

# Zebra HD4000 Kalipso Sample

## 1. Components

- a. **KZebraHud** - Kalipso project that contains the Global Action to interface with Zebra HD4000.
- b. **ZebraHudInterface** – Android Studio Source code that exposes the HD4000 to Kalipso.
- c. **HudViewer** – Application to display the HD4000 glasses on a PC for presentations
- d. **Zebra\_HD4000\_Picking\_Sample\_01** – Sample Kalipso Picking project using HD4000. Inside there is a Sample\_HD4000\_DemoBarcodes.pdf with barcodes that can be used for this demo.
- e. **Darwin\_Android\_Service\_v4\_01.apk** – Zebra HD4000 Android service

## 2. Requirements

- Zebra HD4000 glasses
- Zebra Android device compatible with HD4000 like TC77.
- Zebra Darwin Android Service installed on the device. In this folder you will find Darwin\_Android\_Service\_v4\_01.apk, but you should check with Zebra if a more recent version is available
- Kalipso Designer

## 3. How to use HD4000 in your Kalipso projects

Create your own Kalipso project and link to the provided KZebraHud Kalipso project.

The first thing to do is to call **hudInitialize** to initialize the connection to the HD4000 glasses. When you are done, or before closing your project, you should call **hudDeinitialize** to close the connection to the glasses.

After the connection is established, you can call any of the other Global Actionsets from the **KZebraHud** project to send content to the glasses.

- **hudShowText** – will send a Title and a Description to the HD4000 glasses. The User Interface cannot be modified.
- **hudShowControl** – will generate an image of the current content of a Kalipso control and send it to the HD4000. The best option here is to use a container like the ScrollArea, where you can place multiple Kalipso controls, and then send the entire ScrollArea into the HD4000.
- **hudShowImage** – will send an image file to the HD4000.

- **hudShowJSON** – will end the JSON command to the HD4000. For this you should read Zebra documentation to know what can be displayed on the glasses using JSON.
- **hudClearDisplay** – will clear the HUD display
- **hudSetMicrophoneOn** - can be used to turn on/off the HD4000 microphone
- **hudSetBrightness** – can be used to control the HD4000 display brightness.

#### 4. Use the HD4000 camera

In order to use the HD4000 camera, you need to call **hudStartCameraWithPreview** if you want to display the Camera in your Kalipso App, or **hudStartCameraWithoutPreview**, if you don't want to show the Camera in your Kalipso App.

After you can call **hudGetImageFromCamera** to an image file with the current frame from the Camera. You can call this inside a Loop in a thread and also use Kalipso **BarcodeRecognize** action to try to recognize a barcode, until **BarcodeRecognize** returns something. This can be used for simple scenarios like reading a QRCode, it will not be good for 1D or intensive barcode scanning.

When you are done using the camera you should call **hudStopCamera**, to free the camera and resources.

#### 5. Send the content of the HD4000 display to your PC

If you need to make a presentation where the audience is unable to use the HD4000 to see the display, you can send the contents of the HD4000 display to your PC.

First start the App **HudViewer.exe** on your PC, then in your Kalipso project call **hudStartCasting**, passing your PC IP address as parameter. This should establish a connection to your PC. After this, any content you send to the glasses will also be sent to your PC.

If you want to terminate the connection, call **hudStopCasting**.

#### 6. Presentation Mode

If you open the Darwin Service in the Android device, it has an option called "Presentaion Mode".

If you enable this option, then in Kalipso you can call **PresentForm** to send a form into the HD4000. This will work in stream mode, basically streaming the "Presented Form" into the glasses.

For this you don't have to use the **KZebraHud** project. This is a standard feature in Android, you just need to call **PresentForm**. But do note, that while doing this, no other operation can be performed on the HD4000 glasses.

This operation mode is not advised, since it is a continuous heavy process of streaming a Form into the glasses, but is available for some use cases where it may be a better alternative for using with the HD4000.

## 7. Further extend the connection between Kalipso and HD4000

In folder **ZebraHudInterface**, is the Android Studio source project used to expose the HD4000 API to Kalipso. You can modify it, or create your own, to add additional capabilities to the Kalipso connection. You should then compile a ZebraHudInterface.apk.

We have already done this in this sample, and this compiled APK has been copied to the **FilesToSend** folder in the **KZebraHud**. If you modify this project and compile a new APK, you should put the new APK in the **FilesToSend** folder.