
AVR-IoT Wx Hardware User Guide

Preface



Important: This document is applicable for two different products: AVR-IoT WG (AC164160) and AVR-IoT WA (EV15R70A). Both variants are referred to as AVR-IoT Wx in this document, and both products have identical hardware. AVR-IoT WG is preconfigured to send data through Google Cloud IoT Core, and AVR-IoT WA is preconfigured to send data through Amazon Web Services (AWS). Both products can be reconfigured to send data to different cloud providers.



Tip: For the AWS variant of the board (WA), see the [Microchip IoT Developer Guides for AWS](#) for a list of tutorials describing how to connect the board to an AWS account, and how to design in the cloud through various examples.

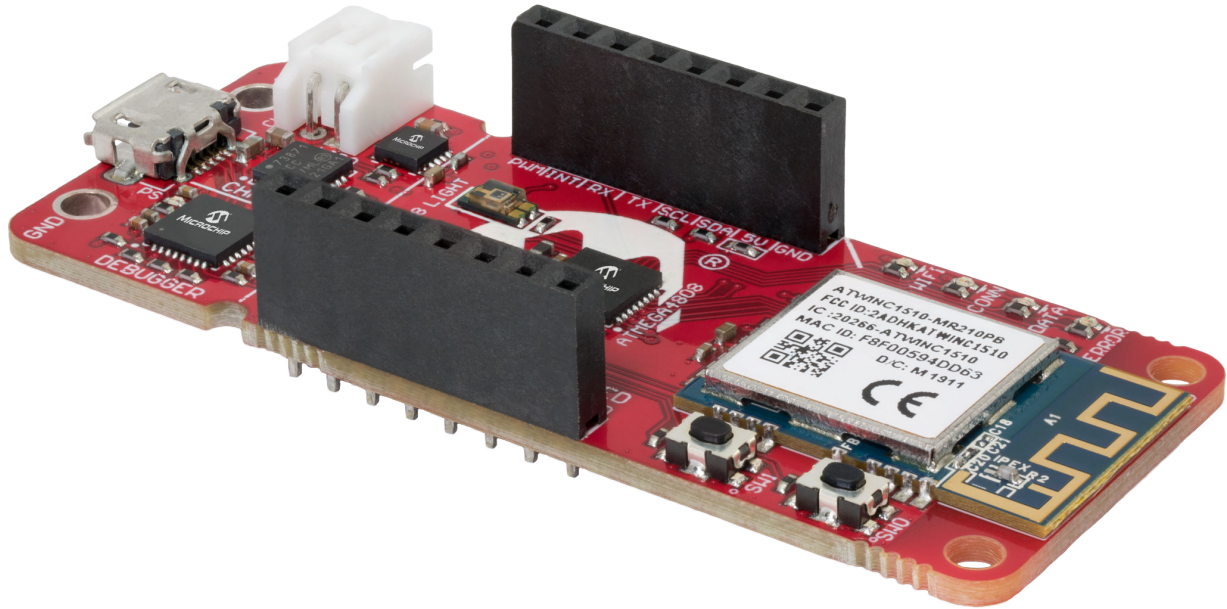
Introduction

The AVR-IoT Wx 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 is 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 [ATWINC1510](#) Wi-Fi controller module

The AVR-IoT Wx Development Board features the following elements:

- The on-board debugger (PKOB nano) supplies full programming and debugging support through Atmel Studio/ MPLAB® X IDE. It also provides access to a serial port interface (serial to USB bridge) and two logic analyzer channels (debug GPIO).
- The on-board debugger enumerates on the PC as a mass storage interface device for easy ‘drag-and-drop’ programming, Wi-Fi configuration, and full access to the microcontroller application Command Line Interface (CLI)
- A mikroBUS™ socket allows for the ability to expand the board capabilities with the selection from 450+ sensors and actuators options offered by MikroElektronika (www.mikroe.com) via a growing portfolio of Click boards™
- A light sensor used to demonstrate published data
- Microchip [MCP9808](#) high-accuracy temperature sensor used to demonstrate published data
- Microchip [MCP73871](#) Li-Ion/LiPo battery charger with power path management



- [MPLAB® X IDE and Atmel Studio](#) - Software to discover, configure, develop, program, and debug Microchip microcontrollers.
- [Application Code on GitHub](#) - Get started with application code.
- [AVR-IoT WG website](#) - Find schematics, design files, and purchase the board. Set up for Google Cloud IoT Core.
- [AVR-IoT WA website](#) - Find schematics, design files, and purchase the board. Set up for Amazon Web Services.

Table of Contents

Preface.....	1
1. Introduction.....	5
1.1. Features.....	5
1.2. Board Overview.....	5
2. Getting Started.....	7
2.1. Quick Start.....	7
2.2. Design Documentation and Relevant Links.....	7
3. Application User Guide.....	8
4. Hardware User Guide.....	9
4.1. On-Board Debugger Overview.....	9
4.2. On-Board Debugger Connections.....	15
4.3. Power.....	15
4.4. Peripherals.....	17
5. Regulatory Approval.....	21
5.1. United States.....	21
5.2. Canada.....	21
5.3. Taiwan.....	22
5.4. List of Antenna Types.....	22
6. Hardware Revision History and Known Issues.....	23
6.1. Identifying Product ID and Revision.....	23
6.2. AVR-IoT WG.....	23
6.3. AVR-IoT WA.....	24
7. Document Revision History.....	25
8. Appendix.....	26
8.1. Schematic.....	26
8.2. Assembly Drawing.....	28
8.3. Mechanical Drawings.....	28
8.4. Getting Started with IAR.....	29
The Microchip Website.....	32
Product Change Notification Service.....	32
Customer Support.....	32
Microchip Devices Code Protection Feature.....	32
Legal Notice.....	32
Trademarks.....	33
Quality Management System.....	33

AVR-IoT Wx Hardware User Guide

Worldwide Sales and Service.....34

1. Introduction

1.1 Features

- [ATmega4808](#) AVR Microcontroller
- [ATWINC1510](#) Wi-Fi Module
- [ATECC608A](#) CryptoAuthentication™ Device
- Preconfigured for Microchip Accounts with Different Cloud Providers
 - Google Cloud IoT Core
 - Amazon Web Services (AWS)
- Four User LEDs
- Two Mechanical Buttons
- TEMENT6000 Light Sensor
- [MCP9808](#) Temperature Sensor
- mikroBUS Socket
- On-board Debugger
 - Board identification in Atmel Studio and Microchip MPLAB X
 - One green board power and status LED
 - Virtual serial port (USB CDC)
 - Two logic analyzer channels (debug GPIO)
- USB and Battery Powered
- [MCP73871](#) Li-Ion/LiPo Battery Charger
- Fixed 3.3V

1.2 Board Overview

The AVR-IoT Wx development board is a hardware platform to evaluate and develop IoT solutions with the Microchip ATmega4808 AVR microcontroller, ATECC608A secure element, and WINC1510 Wi-Fi controller module.

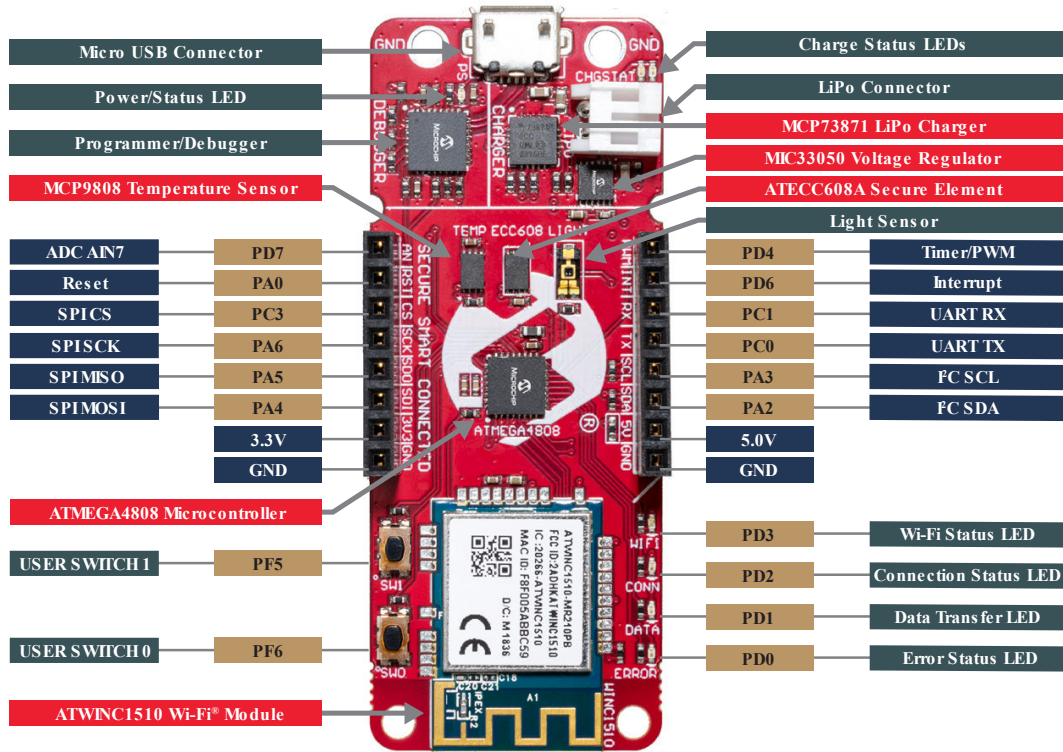
The preprogrammed demo application publishes data from the on-board light and temperature sensor read by the ATmega4808 every second to the cloud. Any data received from the cloud over the subscribed topic is sent to the virtual serial port and can be displayed in a serial terminal application. The WINC1510 needs a connection to a Wi-Fi network with an internet connection. The ATECC608A is used to authenticate the hardware with the cloud to uniquely identify every board. The demo application source code can be modified to publish data to a personal cloud account to get started with a custom cloud application.

The figure below shows the main features and pinout of the board.

