

Synthotron: Live Android Synthesizer

Michael Nye and Michael Ryan

Background

- Android OS has the majority of the smartphone market share but no music production applications suitable for live performance.
- Differences in hardware and software from each manufacturer and service provider mean that devices have wildly different latency times.
- Response times can be as high as 250 ms from the user pressing a button to the speaker emitting sound, which is unacceptable for responsive performance with traditional keyboard instruments.
- We created a music performance app for Android with an interface suited to higher input latency

Design

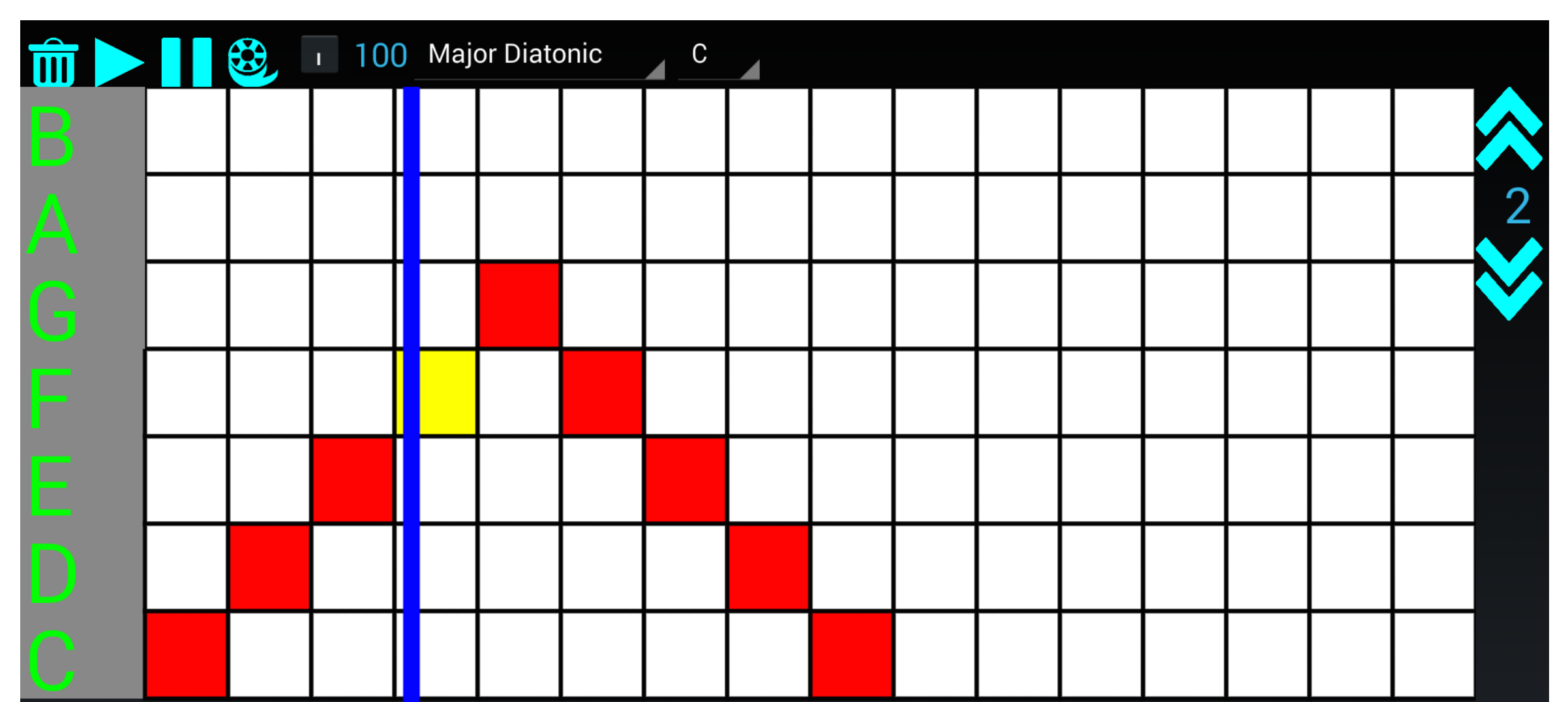
- Sound processing done in native C++ code
 - Composes processing blocks that pass audio between each other with circular buffers
 - OpenSL ES handles embedded audio
- Java interface to the native code is used to change tones and trigger notes
 - Schedulers run in Java and pass MIDI messages to the processing backend

Future work

- Optimize audio processing to lower CPU usage
- Design additional instruments and effects
- Extra MIDI samples (piano, guitar, etc)
- Cleaner and more beautiful user interface

Synthotron Package

- Two synthesizer instruments:
 - Subtractive Synthesizer with two source oscillators using antialiased waveforms
 - FM Synthesizer with adjustable carrier and modulating waveforms
 - Both with included adjustable effects:
 - LFOs for vibrato and tremolo
 - ADSR envelopes
 - Adjustable filters
 - Tone settings can be saved on device
- A sample-based drum machine:
 - Ring modulator to adjust tone frequencies
 - Compressor for reducing dynamic range
- Mastering channel
 - Reverb filter for naturalness
 - Limiter to prevent clipping
 - Four point EQ for tone control
- A piano roll for loop creation
 - Schedule notes in each octave of a scale
 - Hear playback and save for sequencing
 - Loops can be saved on device



- Sequencer for loop playback
 - Schedule loops on both synthesizers and drum machine for percussion, pads, and melody
 - Schedules can be saved on device