CYW943907AEVAL1F evaluation kit guide

About this document

Scope and purpose

This document serves as a guide for using the CYW943907AEVAL1F evaluation kit. The document explains software installation, kit operation, hardware details of the board, and provides a list of code examples.

Intended audience

This guide is intended for all technical specialists who want to use the CYW943907AEVAL1F kit using ModusToolbox™ software (2.4 or later) and are familiar with Wi-Fi connectivity. This board is intended to be used in laboratory conditions.

Reference documents

This user guide should be read in conjunction with the following documents:

- **ModusToolbox™ software user guide**
- **ModusToolbox™ software installation guide**
- **AIROC™ CYW43907 data sheet**

Table of contents

Safety and regulatory compliance information

This kit is intended for development purposes only. The board is an open-system design, which does not include a shielded enclosure. Due to this reason, the board may interfere with other electrical or electronic devices in close proximity. In a domestic environment, this product may cause radio interference. In such cases, take adequate preventive measures. Also, do not use this board near any medical equipment or RF devices.

Safety evaluation for this kit is done in factory default settings using default accessories shipped with the kit. All evaluations for safety are carried out using a 5-V (USB 2.0, @ 500 mA) supply. Attaching additional wiring to this product or modifying the product operation from the factory default may affect its performance and cause interference with other apparatus in the immediate vicinity. If such interference is detected, suitable mitigating measures should be taken.

This kit is not a finished product and when assembled may not be resold or otherwise marketed unless all required authorizations are first obtained. Contact **support@cypress.com** for details.

	These kits contain electrostatic discharge (ESD) sensitive devices. Electrostatic charges readily accumulate on the human body and any equipment, which can cause a discharge without detection. Permanent damage may occur on devices subjected to high-energy discharges. Proper ESD precautions are recommended to avoid performance degradation or loss of functionality. Store unused kits in the protective shipping package.
	End-of-Life/Product Recycling The end-of-life cycle for this kit is five years from the date of manufacture mentioned on the back of the box. Contact your nearest recycler to discard the kit.

General safety instructions

ESD protection

ESD can damage boards and associated components. Infineon recommends that you perform procedures only at an ESD workstation. If an ESD workstation is unavailable, use appropriate ESD protection by wearing an anti-static wrist strap attached to the chassis ground (any unpainted metal surface) on your board when handling parts.

Handling boards

CYW943907AEVAL1F boards are sensitive to ESD. Hold the board only by its edges. After removing the board from its box, place it on a grounded, static-free surface. Use a conductive foam pad if available. Do not slide the board over any surface. Any physical action on the board such as changing wires, jumper settings, or measuring voltages can cause stress on the printed circuit board assembly (PCBA). You must ensure that the PCBA has proper support on the bottom side to avoid stress on the PCBA when the EVK is in operation.

1 Introduction

Thank you for your interest in the CYW943907AEVAL1F evaluation kit (EVK). This EVK enables you to evaluate and develop single-chip Wi-Fi applications using AIROC™ CYW43907 devices.

The EVK uses ModusToolbox™ software (2.4 or later) to develop and debug your CYW43907 project.This EVK offers footprint-compatibility with Arduino shields. In addition, the kit features an RJ45 Ethernet connector, a Micro-SD-card slot, onboard programmer/debugger, and serial bridge chip. This EVK supports only 3.3 V as the operating voltage.

Older revisions of the same kit were named “BCM943907AEVAL1F_2” and “BCM943907AEVAL1F”. “CYW43907” and “BCM43907” refer to the same device.

ModusToolbox™ software (2.4 or later) is compatible with Windows, macOS, and Linux operating systems. Both ModusToolbox™ software and **WICED IDE** can be used for the CYW943907AEVAL1F evaluation kit (EVK) but the default IDE is ModusToolbox™ software.

For more information on WICED IDE, see **https://www.cypress.com/file/370526/download**.

Note: This document applies to ModusToolbox™ software (2.4 or later).

The CYW943907AEVAL1F EVK is available through the **Cypress Online Store** or through distributors.

1.1 Kit contents

The CYW943907AEVAL1F EVK includes the following:

- One CYW943907AEVAL1F evaluation board with assembled headers compatible with Arduino
- One USB 2.0 Type-A to Micro-B cable

Hardware not included with the kit

The CYW943907AEVAL1F EVK does not come with all the hardware needed to perform the demonstrations documented in this guide.

The following hardware is not included with this kit and must be supplied separately:

- RJ45 Ethernet cable
- SD card
- External power supply
- Dual external antenna

Figure 1 Kit contents

Inspect the contents of the kit; if you find any part missing, go to **www.cypress.com/support**.

1.2 Getting started

This guide will help you to get acquainted with this evaluation kit:

- The **Software installation** chapter describes the installation of the kit software. This includes extracting the required files for ModusToolbox™ software (2.4 or later).
- The **Kit operation** chapter describes the major sections of the kit such as the onboard programmer/debugger chip, reset control, headers, programming and debugging of the kit.
- The **Hardware** chapter describes the EVK hardware and its different blocks.
- The **Code examples** chapter provides a list of code examples that will help you understand how to get started with basic peripheral and WLAN examples.

1.3 Board details

The CYW943907AEVAL1F board consists of the blocks shown in **Figure 2**.

1. Reset switch (SW2)
2. RJ45 connector (J14)
3. Micro USB (programming and debugging) (J5)
4. (Optional) 5-12 V power input (J8)
5. 44-pin expansion header with I2C, SDIO, UART, SPI, PWM lines and I/Os (J6)
6. Header compatible with Arduino (J13)
7. User switch 1 (SW3)
8. User switch 2 (SW1)
9. Header compatible with Arduino (J9)
10. PCB antenna - Main (ANT1)
11. Connector for external antenna 1 (J1)
12. AIROC™ CYW43907 Type 1GC module from Murata (U14)
13. PCB antenna - Diversity (ANT0)
14. Connector for external antenna 0 (J2)
15. Onboard/external JTAG switch (SW4)
16. External JTAG header (J3)
17. Header compatible with Arduino (J10)
18. Header compatible with Arduino (J12)
19. Ethernet PHY chip (U12) – BCM5241

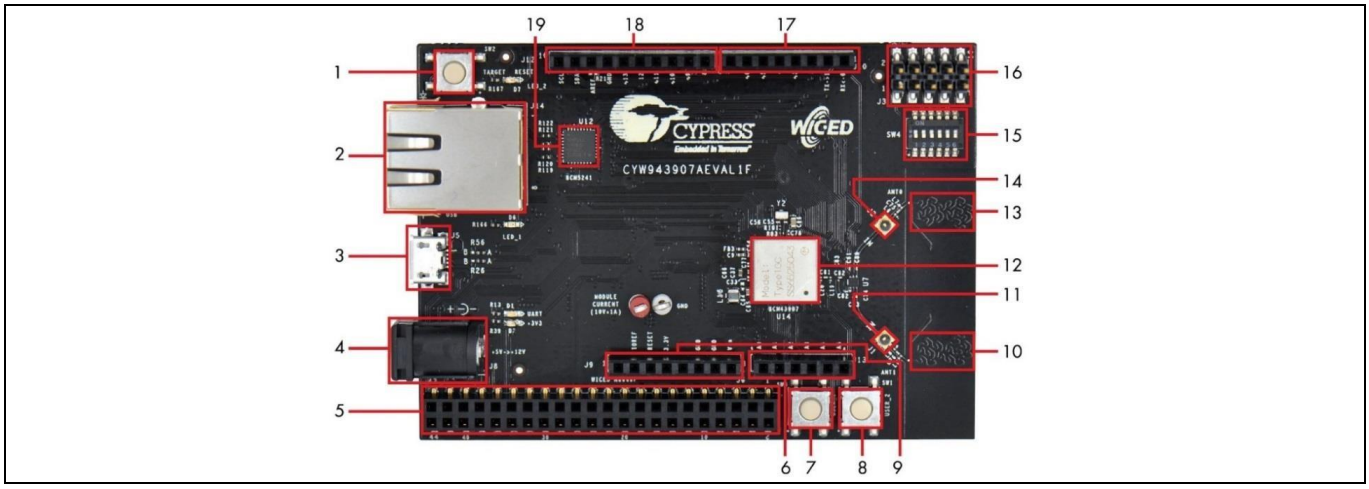

Figure 2 CYW943907AEVAL1F evaluation board – top view

1.4 Software development and system overview

See the **ModusToolbox™ software user guide**.