AVR-IoT Wx Hardware User Guide

Introduction

Figure 1-1. AVR-IoT Wx Development Board Overview



2. Getting Started

2.1 Quick Start

Steps to start exploring the board:

1. Connect the board to a computer.
2. Open the "CLICK-ME.HTM" file on the "CURIOSITY" mass storage disk and follow the instructions.
 - 2.1. Download the latest application .hex firmware.
 - 2.2. Download the Wi-Fi configuration file "WIFI.cfg".
3. Drag and drop the application .hex file on the "CURIOSITY" drive.
4. Drag and drop the "WIFI.cfg" configuration file on the "CURIOSITY" drive.

The board will now connect to a Wi-Fi network and send data to the website opened in step 2 through a cloud provider.

If AWS is used, the next recommended step is the [Microchip IoT Developer Guides for AWS](#) to connect the board to an AWS account and to design in the cloud through various examples.

2.2 Design Documentation and Relevant Links

The following list contains links to the most relevant documents and software for the AVR-IoT Wx.

- [AVR-IoT WG website](#) - Find schematics, design files, and purchase the board. Set up for Google Cloud IoT Core.
- [AVR-IoT WG on microchipDIRECT](#) - Purchase this board on Microchip Direct.
- [AVR-IoT WA website](#) - Find schematics, design files, and purchase the board. Set up for Amazon Web Services.
- [AVR-IoT WA on microchipDIRECT](#) - Purchase this board on Microchip Direct.
- [Microchip IoT Developer Guides for AWS](#) - Learn IoT Design in AWS.
- [MPLAB Data Visualizer](#) - MPLAB Data Visualizer is a program used for processing and visualizing data. The Data Visualizer can receive data from various sources such as serial ports and on-board debugger's Data Gateway Interface, as found on Curiosity Nano and Xplained Pro boards.
- [Atmel Studio](#) - Free IDE for the development of C/C++ and assembler code for microcontrollers.
- [MPLAB® X IDE](#) - MPLAB X IDE is a software program that runs on a PC (Windows®, Mac OS®, Linux®) to develop applications for Microchip microcontrollers and digital signal controllers. It is called an Integrated Development Environment (IDE) because it provides a single integrated "environment" to develop code for embedded microcontrollers.
- [IAR Embedded Workbench® for AVR®](#) - This is a commercial C/C++ compiler that is available for AVR microcontrollers. There is a 30-day evaluation version as well as a 4 KB code-size-limited kick-start version available from their website.
- [Atmel START](#) - Atmel START is an online tool that hosts code examples, helps the user to select and configure software components, and tailor your embedded application in a usable and optimized manner.
- [Microchip Sample Store](#) - Microchip sample store where you can order samples of devices.

3. Application User Guide

The ATmega4808 mounted on AVR-IoT Wx is preprogrammed with an application ready to publish data to a Microchip account with a cloud provider, and subscribe to data sent from avr-iot.com through the cloud provider. AVR-IoT WA is preconfigured for Amazon Web Services (AWS), and AVR-IoT WG is preconfigured for Google Cloud IoT Core. The data is read from the cloud and presented to the user on avr-iot.com.

AVR-IoT WA

The application publishes data through Amazon Web Services, and the firmware is available on GitHub:

- **MPLAB X:** github.com/microchip-pic-avr-solutions/avr-iot-aws-sensor-node-mplab
- **Atmel Studio:** github.com/microchip-pic-avr-solutions/avr-iot-aws-sensor-node-studio

For a set of tutorials on developing IoT applications on AWS, see [Microchip IoT Developer Guides for AWS](#).

AVR-IoT WG

The application publishes data through Google Cloud IoT Core, and the firmware is available on GitHub: github.com/microchip-pic-avr-solutions/avr-iot-google-sensor-node-mplab.

For in-depth information about the preprogrammed demo application and how to develop your application, see the full AVR-IoT WG Application User Guide: www.microchip.com/mymicrochip/filehandler.aspx?ddocname=en607553.

Setup for Different Cloud Accounts

Any AVR-IoT Wx kit can be reprovisioned to publish data to either Microchips sandbox account at AWS, Microchips sandbox account at Google Cloud IoT Core, or to a personal account.

Download the IoT Provisioning Tool package, compatible with Windows, Mac and Linux to get started: www.microchip.com/mymicrochip/filehandler.aspx?ddocname=en1001525. A video on said process is also available here [Converting PIC & AVR IoT Boards Between AWS and GCP](#).

4. Hardware User Guide

4.1 On-Board Debugger Overview

AVR-IoT Wx contains an on-board debugger for programming and debugging. The on-board debugger is a composite USB device consisting of several interfaces:

- A debugger that can program and debug the ATmega4808 in Atmel Studio/MPLAB X IDE
- A mass storage device that allows drag-and-drop programming of the ATmega4808
- A virtual serial port (CDC) that is connected to a Universal Asynchronous Receiver/Transmitter (UART) on the ATmega4808, and provides an easy way to communicate with the target application through terminal software
- A Data Gateway Interface (DGI) for code instrumentation with logic analyzer channels (debug GPIO) to visualize program flow

The on-board debugger controls a Power and Status LED (marked PS) on the AVR-IoT Wx Board. The table below shows how the LED is controlled in different operation modes.

Table 4-1. On-Board Debugger LED Control

Operation Mode	Power and Status LED
Boot Loader mode	The LED blinks slowly during power-up
Power-up	The LED is ON
Normal operation	The LED is ON
Programming	Activity indicator: The LED blinks slowly during programming/debugging.
Drag-and-drop programming	Success: The LED blinks slowly for 2 sec. Failure: The LED blinks rapidly for 2 sec.
Fault	The LED blinks rapidly if a power fault is detected
Sleep/Off	The LED is OFF. The on-board debugger is either in a sleep mode or powered down. This can occur if the board is externally powered.



Info: Slow blinking is approximately 1 Hz, and rapid blinking is approximately 5 Hz.

4.1.1 Debugger

The on-board debugger on the AVR-IoT Wx Board appears as a Human Interface Device (HID) on the host computer's USB subsystem. The debugger supports full-featured programming and debugging of the ATmega4808 using Atmel Studio/MPLAB X IDE, as well as some third-party IDEs.



Remember: Keep the debugger's firmware up-to-date. Firmware upgrades are done automatically when using Atmel Studio/MPLAB X IDE.

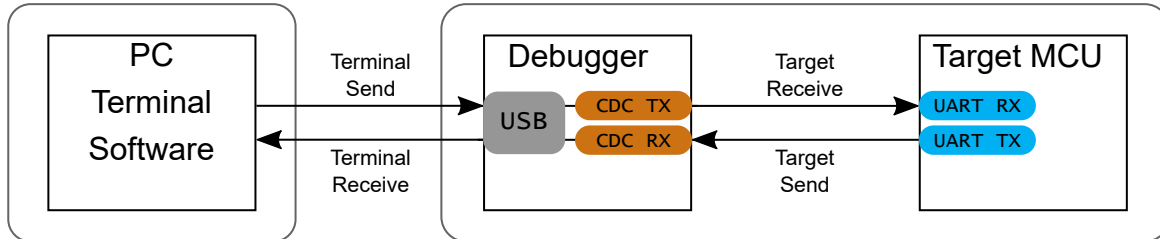
4.1.2 Virtual Serial Port (CDC)

The virtual serial port (CDC) is a general purpose serial bridge between a host PC and a target device.

4.1.2.1 Overview

The on-board debugger implements a composite USB device that includes a standard Communications Device Class (CDC) interface, which appears on the host as a virtual serial port. The CDC can be used to stream arbitrary data in both directions between the host computer and the target: All characters sent through the virtual serial port on the host computer will be transmitted as UART on the debugger's CDC TX pin, and UART characters captured on the debugger's CDC RX pin will be returned to the host computer through the virtual serial port.

Figure 4-1. CDC Connection



Info: As shown in [Figure 4-1](#), the debugger's CDC TX pin is connected to a UART RX pin on the target for receiving characters from the host computer. Similarly, the debugger's CDC RX pin is connected to a UART TX pin on the target for transmitting characters to the host computer.

4.1.2.2 Operating System Support

On Windows machines, the CDC will enumerate as *Curiosity Virtual COM Port* and appear in the Ports section of the Windows Device Manager. The COM port number can also be found there.



Info: On older Windows systems, a USB driver is required for CDC. This driver is included in installations of Atmel Studio/MPLAB X IDE.

On Linux machines, the CDC will enumerate and appear as `/dev/ttyACM#`.



Info: `tty*` devices belong to the "dialout" group in Linux, so it may be necessary to become a member of that group to have permissions to access the CDC.

On MAC machines, the CDC will enumerate and appear as `/dev/tty.usbmodem#`. Depending on which terminal program is used, it will appear in the available list of modems as `usbmodem#`.



Info: For all operating systems: Be sure to use a terminal emulator that supports DTR signaling. See [4.1.2.4 Signaling](#).

4.1.2.3 Limitations

Not all UART features are implemented in the on-board debugger CDC. The constraints are outlined here:

- **Baud rate:** Must be in the range of 1200 bps to 500 kbps. Any baud rate outside this range will be set to the closest limit, without warning. Baud rate can be changed on-the-fly.
- **Character format:** Only 8-bit characters are supported.

- **Parity:** Can be odd, even, or none.
- **Hardware flow control:** Not supported.
- **Stop bits:** One or two bits are supported.

4.1.2.4 Signaling

During USB enumeration, the host OS will start both communication and data pipes of the CDC interface. At this point, it is possible to set and read back the baud rate and other UART parameters of the CDC, but data sending and receiving will not be enabled.

When a terminal connects on the host, it must assert the DTR signal. As this is a virtual control signal implemented on the USB interface, it is not physically present on the board. Asserting the DTR signal from the host will indicate to the on-board debugger that a CDC session is active. The debugger will then enable its level shifters (if available), and start the CDC data send and receive mechanisms.

Deasserting DTR in debugger firmware version 1.20 or earlier has the following behaviour:

- Debugger UART receiver is disabled, so no further data will be transferred to the host computer
- Debugger UART transmitter will continue to send data that is queued for sending, but no new data is accepted from the host computer
- Level shifters (if available) are not disabled, so the debugger CDC TX line remains driven.

