AN1280: RS9116W Power Save Application Notes

This application note provides information about the various Power Saves supported by the RS9116W. It also includes recommended modifications for optimizing application performance. Additionally, this application note provides detailed current consumption metrics for the RS9116W when it is calibrated and operating in different states.

KEY POINTS

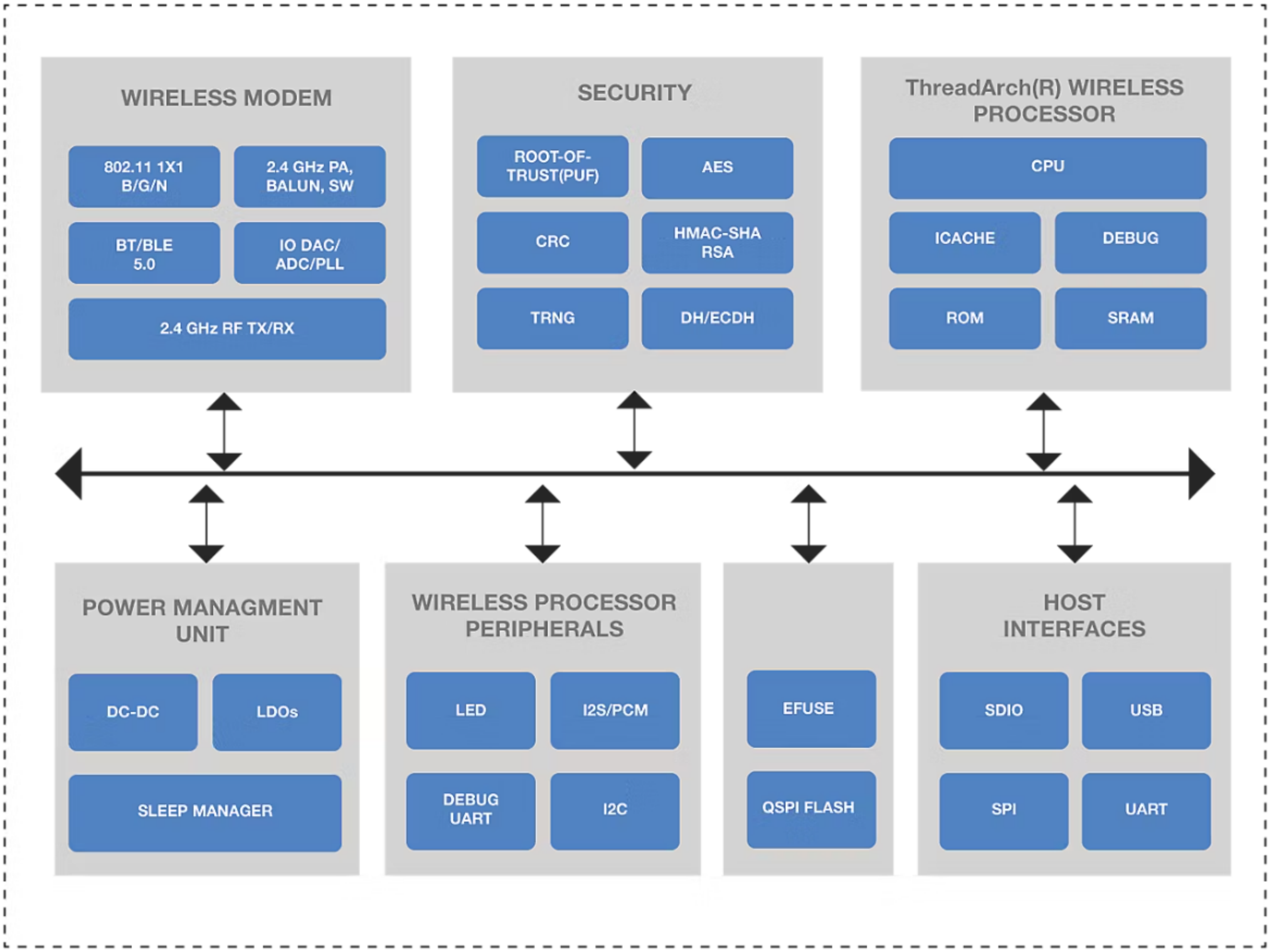
- RS9116W's current consumption during various operational modes
- Refer to Power Save Examples in the WiSeConnect™ SDK
- Choosing Power Save Modes as per application requirements
- Current measurement methods
- Power optimization techniques

Table of Contents

1. Device Architecture

The RS9116W chipsets have an internal Power Managament Unit (PMU) that generates all the voltages required by the chipset to operate from a wide variety of input sources.

The top-level view of power domains subsystem of RS9116W is as below:



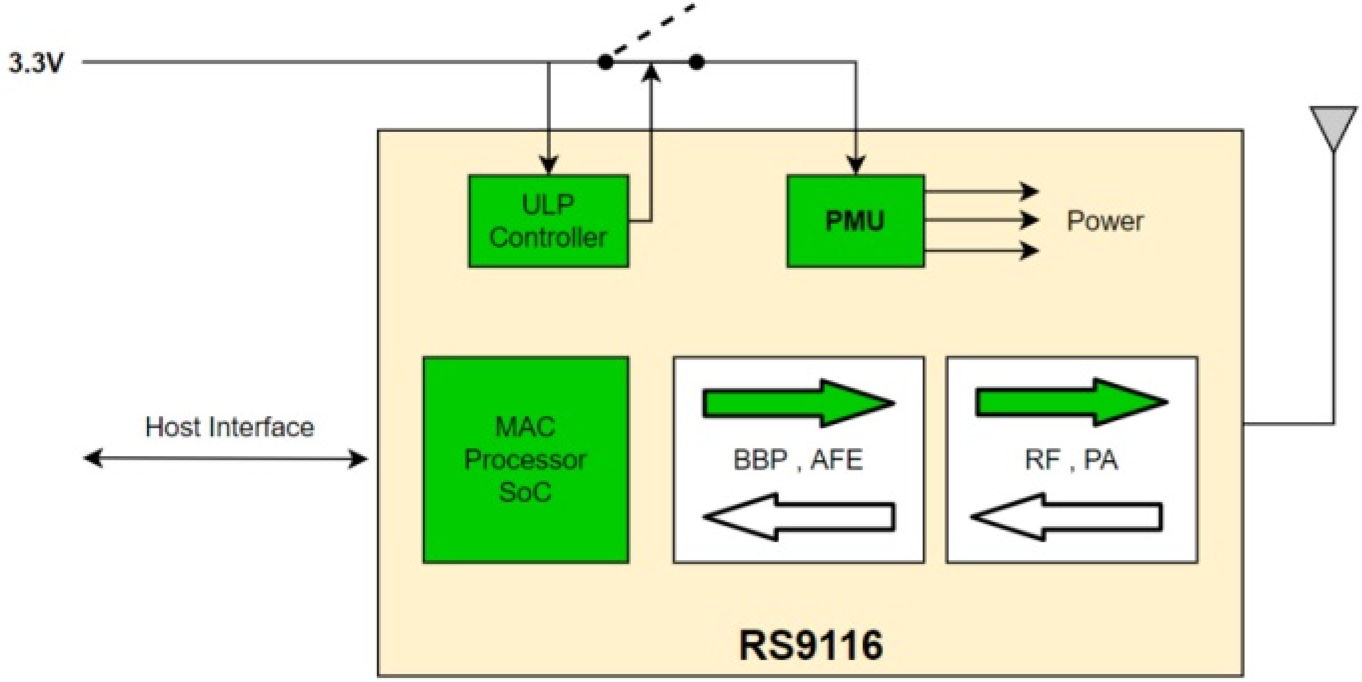
The Low Power Subsystem contains circuitry that operates and gives an extremely low current consumption. This includes a timer running off a 32 kHz clock and input pins monitored for activity enabling and sensing external inputs in Ultra-Low Power (ULP) mode. When the device is put to its lowest power sleep state, only the Low Power Subsystem is active. When the wake-up event occurs, the ULP logic present in the Low Power Subsystem turns on the PMU and/or external power switches and enables bringing the rest of the device into normal operation. In regular operation, the device may still be able to turn on or off selective portions by itself based on the Power Save used.

2. Operational Modes

The RS9116W can be in three different operational modes, each with varying levels of current consumption. The current consumption differs depending on the operational mode.

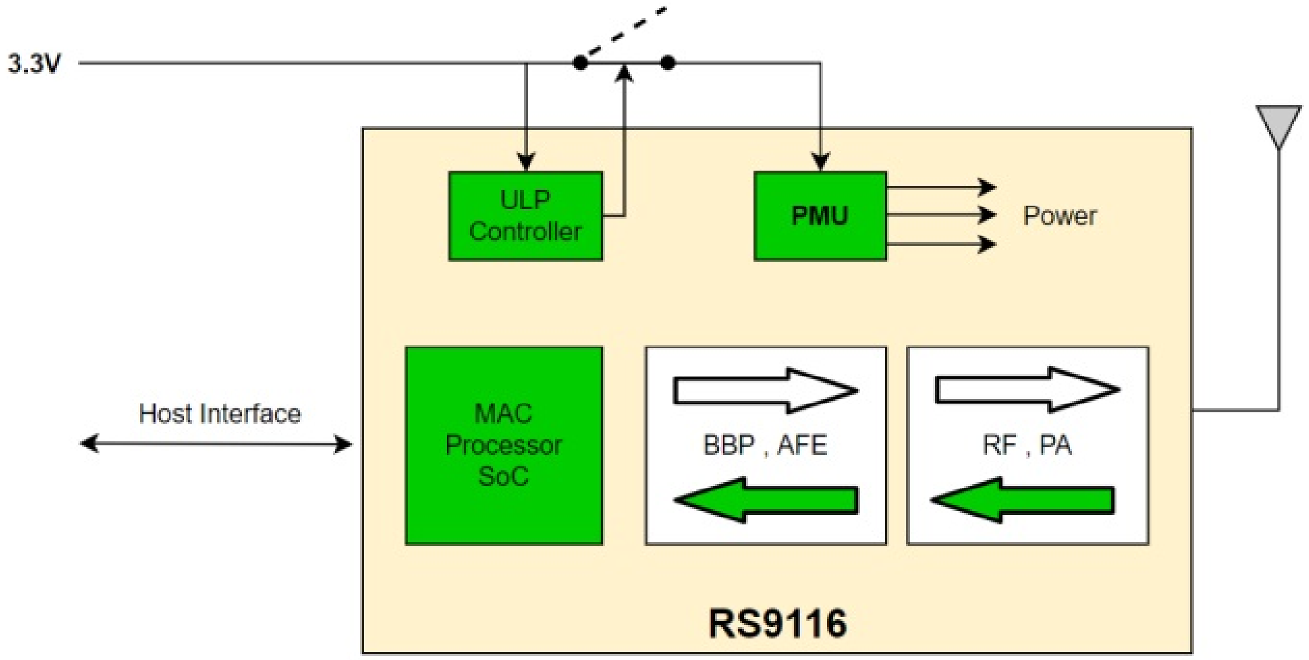
2.1 Transmit Mode

This is generally the highest power-consuming mode. Transmit mode has all components of the RS9116W ON except the receiver section of the RF, AFE, and BBP.



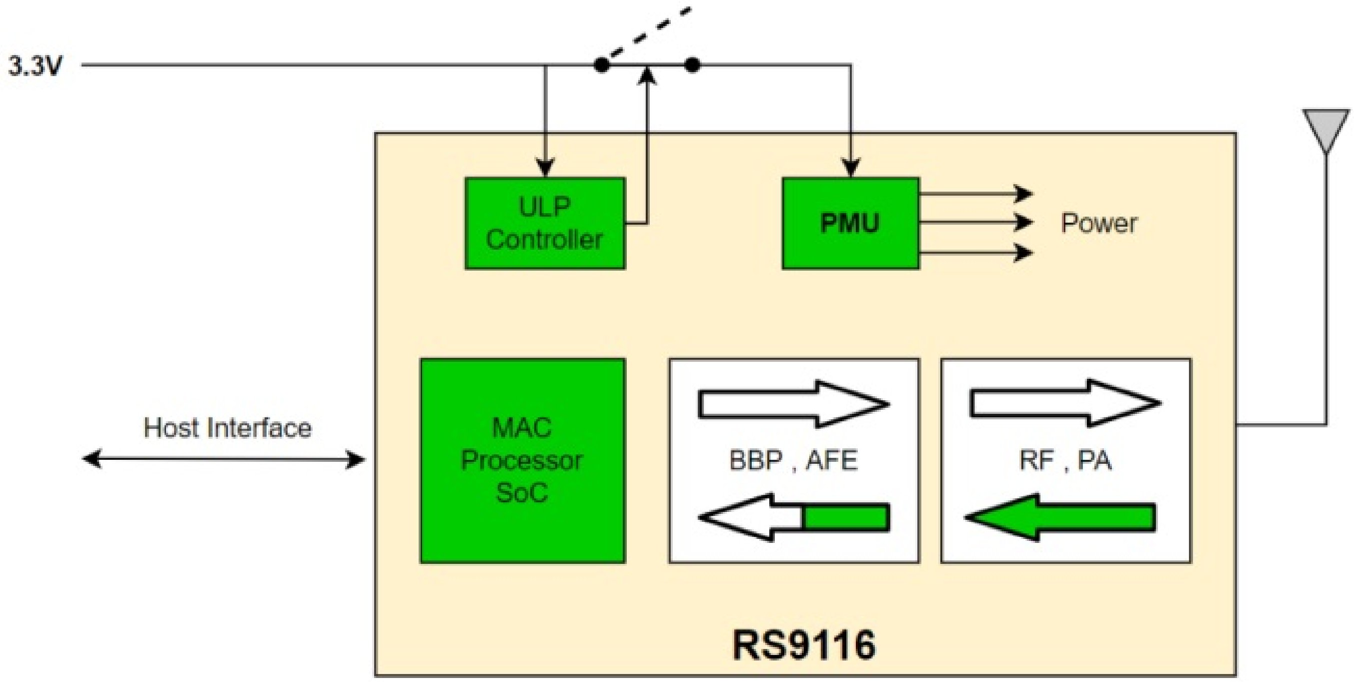
2.2 Receive Mode

In Receive, the transmit sections of the BBP, AFE, and RF are not powered. In addition, the PA is also not powered.



2.3 Listen Mode

This mode is a specialized subset of the Receive mode, where specific sections of the receiver are either powered down or transitioned to a lower power state when no packets are actively being received. A portion of the AFE and BBP remains off until the beginning of a packet is detected in the receiver.



