Prototyping Cloud IoT Products using FRDM-K22F and LSR Wi-Fi/BLE Shield

v2.0

On-Ramp

Technical Sessions™

by Peter Fenn,
Avnet Global Technical Marketing

Course Outline

Course Title:

Prototyping Cloud IoT Products using FRDM-K22F and Wi-Fi/BLE Shield

Course Abstract:

Based on FRDM-K22F board plus a versatile new Arduino-compatible Shield, this course discusses how to implement:

- IoT Apps using Kinetis K2 with Wi-Fi and BLE wireless modules from LS Research
- Streaming to the Cloud using TiWiConnect LIFT client on the K22 MCU
- Sensor subsystems using 3rd party Pmod-compatible boards
- Accelerated project development using pre-certified wireless modules

Agenda

Hardware Overview - FRDM-K22F and Wi-Fi & BLE Shield Wireless Modules

- Wi-Fi (TiWi-C-W) and BLE (SaBLE-x) module details

Interfaces, Indicators and Power

- Pmod and Shield Interfaces
- LED Indicators and Power

Wi-Fi Software Overview

- TiWiConnect Device Designer
- TiWiConnect LIFT Client, Agent and Server
- DeviceView, Dashboard and other Server Apps

BLE Software Overview

- SaBLE-x Developer Tool Suite (Serial-to-BLE Software)
- LSR ModuleLink BLE Mobile App

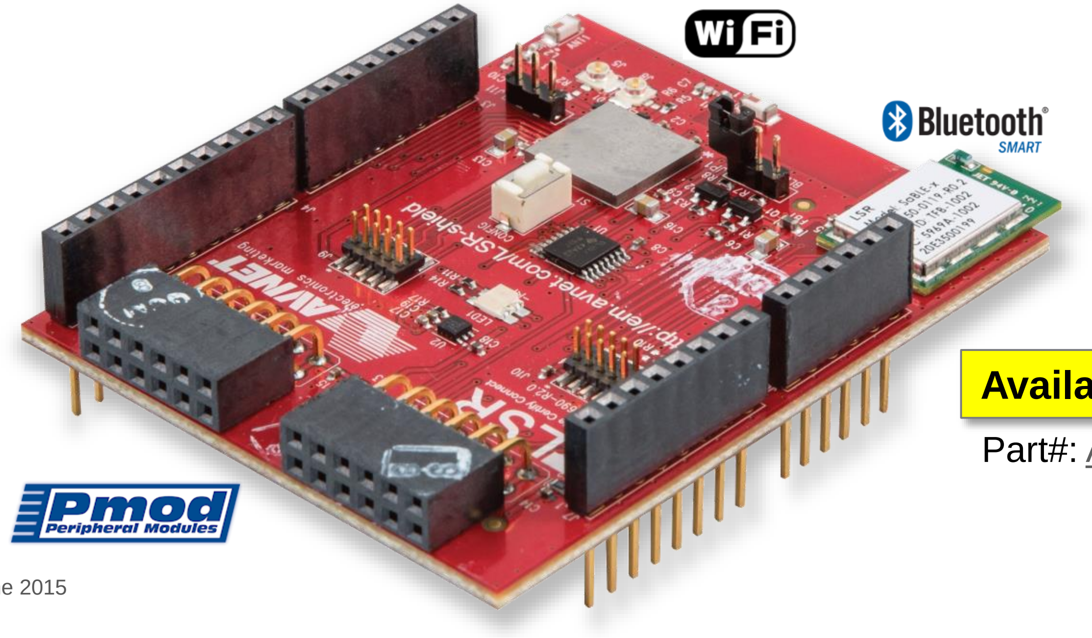
Architecting User Applications

Other Shield and Pmod-compatible Resources from Avnet

Hardware Overview

Avnet LSR Wi-Fi and BLE Shield

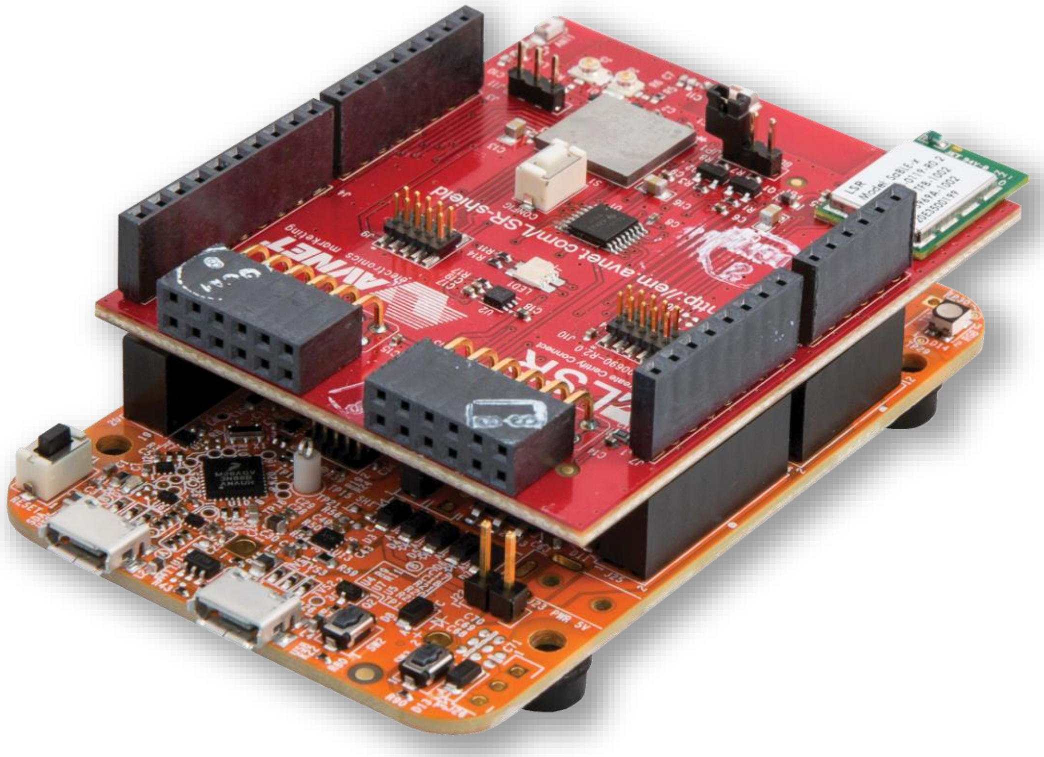
- IoT application prototyping platform
- Pre-certified wireless modules from LS Research:
  - **Ti-Wi-C-W** (802.11bgn) Wi-Fi module
  - **SaBLE-x** (Bluetooth Smart 4.1) BLE wireless module
- Simplified connection to the Cloud (**TiWiConnect LIFT** protocol)
  - Easy-to-use “**LIFT Client**” ported to **Kinetis** (FRDM-K22F reference design)
  - LIFT Agent on wireless module, LIFT Server in the cloud
- Sensor expansion using 3rd-party Pmod-compatible boards (SPI, I2C)

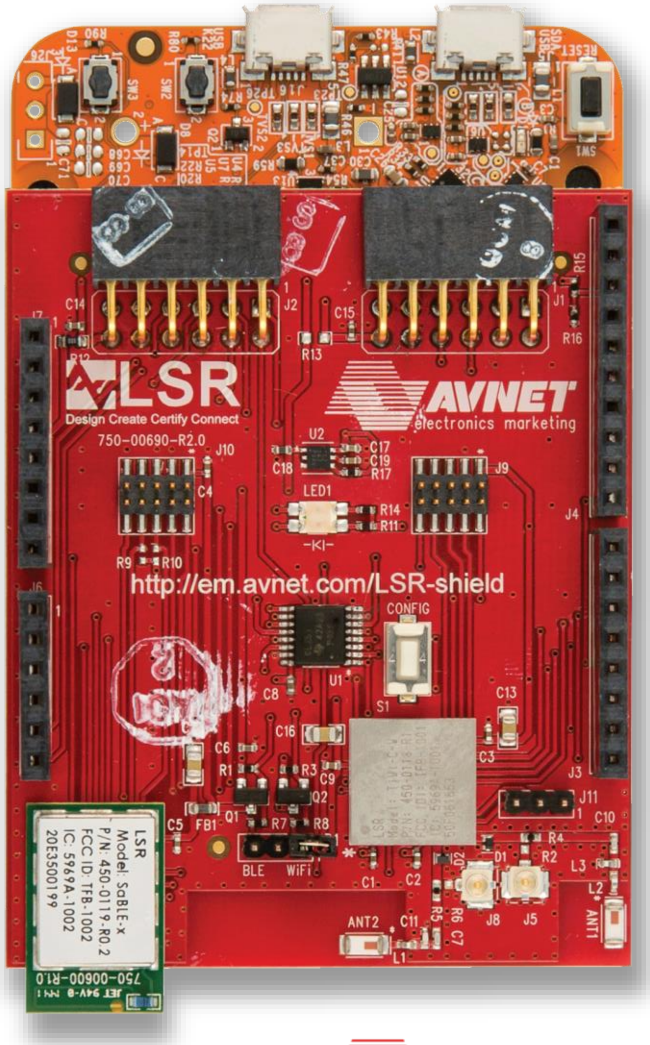


Available 3Q15

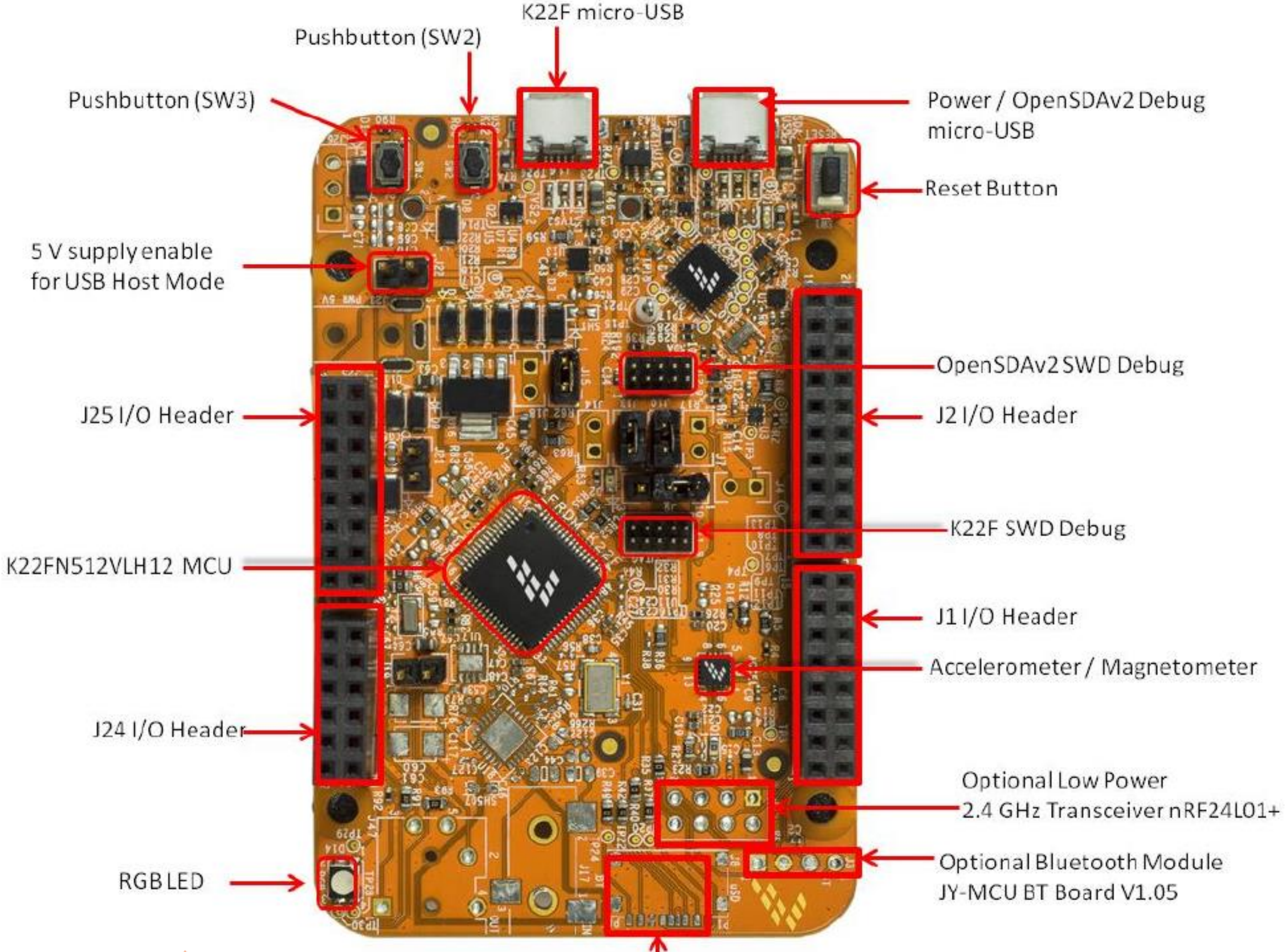
Part#: AES-TBD

Freescale FRDM-K22F Wireless Shield!





HW Platform: FRDM-K22F MCU board



K22F micro-USB

Pushbutton (SW2)

Pushbutton (SW3)

Power / OpenSDAv2 Debug micro-USB

Reset Button

5 V supply enable for USB Host Mode

OpenSDAv2 SWD Debug

J25 I/O Header

J2 I/O Header

K22F SWD Debug

K22FN512VLH12 MCU

J1 I/O Header

Accelerometer / Magnetometer

J24 I/O Header

Optional Low Power 2.4 GHz Transceiver nRF24L01+

RGB LED

Optional Bluetooth Module JY-MCU BT Board V1.05

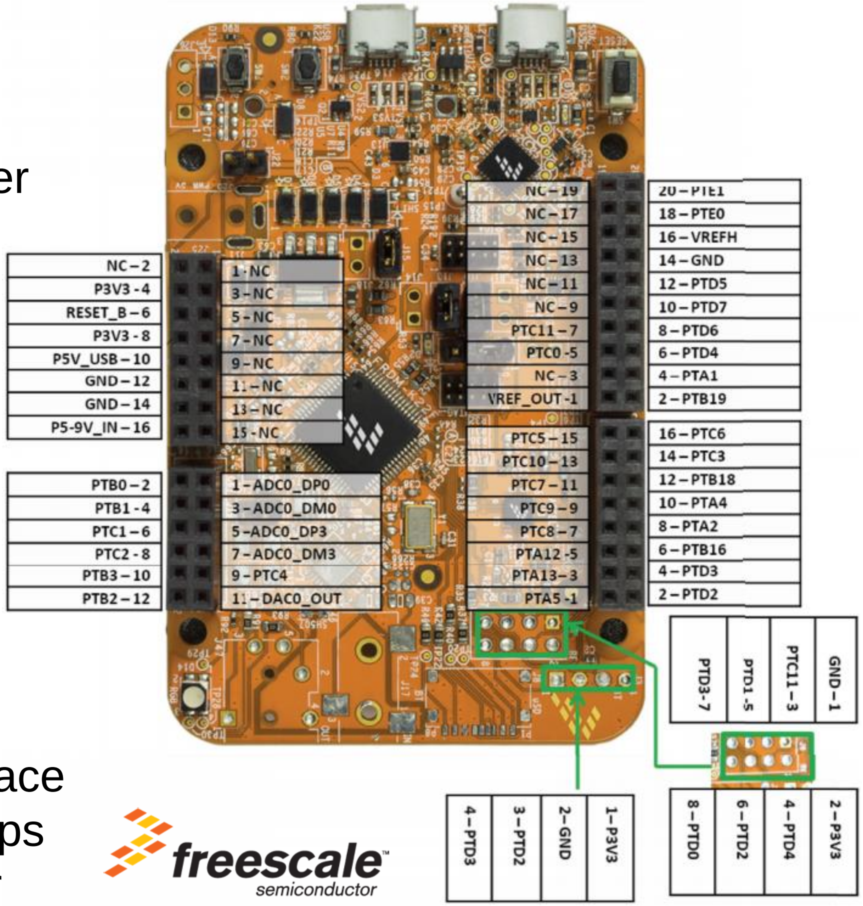
Optional micro-SD card slot

$29.00

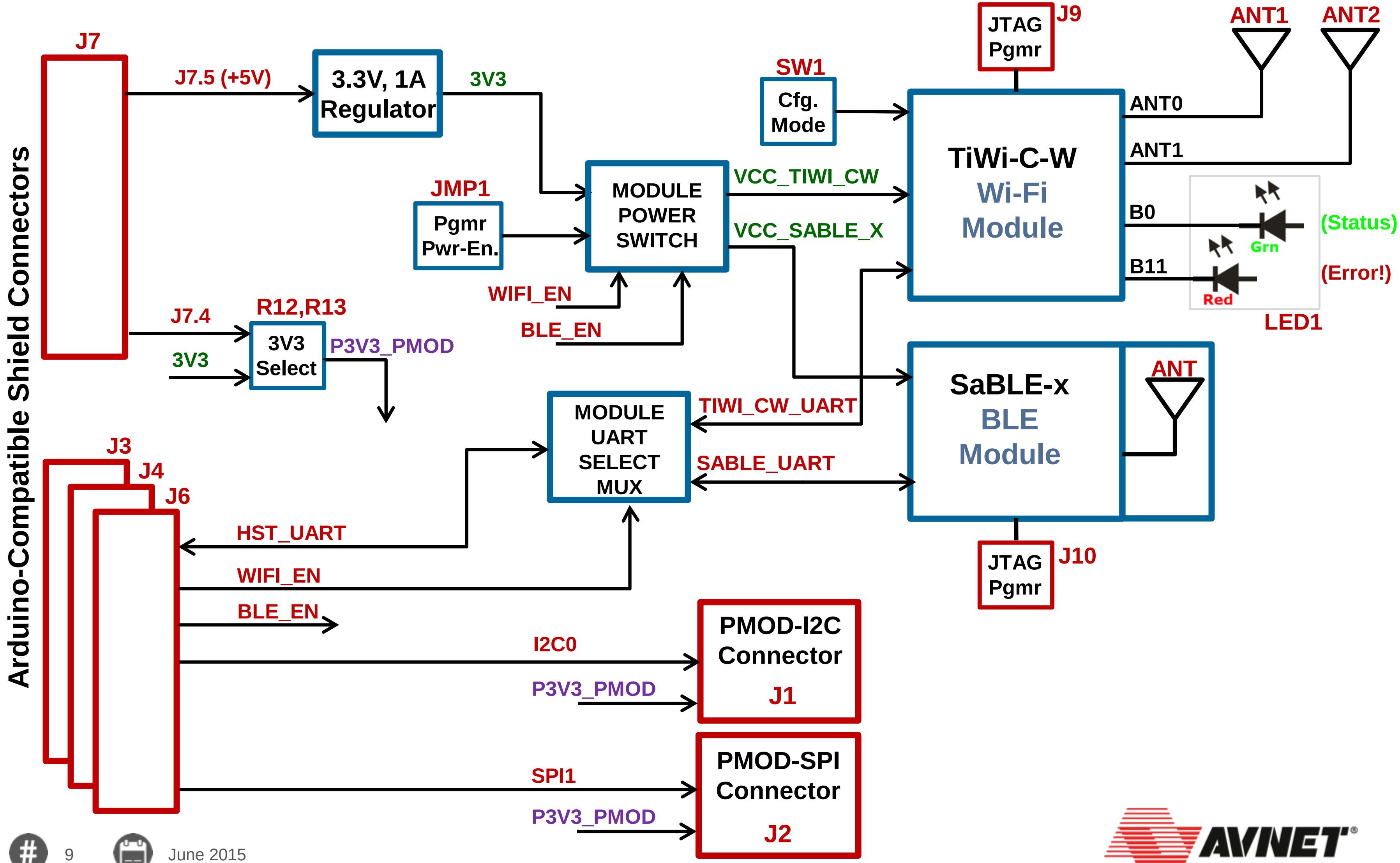
FRDM-K22F MCU board

Features:

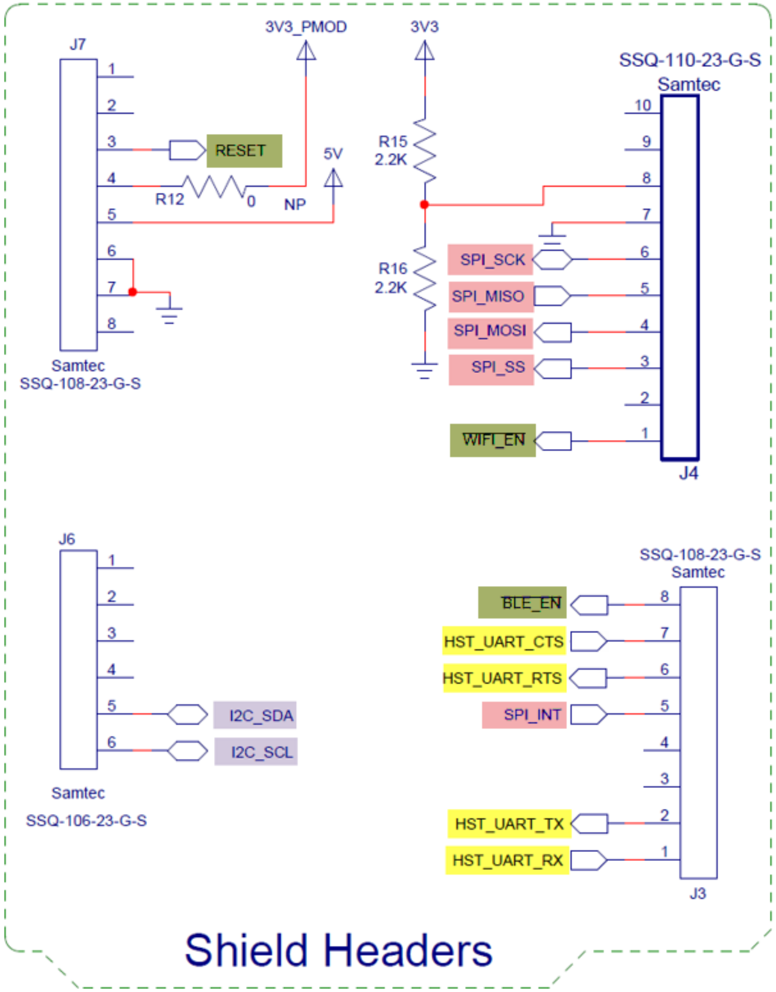
- K22 MCU (ARM Cortex-M4 @ 120 MHz, 512 KB Flash, 128 KB SRAM, FPU, DSP, crystal-less USB (host/device), 64 LQFP
- FXOS8700CQ Accelerometer + Magnetometer
- RGB LED and 2x User Push Button switches
- Flexible power supply: USB or Ext.Source
- Easy access to MCU I/O
- Optional add-ons: microSDcard (SPI), RF24L01+ (RF) and JY-MCU (Bluetooth)
- Arduino R3 form factor + pinout compatible
- OpenSDAv2.1 serial and debug interface
- Open-source hardware design
- Open-source bootloaderand firmware
- Virtual Serial port interface
- Drag 'n Drop (MSD) flash programming interface – no tool install required to evaluate demo apps
- CMSIS-DAP interface: new ARM standard for embedded debug interface



Block Diagram: LSR Wi-Fi & BLE Shield



FRDM-K22F pinout to LSR Shield

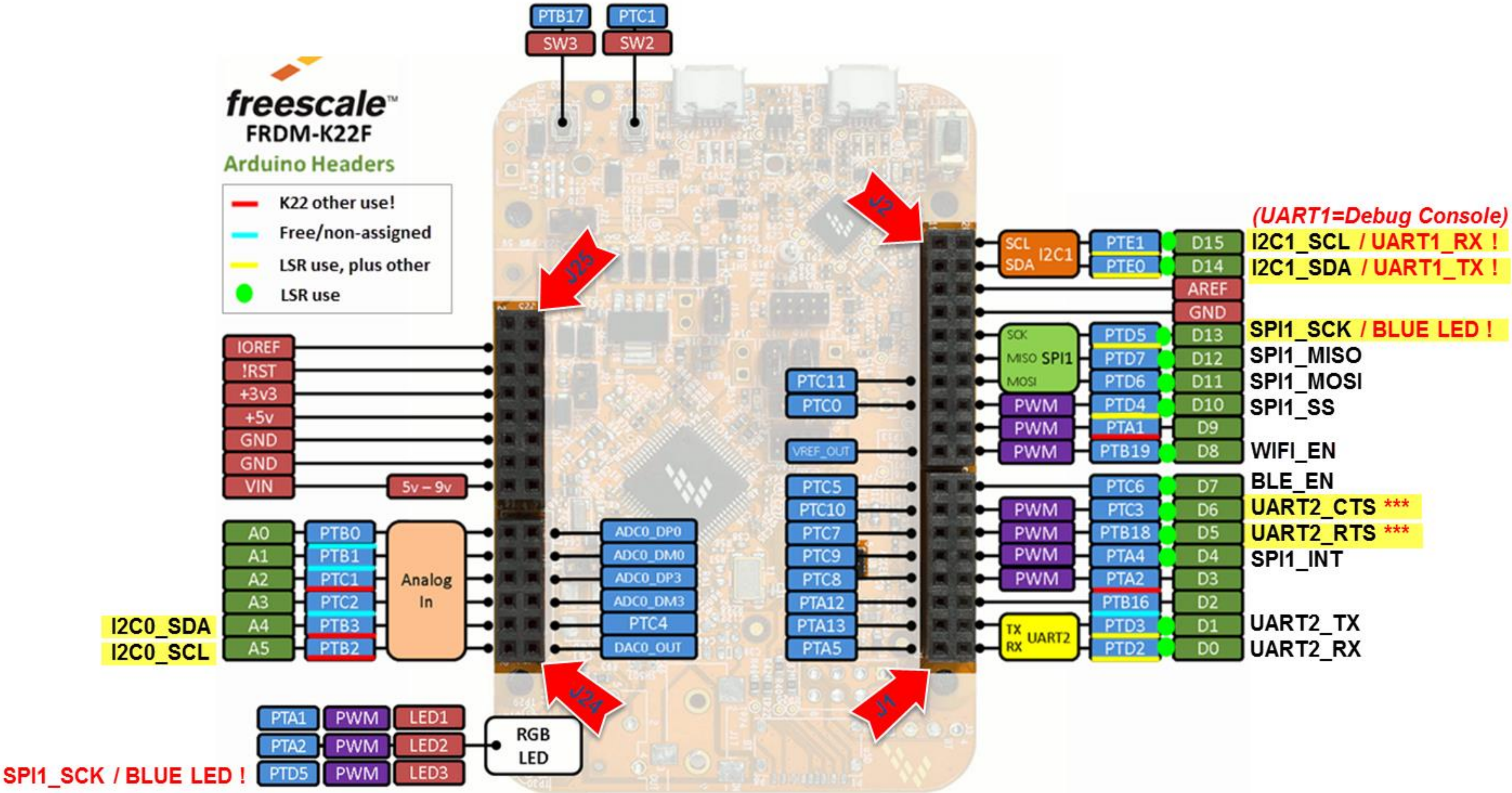


- Three communication interfaces are routed from the Freedom board to the LSR Wireless Shield:
  - **I2C0** (connects to **Pmod J1**)
  - **SPI1** (connects to **Pmod J2**)
  - **UART2** (via mux to wireless modules)
- Two GPIO signals are used to enable which wireless module is active, and which UART signals are connected to the MCU's UART2

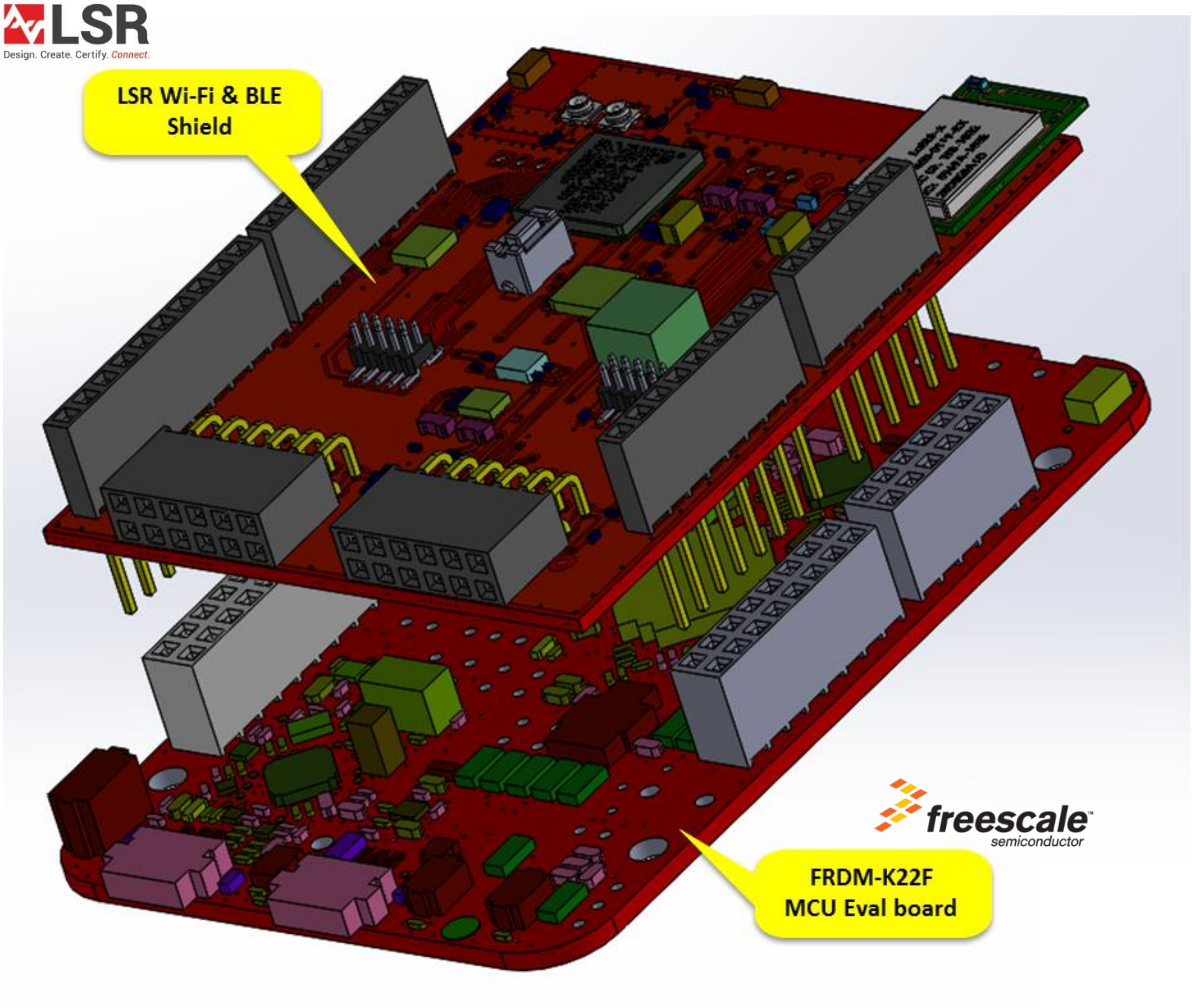
  WI-FI_EN (enables 3.3V to TiWi-C-W, also is the mux select for UART2)

  BLE_EN (enables 3.3V to SaBLE-x)
- The Serial Debug Console uses **UART1** (routed to SDA USB interface as well as Shield connector J4 pins 9 and 10)

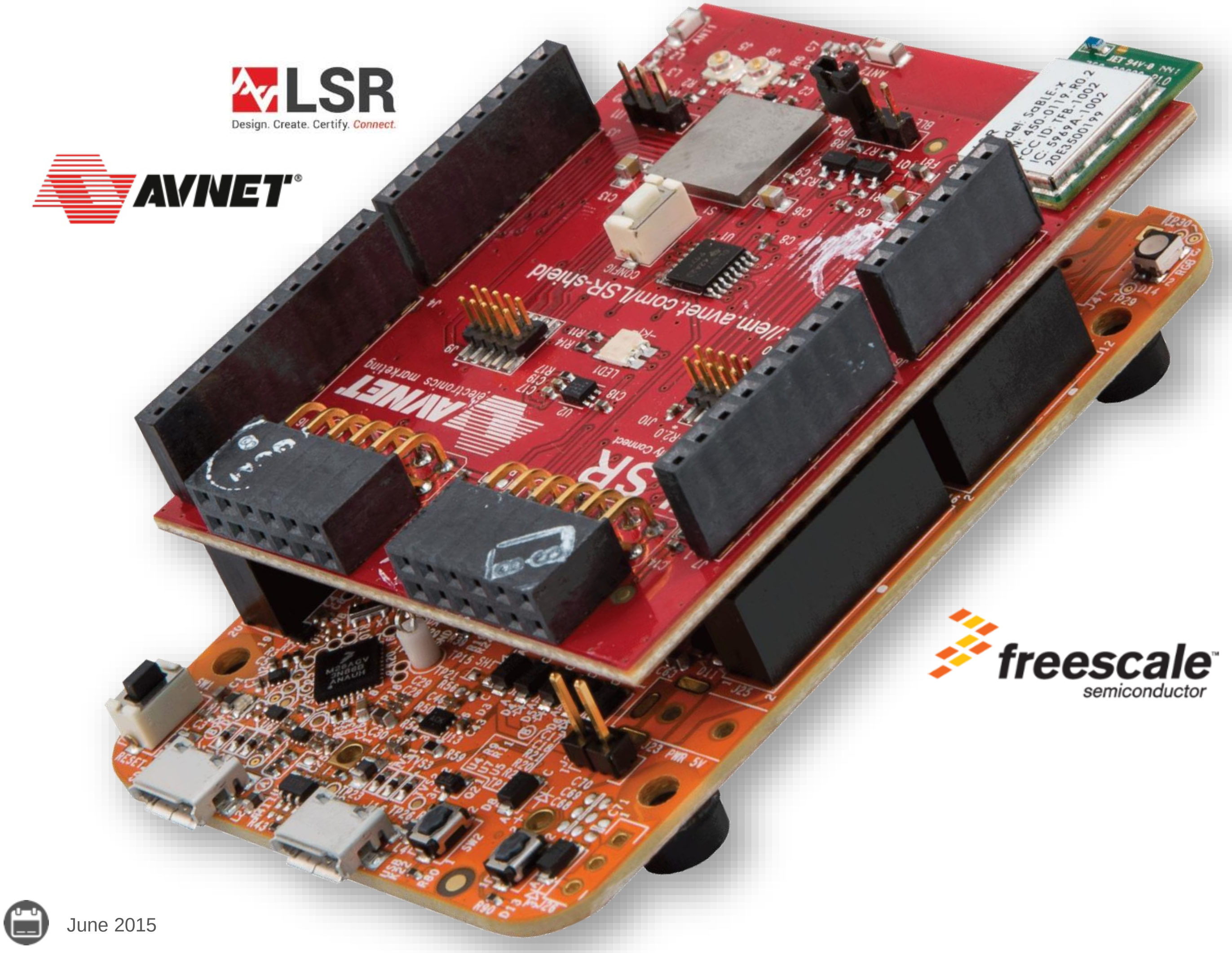
FRDM-K22F pinout to LSR Shield



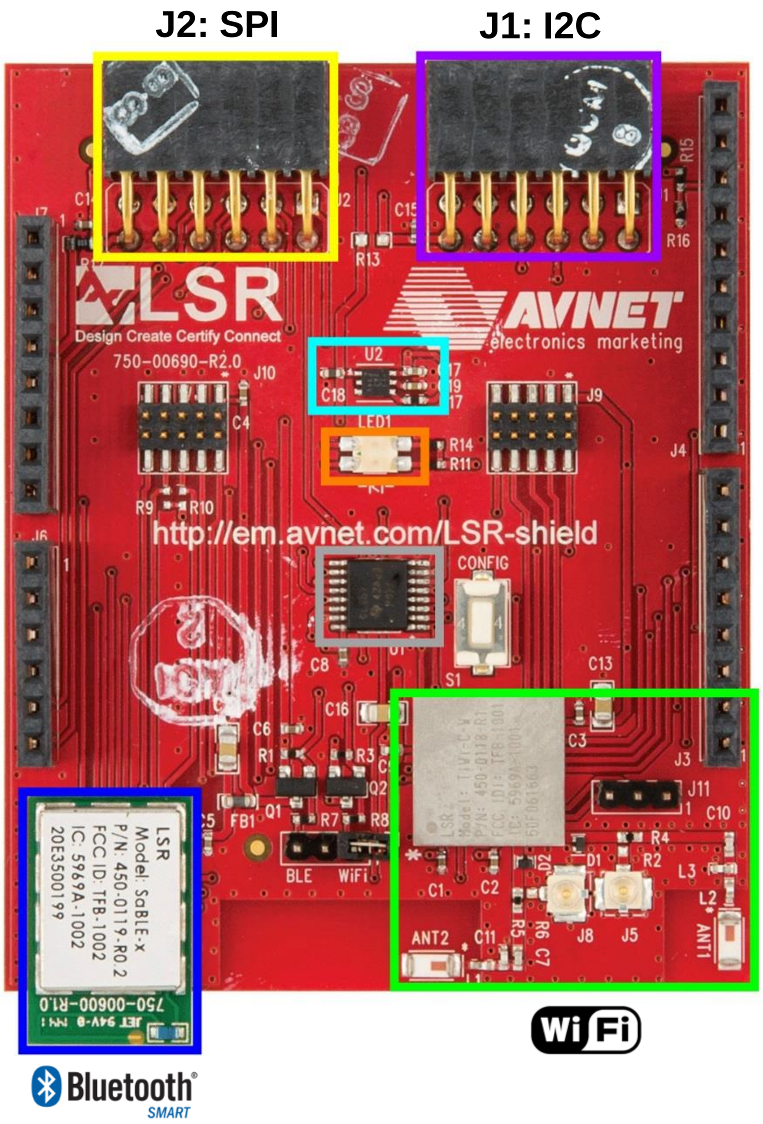
HW Platform: Concept Design…



HW Platform: Board Stack-Up



HW Platform: Shield Detail



J2: SPI pinout

#	SPI	#	SPI
1	SS	7	INT
2	MOSI	8	RST
3	MISO	9	
4	SCK	10	
5	GND	11	GND
6	VCC	12	VCC

J1: I2C pinout

#	I2C	#	I2C
1		7	
2		8	
3	SCL	9	SCL
4	SDA	10	SDA
5	GND	11	GND
6	VCC	12	VCC

- 🟪 I2C Pmod connector (J1)
- 🟨 SPI Pmod connector (J2)
- 🟦 Voltage Regulator (3.3V, 1A)
- 🟧 Wi-Fi Status LEDs (Red, Green)
- ⬜ Wireless Module Selector
- 🟦 LSR SaBLE-x BLE module
- 🟩 LSR TiWi-C-W Wi-Fi module

