

## Release Notes for the Avnet AT&T IoT Cellular Starter Kit

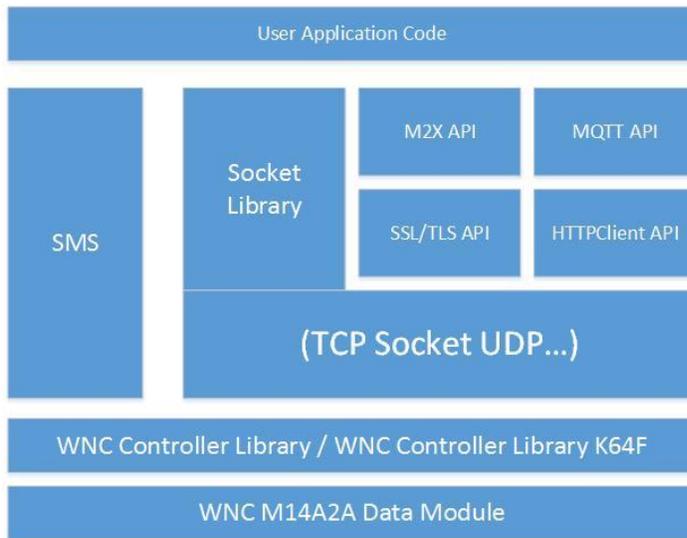
**Release Tag:** att\_cellular\_K64\_wnc\_14A2A\_20161117

**WNC Firmware Requirement:** CM\_MPSS\_M14A2A\_v11.50.164451

The software for the IoT Kit is a combination of software on the NXP FRDM-K64F development board and a WNC 14A2A data module. This data module implements an LTE data module that enables data to be sent over the cellular network.

### New Features

This release adds a communications library interface to the IoT Starter Kit to simplify programming the Kit. Architecturally, the software interface architecture is:



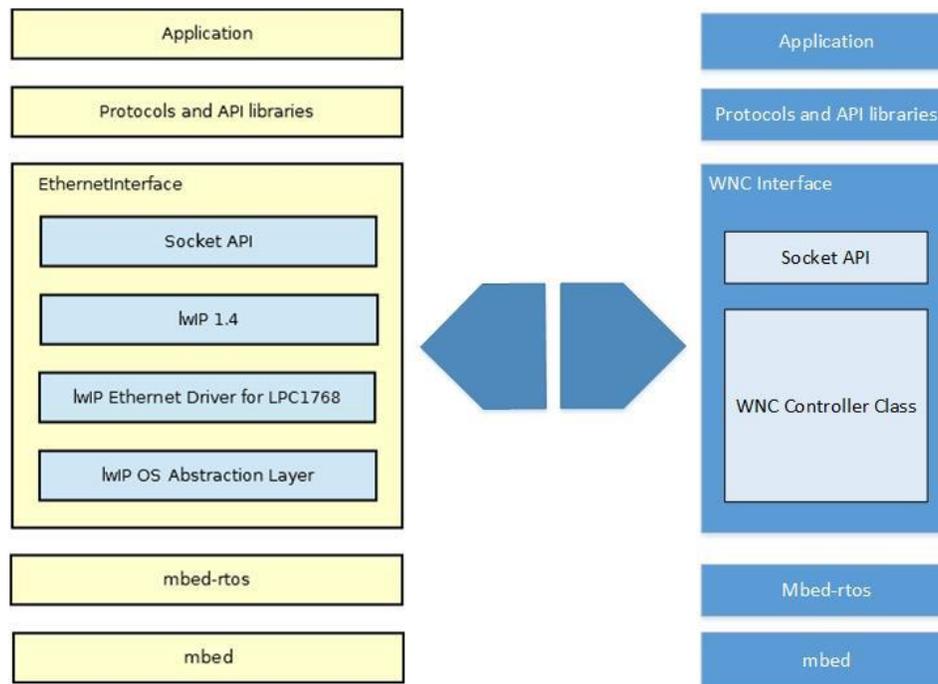
The M14A2A Data Module implements a basic TCP/IP stack and exposes the interface based on the 3GPP TS 27.007/005/010 Serial AT command specifications. The M14A2A Data Module implements Internet Service Commands that aggregate AT command functionality to implement a full set of Internet Services.

The C++ WNC Controller Library implements the necessary control functionality to allow a user to easily utilize TCP/IP and SMS functionality. The WNC Controller Library is implemented as an abstract base class which is OS agnostic. On top of the

abstract base class is an implementation specific class which has been implemented for the NXP FRDM-K64F board (in the sample code, this is the C++ WncControllerK64F class).

The C++ WNCInterface class builds upon the WNC Controller Library to fulfill an mbed OS networking API. This class is similar to the standard EthernetInterface class that has been implemented within the MBED system—in principle, you can change the object instantiation/creation from EthernetInterface to WNCInterface and the program using that networking API will continue to work similarly.

The WNCInterface Library implements a high level socket API based on the Berkeley socket API and comparing the WNCInterface to the EthernetInterface, the architecture compares as shown (<https://developer.mbed.org/handbook/Networking>):



### Example Code

To demonstrate the use of the WNC Interface, several example programs have been implemented and are available on the Avnet Team Site:

<https://developer.mbed.org/teams/Avnet/>

You will also need:

1. An mbed developer account at <https://developer.mbed.org>
2. The FRDM-K64F platform added to your workspace
3. Import an example program and follow the instructions in the readme.md file

Note: the WncControllerLibrary and WNCInterface will be automatically included with each example program so there is no need to explicitly import those two programs, but they are listed below for completeness.

