Bosch Connected Devices and Solutions

XDK110: Getting Started Guide with the Enterprise Network Connection

**Cross-Domain Development Kit XDK110**

Platform for Application Development

Bosch_SL-en_4C_S­­



|  |  |
| --- | --- |
| **XDK110: Getting Started Guide with Enterprise Network Connection** | |
| Document revision | 1.0 |
| Document release date | **August 24, 2016** |
| Document number | BCDS-XDK110- |
| Technical reference code(s) |  |
| Notes | Data in this document is subject to change without notice. Product photos and pictures are for illustration purposes only and may differ from the real product’s appearance.  This document is confidential and under NDA inherent with the purchase of an XDK110.  **Advance information – Subject to change without notice** |

**XDK110**

Platform for Application Development

General Description

The XDK is a wireless sensor device developed to enable rapid prototyping for the internet of things (IoT) space. It allows users to quickly realize their own test case (or “Proof of concept” project) and understand what the requirements of their ideal product are.

The XDK comes with a lithium ion battery, an extension board, a micro USB 2.0 cable and mounting plate and screws. The XDK is equipped with 7 physical sensors: Accelerometer, Gyroscope, Magnetometer, Inertial (Accelerometer and Gyroscope), Environmental (Pressure, Humidity, and Temperature), Ambient Noise, and Ambient Light. The XDK has a Micro-SD card slot and two antennas: one for wireless LAN and one for Bluetooth 4.0 communications. The XDK has 3 programmable LEDs, one LED for charge indication, and two programmable buttons.

Development with the XDK is easily done with the XDK Workbench. A programming suite based on the Eclipse Platform, which requires a PC with a Windows 7 or higher operating system. The XDK Workbench delivers all of the API’s, source code and demos necessary for new users to quickly and easily start development of their application with the XDK.

This document details how to setup the Enterprise Network Connection example code in the XDK Workbench, flash the XDK with the firmware, upgrade the Wi-Fi module, and connect to the WPA2-Enterprise Wi-Fi network.

This document assumes the user has been granted access to the XDK Community and has already downloaded and installed the XDK Workbench 1.6.0. It is also assumes the user knows how to import a demo into the XDK Workbench. For help with how to download and install the XDK workbench or how to import a demo, please navigate to the XDK website ([www.xdk.io](http://www.xdk.io)), click on the Help & Learning tab and view the Workbench Installation guide. The download for the XDK workbench can be found on the XDK website under the Downloads tab.

Table of Contents

[1. XDK Enterprise Network Connection Project Setup 4](#_Toc459797755)

[1.1 WiFi Module upgrade 4](#_Toc459797756)

[1.2 Connecting to Enterprise Network 5](#_Toc459797757)

[1.2.1 Configuration Parameters: 6](#_Toc459797758)

[2. Document History and Modification 7](#_Toc459797759)

[3. Appendix 7](#_Toc459797760)

[3.1 Creating Certificate Files (Optional) 7](#_Toc459797761)

# XDK Enterprise Network Connection Project Setup

The source code for the Enterprise Network Connection demo is located on the XDK website at <http://www.xdk.io>. From the home page select the Downloads tab. Create an account or sign in and select the Demos menu item. Scroll down until you find the Enterprise Network Connection section. Click on Download to download the code (the latest version of this document will also be downloaded). Extract the zip file onto a computer installed with the XDK workbench.

Once you have downloaded all the source code and placed the source files in the proper location, import both projects into the XDK workbench, i.e. WiFiHostProgramming and EnterpriseWiFi project. Please see the XDK Workbench or User Guide tutorials in the Help & Learning section of the XDK website for more information on how to properly import a project. Once this is done insure you also have the following items in order to confirm a working installation of the Enterprise Network Connection Demo project.

* XDK110 – Hardware Development Kit
* XDK Workbench – Necessary to configure the source code for the wireless network and to flash the XDK. The workbench can be downloaded from the XDK Community.
* WPA2-Enterprise Wi-Fi network and along with the following credentials:
  + SSID
  + Username
  + Password

This rest of this section will show the user how to configure the code and flash the XDK using the XDK Workbench.

## Wi-Fi Module upgrade

The Wi-Fi Host Programming project upgrades the Wi-Fi module by serially flashing in the service pack header files. In addition, the project will also flash in a dummy certificate file to enable enterprise network access. First import the Wi-Fi Host Programming project, then flash it to the XDK. Pressing button 1 of the XDK will output the current version of the Wi-Fi module in the XDK-Workbench output console. Pressing button 2 of the XDK will automatically flash in the service pack, wait two seconds, and then flash in the dummy certificate file to the Wi-Fi module. You should expect version 2.6.0.5 along with certificate write as shown in the XDK workbench colsole when pressing button 2.

