



Release Notes

RFID SDK Windows v3.0
August 2021

Contents

Overview	1
Device Compatibility	1
Version History.....	2
Components.....	4
Installation – Requirements.....	4
Using Demo application	5
Known Issues/Limitations	5

Overview

Zebra RFID SDK for Windows provides a .NET application programming interface for Zebra Handheld reader (RFD8500 Standard, RFD8500).

Device Compatibility

For the compatible devices list, please visit following page.

<https://www.zebra.com/us/en/support-downloads/software/developer-tools/zebra-rfid-sdk-for-windows.html>



Version History

Version 3.00.0018 – 08/2021

1. Sample app enhancement – provide an option to select preferred region.
2. Bug Fixes
 - a. When an inventory abort command fails, inventory session is aborted.
 - b. Running inventory with phase info enabled, resulting in overflow exceptions from the SDK and 123RFID Desktop hangs.
 - c. Connecting to device fails if attempted to connect after power cycling, while inventory was running.
 - d. On disconnect of a particular RFD40, disconnect event is being sent for all the connected RFD40 devices.
 - e. Key validation fails if the system locale is changed to a setting other than English.
 - f. Unable to Read/Write tags when a Prefilter setting with INV_B and Singulation STATE B is already set in the Device.

Version 2.01.0009 – 04/2020

1. Bug fix: Unable to Read/Write tags when a Prefilter setting with INV_B and Singulation STATE B is already set in the Device.

Version 2.01.0005 – 07/2019

1. Sample app enhancement – Trigger Mode Setting (allows a user to programmatically switch between barcode scanning and RFID reading) added to sample app.
2. Bug Fixes
 - a. Transmit Power Index not updated after power cycle
 - b. Read/Write operation failure when Trigger is set to handheld
 - c. Connect after Restore defaults failure
 - d. RFID Sample application
3. Port identification improvements for Serial Port Communication

Version 2.01.0002 – 11/2018

1. Access Operations – Tag Read, Tag Write, Tag Lock and Tag Kill
2. Configuration Updates
 - a. Antenna – Link Profile, Tari
 - b. Singulation – Inventory State, SL Flag
 - c. Tag Reporting – PC, RSSI, Phase, Channel Index, Tag Seen Count and Report Unique Tags
 - d. Regulatory - Channel Index
 - e. Power Management – Dynamic Power
3. Pre-Filters
4. Tag Locate



Version 2.00.0001 – 06/2018

1. Windows Desktop new SDK.

Version 1.02.0006 – 02/2018

1. Minor Bug Fixes.

Version 1.02.0005 – 09/2017

1. Batch mode support added.

Version 1.02.0004 – 08/2017

1. Minor Bug Fixes.

Version 1.02.0003 – 08/2017

1. Added serial port number into RFID Device properties.

Version 1.02.0002 – 08/2017

1. Removed serial port communication.
2. Stability improvements.

Version 1.02.0001 – 08/2017

1. Removed serial port number (COMX) from RFID Device properties.
2. Connection logic has been changed when RFID serial port get non-maximum serial port id.
3. Operation summary notification events.
4. ERROR_INSUFFICIENT_POWER exception introduced.
5. More informative exception messages.
6. Diagnostic logs.

Version 1.02.0000 – 07/2017

1. Battery, Power and Temperature notifications.
2. Configure Beeper volume.

Version 1.01.0000 – 07/2017

1. See reader capabilities.
2. PeakRSSI and TagSeenCount in TagData.
3. Configure Start/Stop triggers.
4. Get status events when inventory starts and stops.
5. Get an array of Tags that was read by the reader when data is not attached to ReadNotify event.
6. Purge all tags present in DII and Reader queues when data is not attached to ReadNotify event.



Using Demo application

Note: This SDK has only been tested using the Microsoft Bluetooth Communication stack. Use of proprietary drivers using a different communication stack/layer may have undefined results.

1. The demo application exe (RFIDDemoApplication) can be found inside “DemoApp\Bin” folder. Once you launch the application, if you have an already paired RFD8500, application will populate it and show in the combo box in following format <Bluetooth friendly name><COMX>
2. Bluetooth friendly names consist of device the family name and serial number which users can match with the label available on the device.
3. To pair a new device, the RFD8500 device should be discoverable, in range and the pairing request should be authenticated by pressing the yellow trigger of the device when device blink its blue LED faster. Please watch the video (<https://www.youtube.com/watch?v=JxLkE7GVTrQ>) for more information.
4. If the user restores the device to a factory defaults configuration, the device will disconnect. To connect the reader again, user may have to un-pair and re-pair the device.

Known Issues/Limitations

1. This SDK has only been tested only with Microsoft Bluetooth Driver stack and behavior with any other Bluetooth driver stacks is unknown.
2. New device pair notification (RFIDReaderAppeared) may not occur intermittently. Use the GetAvailableRFIDReaderList for most up-to-date reader list.
3. Session 0 and 1 inventory may reduce the performance as it takes time to process the data provide events. So, subsequent commands may fail (throw exceptions) after stopping the inventory.
4. Only TranmitPowerIndex changes are possible in Antennas.config
5. TagTransitTime implementation is not complete in Antennas.SingulationControl.
6. RFIDReader.VersionInfo property provides only the API DLL version at this time.
7. The device disconnection status notification takes more than 20s to report the disconnection.
8. The SDK uses an emulated Bluetooth serial port to communicate with the device. For the SDK to function properly, the serial port should not be opened by any other application or service.
9. In a situation where the SDK cannot open the serial port, the SDK API will throw the appropriate exception.
10. The RFD8500 automatically switches to a SNAPI USB communication mode when it is plugged in via the USB cable (disables Bluetooth). As such, it cannot be used with this SDK API while plugged-in.
11. GetReadTags and PurgeTags consider only the DLL queue. Batched data inside the reader is not considered.
12. Start/Stop trigger must be configured by setting the TriggerInfo property in RFIDReader.Config. Individually setting properties/Fields in RFIDReader.Config.TriggerInfo has no effect. Please use Demo app source code as a reference.