Presence detection and zoning solution

using XENSIV™ BGT60TR13C radar and CYW55913 Wi-Fi Bluetooth® MCU

About this document

Scope and purpose

This application note describes a proof-of-concept (PoC) for an embedded platform that combines a 60 GHz radar sensor for robust human presence detection and zoning implemented on a Wi-Fi & Bluetooth® LE Connected MCU. The platform comprises Infineon's XENSIV™ BGT60TR13C 60 GHz radar sensor shield mounted onto the AIROC™ CYW55913 Wi-Fi & Bluetooth® LE Evaluation Kit (EVK). In addition, Infineon Presence Sensing and Zoning Android mobile app is used to configure the radar zoning algorithm and view the presence information returned from it. This implementation reduces the need for separate MCUs for radar processing and connectivity, further reducing the board space and BoM cost.

Intended audience

The document is intended for design engineers, technicians, and developers of electronic systems who want to integrate a radar sensor for presence detection with an IoT device such as smart doorbell and security cameras but can extend to other smart home applications such as smart home lighting and smart TV.

Table of contents

1 Introduction

1.1 System overview

Prescence detection and zoning application provides an example of integrating the radar presence/zoning functionality using Infineon's XENSIV™ BGT60TR13C frequency modulated continuous wave (FMCW) radar sensor [1] with Infineon's AIROC™ CYW55913 Wi-Fi & Bluetooth® LE Connected MCU [2]. This solution includes the Infineon Presence Sensing and Zoning Android mobile app (released as an *.apk* file in the FW release package) to configure the radar zoning algorithm parameters, view the radar presence detection/zoning results, and perform Wi-Fi onboarding.

Passive infrared (PIR) sensors are the incumbent solution for presence detection. In applications such as security cameras or smart doorbells, presence detection sensors trigger the system to initiate video recording to capture evidence of presence. While PIR sensors are low-cost, they are susceptible to numerous issues such as:

- False detection – PIR sensors are susceptible to false detection due to moving objects or animals entering the field of view (FoV). Frequent false detections lead to increased power consumption and reduced confidence in the system reliability.
- Multiple hardware requirements - Present solutions require a dedicated sensor hub MCU to carry out radar communication and presence detection, and another MCU to control Wi-Fi & Bluetooth® LE connectivity.

To offset these issues, Infineon offers the AIROC™ CYW55913 MCU, which is Wi-Fi & Bluetooth® LE Connected MCU with a core suitable for applications development. Besides connectivity, it has an additional Arm® Cortex® M33 core that can be used for a custom user application, which was used for developing the radar zoning algorithm. The XENSIV™ BGT60TR13C radar sensor paired with a robust zoning algorithm provides accurate human presence detection up to 8 m with a FoV of ± 60° (total of 120°). This solution eliminates the need for a separate sensor hub MCU to address Wi-Fi & Bluetooth® LE functionality and radar data processing, thus reducing the overall cost and time to market.

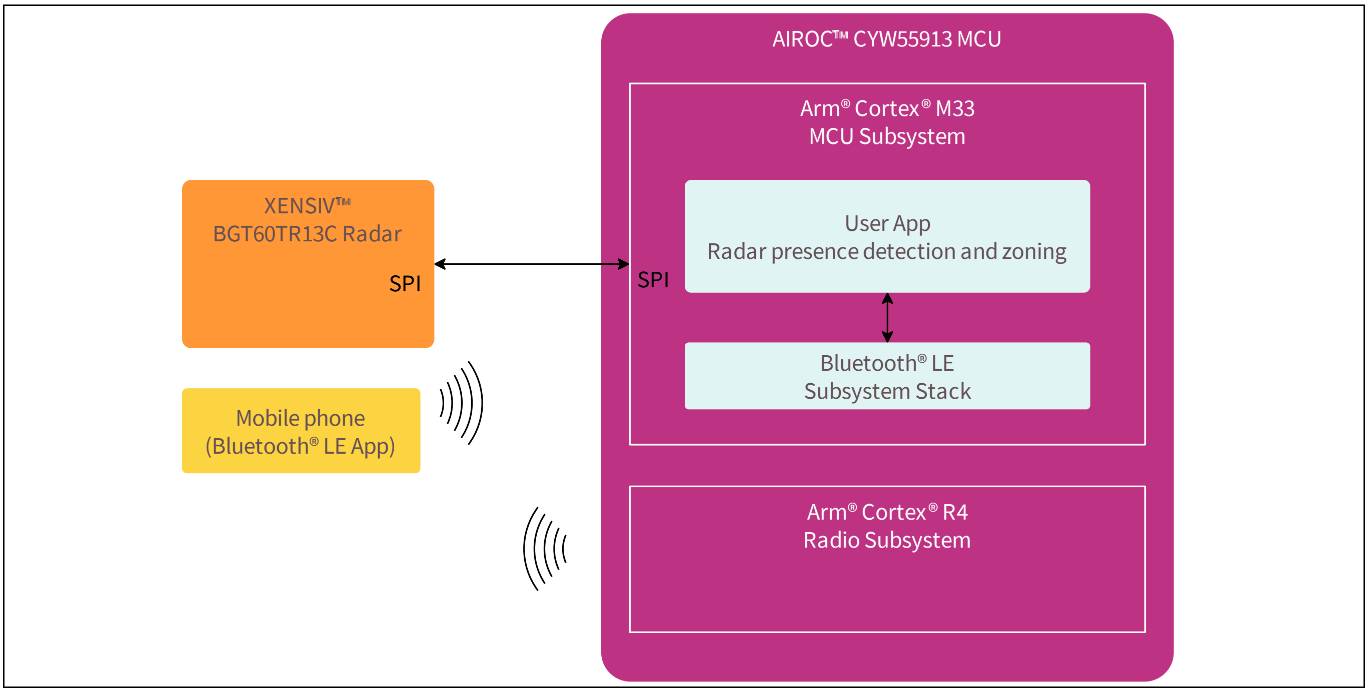


Figure 1 System block diagram

1.2 XENSIV™ BGT60TR13C radar

The XENSIV™ BGT60TR13C MMIC [1] is a 60 GHz FMCW radar with antenna-in-package (AIP), containing one transmit (Tx) and three receive (Rx) antennas, with an FoV of ±60°. Additionally, it has an integrated state machine for low-power and real-time operation, and 5.2 GHz ultra-wide bandwidth that allows FMCW operations with a high resolution.

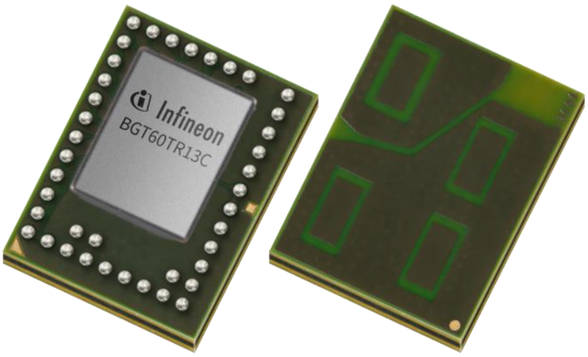


Figure 2 BGT60TR13C radar chip with AIP

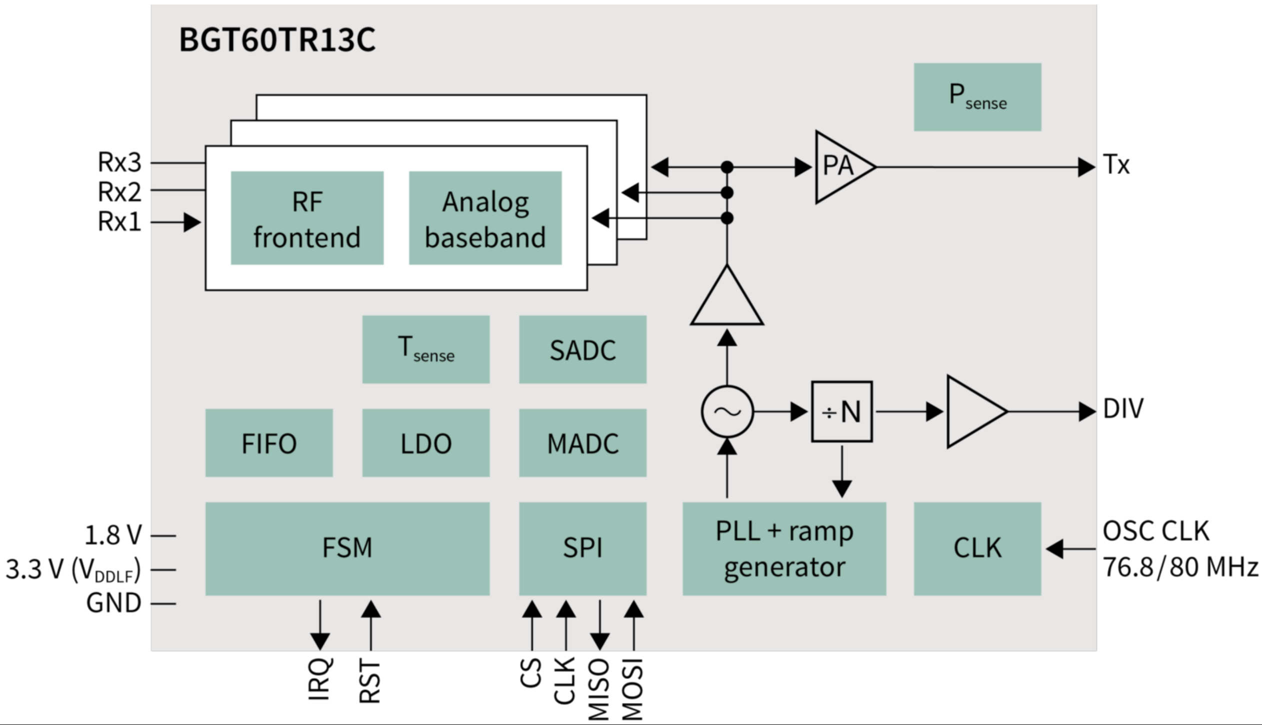


Figure 3 BGT60TR13C system block diagram

1.3 XENSIV™ radar adapter board

To interface with the BGT60TR13C radar shield [3], an XENSIV™ radar adapter board [4] is used. This board allows a connection with a 60 GHz radar shield such as BGT60TR13C (1 Tx, 3 Rx) or BGT60LTR11AIP (1 Tx, 1 Rx), usually used on Infineon's demo platforms, to any system. The XENSIV™ radar adapter board is based on the Adafruit Feather pin specification, which exposes the necessary pins to interface the radar with any platform.

The board includes:

- Serial Peripheral Interface (SPI) pins and various GPIOs that are required to control the radar
- Level shifters to translate the 3.3 V MCU logic level to the radar's 1.8 V logic level and LDOs for generating the necessary 3.3 V and 1.8 V power rails.

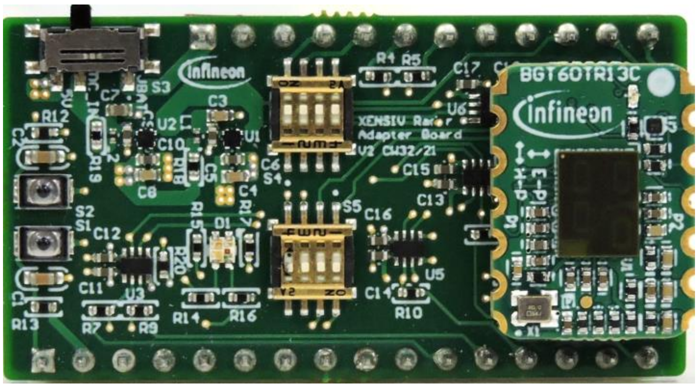


Figure 4 XENSIV™ radar adapter board with BGT60TR13C shield

1.4 AIROC™ CYW55913 Wi-Fi & Bluetooth® LE Connected MCU

Features:

- An ultra-low power single-chip connected MCU
- Supports 1x1 2.4+5+6 GHz Wi-Fi 6E, Bluetooth® Low Energy 5.4, and Matter
- Optimized for IoT including smart home, industrial, and portable devices, which can be used by itself or to offload connectivity from a host processor

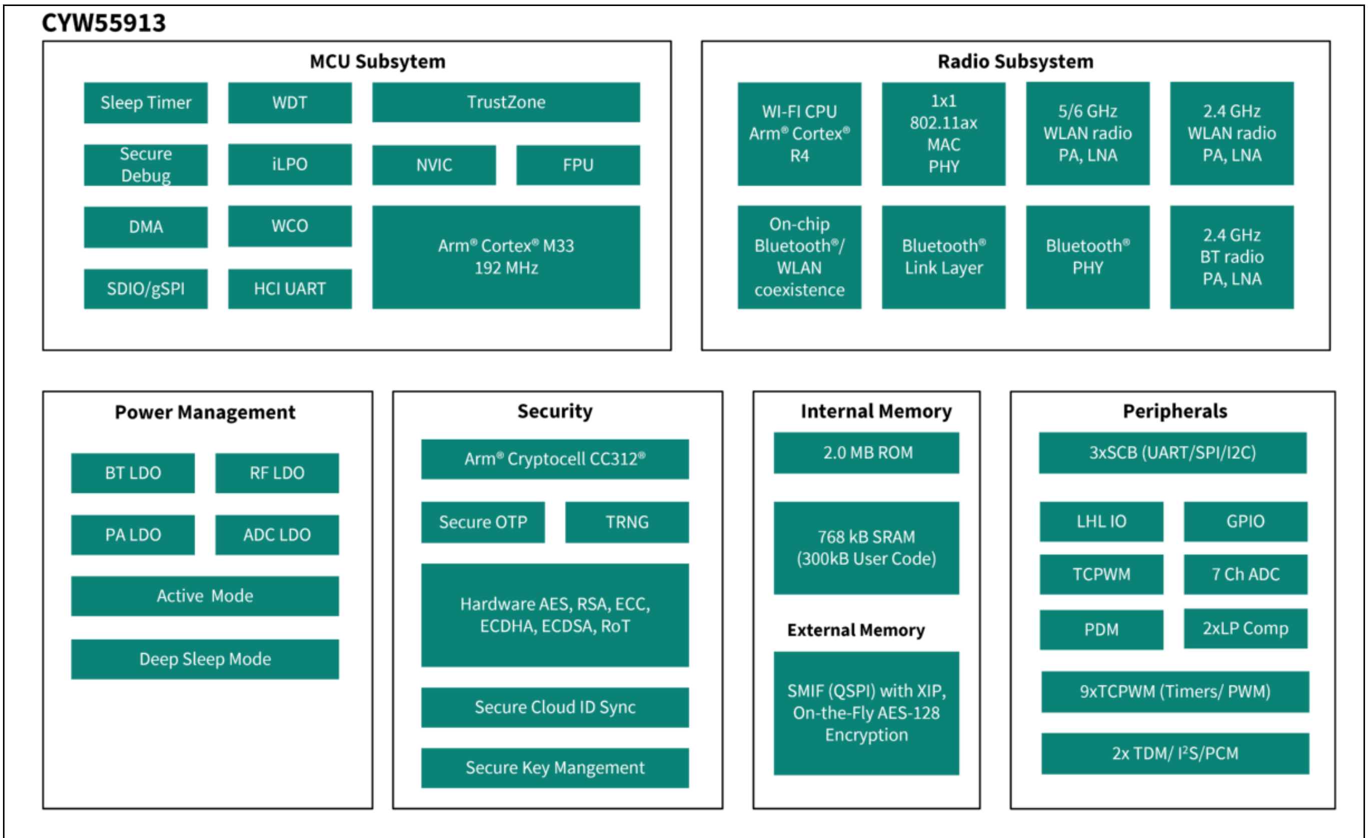


Figure 5 CYW55913 MCU subsystem diagram

1.5 AIROC™ CYW955913EVK-01 Evaluation Kit

The CYW955913EVK-01 Evaluation Kit [5] [6] is used as the baseboard for this application, which includes an M.2 slot to connect the CYW55913 MCU and multiple headers for easy access to different I/O.