1.5 Additional learning resources

Visit **CYW943907AEVAL1F EVK** and **AIROC™ CYW43907** web pages for additional learning resources including datasheets and application notes.

1.6 IoT resources and technical support

Cypress provides a wealth of data at **www.cypress.com/internet-things-iot** to help you select the right IoT device for your design, and quickly and effectively integrate the device into your design. Cypress provides customer access to a wide range of information, including technical documentation, schematic diagrams, product bill of materials, PCB layout information, and software updates.

For assistance, go to **www.cypress.com/support**. Visit **community.infineon.com** to ask your questions in Infineon developer community.

1.7 Documentation conventions

Table 1 Document conventions

Convention	Usage
`Courier New`	Displays user-entered text and source code
Italics	Displays file names and reference documentation: Read about the *sourcefile.hex* file in the *PSoC™ Creator user guide*.
File > Open	Represents menu paths: File > Open > New Project
Bold	Displays commands, menu paths, and icon names in procedures: Click the **File** icon and then click **Open**.
Times New Roman	Displays an equation: $2 + 2 = 4$
Text in gray boxes	Describes Cautions or unique functionality of the product.

1.8 Abbreviations and definitions

Table 2 Abbreviations

Abbreviation	Definition
AWS	Amazon Web Services
CPU	central processing unit
DC	direct current
ESD	electrostatic discharge
EVK	evaluation kit
GPIO	general-purpose input/output
IC	integrated circuit
IDE	integrated development environment

Introduction

Abbreviation	Definition
IoT	Internet of Things
I2C	Inter-integrated Circuit
I2S	Inter-IC Sound
JTAG	Joint Test Action Group
LED	light-emitting diode
LPO	low-power oscillator
MQTT	message queue telemetry transport
PC	personal computer
PSoC™	programmable system-on-chip
SDIO	secure digital input output
SDK	software development kit
SPI	serial peripheral interface
UART	universal asynchronous receiver transmitter
USB	Universal Serial Bus
WLAN	wireless local area network

2 Software installation

2.1 Install software

Go to the ModusToolbox™ software website (**www.cypress.com/modustoolbox**) and download the appropriate software for your platform.

See **ModusToolbox™ software installation guide**.

3 Kit operation

3.1 Theory of operation

Figure 3 illustrates the block diagram of the CYW943907AEVAL1F evaluation board. This board contains a Type 1GC wireless module as a system-in-a-package (SiP) based on the AIROC™ CYW43907 device, which is an embedded network controller solution from Murata. This board also contains a USB-Serial interface, JTAG programmer, and a debugger.

This board features headers that are form-factor-compatible with Arduino, which enable Arduino shields to be plugged on top, extending its capabilities. This board also features two user switches, two user LEDs, an RJ45 connector for Ethernet, and a reset switch for the wireless module.

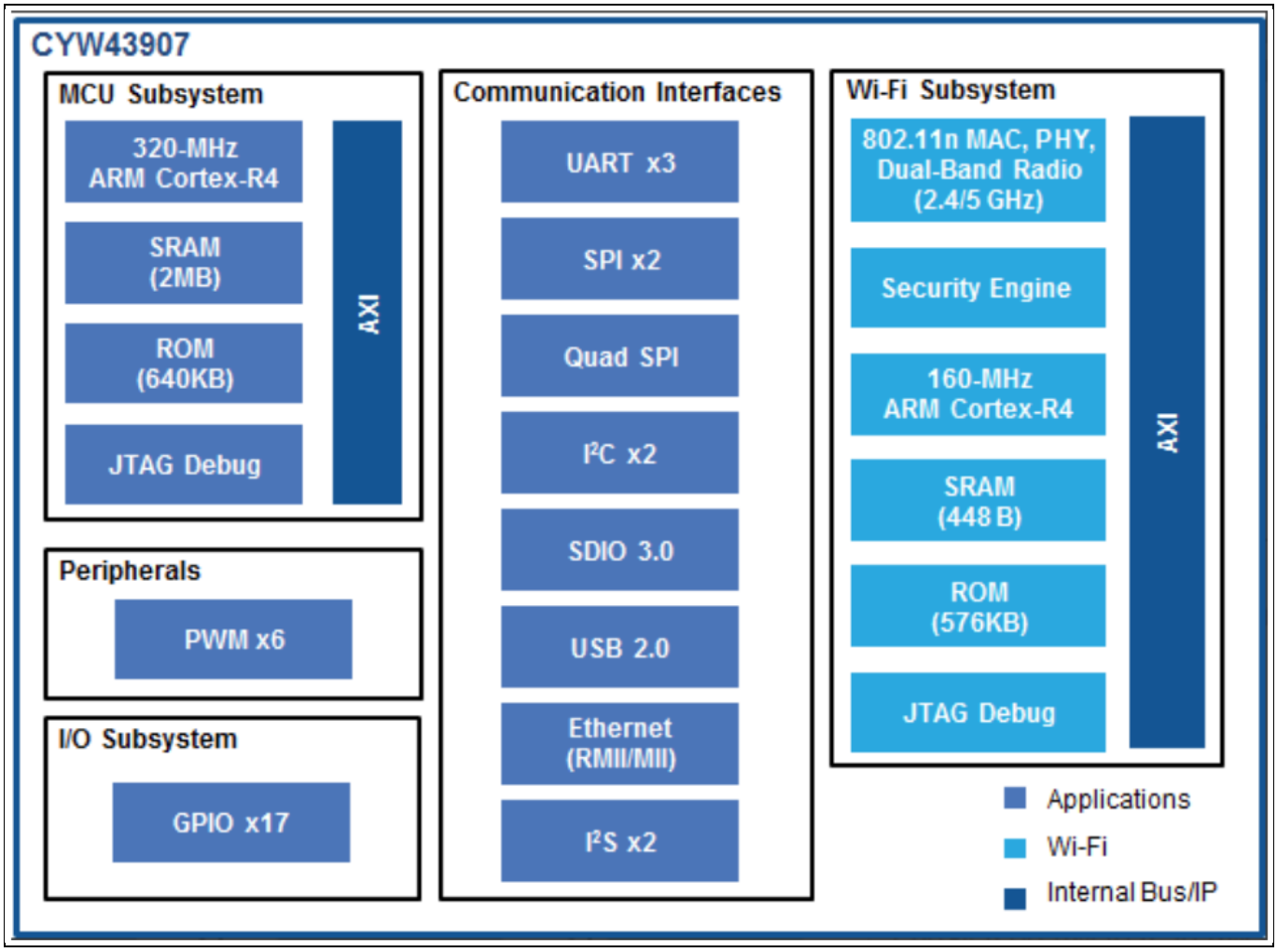

Figure 3 Block diagram

3.2 Onboard programmer/debugger and serial interface chip

An FT-2232-HQ chip is used for onboard programming, debugging, and USB-serial functionality. It connects to the PC over a USB interface and connects to the AIROC™ CYW43907-based SiP module over JTAG and UART pins. Alternatively, you can use the external JTAG connector (J3) along with switch **SW4** (in all closed positions) in order to use JTAG from connectors such as Olimex.

