
Project : *SwitchBot***Project manager** : Davin Sufer
Document title : Developer guide

In this document we present some details related to SwitchBot project implementation and a quick source code walkthrough.

History :

Version	Description	Author	Date
0.1	First outline	Raouf Bensalem	Feb 16th, 2015

Project structure

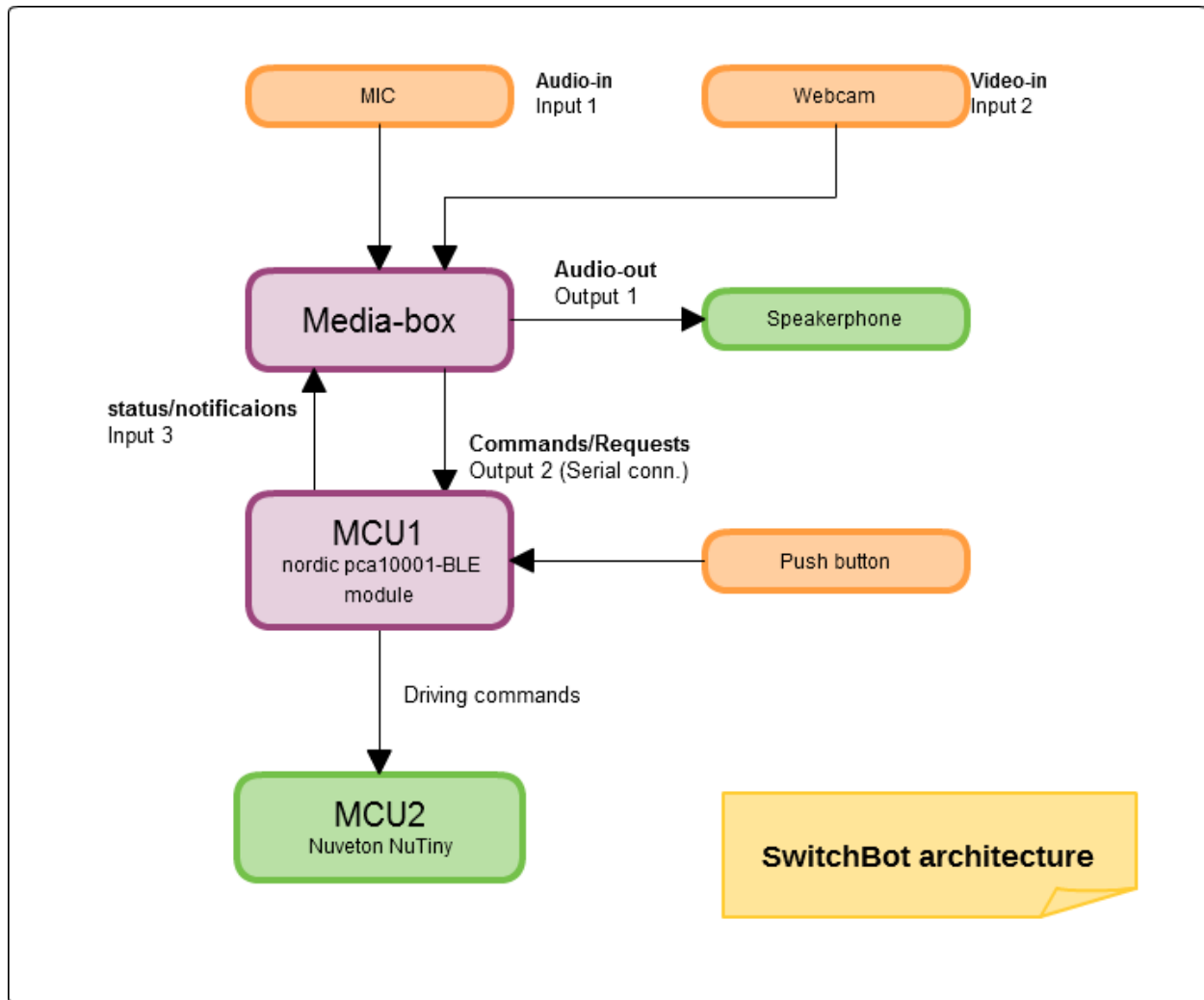
In the project repository we have the following folders :

- **addons** : this folder contains complementary projects
- **android** : this is the main project folder, it contains the application intended for the mediabox
- **documents** : project documentation
- **microcontrollers** : code source intended for the 2 microcontrollers (nordic pca10001 and Nuveton NuTiny)

In the “addons” folder we have two complementary projects **SBTouchpad** and **Telepresence**. **SBTouchpad** is an application that we need to install on a android tablet to interact with SwitchBot. This later provide the following functionalities :

- Remote control for SwitchBot motions and driving (`source: com.wowwee.touchpad.SwitchBotRC.java`)
- Get SwitchBot status (USB connection, Bluetooth, WiFi, Battery level, Telepresence connection - `source : com.wowwee.touchpad.SwitchBotStatus.java`)
- Results related to some requests (for example : show a map) (source : N/A)
- Launch the Telepresence application, (source : addons/Telepresence)

Hardware architecture



This diagram shows the main hardware components of SwitchBot :

- **Media-box** : this is an android board, we're using an **ODROID-C1** board
- **MCU1** : this microcontroller is a **Nordic pca10001** board which provide a BLE module so we can connect to SwitchBot from a tablet (**SBTouchpad** application)
- **MCU2** : this microcontroller is a **Nuveton NuTiny**, it is responsible to control SwitchBot motions and driving.

Main application

We can find the project for the main application in the folder “android”. This is the application that we have to install in SwitchBot media-box. This latter is composed by the following packages :

- **com.she.ai** : it contains the source code for SHE assistant
- **com.wowwee.switchbot** : contains the entry point of the application, which is MainActivity and some classes to manage the USB connection
- **com.wowwee.telepresence** : contains a client for the LightStreamer server
- **com.wowwee.util** : contains SwitchBot protocol (**SBProtocol.java**) and some utility methods (**Utils.java**)

Virtual/Real modes

To be able to test SwitchBot app even if we don't have a USB connection to a microcontroller, we added the interface SBUbDevice (`source :com.wowwee.switchbot.SBUsbDevice.java`). We have two implementation for this interface, **SBRealDevice** when we have a connected microcontroller to the media-box (real mode), **SBVirtualDevice** when we don't (virtual mode). We use the boolean member **IS_REAL_DEVICE** to select which implementation we need (line 32 in **MainActivity.java**) .

For real mode for example we set **IS_REAL_DEVICE** to true and we uncomment lines 40, 44, 45 and 46 in the AndroidManifest.xml file. For virtual mode it's the opposite.

In the virtual mode the physical push button that we use to start a voice command is replaced by the “speak now” button of the application GUI.

Telepresence

The Telepresence application (addons/Telepresence) is an Adobe AIR application which provide Video/Audio streaming from SwitchBot to the user's tablet. It consists of two projects

- **TelepresenceRobot** : to be installed on SwitchBot Media-box
- **TelepresenceUser** : to be installed on the user's tablet

To launch the application we use a [LightStreamer](#) client

(source : `com.wowwee.telepresence.PushServer.java`) which is instantiated in

MainActivity.java line 63 .

When a request to launch the Telepresence app is fired through the LightStreamer server, a new intent for the TelepresenceRobot module is launched (see **PushServerListener.java** line 94 in `com.wowwee.telepresence`).

The LightStreamer server we're using is running in an Amazon EC2 instance, the url of this instance is in `com.wowwee.telepresence.Lightstreamer.java` line 40.

The projects structure is a [FlashDevelop IDE](#) structure.