



ONYXKIT IOS SDK QUICKSTART GUIDE

Thank you for downloading and evaluating the **ONYXKIT for iOS**. Hopefully this guide will provide the information you need to be up and running with biometric security, but please also check our new support portal located at:

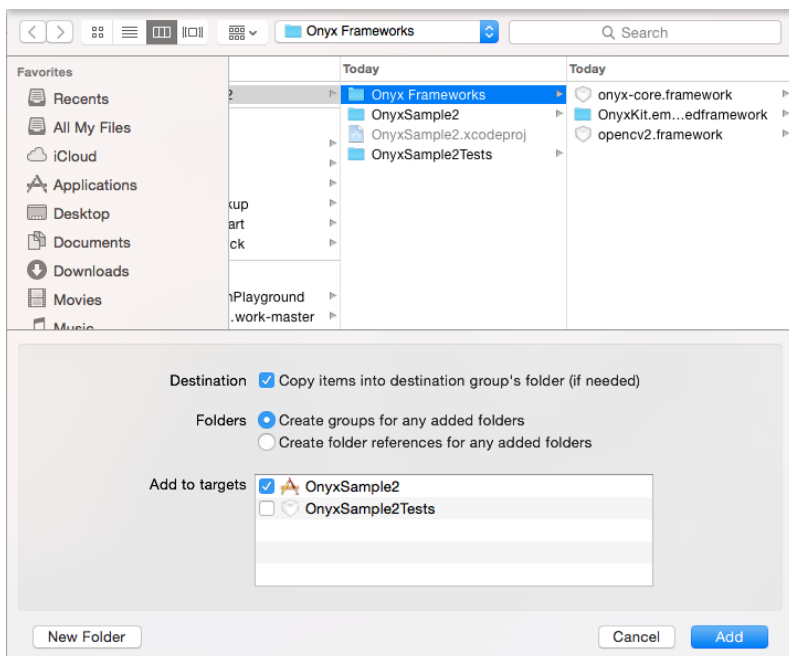
<http://support.diamondfortress.com>

To implement the **OnyxKit iOS SDK, you must be developing using **Xcode** on an **Apple Macintosh**.*

IMPORTING ONYXKIT INTO YOUR PROJECT

In Xcode, Go to the **File** menu and select **Add file to Project**.

When prompted, find and select the **Onyx Frameworks**.



DIAMOND FORTRESS
TECHNOLOGIES

WWW.DIAMONDFORTRESS.COM

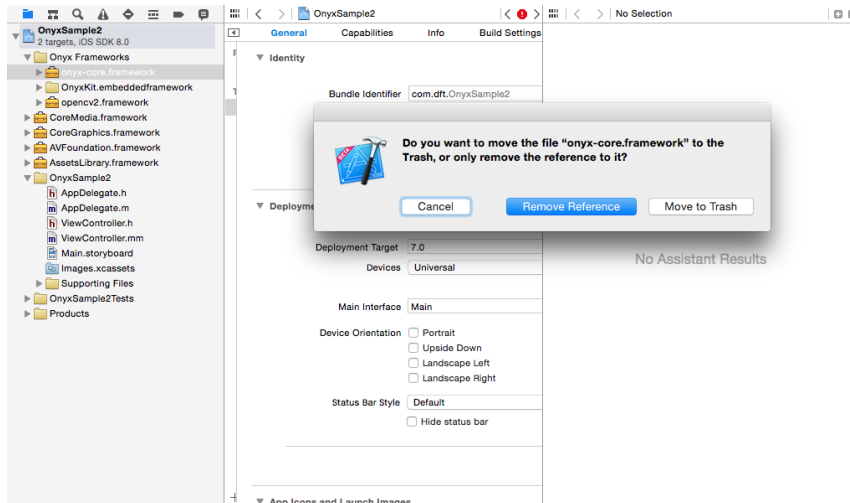


ONYX

Make sure the **Copy items into destination group's folder** option is checked.