3.2.1 Kit connection

The EVK can be powered by an external power supply or though USB.

When using an external power supply, you should use a 5 V–12 V, 2-A power supply with a 2.1-mm DC jack (center pin positive). When powered from USB, there are two logical USB devices: a USB-JTAG device and a USB-UART device. Drivers for the EVK are automatically installed during the ModusToolbox™ SDK installation process.

3.2.2 Building, programming, and debugging

1. Start ModusToolbox™ software.

   When launching the Eclipse IDE, it provides an option to select the workspace location on your machine. This location is used by the IDE for creating and storing the files as part of application creation for a particular platform. The default workspace location is a folder called “mtw” in your home directory. You may add additional folders under the “mtw” folder or to choose any other location for each workspace.
2. Click the **New Application** link in the **Quick Panel** (or, use **File** > **New** > **ModusToolbox™ Application**). This launches the Project Creator tool.

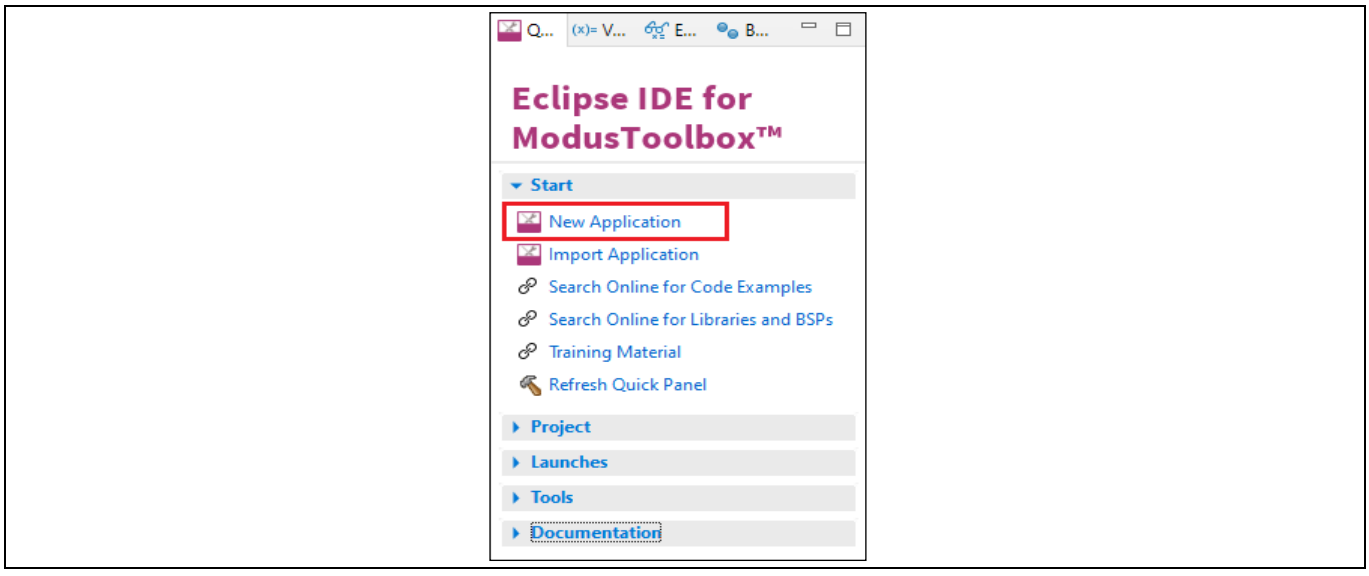

Figure 4 Create new application

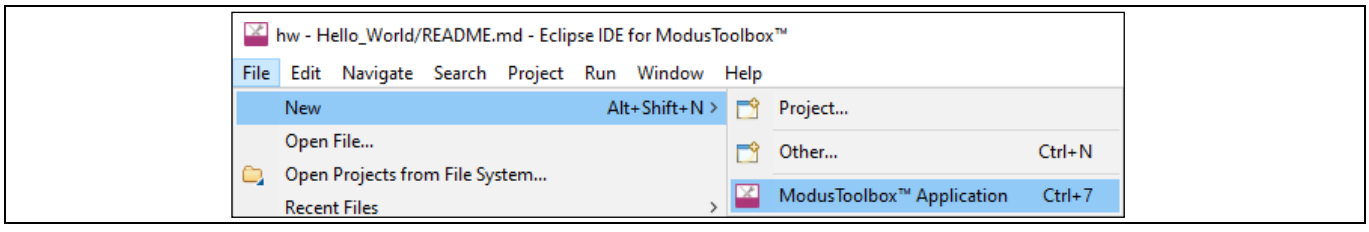

Figure 5 Create new application (alternative method)

Kit operation

3. Select CYW943907AEVAL1F kit in the Project Creator tool and click **Next**.

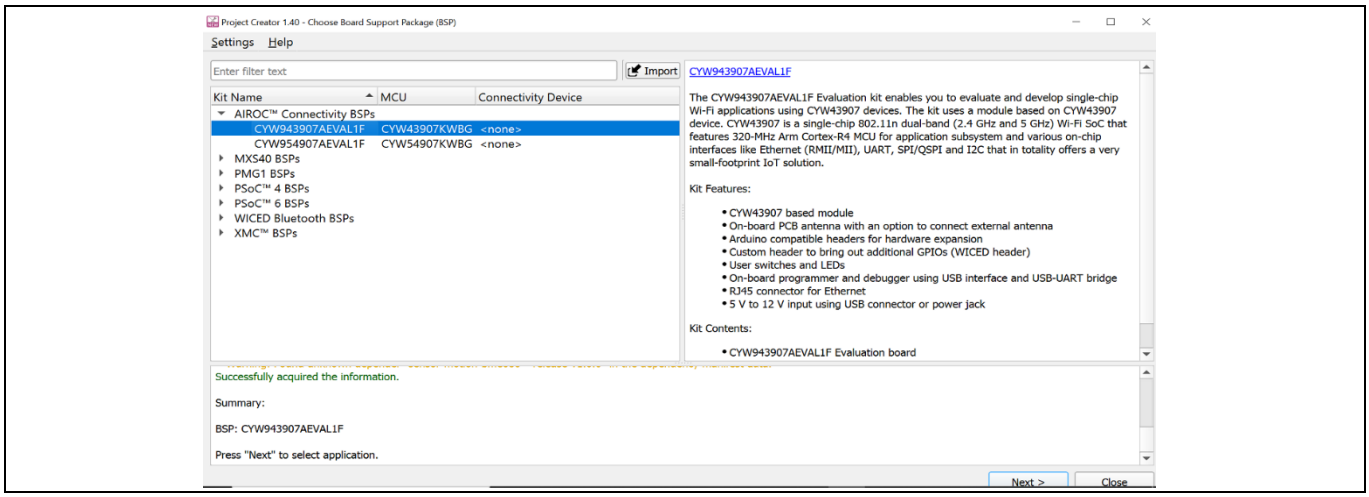

Figure 6 CYW943907AEVAL1F kit selection in Project Creator tool

This page lists various applications available for the CYW943907AEVAL1F kit. As you select an application, a description displays on the right. You can select multiple applications for the selected BSP by enabling the check box next to the applicable applications. Click **Create** to begin the project creation process.

