

MEDIALAB

TEXT INPUT NAAR AUDIO OUTPUT

Johnny Berkman

KRIS KARDINAELS EN LIEVEN MENSCHAERT

FASE 1.

OPDRACHT: standard shape recognition with processing

Voor deze opdracht heb ik me gericht op een filmpje dat ik vond tijdens mijn onderzoek voor mijn thesis. (<http://www.youtube.com/user/juanprocessing#p/u/7/6zQiuNhjgo>) Hier maakten ze gebruik van processing om de vorm van letters te definiëren en daaruit dan bepaalde frequenties te halen.

Mijn idee was dan om ongeveer hetzelfde te doen, maar dan beginnend met de standaard basisvormen (cirkel, vierkant,...)

Probleempunten: Externe camera aansluiten met Processing (oplossing was een driver installeren, maar daarna ging het bergaf. Omdat mijn kennis van processing en algoritmes te laag was, besloot ik het over een andere boeg te gooien.

FASE 2. (NIEUWE DOELSTELLING)

Mijn thesis ging meer naar een input gerichte vorm van tekst, daarom besloot ik om tijdens de lessen medialab een nieuw concept te bedenken dat ik kan gebruiken als prototype van mijn thesis.

Het nieuwe idee was om met een extern toetsenbord input te geven (in de vorm van tekst) en deze door te sturen naar Arduino, waarop een programma staat, die dan deze input omzet naar Pure Data, waar dan uiteindelijk deze data wordt omgezet naar geluid.

STAPPENPLAN:

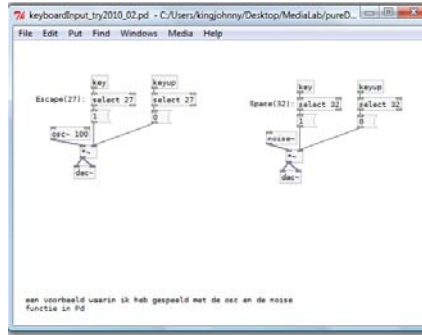
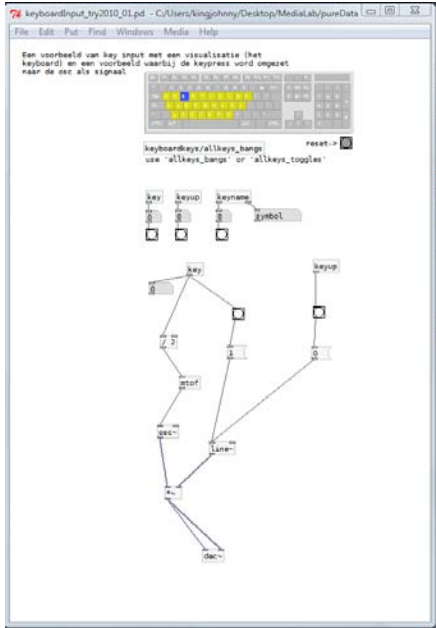
1. Pure data in orde brengen (laatste stap als eerste)
2. Uitzoeken hoe extern toetsenbord aangesloten moet worden op Arduino
3. Libraries zoeken van Arduino voor toetsenbord
4. Arduino prepareren
5. Arduino verbinden met Pure Data (ref Pduino)
6. Quality check

Pure data

Ook dit heb ik in stappen aangepakt, want eerst ben ik op zoek gegaan naar informatie over het aansturen van Pd ahv een klavier. De eerste stappen waren om de keypress te definiëren en er geluid uit te krijgen. Gelukkig ondersteunt Pd een hele helpfunctie waar je heel snel en gemakkelijk info kunt vinden over hoe je zoiets moet aanpakken (zie afb. Helpfunctie Pd)

