Introduction

Out of the BOX when RFD40/90 is connected through USB, RFD40/90 barcode scanner is not available for Datawedge and can be accessed only through integrated RFID SDK. If users need to use RFD40/90 scanner for Datawedge along with integrated RFID applications, then user need to configure the device for new co-existence mode.

RFD40 barcode scanners are available for either SDK or Data wedge. Users cannot create Data wedge profile to access RFD40 scanner and associate to same application which initializes the same scanner interface through SDK.

Whenever a new RFD40 device is connected to terminal through USB. Scanning framework takes priority and enumerates the Scanning interface. During this period SDK will not be able to acquire the scanning interface until scanning framework finishes the enumeration which takes around 10 seconds. During this period scanner init API

```
sdkHandler.setSfCoexistenceModeInit(new SDKHandler.SfCoexistenceResultListener() {
  @Override
  public void onResult(SfCoexistenceResult sfCoexistenceResult) {

    switch (sfCoexistenceResult){
      case SF_COEXISTENCE_FAILURE:
        //Failure case
        break;

      case SF_COEXISTENCE_SUCCESS:
        //Success case
        break;

      case SF_COEXISTENCE_NOT_SUPPORTED:
        //Reader not configured to SSI datawedge mode
        break;
    }
  }
});
```

Note: User needs to call **setSfCoexistenceModeInit** as the first API call immediately after creating sdkhandler instance.

Callback listener will provide the result with success or failure. In case of failure user needs to retry after 10 seconds to set the operation mode again.

Co-existence mode has dependency on the following component version to function

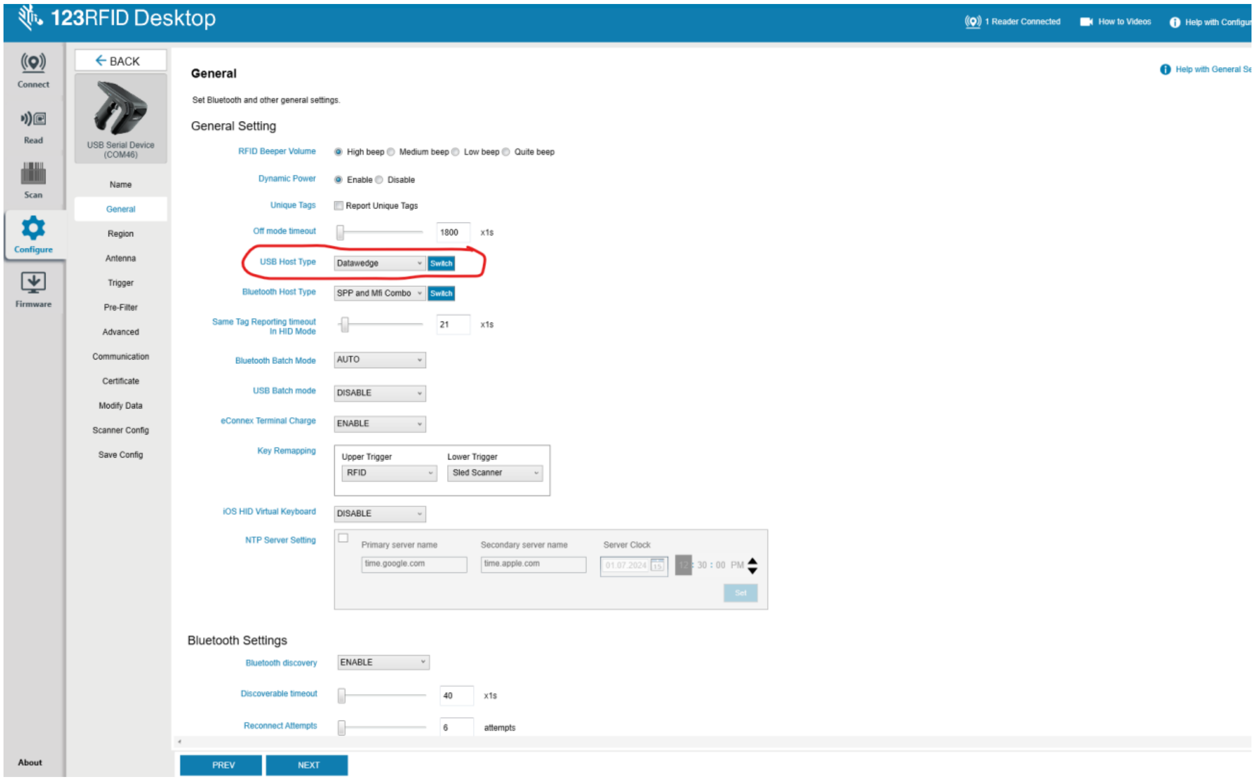
1. Android SDK 92.0.5.236 and above
2. Android Application 92.0.5.236 and above
3. Firmware version
4. Scanning framework 2025 NOV LG release

