

# ON Semiconductor

## Is Now

# onsemi™

To learn more about onsemi™, please visit our website at  
[www.onsemi.com](http://www.onsemi.com)

**onsemi** and **onsemi** and other names, marks, and brands are registered and/or common law trademarks of Semiconductor Components Industries, LLC dba "**onsemi**" or its affiliates and/or subsidiaries in the United States and/or other countries. **onsemi** owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of **onsemi** product/patent coverage may be accessed at [www.onsemi.com/site/pdf/Patent-Marking.pdf](http://www.onsemi.com/site/pdf/Patent-Marking.pdf). **onsemi** reserves the right to make changes at any time to any products or information herein, without notice. The information herein is provided "as-is" and **onsemi** makes no warranty, representation or guarantee regarding the accuracy of the information, product features, availability, functionality, or suitability of its products for any particular purpose, nor does **onsemi** assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. Buyer is responsible for its products and applications using **onsemi** products, including compliance with all laws, regulations and safety requirements or standards, regardless of any support or applications information provided by **onsemi**. "Typical" parameters which may be provided in **onsemi** data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. **onsemi** does not convey any license under any of its intellectual property rights nor the rights of others. **onsemi** products are not designed, intended, or authorized for use as a critical component in life support systems or any FDA Class 3 medical devices or medical devices with a same or similar classification in a foreign jurisdiction or any devices intended for implantation in the human body. Should Buyer purchase or use **onsemi** products for any such unintended or unauthorized application, Buyer shall indemnify and hold **onsemi** and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that **onsemi** was negligent regarding the design or manufacture of the part. **onsemi** is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner. Other names and brands may be claimed as the property of others.

# DVK-2 Quick Start Guide

## AND9698/D

### DVK-2 Introduction

#### DVK-2 Base-kit

The DVK-2 is designed as quick start solution for the development of applications with AX micro-controllers and radio-chips. The DVK-2 base-kit is the development platform for applications based on the AX8052F1xx micro-controller family. With the available radio add-on-kits it becomes the ideal platform for radio performance evaluation and radio system development. The DVK-2 is compatible with AX-RadioLab and AxCode::Blocks development software for the PC.



**ON Semiconductor®**

[www.onsemi.com](http://www.onsemi.com)

### APPLICATION NOTE

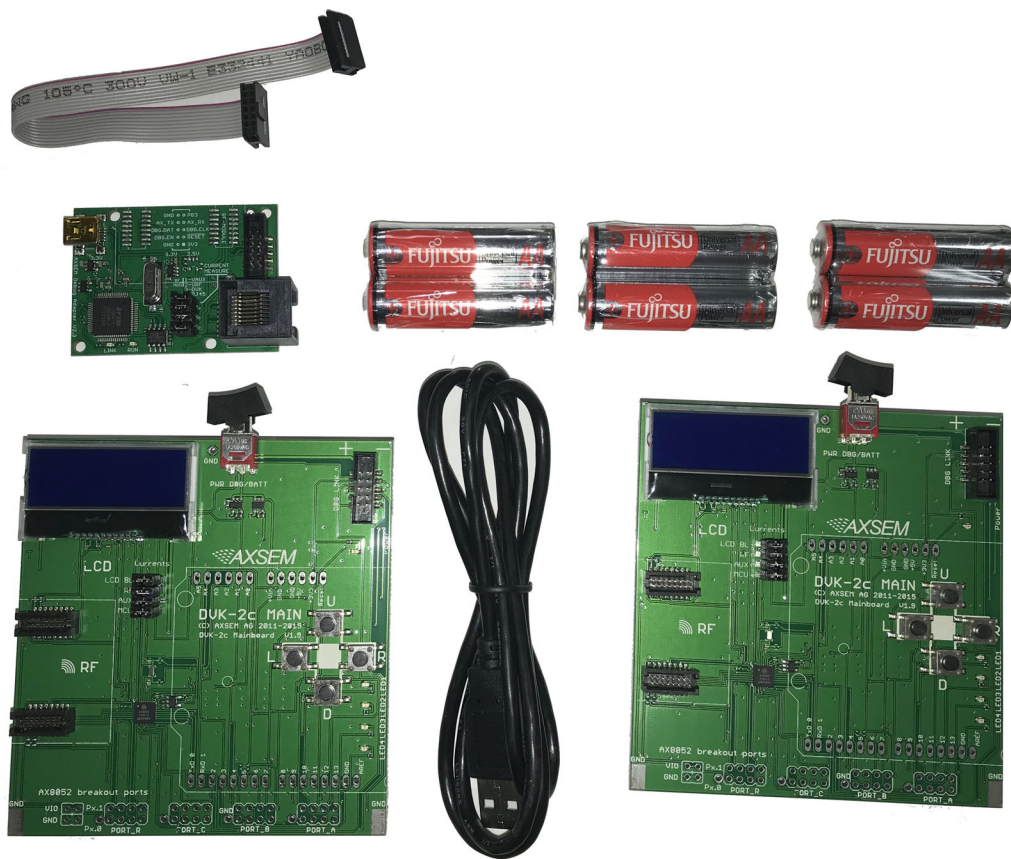


Figure 1. Content of the DVK-2 BASE KIT

The DVK-2 base-kit comes with:

- 2x mainboards with radio-module connectors for one radio add-on-kit module, AX8052F100 20 MIPS ultra-low-power micro-controller, LCD display, 4 buttons, 4 LEDs, AX debug connector and battery connector for 3 AA batteries
- 1x AXDBG debug adaptor
- 1x Mini USB cable
- 1x Debug cable

#### DVK-2 Add-on-kits

For evaluation of AX radio ICs, a variety of add-on-kits are available. A radio add-on-kit consists of a pair of radio add-on modules and antennas. Pictured below is an example of a DVK-2 add-on module. Each add-on kit contains two modules.