3. Basic Low Power Modes

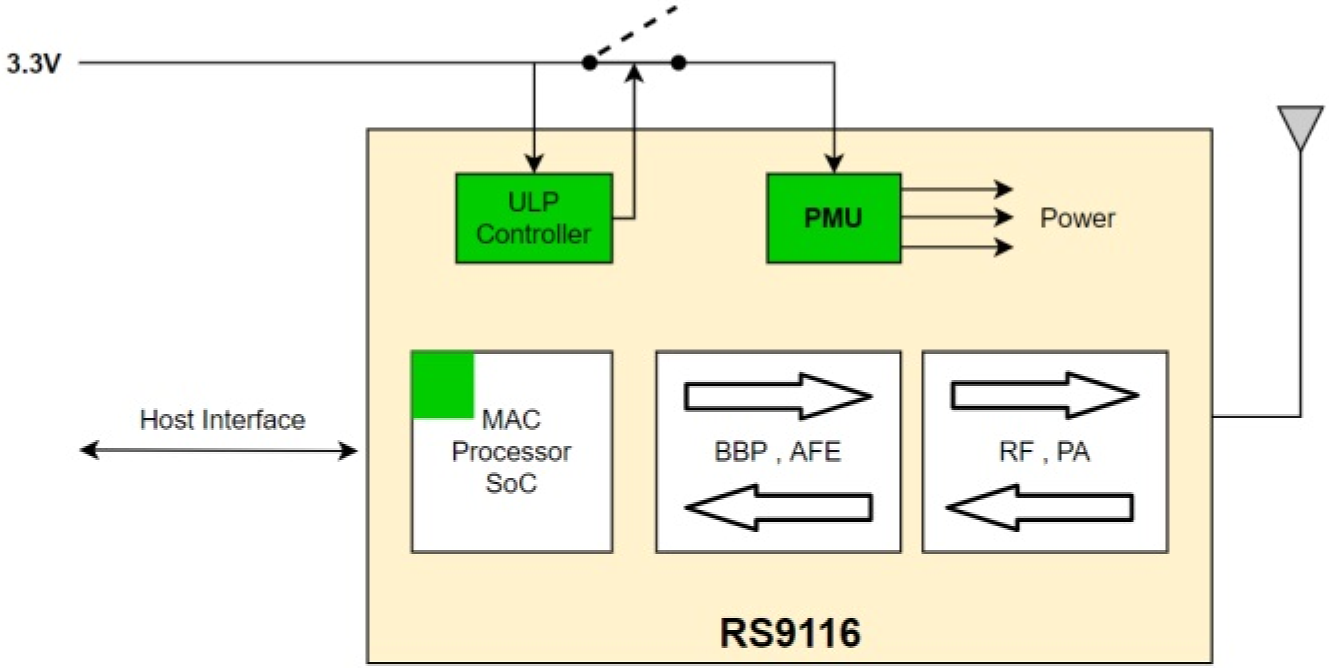
The RS9116W can operate in two fundamental low power modes:

- Low Power Mode
- Ultra-Low Power Mode

3.1 Low Power Mode

In this mode, the internal modem portions are turned off while the SoC is kept on, in a low power state. A Sleep State Machine in the SoC is active and responds to wake-up events. The host interface is active, and the device is ready to receive packets from the host. Network processor maintains system state and gates all internal high-frequency clocks.

In this mode, the BBP, AFE, and RF are powered OFF.

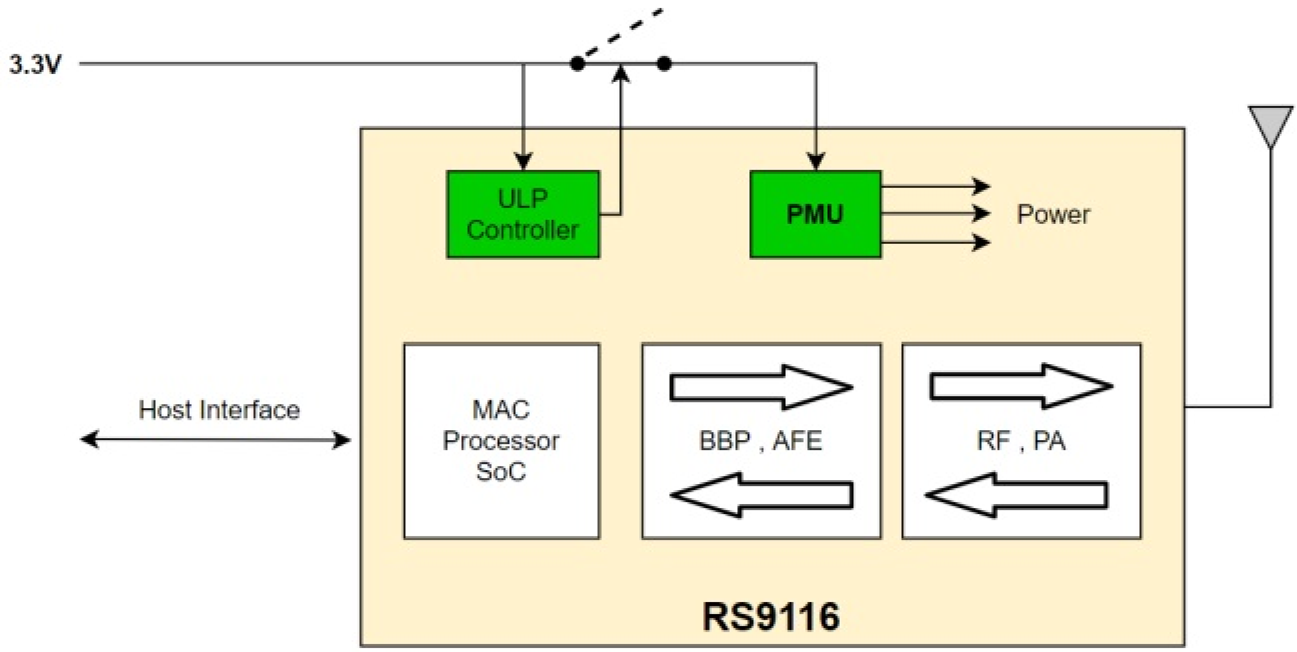


3.2 Ultra-Low Power Mode

In this mode, the entire SoC, except for the ULP sub-system, is shut down. The host interface is not active in this mode. This mode is subdivided into two categories based on RAM retention:

- With RAM Retention
- Without RAM Retention

In this mode, all the section of the SoC are shut down, except the special ULP circuit.



4. Handshake Mechanisms

A Handshake Mechanism refers to the method used by the host to wake up or set the RS9116W into sleep. There are two Handshake Mechanisms used in the RS9116W:

- GPIO-based
- Message-based

4.1 GPIO Based

The GPIO-based Handshake, as the name specifies uses GPIOs for handshake. During its Power Save Mode, the RS9116W uses the following GPIOs for receiving sleep/wake requests from the host, and for sending its sleep/wake indications to the host.

In Ultra Low Power(ULP) Mode

- UULP_VBAT_GPIO_2 (Input to RS9116W): The host uses this GPIO for the following scenarios:
  - If the host application wants to send data/command to RS9116W, it makes a wake request by asserting this GPIO.
  - After communicating with RS9116W, the host lets the RS9116W sleep by de-asserting this GPIO.
- UULP_VBAT_GPIO_0/UULP_VBAT_GPIO_3 (Output from RS9116W): The RS9116W uses either of these GPIOs to send sleep/wake indications to the host.

Software Configuration:

- For making Ultra Low Power configuration with GPIO-based handshake, enable the `FEAT_ULP_GPIO_BASED_HANDSHAKE` feature in the `RSI_FEATURE_BIT_MAP` of Opermode command.
- By default, UULP_VBAT_GPIO_3 is used by the host to receive sleep/wake indications from the RS9116W. To use UULP_VBAT_GPIO_0 instead for receiving these indications, enable the `RSI_FEAT_SLEEP_GPIO_SEL_BITMAP` feature in `RSI_CONFIG_FEATURE_BITMAP`.

In Low Power(LP) Mode

- ULP_GPIO_5 (Input to RS9116W): The host uses this GPIO for the following scenarios:
  - If the host application wants to send data/command to RS9116W, it makes a wake request by asserting this GPIO.
  - After communicating with RS9116W, the host lets the RS9116W sleep by de-asserting this GPIO.
- UULP_VBAT_GPIO_0/UULP_VBAT_GPIO_3 (Output from RS9116W): The RS9116W uses either of these GPIOs to send sleep/wake indications to the host.

Software Configuration:

- For making Low Power configuration with GPIO-based handshake, enable the `FEAT_LP_GPIO_BASED_HANDSHAKE` feature in the `RSI_FEATURE_BIT_MAP`.
- By default, UULP_VBAT_GPIO_3 is used by the host to receive sleep/wake indications from the RS9116W. To switch to using UULP_VBAT_GPIO_0 for these indications, enable the `RSI_FEAT_SLEEP_GPIO_SEL_BITMAP` feature in `RSI_CONFIG_FEATURE_BITMAP`.

Note: The 32 kHz clock source needed in ULP Mode/LP Mode can be configured via software. For available options, refer to the RS9116X-SB-EVK1 data sheet and the RS9116X-DB-EVK1 data sheet.

4.2 Message Based

In cases where Host MCU has limited GPIOs available Message (MSG) based handshake can be used. RS9116W will indicate via a message.

- The RS9116W wakes up every configured listen interval or Beacon Interval or DTIM (Delivery Traffic Indication Message) interval and sends a Wake-up message (WKP) to the host.
- The host sends data/command to the RS9116W.
- The RS9116W requests for going into sleep via a Sleep message (SLP).
- The host gives an acknowledgment (ACK) to the RS9116W to go into the sleep state.
- The RS9116W does not go to sleep until it gets an ACK from the host.
- The RS9116W cannot be woken up asynchronously.

Note:

Message-based power saving involves continuous monitoring and frequent wake-ups to handle communication, which increases power consumption due to the processing required for message reception and response. In contrast, GPIO-based power saving is much more efficient, as it relies on low power GPIOs for sleep wake events, allowing it to remain in a deep sleep state for extended periods, significantly reducing power usage.

5. Power Save Modes of RS9116W

The Power Save Modes (or Power save Procedures) of RS9116W can be broadly classified into three categories:

- Active mode
- Connected sleep modes
- Unconnected sleep (deep sleep) modes. For connected and unconnected Power Save Modes, there are sub Power Save Modes in AT and SAPI modes.

5.1 Classification of Power Save Modes in AT Mode

Table 5.1. Power Save Mode Classifications

Power Save Mode Value	Power Save Mode	Handshake with Host	Description
0	Power Save Mode 0	No Handshake	Active Mode
1	Power Save Mode 1	No Handshake	Standby Associated Power Save Mode (Connected Sleep)
2	Power Save Mode 2	GPIO-based	GPIO-based Standby Associated Power Save Mode (Connected Sleep)
3	Power Save Mode 3	Message-based	Message-based Standby Associated Power Save Mode (Connected Sleep)
8	Power Save Mode 8	GPIO-based	GPIO-based Deep Sleep mode with/without RAM Retention (Unconnected sleep)
9	Power Save Mode 9	Message-based	Message-based Deep Sleep Mode with/without RAM Retention (Unconnected sleep)

5.2 Classification of Power Save Modes in SAPI Mode

Table 5.2. Power Save Modes

Power Save Mode Value	Power Save Mode	Handshake with Host	Description
0	Power Save Mode 0	No Handshake	Active Mode
1	Power Save Mode 1	No Handshake	Standby Associated Power Save Mode (Connected Sleep)
2	Power Save Mode 2	GPIO-based/ Message-based	GPIO-based/Message-based Standby Associated Power Save Mode (Connected Sleep)
8	Power Save Mode 8	GPIO-based/ Message-based	GPIO-based/Message-based Deep Sleep (Unconnected sleep) with RAM retention
10	Power Save Mode 10	GPIO-based/ Message-based	GPIO-based/Message-based Deep Sleep(Unconnected sleep) without RAM retention

Note:

1. In the SAPI framework, RAM retention settings are not user-configurable. Instead, the SAPI driver automatically selects the appropriate RAM retention configuration based on the chosen power save mode.
2. In AT mode, Power Save Mode 2 is equivalent to Power Save Mode 2 - GPIO-based Standby Associated Power Save Mode (Connected Sleep) in SAPI.
3. In AT mode, Power Save Mode 3 is equivalent to Power Save Mode 2 - Message-based Standby Associated Power Save Mode (Connected Sleep) in SAPI.
4. In AT mode, Power Save Mode 8 with RAM retention is equivalent to Power Save Mode 8 - GPIO-based Deep Sleep (Unconnected Sleep) with RAM retention in SAPI.
5. In AT mode, Power Save Mode 9 with RAM retention is equivalent to Power Save Mode 8 - Message-based Deep Sleep (Unconnected Sleep) with RAM retention in SAPI.
6. In AT mode, Power Save Mode 8 without RAM retention is equivalent to Power Save Mode 10 - GPIO-based Deep Sleep (Unconnected Sleep) without RAM retention in SAPI.
7. In AT mode, Power Save Mode 9 without RAM retention is equivalent to Power Save Mode 10 - Message-based Deep Sleep (Unconnected Sleep) without RAM retention in SAPI.
8. The explanation regarding Power Save Modes applies exclusively to AT mode.

5.3 Connected Sleep Modes

5.3.1 Sleep/Wake State Switching Mechanism

1. The connecting station (here RS9116W) makes an Association request to the AP. In the Association frame, it sends the Listen Interval which indicates how often the station wakes to listen to the AP beacon frames. Based on the Listen Interval, the AP holds the data frames destined for the station.
2. The AP shall respond with an Association Response frame in which it specifies the Association Identifier (AID) to its connecting station.
3. The RS9116W can be configured to wake every Delivery Traffic Indication Message (DTIM) Interval or Beacon Interval or Listen interval or Target Wake Time (TWT) Wake Interval during its Connected Power Save Mode.
    - Beacon Interval-based wakeup: Beacon Interval is the period between two subsequent beacon frames transmitted by AP. The station wakes every beacon interval.
    - DTIM Interval-based wakeup: The DTIM period specifies how often an AP beacon includes Buffered Traffic Indication to its connected clients via the TIM element in the beacon frame. When the AP includes TIM information in a beacon frame, the beacon is called a DTIM beacon. DTIM interval is the time between two subsequent DTIM beacons transmit-ted by AP. DTIM Interval = Beacon Interval*DTIM Period.
    - Listen Interval-based wakeup: Based on the listen interval configured in the application, the station wakes up at the nearest integral multiples of DTIM Interval/Beacon Interval broadcasted by the connected AP which is just less than or equal to the Listen Interval.

- An STA can wake up based on the AP DTIM(Delivery Traffic Indication Map) or listen-interval(set on the STA side).
- When you want to save power from unnecessarily waking up every DTIM, you can go with listen-interval based.
- An AP transmits any multicast or broadcast data available every DTIM when a PSP station is associated with the AP. If not, it will send the multicast or broadcast data immediately.

Note: Configuring larger listen intervals greater than 1000 ms might lead to AP disconnecti ng the RS9116W. It is highly recommended to use 1000 ms.

5.3.2 Power Save Mode 1

Power Save Mode 1 is Standby Associated Power Save Mode (Connected Sleep).

Once the RS9116W is configured in Power Save Mode 1, it periodically wakes up based on its configured listen interval or Beacon Interval or the DTIM interval set in the connected access point, allowing only the RF section of the SoC to enter Power Save Mode while the SoC operates normally.This command should only be issued while the RS9116W is connected to the AP, and after the configuration, the host can send subsequent commands, enabling the RS9116W to receive data from the host anytime while data reception of RS9116W from the external/remote devices can happen only when it wakes up at the DTIM interval or listen interval or Beacon interval.

The flow chart below shows the Power Save Mode 1 flow:

- The RS9116W gets connected to the Access Point and goes into Power Save Mode 1.
- If the host has data, it will send it to the RS9116W. The RS9116W can receive data from the host at any point in time.
- The RS9116W will check if it is DTIM time, if not, it will continue to be in Power Save Mode 1. If it is DTIM time, the RS9116W waits for the beacon. Then it will check if there are packets to send or receive from the remote peer or terminal, if there are packets to send/ receive the RS9116W does the appropriate action and repeats this cycle until there are no more packets to send or receive.
- Later, it will go back to Power Save.

