

Unitech

LibRFIDiOSSDK Migration Guide

2024.07.15

Summary

- This document provides basic information about importing a framework, build connection using unitechRFID SDK, and the comparison of *LibRFIDiOSSDK*.

Chapter Overview

- unitechRFID Connection Overview: This chapter introduces the connection interface and basic process of unitechRFID.
- unitechRFID Integration Overview: The chapter guides user to integrate a framework into their project.
- unitechRFID Command Comparing Sheet: This chapter provides comparison sheet between the *LibRFIDiOSSDK* and the *unitechRFID* of public apis
- If you are not familiar with the process about how to import and framework, please refers to the *unitechRFID Integration Overview* chapter.

unitechRFID Connection Overview

The unitechRFID SDK has integrated the connect functions of *ExternalAccessory* framework into following api, so users no longer need to deal with related affairs of the framework themselves.

Function Name	Usage	Note
<i>startDetect</i>	Start detect paired unitech reader device	
<i>stopDetect</i>	Stop detect	
<i>didConnectReader</i>	Triggered by paired unitech device	This is a protocol method of DetectReaderDelegate
<i>didDisconnectReader</i>	Triggered by unpaired unitech device	This is a protocol method of DetectReaderDelegate
<i>connect</i>	Start connect to the UHFReader	
<i>disconnect</i>	Stop the connection of the UHFReader	
<i>stop</i>	Stop current tasks on connected reader device	

Basic connection process:

1. *startDetect*
2. If an unitech device is detected, the unitechRFID will call *didConnectReader* method.
3. Inside *didConnectReader* method, users can specify which reader to connect.

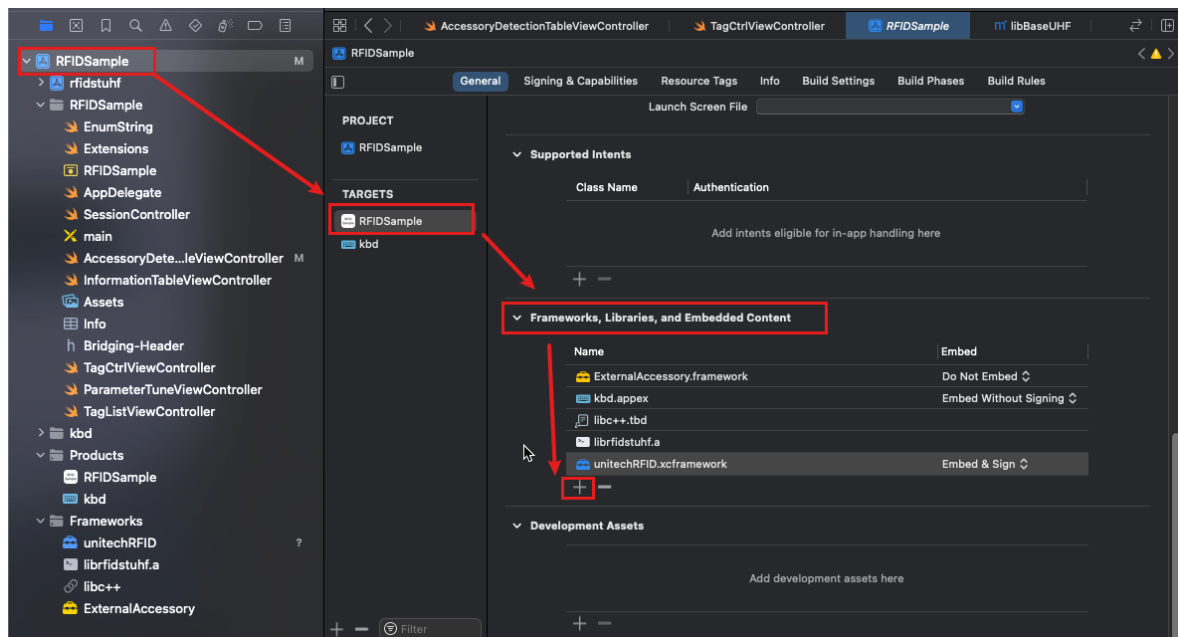
For the detailed connect process, please refers to the Migrating Device Connection from LibRFIDiOSSDK chapter.