Figure 6 AIROC™ CYW955913EVK-01 Evaluation Kit

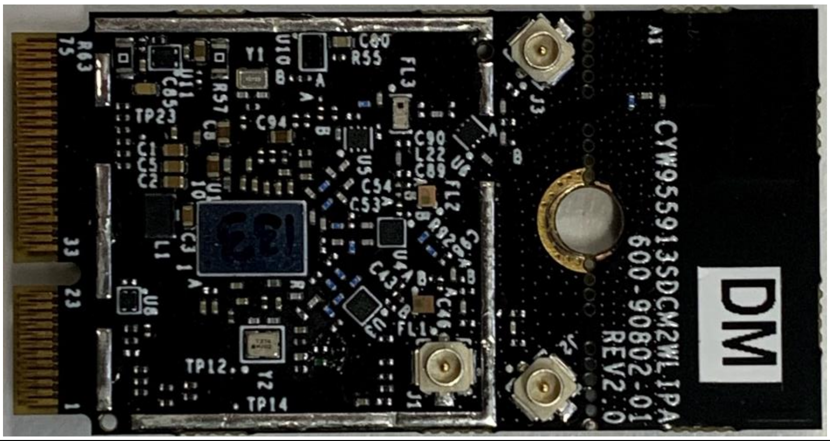


Figure 7 CYW955913SDCM2WLIPA module[1]

1.6 Infineon Presence Sensing and Zoning Android mobile app

The Android mobile app provides an interface for viewing human presence information from the CYW955913EVK-01 system and configuring the characteristics of the radar service (Table 3).

For more information, see Section 5.

[1] Not sold separately, only as part of CYW955913EVK-01.

2 Hardware

2.1 Prerequisites

The following hardware components are required for CYW55913:

- AIROC™ CYW955913EVK-01 Evaluation Kit [5] [6]
- AIROC™ CYW55913 Wi-Fi/Bluetooth® MCU plug-in module
- XENSIV™ BGT60TR13C radar shield [3]
- XENSIV™ radar adapter board

Perform the following steps to connect the hardware.

1. Connect the XENSIV™ BGT60TR13C radar shield to the connectors (P1 and P2) on the adapter board.
2. Ensure to match the dot marker in the upper left corner on the shield and adapter board as shown in Figure 8.

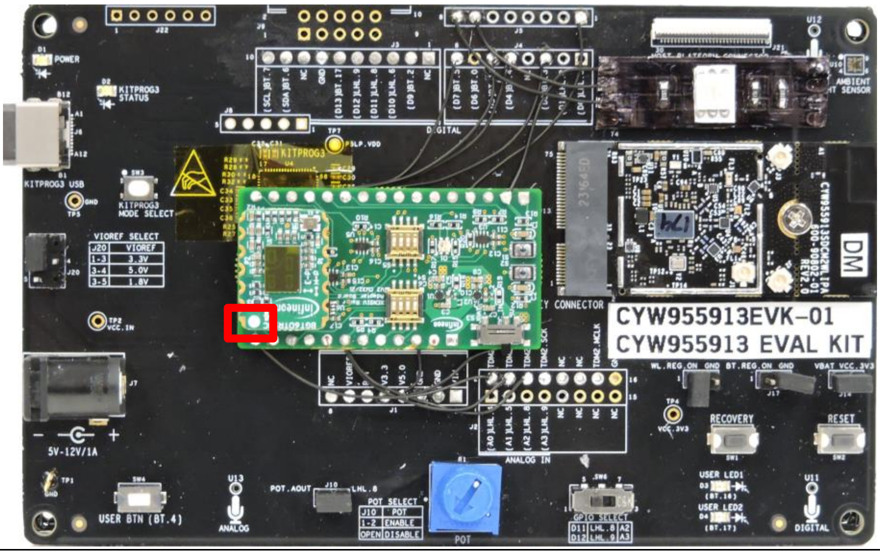


Figure 8 CYW955913EVK-01 with BGT60TR13C radar shield

2.2 Wiring instructions and pin configuration

You can configure and build your own system with the CYW955913EVK-01 baseboard, the BGT60TR13C adapter, and shield boards.

Note: AIROC™ CYW955913EVK-01 Evaluation Kit requires a run through a quick preparation process to achieve the system configuration shown in Figure 8.

2.2.1 Board preparation

1. For a clean wiring process, remove the following headers (identified by both their component identifiers and silkscreen markers on the CYW955913EVK-01 baseboard):

- POWER, J1 (1x8, single row)
- ANALOG IN, J2 (2x8, dual row)
- DIGITAL, J3 (1x10, single row)
- DIGITAL, J4 (1x8, single row)
- J5 (1x8, single row)

Table 1 lists the pin connections between CYW955913EVK-01 and the radar adapter board.

Table 1 Connectivity guide between radar adapter board and CYW955913EVK-01

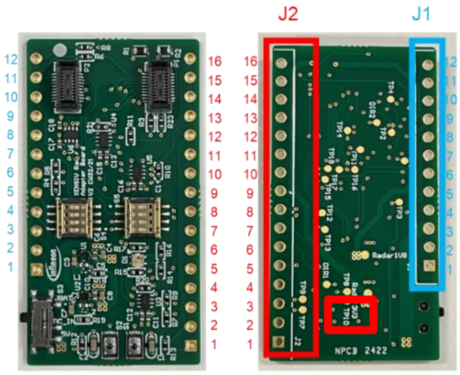
CYW955913EVK-01 signal			Radar adapter board signal		Description
Connector ID	Pin	On board	Connector ID	Pin	
J4.6	LHL_GPIO_6	D5[BT.16]	J2.11	SPI_CLK_Feather[1]	Radar SPI clock[2]
J2.4	TDM2_DO	TDM2_SDO	J2.12	SPI_MOSI_Feather[1]	Radar SPI MOSI
J2.6	TDM2_DI	TDM2_SDI	J2.13	SPI_MISO_Feather[1]	Radar SPI MISO
J2.2	TDM2_WS	TDM2_WS	J2.16	SPI_CSN_Feather[1]	Radar SPI Chip Select
J4.5	BT_GPIO_2	D4[BT.4]	J1.10	en_LDO_Radar[1]	Radar LDO Enable
J4.3	BT_GPIO_3	D2[BT.3]	J1.6	RST_Feather[1]	Radar Reset
J4.8	BT_GPIO_5	D7[BT.5]	J1.7	IRQ_Feather[1]	Radar IRQ
J1.3	–	GND	J2.4	GND[1]	Ground connection
J1.4	–	V5.0	J1.3	5 V[1]	5 V power supply
J1.5	–	V3.3	J2.2	3V3[1]	3.3 V power supply

[1] Signal naming convention as per circuit schematic net names; there are no individual silkscreen indicators on the adapter board.

[2] In Rev. 2 of the baseboard, J3.3 (LHL_GPIO_6) is swapped for J4.6 (BT_GPIO_16). In the firmware, SPI CLK is set to LHL_GPIO_6. However, you should connect it to BT_GPIO_16 on J4.6.

2.2.2 Building the system

1. Once all boards have been prepared, follow Table 1 and Figure 9 wiring instructions to build the system.



BGT60TR13C Adapter Board

CYW955913EVK-01 Base Board

- Connect J1-3 (5V) of the BGT60TR13C adapter board to J1-4 of CYW955913EVK-01 base board (5V)
- Connect J1-6 (RESET) of BGT60TR13C adapter board to J4-3 of CYW955913EVK-01 base board (D2[BT.3])
- Connect J1-7 (INT) of the BGT60TR13C adapter board to J4-8 of CYW955913EVK-01 base board (D7[BT.5])
- Connect J1-10 (LDO_EN) of the BGT60TR13C adapter board to J4-5 of CYW955913EVK-01 base board (D4[BT.4])
- Connect J2-2 (3.3V) of the BGT60TR13C adapter board to J1-5 of CYW955913EVK-01 base board (3.3V)
- Connect J2-4 (GND) of the BGT60TR13C adapter board to J1-2 of CYW955913EVK-01 base board (GND)
- Connect J2-11 (SCLK) of the BGT60TR13C adapter board to J4-6 of CYW955913EVK-01 base board (D5[BT.16])
- Connect J2-12 (MOSI) of the BGT60TR13C adapter board to J2-4 of CYW955913EVK-01 base board (TDM2_SDO)
- Connect J2-13 (MISO) of the BGT60TR13C adapter board to J2-6 of CYW955913EVK-01 base board (TDM2_SDI)
- Connect J2-16 (CS) of the BGT60TR13C adapter board to J2-2 of CYW955913EVK-01 base board (TDM2_WS)

Figure 9 Wiring instructions for CYW955913EVK-01 with BGT60TR13C adapter board and radar shield

2. To observe the presence and absence activity, wire an RGB LED strip into the outlined connections/GPIOs where J5 was located on the CYW955913EVK-01 baseboard (Figure 10).
3. Ensure the following connections:

- Connect the “+” pad of the RGB LED strip to J2.2 (3V3) on the adapter board
- Connect the “G” pad of the RGB LED strip to J5-7 (TDM1.DO on board) on the CYW955913EVK-01
- Connect the “R” pad of the RGB LED strip to J5-8 (TDM1.WS on board) on the CYW955913EVK-01

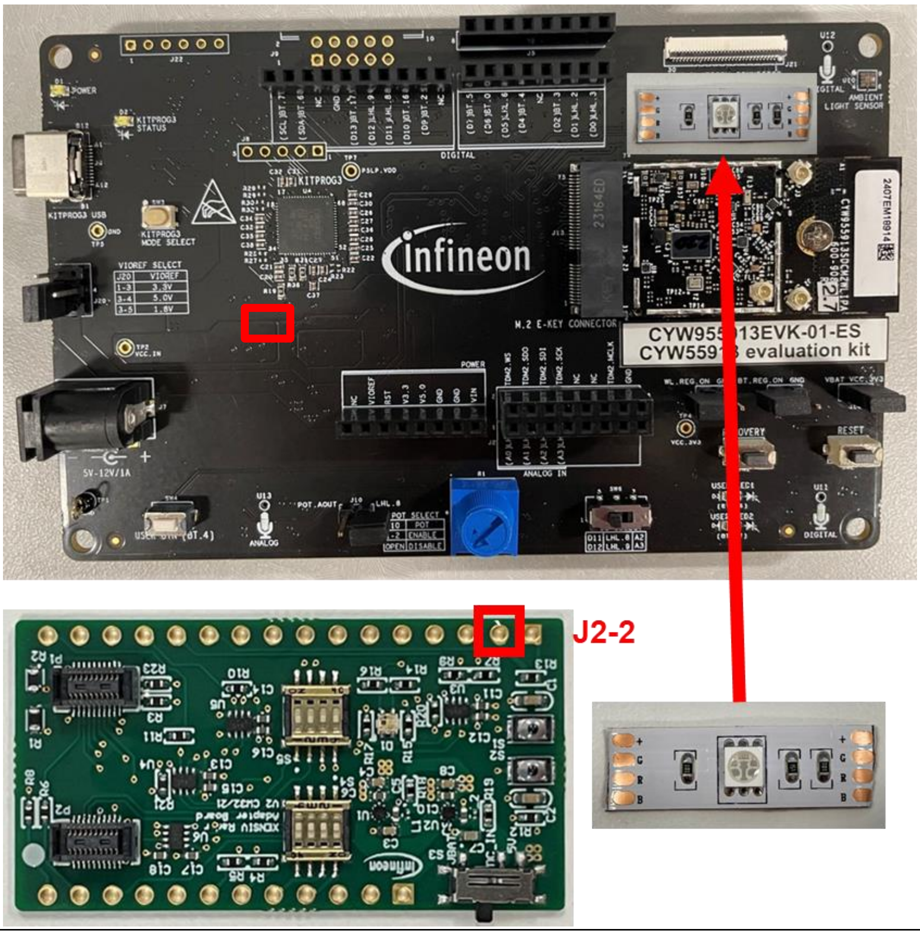


Figure 10 Wiring RGB LED strip into CYW955913EVK-01 baseboard for presence/absence visualization

3 Software

3.1 Prerequisites

The following software components are required for this application:

- ***Infineon_BGT60TR13C_CYW55913_Presence_detection_zoning_Release_Rev0.0.3.zip***.
  This release package includes the following:
  - Firmware *.hex* file and the required support files needed to flash the firmware
  - Infineon Presence Sensing and Zoning *.apk* file to install the application on an Android device (See Section 5)
  - MQTT Subscriber Example as a Python application for Windows PC (See Section 6)

3.2 Software flowchart

During the initial boot, the Bluetooth® Low Energy stack is initialized and started. After the initialization, two radar tasks begin:

- Radar Acquisition task
- Radar Processing task

The Radar Acquisition task reads the radar data from its FIFO buffer when an interrupt signal is received indicating that the data is ready and notifies the Radar Processing task. The Radar Processing task then feeds the frame to the zoning algorithm. When the targets are detected, the Bluetooth® Low Energy stack APIs notify the Android mobile app and an MQTT broker connected through Wi-Fi about the detected presence.