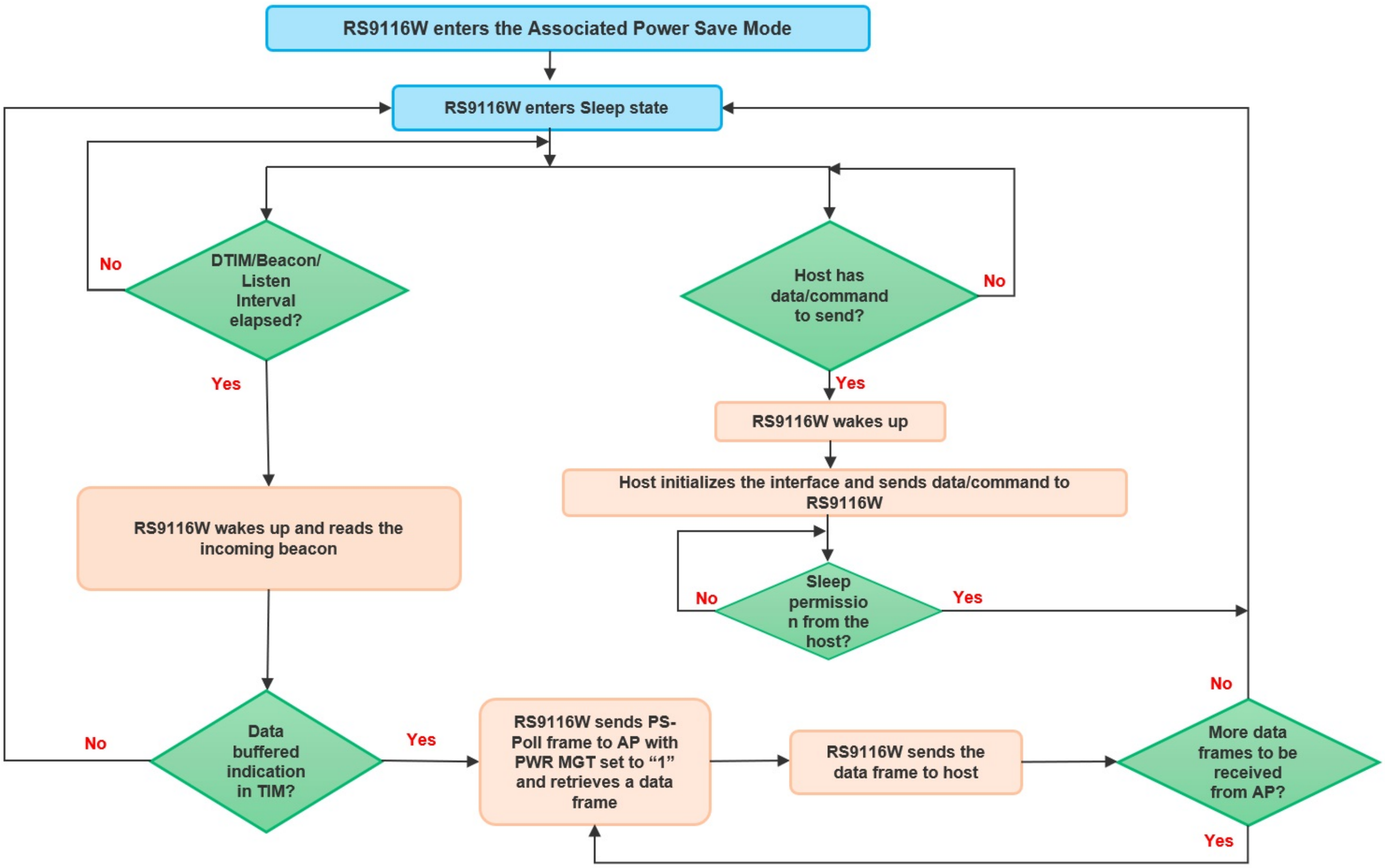


Figure 5.1. Power Save Mode 1 Flowchart

5.3.3 Power Save Mode 2

Power Save Mode 2 is a GPIO-based Standby Associated Power Save Mode (Connected Sleep). Once the RS9116W is configured in Power Save Mode 2, it is woken up by the host using a GPIO-based handshake and it will periodically wake up during its sleep-wake-up cycle based upon the DTIM interval to send or receive data to/from remote-peer.

In this mode, when the host wants to send data to the RS9116W, it gives a wakeup request to the RS9116W by asserting ULP_GPIO_5 in the case of LP mode or UULP_GPIO_2 in case of ULP mode (which make the RS9116W wake up from Power Save). After wakeup, if the RS9116W is ready for data transfer, it sends a wakeup indication to the host by asserting the UULP_GPIO_3 or UULP_GPIO_0.

The host is required to wait until the RS9116W gives the wakeup indication, before sending any data to the RS9116W. After completion of data transfer, the host can give sleep permission to the RS9116W by de-asserting ULP_GPIO_5 in the case of LP mode or UULP_GPIO_2 in the case of ULP mode. After recognizing sleep permission from the host, the RS9116W confirms the host by de-asserting UULP_GPIO_3 or UULP_GPIO_0 and goes back to its sleep-wakeup cycle.

The RS9116W can send a received packet or response to the host at any point in time. No handshake is required on the receive path.

The flowchart below shows the Power Save Mode 2 flow:

- The RS9116W goes to Power Save Mode 2, it will wake up periodically every listen interval or Beacon Interval or DTIM interval, to send or receive data from the remote peer.
- If there is no packet from the host, the RS9116W will continue to stay in the sleep state.
- If there is a packet from the host, the host sends a wake-up request to the RS9116W by asserting the UULP_GPIO_2 in ULP mode and ULP_GPIO_5 in LP mode.
- The RS9116W responds by asserting the UULP_GPIO_3 or UULP_GPIO_0, thereby giving a wake-up indication to the host.
- The host checks if there is a packet from the RS9116W to be received, if it does, the RS9116W will send the packets to the host until there are no more packets to be sent to the host.
- If there is a packet from the host, it sends the packet to the RS9116W.
- If there are no packets to be sent from the host, it gives sleep permission to the RS9116W by de-asserting the UULP_GPIO_2 in ULP mode and ULP_GPIO_5 in LP mode.
- The RS9116W confirms that it’s going to sleep by de-asserting the UULP_GPIO_3 or UULP_GPIO_0 and goes into power save.

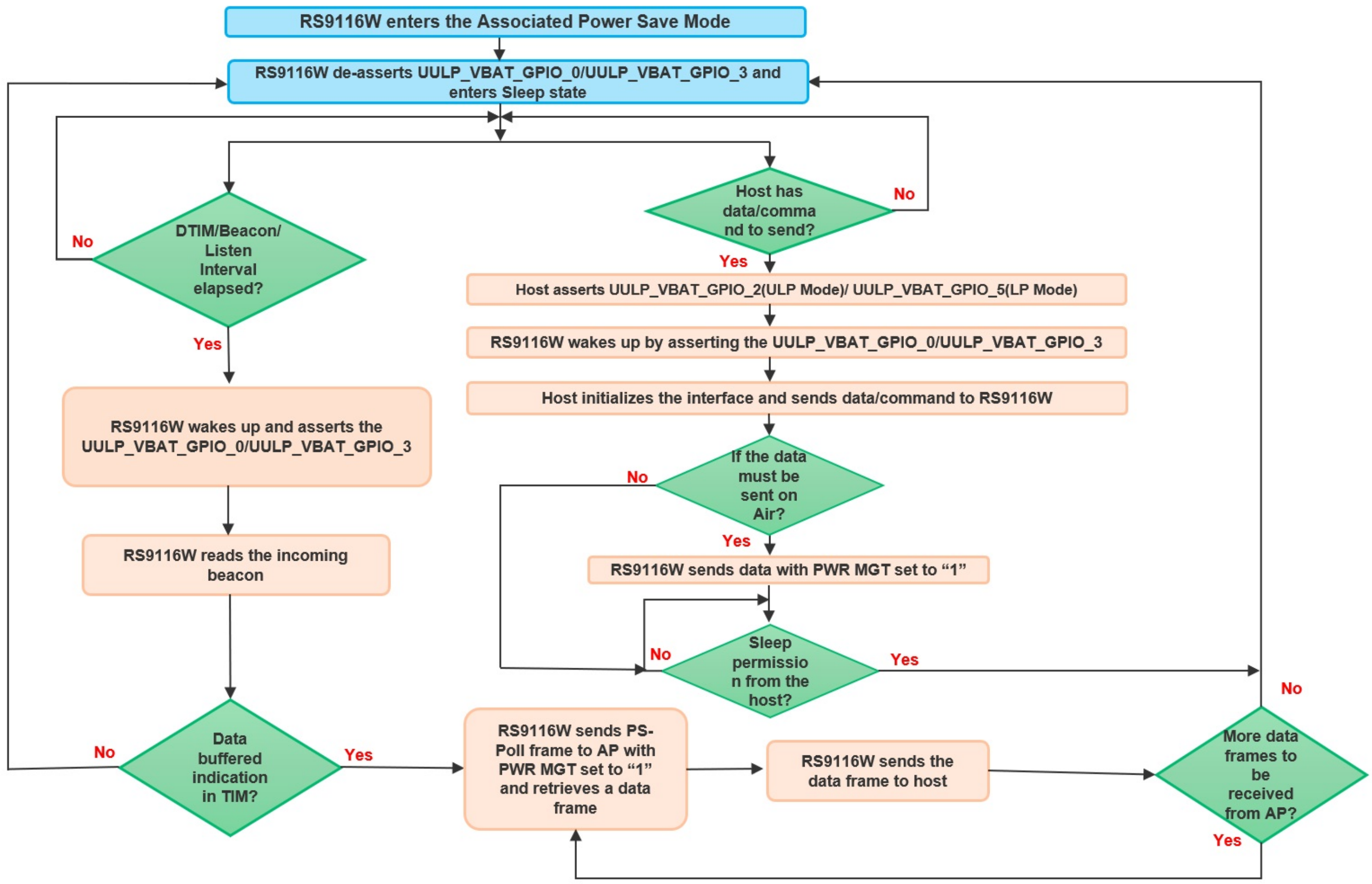


Figure 5.2. Power Save Mode 2 Flowchart

5.3.4 Power Save Mode 3

Power Save Mode 3 is a Message-based Deep Sleep Mode with/without RAM Retention (Unconnected sleep). This mode is significant when the RS9116W is connected to AP. The RS9116W wakes up periodically every DTIM and gives a wakeup message ("WKP") to the host. The RS9116W cannot be woken up asynchronously. Every time the RS9116W intends to go to sleep it sends a sleep request message ("SLP") to the host and expects the host to send the ACK message. Host either sends an acknowledgment message ("ACK") or any other pending message. But once ACK is sent, the host should not send any other message until the next wakeup message from the RS9116W is received. The RS9116W will not go into the complete power save state if ACK is not received from the host for the sent sleep (SLP) message. The RS9116W can send a received packet or response to the host at any point in time. No handshake is required on the receive path.

The flow chart below shows the Power Save Mode 3 flow:

- RS9116W wakes up every DTIM in this case.
- If it is not DTIM time, it continues to stay in power save state.
- When it is DTIM time, the RS9116W sends a wake-up (WKP) message to the host.
- If there is a packet from the host, it sends the packet to the RS9116W.
- When the host has no more packets to send, the RS9116W sends a sleep (SLP) message to the host and waits for an acknowledgment (ACK).
- When ACK is received from the host, the RS9116W goes to power save.

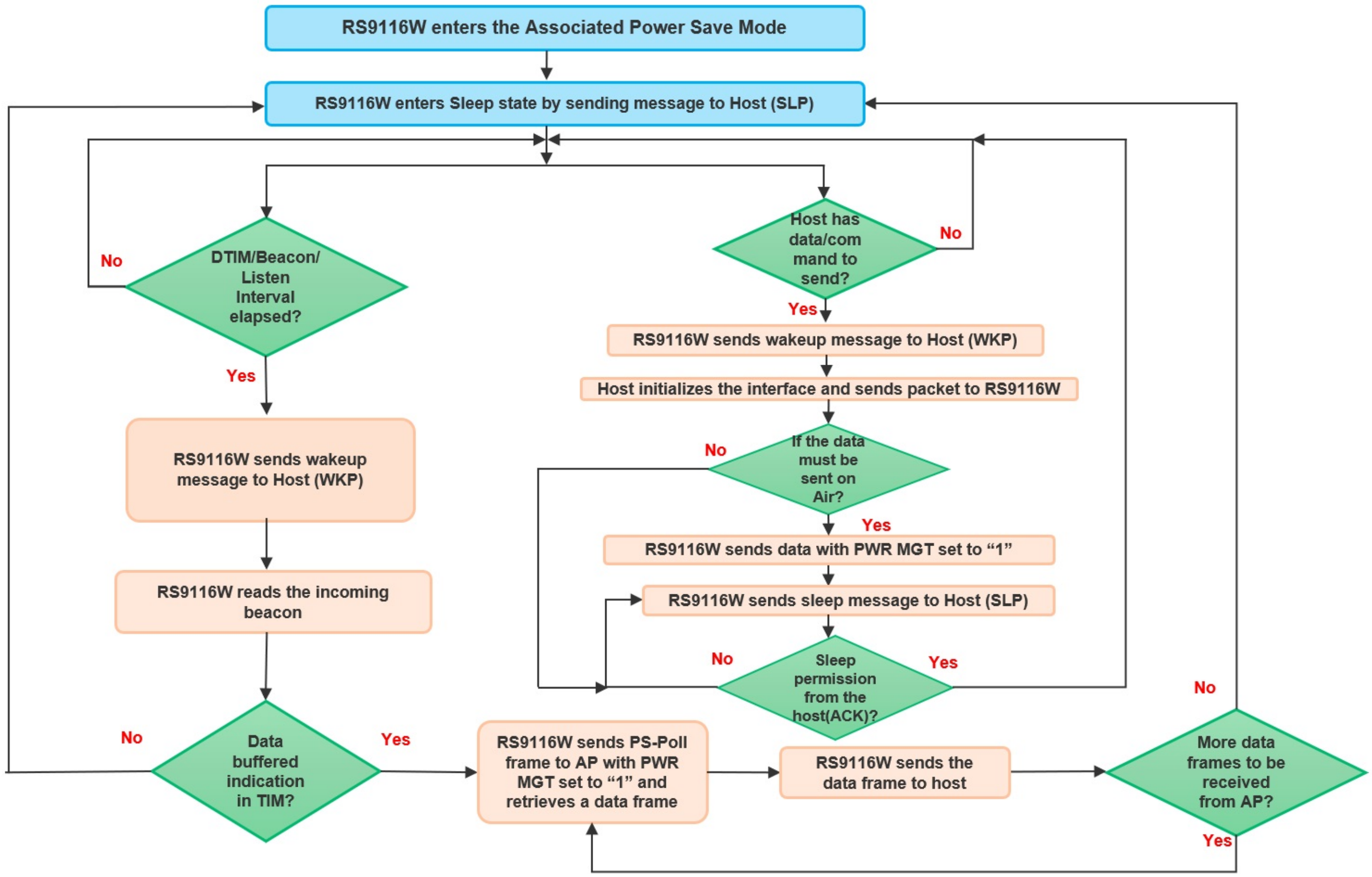


Figure 5.3. Power Save Mode 3 Flowchart

5.4 Unconnected Sleep Modes

5.4.1 Power Save Mode 8

Power Save Mode 8 is GPIO-based Deep Sleep mode with/without RAM Retention (Unconnected sleep). This command must be issued after the Init command. In Power Save Mode 8 both RF and SoC of RS9116-WiSeConnect are in complete power save mode. This mode is significant when the RS9116W is not connected with any AP. In ULP mode, feature_bit_map[4] must be set in opermode command. In the case of LP mode (In power save command when ulp_mode_enable is '0') host can wake up the RS9116W from power save by asserting ULP_GPIO_5. In the case of ULP mode (In power save command when ulp_mode_enable is '1' or '2') host can wake up the RS9116W from power save by asserting UULP_GPIO_2.