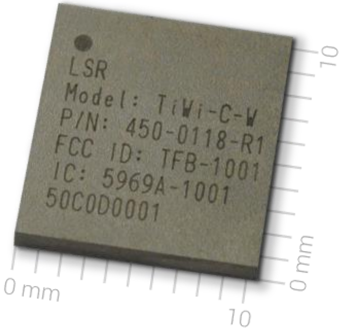
Wireless Modules

Wi-Fi Module: TiWi-C-W

LSR
Design. Create. Certify. *Connect.*

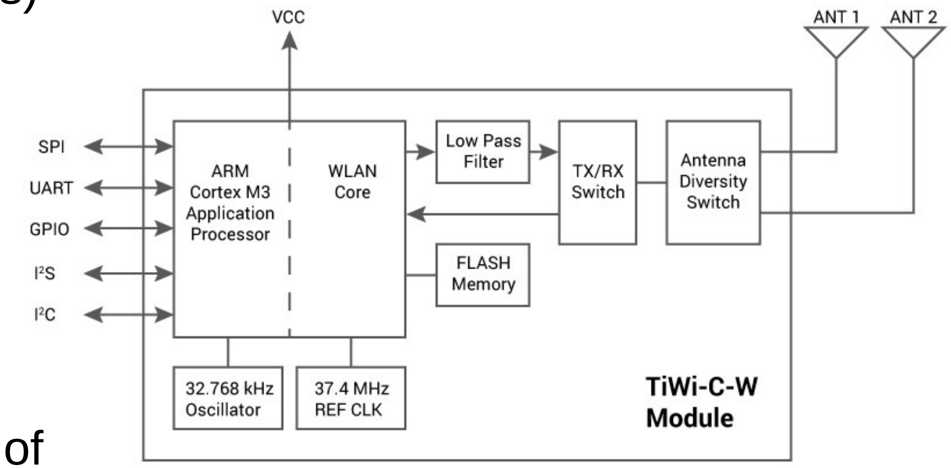
Features:

- 802.11bgn Wi-Fi module with 2 modes:
  - Hostless operation (on-board ARM Cortex-M3)
  - Host Mode via serial UART Interface, using **TiWiConnect LIFT Client** software
- Pre-Integrated **TiWiConnect LIFT Agent**
- Latest Broadcom **BCM4390** SoC device (Broadcom WICED™ Wi-Fi SDK tools)
- Embedded TCP/IP Stack
- Small footprint: 10.5 x 10.5 mm
- Supports true Antenna Diversity
- 2- and 4-layer PCB Ref. Designs
- FCC/IC/CE Module Certification
- Multiple certified antenna options (SMT chip, dipole, FlexPIFA™)
  - LSR offers in-house certification of additional antennas at little or no cost



WICED by BROADCOM

a b g Wi Fi n CERTIFIED



AVNET

BLE Module: SaBLE-x

Features:

- Built-in CC2640F128 Bluetooth Smart (BLE 4.1) SOC
- 128 kB Flash / 20 kB SRAM
- RF Output Power: +5 dBm
- RF Receive Sensitivity: -96 dBm
- Size: 11.6mm x 17.9mm x 2.4mm
- Operating Voltage: 1.8V to 3.8V
- Operating Temperature: -40 to +85C
- 9.1 mA Transmit Mode (+5 dBm), 6.1 mA Receive Mode
- 1µA Standby (SRAM/CPU retention, RTC running) with quick 100µs start up
- 200nA Shutdown
- 61µA/MHz Active CPU Current
- Drivers, BLE Controller, IEEE 802.15.4 MAC and bootloader in ROM
- Flexible peripheral set
- On board 32 kHZ and 24 MHz Crystals.
- Worldwide Certifications: FCC, IC, ETSI, Giteki, C-Tick (all Pending)
- REACH and RoHS compliant

BLE Module: SaBLE-x

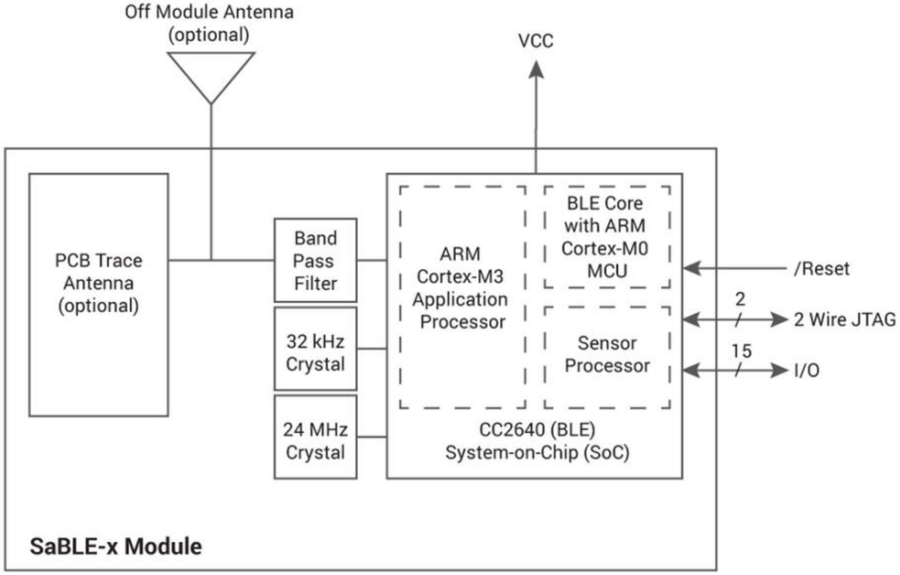
SaBLE-x Module Performance

Specification	Value
Footprint	11.6 mm x 17.9 mm x 2.3 mm
Operating Temp	-40 to +85° C
Operating Voltage	1.8 V to 3.8 V
Output Power	+5 dBm
Receive Sensitivity	-97 dBm, @ 0.1% BER
Average Power Consumption	<10 uA for 1 second BLE connection interval

For full specifications on SaBLE-x, please refer to datasheet.

Technical Details
Bluetooth 4.1 compliant / *Bluetooth* 4.2 capable (future)
Integrated ARM Cortex-M3 MCU, dedicated Cortex-M0 MCU for RF core, and dedicated Sensor Processor Engine
128 kB FLASH and 20 kB SRAM on-module memory

2x signal range, ~ 1/3 the power vs previous gen. BLE module!


Off Module Antenna (optional)
VCC
PCB Trace Antenna (optional)
Band Pass Filter
32 kHz Crystal
24 MHz Crystal
ARM Cortex-M3 Application Processor
BLE Core with ARM Cortex-M0 MCU
Sensor Processor
CC2640 (BLE) System-on-Chip (SoC)
/Reset
2
2 Wire JTAG
15
I/O
SaBLE-x Module

What Differentiates SaBLE-x From Other BLE Modules?

- 1st commercially available module based on new TI **CC2640 BLE SoC**
  - ARM Cortex-M3 for host applications, separate Cortex-M0 for RF core.
  - Dedicated Sensor Processor, collects sensor data while main MCU remains asleep
  - On-board FLASH memory, BLE stack, high-speed and low-speed clocks
- Unmatched RF performance, **2x signal range** and nearly **2/3 less average power** compared to previous generation BLE module (TiWi-uB1)
- LSR’s **Developer Tool Suite** simplifies BLE integration effort:
  - **Serial-to-BLE** API Library with C source code for both module and host
  - **API Command Manager** to quickly add or edit commands for Serial-to-BLE library
  - **PC-as-Host** Test Tool for convenient testing of serial commands
  - **Wired Bootloader** utility (USB-based)
  - **TiWiConnect Cloud Agent** firmware for TiWiConnect IoT platform (Q4 2015)
- Unmatched breadth of **country certification** coverage:
  - FCC/ IC (certified) and CE / C-Tick / Giteki (pending)
  - Multiple certified antenna options:
    - ○ On-board Trace, Dipole, FlexPIFA™, and FlexNotch™
    - ○ In-house certification of additional antennas at little/no cost to customer

Interfaces, Indicators and Power

Multiple Interfaces Available for Sensors

Multiple interfaces on the Shield are available for attachment of sensors:

- **Pmod J1** (I2C0)
- **Pmod J2** (SPI1)
- **Shield** pass-through connectors (SPI, I2C, GPIO…)
- **BLE** wireless (any remote sensor fitted with BLE radio)

FTF-2015 reference design only uses Pmod J1 and Pmod J2 interfaces.

Pmod J1: I2C Interface

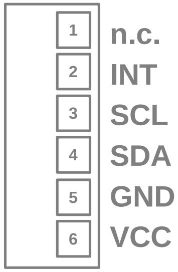
MAX44000 Ambient Light, IR Proximity

I2C based Pmod-compatible Sensor boards are fitted to the Shield via the Pmod J1 connector

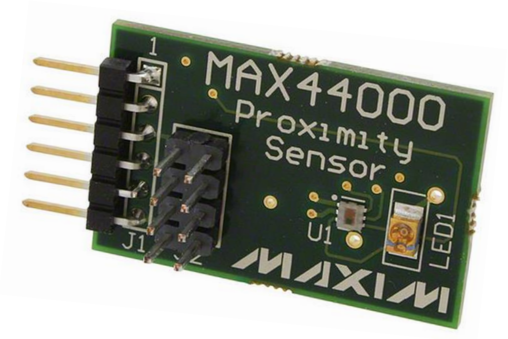
MAX44000 (Maxim Integrated)
Ambient Light and IR Proximity sensor

Pmod1: I2C sensors

#	I2C	#	I2C
1		7	
2		8	
3	SCL	9	SCL
4	SDA	10	SDA
5	GND	11	GND
6	VCC	12	VCC



1 n.c.
2 INT
3 SCL
4 SDA
5 GND
6 VCC



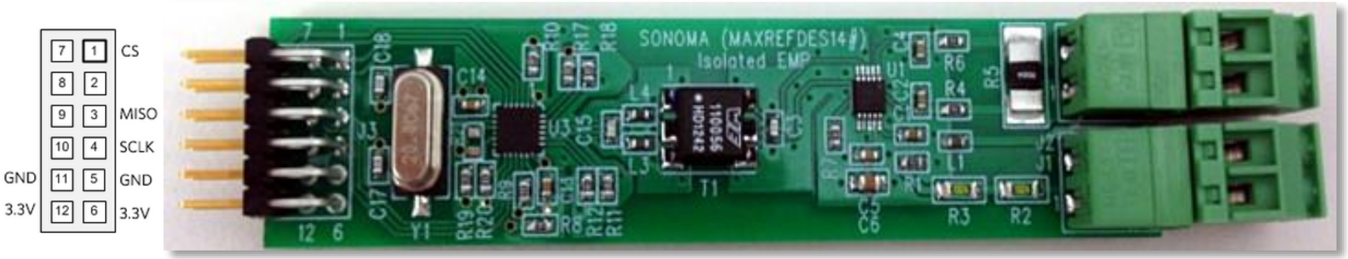
Pmod J2: SPI Interface
MAXREFDES#14 Energy Measurement

A single, extended length Pmod-compatible board, configured to operate in SPI mode, fitted to the Shield via Pmod J2 connector:

MAXREFDES14# (Maxim Integrated)
Isolated Energy Measurement Subsystem
(for AC power measurements)

Pmod2: SPI sensors

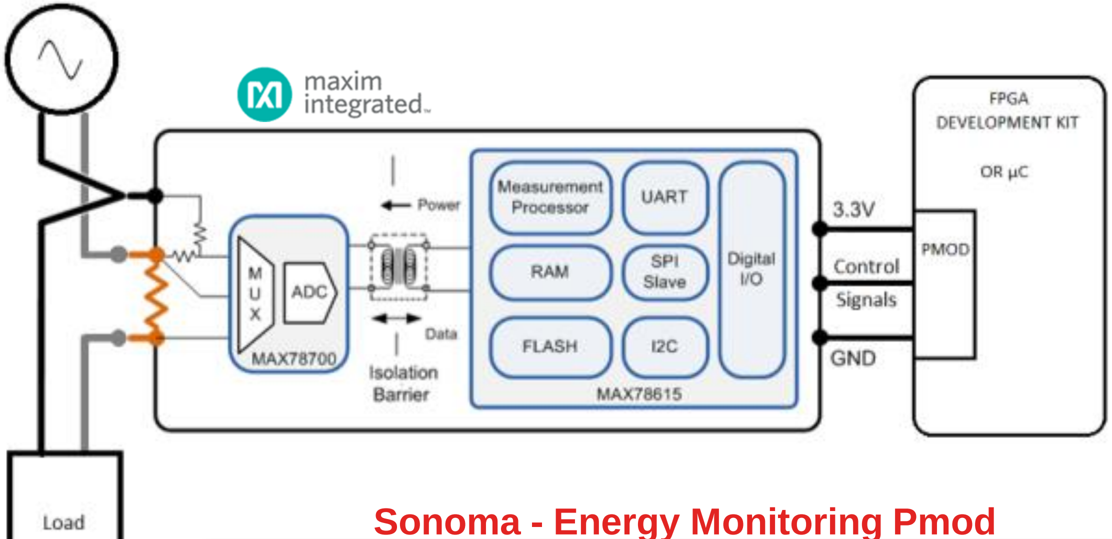
#	SPI	#	SPI
1	SS	7	INT
2	MOSI	8	RST
3	MISO	9	
4	SCK	10	
5	GND	11	GND
6	VCC	12	VCC



MAXREFDES14# (Sonoma)

Isolated Energy Measurement Subsystem Reference Design

- Accurate: 3% without calibration
- Cost optimized isolation
- Small size for immediate system implementation



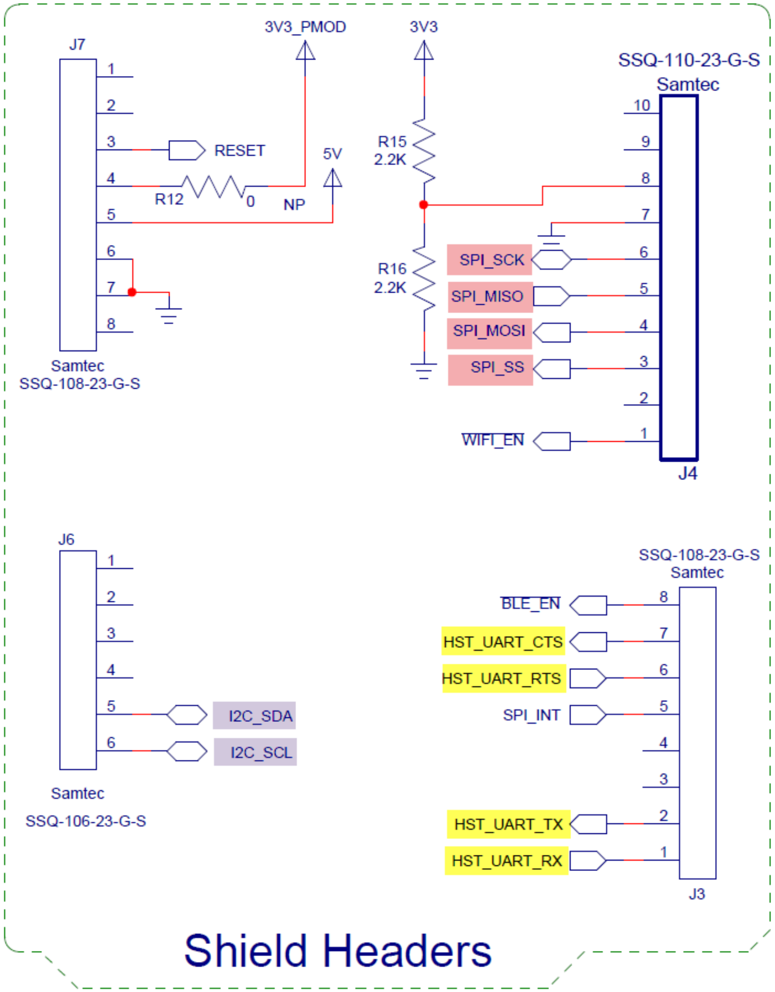
Dimmable AC 120V Desk-Lamp

Sonoma - Energy Monitoring Pmod

www.MaximIntegrated.com/Sonoma

LSR Shield: Arduino-Compatible Interface

- For an application where signals on the Arduino-compatible connectors are not used by LSR Wi-Fi/BLE Shield, these may be used by another stacked Shield.
- I2C bus and unused GPIO are always available for use by another Shield
- The I2C, SPI and UART interfaces and two GPIO enable signals to this board are highlighted here…



LSR Shield: Wi-Fi LED Status Indicators

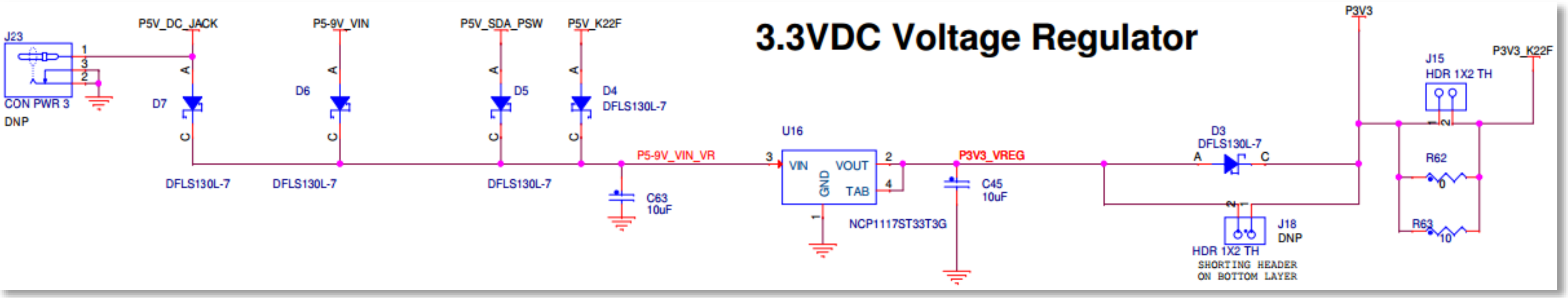
Error LED (Red)	Status LED (Green)	Description
	Green blinks rapidly after startup	Wi-Fi module has successfully joined the configured Wi-Fi network and is ready to communicate
Red blinks rapidly after startup		Wi-Fi module has entered Configuration Mode and is ready to be accessed as an access point for configuration
Red + Green blink simultaneously		Wi-Fi module failed to join the configured network and after 10 seconds will revert to Configuration Mode
	Green LED activity during operation	Wi-Fi module is communicating with the server under normal operation (blinks/pauses of Green LED indicate packets being sent over network)
Red blinks during operation		Wi-Fi module failed to send a message to the server. The network may be down or signal strength too low

LSR Shield: Power Supplies

FRDM-K22F board: +3.3V, 1A LDO (OnSemi **NCP1117**)

Supplies +3.3V power to:

- FRDM-K22F board, plus
- any Pmod boards fitted to LSR Shield and/or additional Shield if fitted



LSR Wireless Shield: +3.3V, 1A LDO (TI **TPS73733**)

Supplies +3.3V power to:

- LSR Wi-Fi and BLE wireless modules, plus
- Selector mux,
- JTAG debug connectors

Wi-Fi Software

TiWiConnect LIFT software
Innovative alternative to AT Command Sets

LSR.com

Wi Fi

Simplicity and Efficiency

LSR's TiWiConnect end-to-end IoT solution is far more than just an IoT cloud provider! By designing the solution from scratch to work seamlessly together, TiWiConnect LIFT's data architecture and software tools abstracts out the complexity of Wi-Fi development and accelerates development time dramatically.

