

# SDK 2.3 Release Package

Quick Start Guide

Release: 08-30-2021

InnoPhase, Inc.

6815 Flanders Drive

San Diego, CA 92121

[innophaseinc.com](http://innophaseinc.com)

## Contents

1	Terms & Definitions .....	2
2	Introduction .....	3
3	Getting Started.....	3
3.1	pc_tools .....	3
3.2	binaries .....	3
3.3	apps.....	3
3.4	examples.....	4
3.5	conf .....	5
3.6	doc .....	5
3.7	components .....	6
3.8	include .....	7
3.9	lib .....	7
3.10	script .....	7
3.11	tools .....	7
3.12	root_fs.....	7
4	Host Packages .....	8
5	Support .....	9
6	Disclaimers.....	10

## 1 Terms & Definitions

API	Application Programming Interface
AWS	Amazon Web Services
BLE	Bluetooth Low Energy
GPIO	General-Purpose Input/Output
HIO	Host Interface
HTTPS	Hypertext Transfer Protocol Secure
MPD	Multi-Purpose Demo
MQTT	Message Queuing Telemetry Transport
TLM	Telemetry
TLS	Transport Layer Security
UART	Universal asynchronous receiver-transmitter

## 2 Introduction

This document provides a brief introduction to the SDK 2.3 release package.

## 3 Getting Started

### 3.1 pc\_tools

The pc\_tools folder consists of the programming (download) and multi-purpose demo GUI tools. These tools are available for both Windows and Linux platforms.

**Note:** For Windows OS, user will need to download the Zadig tool for installation of libUSBK drivers. Stepwise instructions on downloading the Zadig software for libUSBK driver installation is available in `doc/ug_evb_a` and `doc/ug_eclipse_setup_windows`.

### 3.2 binaries

The binaries folder consists of firmware images which can be used to understand the protocols and functionality supported on Talaria TWO™.

The eval folder consists of a firmware image for MPD and Alexa\_Ready application, while the product folder consists of the firmware image for AT commands.

### 3.3 apps

The apps folder consists of production ready example applications that the user can run on Talaria TWO. Following applications are available in the release package:

Protocol	Description
ble_provisioning	Describes multiple APIs to create a provisioning application using BLE as the mode of transferring provisioning data. Includes demo Android smartphone app (.apk)
fota	Describes the FOTA process for the Talaria TWO EVB using the Talaria TWO SDK
gordon	Describes flash helper utility application
helloworld	Describes basic helloworld application
iot_aws	Describes using Talaria TWO board and the SDK with Amazon Web Services (AWS) IoT
iot_azure	Readme file containing the link to IoT_Azure application note present in Github
iperf3	Describes iperf3 server application on Talaria TWO

ssbl	Describes enhancing the flexibility of application on Talaria TWO using SSBL
stw	Describes serial to Wi-Fi application over HIO

## 3.4 examples

The examples folder consists of example applications that can be run on the Talaria TWO EVB to achieve specific functionalities. Following concepts/protocols are available in the release package:

Protocol	Description
at_custom_cmd	Describes custom AT commands which the user can use apart from the standard commands.
ble_beacons	Example codes describing the basic concept of Eddystone Beacon <ul style="list-style-type: none"> <li>- Eddystone UID</li> <li>- Eddystone URL</li> </ul> Eddystone TLM
ble_wifi_bridge	Example code for receiving a text message from a connected BLE client and publishing it to a CloudMQTT broker
crash_handling	Example code for using the crash handler API to handle and debug error cases
http_client	Example codes for using HTTP client APIs to connect to HTTP servers in secured (HTTPS) and non-secured way
innoos_memory_mgmt	Example code for basic memory management and error handling
innoos_msg_q	Example codes describing the use of message queue APIs
innoos_threads_semaphores	Example codes describing the use of threads and semaphore APIs
innoos_timers_callouts	Example codes describing the use of Timers and Callout APIs
innoos_work_q	Example codes describing the use of Work Queue APIs
lp_uart	Example codes describing the use of UART APIs
sleep_enable_disable	Describes the basics of sleep management in InnoOS™ (os_suspend_enable() and os_suspend_disable() functions)
socket_wakeup	Describes the basics of sleep management in InnoOS™ (Talaria TWO wake-up from sleep mode)
spi_flash	Example codes describing using of SPI Flash APIs for Talaria TWO EVK
un-assoc	Example codes describing Wi-Fi un-associated mode transmission APIs available in the SDK, call-back events, notifications and associated data structures
using_ble	Introduction to BLE APIs through code samples consisting of a server and client application