When ulp_mode_enable is set to 0 (LP mode) or 1 (ULP mode -With RAM Retention), once the RS9116W gets a wake-up request from the host, it continues to be in the wake-up state until it gets Power Save Mode 8 command from the host.

When ulp_mode_enable is set to 2 (ULP mode - Without RAM Retention), after the RS9116W wakes up from sleep, the host needs to start giving commands from the beginning(opermode) as the RS9116W state is not retained.

The flow chart below explains the Power Save Mode 8 flow:

- The RS9116W goes to Power Save Mode 8 and if there is no command from the host, the RS9116W continues to be in power save.
- If there is a command from the host, the host sends a wake-up request to the RS9116W by asserting the UULP_GPIO_2 in ULP mode and ULP_GPIO_5 in LP mode.
- The RS9116W responds by asserting the UULP_GPIO_3 or UULP_GPIO_0, thereby giving a wake-up indication to the host.
- The host sends commands to the RS9116W.
- The RS9116W gives sleep confirmation to the host by de-asserting the UULP_GPIO_2 in ULP mode and ULP_GPIO_5 in LP mode.
- The RS9116W confirms that it’s going to sleep by de-asserting the UULP_GPIO_3 or UULP_GPIO_0 and goes into power save.

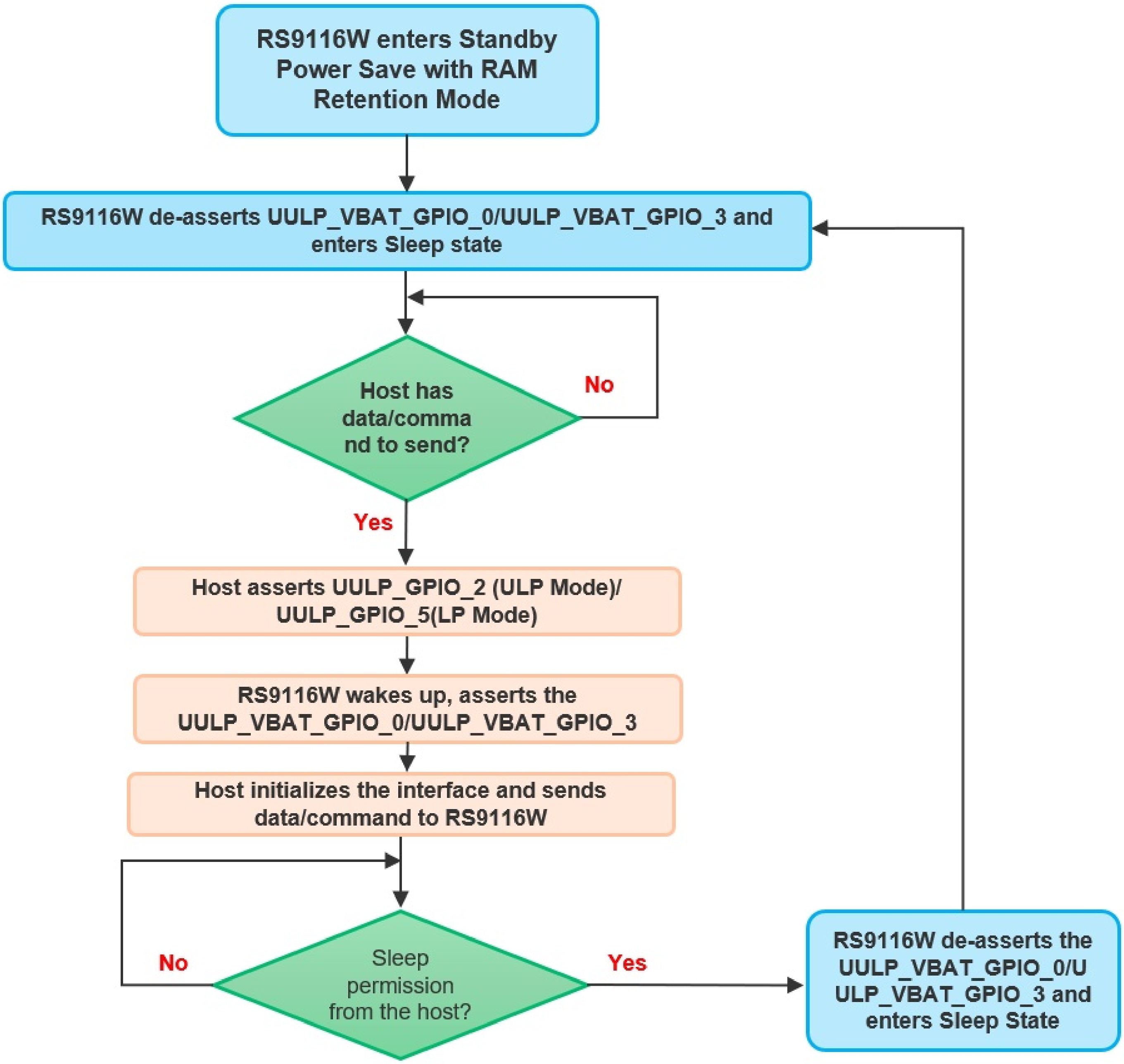


Figure 5.4. Power Save Mode 8 Flowchart

5.4.2 Power Save Mode 9

Power Save Mode 9 is a Message-based Deep Sleep Mode with/without RAM Retention (Unconnected sleep). In Power Save Mode 9 both the radio and SoC of RS9116- WiSeConnect are in complete power save mode. This mode is significant when the RS9116W is not connected with any AP.

Once the Power Save Mode 9 command is given to the RS9116W, it goes to sleep immediately and wakes up after sleep duration configured by the host using the sleep timer command. If the host does not set any time, then the RS9116W wakes up every 3 seconds by default (You can configure this time by using the sleep timer command). Upon wakeup, the RS9116W sends a wakeup message (WKP) to the host and expects the host to give an acknowledgment (ACK) before it goes into the next sleep cycle. Host either sends ACK or any other messages but once ACK is sent to the RS9116W, no other packet should be sent before receiving the next wakeup message from the RS9116W.

When ulp_mode_enable is set to 2 (Without RAM Retention), after waking up from sleep, the RS9116W sends the wakeup from sleep message (WKP FRM SLEEP) to the host. After receiving the message, the host needs to start giving commands from the beginning (opermode) as the RS9116W state is not retained.

The flow chart below explains the Power Save Mode 9 flow:

- RS9116W wakes up with a default configured time of three seconds and sends a wake-up (WKP) message to the host.
- If there is a command from the host, it sends the command to the RS9116W.
- If the host does not have any more commands, the RS9116W sends a sleep (SLP) message to the host
- And the RS9116W waits for an acknowledgment (ACK) from the host.
- Once the RS9116W receives ACK from the host, it will go back to Power Save Mode 9

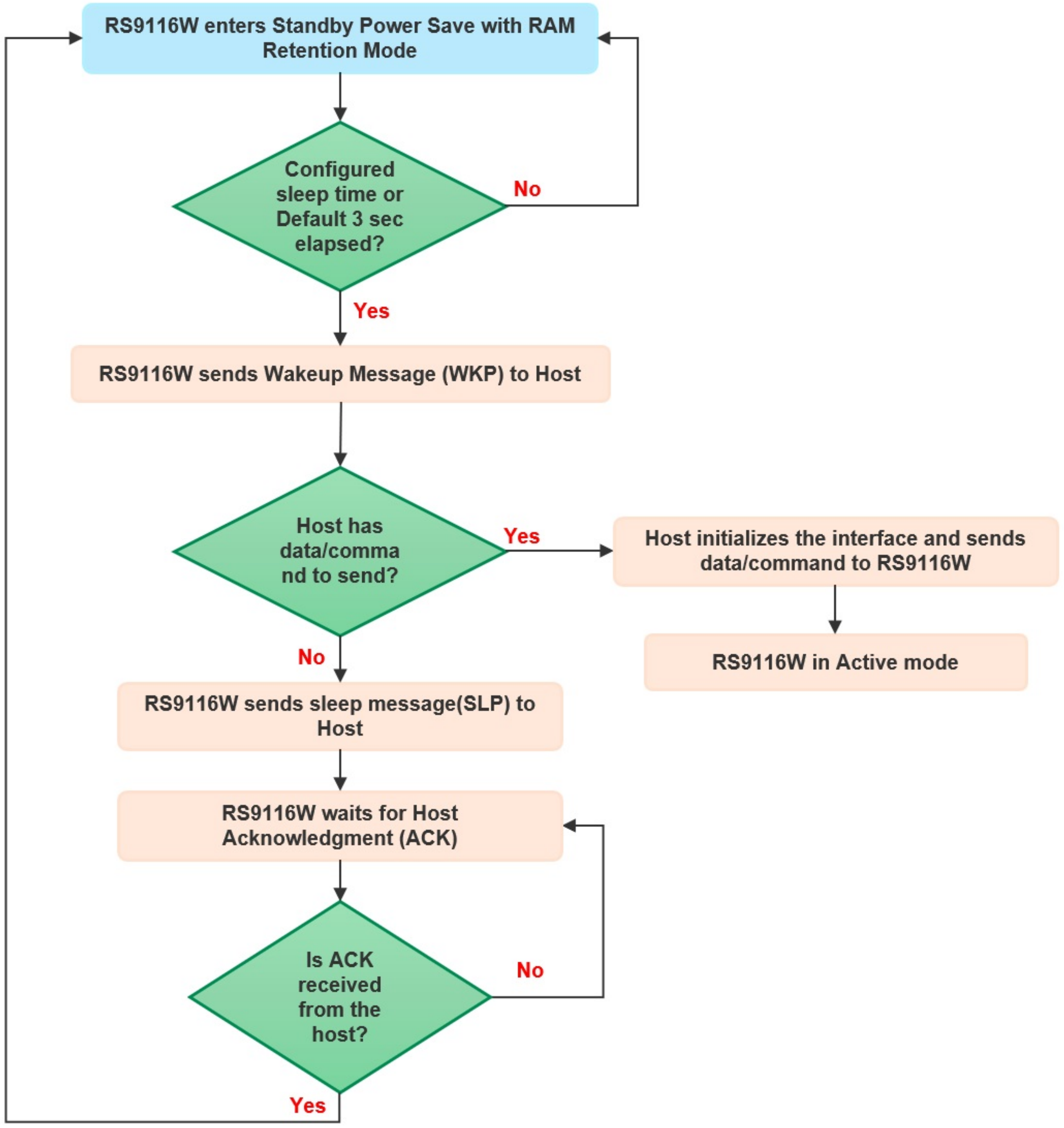


Figure 5.5. Power Save Mode 9 with RAM Retention Flowchart

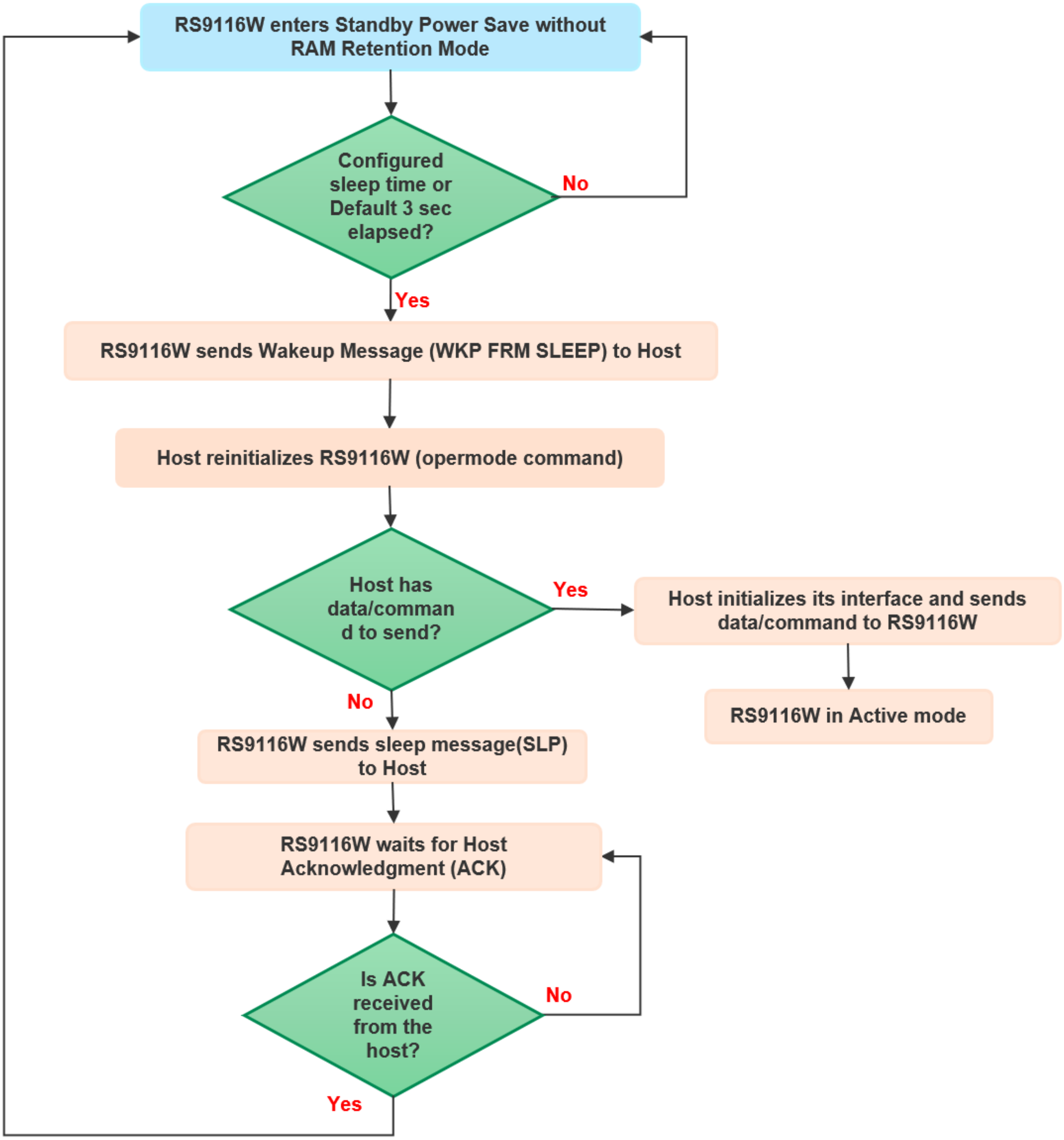


Figure 5.6. Power Save Mode 9 without RAM Retention Flowchart

6. Power Save Procedures

RS9116W supports the following Power save Procedures and based on the application use case, one of the below is chosen.

- MAX Power Save Procedure
- FAST Power Save Procedure

Before getting into the Power save Procedures, let us understand the Power Save polling procedure. The following are the three building blocks of power save polling (PSP):

Wakeup Procedure

There are two reasons for a station (STA) to wake up:

- Transmit pending data from host
- Retrieve buffered data from an Access point (AP)

For receiving data, STA monitors the Partial Virtual Bitmap (PVB) bit in the DTIM beacon frames sent out by the connected AP.