## AND9698/D



**Figure 2. Content of the ADD5043-915-x-GEVK Kit**

The following radio add-on kits are available:

- AX5045-915 add-on kit (ADD5045-915-x-GEVK)
- AX5043-915 add-on kit (ADD5043-915-x-GEVK)
- AX5043-868 add-on kit (ADD5043-868-x-GEVK)
- AX5043-433 add-on kit (ADD5043-433-x-GEVK)
- AX5043-169 add-on kit (ADD5043-169-x-GEVK)
- AX5243-868 add-on kit (ADD5243-868-x-GEVK)

- AX5051-868 add-on kit (ADD5051-868-x-GEVK)

For transmitter only evaluation the following transmitter add-on module is available

- AX5031-868 add-on kit (ADD5031-868-x-GEVK)

### Getting Started With DVK-2

Upon receiving the DVK-2, the add-on modules and batteries can be connected to immediately run the pre-installed packet error rate (PER) test. (See sections “Batteries and Add-on Modules” and “Pre-installed PER Test”).

To begin configuration of the radio and application development, the following steps should be completed in order:

1. Download and install the required software for the desired combination of main board and add-on modules. (See sections “Software” till “Configuration of AxCode::Blocks”)
2. Configure AxCode::Blocks to use the SDCC compiler (See section “Configuration of AxCode::Blocks”)
3. Insert the add-on modules to the mainboard, and connect the AX debug adapter as shown in the image below. Connect the AX debug adapter to the PC. (See section “Connecting the DVK-2 to a PC”)
4. Launch AX-RadioLab or AxCode::Blocks to begin radio configuration, testing or application development



**Figure 3. DVK-2 Main Board Connected to the AXDBG**

### Main Board Overview

A DVK-2 mainboard is the core of the DVK-2 development system. It uses an AX8052F100 as a versatile

ultra-low-power micro-controller. The mainboard is designed to evaluate AX radio ICs as well as being a platform for code development and testing.



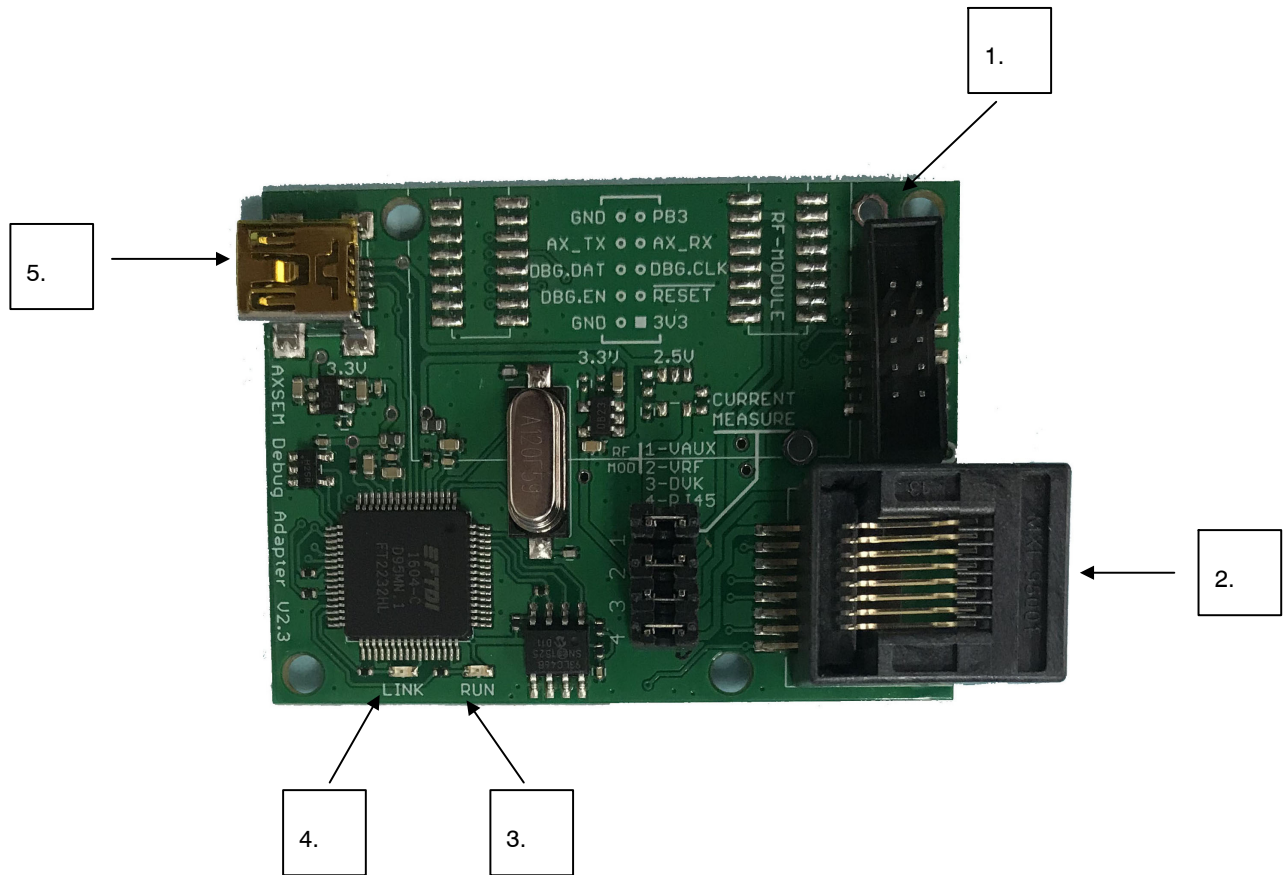


Figure 5. Details of the AXDBG Debugger

The AXDBG has the following interfaces and indicators:

1. Debug link connector
2. RJ45 Connector or legacy AX DVK systems
3. LED green indicating a program is executed on the MCU
4. LED red indicating the debug link is active, MCU in debug mode
5. Mini USB PC interface

#### Batteries and Add-on Modules

##### Batteries

Three AA batteries can be inserted on the bottom side of the DVK-2 mainboard. The orientation is indicated on the

board. The power switch toggles between battery powered and debug-link powered operation. There is no OFF position if batteries are inserted and the debug-link is connected.

##### Add-on Modules

One radio add-on module can be connected to the main board. The DVK-2 add-on modules have to be connected to the main board with the SMA connector facing outside.

NOTE: There is no safeguard for polarity or wrong insertion of add-on modules.

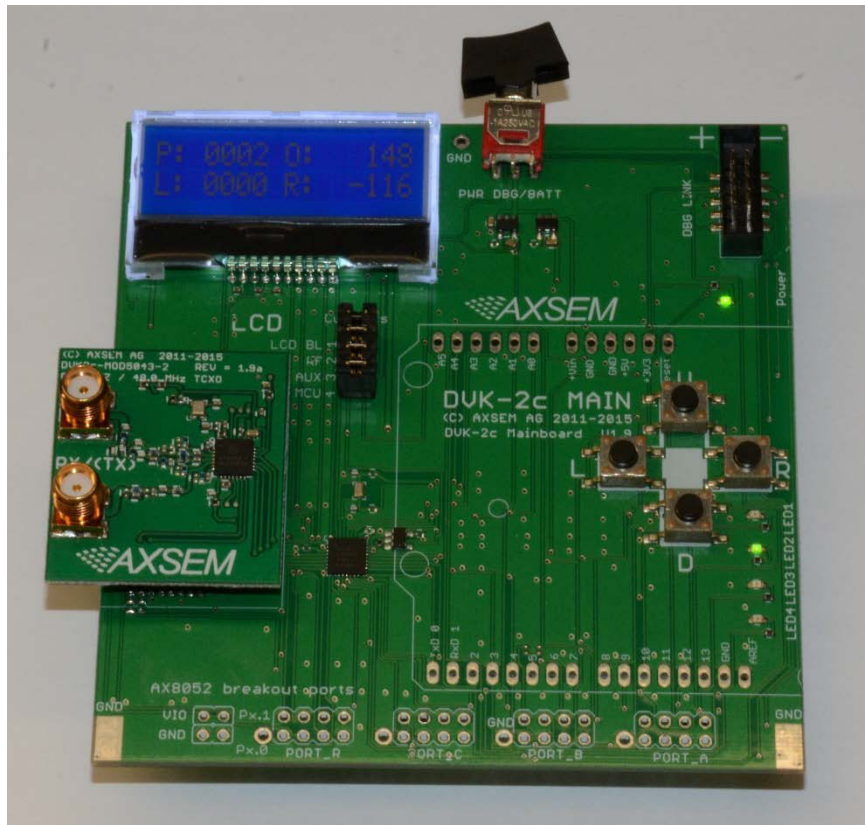


Figure 6. DVK2 Main Board with an ADD5043 RF Module Connected

**Antennas**

Different types of antennas can be used with the AX radio add-on modules. In addition to the antenna that ships with the kit, any 50 Ω antenna designed for a frequency matching the radio add-on module and being equipped with a male SMA connector can be used.

**Software**

Depending on your application and add-on-kit you need to use different ON Semiconductor AX software as listed in the Table 1 below.

Table 1.

Development Kit	Matching Software
DVK-2 base-kit, no add-on module attached	<ul style="list-style-type: none"> <li>• AX8052IDE package                             <ul style="list-style-type: none"> <li>◆ AXCode::Blocks</li> <li>◆ AXSDB</li> <li>◆ LibMF</li> <li>◆ SDCC</li> </ul> </li> </ul>
DVK-2 base-kit with AX5043, AX5243 or AX5045 (Note 1) add-on modules	<ul style="list-style-type: none"> <li>• AX8052IDE package                             <ul style="list-style-type: none"> <li>◆ AXCode::Blocks</li> <li>◆ AXSDB</li> <li>◆ LibMF</li> <li>◆ SDCC</li> </ul> </li> <li>• AX-RadioLab</li> </ul>
DVK-2 base-kit with AX5051 add-on module	<ul style="list-style-type: none"> <li>• AX8052IDE package                             <ul style="list-style-type: none"> <li>◆ AXCode::Blocks</li> <li>◆ AXSDB</li> <li>◆ LibMF</li> <li>◆ SDCC</li> </ul> </li> <li>• AXGen2-RadioLab</li> </ul>
DVK-2 base-kit with AX5031 (Note 2) add-on module	<ul style="list-style-type: none"> <li>• AX8052IDE package                             <ul style="list-style-type: none"> <li>◆ AXCode::Blocks</li> <li>◆ AXSDB</li> <li>◆ LibMF</li> <li>◆ SDCC</li> </ul> </li> <li>• AX-ParamCalc</li> </ul>

1. The AX5045 requires AXCodeBlocks V1.27 (or greater) and AX-RadioLab V2.11a (or greater).  
 2. The AX5031 add-on module is a transmitter only and does not work on a mainboard programmed as a receiver.