Note: Please refer to the component version section below to download corresponding components.

Users can also check or confirm if the RFID reader is in the data wedge mode supporting co-existence using the following API.

boolean isReaderInUsbDatawedgeMode = sdkHandler.isSSIDwInterfaceCDCScannerAvailable();

Please note Device needs to be configured for co-existence mode please refer to the details under firmware configuration.



General

Set Bluetooth and other general settings.

General Setting

RFID Beeper Volume: ● High beep ○ Medium beep ○ Low beep ○ Quite beep

Dynamic Power: ● Enable ○ Disable

Unique Tags: □ Report Unique Tags

Off mode timeout: 1800 x1s

USB Host Type: Datawedge Switch

Bluetooth Host Type: SPP and Mfi Combo Switch

Same Tag Reporting timeout In HID Mode: 21 x1s

Bluetooth Batch Mode: AUTO

USB Batch mode: DISABLE

eConnex Terminal Charge: ENABLE

Key Remapping: Upper Trigger RFID; Lower Trigger Sled Scanner

iOS HID Virtual Keyboard: DISABLE

NTP Server Setting: □ Primary server name time.google.com; Secondary server name time.apple.com; Server Clock 01.07.2024 12 : 30 : 00 PM Set

Bluetooth Settings

Bluetooth discovery: ENABLE

Discoverable timeout: 40 x1s

Reconnect Attempts: 6 attempts

Component version

1. Android SDK (https://artifactory-us.zebra.com/artifactory/local-ny-dcs-dev/RFID_UNIFIED_SDK/Q4R2/2.0.5.238/

2. Android APP (https://artifactory-us.zebra.com/artifactory/local-ny-dcs-dev/RFID_UNIFIED_SDK/Q4R2/2.0.5.238/

3. Firmware https://artifactory-us.zebra.com/artifactory/local-ny-dcs-dev/Handheld/RFD40/FIRMWARE/DAILY/SAAFKS00-010-K05E0.DAT

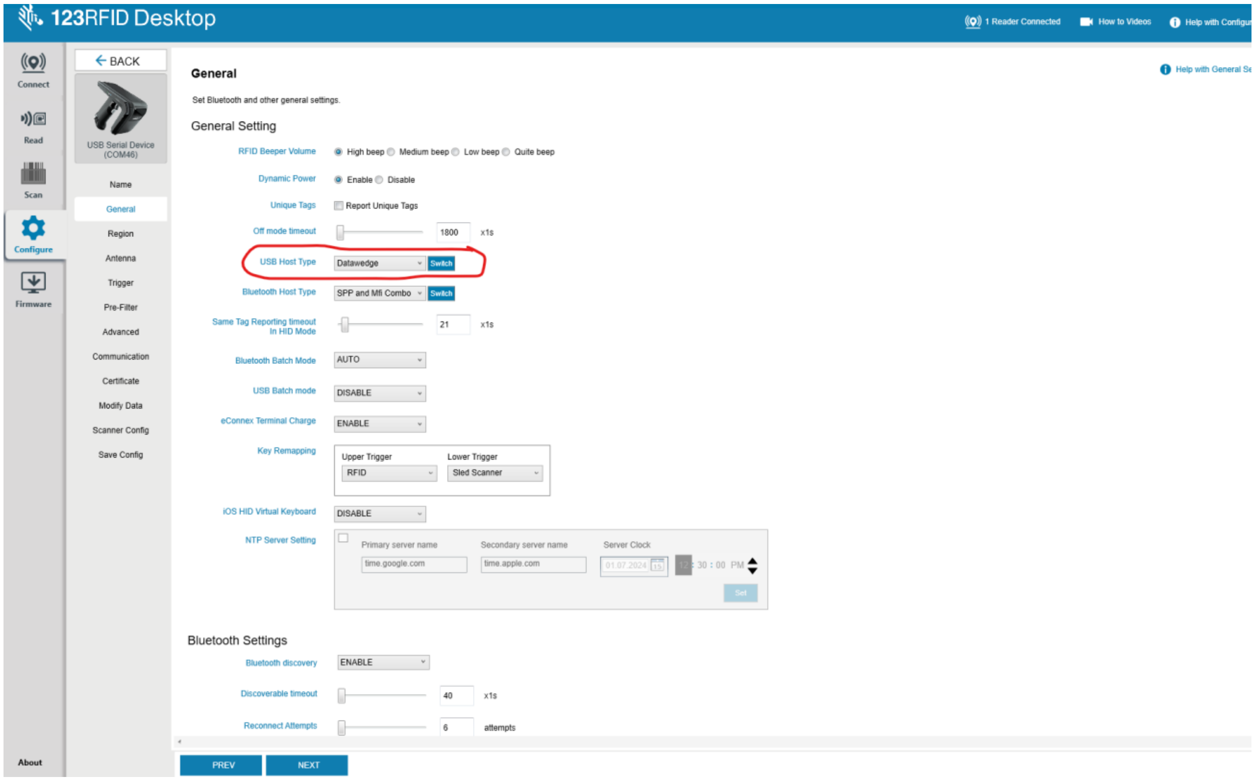
4. Scanning framework 🗋 DataWedgeApp_protected_aligned_signed_v5.apk