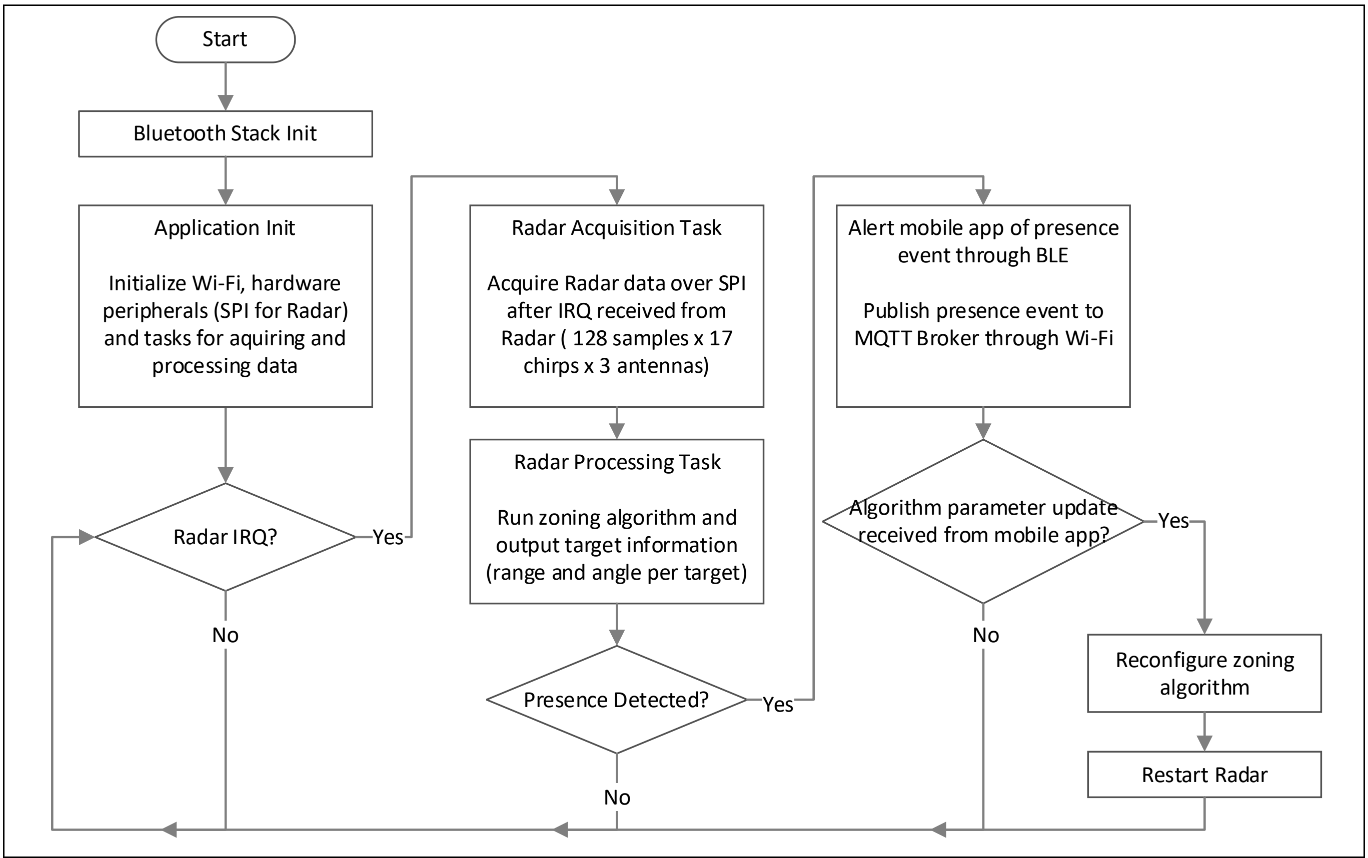


Figure 11 Software flowchart

3.3 Radar algorithm

The radar algorithm used in this design is based on Infineon's zoning algorithm. This algorithm takes the input radar data from BGT60TR13C and estimates the Range Doppler spectrum and angle of arrival (AoA) of the moving targets within the field of view. Additionally, enhanced features such as rejection of unwanted targets (e.g., plants, inanimate moving objects such as a fan) using Doppler masking techniques are also implemented.

This algorithm expects a frame size of 6528 samples (128 samples per chirp, 17 chirps, and 3 antennas). The output of this algorithm includes the following:

- Number of detected targets
- Distance of each detected target
- Angle of detection of each target

3.3.1 Waveform configuration

Table 2 lists the radar waveform configurations used for this application:

Table 2 Waveform configuration

Parameter	Value
Mode	Range Doppler (RD)
Samples per chirp	128
Chirps per frame	17
Number of Rx antennas	3
Frame repetition time	100 ms (10 frames/s)
Chirp repetition time	650 µs
Total frame size	128 x 17 x 3 = 6528 samples

3.3.2 Memory footprint and performance

The algorithm combined with the radar waveform outlined in Section 3.3.1 are profiled to have the following memory footprint and performance:

- **Flash:** 650 KB
- **RAM:** 157 KB
- **Processing time:** 22 ms - 25 ms at 192 MHz system clock on Arm® Cortex® M33

Note: *The Arm® Cortex® M33 CPU is active for 25% of the time as the radar runs at 10 fps (100 ms per frame). Hence, Arm® Cortex® M33 can be in Sleep mode for the remaining duration.*

3.4 GATT configuration (ModusToolbox™)

Generic Attribute Profile (GATT) defines the way two devices (in this case the mobile app and CYW955913EVK-01) communicate and exchange data with each other. This section introduces Bluetooth® device topology and the usage of GATT.

3.4.1 Bluetooth® device topology

Bluetooth® Low Energy is a lightweight subset of the Bluetooth® Classic communications and was introduced as part of the Bluetooth® 4.0 core specification. It is designed for applications that do not require handling heavy data and is widely used in applications such as home automation and fitness tracking.

To transmit data between each other, Bluetooth® Low Energy devices must first form a communication channel using Generic Access Profile (GAP), a framework that defines how devices interact with each other. For two devices to connect and communicate using GAP, one must take the role of **Central**, and the other, **Peripheral**.

The Central device scans for nearby Peripheral devices, while the Peripheral device announces its presence by broadcasting advertising packets. Once the Peripheral device connects to a Central device, it no longer broadcasts data to other Central devices and stays connected to the device that accepted the connection request. In this example, the Central device is the mobile app, and the Peripheral device is CYW955913EVK-01.

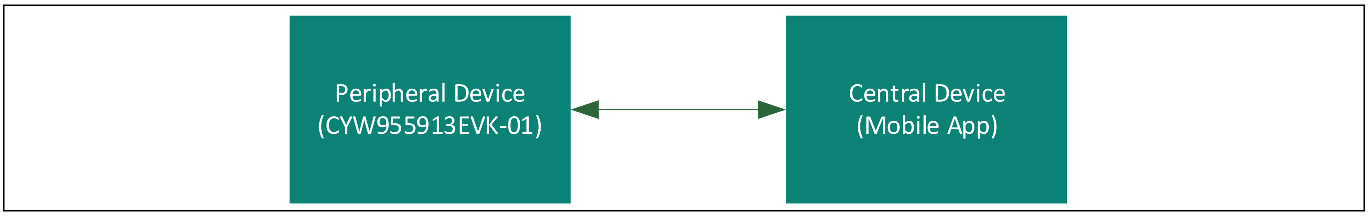


Figure 12 Bluetooth® device topology

3.4.2 GATT description

To exchange data with the Central device, the Peripheral device needs to set up a database with various settings called “Services” and “Characteristics”.

Services - are a way to organize data into different logical blocks and can be thought of as “Classes” and Characteristics as “Class Variables” in object-oriented programming (OOP). Characteristics - are data points within the service that is tied to a specific setting or variable. Each Characteristic has properties for permissions: read, write, or notify.

From a GATT standpoint, when two devices are connected, each is in one of the two roles:

- **GATT Server:** The Peripheral device containing the Characteristic database that is being accessed to by a GATT Client
- **GATT Client:** The Central device that is accessing data from or writing to the GATT Server. All transactions are started by the GATT Client, which receives responses from the GATT Server

3.4.3 GATT settings

Radar Service, used for presence and zoning application consists of the following characteristics:

Table 3 Characteristics of Radar service

Characteristics	Unit	Values supported	Default value	Permissions	Description
Motion sensitivity	–	LOW, MEDIUM, HIGH	MEDIUM	Read/Write	Adjusts the sensitivity of presence detection. HIGH = strong sensitivity.
Minimum range	[m]	0.5 to 9	0.5	Read/Write	Adjusts the minimum distance for presence detection. All targets below the minimum range are filtered out.
Maximum range	[m]	2 to 10	10	Read/Write	Adjusts the maximum distance for presence detection. All targets above maximum range are filtered out.
Doppler mask	–	0 (Deselected: mask on), 1 (Selected: mask off)	1,1,1,1,1,1,1,1, 1,1,1,1,1,1,1,1 Mask off	Read/Write	Provides a way of clutter rejection by filtering targets in configured Doppler grids. Each grid represents 0.23 m/s Doppler speed resolution.
Mode	–	Range Doppler (RD)	RD	Read/Write	Customizes the algorithm mode. RD provides the target count, distance, and angle.
Re-trigger time	[s]	0 to 60	0	Read/Write	Adjusts the time after a presence detection event during which the radar will be inactive.
Zone selection	–	0 (Deselected) 1 (Selected)	All zones selected	Read/Write	Configures the radar FoV via selection of zones. Only targets within the selected zones are shown; others are filtered.
Range angle info	[m], Degrees	0.5 to 10, -60 to 60	–	Read/Notify	Acquires the range and angle information from the zoning algorithm.

3.5 MQTT

MQTT (MQ Telemetry Transport) is a lightweight internet messaging protocol designed to minimize network bandwidth and device resource requirements. It is a publish/subscribe protocol commonly used for M2M (machine-to-machine) and IoT (Internet of Things) applications.

MQTT consists of MQTT Clients and MQTT Brokers. MQTT Clients can publish messages to MQTT Brokers or receive messages from the MQTT Broker. The MQTT Broker acts as a hub for the messages that are published to it from MQTT Clients and dispatches the messages based on the message topics which receiving MQTT Clients are subscribed to.

CYW955913EVK-01 connects to MQTT using Wi-Fi, which uses Bluetooth® LE onboarding to send a Wi-Fi SSID and password from the Android mobile app (See 5.4.5.3 for onboarding steps). CYW955913EVK-01 is a client that publishes the detection events from the radar algorithm to the AWS IoT Core Broker. The MQTT Subscriber Example that runs on Windows PC is a client that subscribes to the detection events from the AWS IoT Core Broker.

3.5.1 MQTT Client and Message

CYW955913EVK-01 connects to the MQTT Broker with a unique client ID prefixed with “CYW955913-client-”. It publishes messages to the “CYW955913EVK-01 RADAR EVENT” topic whenever a new detection event occurs in the radar algorithm. The messages published from CYW955913EVK-01 are formatted with the Client ID, timestamp, detection status, and an array of up to two detected radar targets. Each of the targets have a Range, Range Zone, and Angle Zone field. See Table 4 for the JSON format details and Figure 13 for examples of the MQTT message as viewed on the AWS test client.

The range and angle zones correspond to those used for the Radar Screen in the mobile app, see Section 5.4.3 for details.

Table 4 JSON format of the MQTT Messages published to “CYW955913EVK-01 RADAR EVENT” topic

Name	Unit	Values supported	Type	Description
Client ID	–	-	String	Unique MQTT Client ID number generated in FW, device connects to MQTT with this value prefixed with “CYW955913EVK-01-client-”.
Message ID	–	–	Uint32	ID for tracking MQTT messages, incremented when the CYW955913EVK-01 attempts to publish a message
Timestamp ms	ms	–	UInt32	Timestamp in ms from device start up
Detection	–	PRESENCE, ABSENCE	String	Detection status from radar algorithm
Targets	–	–	–	Array of detected radar targets (up to 2 targets), each targets has a Range, Range Zone, and Angle Zone field
Range	meters		Float	Range of detected target in meters
Range Zone	–	1-15	Uint8	Range zone of detected target
Angle Zone	–	0-5	Uint8	Angle zone of detected target (each zone is

The MQTT Python Subscriber Example in MQTT Python subscriber example is also a MQTT Client that connects to the same MQTT Broker. It subscribes to the “CYW955913EVK-01 RADAR EVENT” topic and is able to receive the MQTT messages the MQTT broker has received from the CYW955913EVK-01.

3.5.2 MQTT Broker

The AWS IoT Core service is used as the MQTT Broker, AWS IoT (Amazon Web Services Internet of Things) provides cloud services and devices support for the implementation of IoT services. It is also used to manage the devices connecting to the MQTT Broker with security certificates and policies which allow for a secure connection.

Instructions for Getting started with AWS IoT Core are available online. AWS IoT Core offers additional services such as a MQTT test client and device management.

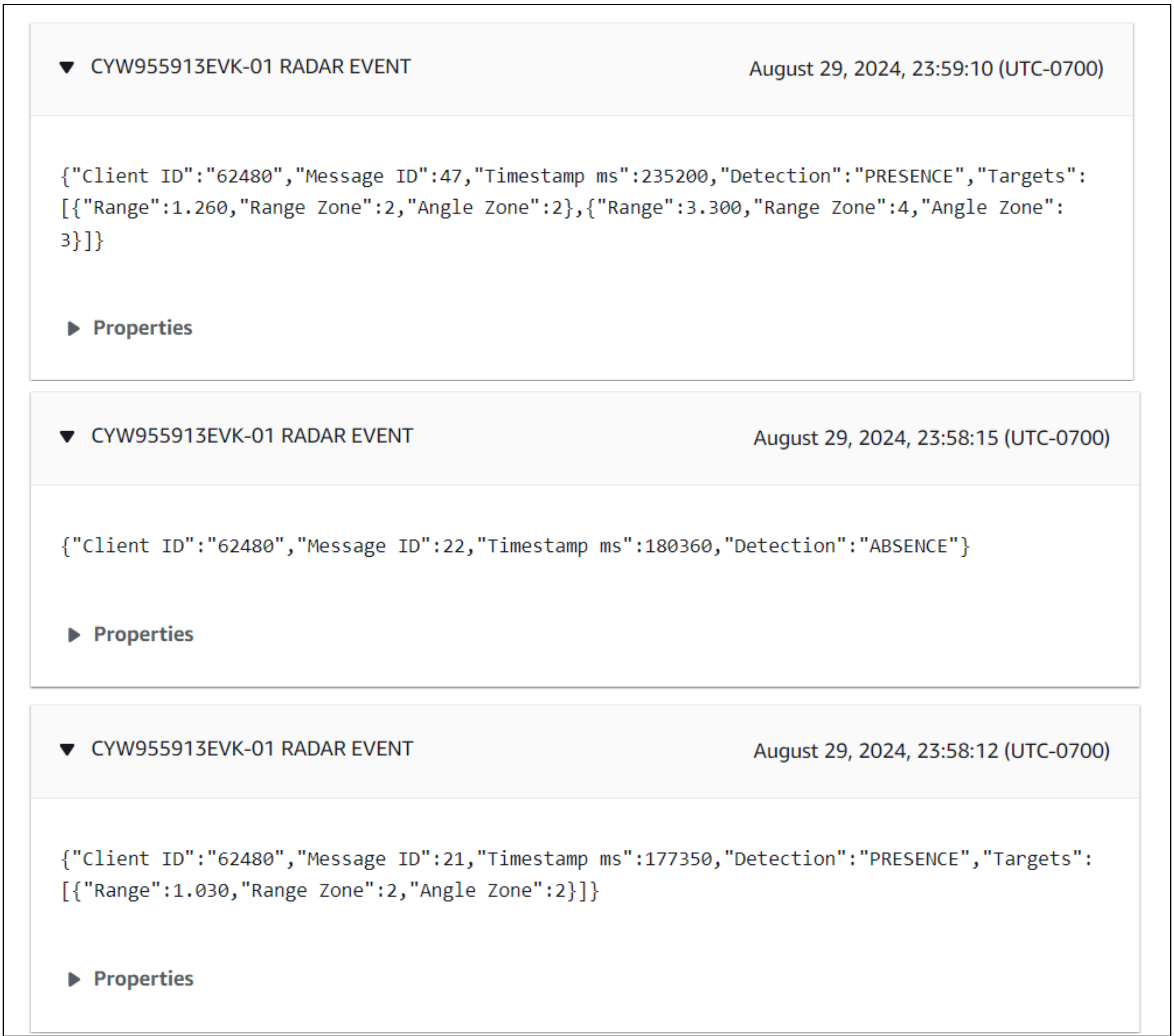


Figure 13 MQTT JSON message examples as seen on AWS IoT Core's MQTT Test Client

4 Flashing the firmware

Perform the following steps to flash the firmware.