Deasserting DTR in debugger firmware version 1.21 or later has the following behaviour:

- Debugger UART receiver is disabled, so no further data will be transferred to the host computer
- Debugger UART transmitter will continue to send data that is queued for sending, but no new data is accepted from the host computer
- Once the ongoing transmission is complete, level shifters (if available) are disabled so the debugger CDC TX line will become high-impedance.



Remember: Set up the terminal emulator to assert the DTR signal. Without the signal, the on-board debugger will not send or receive any data through its UART.



Tip: The on-board debugger's CDC TX pin will not be driven until the CDC interface is enabled by the host computer. Also, there are no external pull-up resistors on the CDC lines connecting the debugger and the target, which means that during power-up, these lines are floating. To avoid any glitches resulting in unpredictable behavior like framing errors, the target device should enable the internal pull-up resistor on the pin connected to the debugger's CDC TX pin.

4.1.2.5 Advanced Use

CDC Override Mode

In normal operation, the on-board debugger is a true UART bridge between the host and the device. However, in certain use cases, the on-board debugger can override the basic operating mode and use the CDC TX and RX pins for other purposes.

Dropping a text file into the on-board debugger's mass storage drive can be used to send characters out of the debugger's CDC TX pin. The filename and extension are trivial, but the text file must start with the characters:

```
CMD:SEND_UART=
```

Debugger firmware version 1.20 or earlier has the following limitations:

- The maximum message length is 50 characters – all remaining data in the frame are ignored.
- The default baud rate used in this mode is 9600 bps, but if the CDC is already active or has been configured, the previously used baud rate still applies.

Debugger firmware version 1.21 and later has the following limitations/features:

- The maximum message length may vary depending on MSC/SCSI layer timeouts on the host computer and/or operating system. A single SCSI frame of 512b (498 characters of payload) is guaranteed, and files of up to 4k will work on most systems. The transfer will complete on the first NULL character encountered in the file.
- The baud rate used is always 9600 bps for the default command

```
CMD:SEND_UART=
```

CDC Override Mode should not be used simultaneously with data transfer over CDC / terminal. If a CDC terminal session is active at the time a file is received via CDC Override Mode, it will be suspended for the duration of the operation and resumed once complete.

- Additional commands are supported with explicit baud rates:

```
CMD:SEND_9600=
```

```
CMD:SEND_115200=
```

```
CMD:SEND_460800=
```

USB-Level Framing Considerations

Sending data from the host to the CDC can be done byte-wise or in blocks, which will be chunked into 64-byte USB frames. Each such frame will be queued up for sending to the debugger's CDC TX pin. Transferring a small amount of data per frame can be inefficient, particularly at low baud rates, because the on-board debugger buffers frames and not bytes. A maximum of four 64-byte frames can be active at any time. The on-board debugger will throttle the incoming frames accordingly. Sending full 64-byte frames containing data is the most efficient method.

When receiving data on the debugger's CDC RX pin, the on-board debugger will queue up the incoming bytes into 64-byte frames, which are sent to the USB queue for transmission to the host when they are full. Incomplete frames are also pushed to the USB queue at approximately 100 ms intervals, triggered by USB start-of-frame tokens. Up to eight 64-byte frames can be active at any time.

If the host (or the software running on it) fails to receive data fast enough, an overrun will occur. When this happens, the last-filled buffer frame will be recycled instead of being sent to the USB queue, and a full frame of data will be lost. To prevent this occurrence, the user must ensure that the CDC data pipe is being read continuously, or the incoming data rate must be reduced.

4.1.3 Mass Storage Device

The on-board debugger includes a simple Mass Storage Device implementation, which is accessible for read/write operations via the host operating system to which it is connected.

It provides:

- Read access to basic text and HTML files for detailed kit information and support
- Write access for programming Intel® HEX formatted files into the target device's memory
- Write access for simple text files for utility purposes

4.1.3.1 Mass Storage Device Implementation

The on-board debugger implements a highly optimized variant of the FAT12 file system that has several limitations, partly due to the nature of FAT12 itself and optimizations made to fulfill its purpose for its embedded application.

The Curiosity Nano USB Device is USB Chapter 9-compliant as a mass storage device but does not, in any way, fulfill the expectations of a general purpose mass storage device. This behavior is intentional.

When using the Windows operating system, the on-board debugger enumerates as a Curiosity Nano USB Device that can be found in the disk drives section of the device manager. The CURIOSITY drive appears in the file manager and claims the next available drive letter in the system.

The CURIOSITY drive contains approximately one MB of free space. This does not reflect the size of the target device's Flash in any way. When programming an Intel® HEX file, the binary data are encoded in ASCII with metadata providing a large overhead, so one MB is a trivially chosen value for disk size.

It is not possible to format the CURIOSITY drive. When programming a file to the target, the filename may appear in the disk directory listing. This is merely the operating system's view of the directory, which, in reality, has not been

updated. It is not possible to read out the file contents. Removing and replugging the board will return the file system to its original state, but the target will still contain the application that has been previously programmed.

To erase the target device, copy a text file starting with “CMD:ERASE” onto the disk.

By default, the CURIOSITY drive contains several read-only files for generating icons as well as reporting status and linking to further information:

- `AUTORUN.ICO` – icon file for the Microchip logo
- `AUTORUN.INF` – system file required for Windows Explorer to show the icon file
- `CLICK-ME.HTM` – redirect to the AVR-IoT Wx web demo application
- `KIT-INFO.HTM` – redirect to the development board website
- `KIT-INFO.TXT` – a text file containing details about the board’s debugger firmware version, board name, USB serial number, device, and drag-and-drop support
- `PUBKEY.TXT` – a text file containing the public key for data encryption
- `STATUS.TXT` – a text file containing the programming status of the board



Info: `STATUS.TXT` is dynamically updated by the on-board debugger. The contents may be cached by the OS and, therefore, do not reflect the correct status.

4.1.3.2 Fuse Bytes

Fuse Bytes (AVR® MCU Targets)

When doing drag-and-drop programming, the debugger masks out fuse bits that attempt to disable Unified Program and Debug Interface (UPDI). This means that the UPDI pin cannot be used in its reset or GPIO modes; selecting one of the alternative functions on the UPDI pin would render the device inaccessible without using an external debugger capable of high-voltage UPDI activation.

4.1.3.3 Limitations of Drag-and-Drop Programming

Lock Bits

Lock bits included in the hex file will be ignored when using drag-and-drop programming. To program lock bits, use Atmel Studio/MPLAB X IDE.

Enabling CRC Check in Fuses

It is not advisable to enable the CRC check in the target device’s fuses when using drag-and-drop programming. This is because a subsequent chip erase (which does not affect fuse bits) will effect a CRC mismatch, and the application will fail to boot. To recover a target from this state, a chip erase must be done using Atmel Studio/MPLAB X IDE, which will automatically clear the CRC fuses after erasing.

4.1.3.4 Special Commands

Several utility commands are supported by copying text files to the mass storage disk. The filename or extension is irrelevant – the command handler reacts to content only.

Table 4-2. Special File Commands

Command Content	Description
<code>CMD:ERASE</code>	Executes a chip erase of the target
<code>CMD:SEND_UART=</code>	Sends a string of characters to the CDC UART. See “ CDC Override Mode ”.

.....continued	
Command Content	Description
CMD:SEND_9600=CMD:SEND_115200=CMD:SEND_460800=	Sends a string of characters to the CDC UART at the baud rate specified. Note that only the baud rates explicitly specified here are supported! See “ CDC Override Mode ” (Debugger firmware v1.21 or newer.)
CMD:RESET	Resets the target device by entering Programming mode and then exiting Programming mode immediately thereafter. Exact timing can vary according to the programming interface of the target device. (Debugger firmware v1.16 or newer.)



Info: The commands listed here are triggered by the content being sent to the mass storage emulated disk, and no feedback is provided in the case of either success or failure.

4.1.4 Data Gateway Interface (DGI)

Data Gateway Interface (DGI) is a USB interface for transporting raw and timestamped data between on-board debuggers and host computer-based visualization tools. [MPLAB Data Visualizer](#) is used on the host computer to display debug GPIO data. It is available as a plug-in for MPLAB® X IDE or a stand-alone application that can be used in parallel with Atmel Studio/MPLAB X IDE.

Although DGI encompasses several physical data interfaces, the AVR-IoT Wx implementation includes logic analyzer channels:

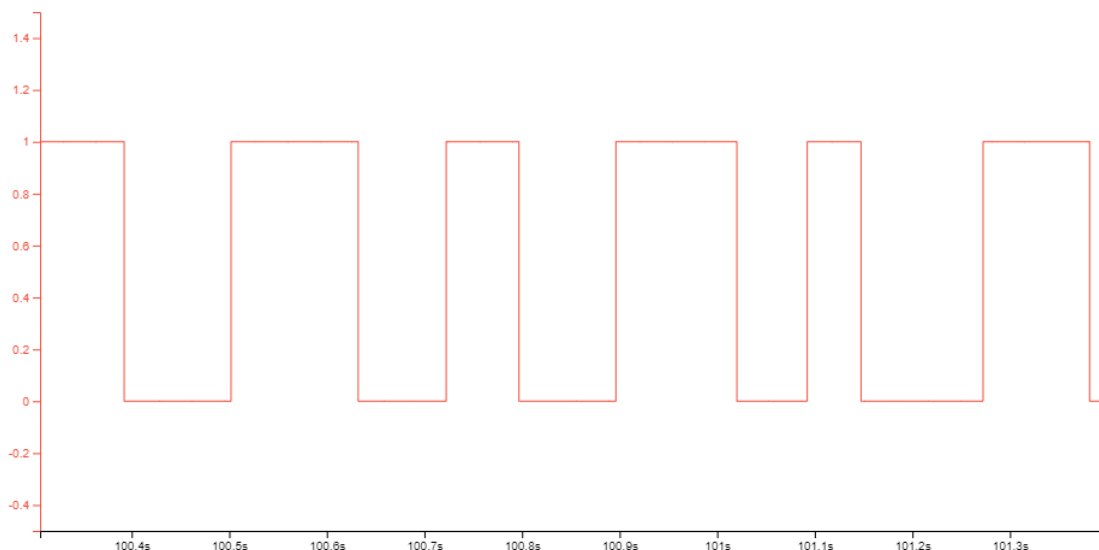
- Two debug GPIO channels (also known as DGI GPIO)

4.1.4.1 Debug GPIO

Debug GPIO channels are timestamped digital signal lines connecting the target application to a host computer visualization application. They are typically used to plot the occurrence of low-frequency events on a time-axis – for example, when certain application state transitions occur.