Firmware Configuration

Use the FW SAAFKS00-010-K05E0.DAT to verify data wedge support

Steps to follow:

1. Update the latest shared FW SAAFKS00-010-K05E0.DAT
2. After successful update perform factory reset
3. By default, out of the box/factory sled is not configured in data wedge mode. User needs to configure the same using below



General

Set Bluetooth and other general settings.

General Setting

RFID Beeper Volume ● High beep ○ Medium beep ○ Low beep ○ Quite beep

Dynamic Power ● Enable ○ Disable

Unique Tags □ Report Unique Tags

Off mode timeout 1800 x1s

USB Host Type Datawedge Switch

Bluetooth Host Type SPP and Mfi Combo Switch

Same Tag Reporting timeout In HID Mode 21 x1s

Bluetooth Batch Mode AUTO

USB Batch mode DISABLE

eConnex Terminal Charge ENABLE

Key Remapping Upper Trigger RFID Lower Trigger Sled Scanner

iOS HID Virtual Keyboard DISABLE

NTP Server Setting □ Primary server name time.google.com Secondary server name time.apple.com Server Clock 01.07.2024 12 : 30 : 00 PM Set

Bluetooth Settings

Bluetooth discovery ENABLE

Discoverable timeout 40 x1s

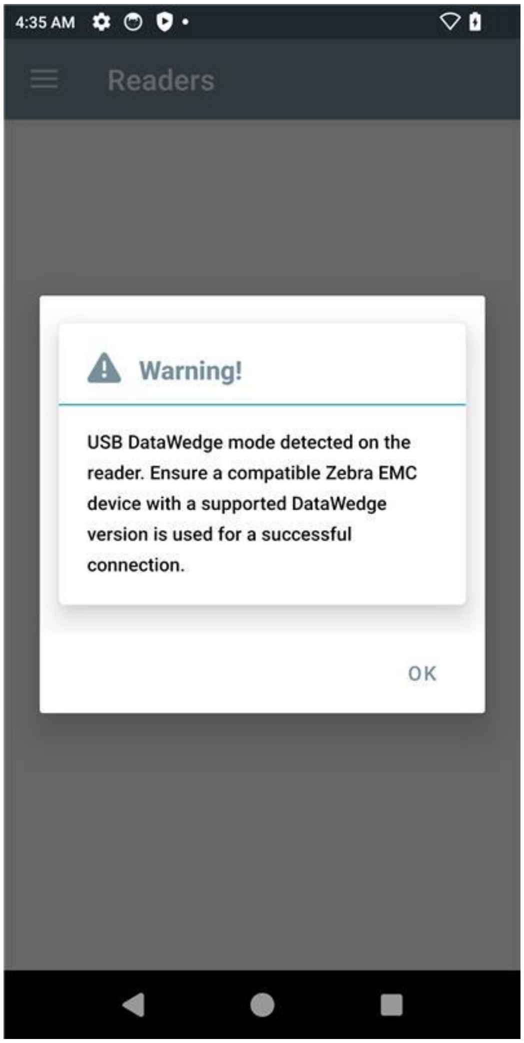
Reconnect Attempts 6 attempts

After above step USB will reset, that is connection to serial terminal will be lost