**Software Installation**

First, download the required software packages (specified in the previous section) from the ON Semiconductor Wireless Transceivers software page:

<http://www.onsemi.com/PowerSolutions/supportDoc.do?type=software&category=16860>

The default location for AX–RadioLab, AxCode::Blocks and AXSDB is in the C:\Program Files (x86)\ON Semiconductor (or C:\Program Files (x86)\AXSEM for AXGen2–RadioLAB and AXParamCalc). LibMF and the supporting libraries are installed in the AXSDB directory. The SDCC C–compiler is typically installed directly into the C:\Program Files directory. Please check the respective documentations for further detail.

**AX–IDE for AX5043, AX5243 and AX5045:**

After unzipping the downloaded package, execute the AXIDE–setup–1.27.exe to start the installation of the multiple tools included in the AX–IDE such as AX–RadioLAB and AxCode::Blocks.

NOTE: The installer requests to reboot the computer, but is not necessary.

**AX8052–IDE for AX5051 and AX5031:**

Important Note: AX–RadioLAB uses the AX8052–IDE software package in the background. Therefore the AX8052–IDE software package has to be installed and configured before installation of the other software. Otherwise you may experience difficulties in installing the driver later.

NOTE: The installer requests to reboot the computer, but is not necessary.

After downloading, each package will need to be unzipped. Each package contains a windows installer executable, which needs administrator privileges to be run. Run the installer for the AX8052–IDE package first. It installs the AxCode::Blocks IDE, the SDCC C–compiler, the LibMF support libraries, the AXSDB command line debugger and low level drivers. Repeat the same process for AXGen2–RadioLab, or AX–ParamCalc, depending on the add–on module. (See Table 1 in previous section).

NOTE: The installer requests to reboot the computer, but is not necessary.

**Configuration of AxCode::Blocks**

After starting AxCode::Blocks for the first time, the user has to select SDCC as the default compiler. The following

window will pop-up. After this configuration process you are ready to go.

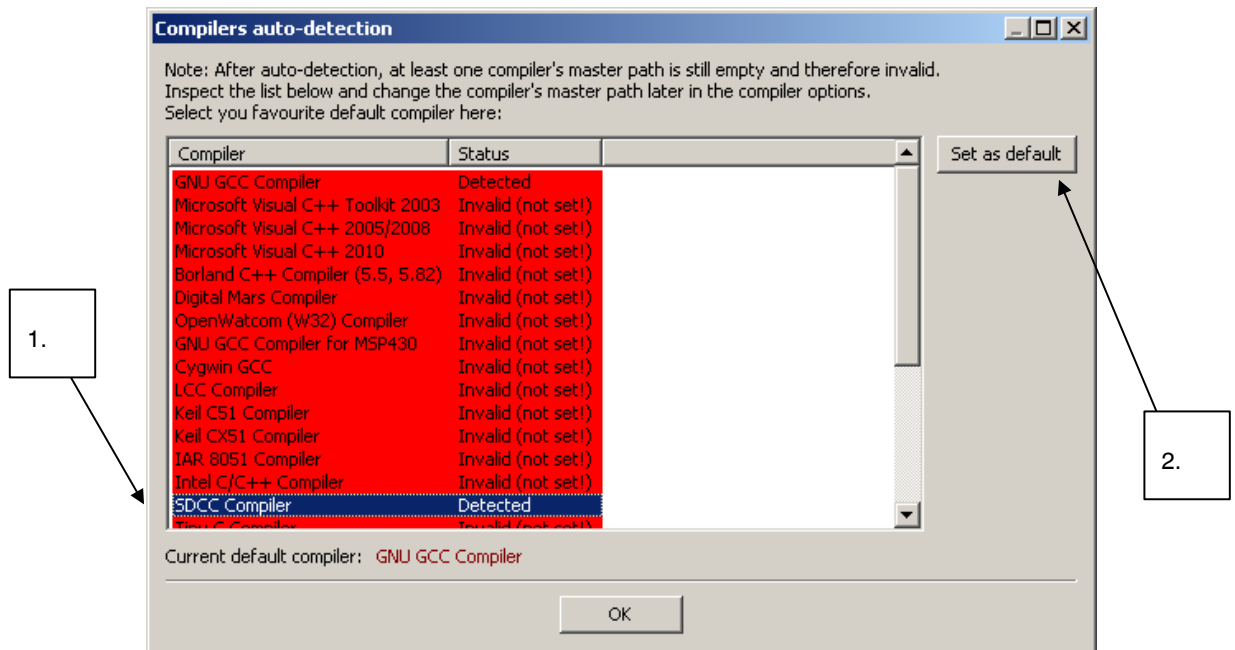


Figure 7.