- TiWiConnect LIFT facilitates **host-to-cloud** communication, not just host-to-module like an AT Command Set
- Use of a data format familiar to the cloud (JSON) accelerates development effort.
- Better development team collaboration is achieved as developers of cloud-server and related apps are already familiar with JSON

TiWiConnect™
Module | Cloud | App

Flexibility and Freedom

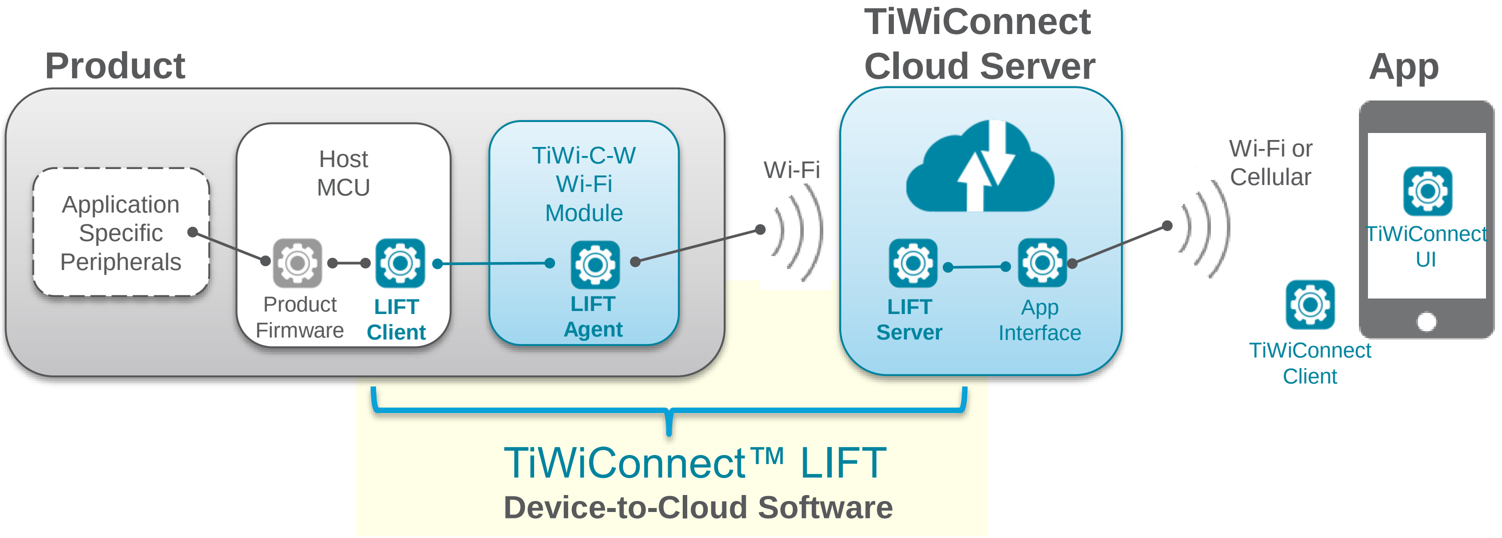
Unlike other IoT Platforms, TiWiConnect was architected to provide flexibility! Designers can now choose to implement the cloud-server aspect of their system based on time-to-market and simplicity vs. cost:

1	LSR implements and hosts Server to User's custom requirements	Subscription Service
2	LSR implements to User's custom requirements but the Server is hosted by User.	1-time Licensing/NRE
3	User implements and hosts, utilizing LSR-provided LIFT Server source code examples (NodeJS, Python)	No additional fees beyond module h/w

Transition from LSR-hosted model to self-hosted model may be done at a future date (ie. rapid time-to-market without up-front costs and commitment)

AVNET®

Overview of TiWiConnect LIFT
LSR's Device-to-Cloud software



TiWiConnect LIFT = LIFT Client + LIFT Agent + LIFT Server

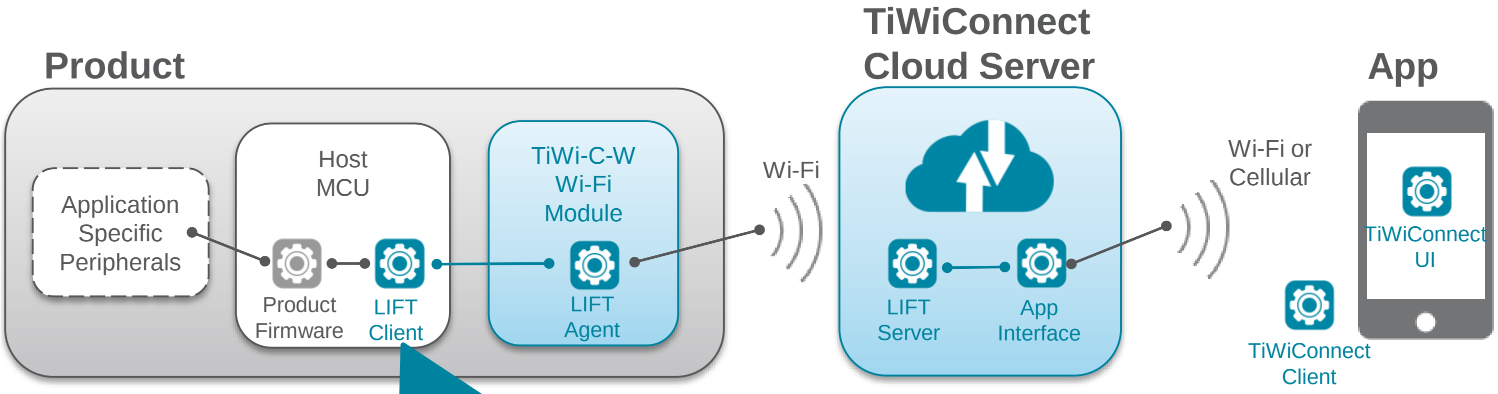
- LIFT Client, Agent and Server pre-engineered software components implement the 'data tunnel' solution between on-board MCU and the TiWiConnect cloud.

TiWiConnect LIFT simplifies your implementation!

(reduced time & complexity vs traditional WiFi implementations)

- No need to learn and write 'AT' Commands
- No dealing with raw TCP socket connections
- Avoid the learning curve with implementing MQTT or other Restful API's

Overview of TiWiConnect LIFT Client



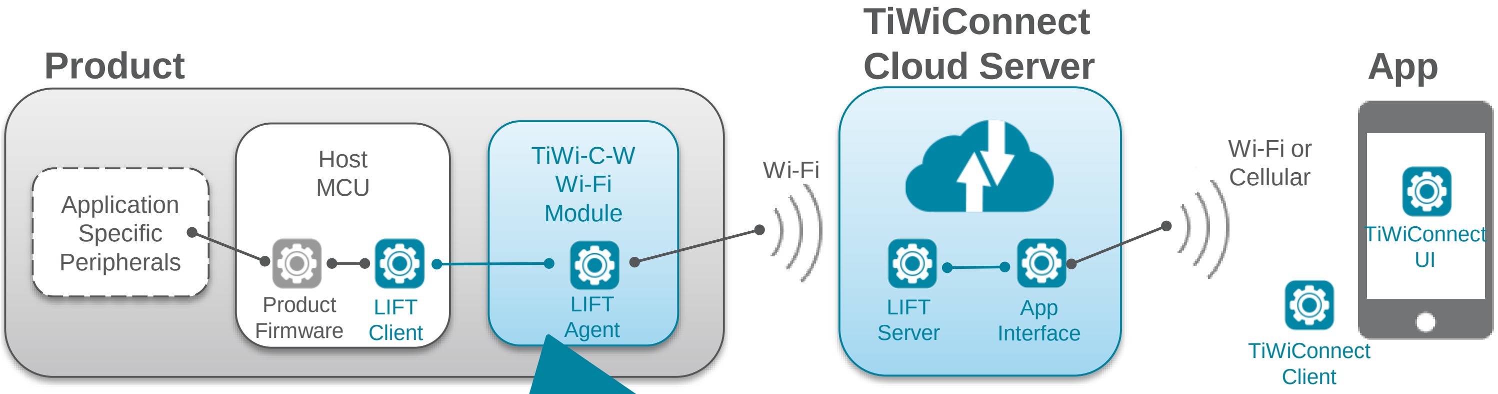
WHAT IS LIFT CLIENT?

- **LIFT Client** refers to the LSR-provided source code to be run on the host microcontroller (MCU).
- Creates a ‘tunnel’ for easy transfer of data between the MCU and a TiWiConnect cloud-server via the Wi-Fi module.
- This code is **auto-generated** by the **LIFT Device Designer** web-based tool

BENEFITS OF LIFT CLIENT

- Auto-generates small footprint source code designed to run on any MCU with a C compiler
- Communicates over a standard serial port interface
- Allows developer to continue working with IDE development tools they are already familiar with
- No proprietary AT Command Set to learn and implement

Overview of TiWiConnect LIFT Agent (Firmware on TiWi-C-W Module)



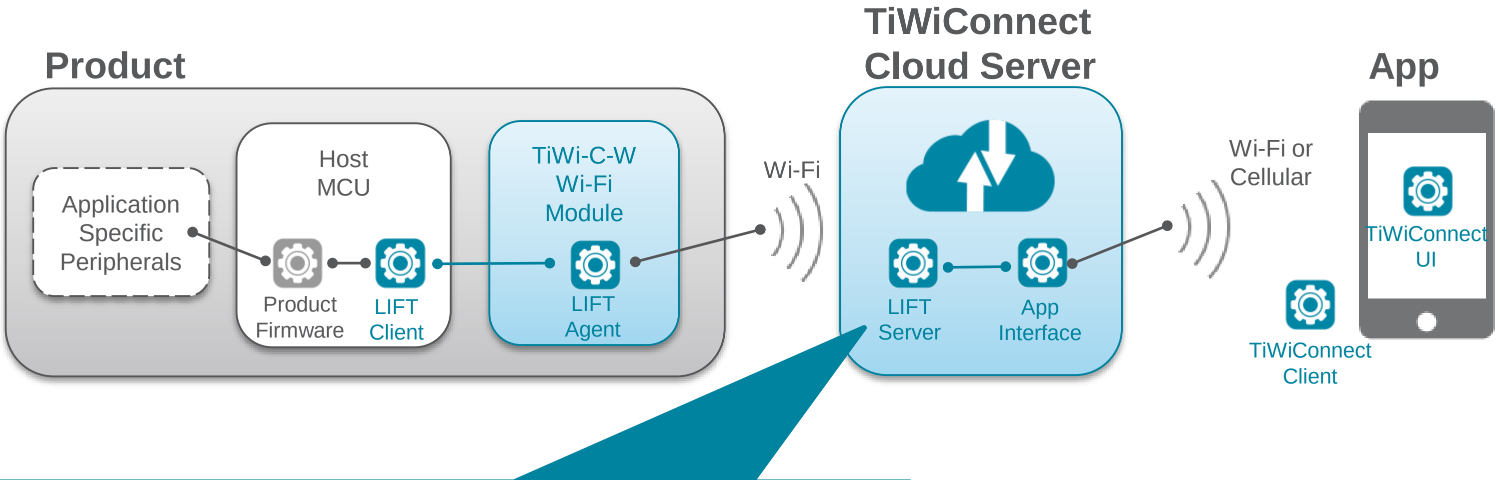
WHAT IS LIFT AGENT?

- **LIFT Agent** is the LSR-provided embedded software that runs on the TiWi-C-W module
- Provides 'transport mechanism' to pass data between LIFT Client on Host MCU (via UART serial port) and LIFT Server in the cloud (via internet).
- LIFT Agent also includes the network provisioning feature for the module (Soft AP).
- LIFT Agent is part of the firmware provided by LSR that customer would flash onto the TiWi-C-W module

BENEFITS OF LIFT AGENT

- Easy data tunneling to the cloud
- Familiar protocols for easy interoperability
- 'Soft AP' Network commissioning features

Overview of TiWiConnect LIFT Server



WHAT IS LIFT SERVER?

- **LIFT Server** is the LSR-designed server-side interface running in the TiWiConnect cloud, receiving/sending data to the device through the simple ‘data tunnel.’
- In cases where customer wishes to host the cloud themselves, LSR would provide the LIFT Server in the form of source code examples (e.g. Python, NodeJS)

BENEFITS OF LIFT SERVER

- Provides web developers the tools they need to send / receive tunneled data from a LIFT Client enabled device to the web
- Example code provided in several familiar web development languages (Python, NodeJS, and more coming)

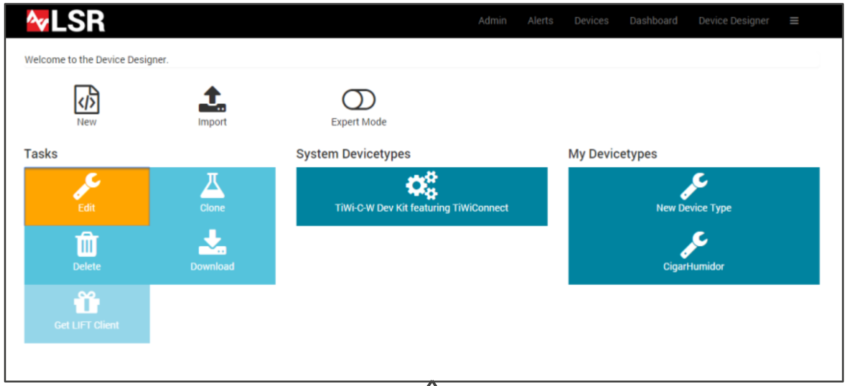
What is “LIFT Device Designer”?

- Web-based drag-and-drop tool to quickly define your product’s unique **Device Type** in terms of **Attributes** and **Actions**

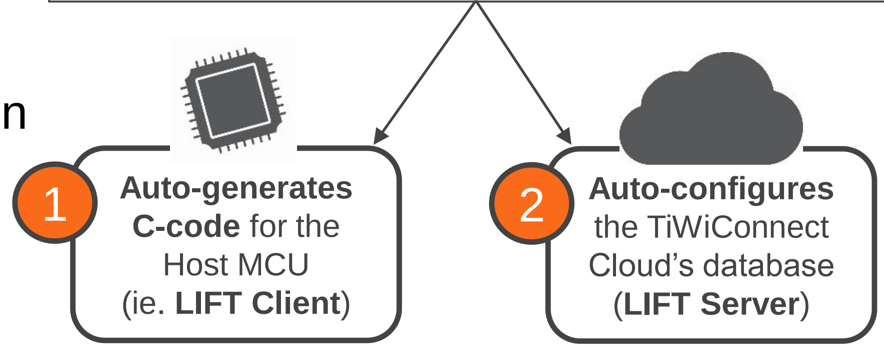
- Eliminates need to program directly in JSON

- Dramatically simplifies 2 key tasks for creating a data tunnel between MCU and Cloud…

- Minimizes the learning curve and provides code that can integrated into user’s IDE, so they can continue developing in the software tools that they’re comfortable with!



