

# CSD2B C++ Synth

Kasper Spijkerman

# Inhoud

- Concept
- Gekozen Focus
- Class diagram
- Audio flow diagrams
- Reflectie

# Concept

- Vibe Synth -> AM synth met sinus oscillatoren
- Additieve Synth -> Square/Saw Synth met detuning oscillators
- Aangestuurd door 1 of 2 melodieën, zo meerdere partijen

# Gekozen focus

- Vooral dieper op melodie en UI ingegaan minder op de synthese zelf
- Flexibiliteit melody generatie -> compositie/productie

## Melody

setScale()  
chooseScale()  
getPitch()

Rootnote  
Scales  
Octavemultiplier  
octave

## Synth

synthTick()  
mtof()  
Virtual getSample()

## Oscillator

setFrequency() frequency  
setSampleRate() sample  
tick() samplerate  
Virtual Calculate() amplitude

## UI

retrieveUserinput()  
retrieveSelection()  
retrieveFloatInRange()  
retrieveDoubleInRange()

## Vibe Synth

createOscillator()  
getSample()

## Additive Synth

createOscillator()  
getSample()

## Square

Calculate()

## Saw

Calculate()

## Sine

Calculate()

Is a

Has a

Is a

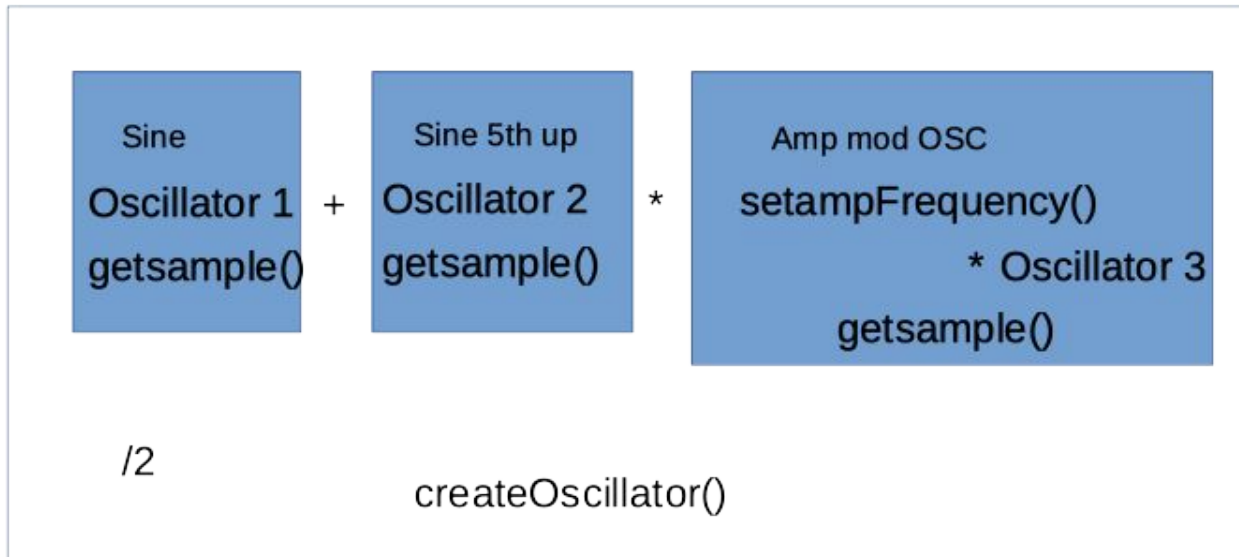
Is a

## Vibe Synth

Executes createOscillator() in constructor → creates a Vibe\_Synth object

All Oscillators get ticked by Synthtick()

Melody triggers the synthobject and updates with updatePitch.



Generated Melody (array)

Mtof()

setpitch()

MySynths[] = New Vibe Synth

MySynths[].getSample()

Synthtick()

Buffer

updatePitch()

## Add Synth

Executes createOscillator() in constructor → creates a Add\_Synth object

All Oscillators get ticked by Synthtick()

Melody triggers the synthobject and updates with updatePitch.

Generated  
Melody  
(array)

Mtof()

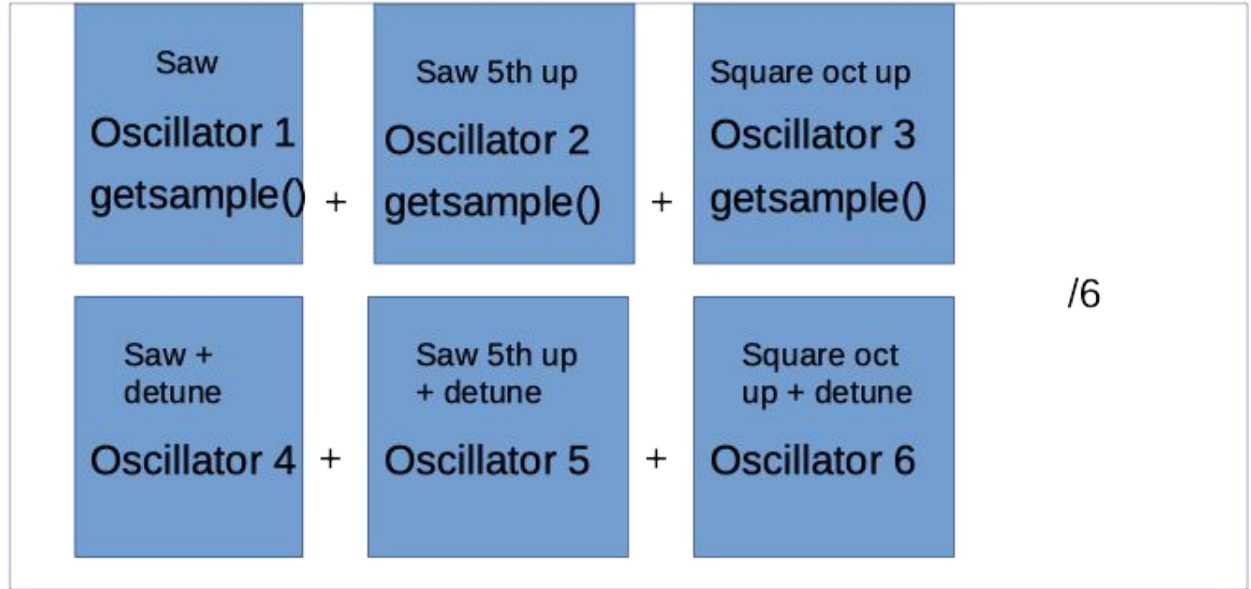
setpitch() createOscillator()

MySynths[] = New Add\_Synth

MySynths[].getSample()

Synthtick()

Buffer



updatePitch()

# Reflectie

- Leuk!
- Soepeler dan vorig blok -> meer vragen stellen/samenwerken
- Classes, C++ compiler, Meerdere bestanden, Synthese en Diagrammen maken
- Soms lastig overzicht houden, volgende keer -> minder duplicate code
- Veel tijd: 10 plus uur per week