1. Download and extract the firmware package *Infineon_BGT60TR13C_CYW55913_Presence_detection_zoning_Release_Rev0.0.3.zip* from Presence detection and zoning solution using XENSIV BGT60TR13C radar.
2. Connect the EVK to a PC via the USB-C connector labelled KITPROG3 USB. See Figure 14.

Figure 14 KITPROG3 USB on CYW955913EVK-01 evaluation baseboard

3. On Windows, navigate to **Device Manager** > **Ports** to identify the associated COM port number.
4. Press and hold the **Recovery** button on the EVK, and then press the **Reset** button. Release the **Reset** button after 1 s, then release the **Recovery** button.

EVK is ready for flashing. See Figure 15.

Figure 15 Recovery and Reset buttons

5. Open the **Command** prompt and navigate to extracted folder/FW_v0.0.3 to run the following:

   Replace ***xyz*** with the COM port number and replace *<firmware hex file>* with filename of the *hex* file you want to flash. A pre-compiled hex file for this application is already provided in *FW_v0.0.3* folder.

Code listing 1 Flashing firmware

```
flash_firmware.bat COM<xyz> <firmware hex file>
```

```
C:\Users[redacted]\Desktop\CYW955913EVK_BGT60TR13C_ZONING_ALGORITHM>flash_firmware.bat COM82 TR13C_17Chirps.hex

C:\Users[redacted]\Desktop\CYW955913EVK_BGT60TR13C_ZONING_ALGORITHM>set arg1=COM82

C:\Users[redacted]\Desktop\CYW955913EVK_BGT60TR13C_ZONING_ALGORITHM>set arg2=TR13C_17Chirps.hex

C:\Users[redacted]\Desktop\CYW955913EVK_BGT60TR13C_ZONING_ALGORITHM>ChipLoad.exe -BLUETOOLMODE -ENTERDOWNLOADMODE -
PORT COM82 -LAUNCHADDRESS 0x00000000 -BAUDRATE 115200 -NOVERIFY -MINIDRIVER minidriver.hex -BTP h1cp.btp -CONFIG TR13C
_17Chirps.hex -DL_TIMEOUT_MULTIPLIER 16
Download minidriver successfully had written 240 bytes to address 0x00300400
Download minidriver successfully had written 240 bytes to address 0x003004F0
Download minidriver successfully had written 240 bytes to address 0x003005E0
```

Figure 16 Flashing firmware: Initiation

```
25 BB 85 1B 9D B1 71 37 C1 AE 42 FC 82 E0 E9 C5 9B F9 41 1C 4B 01 E6 B0 D0
Download config successfully had written 180 bytes to address 0x006AFC40:  81 A0 C4 E9 05 23 C3 BE 02 F2 75 DC 75 9C B4
70 09 68 8B F4 40 7F 4A 16 79 6D 47 9F 14 B2 76 44 82 55 2F 82 AF 69 08 06 42 CF FA 7E F4 AB 3E FA B4 EF 6D 88 52 36 9B
05 99 10 FF 6E D0 C9 B6 88 83 25 4D FA E7 6D 84 D9 AD 60 78 4B 67 91 F9 E7 0D 53 FA BB E6 AB 61 C1 66 4A 3F 71 31 99 BD
5C 91 7F 16 92 BD FF 80 62 5D FE E6 F1 F1 19 46 71 6E 06 BA E2 81 12 18 7D E2 92 D2 16 8A 60 75 7F 1F A8 77 B6 ED B1 8F
B5 BC 16 57 3D 65 2F 93 11 0F 6B A4 4C 1B 6A 54 28 C5 DE E0 9E E3 92 0E E4 F5 E0 89 06 BE A9 C3 A1 FF FF FF FF FC D4 00
00 00 00 00 00
Chip reset to address 0x00000000 succeeded
Downloaded 0 code bytes ( 0.0%) and 195542 data bytes (100.0%). Verified 0 code bytes ( 0.0%) and 0 data bytes ( 0.0%).
  Current state: Completed successfully


A total of 4 contiguous memory areas were filled:
[00000000..0000002F] DATA (48 bytes)
[00600000..00600041] DATA (66 bytes)
[00680000..006AE06F] DATA (188528 bytes)
[006AE200..006AFCF3] DATA (6900 bytes)
Total execution time: 29.27 seconds     Net downloading time: 29.27 seconds
```

Figure 17 Flashing firmware: Completion

5 Infineon Presence Sensing and Zoning Android mobile app

The Infineon Presence Sensing and Zoning mobile app for Android interfaces with CYW955913EVK-01 via Bluetooth® and provides options to configure the characteristics of the radar service (Table 3).

Note: *This Android app is not available on Google Play Store and is released as an .apk file in the FW release package. Ensure that the firmware is flashed on CYW955913EVK-01 before using the app.*

5.1 Software requirements

- **Android OS:** Minimum 12 (API level 32), latest recommended

5.2 Hardware requirements

- Android device with Bluetooth® 4.0 or later

5.3 Installation

1. Connect the Android device to the PC via USB cable.
2. Copy and paste the released *.apk* to the Android device.
3. Click on the *.apk* to install it.

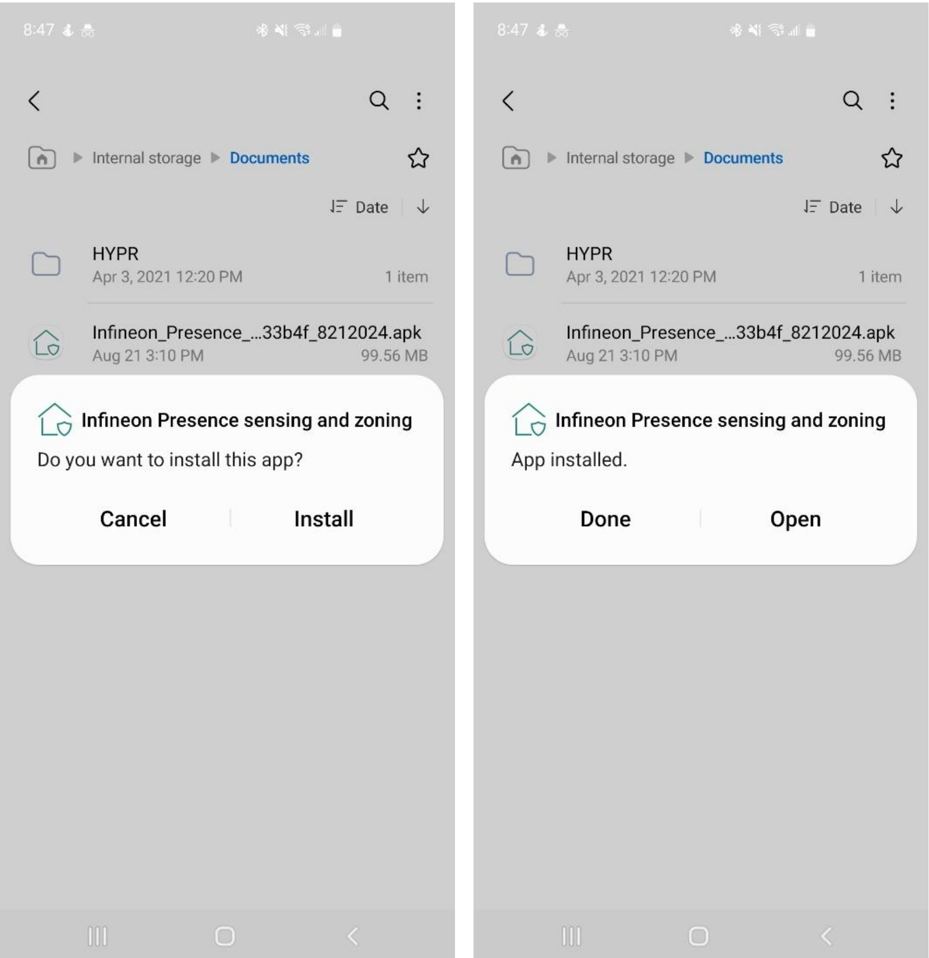


Figure 18 **App installation**

5.4 Application UI description

This section describes the app layout along with various tools and their functionalities.

5.4.1 Navigation bar

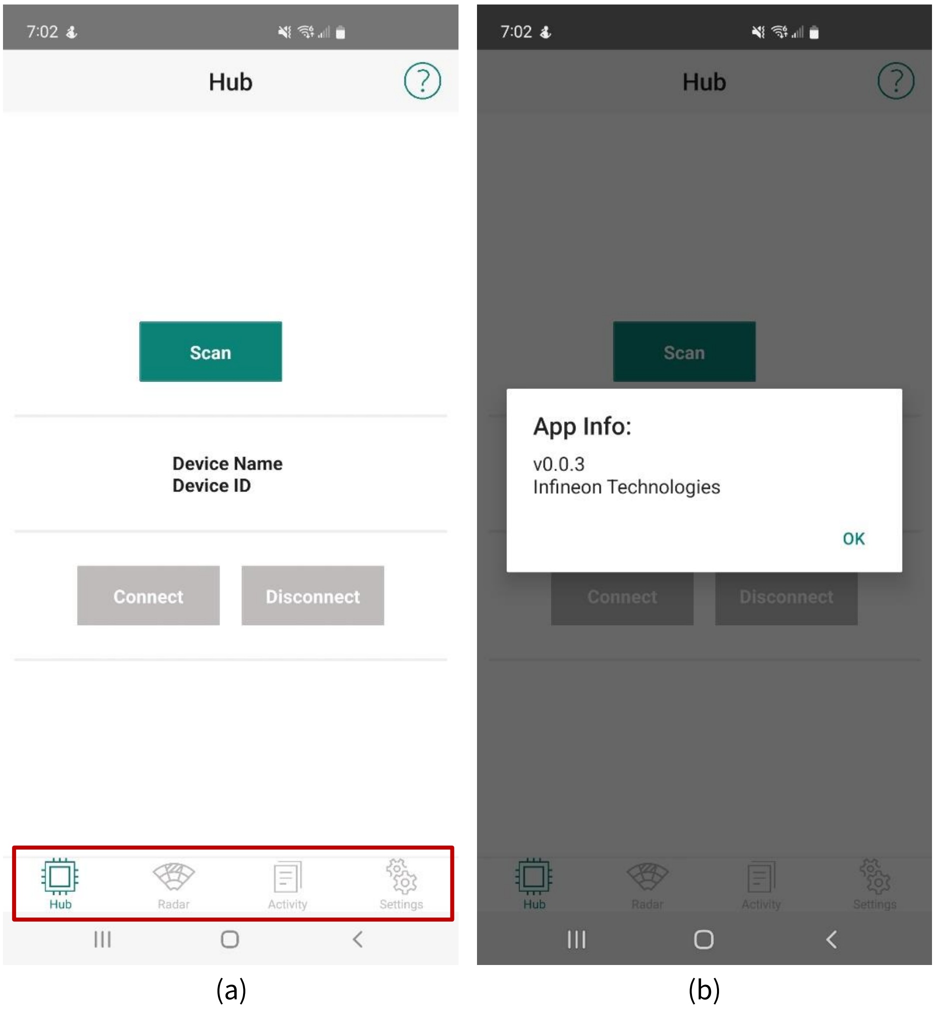


Figure 19 (a) Navigation bar and (b) App Info window

The following are available in the Navigation bar:

- **Hub screen:** Provides option to scan, connect, disconnect
- **Radar screen:** Provides option to read the characteristics of the radar service (Table 3)
- **Activity screen:** Displays logs of activity from each screen
- **Settings screen:** Provides option to configure the characteristics of the radar service (Table 3) and perform Wi-Fi onboarding on CYW955913EVK-01

5.4.2 Hub screen

1. Press the **Reset** button on EVK to start the firmware.
2. Press **Scan** to discover the nearby Bluetooth® LE devices including the EVK. The scanning period lasts for 10 seconds.
3. Allow permissions when asked.

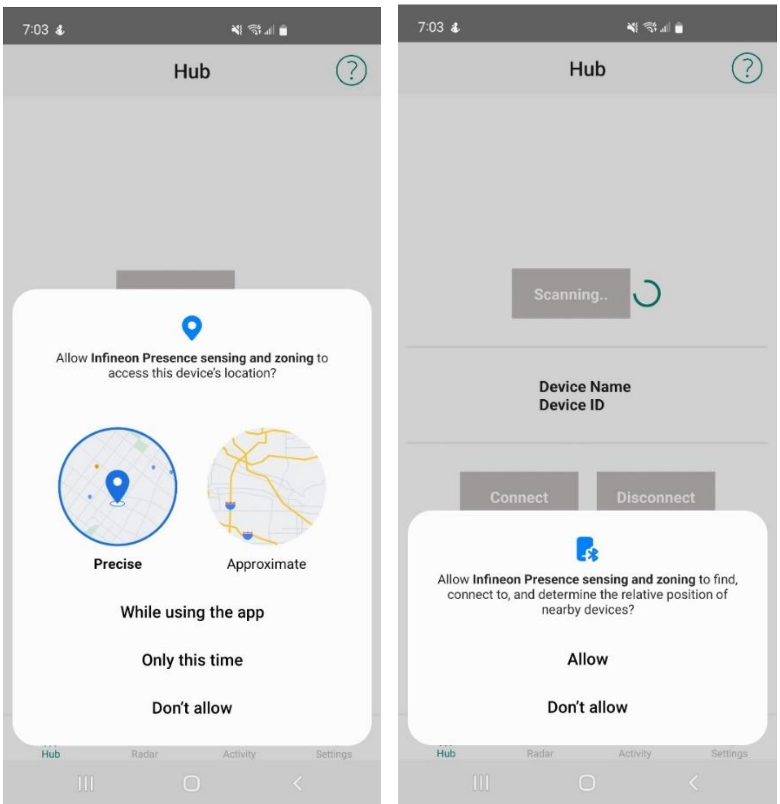


Figure 20 User permissions

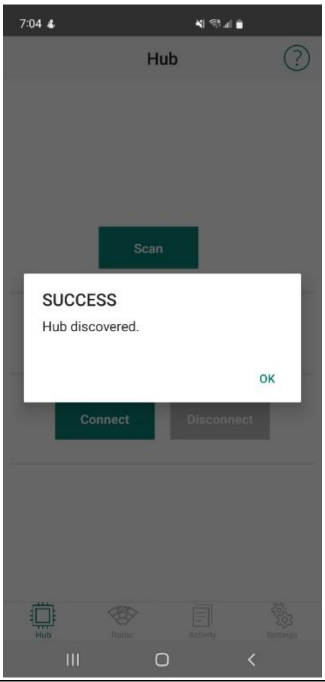


Figure 21 Discovery: Successful

4. After the EVK discovery, an alert is displayed and the **Connect** button is enabled.

Note: If discovery is unsuccessful, repeat scanning for multiple iterations.