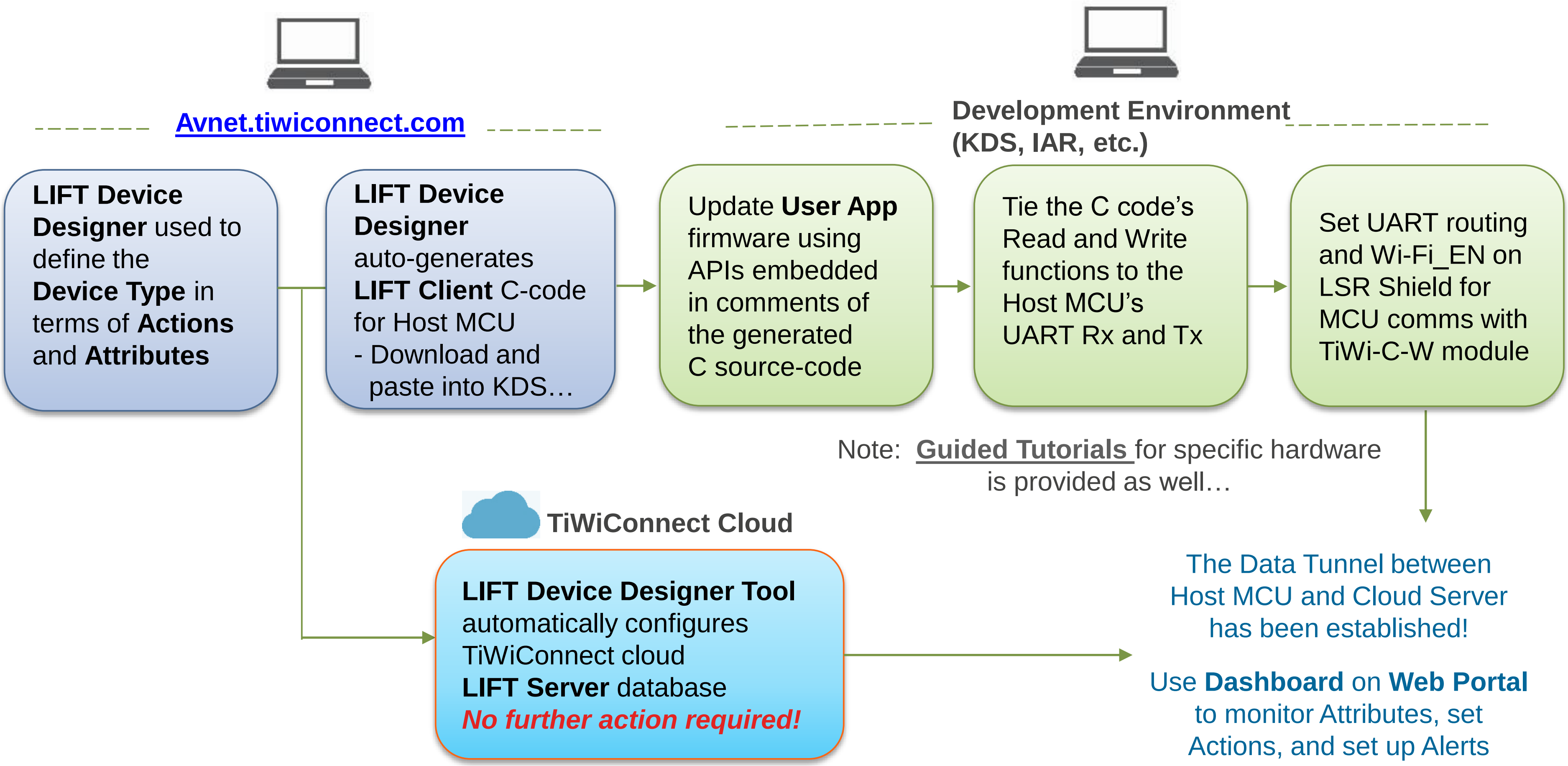
Welcome to the Device Designer.



1 **Auto-generates C-code** for the Host MCU (ie. **LIFT Client**)

2 **Auto-configures** the TiWiConnect Cloud’s database (**LIFT Server**)

TiWiConnect
LIFT Device Designer



Avnet.tiwiconnect.com

LIFT Device Designer used to define the **Device Type** in terms of **Actions** and **Attributes**

LIFT Device Designer auto-generates **LIFT Client** C-code for Host MCU
- Download and paste into KDS…

Development Environment (KDS, IAR, etc.)

Update **User App** firmware using APIs embedded in comments of the generated C source-code

Tie the C code's Read and Write functions to the Host MCU's UART Rx and Tx

Set UART routing and Wi-Fi_EN on LSR Shield for MCU comms with TiWi-C-W module

Note: Guided Tutorials for specific hardware is provided as well…

TiWiConnect Cloud

LIFT Device Designer Tool automatically configures TiWiConnect cloud **LIFT Server** database
No further action required!

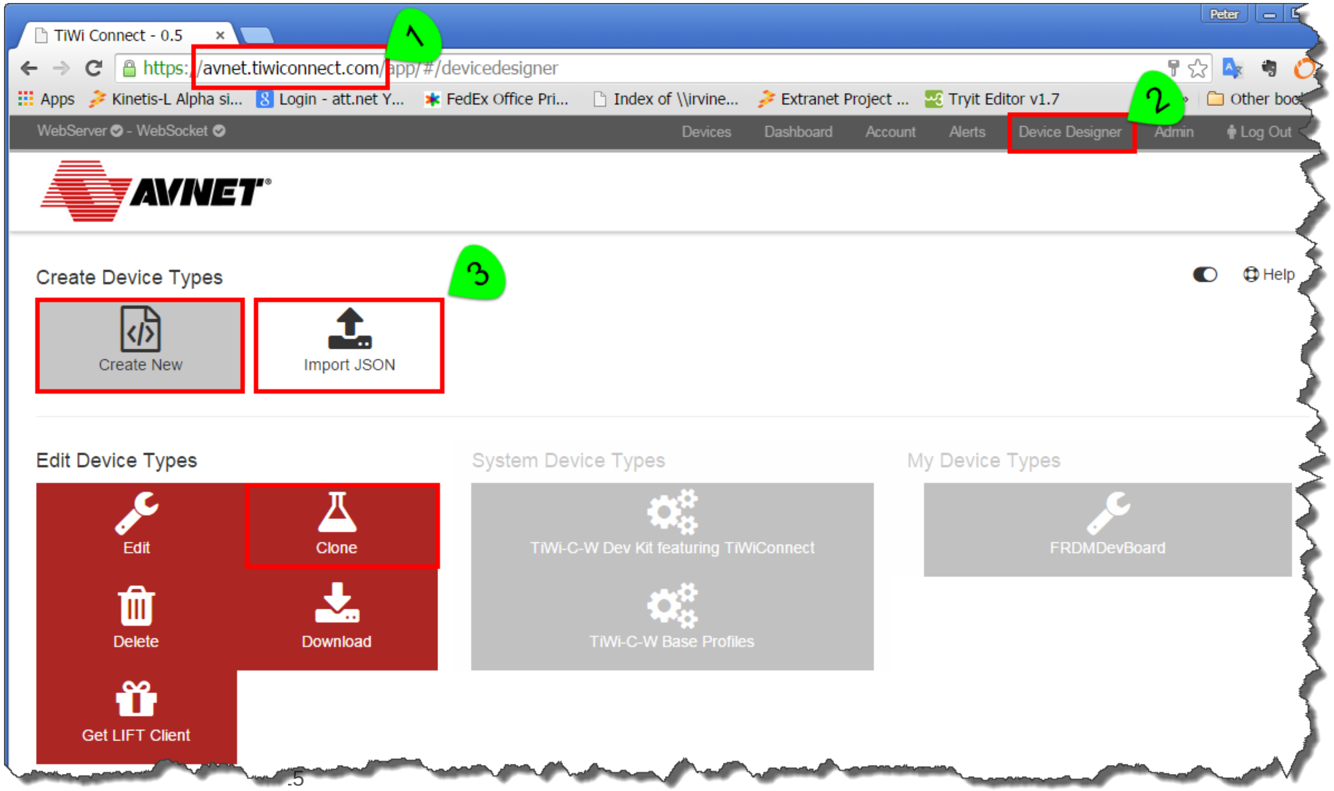
The Data Tunnel between Host MCU and Cloud Server has been established!

Use **Dashboard** on **Web Portal** to monitor Attributes, set Actions, and set up Alerts

TiWiConnect
LIFT Device Designer

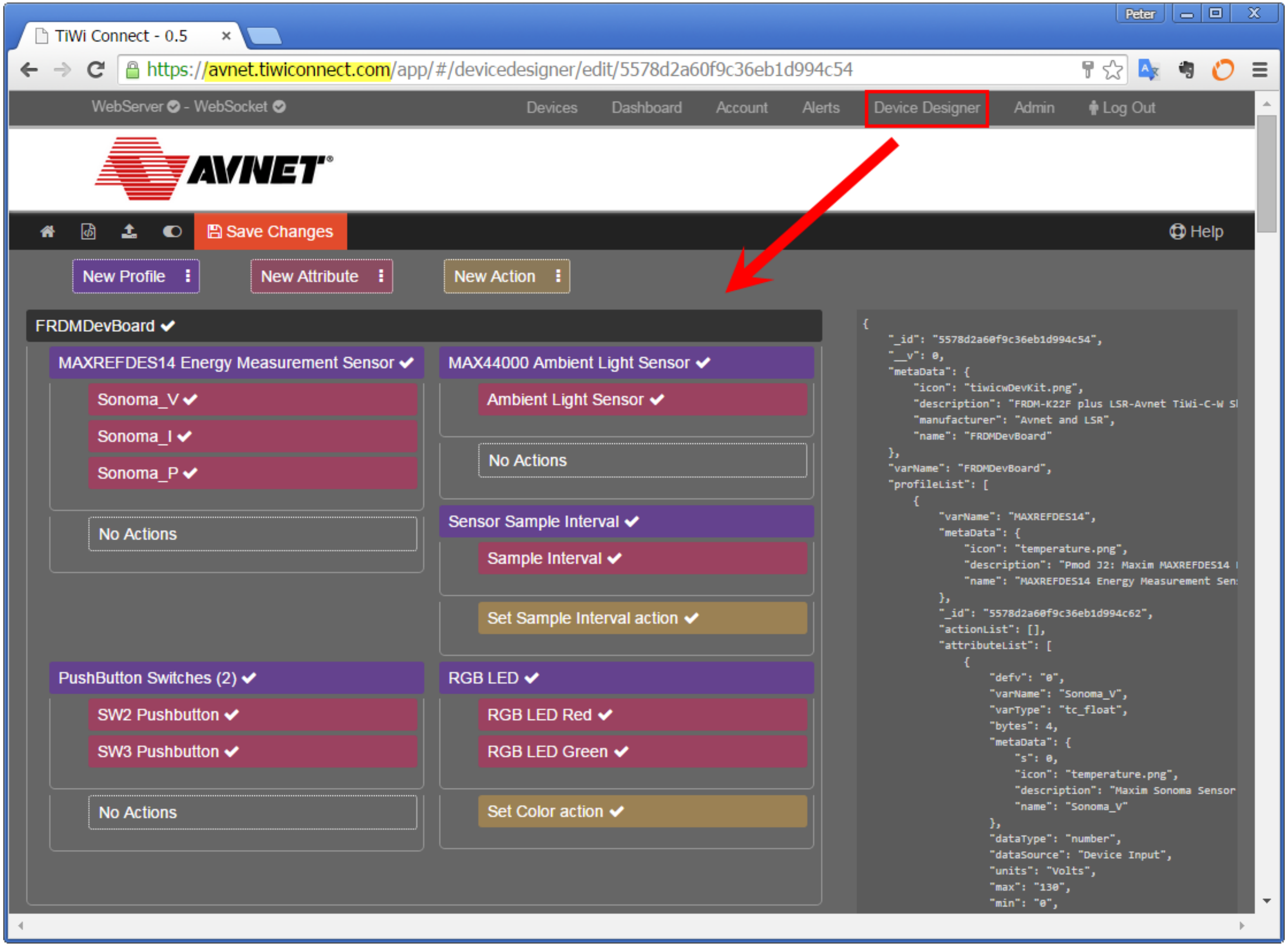
Device Designer is launched from the **Avnet TiWiConnect** web portal
https://avnet.tiwiconnect.com

Definition of your custom “**Device Type**” can be started in 1 of 3 ways:
Create New, **Import JSON** or **Clone** (of an existing Device Type)



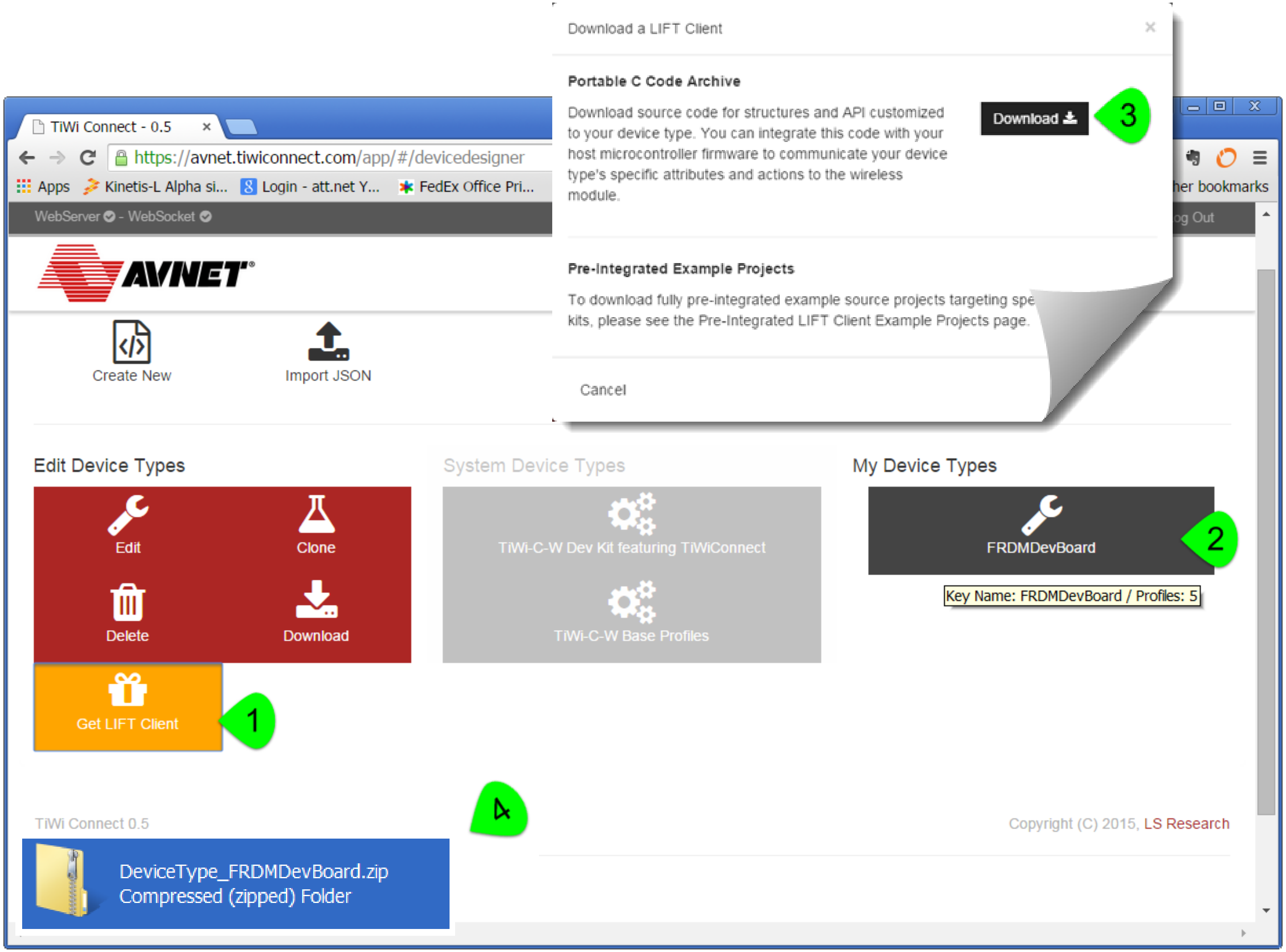
TiWiConnect: Defining your Device Type

Profile, Attributes and Actions…



```
{
    "_id": "5578d2a60f9c36eb1d994c54",
    "__v": 0,
    "metaData": {
        "icon": "tiwicwDevKit.png",
        "description": "FRDM-K22F plus LSR-Avnet TiWi-C-W S
        "manufacturer": "Avnet and LSR",
        "name": "FRDMDevBoard"
    },
    "varName": "FRDMDevBoard",
    "profileList": [
        {
            "varName": "MAXREFDES14",
            "metaData": {
                "icon": "temperature.png",
                "description": "Pmod J2: Maxim MAXREFDES14
                "name": "MAXREFDES14 Energy Measurement Sen
            },
            "_id": "5578d2a60f9c36eb1d994c62",
            "actionList": [],
            "attributeList": [
                {
                    "defv": "0",
                    "varName": "Sonoma_V",
                    "varType": "tc_float",
                    "bytes": 4,
                    "metaData": {
                        "s": 0,
                        "icon": "temperature.png",
                        "description": "Maxim Sonoma Sensor
                        "name": "Sonoma_V"
                    },
                    "dataType": "number",
                    "dataSource": "Device Input",
                    "units": "Volts",
                    "max": "130",
                    "min": "0",
```

Generating the LIFT Client files
(to add to Freescale KDS project…)



Download a LIFT Client

Portable C Code Archive

Download source code for structures and API customized to your device type. You can integrate this code with your host microcontroller firmware to communicate your device type's specific attributes and actions to the wireless module.

Pre-Integrated Example Projects

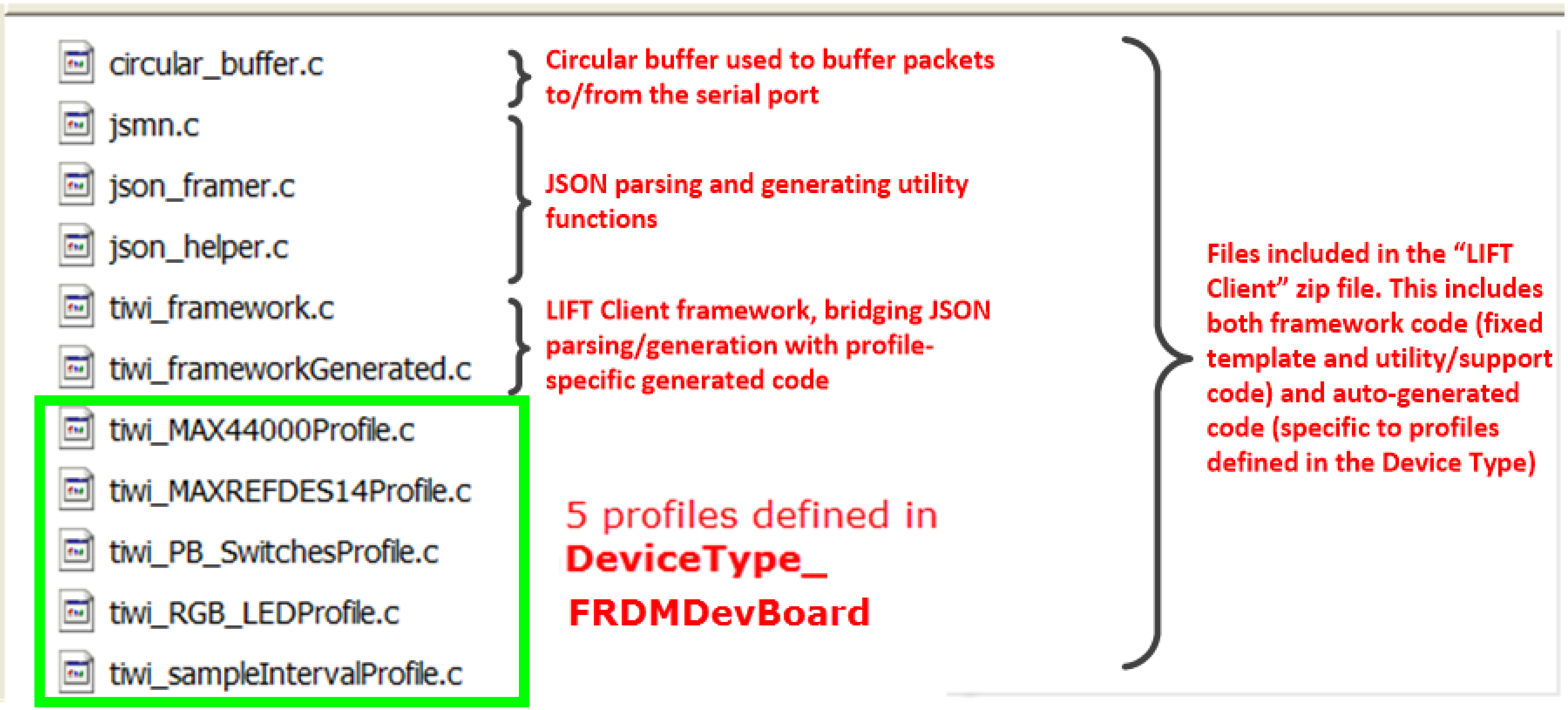
To download fully pre-integrated example source projects targeting spe kits, please see the Pre-Integrated LIFT Client Example Projects page.