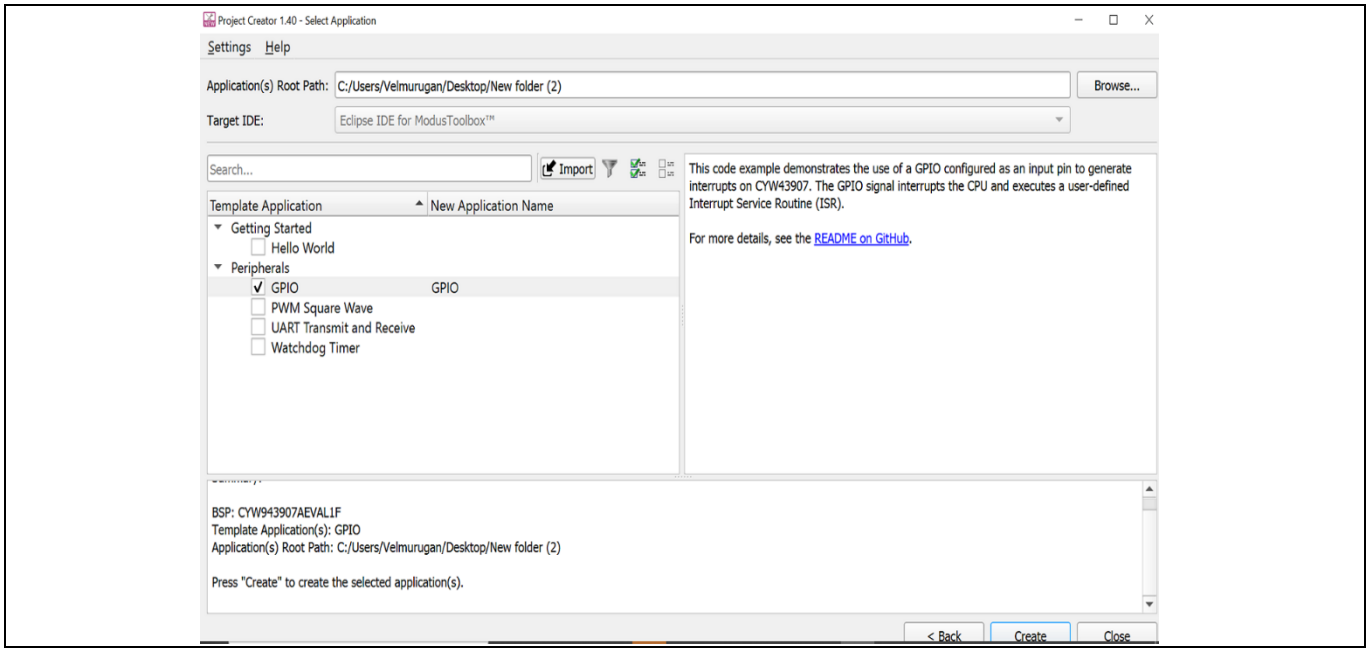

Figure 7 Beginning of the project creation process

For this example:

- Select the checkbox next to the“GPIO” application.
- (Optional) Type a name for the application under **New Application Name**. Do not use spaces in the application name. In this case, we use the default “GPIO” as the name.

Note: The application creation process performs a `git clone` operation, and downloads the selected application from the GitHub repository. Depending on the selected application, this process can take several minutes.

Kit operation

When complete, the Project Creator tool closes automatically. In the IDE, a message displays about importing the project:

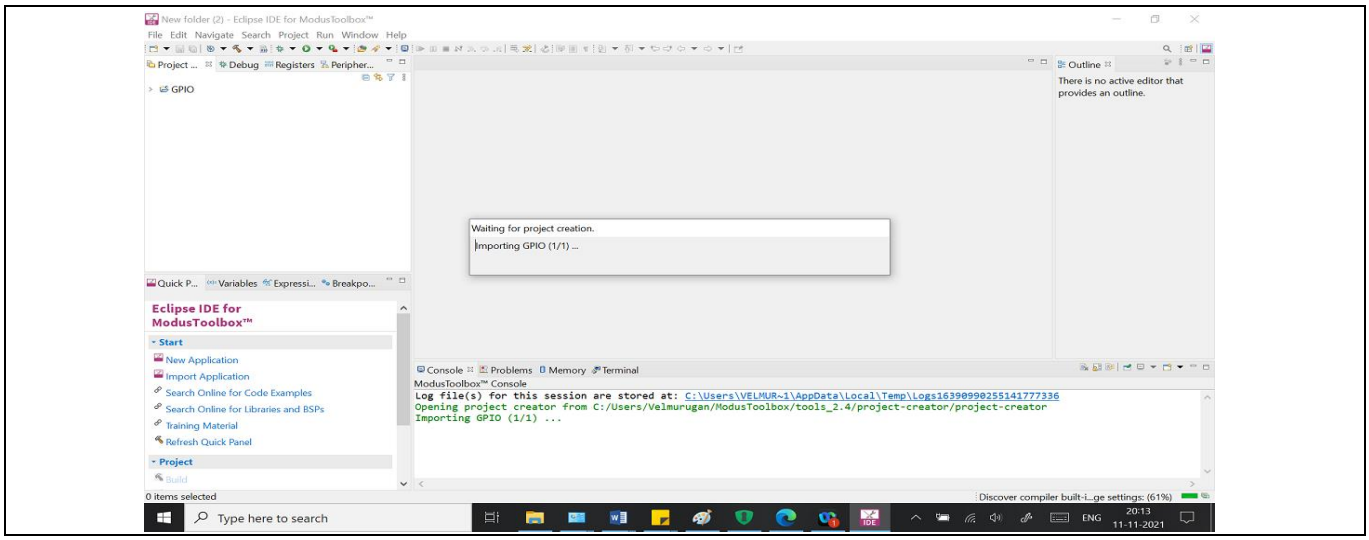

Figure 8 Project creation process complete

After several moments, the application opens with the *GPIO* in the project explorer, and the *README.md* file opens in the file viewer.

*Note: If the README.md file does not render correctly in the IDE, view it from code examples GitHub repo (**https://github.com/Infineon/mtb-example-cyw43907-gpio/blob/master/README.md**).*