5. EVK will close its BLE advertisement after 120 seconds. Ensure that the scanning is started as soon as reset is pressed.
6. Press **Connect** to establish the connection with EVK. If successful, an alert is displayed. If connection is unsuccessful, repeat this step for multiple iterations.

Note: *At this stage, **Connect** button is disabled and the **Disconnect** button is enabled. If the app is forcefully closed and reopened after successful connection, the information on scanned and connected devices is not retained; the entire process must be repeated, starting with resetting the EVK.*

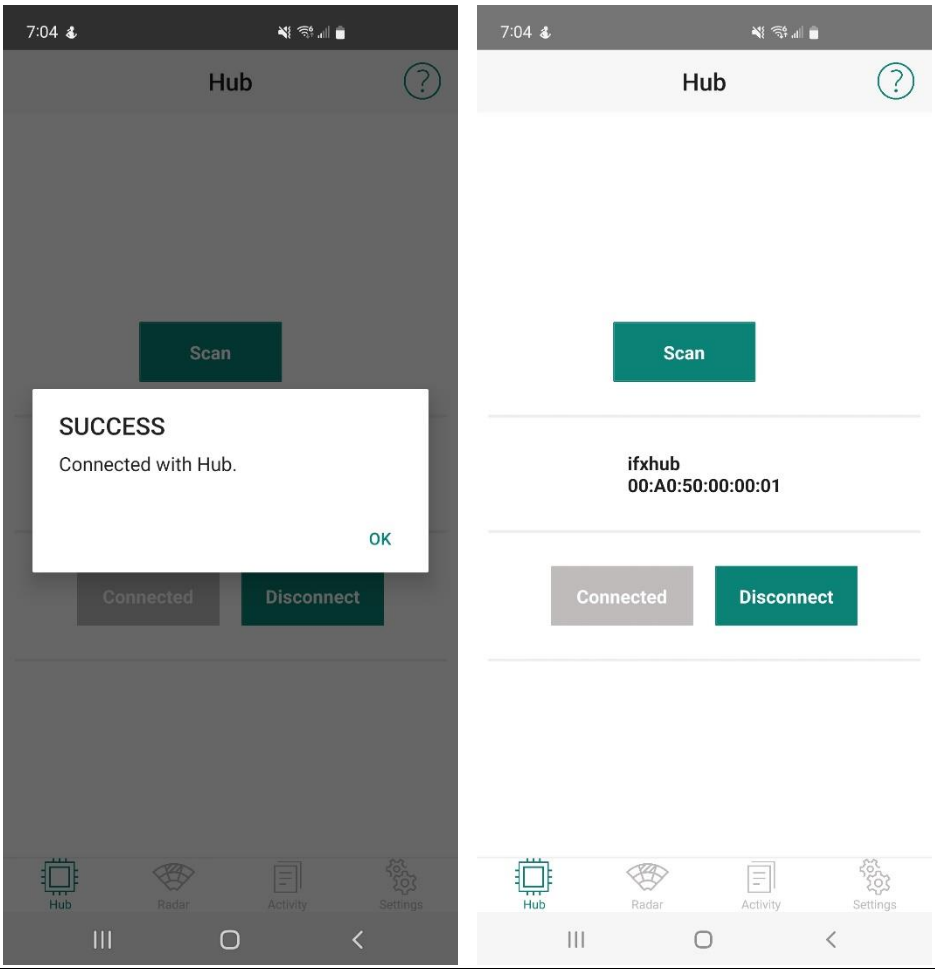


Figure 22 Connection: Successful

Note: *App can only be connected to a single EVK at any given time. To connect with another EVK, disconnect from the already connected EVK.*

5.4.3 Radar screen

The Radar screen provides a drop-down list to read the characteristics of the Radar Service (Table 3).

1. To read the characteristics, select a setting and tap **Read**.
   If successful, the value is displayed in the textbox and a toast message is displayed.

*Note: The **Read** button is disabled by default; it is only enabled when EVK connection is successful.*

Additionally, this screen displays the Radar Field of View (FoV) zones. The zones configured in the radar zone selection are displayed (Figure 29) with each zone representing a range resolution of 1 m and angular resolution of ±10°. The leftmost zone represents 60° while the rightmost zone represents -60°.

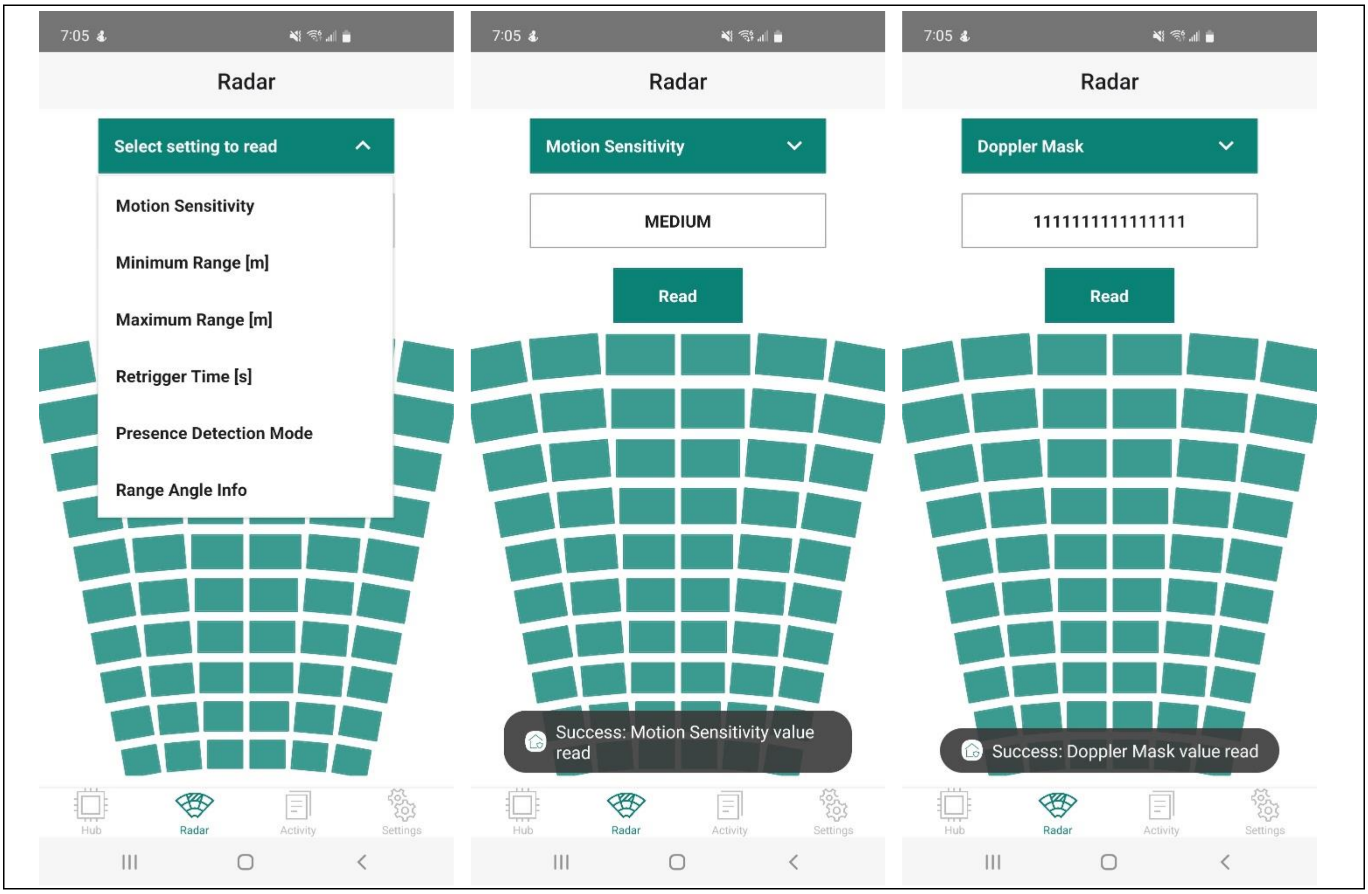


Figure 23 Reading characteristics value: Successful

5.4.3.1 Range Angle Info

The range and angle information of the target in the radar FoV is received every second and displayed in the textbox. Additionally, presence is highlighted by a change of respective zone color. Acquisition is continuous until the **Stop** button is pressed. EVK firmware supports the range and angle information for up to two targets.

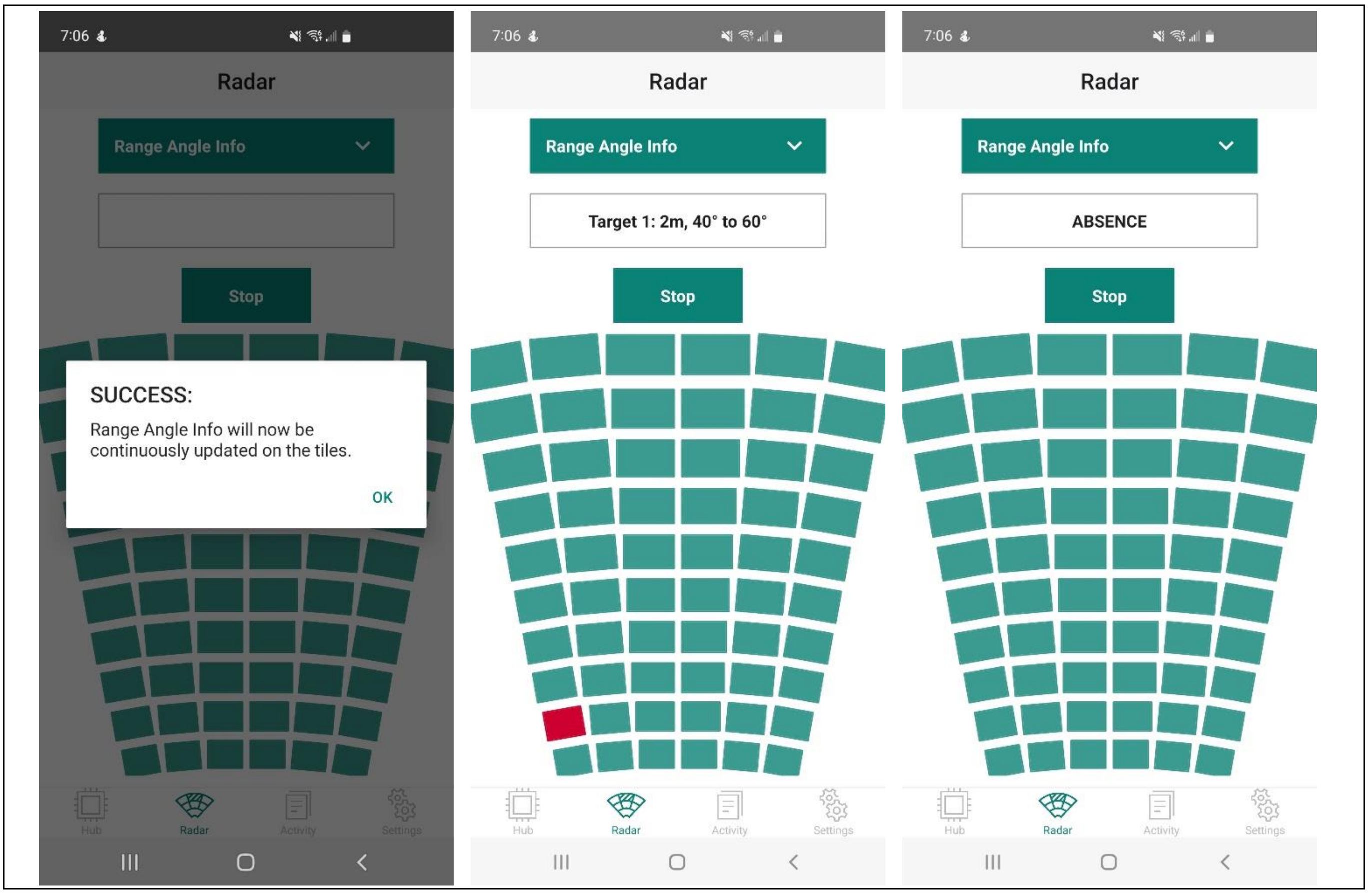


Figure 24 Range angle info: Acquisition

Note:

1. *As data is received continuously every second, UI components like button press may become unresponsive. In this case, press **Stop**.*
2. *App may lag a few seconds before termination is successful. In this case, restart acquisition.*

5.4.4 Activity screen

The Activity screen generates logs from each activity performed on the mobile app, such as button presses, slider movements, etc. along with a timestamp for when each activity was performed. The logs are displayed in the order most recent to the least.

The following is the Log format displayed in the Activity screen:

1. Name of the screen
2. Success/Error message
3. Timestamp of the activity (local region format)

All logs can be erased with the **Clear all** button.

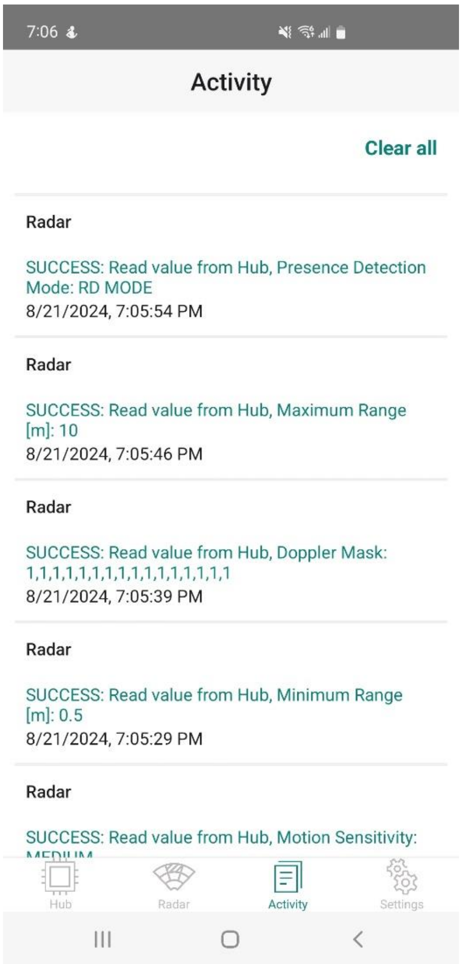


Figure 25 **Activity screen: Logs**

Note: Existing logs on Activity screen get cleared whenever EVK is disconnected.

5.4.5 Settings screen

The Settings screen provides the option to configure Characteristics of Radar service.

5.4.5.1 Radar settings

Settings can only be configured individually and once EVK connection is successful.

Note: *The current version of the app does not restore value of radar settings in the following cases:*
a. App is closed and reopened.
b. Back (<) button is tapped.
c. Radar settings screen is reopened.