Key Name: FRDMDevBoard / Profiles: 5

DeviceType_FRDMDevBoard.zip
Compressed (zipped) Folder

Copyright (C) 2015, LS Research

Auto-Generated LIFT Client source files (that you add to Freescale KDS project…)



circular_buffer.c

jsmn.c

json_framer.c

json_helper.c

tiwi_framework.c

tiwi_frameworkGenerated.c

tiwi_MAX44000Profile.c

tiwi_MAXREFDES14Profile.c

tiwi_PB_SwitchesProfile.c

tiwi_RGB_LEDProfile.c

tiwi_sampleIntervalProfile.c

Circular buffer used to buffer packets to/from the serial port

JSON parsing and generating utility functions

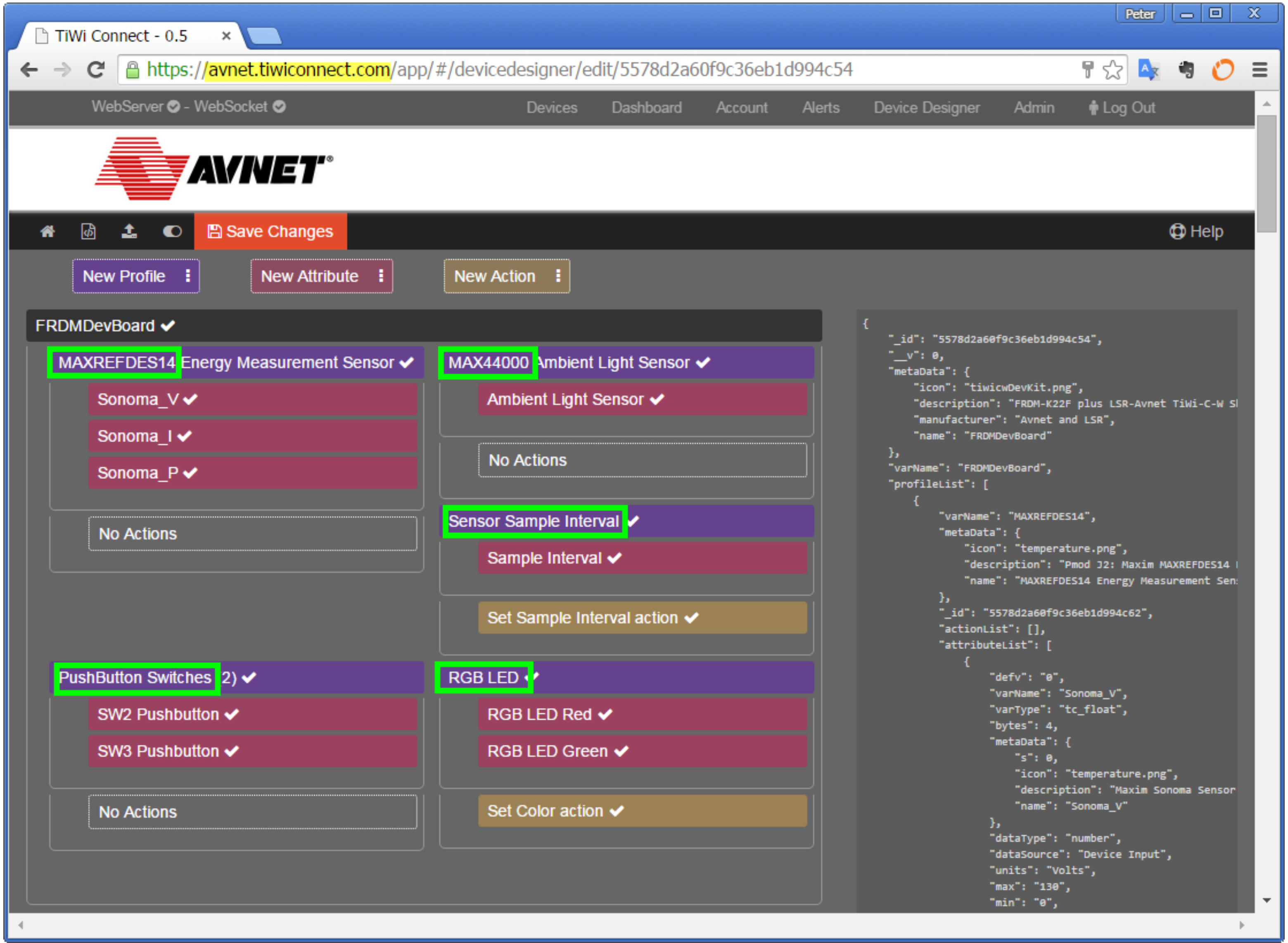
LIFT Client framework, bridging JSON parsing/generation with profile-specific generated code

5 profiles defined in **DeviceType_ FRDMDevBoard**

Files included in the "LIFT Client" zip file. This includes both framework code (fixed template and utility/support code) and auto-generated code (specific to profiles defined in the Device Type)

TiWiConnect: Defining your Device Type

Profile, Attributes and Actions…



```
{
    "_id": "5578d2a60f9c36eb1d994c54",
    "__v": 0,
    "metaData": {
        "icon": "tiwicwDevKit.png",
        "description": "FRDM-K22F plus LSR-Avnet TiWi-C-W S
        "manufacturer": "Avnet and LSR",
        "name": "FRDMDevBoard"
    },
    "varName": "FRDMDevBoard",
    "profileList": [
        {
            "varName": "MAXREFDES14",
            "metaData": {
                "icon": "temperature.png",
                "description": "Pmod J2: Maxim MAXREFDES14
                "name": "MAXREFDES14 Energy Measurement Sen
            },
            "_id": "5578d2a60f9c36eb1d994c62",
            "actionList": [],
            "attributeList": [
                {
                    "defv": "0",
                    "varName": "Sonoma_V",
                    "varType": "tc_float",
                    "bytes": 4,
                    "metaData": {
                        "s": 0,
                        "icon": "temperature.png",
                        "description": "Maxim Sonoma Sensor
                        "name": "Sonoma_V"
                    },
                    "dataType": "number",
                    "dataSource": "Device Input",
                    "units": "Volts",
                    "max": "130",
                    "min": "0",
```

Auto-Generated LIFT Client source files (that you add to Freescale KDS project…)

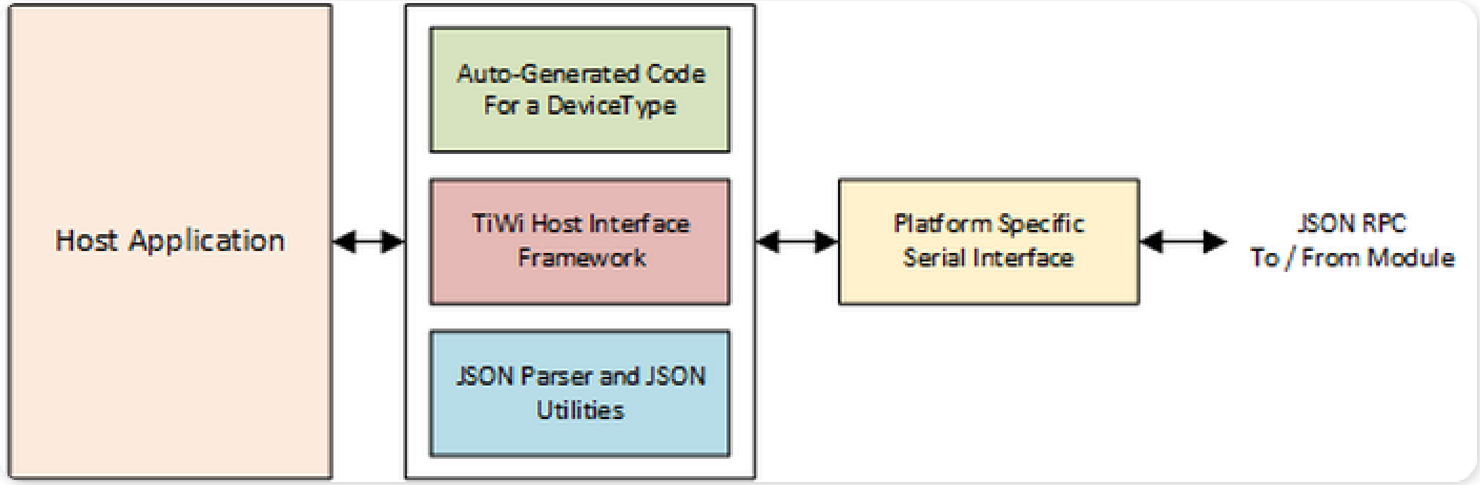
Key patches needed to interface LIFT Client C-code to the platform specific (ie. K22 UART2) User Application are the following:

TransmitToModuleViaUart() - UART TX function call from user code,

RxDataFramerAddByte() - UART RX callback to update circular buffer,

SetAttributes_* - functions called by user code to send attributes to the server,

ActionHandler_* - callback stubs for User code to implement actions from server


Host Application
Auto-Generated Code For a DeviceType
TiWi Host Interface Framework
JSON Parser and JSON Utilities
Platform Specific Serial Interface
JSON RPC To / From Module

The next 2 slides provide more detail…

TiWi Function Calls Added to User Code in Example Reference Design

```
// Initialize the Rx Circular Buffer for the interface UART
InterfaceUartFifo_Config( );

// Reset the JSON parser that determines object boundaries
JsonParserReset( );

// In main(), Kinetis UART TX function called from TiWi code
int TransmitToModuleViaUart(uint8_t* buffer, uint16_t bufLen)

// UART RX Interface Handler to update RX circular buffer
int RxDataFramerAddByte(uint8_t rxByte)

// Framework call to parse packets from RX circular buffer
ParseInterfaceUartRxData()
```

TiWi Function Calls Added to User Code in Example Reference Design

```
// Send Status of Pushbutton Switches SW2 and SW3
SetAttributes_PB_Switches( &Button_AttData );
```

```
// Send measurements from SONOMA Pmod (V, I and P)
SetAttributes_MAXREFDES14( &MAXREFDES14_AttData );
```

```
// Check for pending actions requested by server
GetPendingActionCount_framework();
```

```
// Sends get action message to the server
GetPendingAction_framework();
```

```
// Example Action handler (remotely set RGB LED color)
TC_STATUS ActionHandler_RGB_LED_setColor(stdParseParams_t* pStruct);
```

Example: MCU board to Browser
Send Switch Positions (SetAttributes_)

1) Update "**Button_AttData**" data structure with state of the Pushbuttons:

```
Button_AttData.SW2_ButtonState = GPIO_DRV_ReadPinInput(SW2_PUSH_BUTTON);
Button_AttData.SW3_ButtonState = GPIO_DRV_ReadPinInput(SW3_PUSH_BUTTON);
```

2) Call **Set_Attributes_** function (from interrupt service routine for these GPIO)

```
SetAttributes_PB_Switches( &Button_AttData );
```

Example: Browser to MCU board Set/Clear RGB LEDs (ActionHandler_)

1) Add user code to the provided “stubs” in **UserCallback_RGB_LED_setColor**() function that is called by **ActionHandler_RGB_LED_setColor()** (located in **tiwi_RGB_LEDProfile.c** file generated by Device Designer)

```
GPIO_DRV_SetPinOutput(BOARD_GPIO_LED_GREEN); // GRN = Off
GPIO_DRV_ClearPinOutput(BOARD_GPIO_LED_RED); // RED = On!
```

Define, Build, Debug and Program the MCU User App using Freescale KDS, KSDK and PEx…

<u>Toolchain Used</u>:
Kinetis Design Studio (v3.0.0)
Kinetis Software Development Kit (v1.2.0)

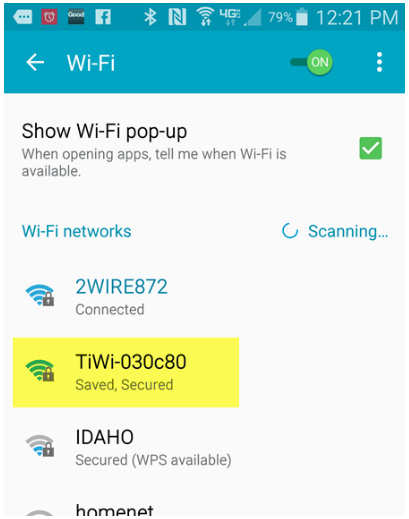
The reference design uses **Processor Expert** for the generation of peripheral configuration, initialization and driver C-code

Once the MCU has been re-flashed with User App and LIFT Client, the board assembly is then ready to establish a WLAN network connection and communicate with the Cloud-based LIFT server…

Wi-Fi Test

"Config Mode" for Easy Wireless A/P Provisioning

1. Hold-down **CONFIG Pushbutton** on LSR Wireless Shield
2. Press and release **RESET Pushbutton** on FRDM-K22F board
3. Keep holding **CONFIG Pushbutton** until Red LED flashes twice
4. From Smartphone Wi-Fi settings, connect to SSID named **TiWi-……**
5. Open SmartPhone Internet browser, connect to **192.168.0.1**
6. Change/verify settings as shown, then click on **Save Settings**
7. Click **Wi-Fi Setup** then select SSID of the network you'd like to connect to
8. Additional documentation/help at https://devkit.tiwiconnect.com



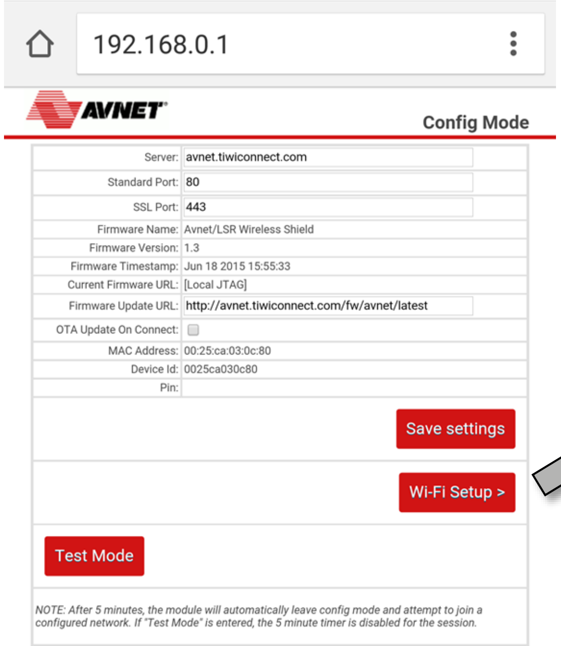
Show Wi-Fi pop-up ☑
When opening apps, tell me when Wi-Fi is available.

2WIRE872
Connected

TiWi-030c80
Saved, Secured

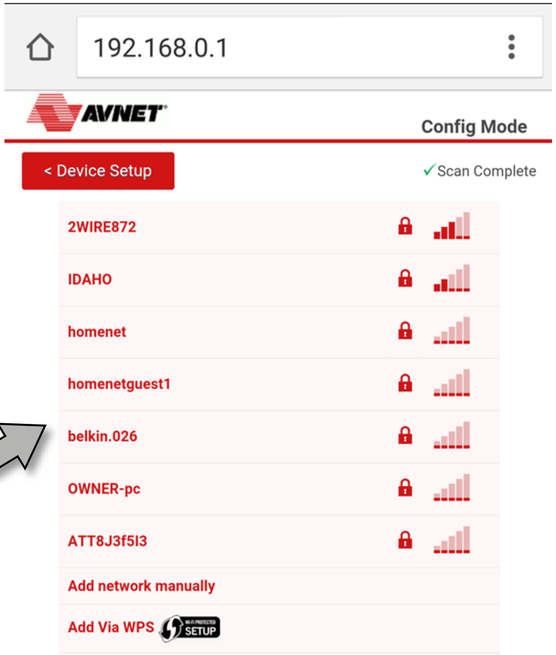
IDAHO
Secured (WPS available)

homenet



Server:	avnet.tiwiconnect.com
Standard Port:	80
SSL Port:	443
Firmware Name:	Avnet/LSR Wireless Shield
Firmware Version:	1.3
Firmware Timestamp:	Jun 18 2015 15:55:33
Current Firmware URL:	[Local JTAG]
Firmware Update URL:	http://avnet.tiwiconnect.com/fw/avnet/latest
OTA Update On Connect:	□
MAC Address:	00:25:ca:03:0c:80
Device Id:	0025ca030c80
Pin:	

NOTE: After 5 minutes, the module will automatically leave config mode and attempt to join a configured network. If "Test Mode" is entered, the 5 minute timer is disabled for the session.