unitechRFID Integration Overview

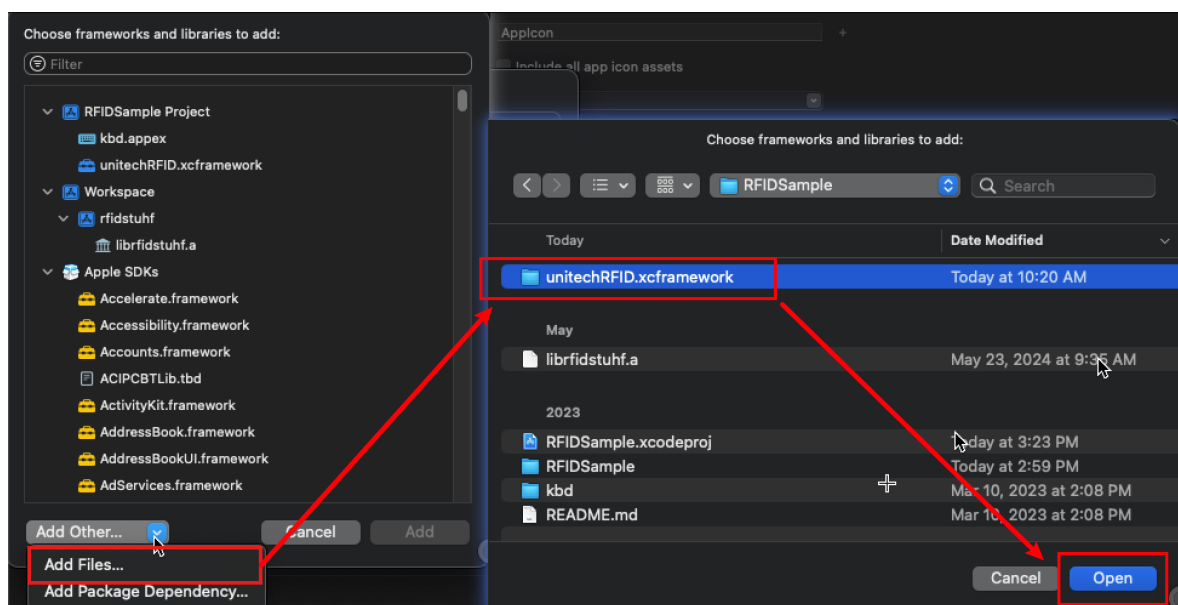
- This chapter walks you through the process of how to add the *unitechRFID.xcframework* to your project, and how to migrate the connection process.

Add unitechRFID to your project

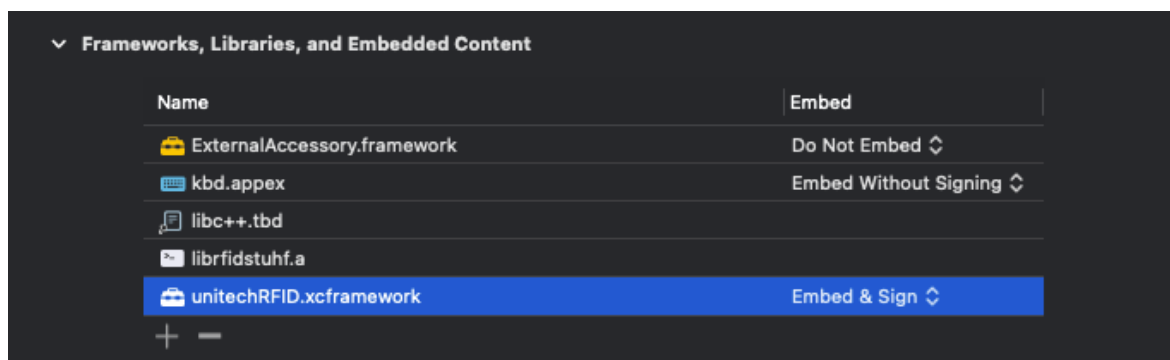
1. To add the unitechRFID.xcframework to your project, locate the project navigator to Project -> Targets -> Frameworks, Libraries Embedded Content section, and press the “+” button.



2. In the new window, click on the *Add Other..* button, and choose the *Add Files* item. Open the *unitechRFID.xcframework* folder.



3. The `unitechRFID.xcframework` file should have been added to your project.



Migrating Device Connection from LibRFIDiOSSDK

- The new unitechRFID framework has implemented accessory connection affairs, users can use straightforward methods to *detect*, *connect*, or *stop* tasks on the reader easily.
- This chapter will guide you to introduce the unitechRFID framework into your project with the RFIDSample project, and replace the ExternalAccessory framework you might used to build connection with the reader.

1. Import unitechRFID to the AccessoryDetectionTableViewController of RFIDSample project (or the place that handles connection affairs in your project), and add these variables:

```
import unitechRFID

let detectReader: DetectReader = DetectReader()
var unitechDevice: UnitechDevice?
var unitechDeviceList: [UnitechDevice] = []
var uhfReader: UhfReader?
```

2. Inherit the `detectReader.delegate` in the `viewWillAppear`, and `startDetect()` readers:

```
override func viewWillAppear(_ animated: Bool) {
    super.viewWillAppear(animated)
    detectReader.delegate = self
    detectReader.startDetect()
}
```

3. The implementation of the *DetecReaderDelegate* protocol is required, use paired *unitechDevice* object to initialize an *UhfReader* instance. With the *UhfReader* instance, user can build connection to reader, and inherits the *UhfReaderDelegate* protocol to receive various device event and detailed status like battery level, reader action state, key state, temperature... etc.

```
extension AccessoryDetectionTableViewController: DetectReaderDelegate {  
    func didConnectReader(_ reader: unitechRFID.UnitechDevice) {  
        unitechDevice = reader  
        uhfReader = UhfReader(reader)  
        uhfReader?.uhfReaderDelegate = self  
        let connectResult = uhfReader?.connect()  
    }  
  
    func didDisconnectReader(_ reader: unitechRFID.UnitechDevice) {  
    }  
}
```

unitechRFID Command Comparing Sheet

- This sheet lists the *unitechRFID* corresponding class name and method name of *LibRFIDiOSSDK*. If you need detailed parameter or usage of each function, please refer to the *RFIDSample* project and the *Scanner iOS Programming Guide* document.