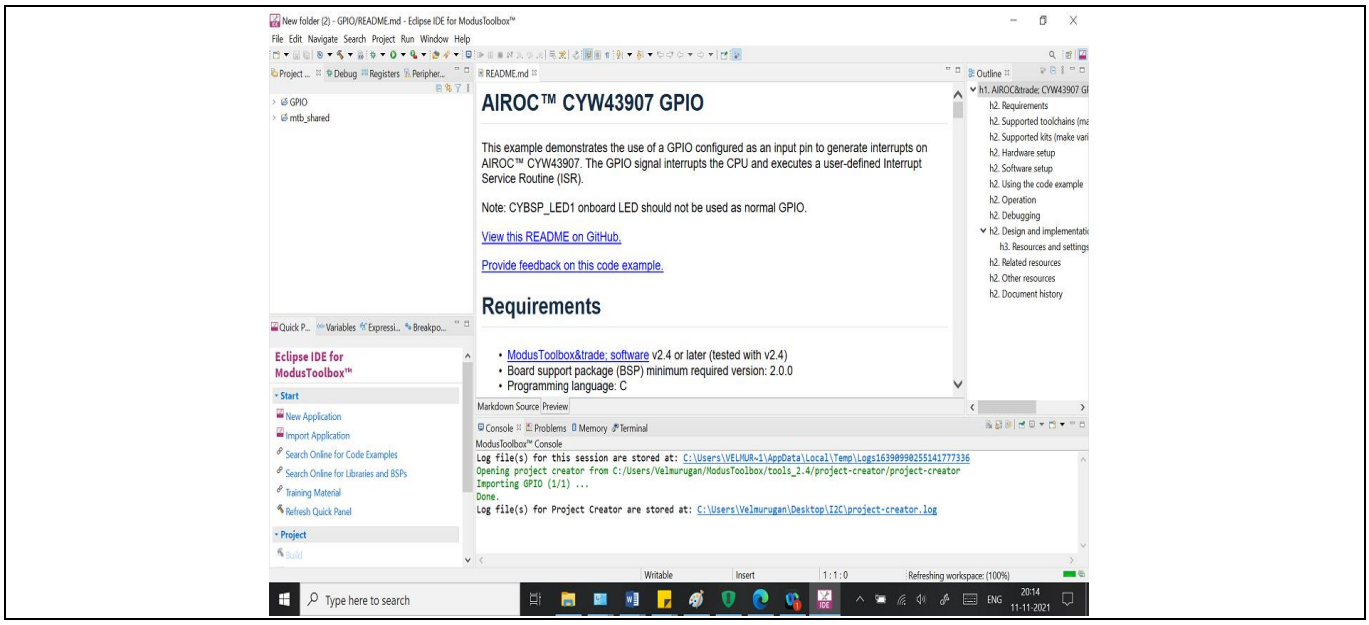

Figure 9 Opening an application in Project Explorer

4. After loading the application, build it to generate the necessary files. Select a project, and then in the **Quick Panel**, click the **Build <project> Application** link. The following images show the Quick Panel for a typical AIROC™ CYW43907 Wi-Fi connectivity processor application.

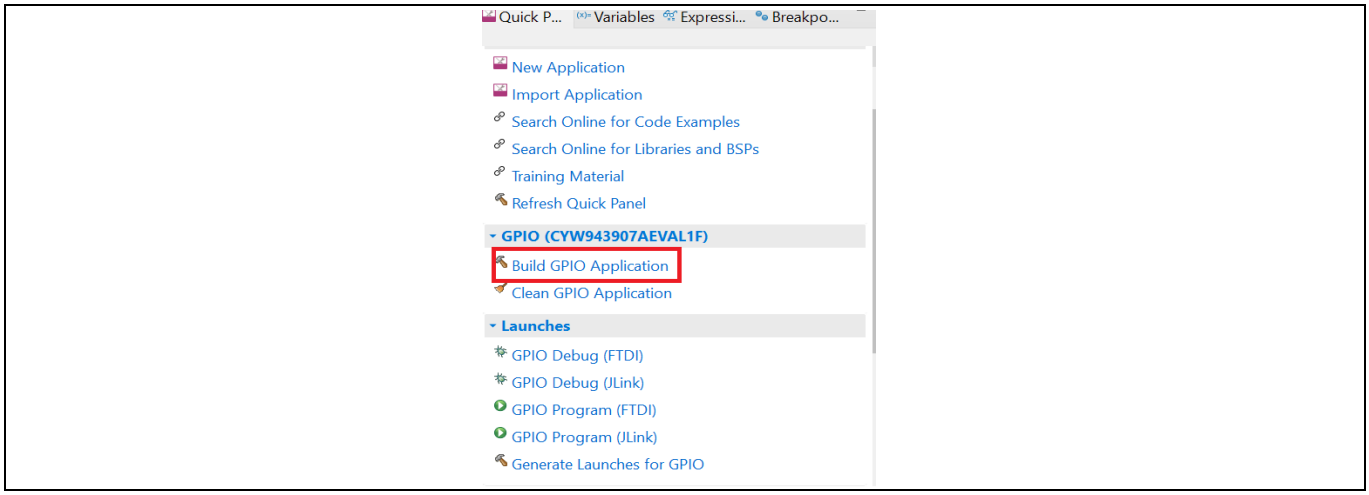

Figure 10 Build the GPIO application

5. In the Project Explorer, select the desired project. Then, in the **Quick Panel**, click the **<app-name> Program (FTDI)** link for programming the AIROC™ CYW43907 Wi-Fi connectivity processor and for debugging select **<app-name> Debug (FTDI)**.

   To add breakpoints, right-click at the point where you want to add the breakpoint and select **Add Breakpoint**.

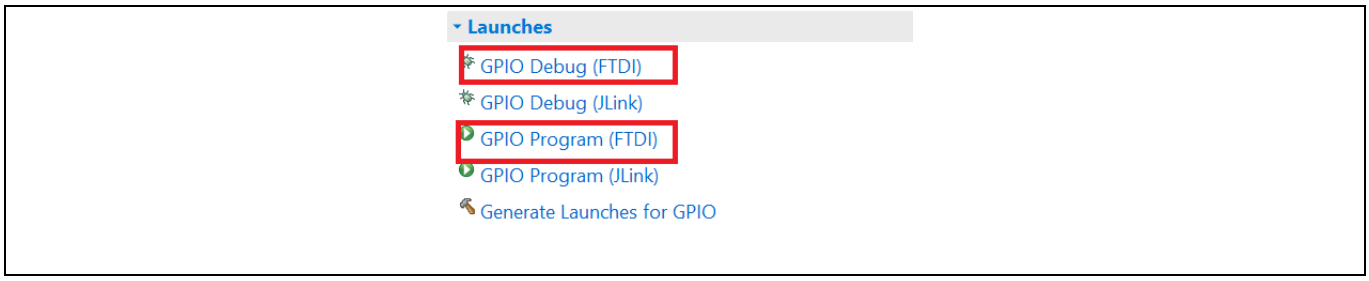

Figure 11 Options for programming and debugging

Kit operation

Controls for debugging like play, step into, step return, and step over are marked in **Figure 12**.

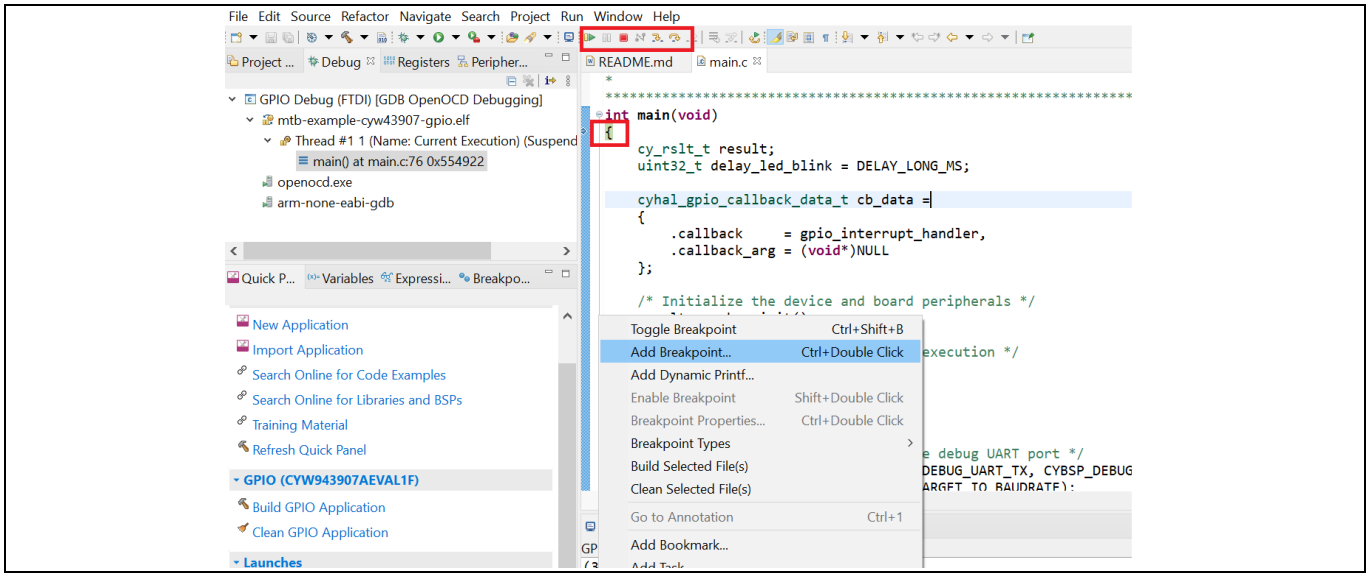

Figure 12 Controls for debugging

If you prefer to use the command-line option to create the project, see the *README.md* file of the code example.

