

# TPad Library Design Tools

## 1. Basic friction control

`sendTPad(amplitude)`

`sendTPadBuffer(amplitude array)`

## 2. Pre-defined “texture tone” waveforms

`sendTPadTexture(type, frequency, amplitude)`

## 3. Friction maps

`FrictionMapView`

## 4. Depth maps

`DepthMapView`

# HelloTPadTablet

- Using raw friction on the screen to make the finger slip or stick



- Good for things like “sticky buttons”, dynamic forces, hitting walls, etc.

# HelloTPadTablet

- Using time-varying friction to produce vibrotactile like stimulus. “Texture tone”

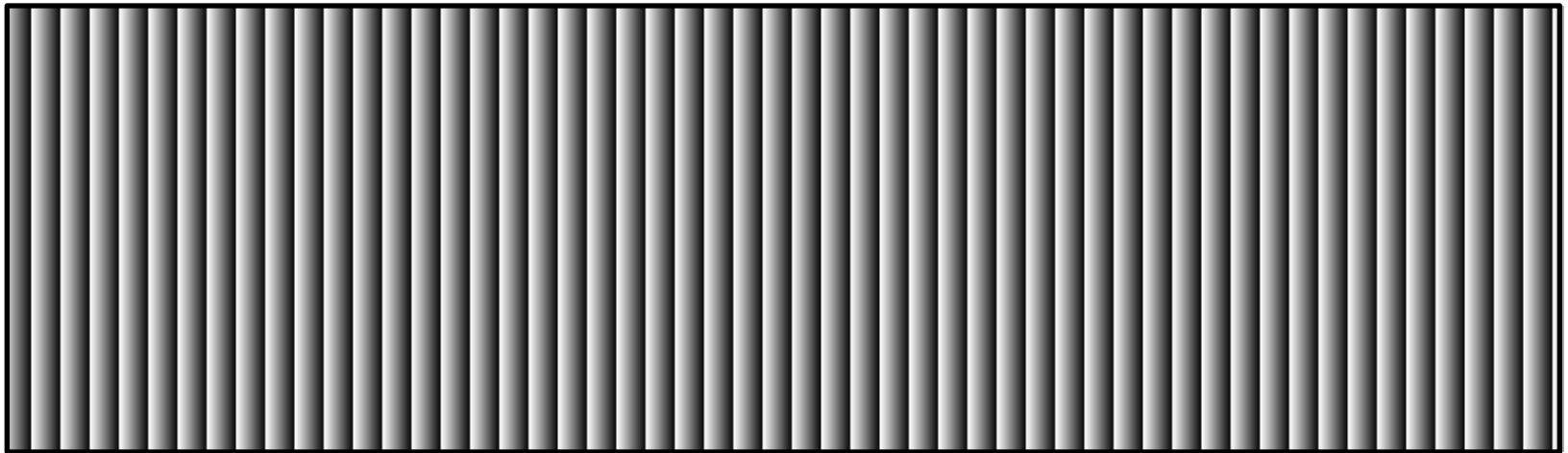
```
sendTPadTexture(sawtooth, 35, 1)
```

- Easy to implement and very strong sensation. Notifications, “active” elements, etc.