using_wifi	Example codes describing the Wi-Fi connection manager APIs
watchdog_timer	Describes managing Talaria TWO watchdog timer using the functions provided by the watchdog driver
wcm_multi_ap	Describes the application example of Wi-Fi Connection with Multi-Access Point (WCM_MULTI_AP) available in the SDK
wcm_pm	Describes the Wi-Fi Connection Manager power management APIs
gpio	Demo application describing the management of GPIO PINS of the Talaria TWO device
adc	Describes using the Analog to Digital Converter (ADC) peripheral of Talaria TWO modules
pwm	Describes using the Pulse Width Modulation (PWM) peripheral of Talaria TWO modules
using_hio	Describes the fundamentals of developing HIO based application on both host and Talaria TWO EVK

## 3.5 conf

The config folder consists of the make and debugger configuration files. These files include ld and linker scripts and make file rules used for firmware configuration.

## 3.6 doc

The doc folder consists of the Talaria TWO SDK API Reference Guide that provides details on the programmer's API and user guides for the following:

1. `ug_evb_a`: Talaria TWO Evaluation Board an overview of the evaluation board explaining its key features and functions
2. `ug_eclipse_setup_windows`: Eclipse setup in Windows describes developing an application using Eclipse and Talaria TWO SDK
3. `ug_env_setup_linux`: Environment set-up with Talaria TWO for Linux describes setting up the development environment for Talaria TWO™ SDK on an Ubuntu VirtualBox with a Windows 10 host
4. `ug_wsl`: Windows Subsystem for Linux describes developing an application using Windows Subsystem for Linux (WSL) and Talaria TWO SDK

### 3.7 components

This folder consists of files which serve as common components which all applications can make use of. It acts as a library, where any application using HTTP, SNTP, etc., can link to this library and directly use the function. The following protocols are available in the components folder serving as a library for applications:

1. checksum
2. fota
3. http
4. json
5. mqtt
6. provisioning
7. secure\_fs
8. ssl\_wrap
9. websocket
10. sntp
11. utils
12. out

### 3.8 include

The include folder consists of the SDK include files (.h files) for all applications/examples in the release package.

### 3.9 lib

The lib folder consists of the SDK library files which can be used by all applications/examples.

### 3.10 script

The script folder consists of helper scripts/utilities used to achieve multiple functionalities like program, read, write and so on.

### 3.11 tools

The tools directory contains certain tools, supporting files and scripts used for development on Talaria TWO. This directory further contains the following three directories:

1. fletcher32 - tool used to create checksum files for checking the integrity of the configuration files. Currently `part.json`, `boot.json` and `fota_config.json` files' integrity is checked using the checksum.
2. mklittlefs - tool used to create the file system image.
3. partition\_files - flash partition files used with SSBL setup (`ssbl_part_table.json`) and without SSBL setup (`standard_part_table.json`).

### 3.12 root\_fs

This folder contains the basic files that need to be present in root\_fs image as required for SSBL. It also contains the script to build the root\_fs.



## 4 Host Packages

The Host packages can be used by a Host to extend the Wi-Fi and BLE capability using Talaria TWO device connected to it. The Host can be connected to Talaria TWO using UART or SPI interface. Talaria TWO SDK supports different types of Host Packages – Command Based, and API Based.