4 Hardware

4.1 User switches

There are two user switches available on the board named USER_1 and USER_2. **Table 3** shows the pin names and enumeration used in ModusToolbox™ software for the switches.

Table 3 User switch

AIROC™ CYW43907 pin name	Enumeration in ModusToolbox™ software
USER_1 (SW3)	CYBSP_SW3
USER_1 (SW1)	CYBSP_SW1

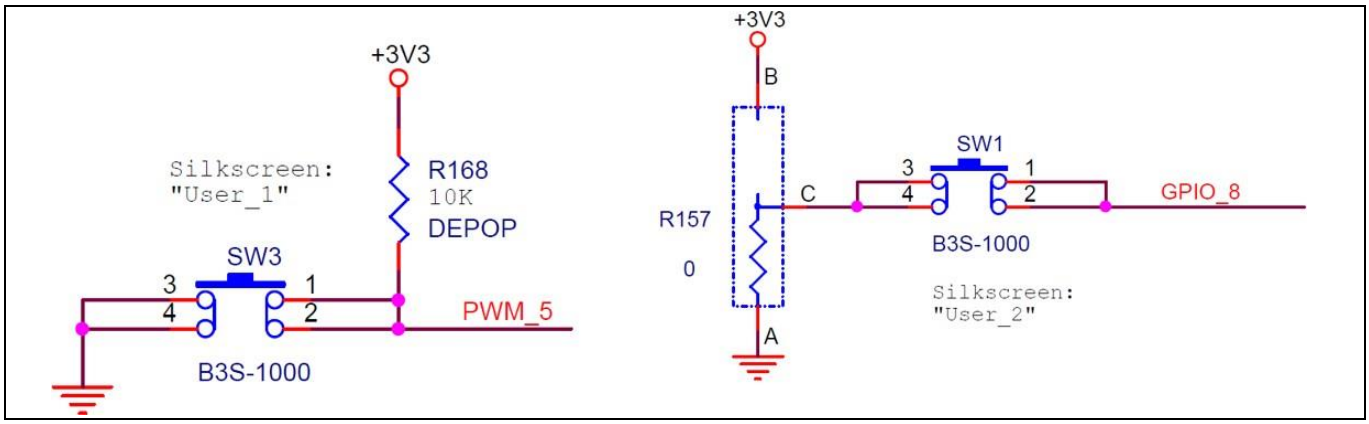

Figure 13 User switch circuit diagram

4.1.1 LEDs

There are two user LEDs available named LED_1 and LED_2. **Table 4** shows the pin name and enumeration used in ModusToolbox™ software for these LEDs.

Note: PIN_PWM_3 LED will not act as a normal GPIO.

Table 4 User LEDs

Pin name on AIROC™ CYW43907	Enumeration in ModusToolbox™ software
LED_1	CYBSP_LED1
LED_2	CYBSP_LED2

4.1.2 Reset control

The AIROC™ CYW43907 device can be reset using the “Target Reset” switch **SW2** or a reset command from the onboard programmer/debugger and serial interface chip, as shown in **Figure 14**. The CYW43907 datasheet states that HIB_REG_ON_IN needs to be delayed by at least two cycles of the 32.768-kHz clock after VBAT and VDDIO have reached 90% of their final values. To ensure proper boot-up, the RC delay circuit for HIB_REG_ON_IN is essential as shown in **Figure 15**. See **4.1 User switches** for details on the RC delay circuit.