# HelloTPadTablet

- Spatial based map of friction values

FrictionMapView

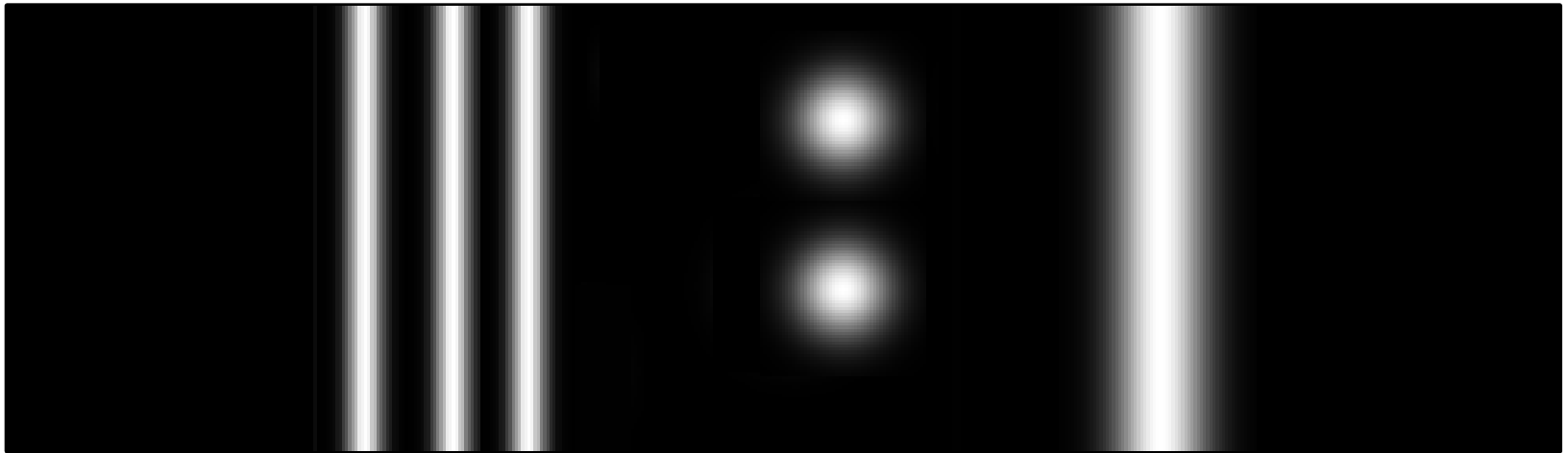


- Most versatile and straightforward tool.  
“Passive” texture elements, UI elements, small straightforward features, almost anything.

# HelloTPadTablet

- Spatial “depth” map. Alternative rendering.

DepthMapView



- Good for large features and simulated contours. Bumps, ridges, shapes.

# TPad App Design Tools

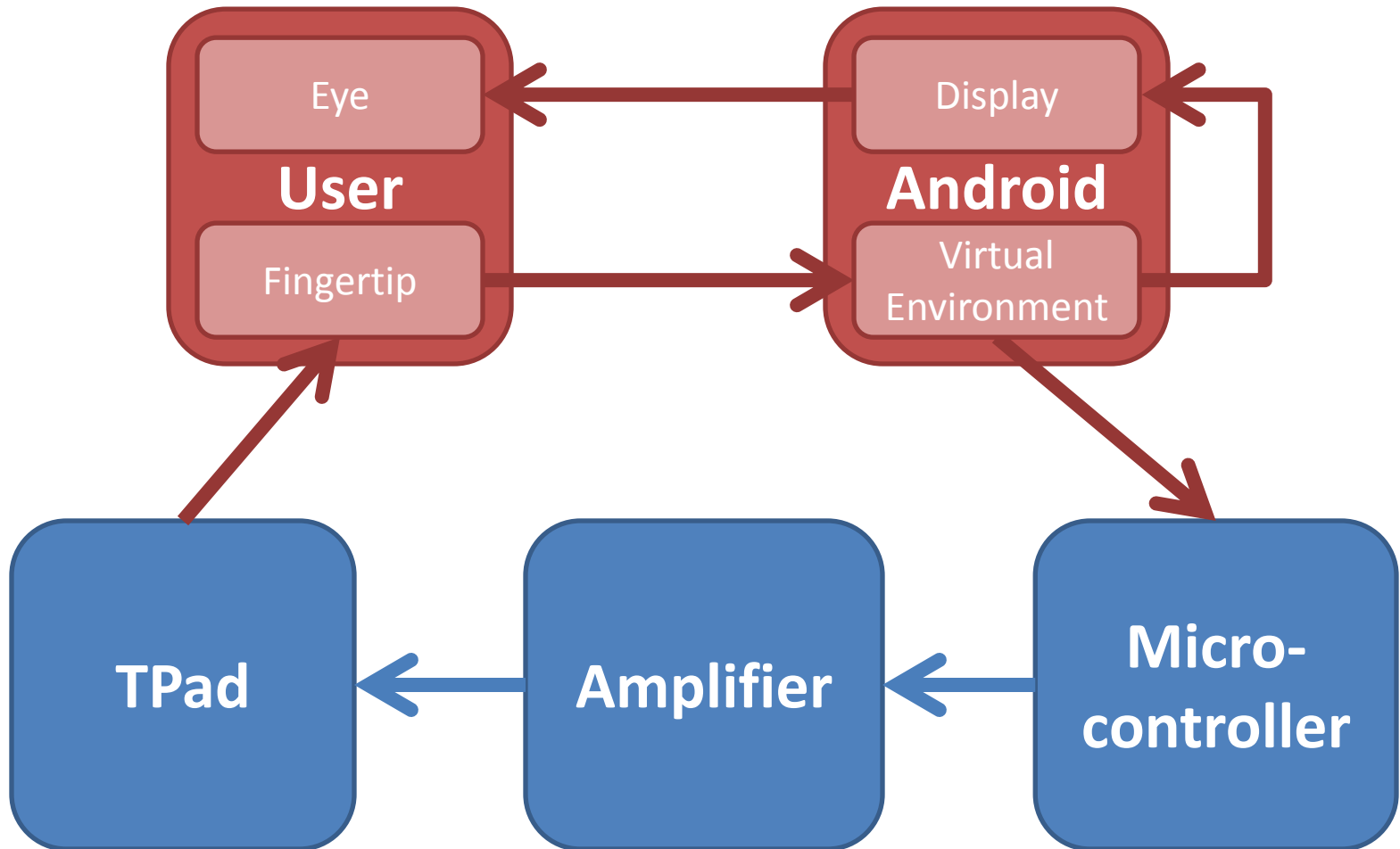
## Pre-built Prototyping Android Apps:

1. Pre-defined “texture tone” waveforms  
TimeTextureSampler
2. Friction maps  
TPadFrictionMap
3. Depth maps  
TPadDepthMap

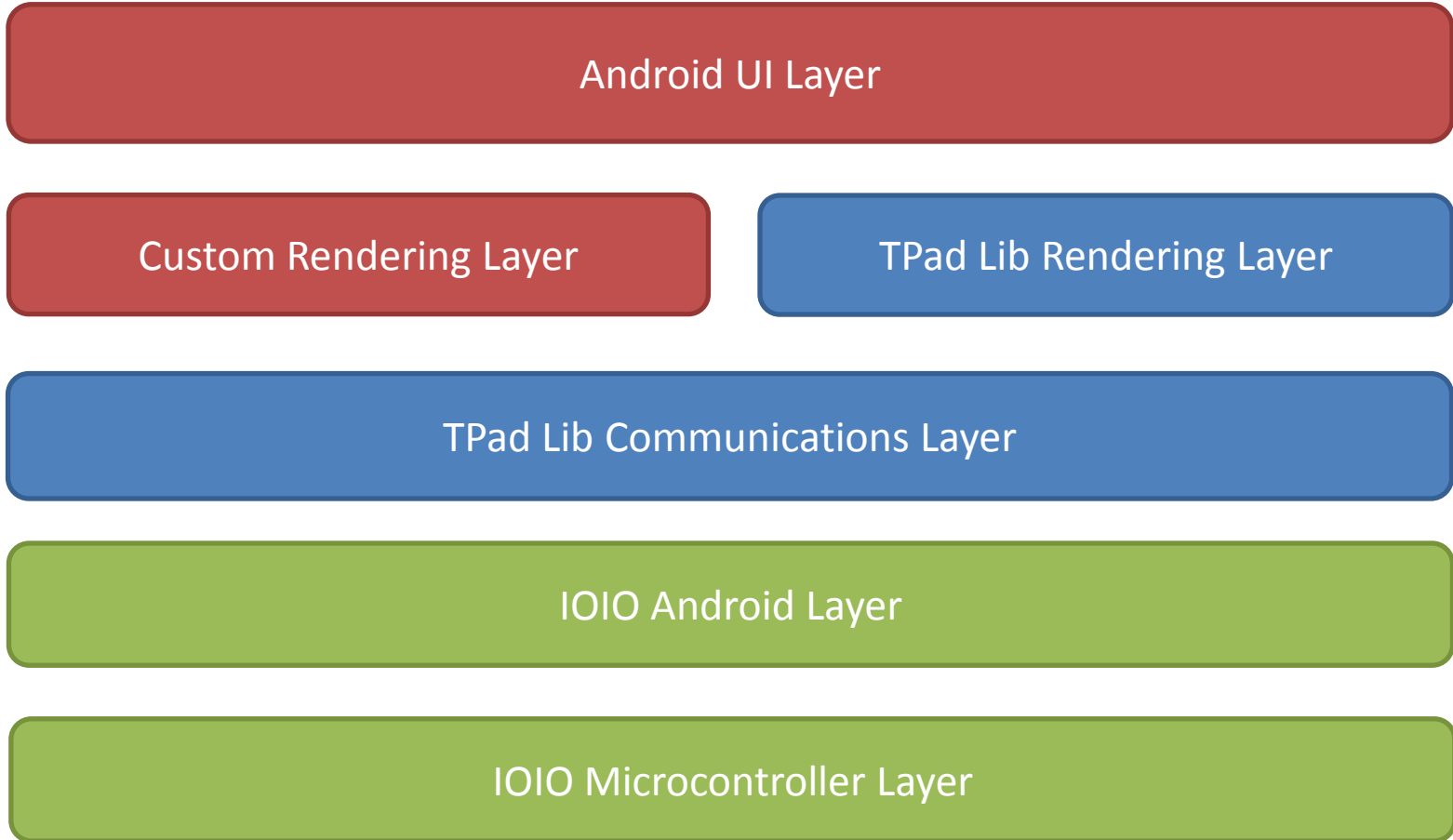
Located and updated on TPadTablet github:

<https://github.com/TPadTabletProject/TPadTablet>

# System Architecture



# Software Layers





# HelloTPadTablet Walkthrough

Line by line

Two main files to focus on:

1. HelloTPadActivity.java

Contains virtual environment and UI

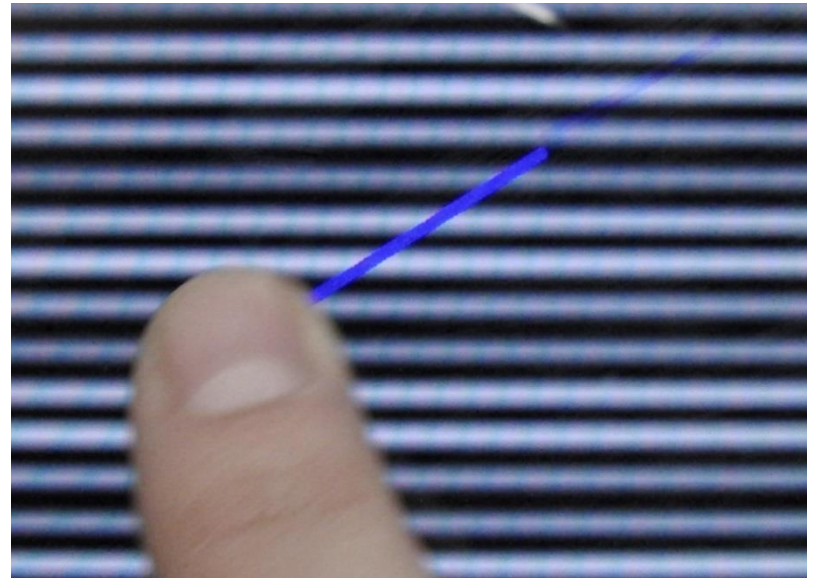
2. activity\_hello\_tpad.xml

Controls UI layout on the screen

# Important Note on Finger Position

The TPad Tablet Library views oversample the friction map/depth map data at a rate of 1kHz.

Fine feature rendering would be impossible this.



# I can't feel/hear the TPad

- Charge the external battery/Make sure it is turned on.
- Close and reopen your TPad application.
- Make sure the TPad is connecting over USB and asking if you would like to launch a TPad enabled app.
- Load a sample application to see if it works with the TPad.
- Check to make sure you are initializing the TPad with the correct frequency

# The TPad feels weak

- Charge the external battery.
- Check to make sure you are initializing the TPad with the correct frequency.
- Check what TPad values you are sending to the library, are they close to 1?
- Check to see if the TPad screen is still mounted securely/correctly.

# The TPad turns on/off randomly

- Charge the external battery.
- Load a sample application to see if it works with the TPad.
- Make sure you aren't sending conflicting values to the TPad, or not enough values
- Turn off the Tablet screen and turn it back on (This recalibrates the finger positioning, which may be miscalibrated)
- Charge the table.

# The TPad dies quickly/gets really hot

- Make sure you aren't leaving the friction value set to 1 for a long time (i.e. the device is staying on). There is a library function to help avoid this, but it can still be accidentally overwritten.

# The TPad is very loud!!

- Something is most likely wrong with the mounting of the TPad glass to the screen. Check for loose wires around the screen, and anything that might be contacting the screen accidentally.
- Check to see if you are initializing the TPad to the correct frequency.

# Design Takeaways

- TPad only works while user is moving their finger
- Only a single friction level is displayed on the entire screen
- Responsiveness of interface matters!  
Oversample finger positioning for best effects.
- Passive effects are more compelling and “intuitive” than active effects



# Thank You!

TPad Tablet Project  
tpadtablet.org

Joe Mullenbach  
mullenbach@u.northwestern.edu

Craig Shultz  
craigdshultz@u.northwestern.edu