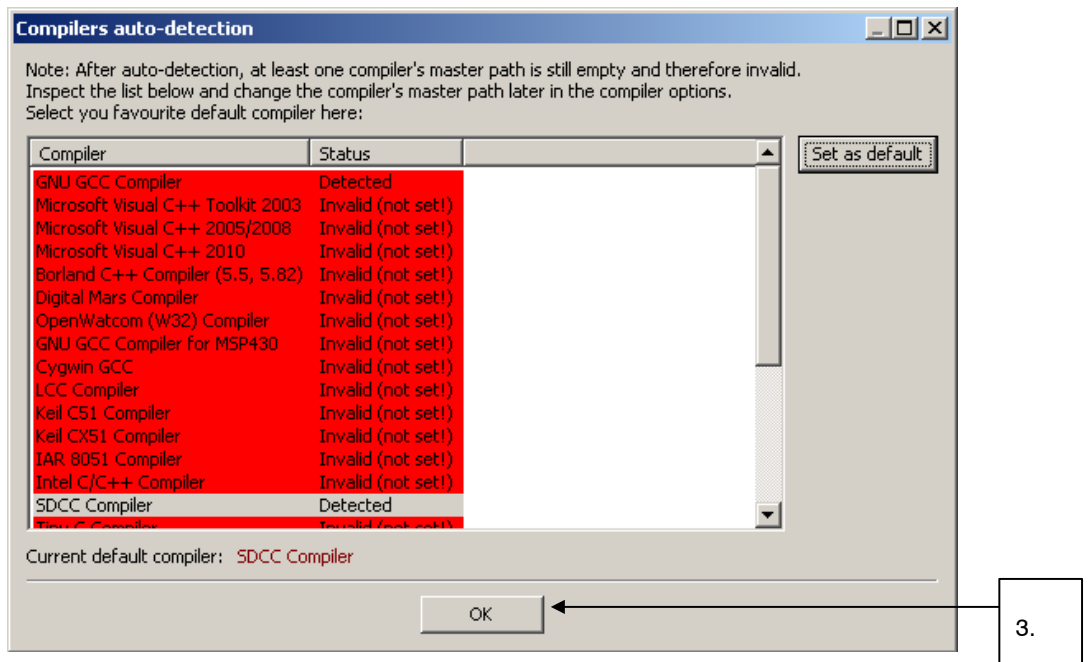


Figure 8.



**Connecting the DVK-2 to a PC**

- Step 1: Verify the proper software is installed and configured, as detailed in the previous sections.
- Step 2: Connect the AXDBG via the USB cable to the computer. Since the driver is already installed, the

computer should report 2 “USB Serial Converters” as well as a “USB Serial Port” in your Device Manager.

