SDK Overview

VINScanner SDK allows to add a quick and precise recognition of VIN barcodes in applications for iOS. The codes are successfully recognized in cases of blurriness and noise, which are often when using the camera in difficult conditions.

SDK carries out all necessary actions on work with the device cam itself. SDK also performs verifications of scanned VIN codes and uses optimal amount of pixels to provide quick and reliable recognition.

SDK can be used on devices with iOS 4.0 or later.

A trial mode with restrictions on the number of codes to be scanned and logo display is also supported.

Please see example project for demonstration of SDK usage.

SDK Structure

B4HScannerController.h
Declaration of classes, protocols and constants.

scannerlicense.plist
License file. Optional.

libB4HScanLib.a
VINScanner SDK static library.

Adding SDK into the project

The file with the static library and header file should be included into the project.

SDK involves dependences from the following frameworks:
- UIKit.framework
- Foundation.framework
- QuartzCore.framework
- CoreVideo.framework
- CoreMedia.framework
- AVFoundation.framework
- Security.framework

Important!
The application should be built with -ObjC key. SDK classes can't be used with iOS versions older than 4.0

SDK Usage

It is possible to check when SDK is ready for work by calling CheckReadyStatus - returns B4HScannerController status (readiness or error). A verification of license file, number of uses in the trial mode, presence of a cam on a device and availability of frameworks necessary for use are carried out. Check of the cam presence is not carried out on a simulator.

Description of classes and protocols

```
@interface B4HScannerController : UIViewController
- (BOOL) isRunning;
- (void) startScanning;
- (void) stopScanning;
- (B4HScannerLibraryStatus) CheckReadyStatus;

@property (nonatomic, retain) B4HCameraOverlayViewController *overlay;
@property (nonatomic, assign) id <B4HScannerDelegate> delegate;

@end
```

The main scanner class. Its view is image from the cam without any control elements. There is a logo in a trial mode. Only landscape orientations are supported. Start of a cam and the process of recognition happens at view controller's viewWillAppear. A stop and cam disability takes places at viewDidDisappear.

By calling these functions:
```
- (BOOL) isRunning;
- (void) startScanning;
- (void) stopScanning;
```
it's possible to track whether scanning and recognition process is working, start and stop scanner.

```
@interface B4HCameraOverlayViewController : UIViewController { }
@property (nonatomic, assign) B4HScannerController *parentScanner;
@end
```

The class serves for modification of scanner interface. Having inherited an arbitrary class from B4HCameraOverlayViewController and having connected to scanner controller as an overlay property, it is possible to put view over scanner's view. Correspondingly, presentation of B4HCameraOverlayViewController should be transparent for a display of a picture from the cam.

The use of classes inherited from B4HCameraOverlayViewController in the application is not a necessary thing, as on default B4HScannerController contains B4HCameraOverlayViewController. At the start on a simulator this controller displays a button in the centre, which imitates code scanning. B4HScannerController is built into the hierarchy of app controllers, but not B4HCameraOverlayViewController.

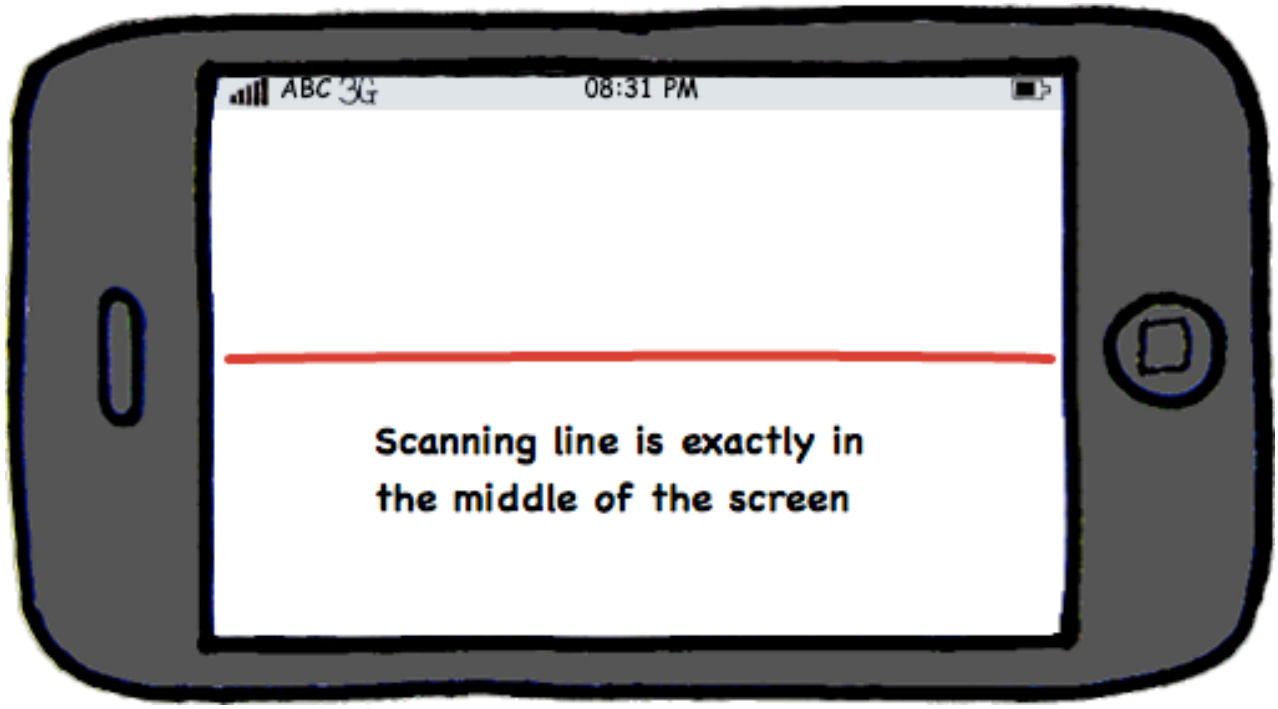
```
@protocol B4HScannerDelegate <NSObject>
```

```
@required
- (void)scanner:(B4HScannerController *)scanner gotCode:(NSString
*)code;
@end
```

The protocol of a delegate which receives the code recognized by the scanner. Any controller, not only B4HCameraOverlayViewController, can be in the role of a delegate.

Scanning of codes

Scanning is always carried out along the long side of the screen, the line passing exactly in the middle.



If a device camera supports autofocus, then focusing happens automatically, and can't be disabled.

At the start on a simulator, scanning of a code is emulated by clicking "return code" button, located in the centre.