- Once the STA decides to transit from sleep to active mode, it notifies AP by sending a frame with Power Management (PWR MGT) bit set to Active.
- The STA remains active to enable, the AP to send the buffered frames.

Sleep Procedure

- An STA (in active mode) must complete an STA-initiated frame exchange sequence, with the PWR MGT bit set to Sleep – this is a successful transition to sleep.
- Following this operation, the AP buffers all the frames destined to this STA.

PS-Poll Procedure

- Instead of waiting for the AP to transmit the buffered frames, an STA (in sleep mode) can solicit an immediate delivery from its access point by using a PS-poll frame.
- Upon receiving this PS-poll, the AP can immediately send buffered frames (immediate data response); or it can send an acknowledgment message and response with a buffered data frame later (delayed data response).

6.1 Maximum PSP

- STA wakes up, listens based on the interval in Association request, and checks if data is buffered in AP for it.
- If data is buffered, it sends a PS poll and retrieves data, and this process continues until the last data frame from AP.
- In this case, the STA will be in power save most of the time.
- If you want to save more power and there is no requirement of multicast data, you will select listen interval- based power save.
- RS9116W retrievals data only in wakeup time due to which it saves more power but produces lower throughputs. In case of receiving bulk amount of data, sending a PS-Poll frame to retrieve each frame affects the throughput and induces a considerable amount of delay.
- You can opt for this procedure when you want the device to be in maximum power save mode, and data retrieval happens less frequently.

6.2 Fast PSP

- When both throughput and power save are important for an application this power save profile is used.
- Whenever there is data buffered on AP, RS9116W sends a Null/data frame with the PWR MGT bit set to 0 (STA will stay up for the monitor interval time set using the Monitor_interval in power save command) and receives the data buffered from AP.
- After successful retrieval of data, we go back to sleep mode.
- This helps in avoiding latency (total round-trip time it takes a data frame to travel), which is seen in MAX PSP.
- The Fast PSP saves less power and produces better throughput when compared to the Max PSP. If both throughput and power save are important for your application, use Fast PSP.

6.2.1 Enhanced Max-PSP

- This is recommended PSP type. This essentially a MAX PSP mode but switches to Fast PSP mode if AP does not deliver data within 20 ms for PS-Poll.
- To enable this mode please follow the below steps.
    1. Enable BIT(26) in CONFIG_FEATURE_BITMAP
    2. Set psp_type to 1 (Fast PSP)

7. Power Save Mode Selection

You can choose the appropriate power save mode based on your application. For example, the following table has few cases mentioned when a particular power save mode must be chosen.

Power Save Mode	GPIO's Available for a Handshake	RS9116W Device is Always-on for Incoming Messages from the Host	RS9116W Wakes up and Receives Data from the Host in a Connected Sleep State	RS9116W Wakes up and Receives Commands from the Host in an Unconnected Sleep State
1	Not Applicable	YES	NO	NO
8	YES	NO	NO	YES
9	NO	NO	NO	YES
2	YES	NO	YES	NO
3	NO	NO	YES	NO

When RS9116W is in a connected or unconnected sleep state, it can be woken up by the host using either GPIO or Message-based handshake.

8. Current Consumption

8.1 Setup Diagram

8.1.1 AT Setup

Hardware Requirements

- Windows PC with UART interface
- Silicon Labs RS9116 Wi-Fi Evaluation Kit - RS9116X-SB-EVK1/RS9116X-DB-EVK1
- Wi-Fi Access point
- 3 wire connectors for GPIO pins (female-to-female connectors)
- Power analyzer

Software Requirements

- Docklight
- Keysight Software

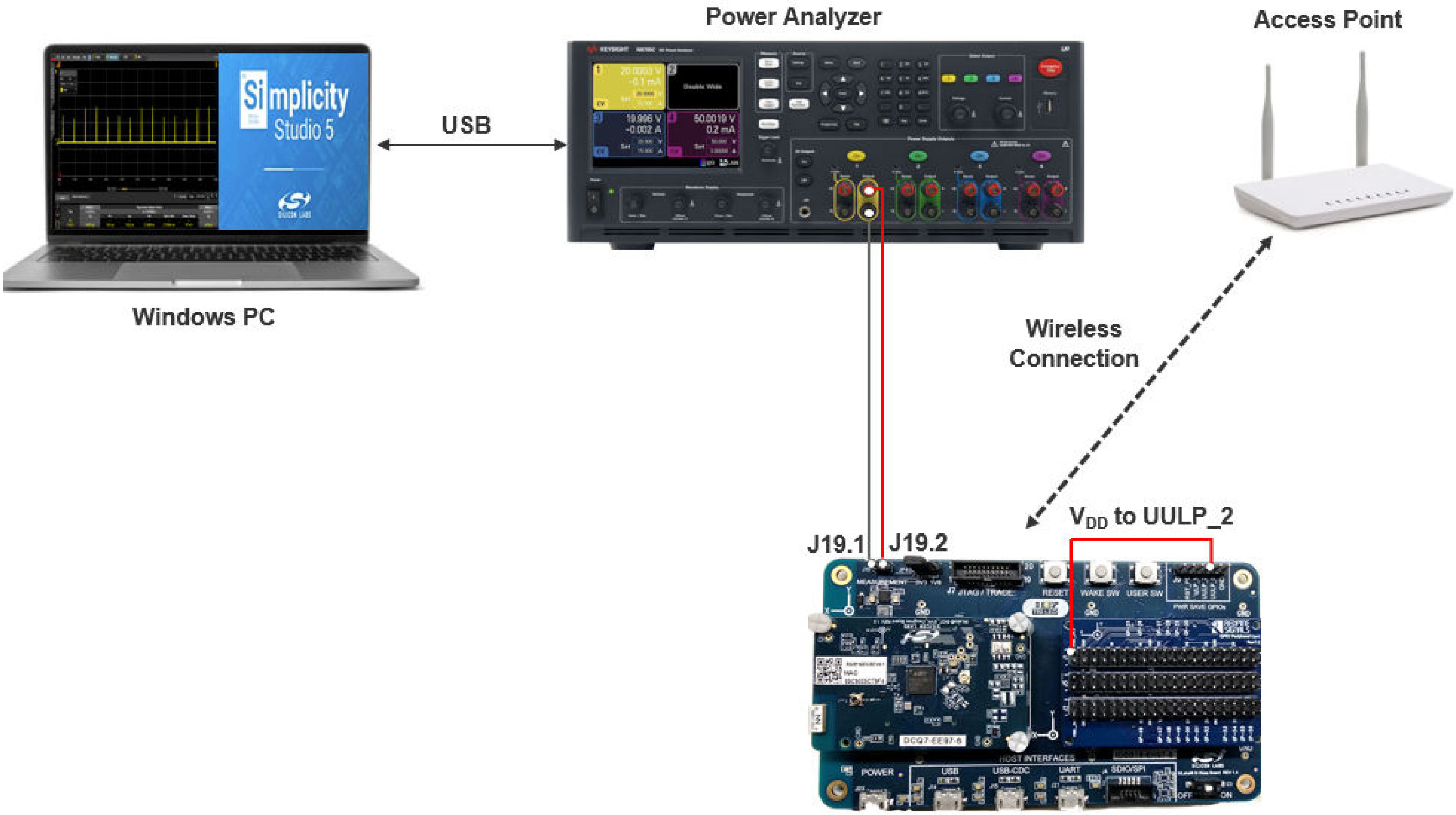
GPIO-based Setup

In AT mode, no Host MCU is required. An EVK GPIO Peripheral Card is connected to the RS9116 EVK, as illustrated below. Whenever data/command is to be sent to RS9116W:

- The UULP_2 pin (UULP_VBAT_GPIO_2 of RS9116W) on J9 should be connected to GPIO Header J1.2 to wake the RS9116W from sleep in case of Ultra-Low Power mode.
- The ULP_5 on J9 should be connected to GPIO Header J1.2 to wake the RS9116W from sleep in case of Low Power mode.

When wake-up is not needed, UULP_2/ULP_5 can be connected to GND (GPIO Header-J1.4) or left idle.

The setup diagram for measuring the RS9116W's current consumption in AT mode using GPIO-based connections is measured by a Keysight N6705B Power Analyzer, as shown below.



8.1.2 SAPI Setup

Hardware Requirements

- EFR32xG21 Starter Kit with Wireless Gecko (Baseboard: BRD4002A, Radio board: BRD4180a or BRD4180b)
- Windows PC with Host interface (UART / SPI/ SDIO).
- Silicon Labs RS9116 Wi-Fi Evaluation Kit
- Wi-Fi Access point
- 5 wire connectors for GPIO pins (female-to-female connectors)
- Power analyzer

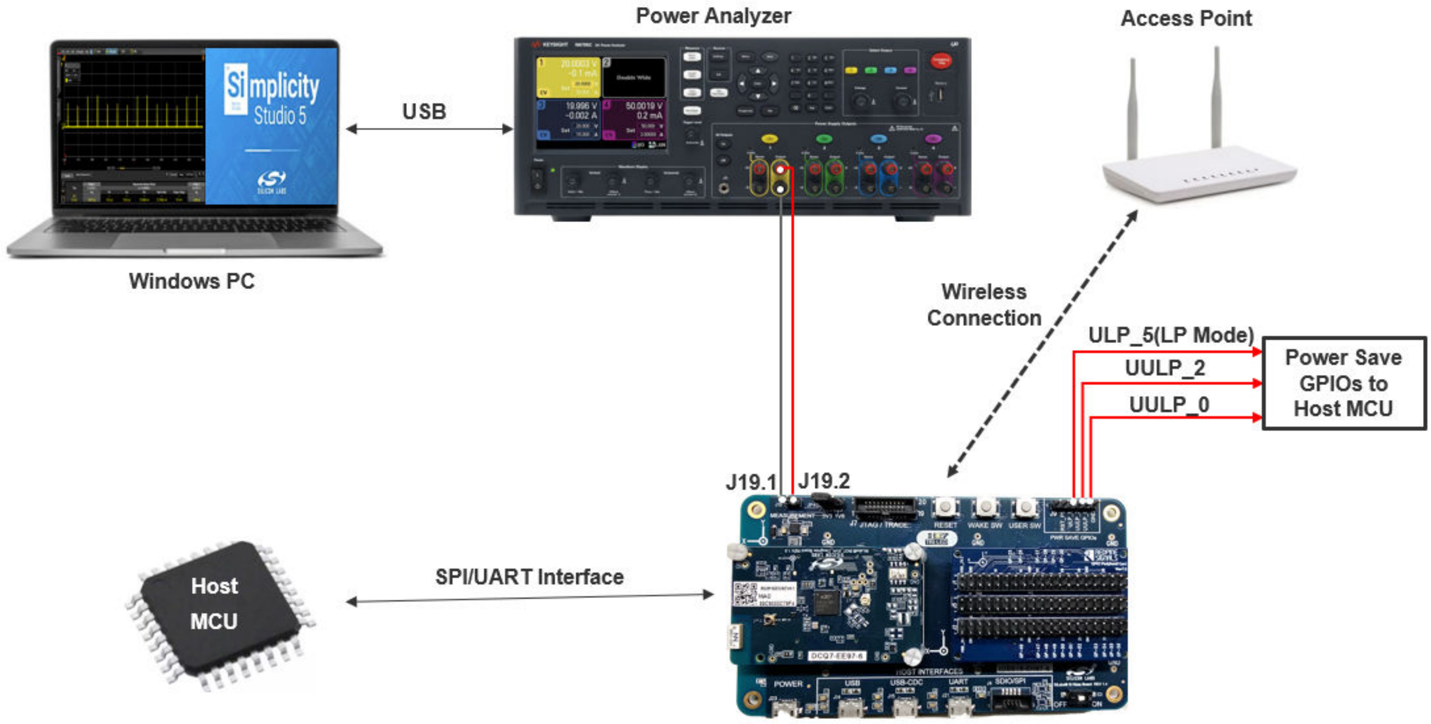
Software Requirements

- WiSeConnect SDK
- Embedded Development Environment
  - For Silicon Labs EFx32, use the latest version of Simplicity Studio.
  - For STM32, use Keil
- Docklight

GPIO-based Setup

In SAPI mode, the UULP_2 GPIO on the RS9116 EVK connects to the Host MCU to control wake-up and sleep requests from the host in Ultra Low Power (ULP) mode. Similarly, the ULP_5 GPIO connects to the Host MCU to manage wake-up and sleep requests in Low Power (LP) mode. Additionally, UULP_0 is connected to the Host MCU to get wake and sleep indications from the RS9116 EVK.

The setup diagram for measuring the RS9116W's current consumption in AT mode using GPIO-based is measured by a Keysight N6705B Power Analyzer, as shown below.



8.2 Network Processor Initialization

Below is the time-taken and current consumption during RS9116W initialization before accepting the first command.

The following configurations are used for the below test cases with RS9116W QMS (1.4 silicon version):
- **EVK:** RS9116X-SB-EVK1 (RS9116W QMS Rev1.4)
- **Band:** 2.4 GHz Channel - 1
- **Test Environment:** Ideal Environment (Shield Room)

RS9116W State	Current (Average Current)	Time Taken	Current (Average Current)	Time Taken
	3.3 V - UART@115200 (Bootloader Bypass)		3.3 V - SPI	
Device Power up to Card-Ready (During Firmware Loading)	18.9 mA	710 ms	18.9 mA	707 ms
After firmware gets loaded (Before Opermode command)	6.4 mA	NA	6.32 mA	NA

8.3 RS9116W Wi-Fi Connection and IP Configuration

This section provides details on current and energy consumption for various Wi-Fi initialization states, such as scanning and joining:

The following configurations are used for the below test cases with RS9116W QMS (1.4 silicon version):
- **EVK:** RS9116X-SB-EVK1 (RS9116W QMS Rev1.4)
- **Band:** 2.4GHz Channel - 1
- **Test Environment:** Ideal Environment (Shield Room)

State	Current (mA)	Time (ms)	Voltage (V)
Scan	40.6	1030	3.3
Join	54.2	137.5	3.3
Scan (Quick Scan)	56.59	6.02	3.3
WPA2 Association (Quick Join)	68.78	20.07	3.3
DHCP	41.51	10.03	3.3

- **Quick Scan:** When quick scan is enabled, the RS9116W scans for the specified Access Point's SSID in the special channel. As soon as it finds the AP, the RS9116W sends scan response to host.
- **Software Configuration:** Enable the BIT(0) in scan_feature_bitmap. This feature aids in reducing the time taken for scanning for APs.
- **Quick Join:** When the Quick Join feature is enabled, the RS9116W bypasses the unicast probe request step. Instead of sending a unicast probe request and waiting for a response from the Access Point, the RS9116W directly sends authentication and association requests to the AP, streamlining the joining process.
- **Software Configuration:** Enable BIT(2) in the join_feature_bit_map to activate this feature. Quick Join helps in reducing the time required to connect to Access Points.

8.4 TCP Data Transfer

This section provides details on current and energy consumption for various TCP throughput states and SSL/TLS connections:

- Collecting the maximum amount of data possible and uploading it at one shot at full throughput.
- Downloading data at maximum throughput allowed by the link.
- Full throughput upload power consumption (maximum) occurs at the lowest on-air data rates.
- Full throughput download power consumption (maximum) occurs at the highest on-air data rates.