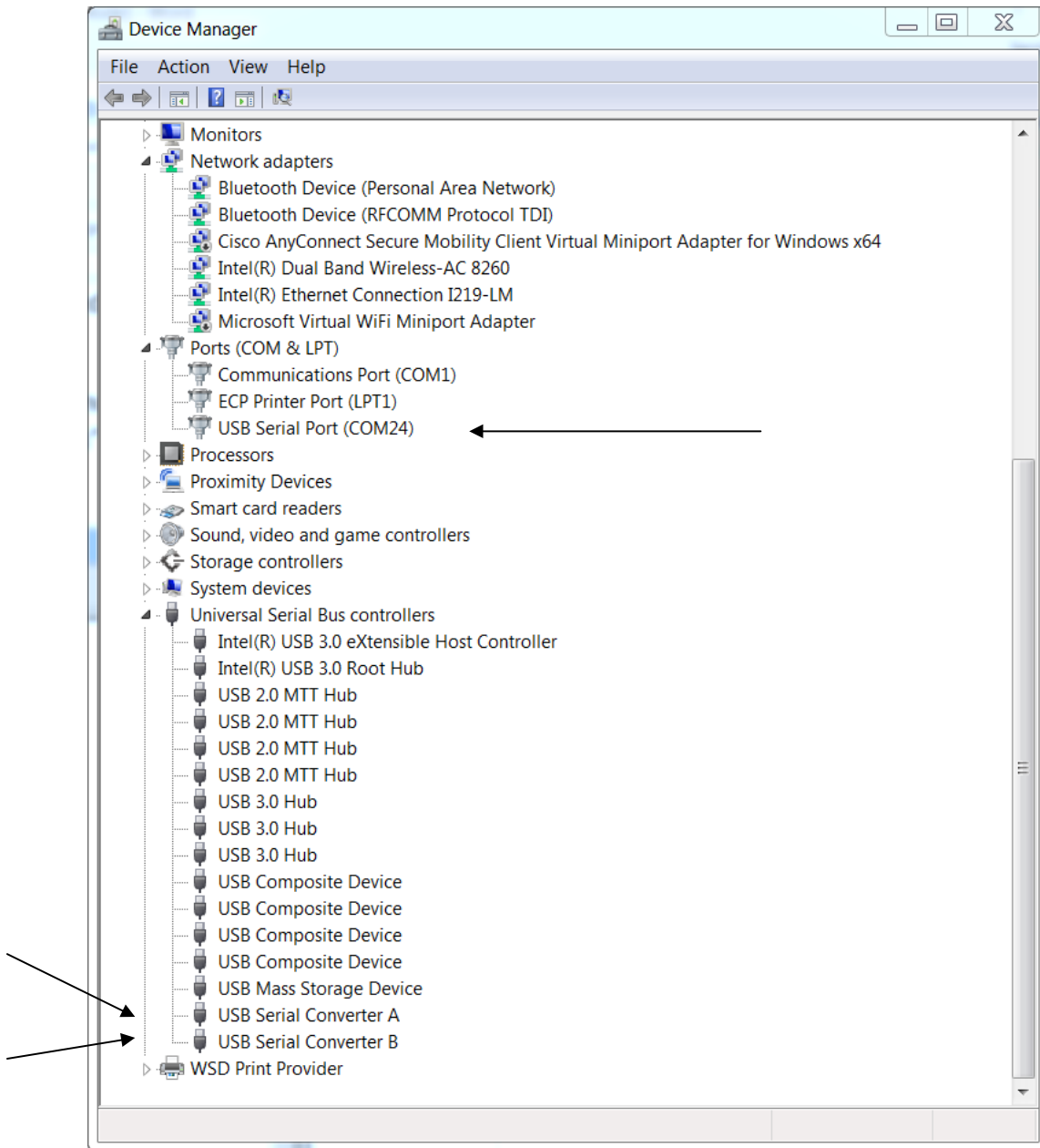


Figure 9.

- Step 3: Connect a DVK-2 mainboard via the debug cable to the AXDBG debug adapter.

**Pre-installed PER Test**

The DVK-2 mainboards come with a pre-installed software for tests with DVK-2 radio add-on modules. The pre-installed software recognizes most AX radio add-on modules and automatically starts testing packet error rate. One DVK-2 mainboard is programmed to be a transmitter and the other to be a receiver. After inserting batteries, attach radio add-on modules and set the power supply switch to battery. For longer range, use antennas. The transmitter indicates the number of the last transmitted packet, while the receiver indicates the packet error rate (PER), the RSSI and the frequency offset.

The pre-installed software is the same as the default master/slave application that can be configured in AX-RadioLab. See the following sections for details on how to use AX-RadioLab to configure this application.

*Transmitter*

The transmitter sends a standard packet every second.

The LCD display shows:

**MASTER**  
**TX 0051** (the last transmitted packet number)

Transmit parameter:

- 868.3 MHz
- 100 kbps
- max power
- FSK, deviation h = 0.5
- duty cycle ~ 1 s

*Receiver*

The application on the AX8052F100 receives the data packets from the transmitter and shows packet error rate, signal strength and frequency offset.

The LCD display shows:

**P:0014 O:00397**  
**L:0000 R: -034**

P is the number of received packets

L is the number of lost packets

O is the frequency offset in Hz

R is the receiver strength indicator (RSSI) in dBm

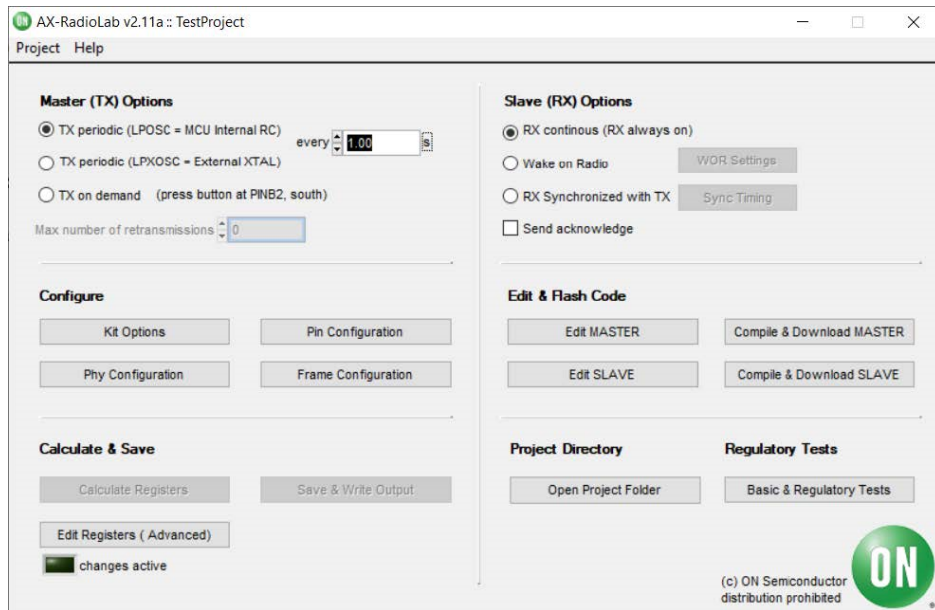
NOTES: The receiver screen updates only if a valid packet was received.

If the transmitter is reset during the test, the P and L values become invalid.

**Working With AX-RadioLab**

AX-RadioLab is the most advanced development tool for SDR radio applications. It offers a variety of transmitter and receiver options such as

- Periodic transmission of packets timed with the RC oscillator of the AX5043
- Periodic transmission of packets timed with the 32 kHz XTAL oscillator of the MCU
- Transmit on push-button
- Wake-on-radio reception with programmable wake-up interval
- Synchronous transmit and receive with programmable wake-up interval
- Optional acknowledge package send for all modes



**Figure 10. Screenshot of AX-RadioLab Main Panel GUI**

Additionally, AX-RadioLab allows the user to configure the packet format, as well as PHY parameters.

Since AX–RadioLab is a source code generator, the developer can use the generated C code example project as a robust foundation upon which to build the end application.

AX–RadioLab also estimates the power consumption of the radio device. For most modes the AX8052F100 power consumption is negligible. Average and peak currents are indicated for transmitter and receiver.

Additional documentation for AX–RadioLab can be found on the AX5043 product page.