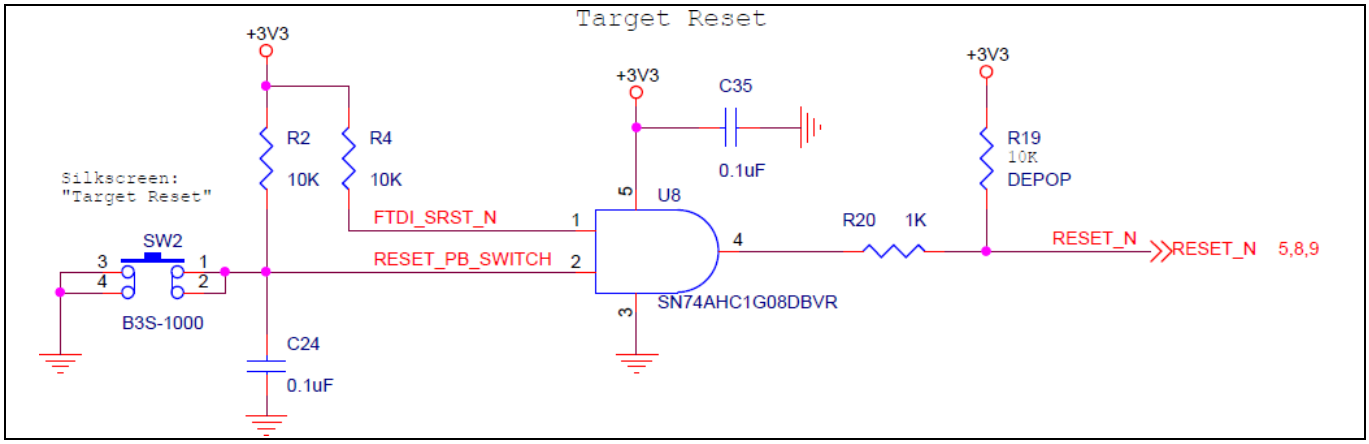

Figure 14 Reset circuit

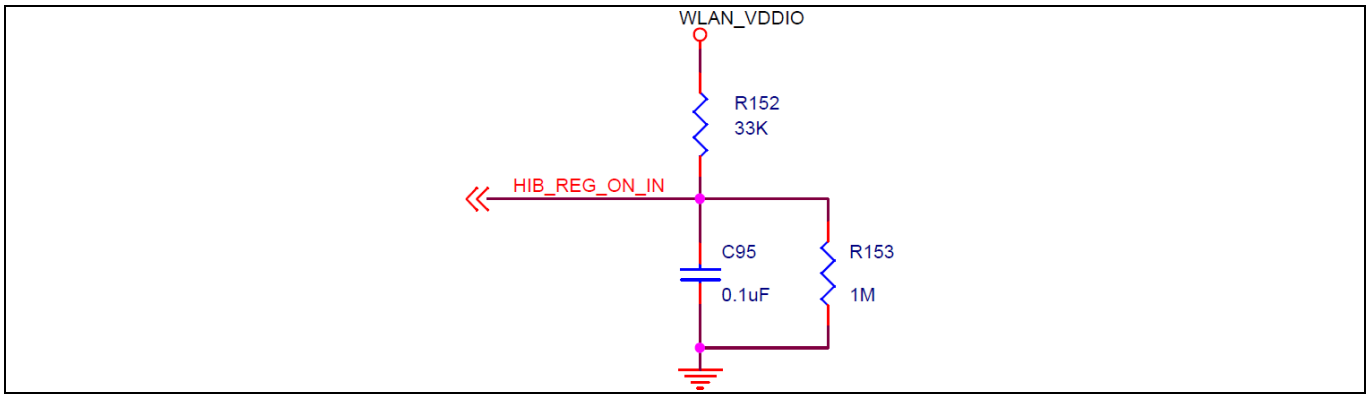

Figure 15 HIB_REG_ON_IN RC delay circuit

4.2 Connectors

Header J6 on the CYW943907AEVAL1F EVK is a 44-pin header containing I2C, SDIO, UART, SPI, PWM lines, and I/Os. Note that some signals are shared with the header compatible with Arduino (UART0 Tx/Rx) and onboard programmer/debugger chip (UART1). **Table 5** illustrates the J6 pinout.

Table 5 J6 header pinout

Evaluation board header	CYW43907 pin name	Enumeration in ModusToolbox™ software
J6.1	PWM_4	PIN_PWM_4
J6.2	PWM_5	PIN_PWM_5
J6.3	I2S0_MCK	PIN_I2S_MCLK0
J6.4	I2S0_SD_OUT	PIN_I2S_SCLK0
J6.5	I2S0_SCK_BCLK	PIN_I2S_SDATAO0
J6.6	I2S0_WS_LRCLK	PIN_I2S_LRCLK0
J6.7	PWM_3	PIN_PWM_3
J6.8	GND	GND
J6.9	SPI_1_CLK	PIN_SPI_1_CLK
J6.10	I2S1_SD_OUT	PIN_I2S_SDATAO1
J6.11	SPI_1_MISO	PIN_SPI_1_MISO
J6.12	SPI_0_CLK	PIN_SPI_0_CLK
J6.13	SPI_1_MOSI	PIN_SPI_1_MOSI
J6.14	SPI_0_MOSI	PIN_SPI_0_MOSI
J6.15	SPI_1_CS	PIN_SPI_1_CS
J6.16	SPI_0_CS	PIN_SPI_0_CS
J6.17	SPI_0_MISO	PIN_SPI_0_MISO
J6.18	UART0_RXD_IN	PIN_UART0_RXD
J6.19	GND	GND
J6.20	UART0_TXD_OUT	PIN_UART0_TXD
J6.21	USB2_HOST_DEV_SEL	USB2_HOST_DEV_SEL
J6.22	UART0_CTS_IN	PIN_UART0_CTS
J6.23	I2C_0_SCL	PIN_I2C0_CLK
J6.24	UART0_RTS_OUT	PIN_UART0_RTS
J6.25	I2C_0_SDA	PIN_I2C0_SDATA
J6.26	I2S1_MCK	PIN_I2S_MCLK1
J6.27	I2S1_WS_LRCLK	PIN_I2S_LRCLK1
J6.28	GND	GND
J6.29	I2S1_SCK_BCLK	PIN_I2S_SCLK1
J6.30	SDIO_DATA_1	PIN_SDIO_DATA_1
J6.31	SDIO_DATA_0	PIN_SDIO_DATA_0
J6.32	SDIO_CLK	PIN_SDIO_CLK

Evaluation board header	CYW43907 pin name	Enumeration in ModusToolbox™ software
J6.33	SDIO_CMD	PIN_SDIO_CMD
J6.34	SDIO_DATA_3	PIN_SDIO_DATA_3
J6.35	SDIO_DATA_2	PIN_SDIO_DATA_2
J6.36	RF_SW_CTRL_6_UART1_RXD	PIN_RF_SW_CTRL_6
J6.37	UART1_TXD	PIN_RF_SW_CTRL_7
J6.38	RF_SW_CTRL_8_UART2_RXD	PIN_RF_SW_CTRL_8
J6.39	UART2_TXD	PIN_RF_SW_CTRL_9
J6.40	HIB_WAKE	HIB_WAKE
J6.41	HIB_LPO_SEL	HIB_LPO_SEL
J6.42	HIB_REG_ON_IN	HIB_REG_ON_IN
J6.43	USB2_DN	USB2_DN
J6.44	USB2_DP	USB2_DP

4.2.1 Headers compatible with Arduino

J9, J13, J12, and J10 are headers compatible with Arduino available in the CYW943907AEVAL1F EVK. **Table 6** shows the pinout. Note the following points while connecting an Arduino shield to the board:

- 5-V pin of header (J9) is not connected to the board.
- The maximum current that an Arduino shield can sink from the board depends on the application that is running. In general, 100 mA is the worst-case scenario.
- The Arduino analog reference is connected to the 3V3 (3.3 V) power supply through R21, which is not populated by default. In other words, the analog reference is not driven by default.

Table 6 Arduino header pinout

Eval board header	CYW43907 pin name/ kit signal name	Arduino header name	Enumeration in ModusToolbox™ software
J10.1	GPIO_0	D0	PIN_GPIO_0
J10.2	GPIO_1	D1	PIN_GPIO_1
J10.3	GPIO_13	D2	PIN_GPIO_13
J10.4	GPIO_7	D3	PIN_GPIO_7
J10.5	GPIO_14	D4	PIN_GPIO_14
J10.6	GPIO_16	D5	PIN_GPIO_16
J10.7	GPIO_15	D6	PIN_GPIO_15
J10.8	I2S0_SD_IN	D7	PIN_I2S_SDATAI0
J12.1	I2S1_SD_IN	D8	PIN_I2S_SDATAI1
J12.2	PWM_4	D9	PIN_PWM_4
J12.3	GPIO_11	D10	PIN_GPIO_11
J12.4	GPIO_10	D11	PIN_GPIO_10
J12.5	GPIO_12	D12	PIN_GPIO_12
J12.6	GPIO_9	D13	PIN_GPIO_9

Eval board header	CYW43907 pin name/ kit signal name	Arduino header name	Enumeration in ModusToolbox™ software
J12.7	GND	GND	N/A
J12.8	ARD_AREF	AREF	N/A
J12.9	I2C_1_SDA	SDA	PIN_I2C1_SDATA
J12.10	I2C_1_SCL	SCL	PIN_I2C1_CLK
J13.1	ARD_AD0	A0	N/A
J13.2	ARD_AD1	A1	N/A
J13.3	ARD_AD2	A2	N/A
J13.4	ARD_AD3	A3	N/A
J13.5	ARD_AD4_SDA	A4	N/A
J13.6	ARD_AD5_SCL	A5	N/A
J9.1	NC	NC	N/A
J9.2	ARD_IOREF	IOREF	N/A
J9.3	ARD_RESET	RESET	N/A
J9.4	3V3	3.3V	N/A
J9.5	NC	5V	N/A
J9.6	GND	GND	N/A
J9.7	GND	GND	N/A
J9.8	VIN_EXT	VIN	N/A

4.3 UART port configuration

The CYW943907AEVAL1F kit has three UART ports: slow UART, fast UART, and GCI UART. Slow UART and GCI UART are 2-wire interfaces while fast UART is a 4-wire interface that can support up to a 3 Mbps baud rate. Slow UART is routed to the onboard programmer/debugger chip for UART to USB communication.

Note: For using debug UART, call the `cy_retarget_io_init( )` function with the slow UART pins in your application.