The figure below shows the monitoring of the digital state of a mechanical switch connected to a debug GPIO in MPLAB Data Visualizer.

Figure 4-2. Monitoring Debug GPIO with MPLAB® Data Visualizer



Debug GPIO channels are timestamped, so the resolution of DGI GPIO events is determined by the resolution of the DGI timestamp module.



Important: Although bursts of higher-frequency signals can be captured, the useful frequency range of signals for which debug GPIO can be used is up to about 2 kHz. Attempting to capture signals above this frequency will result in data saturation and overflow, which may cause the DGI session to be aborted.

4.1.4.2 Timestamping

DGI sources are timestamped as they are captured by the debugger. The timestamp counter implemented in the Curiosity Nano debugger increments at 2 MHz frequency, providing a timestamp resolution of a half microsecond.

4.2 On-Board Debugger Connections

The table below shows the connections between the target and the debugger section. All connections between the target and the debugger are tri-stated as long as the debugger is not actively using the interface. Hence, since there are little contaminations of the signals, the pins can be configured to anything the user wants.

For further information on how to use the capabilities of the on-board debugger, see [4.1 On-Board Debugger Overview](#).

Table 4-3. On-Board Debugger Connections

ATmega4808 Pin	Debugger Pin	Function	Shared Functionality
PF1	CDC TX	UART2 RX (ATmega4808 RX line)	—
PF0	CDC RX	UART2 TX (ATmega4808 TX line)	—
UPDI	DBG0	UPDI	—
PF6	DBG1	DGI GPIO0	SW0
PF5	DBG2	DGI GPIO1	SW1

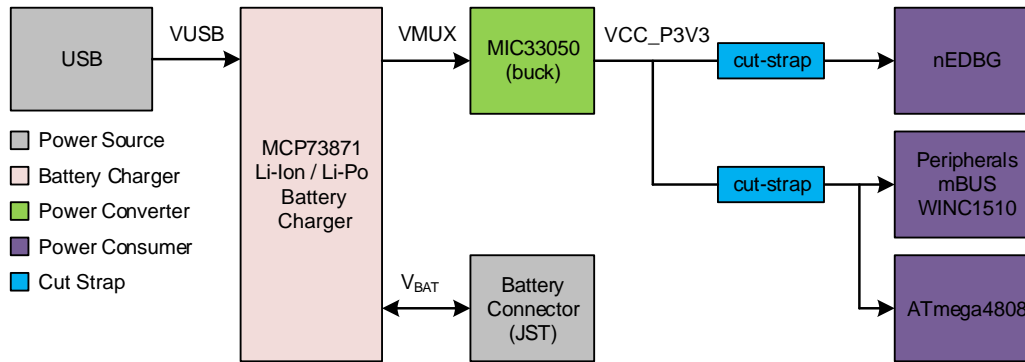
4.3 Power

4.3.1 Power Source

The board can be powered through the USB port or by a Li-Ion/LiPo battery. The board contains one buck converter for generating 3.3V for the debugger, target, and peripherals.

The maximum available current from the USB is limited to 500 mA. The current will be shared between charging the battery (if connected) and the target application section.

Figure 4-3. Power Supply Block Diagram



4.3.2 Battery Charger

AVR-IoT Wx features an MCP73871 Li-Ion/LiPo charger and JST battery connector on board. The charger is configured to limit the charge current to 100 mA to prevent the overcharging of small capacity batteries. Minimum recommended battery capacity is 400 mAh.



WARNING The MCP73871 has a battery charge voltage of 4.2V. Make sure your battery has the same charge voltage.

Table 4-4. Charger Status LEDs

LEDs	Function
Red (charging)	The battery is being charged by the USB
Red (discharging)	The battery voltage is low. Triggers if the voltage is under 3.1V.
Green	Charge complete
Red and Green	Timer Fault. The six-hour charge cycle has timed out before the complete charge.

4.3.3 Hardware Modifications

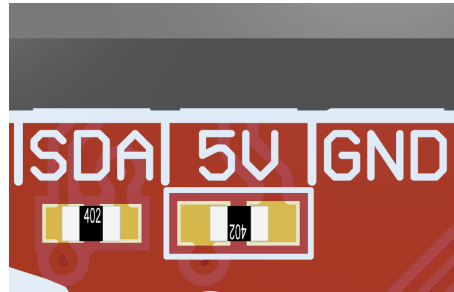
On the bottom side of the AVR-IoT Wx board there are two cut-straps, as shown in the figure below. These are intended for current measurement purposes. Do not leave these unconnected as the microcontrollers might get powered through the I/O's.

Figure 4-4. V_{CC} Cut-Straps



The 5V supply to the mikroBUS socket is connected by default. To remove 5V to the socket, desolder the 0-ohm resistor (0402) below the 5V text, as shown in the figure below.

Figure 4-5. mikroBUS™ 5V Footprint



4.4 Peripherals

4.4.1 ATmega4808

Microchip ATmega4808 is a microcontroller featuring the 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.

4.4.2 mikroBUS™ Socket

AVR-IoT Wx features a mikroBUS socket for expanding the functionality of the development board using MikroElektronika Click boards and other mikroBUS add-on boards. The socket is populated with two 1x8 2.54 mm pitch female headers and is ready to mount add-on boards.

Table 4-5. mikroBUS™ Socket Pinout

mikroBUS™ Socket Pin	ATmega4808 Pin	Function	Shared Functionality
AN	PD7	ADC AIN7	—
RST	PA0	GPIO	—
CS	PC3	GPIO	—
SCK	PA6	SPI0 SCK	WINC1510 SPI
MISO	PA5	SPI0 MISO	WINC1510 SPI
MOSI	PA4	SPI0 MOSI	WINC1510 SPI
+3.3V	V _{DD}	VCC_TARGET, 3.3V supply	—
GND	GND	Ground	—
PWM	PD4	TCA0 WO4	—
INT	PD6	GPIO	—
RX	PC1	UART1 RX	—
TX	PC0	UART1 TX	—
SCL	PA3	TWI0 SCL	MCP9808 and ATECC608A
SDA	PA2	TWI0 SDA	MCP9808 and ATECC608A
+5V	—	VCC_MUX ¹ , MCP73871 output	—

.....continued

mikroBUS™ Socket Pin	ATmega4808 Pin	Function	Shared Functionality
GND	GND	Ground	—



Info:

1) A 0-ohm resistor has been soldered to connect the VCC_MUX pin to the mikroBUS socket. If an add-on module cannot handle 5V on this pin, the 0-ohm resistor has to be removed. For more information, see [4.3.3 Hardware Modifications](#).

4.4.3 WINC1510 Wi-Fi® Module

Microchip's WINC1510 is a low-power consumption 802.11 b/g/n IoT module, specifically optimized for low-power IoT applications. The module integrates 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 (21.7x14.7x2.1 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.

WINC1510 provides internal Flash memory as well as multiple peripheral interfaces, including UART and SPI. The only external clock source needed for WINC1510 is the built-in, high-speed crystal or oscillator (26 MHz). WINC1510 is available in a QFN package or as a certified module.

The communication interface between the ATmega4808 and the WINC1510 Wi-Fi module is SPI, together with some enable signals and interrupt. The rest of the connections are left unconnected.

Table 4-6. WINC1510 Connections

WIN1510 Pin	ATmega4808 Pin	Function	Shared Functionality
4 RESET_N	PA1	GPIO	—
9 GND	GND	Ground	—
10 SPI_CFG	VCC_TARGET	—	—
11 WAKE	PF4	GPIO	—
12 GND	GND	Ground	—
13 IRQN	PF2	ASYNC EXT INT	—
15 SPI_MOSI	PA4	SPI0 MOSI	mikroBUS
16 SPI_SSN	PA7	SPI0 SS	—
17 SPI_MISO	PA5	SPI0 MISO	mikroBUS
18 SPI_SCK	PA6	SPI0 SCK	mikroBUS
20 VBAT	VCC_TARGET	3.3V supply	—
22 CHIP_EN	PF3	GPIO	—
23 VDDIO	VCC_TARGET	3.3V supply	—
28 GND	GND	Ground	—
29 PADDLE	GND	—	—

4.4.4 ATECC608A

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 Internet of Things (IoT) market by easily supplying the full range of security, such as confidentiality, data integrity, and authentication to systems with MCU 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 that eliminate any potential backdoors linked to software weaknesses.

The ATECC608A CryptoAuthentication device on the AVR-IoT Wx board is used to authenticate the hardware with cloud providers to uniquely identify every board.

Note: 7-bit I²C address: 0x58.

Table 4-7. ATECC608A Connections

ATECC608A Pin	ATmega4808 Pin	Function	Shared Functionality
SDA	PA2	TWI0 SDA	MCP9808 and mikroBUS
SCL	PA3	TWI0 SCL	MCP9808 and mikroBUS

4.4.5 Temperature Sensor

The MCP9808 digital temperature sensor converts temperatures between -20°C and +100°C to a digital word with $\pm 0.25^\circ\text{C}/\pm 0.5^\circ\text{C}$ (typical/maximum) accuracy.

Additional features:

- Accuracy:
 - $\pm 0.25^\circ\text{C}$ (typical) from -40°C to $+125^\circ\text{C}$
 - $\pm 0.5^\circ\text{C}$ (maximum) from -20°C to $+100^\circ\text{C}$
- User Selectable Measurement Resolution:
 - 0.5°C , 0.25°C , 0.125°C , 0.0625°C
- User Programmable Temperature Limits:
 - Temperature Window Limit
 - 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)

The MCP9808 temperature sensor is connected to the ATmega4808 through I²C and a GPIO for the user-configurable alert output.

Note: 7-bit I²C address: 0x18.