Each program has an associated Wiki/Readme which explains how to build and what the expected output is.

Library	Description
WncControllerLibrary / WncControllerK64F <a href="https://developer.mbed.org/teams/Avnet/code/WncControllerLibrary/">https://developer.mbed.org/teams/Avnet/code/WncControllerLibrary/</a> <a href="https://developer.mbed.org/teams/Avnet/code/WncControllerK64F/">https://developer.mbed.org/teams/Avnet/code/WncControllerK64F/</a>	This library is implemented as an abstract class for basic functionality, and a device specific component for the AT&T IoT kit which uses the FRDM-K64F board.
WNCInterface <a href="https://developer.mbed.org/teams/Avnet/code/WNCInterface/">https://developer.mbed.org/teams/Avnet/code/WNCInterface/</a>	This is the interface class that implements the sockets API, SMS API, and SSL/TLS API's
<b>Example Programs using the Core Components:</b>	
BluemixQS <a href="https://developer.mbed.org/teams/Avnet/code/BluemixQS/">https://developer.mbed.org/teams/Avnet/code/BluemixQS/</a>	This is the 'Bluemix Quick Start' example program. Using the AT&T IoT Kit, The temperature/humidity is sent to IBM's Internet of things quick start website ( <a href="https://quickstart.internetofthings.ibmcloud.com/#/">https://quickstart.internetofthings.ibmcloud.com/#/</a> ) and the time series values are automatically plotted.
BluemixDemo <a href="https://developer.mbed.org/teams/Avnet/code/BluemixDemo/">https://developer.mbed.org/teams/Avnet/code/BluemixDemo/</a>	Similar to the Bluemix Quick Start, this example sends data to IBM's Bluemix console. The data is then processed using Node-Red and commands sent back to the IoT Kit to drive the onboard LED.
WNCInterface_M2XMQTTdemo <a href="https://developer.mbed.org/teams/Avnet/code/WNCInterface_M2XMQTTdemo/">https://developer.mbed.org/teams/Avnet/code/WNCInterface_M2XMQTTdemo/</a>	The M2XMQTT example program interacts with the AT&T M2X online service using the MQTT protocol.
WNCInterface_M2Xdemo <a href="https://developer.mbed.org/teams/Avnet/code/WNCInterface_M2Xdemo/">https://developer.mbed.org/teams/Avnet/code/WNCInterface_M2Xdemo/</a>	The M2X example program is similar to the M2XMQTT example program but it uses an HTTP Client rather than MQTT for the underlying transport.
WNCInterface_MQTT_hivemq <a href="https://developer.mbed.org/teams/Avnet/code/WNCInterface_MQTT_hivemq/">https://developer.mbed.org/teams/Avnet/code/WNCInterface_MQTT_hivemq/</a>	The MQTT HiveMQ example program demonstrates how to interact with an Enterprise MQTT broker service. This example program allows a user to publish and subscribe to various topics and uses a PC based tool to allow for interaction along with the web-site monitoring system.
WNCInterface_HTTP_example <a href="https://developer.mbed.org/teams/Avnet/code/WNCInterface_HTTP_example/">https://developer.mbed.org/teams/Avnet/code/WNCInterface_HTTP_example/</a>	This example demonstrates basic socket capabilities of the WNC device. It implements the following interfaces: 1. TCP Socket 2. UDP Socket 3. HTTP Client
WNCsms_demo <a href="https://developer.mbed.org/teams/Avnet/code/WNCsms_demo/">https://developer.mbed.org/teams/Avnet/code/WNCsms_demo/</a>  Also see: <a href="https://starterkit.att.com/tutorials">https://starterkit.att.com/tutorials</a> for instructions on using SMS.	This is an example program that interacts with the AT&T SIM Control Center. The program sends a fixed SMS message to the control center every 30 seconds. From within the Control Center, a user can send a SMS message back to the IoT Kit. When the program first starts, it displays the users SMS number that can be used.
<b>Original Release v1 code:</b>	
Avnet_ATT_Cellular_IOT	This is the original software used to demonstrate basic AT&T IoT Kit operation. It does not use any of the Controller or Interface API's and interacts directly with the M14A2A

## Bug Fixes

- N/A

## Known Issues

- Incoming internet data notifications are not yet implemented.
- Timing for WNC/K64F interaction requires a minimum of 500msec inter-command delay. This delay is handled internal to the Socket classes but if a user wants to interact with WNC Controller directly, this delay should be maintained.
- The SMS example requires the user to have an AT&T SIM Control Center Account.

## Accessing this release

- The libraries and example programs are all provided at: <https://developer.mbed.org/teams/Avnet/>
- Further tutorials are available at: <https://starterkit.att.com/tutorials>

## How to report problems

- It is most effective to post questions and/or problems to the [cloudconnectkits.org](https://cloudconnectkits.org) forum page. This allows others to reference and learn from already discovered issues and their resolution.