The following configurations are used for the below test cases with RS9116W QMS (1.4 silicon version):

- **EVK:** RS9116X-SB-EVK1 (RS9116W QMS Rev1.4)
- **Test Environment:** Ideal Environment (Shield Room)

State	Current (mA)	Time (ms)	Voltage (V)	Charge (uC)	Energy (mJ)
Throughput- TCP-TX (100Kbps)	1.67	1000	3.3	1670	5.51
Throughput- TCP-RX (100Kbps)	1.93	1000	3.3	1930	6.37
TCP & SSL/TLS Connection (Cipher - ECDHE- RSA-AES128-SHA)	9.23	426.00	3.3	3931.98	12.97553
TCP & SSL/TLS Connection (Cipher - AES128- SHA)	7.43	347.00	3.3	2578.21	8.50809
TCP & SSL/TLS Connection (Cipher - DHE-RSA- AES128-SHA)	9.16	1012.00	3.3	9269.92	30.59074

8.5 RS9116W TX/RX Active State

RS9116W Wi-Fi subsystem (Radio and Network processor) can be in the following atomic states:

- **TX_ACTIVE:** Active Transmission at a given on-air data rate e.g., 6 Mbps,54 Mbps, MCS-7, etc., and a given output power level (e.g., 8 dBm, 18 dBm)
- **RX_ACTIVE:** Active Reception at a given on-air data rate

The following configurations are used for the below test cases with RS9116W QMS (1.4 silicon version):

- **EVK:** RS9116X-SB-EVK1 (RS9116W QMS Rev1.4) and RS9116X-DB-EVK1 (RS9116W-DB00-CC1 Rev1.4)

  Band: 2.4 GHz Channel - 1

  Duration: 5 min Average

  Test Environment: Ideal Environment (Shield Room)

The table below shows the RS9116W in RX Active and TX Active states:

Parameter	Data Rate	Description	Value (QMS)	Value (CC1)	Units
RX Active	1 Mbps	LP Chain	22	NA	mA
RX Active	6 Mbps	HP Chain	36	115	mA
RX Active	72 Mbps	HP Chain	37	132	mA
TX Active	6 Mbps	Tx Power = Maximum (18 dBm)	208	339	mA
		Tx Power = 8 dBm	113	285	mA
TX Active	54 Mbps	Tx Power = Maximum (18 dBm)	149	274	mA
		Tx Power = 8 dBm	87	237	mA

These values were measured using the WLAN RF Test example. For more information, please refer to Examples.

8.6 Standby Associated Mode: Connected Sleep

Wireless LAN (WLAN) protocol permits a client device to sleep during its idle periods. However, the client is expected to wake up periodically at intervals specified as listen interval or Beacon Interval or DTIM interval in the power save command mentioned in the application. During the period the client is asleep, the Access Point holds packets destined for them. These buffered packets are delivered to the clients when the clients wake up at DTIM, observes an indicator of traffic pending for it in the AP's beacon, and send a power-save poll message to request the AP to retrieve the buffered packets.

Below are the following configurations are used for the below test cases with RS9116W QMS (1.4 silicon version):

- **EVK:** RS9116X-SB-EVK1 (RS9116W QMS Rev1.4)
- **DTIM Period of Access Point :** 10
- **Listen Interval :** 1000 Milli seconds (1 sec)
- **Power Save Mode:** Power Save Mode 2
- **Test Environment:** Ideal Environment (Shield Room)

8.6.1 Modes of Configuring Power Save in Standby Associated Mode

8.6.1.1 AT Mode: Command Sequence

Power Save Mode	Command Sequence	Average Current Consumption	RS9116W Wake interval
2	at+rsi_opermode=0,20,2147483652,2147483648,12058624,0,2147483648,0,0,1 at+rsi_feat_frame=0,1,0,0,1,49 at+rsi_band=0 at+rsi_init at+rsi_scan=0,SILABS_AP at+rsi_psk=1,password at+rsi_join=SILABS_AP,0,2,2 at+rsi_ipconf=1 at+rsi_pwmode=2,1,0,0	56.8 µA	1000 milliseconds
Note: Once the module enters power save mode, to disable power save, connect the **UULP_2 pin** (UULP_VBAT_GPIO_2 of RS9116W) on **J9** to the **GPIO Header-J1.2** to wake the RS9116W from sleep. Then, execute the command: `at+rsi_pwmode=0`			

8.6.1.2 SAPI Mode: API Sequence

Power Save Mode	Command Sequence	wlan_config.h configurations	Average Current Consumption	RS9116W Wake interval

2	rsi_driver_init(global_buf, GLOBAL_BUFF_LEN)
	
	rsi_device_init(LOAD_NWP_FW)
	
	rsi_wireless_init(0, 0)
	
	rsi_send_feature_frame()
	
	rsi_wlan_connect((int8_t *)SSID, SECURITY_TYPE, PSK)
	
	rsi_config_ipaddress(RSI_IP_VERSION_4, dhcp_mode, 0, 0, 0, ip_buff, sizeof(ip_buff), 0)
	
	rsi_wlan_filter_broadcast(5000, 1, 1)
	
	rsi_wlan_power_save_profile(RSI_SLEEP_MODE_2, PSP_TYPE) or rsi_wlan_power_save_with_listen_interval(RSI_SLEEP_MODE_2, PSP_TYPE, listen_interval)
		Configure RSI_HAND_SHAKE_TYPE as GPIO_BASED
		
		For listen interval based power save: configure as,
		
		`#define RSI_JOIN_FEAT_BIT_MAP`
		
		`RSI_JOIN_FEAT_PS_CMD_LISTEN_INTERVAL_VALID`
		
		`#define RSI_LISTEN_INTERVAL 1000`
			56 uA	1000 milliseconds

The following figure shows the power trace for Power save Standby Associated:

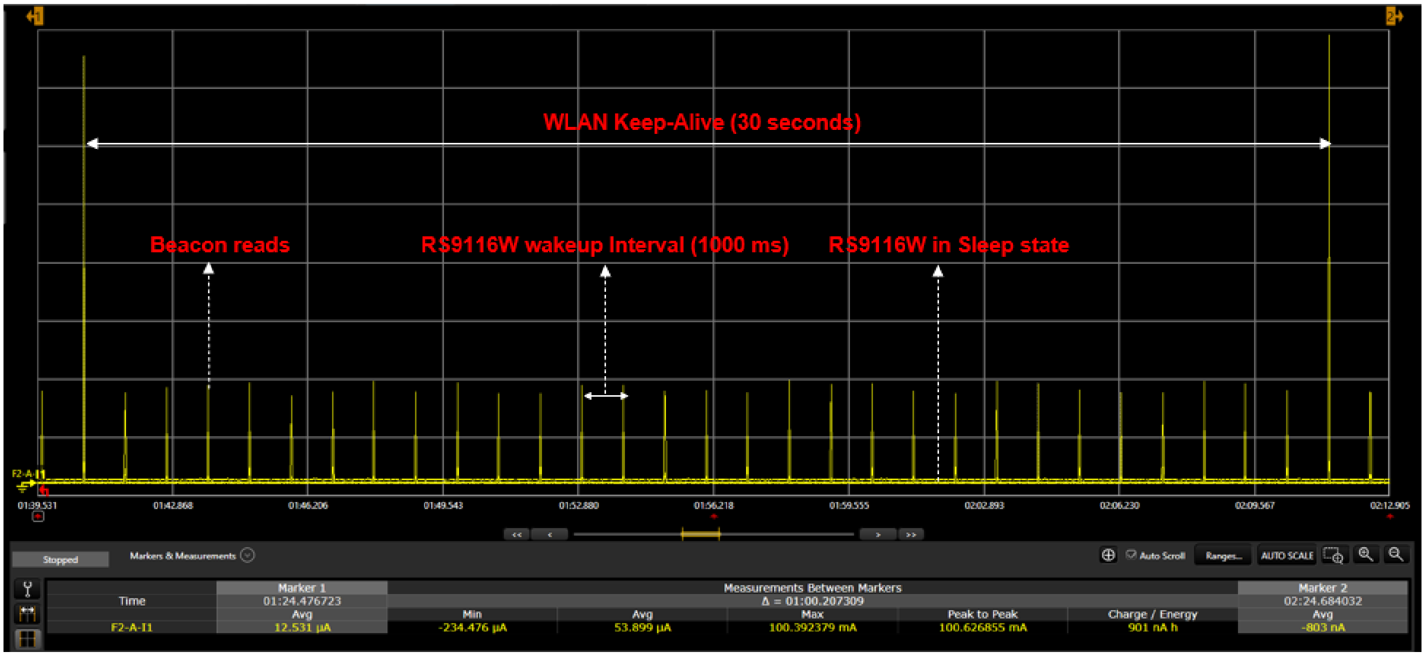


Figure 8.1. Associated Standby Power Trace

Note:

1. The **rsi_wlan_power_save_profile** API is used to set the power save profile in WLAN mode without specifying a listen interval whereas **rsi_wlan_power_save_with_listen_interval** API is used to set the power save profile in WLAN mode with a listen interval-based wakeup.
2. We highly recommend to use rsi_wlan_filter_broadcast API when using connected sleep mode. This API is used to ignore threshold levels of the broadcast packet when station is in power save and is used to achieve low current consumption in standby associated mode. This is blocking API. For more information refer to the link.

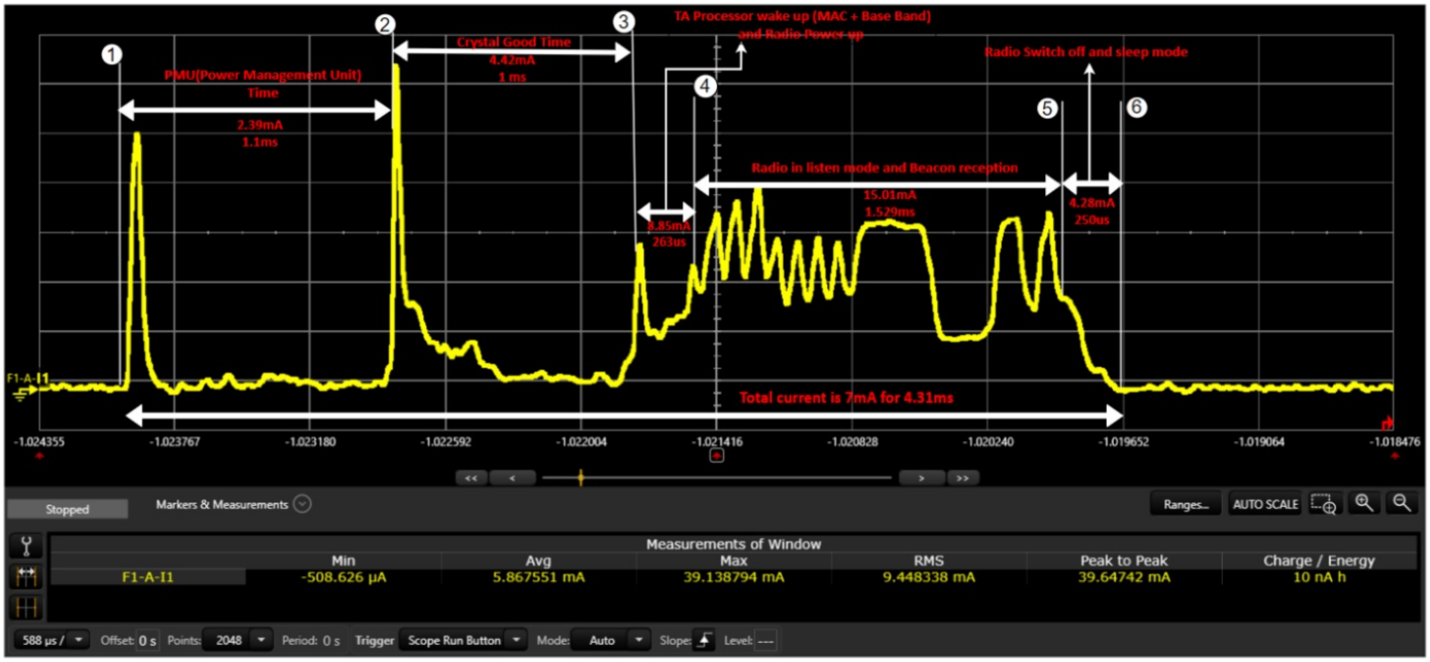
For more information, refer to the API documentation.

For more information refer to the Power Save Standby Associated Example.

8.6.2 Beacon Current Profile

The following table depicts the typical current consumption and time taken at different stages while receiving a single beacon in an Ideal Environment (Shield Room).

S. No	State	Time Taken	Average Current Consumption
1	Power Management Unit (PMU) Time	1.1 ms	2.39 mA
2	Crystal good Time	1 ms	4.42 mA
3	TA Processor Wakeup (MAC + Base Band) and Radio Power up	263 µs	8.85 mA
4	Radio in listen mode and beacon reception	1.529 ms	15.01 mA
5	Radio Switch off and sleep mode	250 µs	4.28 mA



Note: These values were measured using the Power Save Standby Associated example. For more information, click here.

8.6.3 WLAN Connection to Disconnection and back to Connected Sleep: API Sequence

Below is the API sequence for setting the RS9116W back to Power Save mode 2 (Connected Sleep Mode) whenever WLAN disconnection occurs.

Power Save Mode	Command Sequence	wlan_config.h Configurations

2	void rsi_join_fail_handler(uint16_t status, uint8_t *buffer, const uint32_t length) `//register a callback // Disconnect event handler should make connection_state to NETWORKDISCONNECT`
	
	rsi_driver_init(global_buf, GLOBAL_BUFF_LEN)
	
	rsi_device_init(LOAD_NWP_FW)
	
	rsi_wireless_init(0, 0)
	
	rsi_send_feature_frame()
	
	rsi_wlan_register_callbacks(RSI_JOIN_FAIL_CB, rsi_join_fail_handler)//Intialize a callback
	
	rsi_wlan_scan((int8_t *)SSID,0,NULL,0);
	
	rsi_wlan_connect((int8_t *)SSID, SECURITY_TYPE, PSK)
	
	rsi_config_ipaddress(RSI_IP_VERSION_4, dhcp_mode, 0, 0, 0, ip_buff, sizeof(ip_buff), 0)
	
	rsi_wlan_filter_broadcast(5000, 1, 1)
	
	rsi_wlan_power_save_profile(RSI_SLEEP_MODE_2, PSP_TYPE) or rsi_wlan_power_save_with_listen_interval(RSI_SLEEP_MODE_2, PSP_TYPE, listen_interval)
	
	if(state=NETWORKDISCONNECT){if wlan_disconnection happens, go to rsi_wlan_scan and execute from there}
		Configure RSI_HAND_SHAKE_TYPE as GPIO_BASED

Note: Once after Wlan disconnection, the RS9116W will not be going back to associated Power save. User should configure the RS9116W to associated power save if they want.		

8.6.4 CO-EX (WLAN + BLE) Low Power Applications

This API sequence provides the recommended steps to optimize current consumption in Standby Associated Mode for Coexistence (Coex) applications using both WLAN and BLE. By adhering to this sequence, developers can effectively manage and minimize power usage while ensuring seamless connectivity across both wireless interfaces.