Table 4-8. MCP9808

MCP9808 Pin	ATmega4808 Pin	Function	Shared Functionality
SDA	PA2	TWI0 SDA	ATECC608A and mikroBUS
SCL	PA3	TWI0 SCL	ATECC608A and mikroBUS
Alert	PC2	ASYNC External Interrupt	—

4.4.6 Light Sensor

A TEMT6000X01 light sensor is mounted on the AVR-IoT Wx board for measuring the light intensity. The sensor is a current source that will induce a voltage across the series resistor, which in turn can be measured by the ATmega4808 ADC. The current is exponentially relative to illuminance, from about 10 $\mu\text{A}@20\text{lx}$ to 50 $\mu\text{A}@100\text{lx}$. The series resistor has a value of 10 k Ω .

Table 4-9. Light Sensor Connection

ATmega4808 Pin	Function	Shared Functionality
PD5	ADC AIN5	—

4.4.7 LED

There are four LEDs available on the AVR-IoT Wx board that can be controlled with PWM or GPIO. The LEDs can be activated by driving the connected I/O line to GND.

Table 4-10. LED Connections

ATmega4808 Pin	Function	Description
PD0	TCA0 WO0	Red LED
PD1	TCA0 WO1	Yellow LED
PD2	TCA0 WO2	Green LED
PD3	TCA0 WO3	Blue LED

4.4.8 Mechanical Buttons

AVR-IoT Wx contains two mechanical buttons. These are generic user-configurable buttons. When a button is pressed, it will drive the connected I/O line to ground (GND).



Info: There are no pull-up resistors connected to the generic user buttons. Remember to enable the internal pull-up in the ATmega4808 to use the buttons.

Table 4-11. Mechanical Button

ATmega4808 Pin	Description	Shared Functionality
PF6	User switch 0 (SW0)	On-board debugger
PF5	User switch 1 (SW1)	On-board debugger

5. Regulatory Approval

The AVR-IoT Wx development board has been tested by the following standards:

Emission:

- FCC Part 15 subpart B:2018 (Class B)
- EN55032:2015 (Class B)

Immunity:

- EN55024:2010+A1:2015
- EN61000-4-2:2009 (contact: level 2 (± 4 kV), air: level3 (± 8 kV))
- EN61000-4-3:2006+A2:2010 (80 - 1000 MHz, level 2 (3 V/M))
- EN61000-4-8:2010 (level 2 (3 A/m), continuous field)

The development board contains the wireless transmitter module ATWINC1510-MR210PB, which has the following approval and/or registrations:

- United States/FCC ID: 2ADHKATWINC1510
- Canada
 - IC: 20266-ATWINC1510
 - HVIN: ATWINC1510-MR210PB
 - PMN: ATWINC1510-MR210PB
- Europe – CE
- Japan/MIC: 005-101762
- Korea/KCC: R-CRM-mcp-WINC1510MR210P
- Taiwan/NCC: CCAN18LP0320T0
- China/SRRC: CMIIT ID: 2018DJ1310

5.1 United States

Contains Transmitter Module FCC ID: 2ADHKATWINC1510.

This equipment has been tested and found to comply with the limits for a Class B digital device, pursuant to part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference in a residential installation. This equipment generates, uses, and can radiate radio frequency energy, and if not installed and used in accordance with the instructions, may cause harmful interference to radio communications. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause harmful interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient or relocate the receiving antenna
- Increase the separation between the equipment and receiver
- Connect the equipment into an outlet on a circuit different from that to which the receiver is connected
- Consult the dealer or an experienced radio/TV technician for help

5.2 Canada

Contains IC: 20266-ATWINC1510.

This device complies with Industry Canada's license-exempt RSS standard(s). Operation is subject to the following two conditions:

- (1) This device may not cause interference, and
- (2) This device must accept any interference, including interference that may cause undesired operation of the device.

Le présent appareil est conforme aux CNR d'Industrie Canada applicables aux appareils radio exempts de licence. L'exploitation est autorisée aux deux conditions suivantes:

- (1) l'appareil ne doit pas produire de brouillage, et
- (2) l'utilisateur de l'appareil doit accepter tout brouillage radioélectrique subi, même si le brouillage est susceptible d'en compromettre le fonctionnement.

Guidelines on Transmitter Antenna for License Exempt Radio Apparatus:

Under Industry Canada regulations, this radio transmitter may only operate using an antenna of a type and maximum (or lesser) gain approved for the transmitter by Industry Canada. To reduce potential radio interference to other users, the antenna type and its gain should be so chosen that the equivalent isotropically radiated power (e.i.r.p.) is not more than that necessary for successful communication.

Conformément à la réglementation d'Industrie Canada, le présent émetteur radio peut fonctionner avec une antenne d'un type et d'un gain maximal (ou inférieur) approuvé pour l'émetteur par Industrie Canada. Dans le but de réduire les risques de brouillage radioélectrique à l'intention des autres utilisateurs, il faut choisir le type d'antenne et son gain de sorte que la puissance isotrope rayonnée équivalente (p.i.r.e.) ne dépasse pas l'intensité nécessaire à l'établissement d'une communication satisfaisante.

5.3 Taiwan

Contains module: CCAN18LP0320T0.

注意！

依據 低功率電波輻射性電機管理辦法

第十二條 經型式認證合格之低功率射頻電機，非經許可，公司、商號或使用者均不得擅自變更頻率、加大功率或變更原設計之特性及功能。

第十四條 低功率射頻電機之使用不得影響飛航安全及干擾合法通信；經發現有干擾現象時，應立即停用，並改善至無干擾時方得繼續使用。

前項合法通信，指依電信規定作業之無線電信。

低功率射頻電機須忍受合法通信或工業、科學及醫療用電波輻射性電機設備之干擾。

5.4 List of Antenna Types

ATWINC1510-MR210 does not allow the use of external antennas and is tested with the PCB antenna on the module.

6. Hardware Revision History and Known Issues

This user guide is written to provide information about the latest available revision of the board. The following sections contain information about known issues, a revision history of older revisions, and how older revisions differ from the latest revision.

6.1 Identifying Product ID and Revision

The revision and product identifier of the AVR-IoT Wx boards can be found in two ways: Either by utilizing the Atmel Studio/MPLAB X IDE Kit Window or by looking at the sticker on the bottom side of the PCB.

By connecting an AVR-IoT Wx to a computer with Atmel Studio/MPLAB X IDE running, the Kit Window will pop up. The first six digits of the serial number, which is listed under kit information, contain the product identifier and revision.



Tip: The Kit Window can be opened in MPLAB® X IDE through the menu bar *Window > Kit Window*.

The same information can be found on the sticker on the bottom side of the PCB. Most boards will have the identifier and revision printed in plain text as A09-nnnn\rr, where “nnnn” is the identifier, and “rr” is the revision. Boards with limited space have a sticker with only a data matrix code, containing the product identifier, revision, and serial number.

The serial number string has the following format:

```
"nnnnrrssssssss"  
n = product identifier  
r = revision  
s = serial number
```

The product identifier for AVR-IoT WG is A09-3203.

The product identifier for AVR-IoT WA is A09-3349.

6.2 AVR-IoT WG

6.2.1 Revision 9

New PCB revision 5 with minor silkscreen changes.

6.2.2 Revision 8

The Wi-Fi module WINC1510 order code used on revision 8 is ATWINC1510-MR210PB1961 (firmware 19.6.1).

J201 and J202 (mikroBUS socket), and R204 (0-ohm resistor to apply 5V to mikroBUS socket) are not populated on this revision.

PCB revision 4.

6.2.3 Revision 7

Revision 7 is the initial revision available on microchipDIRECT.

The Wi-Fi module WINC1510 order code used on revision 7 is ATWINC1510-MR210PB1952 (firmware 19.5.2). The firmware was upgraded to version 19.6.1 in production.

J201 and J202 (mikroBUS socket), and R204 (0-ohm resistor to apply 5V to mikroBUS socket) are not populated on this revision.

AVR-IoT Wx Hardware User Guide

Hardware Revision History and Known Issues

New PCB revision 4, fixing issues with the MCP73871 Li-Ion/LiPo charger connections.

6.2.4 Revision 6

Revision 6 is the early adopter revision. It does not have the MCP73871 Li-Ion/LiPo charger or battery connector components mounted, and can only be powered through USB.

J201 and J202 (mikroBUS socket), and R204 (0-ohm resistor to apply 5V to mikroBUS socket) are not populated on this revision.

The Wi-Fi module WINC1510 order code used on revision 6 is ATWINC1510-MR210PB1952 (firmware 19.5.2). The firmware was upgraded to version 19.6.1 in production.

PCB revision 3.

Related Links

[4.3.2 Battery Charger](#)

6.3 AVR-IoT WA

6.3.1 Revision 1

Revision 1 of AVR-IoT WA is the initial released revision. The hardware is identical to AVR-IoT WG revision 9.

Drag-and-drop configuration of Wi-Fi credentials does not work with the application firmware preprogrammed on this revision of the kit. Follow the instructions in [2.1 Quick Start](#) to download and upgrade the application firmware.

PCB revision 5.

