

Email: ankitdaf@gmail.com

Short description: This project aims to allow the user to easily switch between controlling multiple paired Bluetooth devices with the Tecla Shield , and provide a client application and UI to allow the Shield to be used with Accessibility software on PC / Mac across multiple OSes

## Summary

This project aims to allow the user to easily switch between controlling multiple paired Bluetooth devices with the Tecla Shield , and provide a client application and UI to allow the Shield to be used with Accessibility software on PC / Mac across multiple OSes.

## Implementation :

### 1. Switch devices manually

The Shield can switch which device it is connected to by the following process :

- a. Each device polls the Bluetooth Server socket (Tecla Shield ) at fixed time intervals, say 30 seconds
- b. Each device has a default timeout, which is the time for which the device will not attempt to poll the Server Socket after it has just been disconnected, to allow the other device to poll and successfully allow the switch to take place

This switching scheme will be implemented manually as follows :

#### For Android :

Provide a menu preference of the type CheckBoxPreference titled "Temporarily Disconnect Bluetooth connection with Shield" . If this is checked, it disables the automatic polling of Server Socket by the Android device

#### For PC :

Provide a simple one screen Preference menu that has a similar checkbox. This checkbox will temporarily disable code to poll the Shield allowing sufficient time for the other device to connect to the Shield.

Additionally, it might be possible to have multiple Bluetooth connections using Threads, as some discussions on forums show. The game "Air Hockey" seems to have an implementation of such a connection in 4 player mode, by using unique UUIDs.

This approach could be implemented if feasible.

Multiple devices might stay connected, but they may choose whether to act on the commands received from the shield by means of a simple checkbox, much like the Temporary Disconnect. In this case, the Disconnect need not be a socket disconnection, and would just involve discarding of any data received.

### 2. Thin translation layer for PC / Mac

This stage of the project will involve writing code that converts key events sent by the Shield over Bluetooth to corresponding Key press actions used by Accessibility software.

Different Accessibility tools use different key schemes to control the underlying OS . The layer must provide translation from the Tecla Shield scheme to control schemes used by at least a few popular accessibility software. This is proposed to be done using a cross-platform language, preferably Python. The PyBlueZ library can be used with Linux and Windows, and LightBlue library can be used for Mac.

[ The LightBlue project is no longer maintained, so another language may be used instead in the absence of a workable Python library for Mac ]

Some popular accessibility tools to be included for support :

1. Microsoft On-Screen Keyboard (Windows)
2. WiVik OSK (Windows)
3. SwitchXS (Mac OSX )
4. GNOME OSK , GOK (Linux Ubuntu )

## Deliverables :

1. Code to add a Temporary Disconnect preference to the Android application , and associated code to disable polling  
( 2 weeks )
2. Python code on PC/Mac to poll Bluetooth ServerSocket if free  
( 1 week x 3 ( Linux,Windows, Mac ) = 3 weeks )
3. Python code to translate received Bluetooth Data to corresponding click / keypress event on PC / Mac based on Accessibility software in use

( 1 week x 3 ( Linux,Windows, Mac ) = 3 weeks )

4. Schemes to convert Tecla key events to other corresponding events on PC/Mac

5. Documentation, bug reports and fixes will be a part of the development cycle and may or may not add to the time estimated above

6. If time permits : Explore possibility of implementing Thread based Bluetooth connections to avoid having to disconnect devices.

The above time estimates are rough and depend on the availability of features on external libraries.

Contact details :

Skype : ankitdaf

GTalk : ankitdaf@gmail.com

IRC : ankitdaf

Link to pull request :

<https://github.com/jorgesilva/TeclaAccess/pull/44>

Portfolio :

All my projects have been documented here :

Portfolio : <http://www.ankitdaf.com/>

Github : <https://github.com/ankitdaf/>

About Me :

I have worked on a broad range of technologies and tools, both hardware and software. I have worked on projects involving micro-controller based Robotics, including line following robots and wired and wireless serial communication protocols.

I have decent working knowledge of web technologies and tools, and have written code-samples to implement them as a part of my various projects.

I have also worked for more than a year on Android applications, all of which involve a novel user interface and automation and simplification of useful tasks. These applications include gesture-based text sharing between a mobile device and a laptop over bluetooth, a prototype for an Android Braille keyboard for the blind and a touch-navigation-based Feedback gathering tool.

I am also well-versed with Python and have used it for more than a year. I developed a kernel trace visualization tool using Python and Matplotlib for CISCO Systems, Bangalore.