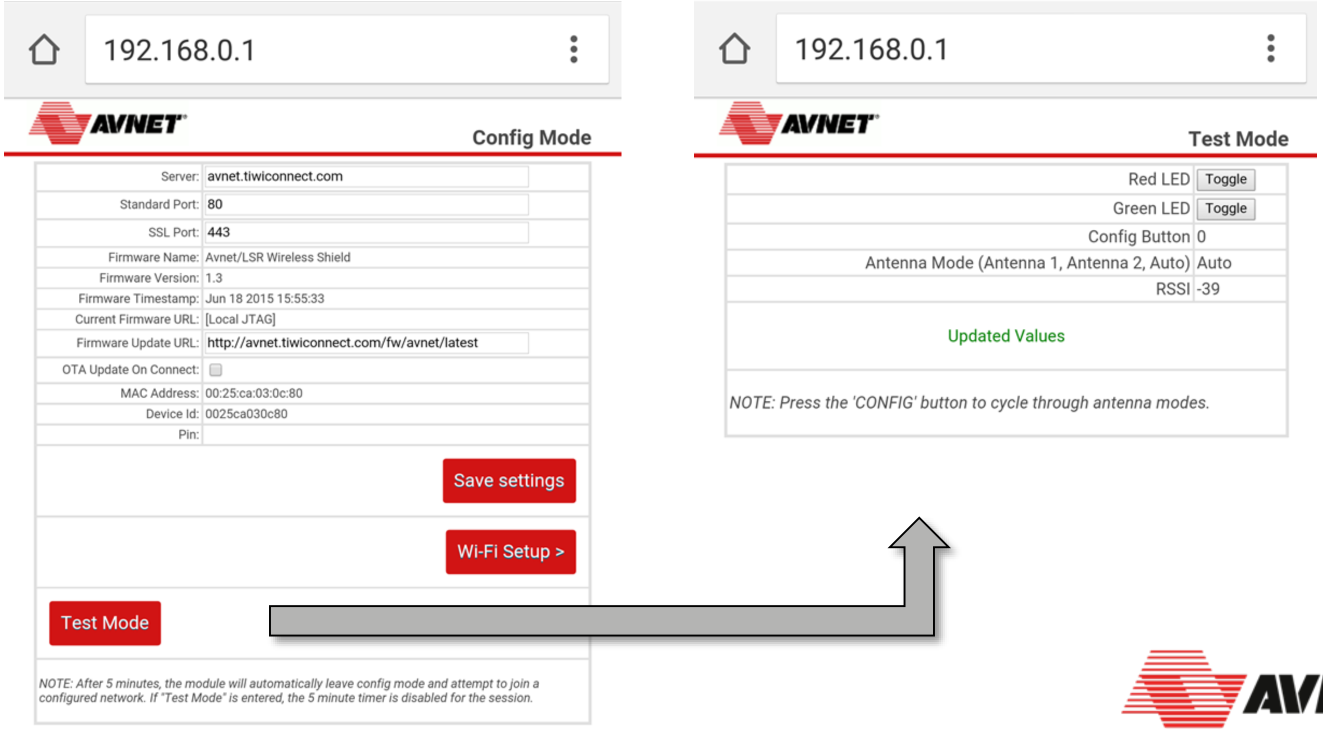


✓Scan Complete

2WIRE872
IDAHO
homenet
homenetguest1
belkin.026
OWNER-pc
ATT8J3f5I3
Add network manually
Add Via WPS

“Test Mode” for Basic Test of Wi-Fi module functionality

- Use the TiWi-CW module’s useful “Test Mode” for quick verification of Wi-Fi communication between your SmartPhone and the Wireless Shield
- Select “Test Mode” from Config Mode screen (SmartPhone Internet browser)
- Click the “Toggle” buttons to control Red and Green LEDs on Wireless Shield
- Press Config button on Wireless Shield to cycle through the antenna options.
- RSSI reports the currently received Wi-Fi signal strength (in dB)



192.168.0.1

AVNET Config Mode

Server:	avnet.tiwiconnect.com
Standard Port:	80
SSL Port:	443
Firmware Name:	Avnet/LSR Wireless Shield
Firmware Version:	1.3
Firmware Timestamp:	Jun 18 2015 15:55:33
Current Firmware URL:	[Local JTAG]
Firmware Update URL:	http://avnet.tiwiconnect.com/fw/avnet/latest
OTA Update On Connect:	☐
MAC Address:	00:25:ca:03:0c:80
Device Id:	0025ca030c80
Pin:	

NOTE: After 5 minutes, the module will automatically leave config mode and attempt to join a configured network. If "Test Mode" is entered, the 5 minute timer is disabled for the session.

192.168.0.1

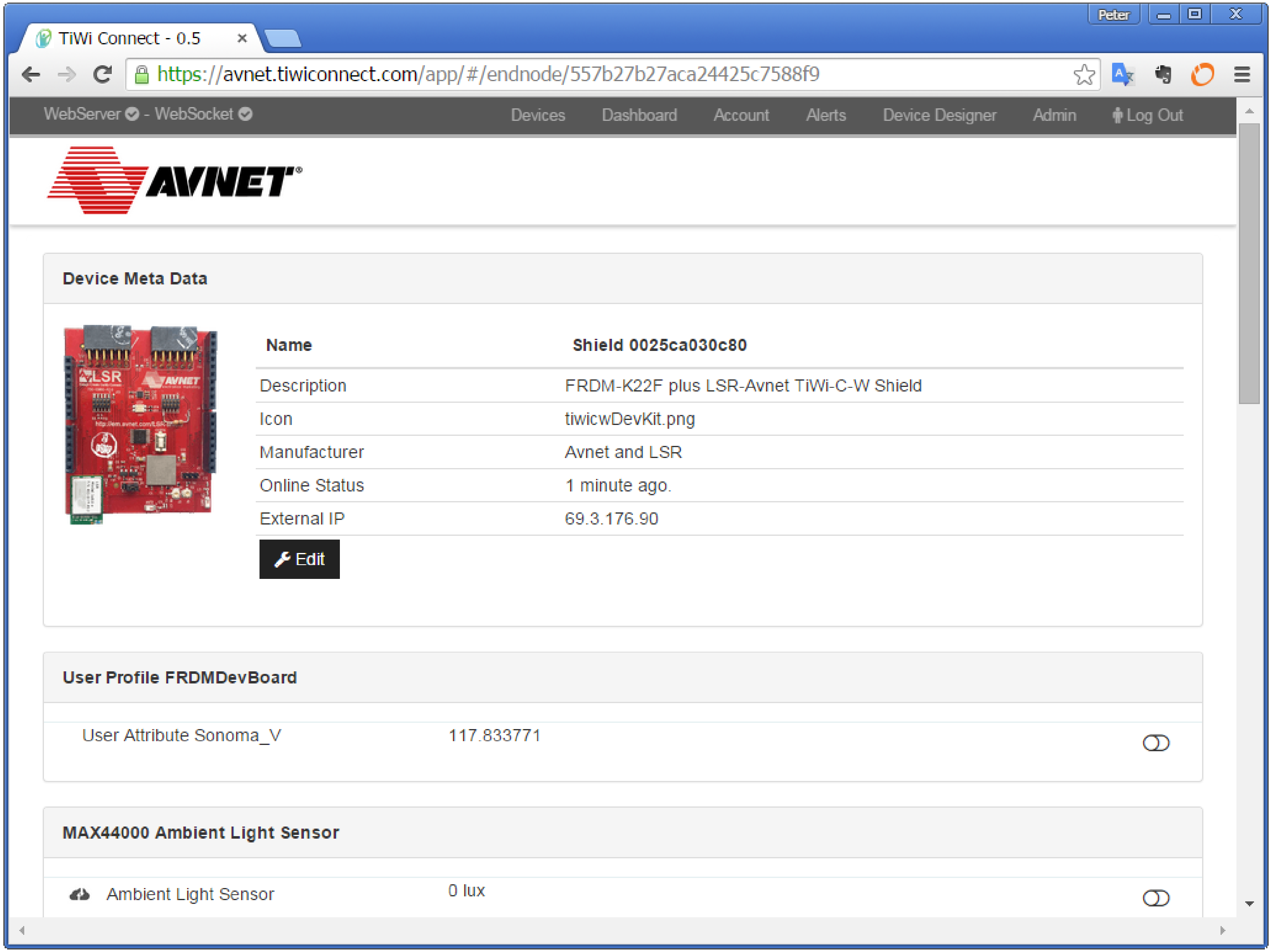
AVNET Test Mode

Red LED	Toggle
Green LED	Toggle
Config Button	0
Antenna Mode (Antenna 1, Antenna 2, Auto)	Auto
RSSI	-39

Updated Values

NOTE: Press the 'CONFIG' button to cycle through antenna modes.

TiWiConnect "Device View"



Device Meta Data

Name	Shield 0025ca030c80
Description	FRDM-K22F plus LSR-Avnet TiWi-C-W Shield
Icon	tiwicwDevKit.png
Manufacturer	Avnet and LSR
Online Status	1 minute ago.
External IP	69.3.176.90

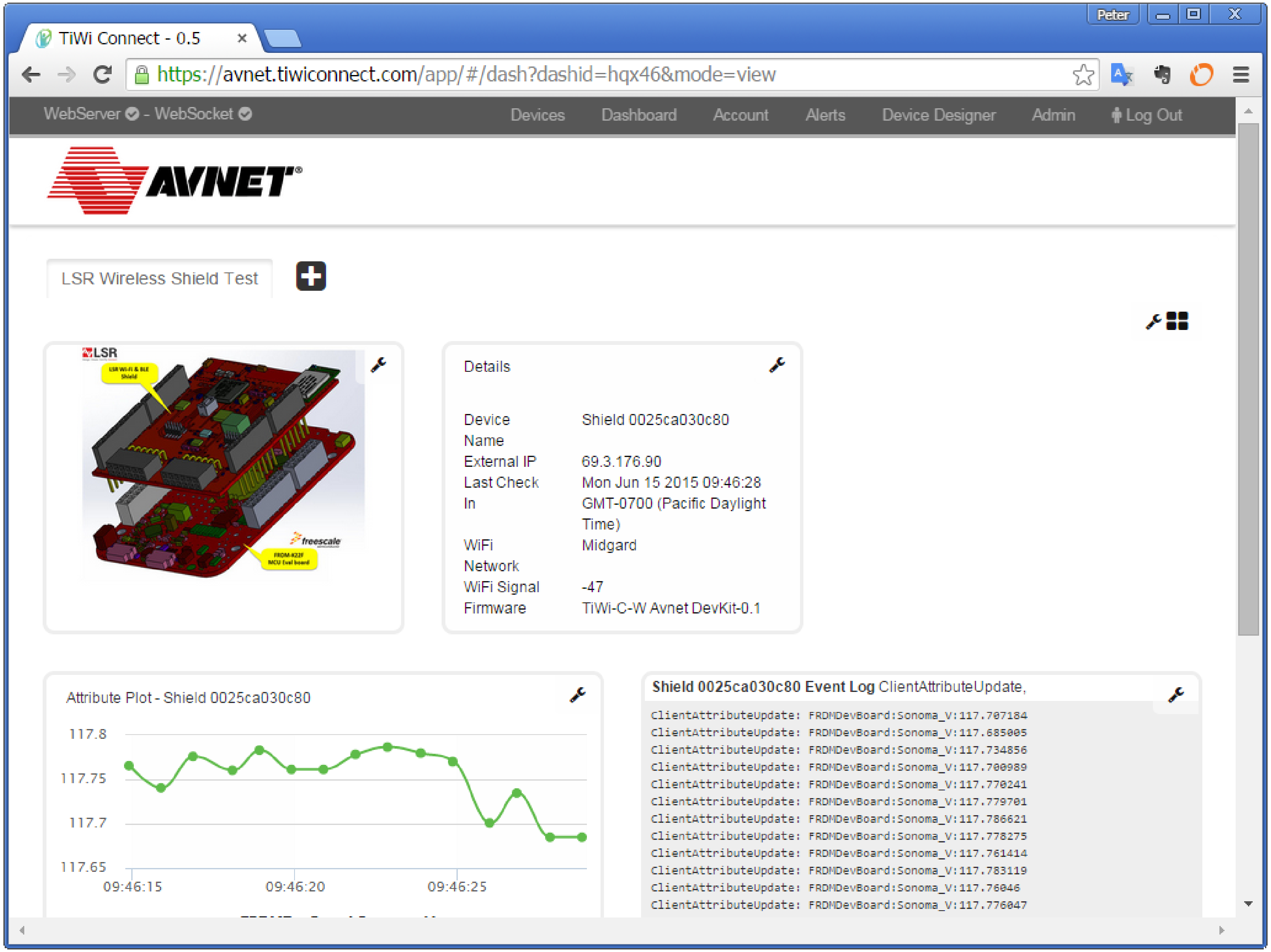
User Profile FRDMDevBoard

User Attribute Sonoma_V 117.833771

MAX44000 Ambient Light Sensor

Ambient Light Sensor 0 lux

TiWiConnect “Dashboard”



Details

Device Name	Shield 0025ca030c80
External IP	69.3.176.90
Last Check In	Mon Jun 15 2015 09:46:28 GMT-0700 (Pacific Daylight Time)
WiFi Network	Midgard
WiFi Signal	-47
Firmware	TiWi-C-W Avnet DevKit-0.1

Attribute Plot - Shield 0025ca030c80

Shield 0025ca030c80 Event Log ClientAttributeUpdate,

```
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.707184
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.685005
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.734856
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.700989
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.770241
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.779701
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.786621
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.778275
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.761414
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.783119
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.76046
ClientAttributeUpdate: FRDMDevBoard:Sonoma_V:117.776047
```

BLE Software

SaBLE-x Developer Tool Suite

Soon to be released **Developer Tool Suite** from LSR for SaBLE-x include:

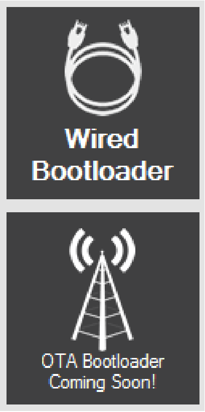
- **Serial-to-BLE**
  Release 1: Simple API and Source Code, wired Bootloader, PC-as-Host tool (July 2015)
  Release 2: Advanced API and Source Code, OTA Bootloader

- **BLE Profile Designer**
  Intuitive GUI tool that generates C-code of custom Bluetooth profile to be used on host

- Pre-Integrated **Cloud Agent** for TiWiConnect™ IoT Platform

- **Host Source Code** for common host processors
  Includes TI EM and Arduino-style adaptor boards to further support quick and easy development

Serial-to-BLE
API
API Library & Source Code
API Command Manager Coming Soon!
PC Host for Serial-to-BLE Coming Soon!



Serial-to-BLE Software

Simplifies SaBLE-x development with external host

Serial-to-BLE API Library

C-code for both module and host

API Command Manager

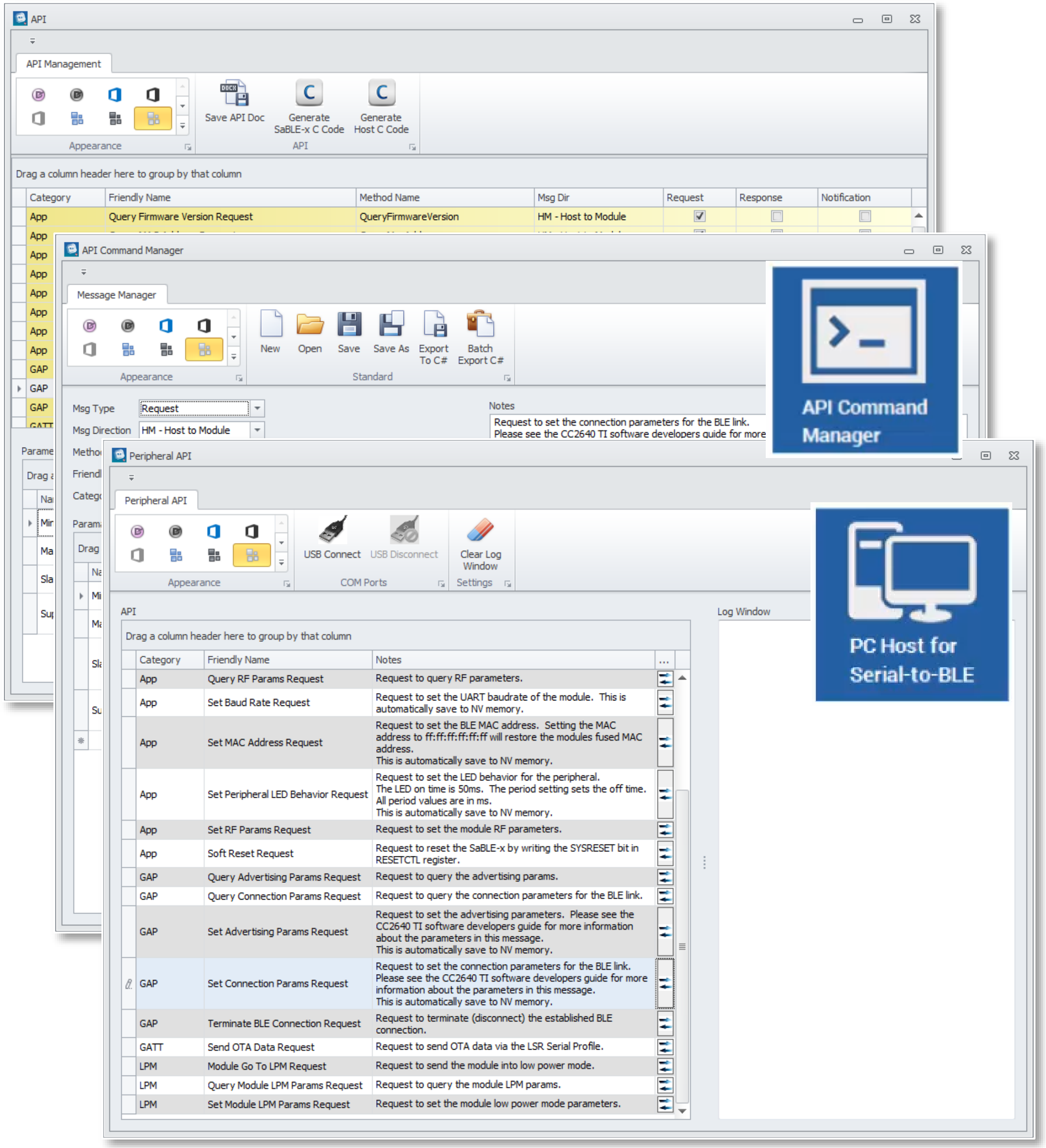
Graphical interface facilitating quick add/edit of API library commands

PC-to-Module

Direct connection for convenient testing of serial commands

Boot Loader Utility

Wired (via USB) and OTA (future)



Drag a column header here to group by that column

Category	Friendly Name	Method Name	Msg Dir	Request	Response	Notification
App	Query Firmware Version Request	QueryFirmwareVersion	HM - Host to Module	☑	□	□

Msg Type: Request

Msg Direction: HM - Host to Module

Notes: Request to set the connection parameters for the BLE link. Please see the CC2640 TI software developers guide for more

API

Drag a column header here to group by that column

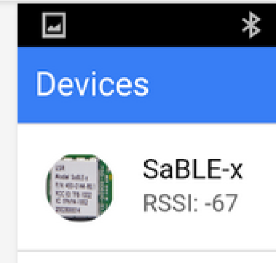
Category	Friendly Name	Notes
App	Query RF Params Request	Request to query RF parameters.
App	Set Baud Rate Request	Request to set the UART baudrate of the module. This is automatically save to NV memory.
App	Set MAC Address Request	Request to set the BLE MAC address. Setting the MAC address to ff:ff:ff:ff:ff:ff will restore the modules fused MAC address. This is automatically save to NV memory.
App	Set Peripheral LED Behavior Request	Request to set the LED behavior for the peripheral. The LED on time is 50ms. The period setting sets the off time. All period values are in ms. This is automatically save to NV memory.
App	Set RF Params Request	Request to set the module RF parameters.
App	Soft Reset Request	Request to reset the SaBLE-x by writing the SYSRESET bit in RESETCTL register.
GAP	Query Advertising Params Request	Request to query the advertising params.
GAP	Query Connection Params Request	Request to query the connection parameters for the BLE link.
GAP	Set Advertising Params Request	Request to set the advertising parameters. Please see the CC2640 TI software developers guide for more information about the parameters in this message. This is automatically save to NV memory.
GAP	Set Connection Params Request	Request to set the connection parameters for the BLE link. Please see the CC2640 TI software developers guide for more information about the parameters in this message. This is automatically save to NV memory.
GAP	Terminate BLE Connection Request	Request to terminate (disconnect) the established BLE connection.
GATT	Send OTA Data Request	Request to send OTA data via the LSR Serial Profile.
LPM	Module Go To LPM Request	Request to send the module into low power mode.
LPM	Query Module LPM Params Request	Request to query the module LPM params.
LPM	Set Module LPM Params Request	Request to set the module low power mode parameters.