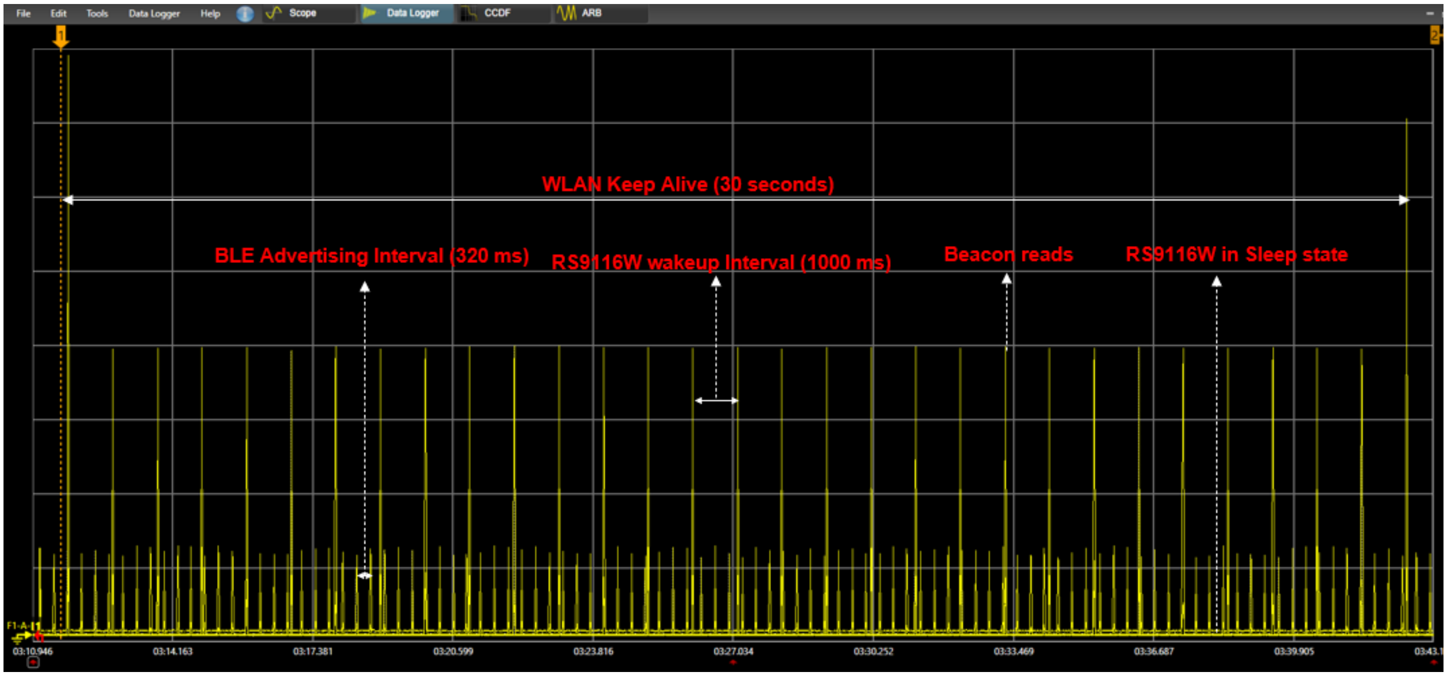
8.6.4.1 AT Command Sequence

Power Save Mode	Command Sequence	RS9116W Wake interval
2	at+rsi_opermode=851968,20,2147483652,2147483648,2159542272,3221225472,2147483648,1075773440,0,1 at+rsi_feat_frame=0,1,0,0,1,49 at+rsi_band=0 at+rsi_init at+rsi_scan=0,SSID at+rsi_psk=4,PSK_LENGTH,PSK at+rsi_join=SILABS_AP,0,2,2,2,1000 at+rsi_ipconf=1 at+rsibt_setlocalname=DEVICE_NAME_LENGTH,DEVICE_NAME at+rsibt_getlocalbdaddr? at+rsibt_setadvertisedata=10,2,1,6,C,9,52,53,39,31,31,36,57,5F,42,4C,45 at+rsibt_advertise=1,128,0,0,0,2048,2048,0,7 at+rsi_pwmode=2,1,1	WLAN - 1 second Bluetooth advertising interval - 1.28 seconds

8.6.4.2 SAPI mode: API sequence

Power Save Mode	Command Sequence	wlan_config.h configurations	ble_config.h configurations	RS9116W Wake interval

2	rsi_driver_init(global_buf, GLOBAL_BUFF_LEN)
	
	rsi_device_init(LOAD_NWP_FW)
	
	rsi_wireless_init(0, 13)
	
	rsi_send_feature_frame()
	
	rsi_wlan_connect((int8_t *)SSID, SECURITY_TYPE, PSK)
	
	rsi_config_ipaddress(RSI_IP_VERSION_4, dhcp_mode, 0, 0, 0, ip_buff, sizeof(ip_buff), 0)
	
	rsi_bt_set_local_name((uint8_t *)RSI_BLE_LOCAL_NAME)
	
	rsi_bt_get_local_name(&rsi_app_resp_get_local_name)
	
	rsi_ble_start_advertising()
	
	rsi_wlan_filter_broadcast(5000, 1, 1)
	
	rsi_wlan_power_save_profile(RSI_SLEEP_MODE_2, PSP_TYPE)
	
	rsi_bt_power_save_profile(RSI_SLEEP_MODE_2, PSP_TYPE)
		Configure RSI_HAND_SHAKE_TYPE as GPIO_BASED
		
		For listen interval based power save: configure as,
		
		`#define RSI_JOIN_FEAT_BIT_MAP RSI_JOIN_FEAT_PS_CMD_LISTEN_INTERVAL_VALID`
		
		`#define RSI_LISTEN_INTERVAL 1000`
			Configure RSI_BLE_ADV_INT_MIN as 0x800 and RSI_BLE_ADV_INT_MAX as 0x800	WLAN - 1 second
				
				Bluetooth advertising interval - 1.28 seconds
				
				Note:
				
				0x800 in hexadecimal is equal to 2048 in decimal. Time calculation: Time = N * 0.625 ms Where N is the value set (2048 in this case)=> 2048 * 0.625 ms = 1280 ms => 1.28 seconds



Note:

1. The above API sequence is with respect to the WLAN BLE Power save example. For more information, please refer to the link.
2. If the BLE is enabled in the opermode, there is no provision to disable BLE to make WLAN standalone.

8.6.5 Current Consumption vs. Listen Interval

The current consumption values for various power save modes with different listen intervals are mentioned in this section.

- **EVK:** RS9116X-SB-EVK1 (RS9116W QMS Rev1.4) and RS9116X-DB-EVK1 (RS9116W-DB00-CC1 Rev1.4)
- **Band:** 2.4GHz Channel - 1
- **Band:** 5GHz, Channel -36
- **Duration:** 5 min Average
- **Test Environment:** Ideal Environment (Shield Room)
- **Access Point:** Asus RT-AC1200G

Power Save Mode	Listen Interval (milliseconds)	Current Consumption (QMS)	Current Consumption (CC1)		Units
		2.4 GHz, Channel 1	2.4 GHz, Channel 1	5 GHz, Channel 36	
1	100	7.3	10.3	10.6	mA
	200	7.1	10.1	10.6	mA
	300	7	10	10.6	mA
	400	7	10	10.5	mA
	500	7	9.9	10.5	mA
	600	6.9	9.9	10.5	mA
	700	6.9	9.9	10.5	mA
	800	6.9	9.9	10.5	mA
	900	6.9	9.9	10.5	mA
	1000	6.9	9.9	10.5	mA
2	100	334.1	527.6	1.4	uA
	200	179.2	300.4	763.4	uA
	300	129.1	227.3	565.1	uA
	400	102.1	175.5	431.4	uA
	500	84.8	154.1	341	uA
	600	76.3	133.6	307.9	uA
	700	67.6	124.9	258	uA
	800	63.3	113.6	223.6	uA
	900	60.8	102.5	199.1	uA
	1000	56.3	101.3	184.1	uA

You can configure the **listen interval** value using the **listen_interval** parameter in the power save command. The maximum listen interval in the power save command should be less than the listen interval configured in join command.

If you want to change the listen_interval dynamically, you need to disable power save and enable power save again with a new **listen_interval**.

Note: The above power numbers were measured using the Power Save Standby Associated example. For more information, please refer to the link.

For additional documentation and release information, visit Silicon Labs RS9116 documentation.

8.7 Deep Sleep Mode: Unconnected Sleep

Below are the following configurations are used for the below test cases with RS9116W QMS (1.4 silicon version):

- **EVK:** RS9116X-SB-EVK1 (RS9116W QMS Rev1.4)
- **Band:** 2.4GHz Channel - 1
- **Duration:** 5 min Average
- **Access Point used:** TP-Link Archer AX10
- **Power Save Modes:** Power Save Mode 8 and 10
- **Test Environment:** Ideal Environment (Shield Room)

The following sections show the current consumption values with a TP-Link Archer AX10 access point in an idle environment.

8.7.1 With RAM Retention

8.7.1.1 AT Command Sequence

Below is the AT mode command sequence for Power Save Mode 8 along with the current consumption values.

Power Save Mode	Command Sequence	Average Current Consumption
8 (With RAM Retention)	at+rsi_opermode=0,20,2147484806,2147483648,11010048,0,2147614728,0,0,1 at+rsi_band=0 at+rsi_init at+rsi_pwmode=8,1	11.6 µA

8.7.1.2 SAPI Mode: API sequence

Below is the SAPI command sequence for Power Save Mode 8 with AP connection along with the current consumption values.

Power Save Mode	Command Sequence	wlan_config.h configurations	Average Current Consumption
8	rsi_driver_init(global_buf, GLOBAL_BUFF_LEN) rsi_device_init(LOAD_NWP_FW) rsi_wireless_init(0, 0) rsi_wlan_radio_init() rsi_wlan_power_save_profile(RSI_SLEEP_MODE_8, PSP_TYPE) rsi_wlan_power_save_profile(RSI_ACTIVE, PSP_TYPE) rsi_wlan_connect((int8_t *)SSID, SECURITY_TYPE, PSK) rsi_config_ipaddress(RSI_IP_VERSION_4, dhcp_mode, 0, 0, 0, ip_buff, sizeof(ip_buff), 0)	Configure RSI_HAND_SHAKE_TYPE as GPIO_BASED	11.6 µA

8.7.2 Without RAM Retention

8.7.2.1 AT Command Sequence

Below is the AT mode command sequence for Power Save mode 8 along with the current consumption values.

Power Save Mode	Command Sequence	Average Current Consumption
8	at+rsi_oper-mode=0,20,2147484806,2147483648,11010048,0,2147614728,0,0,1 at+rsi_band=0 at+rsi_init at+rsi_pwmode=8,2	3.6 µA
Note: Once the module enters power save mode, connect the **UULP_2 pin** (UULP_VBAT_GPIO_2 of RS9116W) on **J9** to the **GPIO Header-J1.2** to wake the RS9116W from sleep and gives sleep message (WKP FRM SLEEP). Then, the host needs to start giving commands from the beginning (opermode) as the module's state is not retained.		

8.7.2.2 SAPI mode: API Sequence

Below is the AT mode command sequence for Power Save mode 10 along with the current consumption values.

Power Save Mode	Command Sequence	wlan_config.h configurations	Average Current Consump-tion

10 (Without RAM Reten-tion)	rsi_driver_init(global_buf, GLOBAL_BUFF_LEN)
	rsi_device_init(LOAD_NWP_FW)
	rsi_wireless_init(0, 0)
	rsi_wlan_radio_init()
	rsi_wlan_power_save_profile(RSI_SLEEP_MODE_10, PSP_TYPE)
	rsi_req_wakeup() //to disable the power save
	rsi_wireless_init(0, 0)
	rsi_wlan_connect((int8_t *)SSID, SECURITY_TYPE, PSK)
	rsi_config_ipaddress(RSI_IP_VERSION_4, dhcp_mode, 0, 0, 0, ip_buff, sizeof(ip_buff), 0)
		Configure RSI_HAND_SHAKE_TYPE as GPIO_BASED</p>	3.6 µA

The figure below shows the power trace for Deep Sleep mode with RAM: retention:

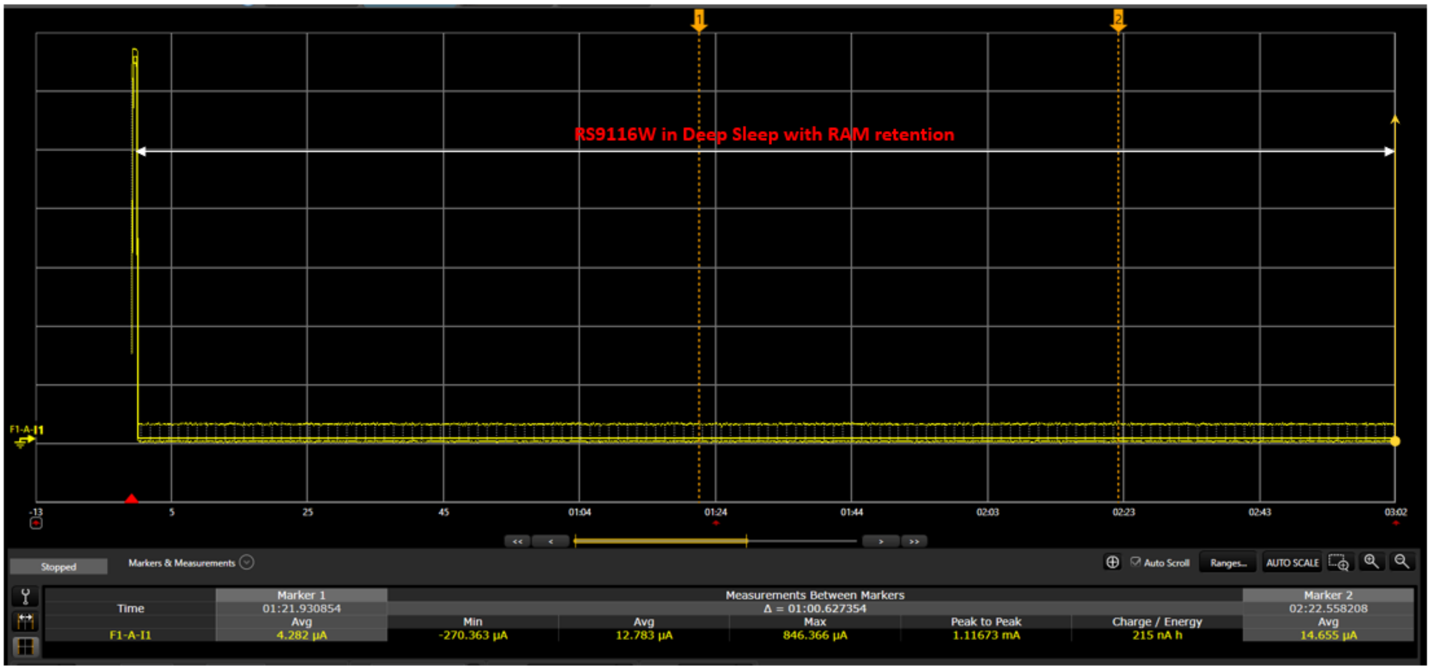


Figure 8.2. Deep Sleep Power Capture

For more information refer to the link.

Note:

1. For AT mode, it is mandatory to call at+rsi_init command before setting the device in unconnected sleep mode.
2. For SAPI mode, it is mandatory to call rsi_wlan_radio_init API before setting the device in unconnected sleep mode.

8.8 Unconnected Power Save Mode Current Consumption

The current consumption values for unconnected power save modes are mentioned in this section.

