

SUNTKVA

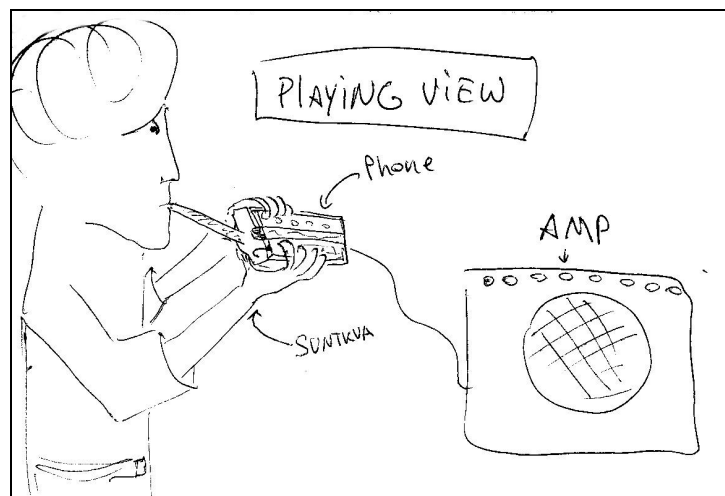
[georgian - “howl”]

Introduction

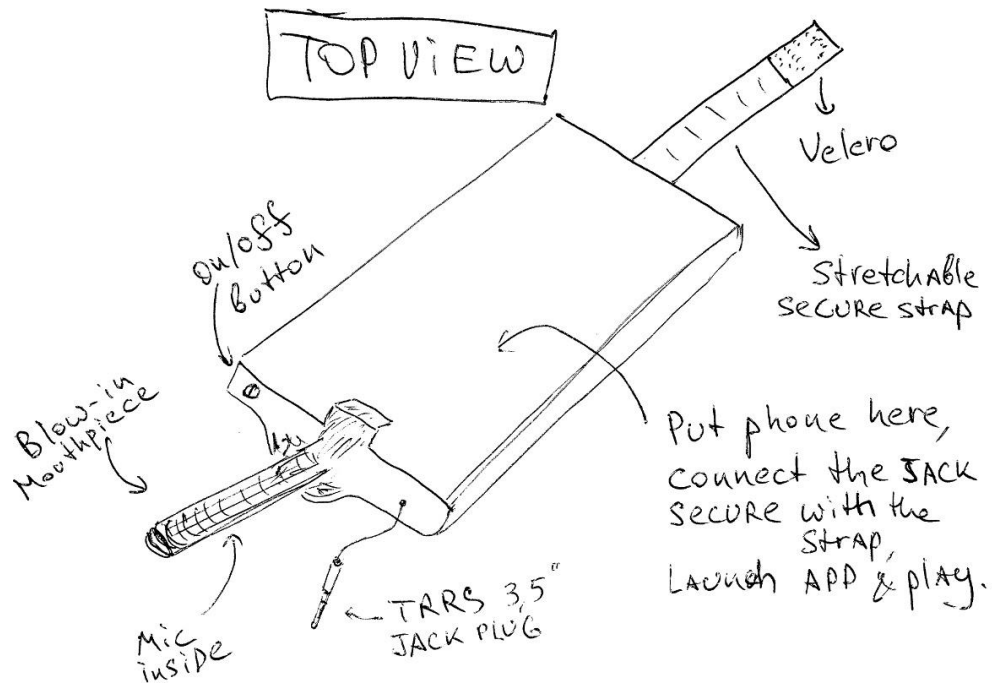
Suntkva is a hackable expressive hyperacoustic musical instrument. Download an app, plug your Suntkva into your smartphone's headphone jack and start playing right away. You can make music regardless of your skill level: even a five year old can pick it up and play, while advanced musicians can enjoy limitless customization and expression. You can choose one of the existing sounds from the soundstore, customize your sound using intuitive controls and connect to external speakers or amp via standard jack for better output. Musical hackers can customize Suntkva ad infinitum: write their own sounds and interfaces or even extend hardware, since it's built on top of Arduino and both software and hardware are open sourced.