LSR ModuleLink for BLE
iOS and Android Mobile App

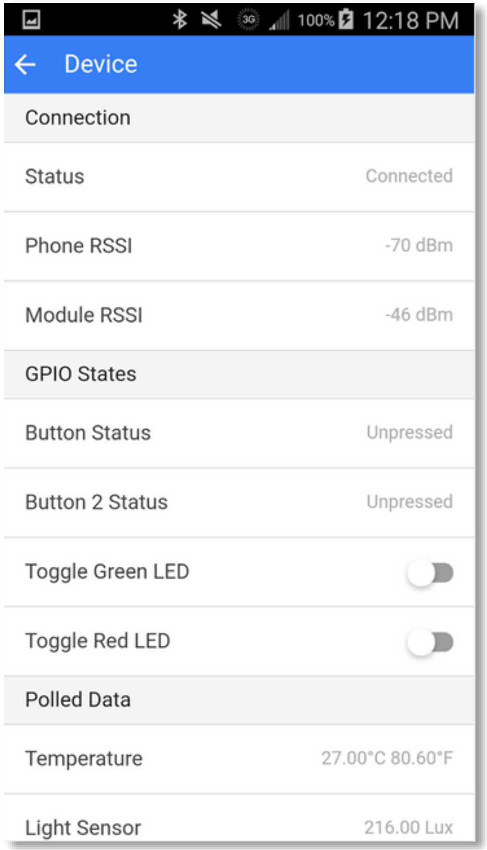
Mobile app available for both iOS and Android platforms.

Predefined functions are provided for remote monitoring and control of SaBLE-x Dev Kit hardware, but this can be modified to match user hardware.

Source code for app is available in both
- native (iOS, Android) and
- non-native (Ionic) formats
to accelerate custom mobile app development efforts



SaBLE-x
RSSI: -67



Connection	
Status	Connected
Phone RSSI	-70 dBm
Module RSSI	-46 dBm
GPIO States	
Button Status	Unpressed
Button 2 Status	Unpressed
Toggle Green LED	
Toggle Red LED	
Polled Data	
Temperature	27.00°C 80.60°F
Light Sensor	216.00 Lux

BLE Testing:
Simple Hosted Peripheral Firmware

“Hosted mode” firmware needs to be flashed into the SaBLE-X module (Use a JTAG programmer connected to **J10** on the LSR Wireless Shield)

Default UART settings for BLE are 19200 baud, 8-N-1
The module will send out the following JSON reset notification after reset:
{"jsonrpc": "2.0","method": "ResetAlert","params": {"Source": 0}}

The module then stays awake for 3 seconds after boot then goes to sleep.
To wake the module, the JSON wakeup command must be sent:
{"jsonrpc": "2.0","method": "WakeupModule"}

Once awake the module will respond with:
{"jsonrpc": "2.0","method": "ModuleAwake"}

Architecting User Applications

LSR Shield
Wireless Software Overview

Wi-Fi interface via TiWi-C-W module:

- Acts as wireless network processor with Kinetis MCU as host
- TiWiConnect LIFT software running on Kinetis MCU (LIFT Client), TiWi-C-W module (LIFT Agent) and TiWiConnect Cloud Server (LIFT Server) creates a simple tunnel to send data from MCU to the cloud and on to web portal or mobile app
- Simplified Soft AP config mode for connecting to local network

Tools to support WiFi & Cloud development:

- LIFT Device Designer GUI tool
- Auto-generated C-code and examples for LIFT Client on MCU
- **Avnet.tiwiconnect.com**
  portal for device management, dashboards, and alerts

Bluetooth SMART

BLE interface via SaBLE-x module:

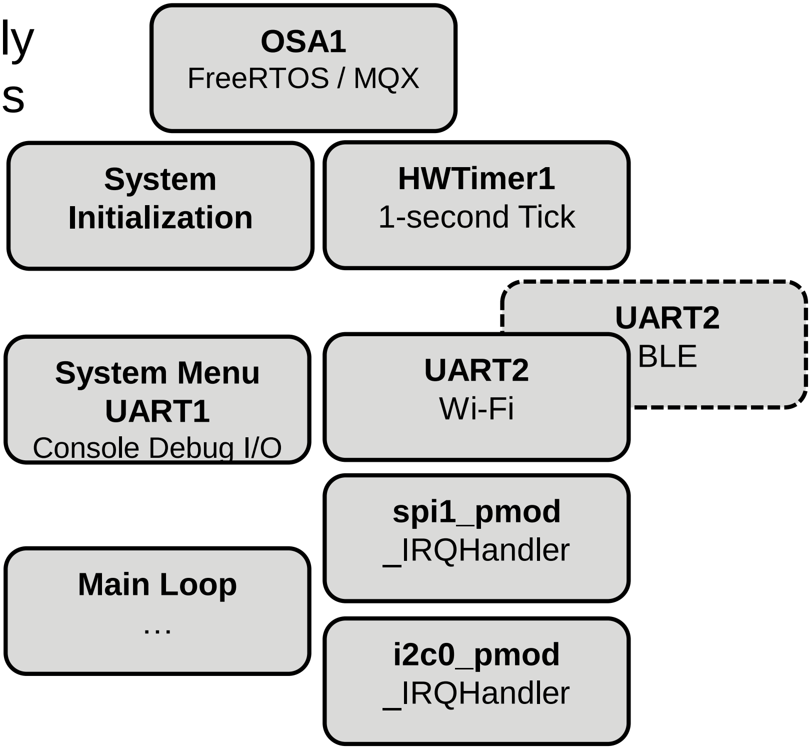
- Acts as wireless network processor with Kinetis MCU as host
- Operates in peripheral mode
- Enables wireless-based peripheral expansion

Tools to support BLE development:

- LSR Serial-to-BLE API & C-Code
- Other Developer Tool Suite tools:
  - Wired Bootloader
  - PC-as-Host testing tool

Software:
Wi-Fi User Application Design

- **DbgCs1** (UART1) continuously monitored for new selection from Test Menu user interface
- **SW2** Freedom board pushbutton continuously monitored, RED LED toggled on button press
- **SW3** Freedom board pushbutton continuously monitored, GRN LED toggled on button press
- **HWTimer1** used for 1-second tick that sequences the following actions:
  - Read the **SPI1 Pmod** sensors
  - Read the **I2C0 Pmod** sensors
  - **UART2** messaging between LIFT Client on MCU and LIFT Server in the Cloud (via LIFT Agent on TiWi-C-W)

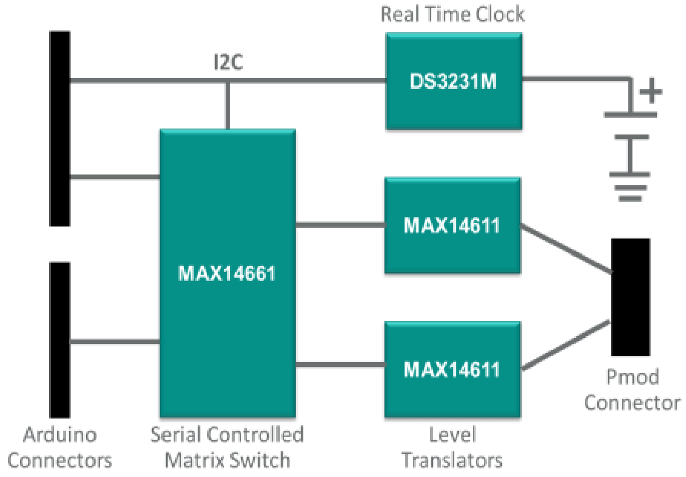


Other Shield & Pmod Resources

Maim: Arduino to Pmod Adaptor Shield

MAXREFDES72# PMOD Adapter for Arduino Platforms

- Arduino® UNO R3 compatible
- IOREF support for 3.3V or 5V signal levels
- 2x6 Digilent® Pmod™ compatible interface
- Supports Pmod types 1, 2, 3, and 4, I2C and nonstandard pinouts
- DS14611 level-shifts from Shield to Pmod pins
- MAX14661 16:2 multiplexer
- DS3231M Precision Real Time Clock (with integrated MEMS resonator)



$25.00

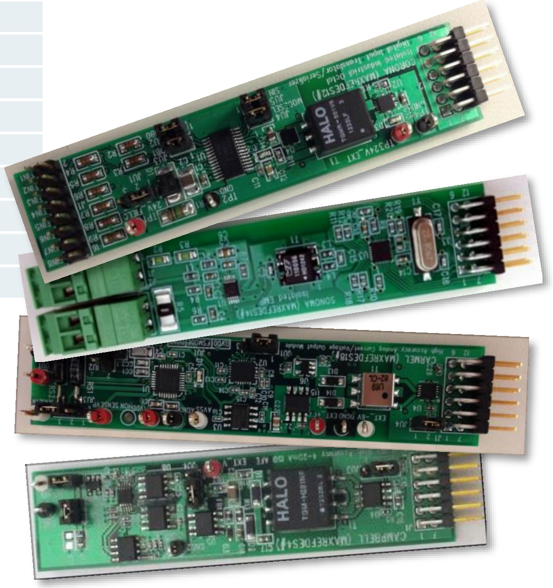
Part#: MAXREFDES72#

www.maximintegrated.com/en/design/reference-design-center

Maxim Pmod™-Compatible Modules

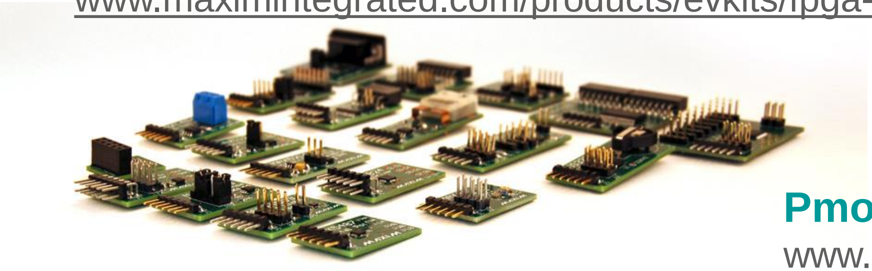
- Maxim Analog Essentials Collection
- Maxim multi-device Subsystem Reference Designs…

Name	Part#	Description
Alcatraz	MAXREFDES34#	SHA-256 Secure Authentication Design
Carmel	MAXREFDES18#	High Accuracy Analog Current/Voltage Output
Sonoma	MAXREFDES14#	Isolated Energy Measurement Subsystem Reference Design
Fremont	MAXREFDES6#	16-Bit, High-Accuracy, 0 to 100mV Input, Isolated Analog Front-End (AFE)
Corona	MAXREFDES12#	Isolated Industrial Octal Digital Input Translator/Serializer
Fresno	MAXREFDES11#	16-Bit High-Accuracy 0 to 10V Input Isolated Analog Front End (AFE)
Campbell	MAXREFDES4#	16-Bit High-Accuracy 4-20mA Input Isolated Analog Front End (AFE)
Santa Fe	MAXREFDES5#	16-Bit High Accuracy Multi-Input Isolated Analog Front End (AFE)
Riverside	MAXREFDES8#	3.3V Input, 12V (15V) Output Isolated Power Supply
Lakewood	MAXREFDES7#	3.3V Input, ±12V (±15V) Output Isolated Power Supply
Oceanside	MAXREFDES9#	3.3V to 15V Input, ±15V (±12V) Output, Isolated Power Supply
	MAXREFDES43#	I2C SHA-256 secure authentication (DS28C22 Secure Memory)
	MAXREFDES44#	1-Wire Asymmetric secure authentication (DS28E35 Secure Memory)



Analog Essentials Collection

www.maximintegrated.com/products/evkits/fpga-modules



Pmod™-compatible Subsystem Reference Designs

www.maximintegrated.com/design/reference-design-center

Next Steps and More Info…

Optional Use Of External Antennas

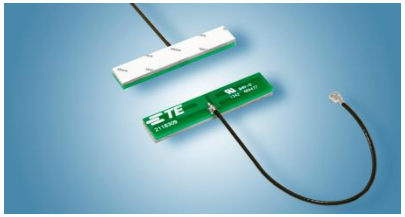
- TiWi-C-W module implementation on Shield includes dual **U.FL connectors** for attachment of external antennas.
- Use of a high Gain external antenna ensures maximum range and data throughput.
- Suitable *"Standard Cabled Antennas"* from *TE Connectivity* are shown tabled below:

Part Number	Frequency Range (MHz)	Standard	Peak Gain (dB)	VSWR
1513381-1	3100 – 6000	UWB	+4	<3.0:1
1513472-5	2400 – 2485.5, 5150 – 5875	BT, Wi-FI, ZigBee	+3	<3.0:1
2118059-1	2300 – 3800	BT, Wi-FI, ZigBee	+4	<3.0:1
2118060-1	2300 – 3800, 5150 – 5875	BT, Wi-FI, ZigBee	+2	<3.0:1
2118326-1	4900 – 5875	Wi-Fi	+2.4	<2.5:1
2118309-1	2400 – 2483.5, 4900 – 5875	BT, Wi-FI, ZigBee	+3.7	<2.0:1

1513381-1, 1513472-5

2118059-1, 2118060-1

2118326-1



2118309-1

Why Choose Pre-Certified Wireless Modules from LS Research?

Beyond wireless modules, LSR's unmatched breadth of services ensures rapid progress to production

Design Services | RF Products | EMC Compliance Testing

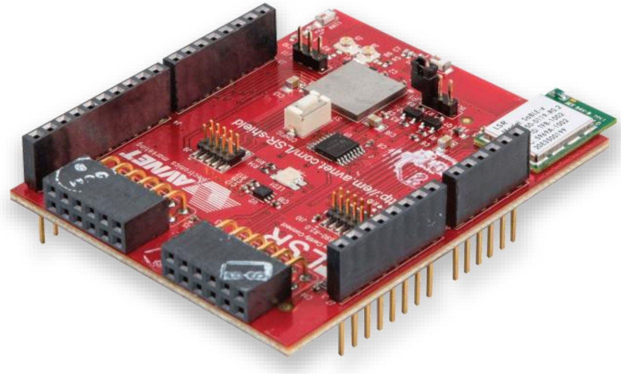
- RF Hardware Design
- Custom Antenna Design
- Embedded Software Development
- Cloud Platform Development
- Mobile App Development
- LSR Design Studio
  - Industrial Design, Mech. Engineering and 3D Prototype Lab
- FCC/IC/CE and Global EMC Testing Services
- Turn-Key Test Fixture Development and Implementation

> *"The extent of LSR's design service capabilities is a major differentiator from other wireless module vendors" - Avnet FAE*

Avnet EMA – Wireless Module Linecard

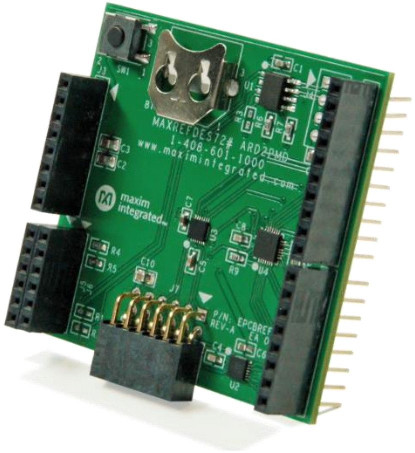
	WLAN	BT/BLE	Combo	Sub-GHz	802.15.4	Cellular	GPS
Module Supplier	Texas Instruments	Texas Instruments	Texas Instruments				Texas Instruments
	Broadcom	Broadcom	Broadcom			MultiTech Systems	
	muRata	muRata	muRata	muRata	muRata		
	LS Research	LS Research	LS Research	LS Research	LS Research		
	Inventek Systems	Inventek Systems	Inventek Systems				Inventek Systems
	Taiyo Yuden	Taiyo Yuden	Taiyo Yuden				Taiyo Yuden
	Panasonic	Panasonic	Panasonic	Panasonic	Panasonic		antenova m2m
	Microchip	Microchip		Microchip	Microchip		
	ST	ST		ST		Quectel	Quectel
		Cypress					
	bluegiga	bluegiga					
	wizwi	wizwi	wizwi				wizwi
	Digi	Digi	Digi	Digi	Digi		
		Anaren		Anaren	Anaren		
	TDK	TDK			MMB Networks		
	Atmel	Atmel			CEL		
		ROHM					

FTF-2015 Avnet Booth Demos



Prototyping Cloud IoT Products using FRDM-K22F and LSR Wireless Shield

- Freescale **FRDM-K22F** Freedom board
- Avnet LSR Wi-Fi/BLE Shield
- Maxim Sonoma Pmod



Prototyping using mbed.org and Maxim's Ard2Pmod Shield

- Freescale **FRDM-K64F** Freedom board
- Maxim ARD2Pmod Shield (MAXREFDES2#)

Preview website for Avnet LSR Wireless Shield:

http://em.avnet.com/LSR-shield

Thank
You
Vinaka
köszönöm
Kiitos
Maake
Asante
Shukria
Dhanyavadagalu
감사합니다
Dankscheen
спасибо
Kam Sah Hammida
اركش
Manana
Dankon
Diolch i Chi
Terima Kasih
Matondo
Dank Je
Mauruuru
Biyan
Taiku
Tack
Blagodaram
Ngiyabonga
Dziekuje
Chokrane
Arigato
Grazie
Juspaxar
Gracias
Mochchakkeram
நன்றி
Bedankt
Dakujem
धन्यवाद
Grazas
cảm ơn bạn
Khap
Paldies
Tingki
Gratias Tibi
Ua Tsaug Rau Koj
Di Ou Mèsi
Kia Ora
Kop Khun
Obrigado
Děkuji
Nuwun
Nirringrazzjak
ありがとう
Suksama
Rahmat
Hvala
Welalin
Matur
谢谢
хвала
Djiere Dieuf
Misaotra
Danke
Merci
Go Raibh Maith Agat
Tuke
Eskerrik Asko
Salamat
ขอขอบคุณคุณ
Najis