(<http://www.onsemi.com/PowerSolutions/product.do?id=AX5043> )

NOTE: AX–RadioLab is designed for use only with the AX5043, AX5243 and AX5045 based radio applications. For AX5051–based applications, see AXGen2–RadioLab, available on the AX5051 product page.  
(<http://www.onsemi.com/PowerSolutions/product.do?id=AX5051-1-TA05> )

**Working With AxCode::Blocks**

AxCode::Blocks is the graphical IDE for code development on AX micro–controllers. It enables the developer to access all the debugging features of the AX micro–controllers, in particular:

- Unlimited (limited only by memory size) number of break points
- Access to all AX8052F1xx MCU registers as well as to all AX radio chip registers
- Debug link UART in a window for debugging printf–style without the need of extra hardware
- SDCC C–compiler pre–installed and ready to go

Additional documentation for AxCode::Blocks can be found on the AX8052F100 product page.

(<http://www.onsemi.com/PowerSolutions/product.do?id=AX8052F100> )

**Interaction Between Development Tools**

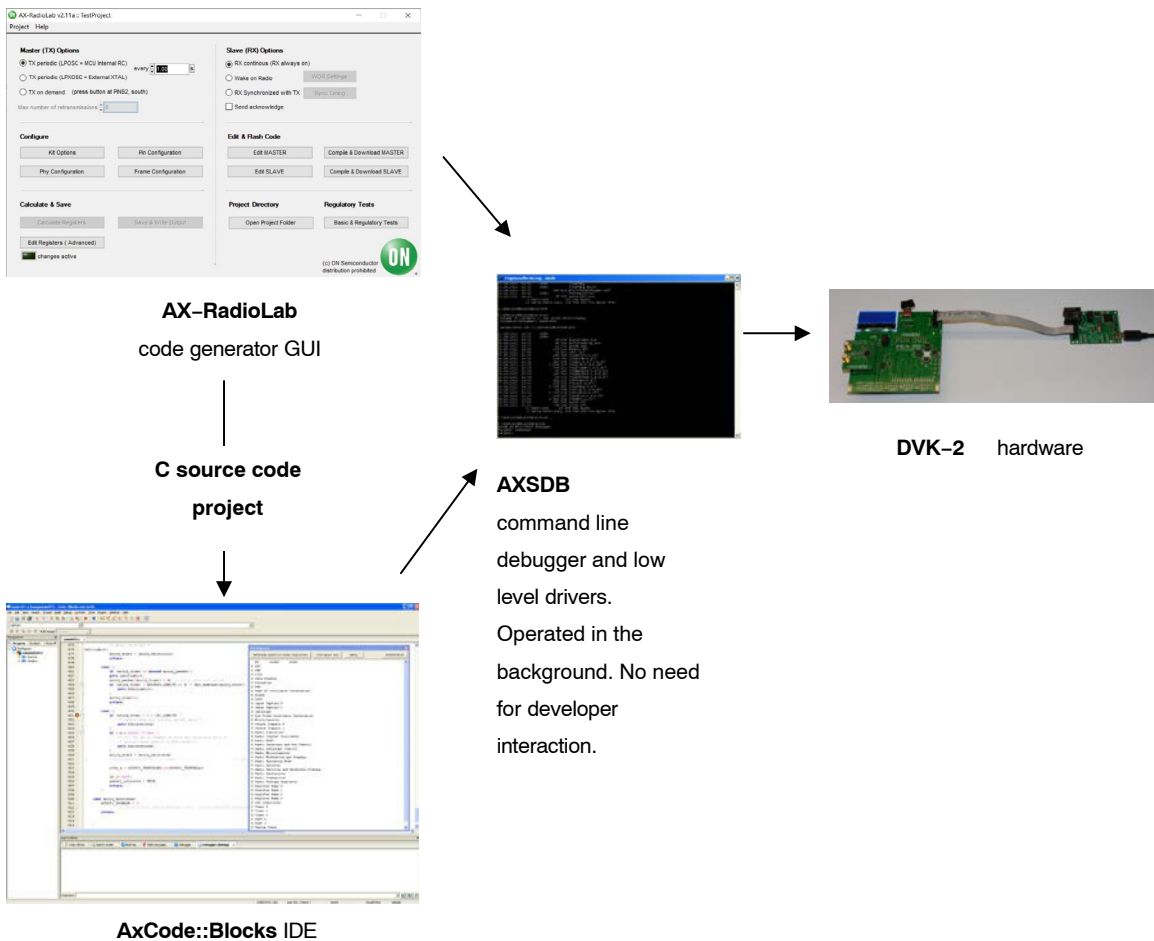


Figure 11.

The AX8052–IDE–setup.exe contains all tools for developing and debugging C source code applications for the AX8052F1xx family of micro–controllers.

The AX–RadioLab GUI is a code generator, which configures C code, which can then be compiled and

downloaded directly from the AX–RadioLab or it can be edited, modified and debugged in the AxCode::Blocks IDE.

The user normally starts in the AX–RadioLab GUI, which generates all necessary configuration and program files. The project can optionally be viewed and modified in

## AND9698/D

AxCode::Blocks. Be aware that there are no protective mechanisms in the AxCode::Blocks to prevent mal function of modified C code.

For details look into the documentation of AX–RadioLab (<https://www.onsemi.com/pub/Collateral/AND9356–D.PDF>) and AxCode::Blocks.