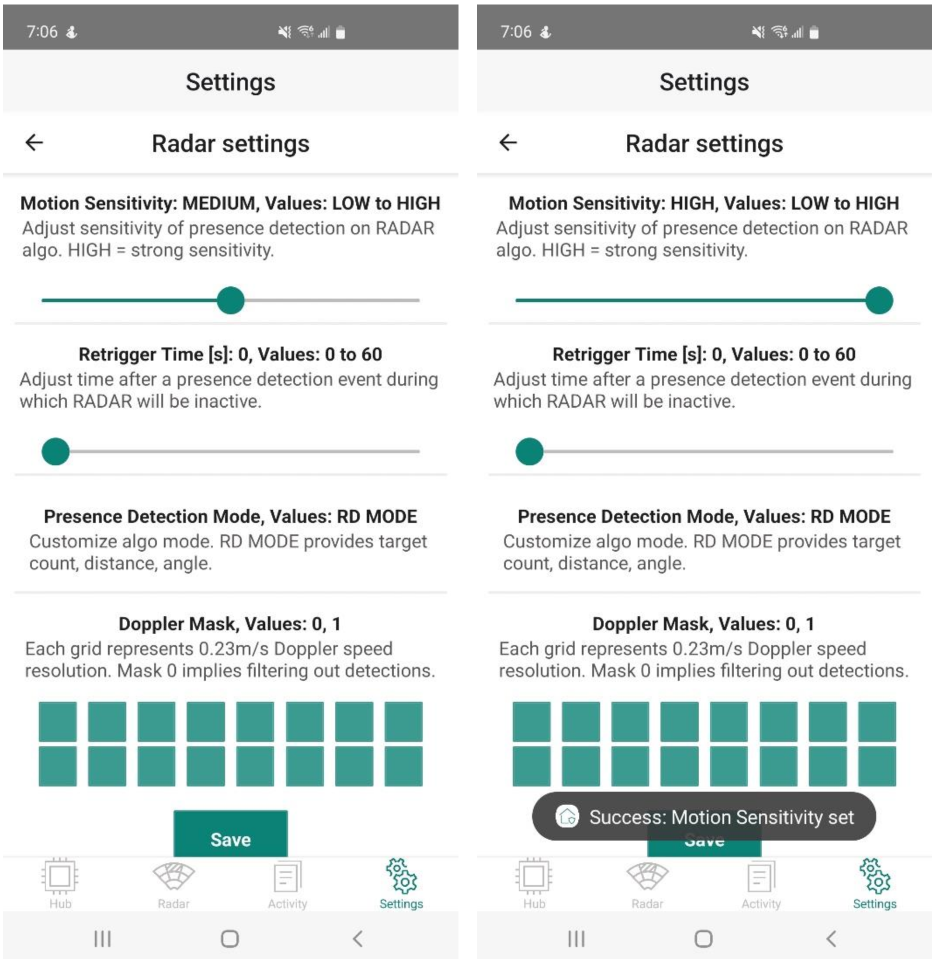


Figure 26 Radar settings: Successful configuration

Following are the settings available in the app:

- **Motion Sensitivity, Retrigger time:** Move the slider to select the required values.
- **Mode:** Select the required option and tap **Save**. Only Range Doppler (RD) is supported with current release.
- **Doppler Mask:** Select the respective Doppler grids and tap **Save**. Depending on the selection, the display changes:
  - **Selected grids appears in green:** Mask Off, which means that the selected Doppler grids are enabled and clutter rejection is disabled.

- **Selected grids appears in grey:** Mask On, which means that the selected Doppler grids are disabled; clutter rejection is applied for those grids.
- Every grid is based on ~ 0.23m/s Doppler speed resolution calculated from the number of chirps and pulse repetition time.
- Each grid filters out in terms of Doppler speed (m/s) as follows: 1.8838 (starting grid, top left), 1.6484, 1.4129, 1.774, 0.9419, 0.7064, 0.4710, 0.2355, 0, -0.2355, -0.4710, -0.7064, -0.9419, -1.774, -1.4129, -1.6484 (ending grid, bottom right).

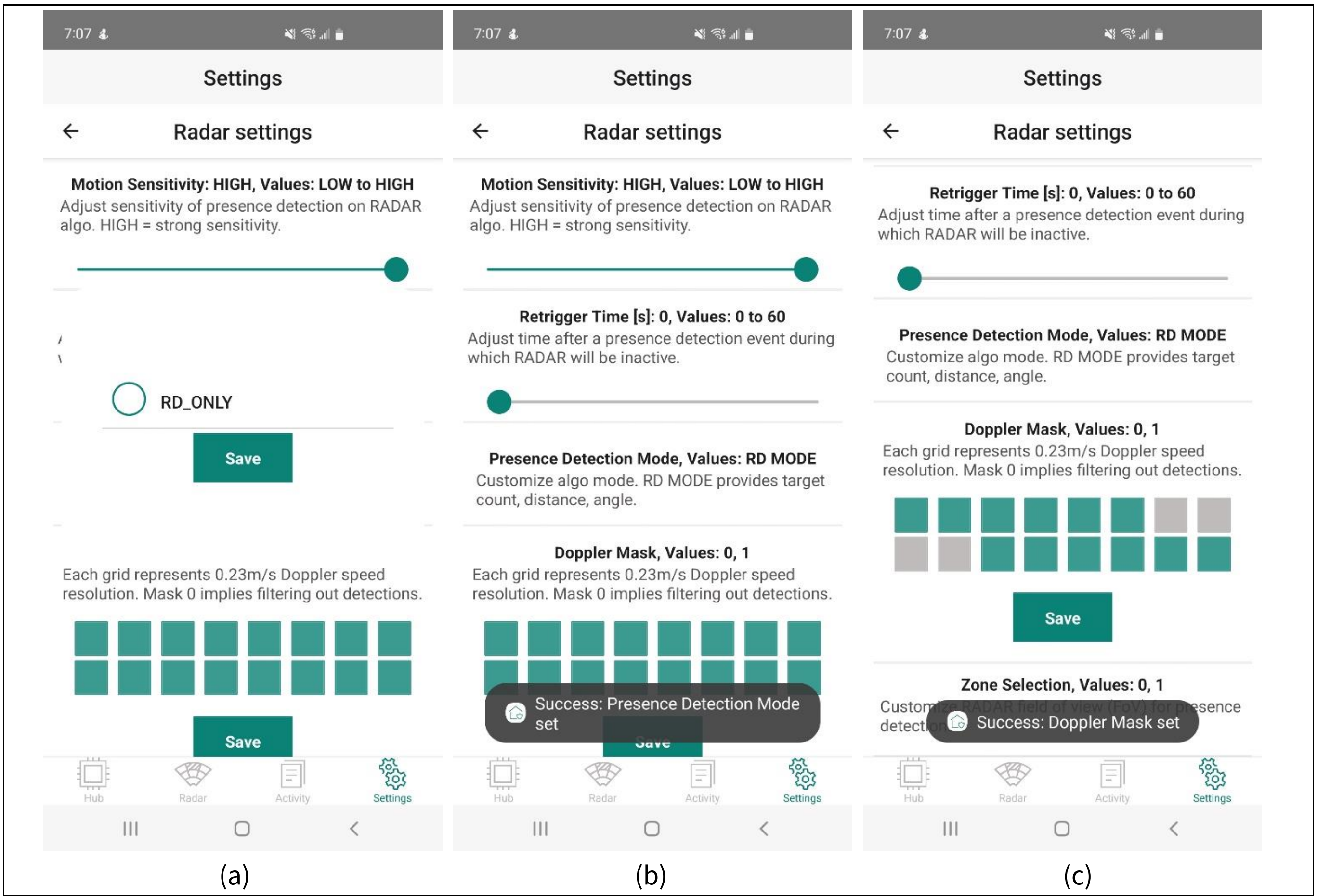


Figure 27 **(a) Mode selection; (b) Presence Detection mode; (c) Doppler Mask successful configuration**

5.4.5.2 Radar zone selection

The Radar zone selection setting configures the radar FoV through the selection of zones and configuration of minimum and maximum range. The slider can be used to set the values to adjust the radar FoV and the **Save** button to save the values.

Note: *Minimum Range must be less than the Maximum Range.*
*Example: If Maximum Range is set to 4 m, it will get configured in the EVK FW as 4 m but if the user goes back to Radar settings screen by pressing the **Back** button and comes back again or if app is forcefully closed and reopened, Maximum Range on app will reset to the default value. The value in FW is still 4 m if there is no power reset. Same applies for Minimum Range and zone selection.*

- Selected zones are indicated in green and deselected in grey as shown in Figure 29.
- By default, all zones are selected (green) when range is configured. Press **Reset zones** to clear selection.
- An empty selection cannot be saved. At least one zone must be selected or else, an Error message will be displayed as shown in Figure 30.

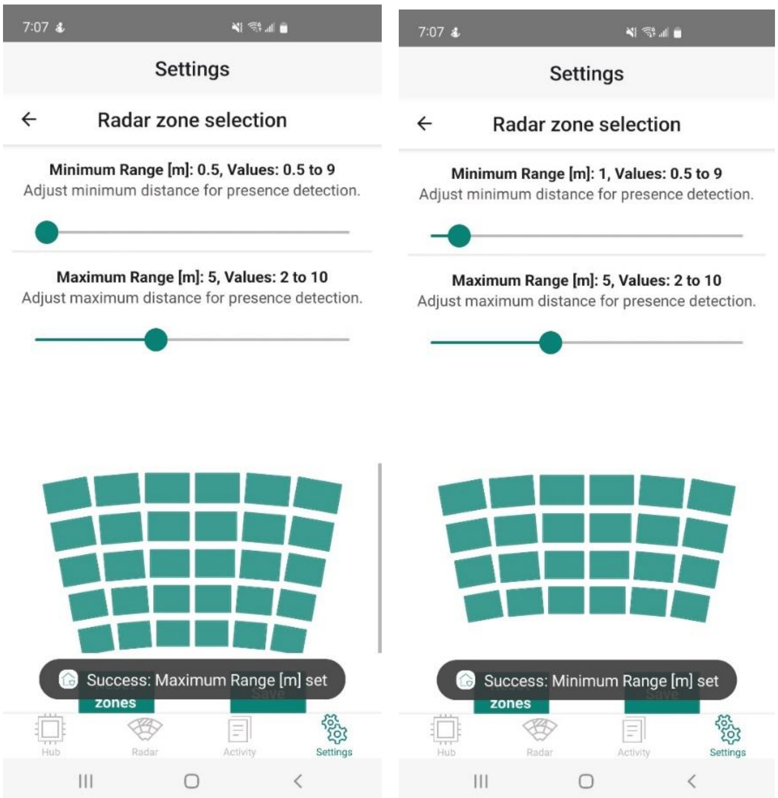


Figure 28 **Minimum and Maximum range successful configuration**

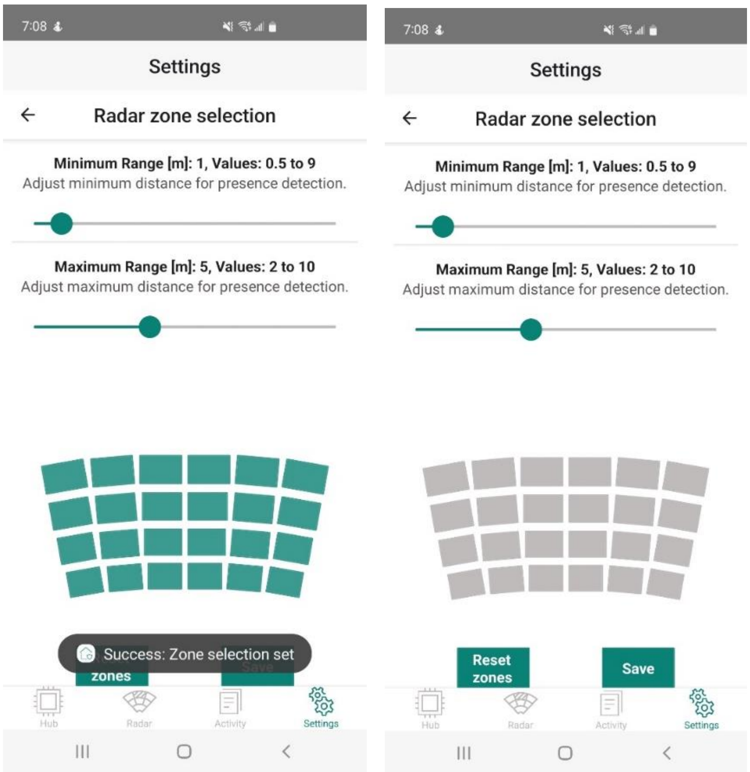


Figure 29 **Zone set and reset**

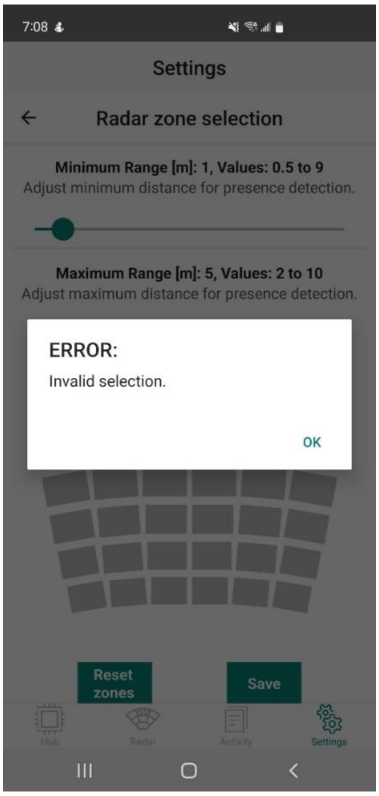


Figure 30 **Empty selection error**

5.4.5.3 Wi-Fi settings

The Wi-Fi settings screen is used to perform Wi-Fi onboarding. It includes options to send the SSID and password of a Wi-Fi Access Point (AP) to the EVK along with a command to connect to/disconnect from the AP.

Perform the following steps for Wi-Fi onboarding.

1. Go to the **Wi-Fi settings** screen once EVK connection is successful. If hotspot is used, make sure it is turned on.
2. Enter the SSID and password and press **Connect**. Transfer will be indicated by Toast messages. Empty SSID will result in an error.

The firmware connects to AP and then to MQTT Broker: AWS IoT Core to perform MQTT publish operations and updates the status of the connection.

3. Select an option from drop-down to read it. It may take several seconds for a successful connection to both AP and MQTT Broker. The MQTT Client Identifier is used to identify different devices using their MQTT Client ID, which is generated by the firmware.

FW publishes the MQTT Client ID number, message ID number, a timestamp, the detection, and range and angle information of two targets to the MQTT topic.