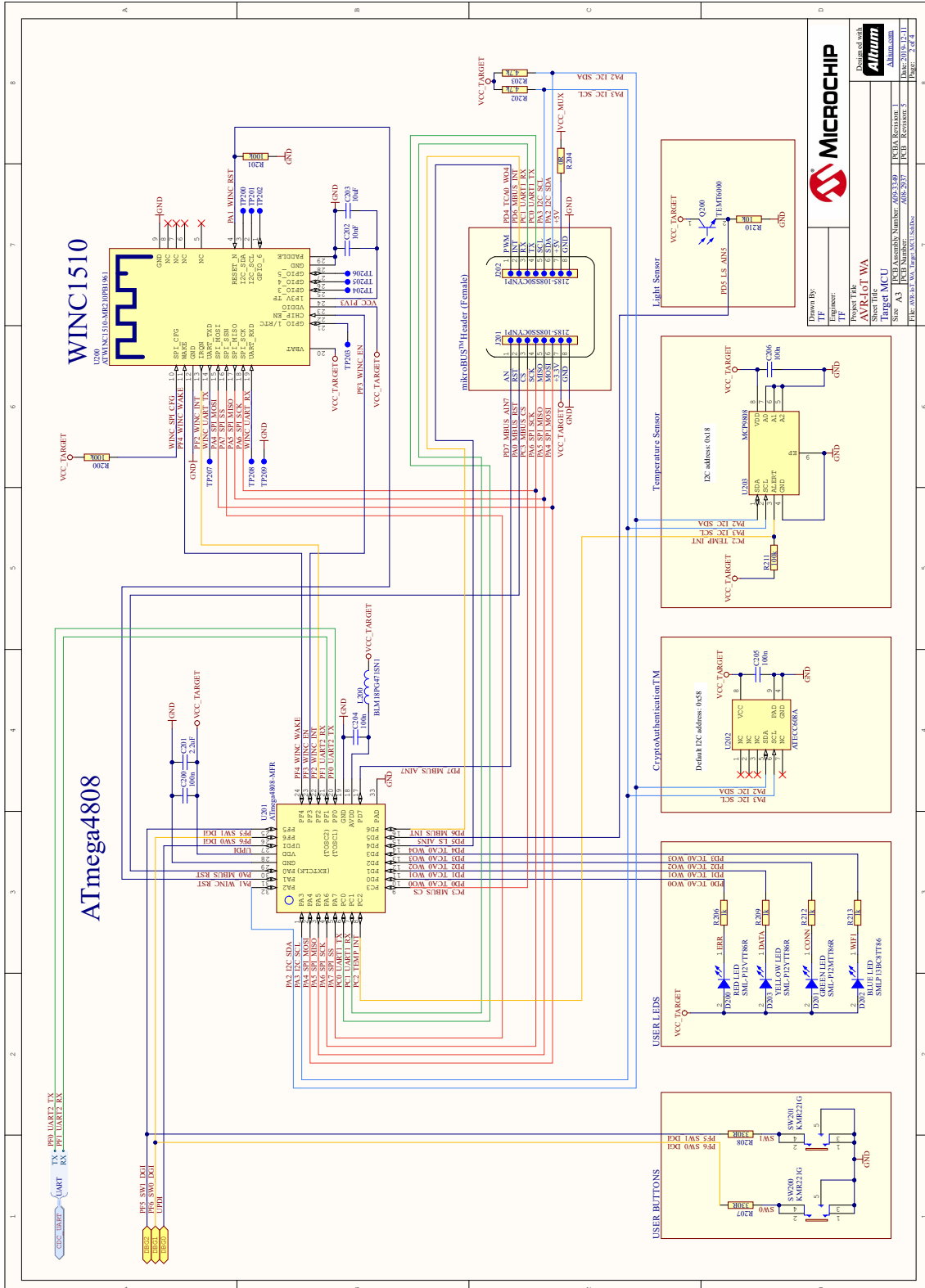
7. Document Revision History

Doc. rev.	Date	Comment
C	07/2020	Added references to the Microchip IoT Developer Guides.
B	03/2020	Updated with AVR-IoT WG revision 9, updated with information about AVR-IoT WA.
A	10/2018	Initial document release.

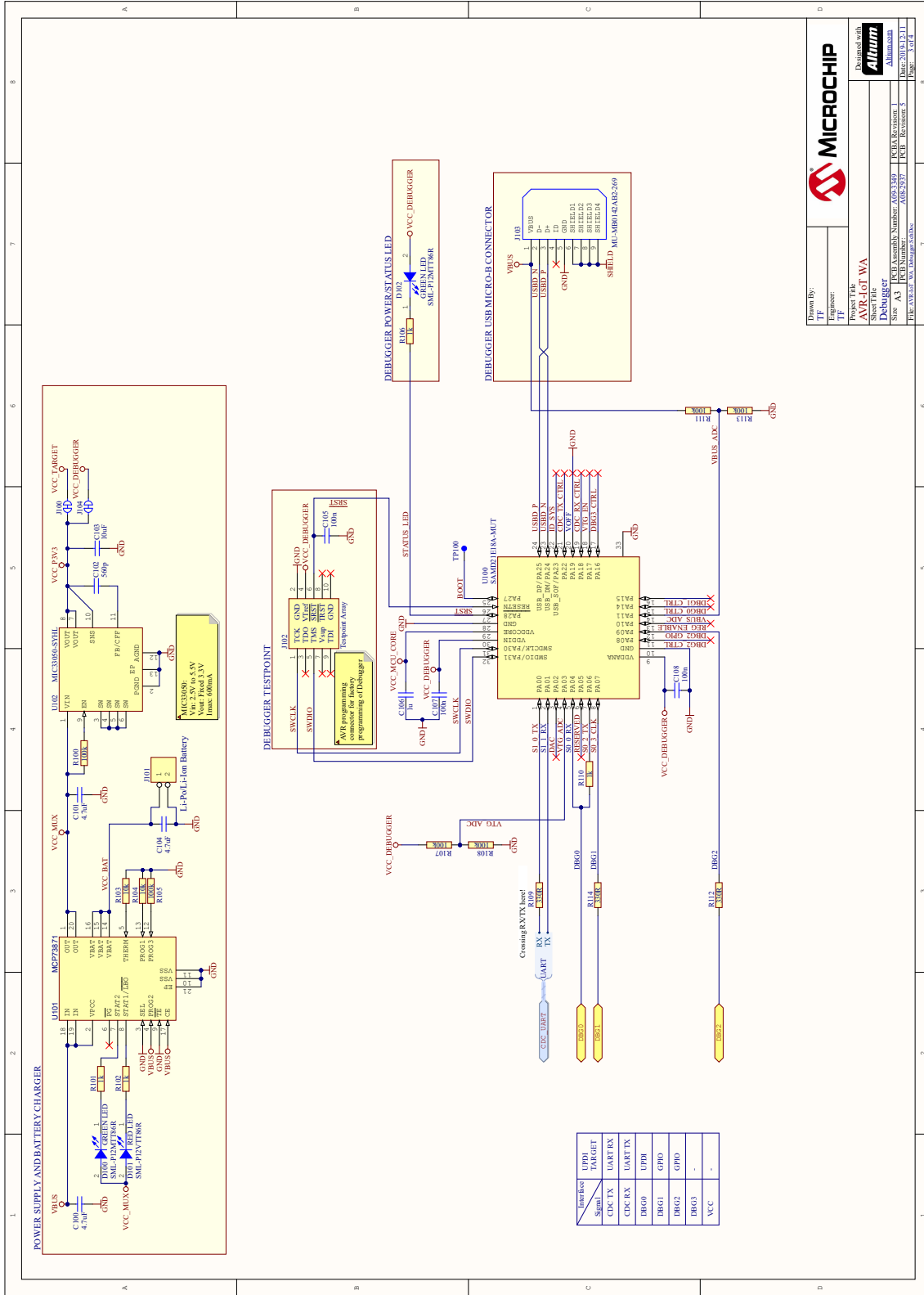
8. Appendix

8.1 Schematic

Figure 8-1. AVR-IoT Wx Schematic



MICROCHIP
 Design of tools
Altium
 Project File
AVR-IoT WA
 Sheet Path: AVR-IoT WA
AVR-IoT MCU
 Size: A3 | PCB Assembly Number: 40523349 | PCB Revision: 1
 Date: 2019-12-11
 PCB Number: 405237 | PCB Revision: 3
 File: AVR-IoT Wx Target.ucb (Auto) | Page: 2 of 4



Drawn By: **Microchip**
 File Name: **AVR-IoT WA**
 Project Title: **AVR-IoT WA**
 Schematic File: **AVR-IoT WA**
 Date: **2020/02/12**
 Size: **A3** | PCB Assembly Number: **AVR-IoT WA** | PCB Revision: **1**
 Part Number: **AVR-IoT WA** | PCB Revision: **1**
 File Location: **AVR-IoT WA**

MICROCHIP
 Divided by
Atmel
 Project Title: **AVR-IoT WA**
 Schematic File: **AVR-IoT WA**
 Date: **2020/02/12**
 Size: **A3** | PCB Assembly Number: **AVR-IoT WA** | PCB Revision: **1**
 Part Number: **AVR-IoT WA** | PCB Revision: **1**
 File Location: **AVR-IoT WA**

8.2 Assembly Drawing

Figure 8-2. AVR-IoT Wx Assembly Drawing Top

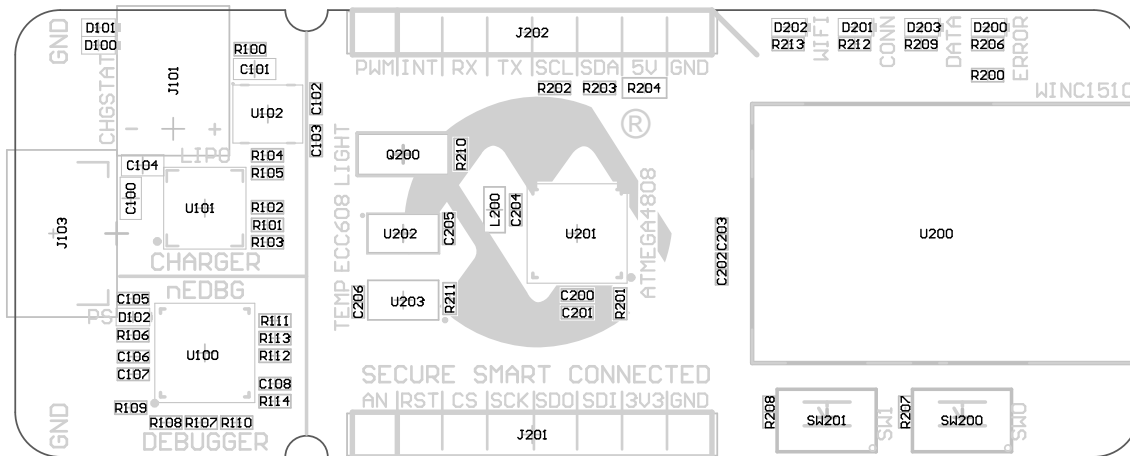
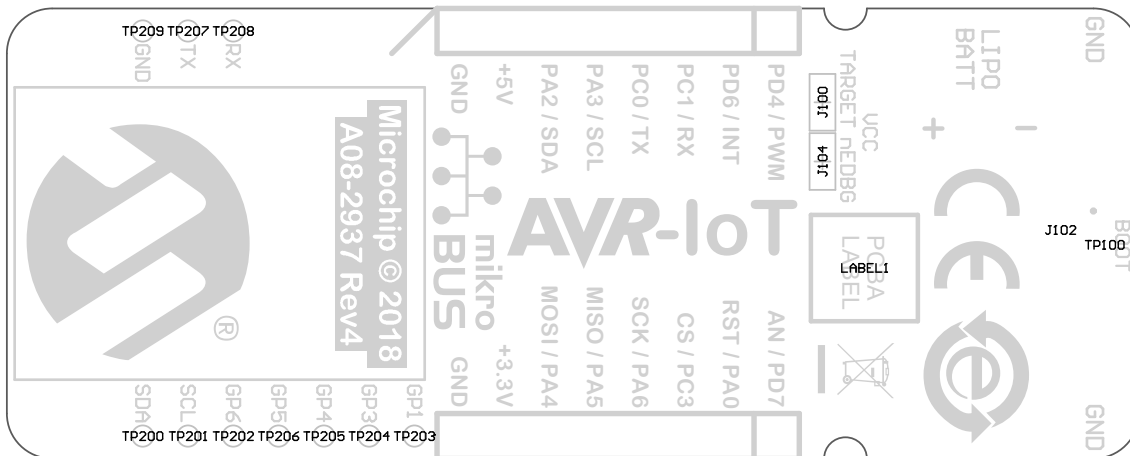