### Saving Factory Calibration Data

The AX8052F1xx MCU contains factory calibration data. This data is stored in the topmost sector of the flash memory (address 0xFC00). When programming the MCU using the axsdb command line debugger, the AxCode::Blocks IDE or AX–RadioLAB, the calibration data is automatically saved and written back to the flash memory. However, experience shows that in a laboratory environment the calibration data occasionally gets lost due to software crashes, removing a cable in the wrong moment etc. It is recommended to back up the calibration data to a file, so it can be recovered in the case of accidental loss. The calibration data can be saved by opening a windows command window (enter cmd) in the windows start menu and entering the command

```
axsdb --savecalib=caldata_board1.txt
```

Repeat this for the second DVK2 main board:

```
axsdb --savecalib=caldata_board2.txt
```

Please mark the DVK2 boards (board1, board2), so you know which file belongs to which board.

The calibration data can be restored from the file using

```
axsdb --loadcalib=caldata_board1.txt
```

NOTE: For further documentation, see the AXSDB user manual.

(<http://www.onsemi.com/pub/Collateral/AND9370–D.PDF>)

### Glossary

**AxCode::Blocks:** An IDE for AX8052F1xx micro-controllers

**AX–RadioLab:** An application C code and settings generator for the AX5043, AX5243 and AX5045 radio chip running on the AX8052F100 micro-controller

**AX–RadioLab:** An application C code and settings generator for the AX5051 radio chip running on the AX8052F100 micro-controller

**Debug link:** The interface between AXDBG debug adapter and AX8052F1xx micro-controllers

**AXDBG:** The USB debug adapter for AX8052F1xx micro-controllers

**AXSDB:** A command line debugger using AXDBG. It is also fully suitable for use in mass production

**SDCC:** A C-compiler for AX8052F1xx micro-controllers

**Main board:** The DVK–2 development board with AX8052F100 micro-controller and LCD display

**DVK–2:** The development system for AX micro-controller and radio applications

**DVK–2 base-kit:** Two mainboards, an AXDBG and cables

**DVK–2 add-on:** Set of two modules with AX radio chips  
Radio module: Plug-in module for mainboards with AX radio chips


**AX8052F100:** The ultra-low-power micro-controller used on the DVK–2 mainboards

**LibMF:** Support libraries for the AX8052F100 and the DVK–2. It normally includes the libraries libax5031, libax5031, libax5042, libax5051, libax5043, which are in separate directories

**LibMFCrypto:** A library which enables the use of the AES or DES functionality of the AX8052F1xx micro-controllers

**Pre-installed PER test:** A software which is programmed on the DVK–2 mainboards

**AX–ParamCalc:** GUI application to generate configuration files for AX5042, AX5051, AX5151, AX5031, AX5131, AX50424. The configuration files are compatible with AX–Easy\_Packet but can also be used in customer specific code

ON Semiconductor and  are trademarks of Semiconductor Components Industries, LLC dba ON Semiconductor or its subsidiaries in the United States and/or other countries. ON Semiconductor owns the rights to a number of patents, trademarks, copyrights, trade secrets, and other intellectual property. A listing of ON Semiconductor's product/patent coverage may be accessed at [www.onsemi.com/site/pdf/Patent-Marking.pdf](http://www.onsemi.com/site/pdf/Patent-Marking.pdf). ON Semiconductor reserves the right to make changes without further notice to any products herein. ON Semiconductor makes no warranty, representation or guarantee regarding the suitability of its products for any particular purpose, nor does ON Semiconductor assume any liability arising out of the application or use of any product or circuit, and specifically disclaims any and all liability, including without limitation special, consequential or incidental damages. Buyer is responsible for its products and applications using ON Semiconductor products, including compliance with all laws, regulations and safety requirements or standards, regardless of any support or applications information provided by ON Semiconductor. "Typical" parameters which may be provided in ON Semiconductor data sheets and/or specifications can and do vary in different applications and actual performance may vary over time. All operating parameters, including "Typicals" must be validated for each customer application by customer's technical experts. ON Semiconductor does not convey any license under its patent rights nor the rights of others. ON Semiconductor products are not designed, intended, or authorized for use as a critical component in life support systems or any FDA Class 3 medical devices or medical devices with a same or similar classification in a foreign jurisdiction or any devices intended for implantation in the human body. Should Buyer purchase or use ON Semiconductor products for any such unintended or unauthorized application, Buyer shall indemnify and hold ON Semiconductor and its officers, employees, subsidiaries, affiliates, and distributors harmless against all claims, costs, damages, and expenses, and reasonable attorney fees arising out of, directly or indirectly, any claim of personal injury or death associated with such unintended or unauthorized use, even if such claim alleges that ON Semiconductor was negligent regarding the design or manufacture of the part. ON Semiconductor is an Equal Opportunity/Affirmative Action Employer. This literature is subject to all applicable copyright laws and is not for resale in any manner.

### PUBLICATION ORDERING INFORMATION

LITERATURE FULFILLMENT:  
Email Requests to: [orderlit@onsemi.com](mailto:orderlit@onsemi.com)

TECHNICAL SUPPORT  
North American Technical Support:  
Voice Mail: 1 800-282-9855 Toll Free USA/Canada  
Phone: 011 421 33 790 2910

Europe, Middle East and Africa Technical Support:  
Phone: 00421 33 790 2910  
For additional information, please contact your local Sales Representative

# Mouser Electronics

Authorized Distributor

Click to View Pricing, Inventory, Delivery & Lifecycle Information:

[onsemi:](#)

[DVK-BASE-2-GEVK](#) [ADD5043-915-1-GEVK](#)