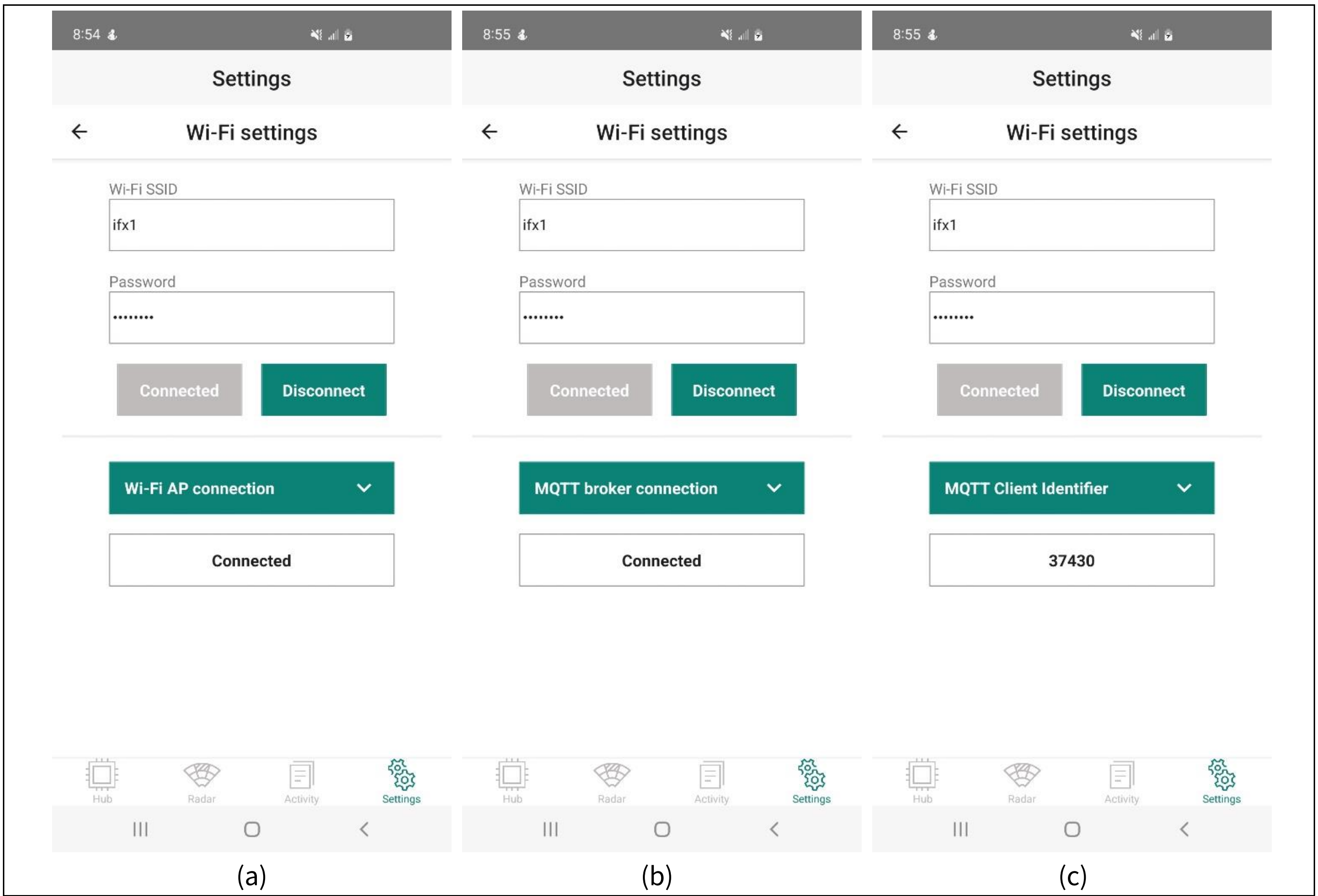


Figure 31 Wi-Fi settings: (a) Confirmation of successful connection to AP, (b) MQTT broker, and (c) MQTT Client Identifier display

Connect is disabled and **Disconnect** is enabled as soon as app successfully transfers SSID, password and connect command to EVK. It may still take several seconds before connection is successful. So, it is recommended to read status for confirmation. Similarly, **Disconnect** is disabled and **Connect** is enabled as soon as app successfully transfers disconnect command to EVK.

Note: *Wi-Fi settings screen values are not restored if the user goes back to the Settings screen by pressing the **Back** button and comes back again or if the app is forcefully closed and reopened. In this case, EVK FW and its connection with the app needs to be restarted.*

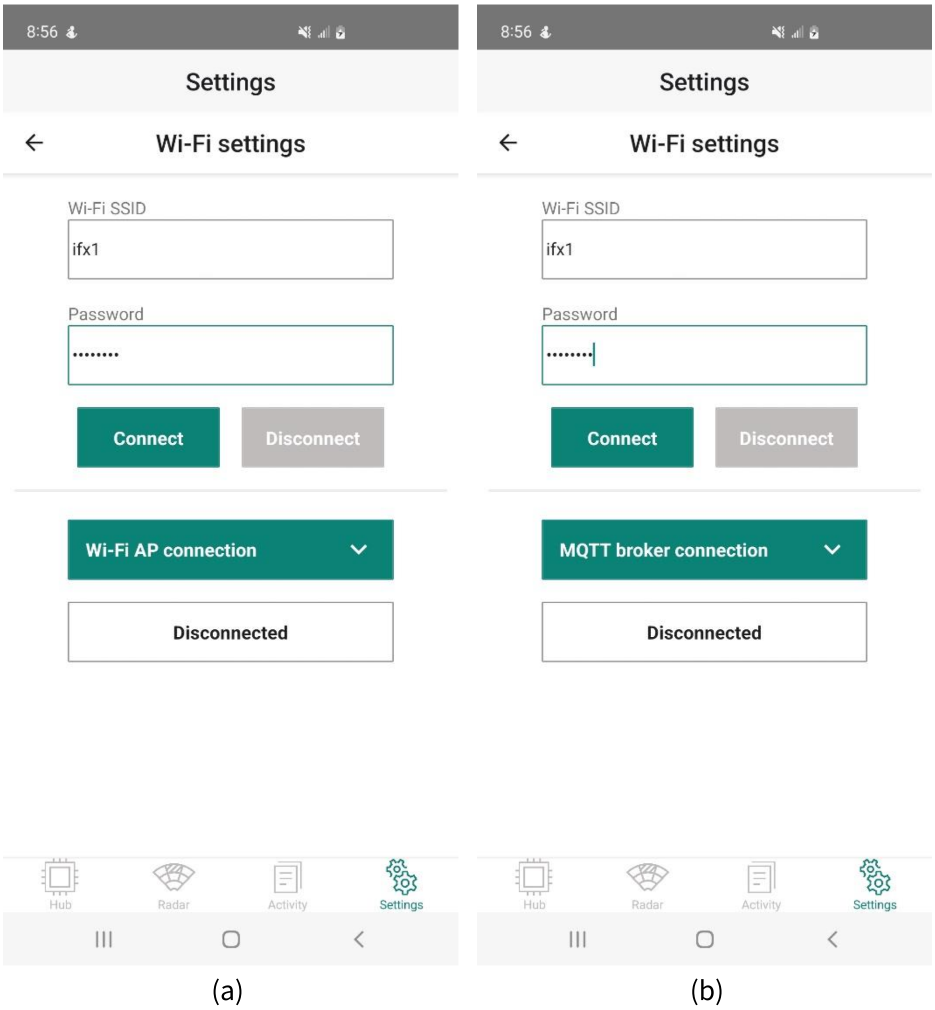


Figure 32 **Confirmation of successful disconnection from (a) AP and (b) MQTT Broker**

5.5 Operation

This section covers some important operational/troubleshooting steps for the CYW955913EVK-01 with the mobile app to view the target information within the FOV.

1. Open the mobile app on your mobile device. The Hub Screen will be displayed.
2. Restart your CYW955913EVK-01 by pressing the **Reset** button on the board.
   Refer to Section 5.4.2 for instructions on discovering and connecting to the CYW955913EVK-01.
3. Once connected, configure the system by modifying the settings outlined in previous sections in the Settings screen, if required.
   Refer to Section 5.4.5 for instructions on how to configure the system.

4. Once configured, navigate to the **Radar** screen.
5. Verify all the configured settings from the drop-down menu and press the **Read** button.
6. Once all settings are verified, select the **Range Angle Info** option from the drop-down menu and press the **Read** button.

The system should now communicate with the mobile app and send target information once targets are detected in the FoV. Refer to Section 5.4.3.1 for details on how target information is displayed.

6 MQTT Python subscriber example

The MQTT Python subscriber example is a simple Python example for the PC that can be used to read the Radar Presence Events from the MQTT messages published by the CYW955913EVK-01 to an MQTT Broker.

The AWS IoT Core service is used as the MQTT Broker in this example and certificates to subscribe to the broker are included in the package. The example consists of the certificates, the Python source file (*CYW955913EVK-01_MQTT_Subscriber.py*), and batch scripts to install and run the example. The installer batch file creates a virtual environment and installs the AWS IoT SDK. The run batch file activates that virtual environment and runs the Python source code.

The subscriber example subscribes to the "CYW955913EVK-01 RADAR EVENT" topic that the CYW955913EVK-01 is publishing MQTT messages to. It parses the JSON messages that are received in the format outlined in Section 3.5.1. The example will determine whether or not to print the output based on the Client ID in the MQTT message, this allows the user to only see the messages published by a particular device.

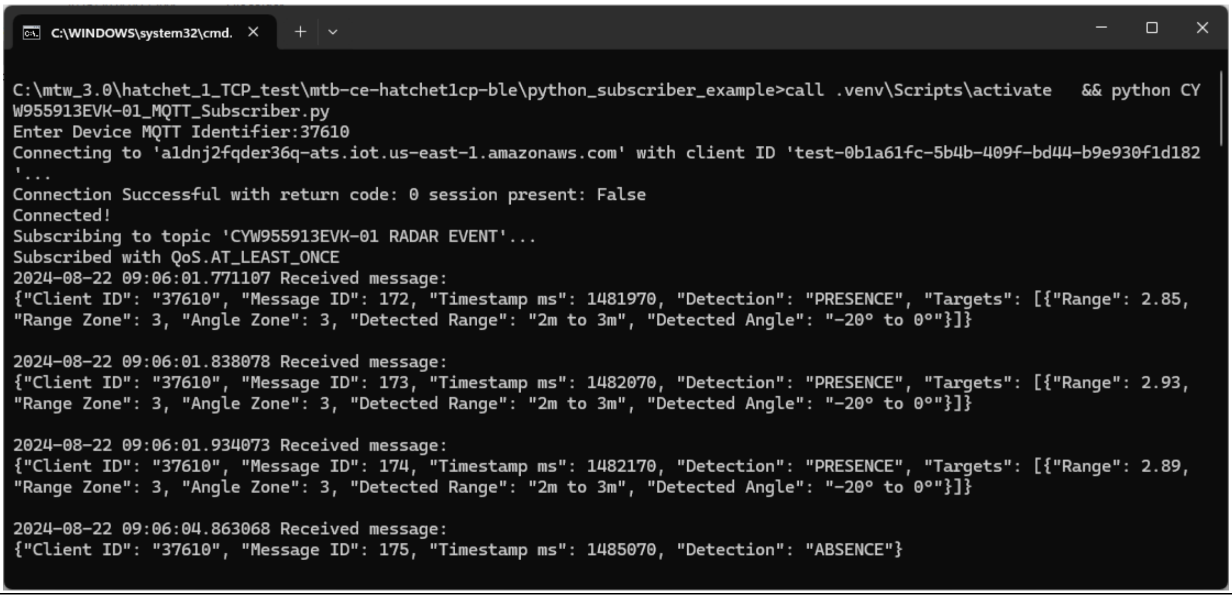
The output of the subscriber example is also in JSON format and includes the same data as the MQTT message with Detected Range and Detected Angle added. Detected Range and Detected Angle are human readable messages that show where detection has occurred based on the Range Zone and Range Angle values from the MQTT message.

6.1 Requirements

- Infineon Presence Sensing and Zoning Android mobile app
- Python 3.7 or later

6.2 Running the MQTT Python subscriber example

1. Connect to CYW955913EVK-01 with Android mobile app using Bluetooth® (Section 5.4.2).
2. Use the mobile app to connect to a Wi-Fi network (Section 5.4.5.3).
3. Enter "ping a1dnj2fqder36q-ats.iot.us-east-1.amazonaws.com" into a terminal to check the connection to the MQTT broker endpoint on a PC. Wi-Fi settings or network may need to be changed if there is no response.
4. Click **CYW955913EVK-01_MQTT_Installer.bat** to install. A *.venv* folder will be created.
5. Click **CYW955913EVK-01_MQTT_Run.bat** to run the example, read the MQTT Client Identifier from the Android mobile app (5.4.5.3) and enter the ID number into the subscriber example when prompted.



```
C:\mtw_3.0\hatchet_1_TCP_test\mtb-ce-hatchet1cp-ble\python_subscriber_example>call .venv\Scripts\activate   && python CY
W955913EVK-01_MQTT_Subscriber.py
Enter Device MQTT Identifier:37610
Connecting to 'a1dnj2fqder36q-ats.iot.us-east-1.amazonaws.com' with client ID 'test-0b1a61fc-5b4b-409f-bd44-b9e930f1d182
'...
Connection Successful with return code: 0 session present: False
Connected!
Subscribing to topic 'CYW955913EVK-01 RADAR EVENT'...
Subscribed with QoS.AT_LEAST_ONCE
2024-08-22 09:06:01.771107 Received message:
{"Client ID": "37610", "Message ID": 172, "Timestamp ms": 1481970, "Detection": "PRESENCE", "Targets": [{"Range": 2.85,
"Range Zone": 3, "Angle Zone": 3, "Detected Range": "2m to 3m", "Detected Angle": "-20° to 0°"}]}

2024-08-22 09:06:01.838078 Received message:
{"Client ID": "37610", "Message ID": 173, "Timestamp ms": 1482070, "Detection": "PRESENCE", "Targets": [{"Range": 2.93,
"Range Zone": 3, "Angle Zone": 3, "Detected Range": "2m to 3m", "Detected Angle": "-20° to 0°"}]}

2024-08-22 09:06:01.934073 Received message:
{"Client ID": "37610", "Message ID": 174, "Timestamp ms": 1482170, "Detection": "PRESENCE", "Targets": [{"Range": 2.89,
"Range Zone": 3, "Angle Zone": 3, "Detected Range": "2m to 3m", "Detected Angle": "-20° to 0°"}]}

2024-08-22 09:06:04.863068 Received message:
{"Client ID": "37610", "Message ID": 175, "Timestamp ms": 1485070, "Detection": "ABSENCE"}
```