Figure 8-3. AVR-IoT Wx Assembly Drawing Bottom



8.3 Mechanical Drawings

The figures below show the board's mechanical drawing and connector placement.

Figure 8-4. Mechanical Drawing

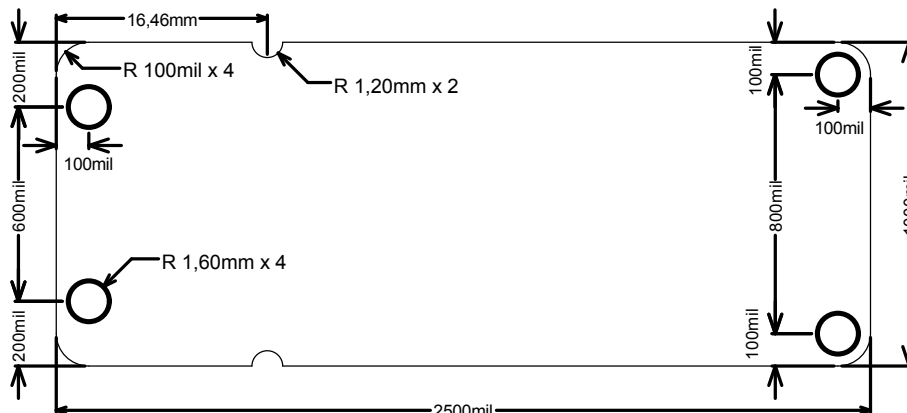
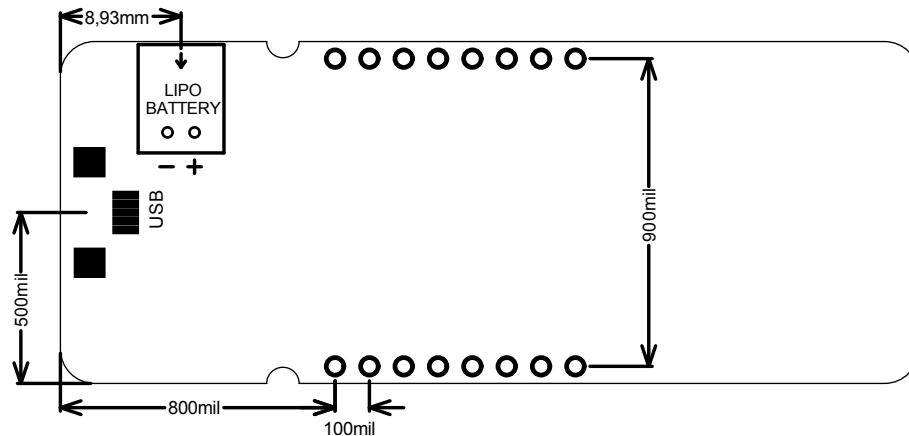


Figure 8-5. Connector Placement



8.4 Getting Started with IAR

IAR Embedded Workbench® for AVR® is a proprietary high-efficiency compiler not based on GCC. Programming and debugging of AVR-IoT Wx is supported in IAR™ Embedded Workbench for AVR using the Atmel-ICE interface. Some initial settings must be set up in the project to get the programming and debugging to work.

The following steps will explain how to get your project ready for programming and debugging:

1. Make sure you have opened the project you want to configure. Open the **OPTIONS** dialog for the project.
2. In the category **General Options**, select the **Target** tab. Select the device for the project, or if not listed, the core of the device, as shown in [Figure 8-6](#).
3. In the category **Debugger**, select the **Setup** tab. Select **Atmel-ICE** as the driver, as shown in [Figure 8-7](#).
4. In the category **Debugger > Atmel-ICE**, select the **Atmel-ICE 1** tab. Select **UPDI** as the interface and, optionally, select the **UPDI** frequency, as shown in [Figure 8-8](#).



Info: If the selection of Debug Port (mentioned in step 4) is grayed out, the interface is preselected, and the user can skip this configuration step.

Figure 8-6. Select Target Device

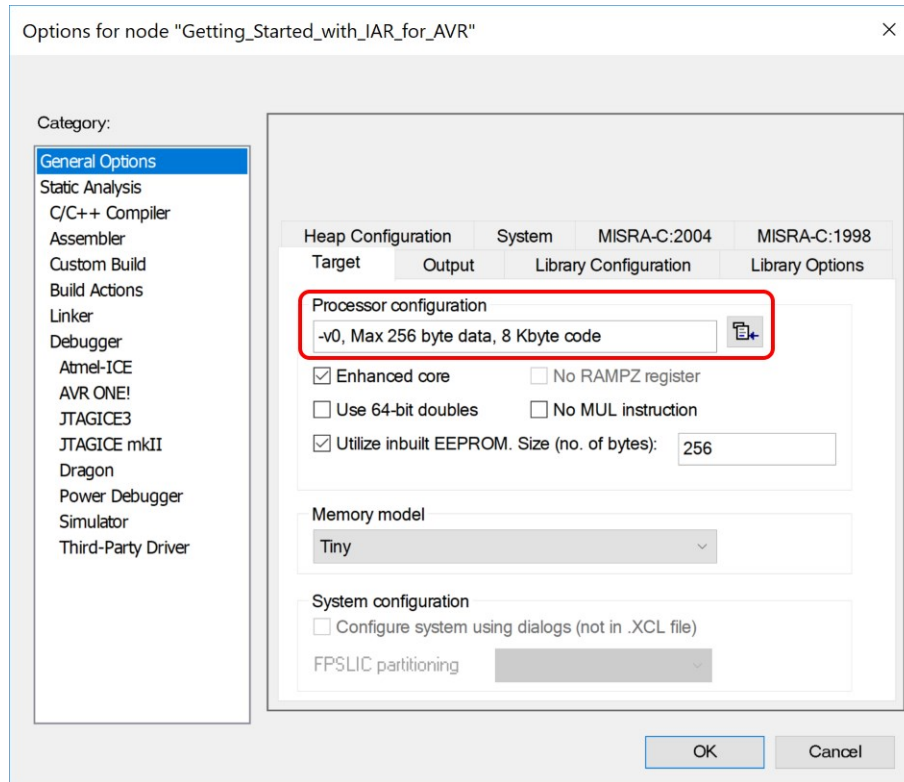


Figure 8-7. Select Debugger

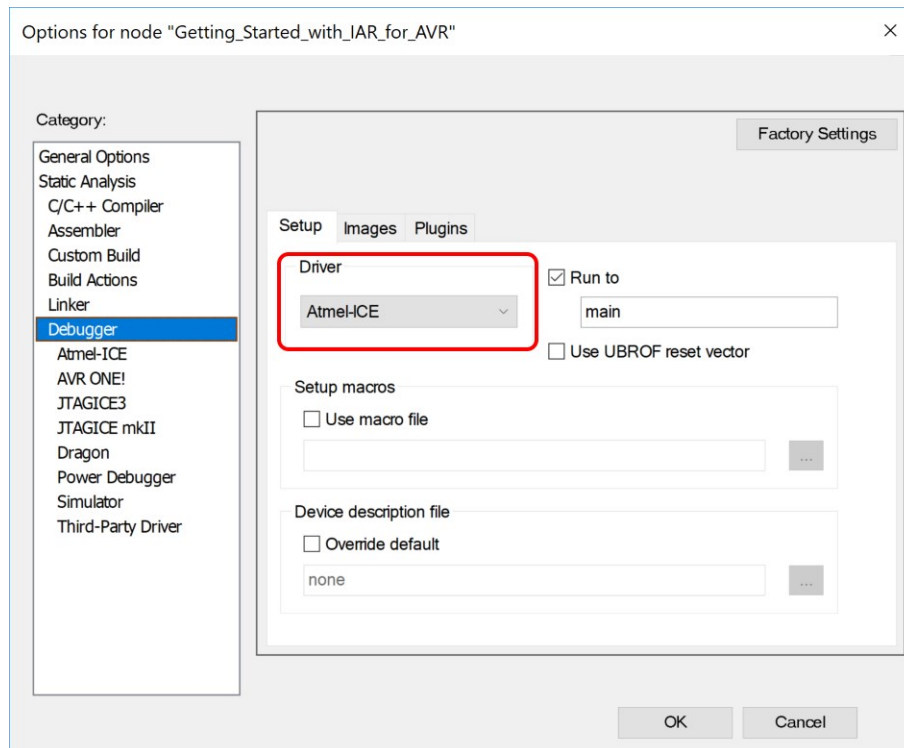
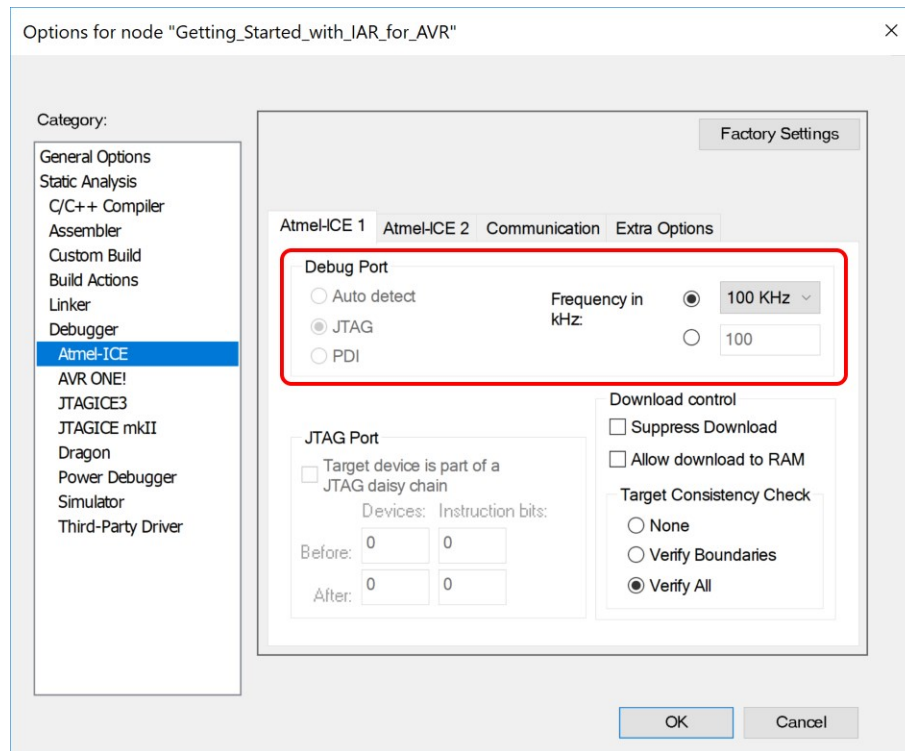