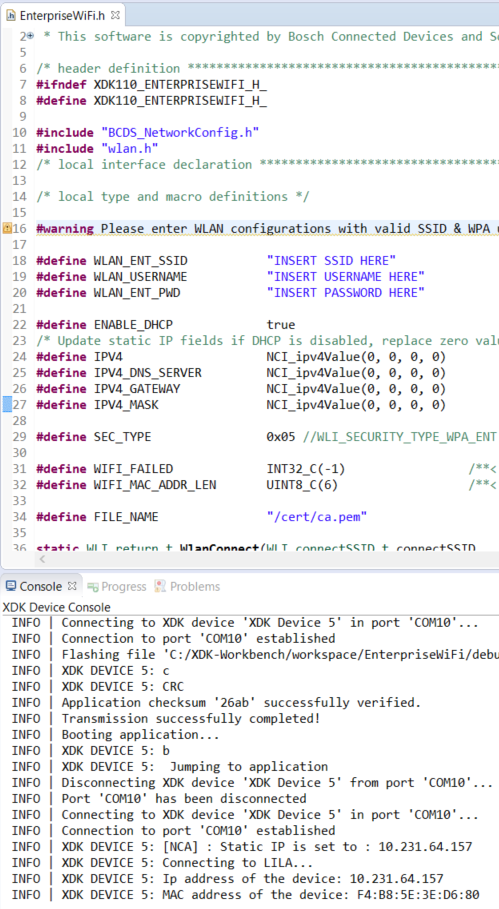


## Connecting to Enterprise Network

The Enterprise Wi-Fi project connects to WPA2-Enterprise Wi-Fi network given specific network parameters. You will need the following information:

1. SSID of the Wi-Fi network
2. Username
3. Password
4. Network IP configuration (static or DHCP)
   1. If static, you will need:
      1. Open IPv4 network address
      2. DNS server address
      3. Gateway address
      4. Subnet mask
5. Network EAP method

Once you have gather all the necessary information, configure these settings in EnterpriseWiFi.h header file:



## 1.2.1 Configuration Parameters:

Header file configuration tips/comments:

1. WLAN\_ENT\_SSID
   1. SSID of the available enterprise network (WPA2-Enterprise)
2. WLAN\_USERNAME
   1. Login credential (username)
3. WLAN\_ENT\_PWD
   1. Login credential (password)
4. ENABLE\_DHCP
   1. Toggle DHCP/Static IP configuration. True means connecting to DHCP enabled network. If false, configure IPV4 parameters 5-8 below.
5. IPV4
   1. IP address for static IP configuration
6. IPV4\_DNS\_SERVER
   1. Static IP DNS Server address
7. IPV4\_GATEWAY
   1. Static IP Gateway Server address
8. IPV4\_MASK
   1. Static IP subnet mask
9. SEC\_TYPE
   1. EAP method security type. Please choose one that matches your enterprise network configuration (#defines available in wlan.h):
      1. SL\_ENT\_EAP\_METHOD\_TLS
      2. SL\_ENT\_EAP\_METHOD\_TTLS\_TLS
      3. SL\_ENT\_EAP\_METHOD\_TTLS\_MSCHAPv2
      4. SL\_ENT\_EAP\_METHOD\_TTLS\_PSK
      5. SL\_ENT\_EAP\_METHOD\_PEAP0\_TLS
      6. SL\_ENT\_EAP\_METHOD\_PEAP0\_MSCHAPv2 (default)
      7. SL\_ENT\_EAP\_METHOD\_PEAP0\_PSK
      8. SL\_ENT\_EAP\_METHOD\_PEAP1\_TLS
      9. SL\_ENT\_EAP\_METHOD\_PEAP1\_MSCHAPv2
      10. SL\_ENT\_EAP\_METHOD\_PEAP1\_PSK
      11. SL\_ENT\_EAP\_METHOD\_FAST\_AUTH\_PROVISIONING
      12. SL\_ENT\_EAP\_METHOD\_FAST\_UNAUTH\_PROVISIONING
      13. SL\_ENT\_EAP\_METHOD\_FAST\_NO\_PROVISIONING

Save the changes and build/flash the project onto the XDK. Once flashed, the XDK will automatically attempt to connect to the enterprise Wi-Fi. This may take 1-2 minutes to complete. If successfully connected, you will see the IPv4 & MAC addresses being printed on the XDK workbench console. Otherwise, the application will hang until a connection is established.

# Document History and Modification

|  |  |  |  |
| --- | --- | --- | --- |
| **Rev. No.** | **Chapter** | **Description of modification/changes** | **Date** |
| 1.0 |  | Initial Release | 2016-08-24 |
|  |  |  |  |

# Appendix

## Creating Certificate Files (Optional)

Reference CC3100/CC3200 SimpleLink Wi-Fi Interneton-a-Chip User’s Guide: <http://www.ti.com/lit/ug/swru368a/swru368a.pdf>

Page 171 Appendix C gives detailed explanation of generating new certificate files. The Wi-Fi Host Programming project already contains a dummyCertificate.h file which has a C array version of the pem certificate string.