- **EVK:** RS9116X-SB-EVK1 (RS9116W QMS Rev1.4) and RS9116X-DB-EVK1 (RS9116W-DB00-CC1 Rev1.4)
- **Band:** 2.4 GHz Channel -1
- **Band:** 5 GHz, Channel-36
- **Duration:** 5 min Average
- **Test Environment:** Ideal Environment (Shield Room)
- **Access Point:** Asus RT-AC1200G

Power Save Mode	Current Consumption (QMS)	Current Consumption **(CC1)**		Units
	2.4 GHz, Channel 1	2.4 GHz, Channel 1	5 GHz, Channel 36	
8 (With RAM Retention)	11.9	9.7	9.5	uA
8 (Without RAM Retention)	3.6	9.7	9.5	uA
9 (With RAM Retention)	12.1	9.6	11.5	uA
9 (Without RAM Retention)	3.6	9.6	11.5	uA

9. Channel Utilization vs Current Consumption across Different APs

Following are the current consumption values with various Access points versus channel utilization.

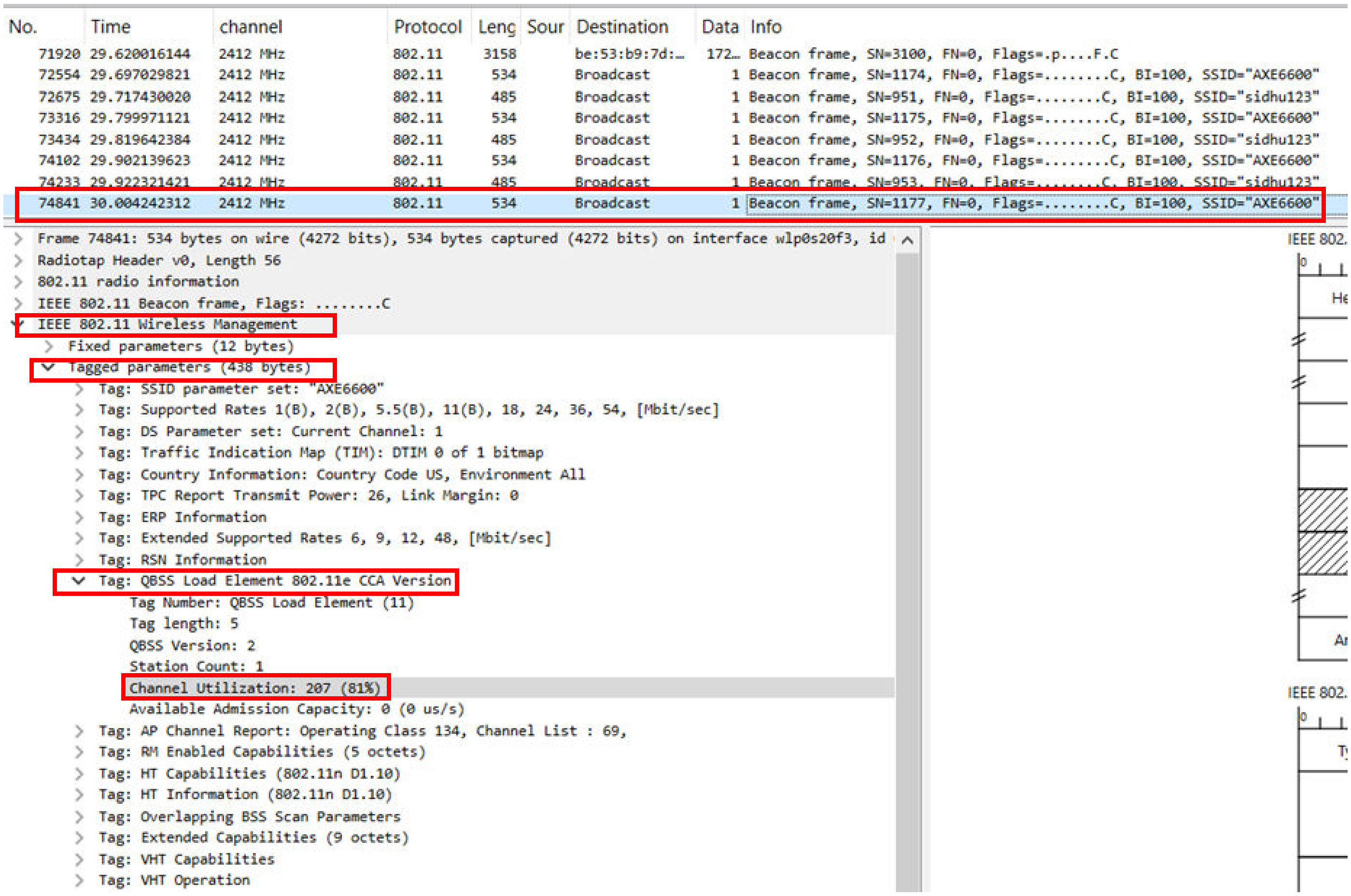
- **EVK:** RS9116X-SB-EVK1 (RS9116W QMS Rev1.4)
- **Band:** 2.4 GHz, Channel - 1
- **Duration:** 5 min Average
- **Power Save Mode:** Standby associated mode with listen interval as 1 second
- **Application:** TCP client sends application message from TCP Client to the server for every 55 seconds.
- **Test Environment:** Ideal Environment (Shield Room)

Access Point	Model	Access Point Firmware Version	Current Consumption for 0% Channel Utilization	Current Consumption for 81% Channel Utilization
TP-Link	AC1750	1.0.4 build 20180425	74.8 µA	138.9 µA
ASUS	RTAC68U	3.0.0.4.384	68.4 µA	155.8 µA
ASUS	AC1200	3.0.0.4.382_70222	71.7 µA	128.5 µA
D-Link	855	1.12	75.6 µA	172.5 µA
Tenda	AC1200 ac5	15.03.06.50_multi	74.3 µA	134.8 µA
D-link	882	1.01	72.5 µA	130.3 µA
MI	R3L		71.1 µA	141.8 µA
Netgear	R8000	1.0.3.46_1.1.32	88 µA	166.4 µA
Tenda	AC1200 ac6	v15.03.06.50_multi	74.05 µA	132.5 µA
D-link	810L	1.01	63.8 µA	149 µA
TPLink	AC750	1.20	61.7 µA	113.55 µA
TPLink	Archer AX10	0.9.1 1.4 v003a.0	71.2 µA	174.65 µA
TPLink	Archer A7 (WiFi 5 enabled)	1.1.0 build 20201120 rel. 50399(4555)	118.65 µA	225.15 µA
ASUS	AXE6600 (Wi-Fi-6 enabled)	3.0.0.4.386_49873	83.3 µA	117.3 µA

Note:

1. The standby-associated current consumption may vary depending on the behavior of the Access Point (AP).

2. To assess the channel utilization of a network, examine the beacon frame of an AP in a Wireshark capture, as illustrated below.



3. If you want to use any tool to check Channel Utilization, refer to the Inssider application.
4. The RS9116W supports Wi-Fi 4. For Wi-Fi 6 (IEEE 802.11ax), the current consumption may be higher.

10. Factors Affecting Power Consumption

- **Type of Access Point (Make/Model)**: Some Access Points (APs) may send unicast frames to the RS9116W, which then processes these frames and sends response frames. This interaction can lead to increased current consumption on the RS9116W due to the additional data traffic.
- **Distance Between the Access Point and RS9116W**: The physical distance can impact the power needed for communication, as the device may need to boost its transmission power to maintain a reliable connection.
- **Channel Congestion**: High levels of congestion on the channel can lead to increased retransmissions and higher power usage, as the device expends more energy trying to send and receive data amidst competing traffic.
- **Roundtrip Time to a Remote TCP Server**: The time it takes for data to travel to a remote TCP server and back, as well as the TCP/TLS connection mechanism, can affect power consumption. Longer roundtrip times may lead to the device remaining active for extended periods, increasing energy use.
- **Type of Power-Save Mechanism Used**: Different power-saving techniques can result in varying levels of power consumption. Ineffective power-saving strategies can cause the device to remain in higher power states longer than necessary.
- **Calibration of the Board**: If the board is not calibrated correctly, it can affect performance and power consumption. Miscalibrated components may require additional power to function optimally.
- **Bulk Data Transfers**:
  - An example of bulk data transfer to IoT devices is an Over-The-Air (OTA) firmware upgrade. These operations typically involve large amounts of data being transmitted, requiring the device to stay awake and actively process incoming data for an extended period, which significantly increases power consumption.
  - Conversely, an example of data transfer from an IoT device is the sensor data captured by the device. Frequent or large volumes of data transmission can keep the device in an active state longer, consuming more power during these operations.
  - Additionally, bulk transfers can lead to increased overhead in terms of connection management, error checking, and acknowledgment processes, all of which contribute to higher current consumption.

11. Recommendations for Low Power and Interoperability Optimization

Depending on the specific application use case, the RS9116W can be configured into various power save modes and follow different power save procedures as outlined in this document. We recommend the following to be considered for battery-operated devices to optimize battery life.

1. Choose a Listen Interval value that aligns with the application's requirements while allowing the RS9116W to maximize sleep time. This interval determines how often the RS9116W will wake up to check for any data from the Server/AP.

   Note: When power saving based on the Listen Interval is enabled, the DTIM period will not affect sleep duration.

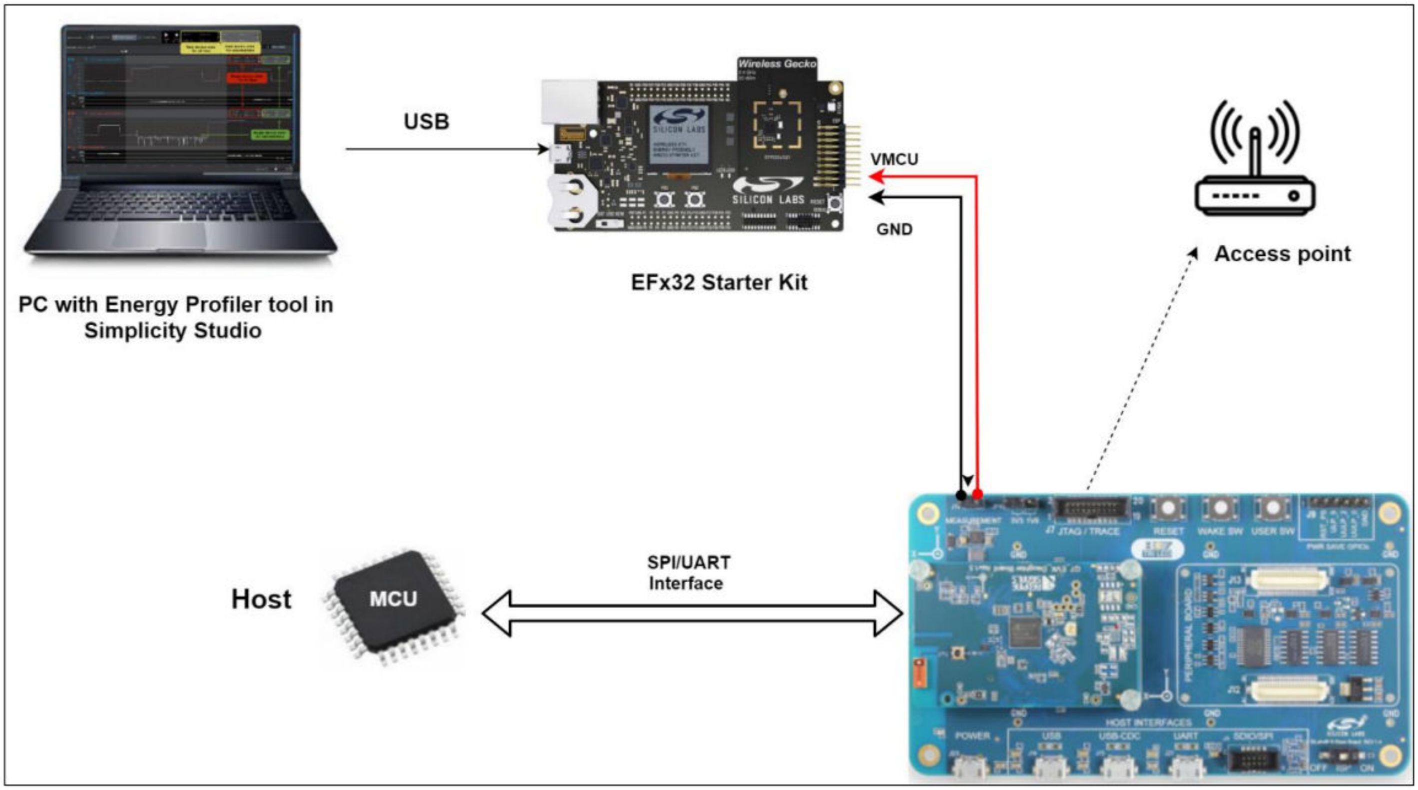
2. A 1000-milliseconds listen interval is highly recommended for optimal low-power consumption.
3. After a rejoin or disconnection, disable power save mode, then perform Scan, Join, and Ipconfig operations. Once complete, re-enable power save mode. This approach can help reduce connection time with certain Access Points.
4. It is recommended to use connected power save mode only after executing the Ipconfig command.
5. If a rejoin attempt fails, instead of scanning immediately for the desired AP, it is recommended to try scanning the network a few times. If still not found, put the RS9116W into deep sleep power save mode (e.g., Power Save Mode 8). After some interval, disable power save, and resume scanning for the desired AP.
6. For optimal power management, we highly recommend using the Enhanced Max PSP type in Wi-Fi-alone mode, while using MAX PSP is advised for Co-ex mode.
7. For applications where throughput is not a key requirement, consider disabling higher data rates such as MCS5, MCS6, and MCS7 by enabling `config_feature_bit_map[19]` in the opermode command.
8. Use the WLAN Keep-Alive, TCP Keep-Alive, and MQTT Keep-Alive settings effectively to reduce current consumption.
9. If DNS failures occur with certain APs, enable aggregation by setting `feature_bit_map[2]` in the opermode command. If enabling aggregation does not resolve DNS issues, it is recommended to disable power save mode during DNS queries, then return to power save mode afterward.
10. In applications where the RS9116W remains in low-power save mode for an extended listen interval, if a rejoin or disconnection occurs, the listen interval value can be adjusted without disconnecting from the AP. To enable this, set `join_feature_bitmap[7]` in the Join command parameters. To apply the new listen interval, disable and then re-enable power save mode.
11. If the application does not require broadcast data, power consumption can be further reduced by using the broadcast filter command.

12. Appendix

EFM32 and EFR32 as Power Analyzers

The EFM32 (SLSSTK3701A) and EFR32 (SLWSTK6006A) starter kits can be used to measure current consumption via the Energy Profiler tool in Simplicity Studio 5.

For more information, visit: Silicon Labs Community and UG343



13. Revision History

Revision 1.5

December, 2024

- Updated and modified details regarding Power Save Modes.
- Included current consumption values for the latest Access Points (APs).
- Added sections on the SAPI APIs utilized.
- Expanded on additional factors influencing power consumption.
- Updated the flow charts of all power save modes.

Revision 1.4

August, 2021

- Added details of Power Save Modes.
- Added latest Power Numbers.
- Edited/removed some sections.
- Added low power and interoperability considerations.

Revision 1.3

January, 2021

- Updated the Latest Current Measurements in the Expected Result table.

Revision 1.2

October, 2020

- Updated the band image as per new TTL script info.
- Updated the new name of TTL script in the document.
- Updated the links and document names.

Revision 1.1

May, 2020

Initial release.

Simplicity Studio

One-click access to MCU and wireless tools, documentation, software, source code libraries & more. Available for Windows, Mac and Linux!

IoT Portfolio
www.silabs.com/IoT

SW/HW
www.silabs.com/simplicity

Quality
www.silabs.com/quality

Support & Community
www.silabs.com/community

SILICON LABS

Silicon Laboratories Inc.
400 West Cesar Chavez
Austin, TX 78701
USA

www.silabs.com