LibRFIDiOSSDK		unitechRFID		
Class Name	Method Name	Class Name	Method Name	Objective-C Method Name
baseUHF	setFastMode()	BaseUhf	setFastMode()	-
baseUHF	getFastMode()	BaseUhf	getFastMode()	getFastMode_OC()
baseUHF	setContinuousMode()	BaseUhf	setContinuousMode()	-
baseUHF	getContinuousMode()	BaseUhf	getContinuousMode()	getContinuousMode_OC()
baseUHF	setIdleTime()	BaseUhf	setIdleTime()	-
baseUHF	getIdleTime()	BaseUhf	getIdleTime()	getIdleTime_OC()
baseUHF	inventory6c()	BaseUhf	inventory()	-
baseUHF	stop()	BaseUhf	stop()	-
baseUHF	setInventoryCfg()	BaseUhf	setIdleTime()	-
baseUHF	setProtocolCfg()	BaseUhf	setIdleTime()	-
baseUHF	setPower()	BaseUhf	setPower()	-
baseUHF	getPower()	BaseUhf	getPower()	getPower_OC()
baseUHF	setEncoding()	BaseUhf	setEncoding()	-
baseUHF	getEncoding()	BaseUhf	getEncoding()	getEncoding_OC()
baseUHF	setAlgorithmType()	BaseUhf	setAlgorithmType()	-
baseUHF	getAlgorithmType()	BaseUhf	getAlgorithmType()	getAlgorithmType_OC()
baseUHF	setTARI()	BaseUhf	setTari()	-
baseUHF	getTARI()	BaseUhf	getTari()	getTari_OC()
baseUHF	setSession()	BaseUhf	setSession()	-
baseUHF	getSession()	BaseUhf	getSession()	getSession_OC()
baseUHF	setBLF()	BaseUhf	setBlf()	-
baseUHF	getBLF()	BaseUhf	getBlf()	getBlf_OC()
baseUHF	setStartQ()	BaseUhf	setStartQ()	-
baseUHF	getStartQ()	BaseUhf	getStartQ()	getStartQ_OC()
baseUHF	setMaxQ()	BaseUhf	setMaxQ()	-
baseUHF	getMaxQ()	BaseUhf	getMaxQ()	getMaxQ_OC()
baseUHF	setMinQ()	BaseUhf	setMinQ()	-

LibRFIDiOSSDK		unitechRFID		
Class Name	Method Name	Class Name	Method Name	Objective-C Method Name
baseUHF	getMinQ()	BaseUhf	getMinQ()	getMinQ_OC()
baseUHF	setToggleTarget()	BaseUhf	setToggleTarget()	-
baseUHF	getToggleTarget()	BaseUhf	getToggleTarget()	getToggleTarget_OC()
baseUHF	setTarget()	BaseUhf	setTarget()	-
baseUHF	getTarget()	BaseUhf	getTarget()	getTarget_OC()
baseUHF	setInventoryTime()	BaseUhf	setInventoryTime()	-
baseUHF	getInventoryTime()	BaseUhf	getInventoryTime()	getInventoryTime_OC()
baseUHF	getSupportedGlobalBands()	BaseUhf	supportedGlobalBand()	supportedGlobalBand_OC()
baseUHF	getSKUString()	BaseUhf	SkuType.description	getSkuTypeDescription()
baseUHF	getGlobalBandString()	BaseUhf	GlobalBandType.description	-
baseUHF	setGlobalBand()	BaseUhf	setGlobalBand()	-
baseUHF	getGlobalBand()	BaseUhf	getGlobalBand()	getGlobalBand_OC()
baseUHF	kill6c()	BaseUhf	killTag()	-
baseUHF	readMemory6c()	BaseUhf	readMemory()	-
baseUHF	writeMemory6c()	BaseUhf	writeMemory()	-
baseUHF	setSelectMask6c()	BaseUhf	setSelectMask()	-
baseUHF	setSelectMask6cEnabled()	BaseUhf	setSelectMaskEnabled()	-
baseUHF	unlock6c()	BaseUhf	lockTag()	-
baseUHF	lock6c()	BaseUhf	lockTag()	-
baseUHF	permaLock6c()	BaseUhf	permaLockTag()	-