### 1. AT Commands (Command based)

- a. This enables the Host to execute a sequence of commands to enable, configure and utilize the Wi-Fi and BLE capability of Talaria TWO. This binary is present under `<sdk_x.y>/binaries/product/at/bin/t2_atcmds.elf`. The commands to be used from the host are mentioned in the user guide - `<sdk_x.y>/binaries/product/at/doc/`.
- b. This interface can be extended to build custom commands as per user needs and an example is present in `<sdk_x.y>/examples/at_custom_cmd`.

### 2. Host API (API based)

- a. The Host application can use the Host APIs which communicate with Talaria TWO to enable, configure, and utilize the Wi-Fi and BLE capability of Talaria TWO.
- b. The Host APIs can be used by either a Microcontroller Host (like STM32) or a Linux Host (like R-Pi). For Linux Hosts, this binary is present under `<sdk_x.y>/apps/stw/bin/stw.elf`. There is a test application - `<sdk_x.y>/script/talaria_cli.py` which can be run from the host to test the capabilities of the Host API.
- c. This interface can be extended to build custom Host APIs as per needs of the user and an example is present in `<sdk_x.y>/examples/using_hio`.
- d. For STM32 host, the package to be used is `I-CUBE-T2-STW-lib.zip` which contains both the Talaria TWO binary (`stw_multi_proto.elf`) and the Host applications.

## 5 Support

1. Sales Support: Contact an InnoPhase sales representative via email – [sales@innophaseinc.com](mailto:sales@innophaseinc.com)
2. Technical Support:
  - a. Visit: <https://innophaseinc.com/contact/>
  - b. Also Visit: <https://innophaseinc.com/talaria-two-modules>
  - c. Contact: [support@innophaseinc.com](mailto:support@innophaseinc.com)

InnoPhase is working diligently to provide customers outstanding support to all customers.

## 6 Disclaimers

**Limited warranty and liability** — Information in this document is believed to be accurate and reliable. However, InnoPhase Incorporated does not give any representations or warranties, expressed or implied, as to the accuracy or completeness of such information and assumes no liability associated with the use of such information. InnoPhase Incorporated takes no responsibility for the content in this document if provided by an information source outside of InnoPhase Incorporated.

InnoPhase Incorporated disclaims liability for any indirect, incidental, punitive, special or consequential damages associated with the use of this document, applications and any products associated with information in this document, whether or not such damages are based on tort (including negligence), warranty, including warranty of merchantability, warranty of fitness for a particular purpose, breach of contract or any other legal theory. Further, InnoPhase Incorporated accepts no liability and makes no warranty, express or implied, for any assistance given with respect to any applications described herein or customer product design, or the application or use by any customer's third-party customer(s).

Notwithstanding any damages that a customer might incur for any reason whatsoever, InnoPhase Incorporated' aggregate and cumulative liability for the products described herein shall be limited in accordance with the Terms and Conditions of identified in the commercial sale documentation for such InnoPhase Incorporated products.

**Right to make changes** — InnoPhase Incorporated reserves the right to make changes to information published in this document, including, without limitation, changes to any specifications and product descriptions, at any time and without notice. This document supersedes and replaces all information supplied prior to the publication hereof.

**Suitability for use** — InnoPhase Incorporated products are not designed, authorized or warranted to be suitable for use in life support, life-critical or safety-critical systems or equipment, nor in applications where failure or malfunction of an InnoPhase Incorporated product can reasonably be expected to result in personal injury, death or severe property or environmental damage. InnoPhase Incorporated and its suppliers accept no liability for inclusion and/or use of InnoPhase Incorporated products in such equipment or applications and such inclusion and/or use is at the customer's own risk.

All trademarks, trade names and registered trademarks mentioned in this document are property of their respective owners and are hereby acknowledged.