After OnyxKit has been imported, Xcode will try to automatically set up framework search paths for you. Please verify these paths are correct before continuing. The OnyxKit framework already has the **OpenCV** and **onyx-core** frameworks. If either of these frameworks are already a part of your project you must remove them and use the OnyxKit versions to prevent duplicate symbols.



IMPORTING THE APPLE FRAMEWORKS

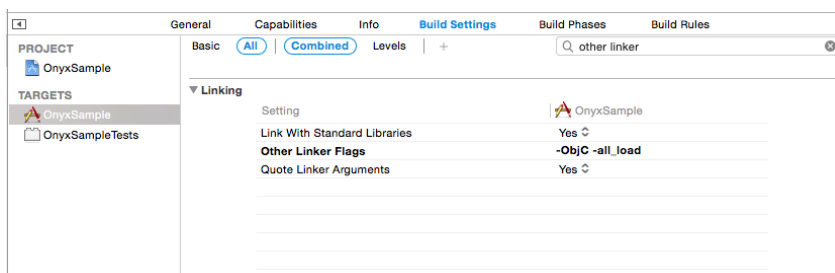
You must include these frameworks in your **Project Settings**:

- **CoreMedia**
- **AVFoundation**
- **AssetsLibrary**

SETTING LINKER FLAGS

Make sure you set **Other Linker Flags** in your target's **Build Settings** to:

```
-ObjC -all_load
```



CHANGE METHOD FILE EXTENSIONS

All method files that use ONYX must have an **.mm** extension. Xcode will compile **.mm** files with C++ flags. Any method files not using ONYX can keep their **.m** extension.



IMPORTING THE ONYX VIEW CONTROLLER

After you have the framework added to your project, you must then import the **OnyxViewController** header file.

```
#import <OnyxKit/OnyxViewController.h>
```

Make sure whatever view controller you wish to use ONYX is also a delegate for **OnyxViewController**.

```
#import <OnyxKit/OnyxViewController.h>
@interface ViewController : UIViewController <OnyxViewControllerDelegate>
```

USING THE ONYX VIEW CONTROLLER

The **OnyxViewController** is a modal that can be presented programmatically at any time.

```
OnyxViewController *vc = [[OnyxViewController alloc] initWithNibName:@"OnyxViewController"
                        bundle:nil];
vc.delegate = self;
vc.state = ONYX_SINGLE;
[self presentViewController:vc animated:YES completion:nil];
```

That's it. That's all you need to do to present Onyx.

ONYX VIEW CONTROLLER STATES

ONYX_SINGLE [Default] - The single state will capture one print and perform a callback with this print.

ONYX_QUAD - The quad state will capture four prints and perform a callback with the one best quality print.

ONYX DELEGATE METHODS



This method will return the raw image:

```
- (void) Onyx:(OnyxViewController *)controller  
didOutputRaw:(UIImage *)image;
```

This method will return the processed image:

```
- (void) Onyx:(OnyxViewController *)controller  
didOutputProcessed:(UIImage *)image;
```

This method will return the enhanced image:

```
- (void) Onyx:(OnyxViewController *)controller  
didOutputEnhanced:(UIImage *)image;
```

This method will return the template: (This data can be stored and used later with our match function.)

```
- (void) Onyx:(OnyxViewController *)controller  
didOutputTemplate:(NSData *)data;
```

This function will be called on controller completion.

```
- (void) Onyx:(OnyxViewController *)controller  
didComplete;
```

All of these delegate methods are optional. Place the ones that you need in your view controller.

MATCHING FINGERPRINT TEMPLATES

```
#import <OnyxKit/Onyx.h>
```

You can match 2 fingerprint data templates with the match method. It will return "YES" if they match and "NO" if they don't.

```
[Onyx match:ft1 with:ft2]
```

FOR ADDITIONAL SUPPORT

Please see our support portal at:

<http://support.diamondfortress.com>

Email support is available by contacting:

support@diamondfortress.com

We also have premium support packages available at all levels for additional cost. Please contact your DFT representative for more details.