4. Now user should be able to perform following actions
   a. Switch to 123RFID sample app and perform barcode scan
   b. Switch back to DW demo, and scan barcodes
   c. Steps I & j can be repeated

Connection failure scenario

In scenario where RFDXX device is configured for USB Datawedge and when you try to connect using incompatible or the old BSP user will observe connection failures.



Warning!

USB DataWedge mode detected on the reader. Ensure a compatible Zebra EMC device with a supported DataWedge version is used for a successful connection.

Issues seen during testing

1. When user disables beeper completely in DW demo app, Sled still beeps upon decode
2. When decode button clicked very fast decode is not happens at the same rate & error beep heard from sled

Q&A

Q. What happens when DW profile other than RFID profile needs to access the USB scanner, i.e. SF needs to access the interface when the RFID application is in background?

A. When RFID application goes to background, it releases the scanning interface. Also, since new "RFID" DW profile is not active, SF will access the device like normal as per the configuration in the active DW profile.

Q. What happens when RFD40/90 scanner (setup in custom DW mode) is attached when RFID application is in background and SF has an active user waiting for using the scanner

A.

- When RFID application is in the background, will not attempt to enumerate/access the scanner interface (SSI interface) of the RFD40/90.
- SF can enumerate the scanner interface and notify the DW for scanner connection.
- DW will enable and use the scanner interface for scanning function as per the DW profile settings configured

Q. What happens when USB scanner is attached when RFID application is in background

A.

- When RFID application is in the background, will not attempt to enumerate/access the scanner interface (SSI interface) of the RFD40/90.
- SF can enumerate the scanner interface and notify the DW for scanner connection.
- SF will acquire or release the SSI interface based on the settings on active DW profile..

Q. What happens when USB scanner is detached

A. Both SF and RFID SDK need to release the SSI interfaces

Q. What happens when RFID application shifts from foreground to background

A. New DW profile "RFID" becomes inactive and RFID application releases the scanning (SSI) interface

Q. What happens if RFID application crashes

A. Below is expected to happen:

- From inter-operability point of view, new “RFID” profile becomes automatically inactive as this counts as RFID application exit
- SF will access the SSI interface based on any active DW profile configuration
- Upon next spawn, RFID application/aar might need to **re-acquire** the RFD40/90 and follow the device access flow.

Q. Is this the default configuration of RFD40/90 scanner

A. No. This is custom RFD40/90 configuration where the mode is changed for DW and RFID applications both to access the scanner. In regular RFD40/90 “out of the box” mode, SF will not recognize the RFD40/90 as a USB scanner. That mode is reserved for exclusive RFID apps

Q. What happens when EMDK application needs to access the USB scanner interface?

A. When EMDK application is in foreground, RFID application is in background. Hence, RFID application will not try to access or enumerate the scanner

Q. What happens when “DW” mode RFD device is connected when RFID application is in the background

A. RFID application will not attempt to enumerate or initialize the scanning interface. It will only attempt to connect to the scanner when the RFID application is in the foreground.

Next time when the application is in the foreground. RFID application will have to follow process create a profile

Q. What happens when the RFID application is opened and RFD40/90 device is non-DW mode?

A. If device is not in DW mode, RFID application proceed to access scanner interface normally like today. DW/SF is not going to access scanning interface when RFD40/90 is in non-DW mode

Q. What happens when non “RFID” DW profile based application or EMDK application is opened and RFD40/90 device is not configured in DW mode?

A. SF is not going to access scanning interface when RFD40/90 device is not in DW mode configuration. Hence, EMDK or DW application will not be able to use the scanning functionality as device is not configured in compatible mode. Device needs to be configured in compatible “DataWedge” mode where it exposes “SSI DataWedge Interface”