Table 7 UART header pinout

Pin name on CYW43907	Header pin number	Enumeration in ModusToolbox™ software
RF_SW_CTRL_6	J6:36 (slow UART RXD)	PIN_RF_SW_CTRL_6
RF_SW_CTRL_7	J6:37 (slow UART TXD)	PIN_RF_SW_CTRL_7
UART0_RXD	J6:18 (fast UART RXD)	PIN_UART0_RXD
UART0_TXD	J6:20 (fast UART TXD)	PIN_UART0_TXD
UART0_CTS	J6:22 (fast UART CTS)	PIN_UART0_CTS
UART0_RTS	J6:24 (fast UART RTS)	PIN_UART0_RTS
RF_SW_CTRL_8	J6:38 (GCI UART RXD)	PIN_RF_SW_CTRL_8
RF_SW_CTRL_9	J6:39 (GCI UART TXD)	PIN_RF_SW_CTRL_9

4.4 PWM

There are three dedicated PWM outputs available on CYW943907AEVAL1F.

Table 8 PWM header pinout

Pin name on CYW43907	Header pin number	Enumeration in ModusToolbox™ software
J6.7	PIN_3	PIN_PWM_3
J6.1	PIN_4	PIN_PWM_4
J6.2	PIN_5	PIN_PWM_5

5 Code examples

The following code examples are available for the AIROC™ CYW43907 Wi-Fi connectivity processor in ModusToolbox™ software.

Table 9 Code examples

Code example name	Description	Link
AIROC™ CYW43907 Hello World	This code example demonstrates the simple UART communication by printing the “Hello world” message on a terminal and blinks an LED with a timer resource using AIROC™ CYW43907 Wi-Fi connectivity processor.	**https://github.com/Infineon/mtb-example-cyw43907-hello-world/blob/master/README.md**
AIROC™ CYW43907 UART transmit and receive	This code example demonstrates the UART (HAL) transmit and receive operation in AIROC™ CYW43907 Wi-Fi connectivity processor. The application uses a serial terminal to read data and to echo back the received data. The UART resource is configured to do both transmit and receive operations.	**https://github.com/Infineon/mtb-example-cyw43907-uart-tx-rx/blob/master/README.md**
AIROC™ CYW43907 GPIO	This example demonstrates the use of a GPIO configured as an input pin to generate interrupts on AIROC™ CYW43907 Wi-Fi connectivity processor. The GPIO signal interrupts the CPU and executes a user-defined ISR.	**https://github.com/Infineon/mtb-example-cyw43907-gpio/blob/master/README.md**
AIROC™ CYW43907 PWM	This code example generates a square wave using the PWM driver. An LED connected to the PWM output pin blinks at 1000 Hz with variable duty cycle.	**https://github.com/Infineon/mtb-example-cyw43907-pwm/blob/master/README.md**
AIROC™ CYW43907 watchdog timer	This example explains how to set up a watchdog timer (WDT) using the WDT HAL resource. The WDT resets the device if it is not serviced within the configured timeout interval. This helps in recovering the program from an unintended lock-up.	**https://github.com/Infineon/mtb-example-cyw43907-wdt/blob/master/README.md**
AIROC™ CYW43907 I2C master	This code example demonstrates the I2C (HAL) operation in AIROC™ CYW43907 Wi-Fi connectivity processor. The application uses I2C to read the data from **CY8CKIT-032** PSoC™ analog front end (AFE) Arduino shield and displays the result in the UART serial terminal every time the user presses the button.	**https://github.com/Infineon/mtb-example-cyw43907-i2c/blob/master/README.md**

Code examples

Code example name	Description	Link
AIROC™ CYW43907 Wi-Fi scan	This example demonstrates how to configure different scan filters provided in the Wi-Fi Connection Manager (WCM) middleware and scan for the available Wi-Fi networks.	**https://github.com/Infineon/mtb-example-cyw43907-wifi-scan/blob/master/README.md**
AIROC™ CYW43907 TCP server	This code example demonstrates the implementation of a TCP server with AIROC™ CYW43907 connectivity processor.	**https://github.com/Infineon/mtb-example-cyw43907-tcp-server/blob/master/README.md**
AIROC™ CYW43907 TCP client	This code example demonstrates the implementation of a TCP client with AIROC™ CYW43907 connectivity processor.	**https://github.com/Infineon/mtb-example-cyw43907-tcp-client/blob/master/README.md**
AIROC™ CYW43907 UDP server	This code example demonstrates the implementation of a UDP server with AIROC™ CYW43907.	**https://github.com/Infineon/mtb-example-cyw43907-udp-server/blob/master/README.md**
AIROC™ CYW43907 UDP client	This code example demonstrates the implementation of a UDP client with AIROC™ CYW43907.	**https://github.com/Infineon/mtb-example-cyw43907-udp-client/blob/master/README.md**
AIROC™ CYW43907 Secure TCP server	This code example demonstrates the implementation of a secure TCP server with AIROC™ CYW43907 Wi-Fi connectivity processor.	**https://github.com/Infineon/mtb-example-cyw43907-secure-tcp-server/blob/master/README.md**
AIROC™ CYW43907 secure TCP client	This code example demonstrates the implementation of a secure TCP client with AIROC™ CYW43907 Wi-Fi connectivity processor.	**https://github.com/Infineon/mtb-example-cyw43907-secure-tcp-client/blob/master/README.md**
AIROC™ CYW43907 secure HTTP server	This code example demonstrates the implementation of an HTTPS server with AIROC™ CYW43907 Wi-Fi connectivity processor.	**https://github.com/Infineon/mtb-example-cyw43907-secure-http-server/blob/master/README.md**
AIROC™ CYW43907 MQTT client	This code example demonstrates implementing an MQTT client using the MQTT client library. The library uses the AWS IoT device SDK MQTT client library that includes an MQTT 3.1.1 client.	**https://github.com/Infineon/mtb-example-cyw43907-mqtt-client/blob/master/README.md**

Revision history

Major changes since the last revision

Date	Version	Description
2021-11-25	**	Initial release.
2022-10-10	*A	No technical updates. Completing Sunset Review.

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

Edition 2022-10-10
Published by

Infineon Technologies AG

81726 Munich, Germany

Do you have a question about this document?

Go to www.cypress.com/support

Document reference
002-34339 Rev. *A

IMPORTANT NOTICE
The information given in this document shall in no event be regarded as a guarantee of conditions or characteristics ("Beschaffenheitsgarantie").

With respect to any examples, hints or any typical values stated herein and/or any information regarding the application of the product, Infineon Technologies hereby disclaims any and all warranties and liabilities of any kind, including without limitation warranties of non-infringement of intellectual property rights of any third party.

In addition, any information given in this document is subject to customer's compliance with its obligations stated in this document and any applicable legal requirements, norms and standards concerning customer's products and any use of the product of Infineon Technologies in customer's applications.

The data contained in this document is exclusively intended for technically trained staff. It is the responsibility of customer's technical departments to evaluate the suitability of the product for the intended application and the completeness of the product information given in this document with respect to such application.

For further information on the product, technology, delivery terms and conditions and prices please contact your nearest Infineon Technologies office (**www.infineon.com**).

WARNINGS
Due to technical requirements products may contain dangerous substances. For information on the types in question please contact your nearest Infineon Technologies office.

Except as otherwise explicitly approved by Infineon Technologies in a written document signed by authorized representatives of Infineon Technologies, Infineon Technologies' products may not be used in any applications where a failure of the product or any consequences of the use thereof can reasonably be expected to result in personal injury.