Figure 8-8. Configure Interface



The Microchip Website

Microchip provides online support via our website at www.microchip.com/. This website is used to make files and information easily available to customers. Some of the content available includes:

- **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 (FAQs), technical support requests, online discussion groups, Microchip design partner 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

Product Change Notification Service

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

To register, go to www.microchip.com/pcn and follow the registration instructions.

Customer Support

Users of Microchip products can receive assistance through several channels:

- Distributor or Representative
- Local Sales Office
- Embedded Solutions Engineer (ESE)
- Technical Support

Customers should contact their distributor, representative or ESE for support. Local sales offices are also available to help customers. A listing of sales offices and locations is included in this document.

Technical support is available through the website at: www.microchip.com/support

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, Adaptec, AnyRate, AVR, AVR logo, AVR Freaks, BesTime, BitCloud, chipKIT, chipKIT logo, CryptoMemory, CryptoRF, dsPIC, FlashFlex, flexPWR, HELDO, IGLOO, JukeBlox, KeeLoq, Kleer, LANCheck, LinkMD, maXStylus, maXTouch, MediaLB, megaAVR, Microsemi, Microsemi logo, MOST, MOST logo, MPLAB, OptoLyzer, PackeTime, PIC, picoPower, PICSTART, PIC32 logo, PolarFire, Prochip Designer, QTouch, SAM-BA, SenGenuity, SpyNIC, SST, SST Logo, SuperFlash, Symmetricom, SyncServer, Tachyon, TempTrackr, TimeSource, tinyAVR, UNI/O, Vectron, and XMEGA are registered trademarks of Microchip Technology Incorporated in the U.S.A. and other countries.

APT, ClockWorks, The Embedded Control Solutions Company, EtherSynch, FlashTec, Hyper Speed Control, HyperLight Load, IntelliMOS, Libero, motorBench, mTouch, Powermite 3, Precision Edge, ProASIC, ProASIC Plus, ProASIC Plus logo, Quiet-Wire, SmartFusion, SyncWorld, Temux, TimeCesium, TimeHub, TimePictra, TimeProvider, Vite, WinPath, and ZL 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, BlueSky, 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, 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.

The Adaptec logo, Frequency on Demand, Silicon Storage Technology, and Symmcom are registered trademarks 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.

All other trademarks mentioned herein are property of their respective companies.

© 2020, Microchip Technology Incorporated, Printed in the U.S.A., All Rights Reserved.

ISBN: 978-1-5224-6372-6

Quality Management System

For information regarding Microchip's Quality Management Systems, please visit www.microchip.com/quality.

Worldwide Sales and Service

AMERICAS	ASIA/PACIFIC	ASIA/PACIFIC	EUROPE
<p>Corporate Office 2355 West Chandler Blvd. Chandler, AZ 85224-6199 Tel: 480-792-7200 Tel: 480-792-7277 Technical Support: www.microchip.com/support Web Address: www.microchip.com</p> <p>Atlanta Duluth, GA Tel: 678-957-9614 Fax: 678-957-1455</p> <p>Austin, TX Tel: 512-257-3370</p> <p>Boston Westborough, MA Tel: 774-760-0087 Fax: 774-760-0088</p> <p>Chicago Itasca, IL Tel: 630-285-0071 Fax: 630-285-0075</p> <p>Dallas Addison, TX Tel: 972-818-7423 Fax: 972-818-2924</p> <p>Detroit Novi, MI Tel: 248-848-4000</p> <p>Houston, TX Tel: 281-894-5983</p> <p>Indianapolis Noblesville, IN Tel: 317-773-8323 Fax: 317-773-5453 Tel: 317-536-2380</p> <p>Los Angeles Mission Viejo, CA Tel: 949-462-9523 Fax: 949-462-9608 Tel: 951-273-7800</p> <p>Raleigh, NC Tel: 919-844-7510</p> <p>New York, NY Tel: 631-435-6000</p> <p>San Jose, CA Tel: 408-735-9110 Tel: 408-436-4270</p> <p>Canada - Toronto Tel: 905-695-1980 Fax: 905-695-2078</p>	<p>Australia - Sydney Tel: 61-2-9868-6733</p> <p>China - Beijing Tel: 86-10-8569-7000</p> <p>China - Chengdu Tel: 86-28-8665-5511</p> <p>China - Chongqing Tel: 86-23-8980-9588</p> <p>China - Dongguan Tel: 86-769-8702-9880</p> <p>China - Guangzhou Tel: 86-20-8755-8029</p> <p>China - Hangzhou Tel: 86-571-8792-8115</p> <p>China - Hong Kong SAR Tel: 852-2943-5100</p> <p>China - Nanjing Tel: 86-25-8473-2460</p> <p>China - Qingdao Tel: 86-532-8502-7355</p> <p>China - Shanghai Tel: 86-21-3326-8000</p> <p>China - Shenyang Tel: 86-24-2334-2829</p> <p>China - Shenzhen Tel: 86-755-8864-2200</p> <p>China - Suzhou Tel: 86-186-6233-1526</p> <p>China - Wuhan Tel: 86-27-5980-5300</p> <p>China - Xian Tel: 86-29-8833-7252</p> <p>China - Xiamen Tel: 86-592-2388138</p> <p>China - Zhuhai Tel: 86-756-3210040</p>	<p>India - Bangalore Tel: 91-80-3090-4444</p> <p>India - New Delhi Tel: 91-11-4160-8631</p> <p>India - Pune Tel: 91-20-4121-0141</p> <p>Japan - Osaka Tel: 81-6-6152-7160</p> <p>Japan - Tokyo Tel: 81-3-6880-3770</p> <p>Korea - Daegu Tel: 82-53-744-4301</p> <p>Korea - Seoul Tel: 82-2-554-7200</p> <p>Malaysia - Kuala Lumpur Tel: 60-3-7651-7906</p> <p>Malaysia - Penang Tel: 60-4-227-8870</p> <p>Philippines - Manila Tel: 63-2-634-9065</p> <p>Singapore Tel: 65-6334-8870</p> <p>Taiwan - Hsin Chu Tel: 886-3-577-8366</p> <p>Taiwan - Kaohsiung Tel: 886-7-213-7830</p> <p>Taiwan - Taipei Tel: 886-2-2508-8600</p> <p>Thailand - Bangkok Tel: 66-2-694-1351</p> <p>Vietnam - Ho Chi Minh Tel: 84-28-5448-2100</p>	<p>Austria - Wels Tel: 43-7242-2244-39 Fax: 43-7242-2244-393</p> <p>Denmark - Copenhagen Tel: 45-4485-5910 Fax: 45-4485-2829</p> <p>Finland - Espoo Tel: 358-9-4520-820</p> <p>France - Paris Tel: 33-1-69-53-63-20 Fax: 33-1-69-30-90-79</p> <p>Germany - Garching Tel: 49-8931-9700</p> <p>Germany - Haan Tel: 49-2129-3766400</p> <p>Germany - Heilbronn Tel: 49-7131-72400</p> <p>Germany - Karlsruhe Tel: 49-721-625370</p> <p>Germany - Munich Tel: 49-89-627-144-0 Fax: 49-89-627-144-44</p> <p>Germany - Rosenheim Tel: 49-8031-354-560</p> <p>Israel - Ra'anana Tel: 972-9-744-7705</p> <p>Italy - Milan Tel: 39-0331-742611 Fax: 39-0331-466781</p> <p>Italy - Padova Tel: 39-049-7625286</p> <p>Netherlands - Drunen Tel: 31-416-690399 Fax: 31-416-690340</p> <p>Norway - Trondheim Tel: 47-72884388</p> <p>Poland - Warsaw Tel: 48-22-3325737</p> <p>Romania - Bucharest Tel: 40-21-407-87-50</p> <p>Spain - Madrid Tel: 34-91-708-08-90 Fax: 34-91-708-08-91</p> <p>Sweden - Gothenberg Tel: 46-31-704-60-40</p> <p>Sweden - Stockholm Tel: 46-8-5090-4654</p> <p>UK - Wokingham Tel: 44-118-921-5800 Fax: 44-118-921-5820</p>