Figure 33 MQTT Python subscriber example with output

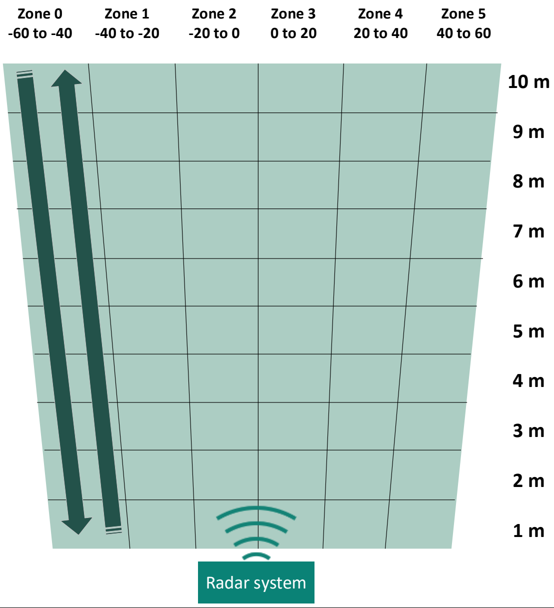
7 Testing and observations

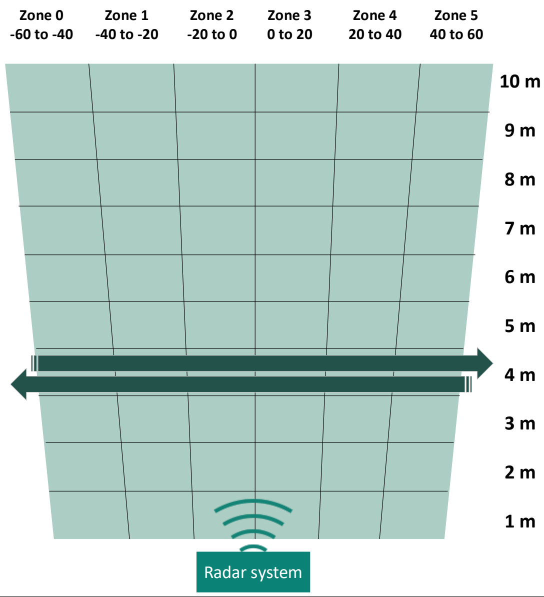
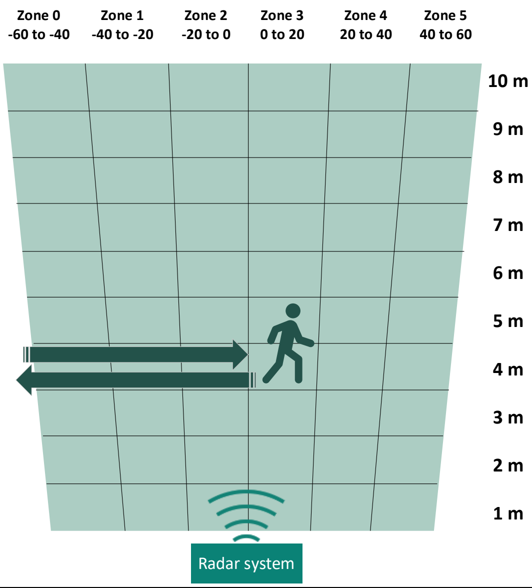
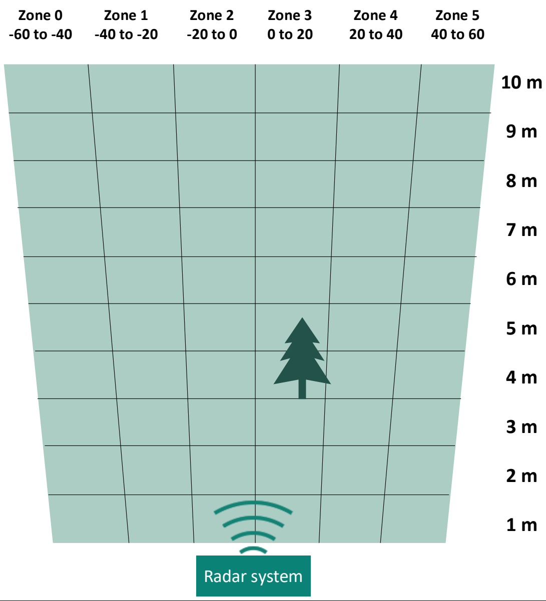
Outdoor system tests were conducted with the following parameters:

- Algorithm parameters
  - **Minimum range:** 0.5 m
  - **Maximum range:** 10 m
  - **Motion sensitivity:** MEDIUM
- System parameters
  - **Radar mounting:** 1.2 m from the floor
  - **Radar direction:** Front-facing
  - **Tilting:** None

Defined test cases

Table 5 Test cases

Type	Figure	Description
ZONE_RADIAL_WALK		• Person enters FoV from **Zone N** at 10 m and walks up to 1 m. • Person turns around and walks back toward 10 m and exits the FoV. • Person does this for all six zones.

Type	Figure	Description
CROSS_WALK		• Person enters the FoV from Zone 0 xm to Zone 5 xm (x representing 1 m to 10 m). • Person turns around and goes back the same way and exits the FoV. • Person does this for all distances (xm).
MAJOR_MOTION		• Person enters the FoV from **Zone M @ xm** (M representing Zone 0 to 5 and x representing 1 m to 10 m) and stands in a zone of preference. Major motion such as arm swings are performed for ~15 s. • Person then walks back the same way and exits the FoV.
CLUTTER_REJECTION		"Clutter" or unwanted targets in the FoV (such as indoor plants, cleaning robots, or curtains) are kept at a zone of preference.

7.1 Outdoor testing

7.1.1 Test setup

Figure 34 **Outdoor FoV ±60° (total FoV 120°) up to 10 m**

Figure 35 **Outdoor test with plant used for robustness (clutter rejection) testing**

7.1.2 Test observations

7.1.2.1 ZONE_RADIAL_WALK test

The following test results included the ZONE_RADIAL_WALK, where the test subject walked in every zone from 10 m to 1 m and then back to 10 m. Following were the observations:

- All tests were able to detect human presence; however, some fluctuations were observed with the pace at which presence was detected.
- Some zones detected presence as far as 10 m, whereas some zones detected only a range around 7-8 m (Table 6). See Figure 36 for plot of maximum range detected at each zone.

Table 6 Outdoor test results

Test N°	Objective	Type	Iterations	Expected result	Actual result	Inference	Comment
1	Single target PRESENCE: Zone 0	ZONE_RADIAL_WALK: Zone 0	2	Presence, then absence	Presence, then absence	PASS	Maximum range detected was 8 m (Sensitivity decreases as distance increases)
2	Single target PRESENCE: Zone 1	ZONE_RADIAL_WALK: Zone 1	2	Presence, then absence	Presence, then absence	PASS	Maximum range detected was 8 m
3	Single target PRESENCE: Zone 2	ZONE_RADIAL_WALK: Zone 2	2	Presence, then absence	Presence, then absence	PASS	Maximum range detected was 8 m
4	Single target PRESENCE: Zone 3	ZONE_RADIAL_WALK: Zone 3	2	Presence, then absence	Presence, then absence	PASS	Maximum range detected was 10 m
5	Single target PRESENCE: Zone 4	ZONE_RADIAL_WALK: Zone 4	2	Presence, then absence	Presence, then absence	PASS	Maximum range detected was 10 m.
6	Single target PRESENCE: Zone 5	ZONE_RADIAL_WALK: Zone 5	2	Presence, then absence	Presence, then absence	PASS	Maximum range detected was 7 m.

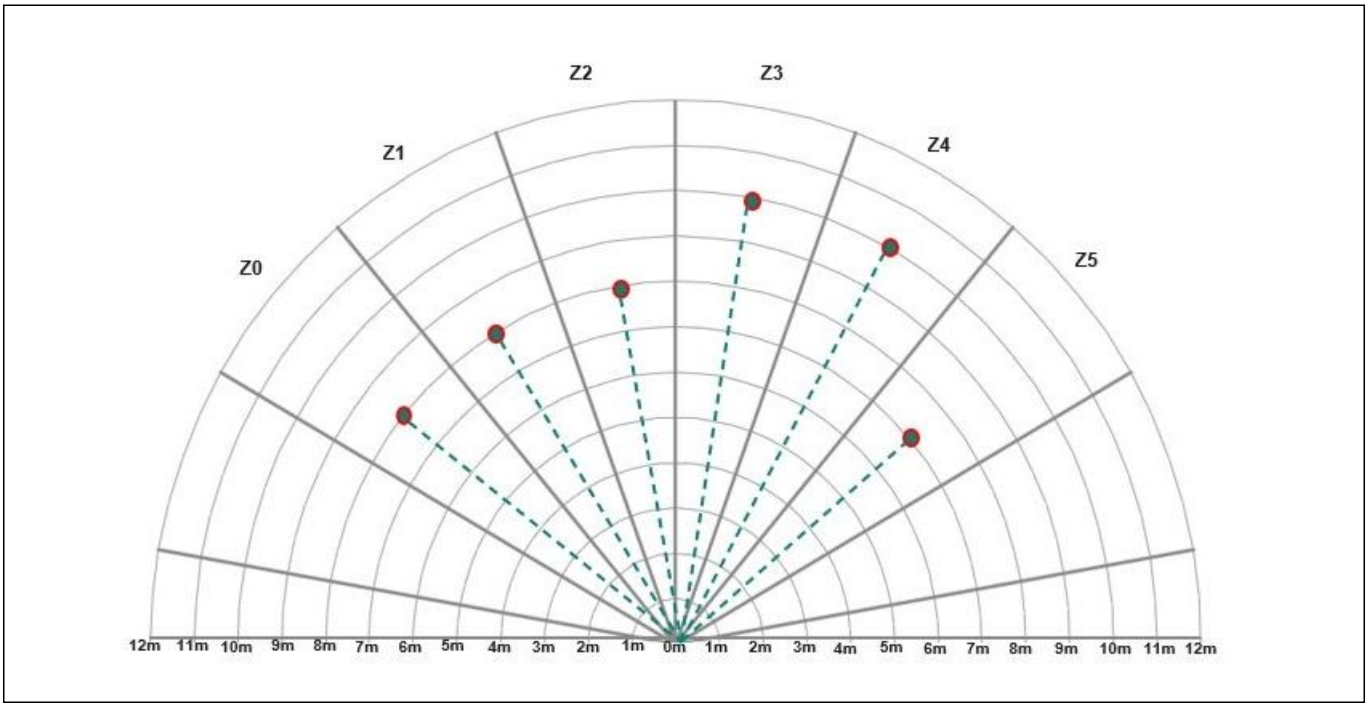


Figure 36 **Maximum range detected at each zone**

7.1.2.2 Clutter rejection test

The following outdoor test was done only to assess "clutter rejection".

Test preparation:

A house plant was placed in zones described in Table 7 at high, medium, and low sensitivity levels. A string was attached to the plant to force movement to simulate non-human targets moving in the background, such as tree branches moving due to wind. After 30 s of moving the plant, a human target entered the FoV at a different zone to confirm that the plant was rejected at its respective zone, but the human target was still detected.

The Doppler mask setting must be updated to reject the clutter at Doppler values.

- Doppler mask setting: 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1
- The 0's represent filtering out targets with estimated Doppler values from 1.15 m/s to 2.76 m/s

Test observation:

The test passed when plant was filtered out. A failure occurred when plant was not filtered out and got detected by the algorithm.

Overall, the tests showed that in all cases, the house plant was not detected and was considered "clutter" and thus filtered out from the algorithm's target output, and the human target was still detected in high and medium sensitivity levels. With low sensitivity, both plant and human target were not detected; however, our pass criteria was to filter out the plant. See Table 7.

Table 7 Outdoor robustness (clutter rejection) test results

Test No.	Sensitivity Level	Type	Iterations	Expected result	Actual result	Inference	Comment
001	High	CLUTTER_REJECTION: Zone 0, 1 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
002	High	CLUTTER_REJECTION: Zone 3, 1 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
003	High	CLUTTER_REJECTION: Zone 5, 1 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
004	High	CLUTTER_REJECTION: Zone 0, 3 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
005	High	CLUTTER_REJECTION: Zone 3, 3 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
006	High	CLUTTER_REJECTION: Zone 5, 3 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
007	High	CLUTTER_REJECTION: Zone 0, 8 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
008	High	CLUTTER_REJECTION: Zone 3, 8 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
009	High	CLUTTER_REJECTION: Zone 5, 8 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
010	Medium	CLUTTER_REJECTION: Zone 0, 1 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
011	Medium	CLUTTER_REJECTION: Zone 3, 1 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
012	Medium	CLUTTER_REJECTION: Zone 8, 1 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
013	Medium	CLUTTER_REJECTION: Zone 0, 3 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.

Test No.	Sensitivity Level	Type	Iterations	Expected result	Actual result	Inference	Comment
014	Medium	CLUTTER_REJECTION: Zone 3, 3 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
015	Medium	CLUTTER_REJECTION: Zone 8, 3 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
016	Medium	CLUTTER_REJECTION: Zone 0, 8 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
017	Medium	CLUTTER_REJECTION: Zone 3, 8 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
018	Medium	CLUTTER_REJECTION: Zone 5, 8 m	1	Absence	Absence	PASS	Human target detected once they entered FoV.
019	Low	CLUTTER_REJECTION: Zone 0, 1 m	1	Absence	Absence	PASS	Plant was rejected successfully but human target not detected due to low sensitivity threshold being too low.
020	Low	CLUTTER_REJECTION: Zone 3, 1 m	1	Absence	Absence	PASS	Plant was rejected successfully but human target not detected due to low sensitivity threshold being too low.
021	Low	CLUTTER_REJECTION: Zone 5, 1 m	1	Absence	Absence	PASS	Plant was rejected successfully but human target not detected due to low sensitivity threshold being too low.
022	Low	CLUTTER_REJECTION: Zone 0, 3 m	1	Absence	Absence	PASS	Plant was rejected successfully but human target not detected due to low sensitivity threshold being too low.

Test No.	Sensitivity Level	Type	Iterations	Expected result	Actual result	Inference	Comment
023	Low	CLUTTER_REJECTION: Zone 3, 3 m	1	Absence	Absence	PASS	Plant was rejected successfully but human target not detected due to low sensitivity threshold being too low.
024	Low	CLUTTER_REJECTION: Zone 5, 3 m	1	Absence	Absence	PASS	Plant was rejected successfully but human target not detected due to low sensitivity threshold being too low.
025	Low	CLUTTER_REJECTION: Zone 0, 8 m	1	Absence	Absence	PASS	Plant was rejected successfully but human target not detected due to low sensitivity threshold being too low.
026	Low	CLUTTER_REJECTION: Zone 3, 8 m	1	Absence	Absence	PASS	Plant was rejected successfully but human target not detected due to low sensitivity threshold being too low.
027	Low	CLUTTER_REJECTION: Zone 5, 8 m	1	Absence	Absence	PASS	Plant was rejected successfully but human target not detected due to low sensitivity threshold being too low.

8 Future upgrades and capabilities

Future upgrades will be implemented to further reduce the overall power consumption of the system. During initial testing, increased current consumption was observed in the idle state, where the system is not processing data and was waiting to new radar data to be ready. Usually, in an RTOS environment, these idle states are utilized best to put the CPU to sleep to reduce the overall power consumption. Enhanced power optimization techniques are in progress and will be implemented in future releases.

As this module is a Wi-Fi/Bluetooth® combo MCU, future releases will focus on integration of a camera module and use of Wi-Fi core for streaming camera's video from this MCU directly to a cloud service. Infineon is currently investigating several possibilities of what can be achieved by this powerful MCU.

Future upgrade of the mobile app will focus on discovery and listing of multiple EVKs (if present) during scanning, thereby opening possibility for connecting to each one at a given time.

References

[1] Infineon Technologies AG: *XENSIV™ BGT60TR13C 60 GHz radar sensor*; Available online

[2] Infineon Technologies AG: *AIROC™ CYW55913 Wi-Fi & Bluetooth® LE Connected MCU*; Available online

[3] Infineon Technologies AG: *XENSIV™ BGT60TR13C 60 GHz radar shield;* Available online

[4] Infineon Technologies AG: *XENSIV™ radar adapter board;* Available upon request

[5] Infineon Technologies AG: *CYW955913EVK-01 Evaluation Kit user guide*; Available online

[6] Infineon Technologies AG: *CYW955913EVK-01 Evaluation Kit;* Available online

Revision history

Document revision	Date	Description of changes
1.00	2024-09-30	Initial release

Trademarks

All referenced product or service names and trademarks are the property of their respective owners.

Edition 2024-09-30

Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Email: erratum@infineon.com

Document reference

AN003623