How it works

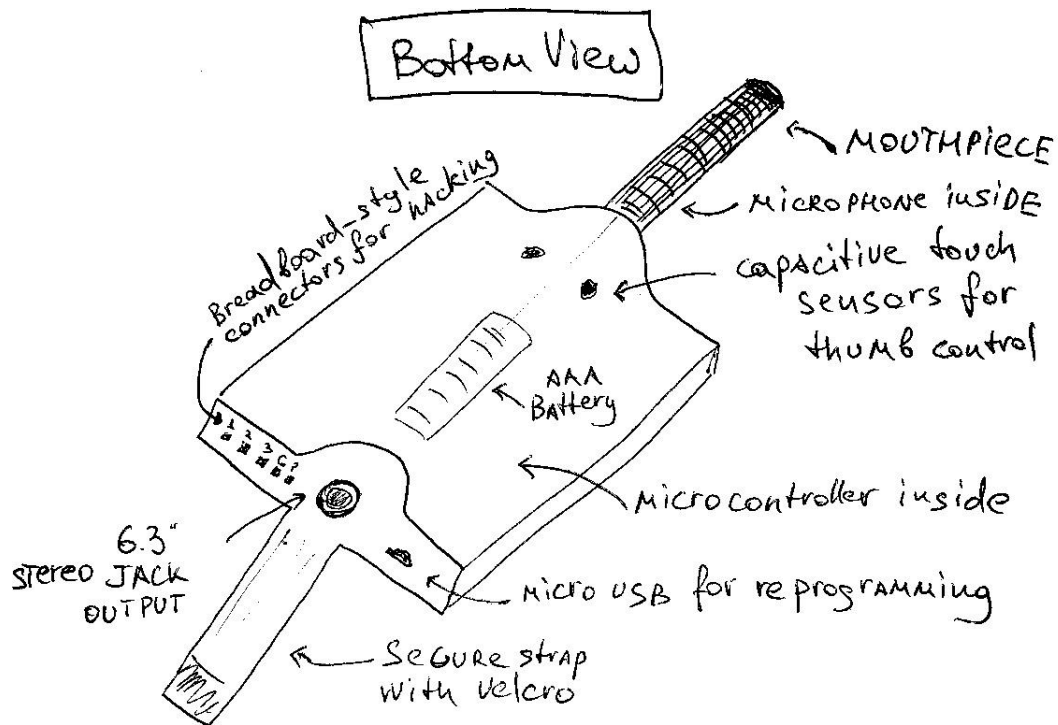
Suntkva is a microcontroller-powered device that you strap onto your smartphone (screen-up), connect the two via headphone jack and use a special app to play. Primary hardware controls include a blow-in mouthpiece and two capacitive touch sensors on the back(for thumbs).



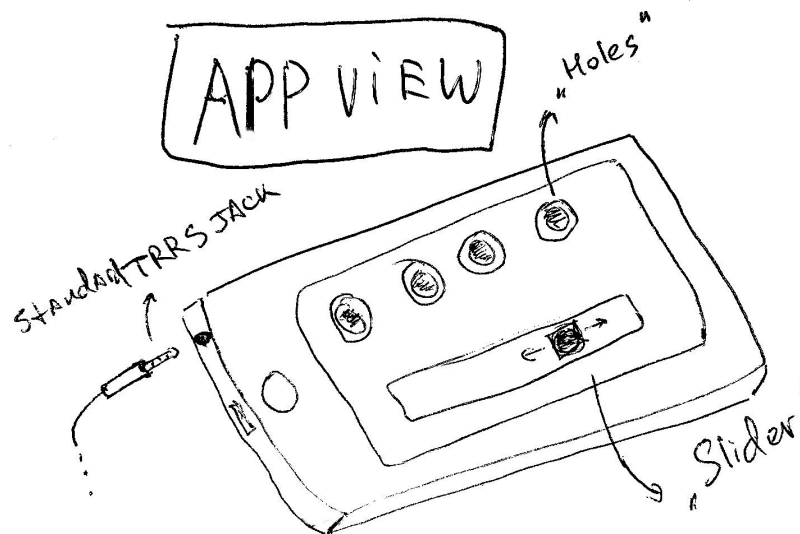
[Diagram 1 : Playing view]



[Diagram 2 : Top View without the smartphone]



[Diagram 3 : Bottom View without the smartphone]



[Diagram 4 : App View]

Software controls can be anything you want, but the primary mode includes 4 “holes” and one “slider”. Various sounds and control modes can be selected from the in-app soundstore.

Prior art & references

- [Koka's Bow Sensor](#)
- Open Pipe (<http://openpipe.cc/>)
- Ryan Janzen's [SqueaKEYS](#)
- [Mune - expressive electronic music instrument](#)
- [Smulle Ocarina](#)
- [iTrump and iBone](#)
- [Steve Mann's Hyperacoustic instruments](#)
 - In particular: [The electric hydraulophone: A hyperacoustic instrument with acoustic feedback](#) and http://www.eyetap.org/papers/docs/mann_physiphones.pdf
- [Echobo](#) - audience participation in a music performance
- [Expressiveness and digital musical instrument design \(paper\)](#)
- [HiJack](#) - hacking Jack port on a smartphone

Implementation notes & links

First prototype can be built with Arduino / MakeyMakey for hardware and Processing for Android. **NB! Looks like Processing for Android doesn't support Minim. So need to figure out how to use Tarsos DSP or similar.**

Useful links & example code:

- [Examples of audio processing in processing](#)
- [Rapid Android Development \(processing for Android\) \(book\)](#)