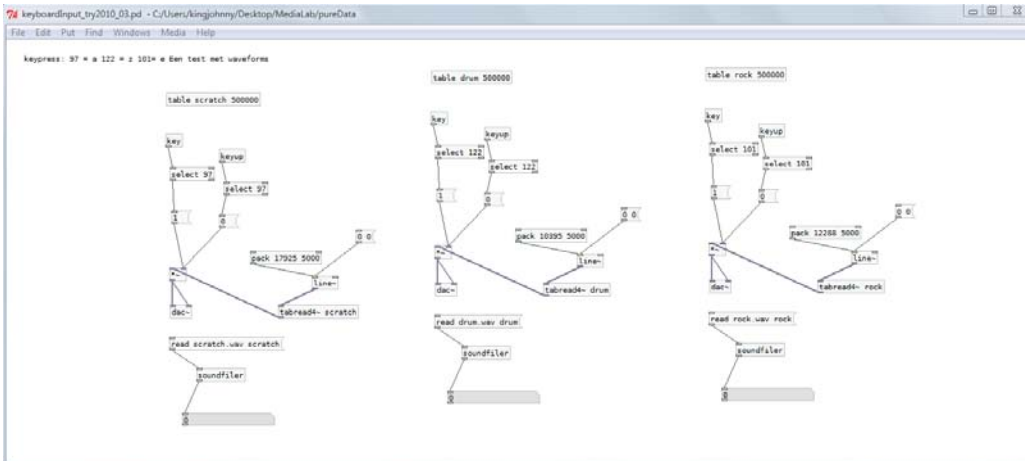
Daarna ben ik zelf wat op zoek gegaan naar andere voorbeelden en specifiek om te kijken hoe ik bepaalde knoppen kan specificeren om met 1 bepaalde knop (bvb de spatiebalk) een sound te triggeren. (zie afb. Esc en Espace).

Uiteindelijk wilde ik een soort synthesizer creëren waarbij door middel van keypresses waveforms (drums, snares, hihats,...) kon laten afspelen (zie afb Waveforms). Maar dit was een te omslachtig werk (omdat ik nog niet heb uitgedokterd hoe ik deze werkwijze universeel kan maken), daarom besloot ik het bij de 'eenvoudige' oplossing te houden, dus met een 'universele' keypress verschillende frequenties aanhalen.

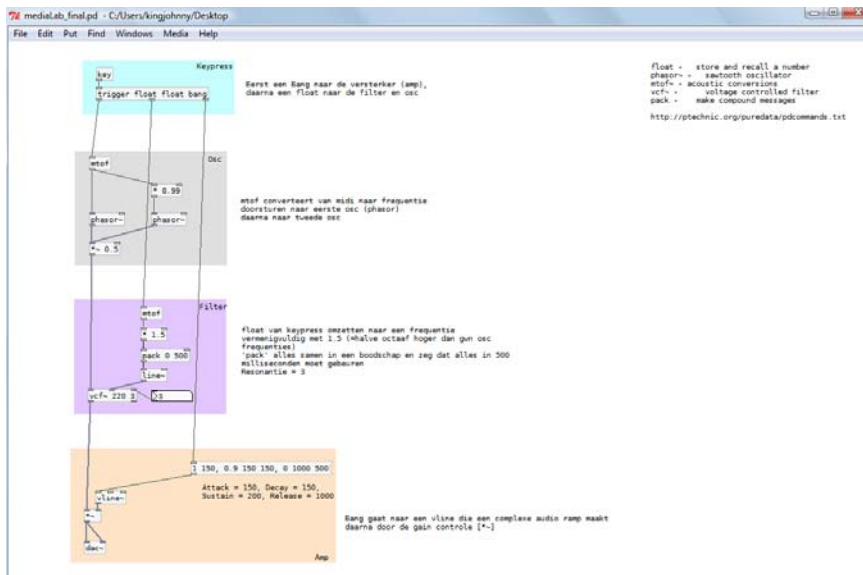


Esc en Espace 1

Helpfunctie Pd 1



Waveforms 1



Pdfinal 1

Ps/2 TOETSENBORDEN



Omdat ik dit project wil gebruiken voor mijn thesis (of toch als startpunt) besloot ik het als een afzonderlijke 'installatie' te fabriceren. Daarvoor had ik een extern toetsenbord nodig dat ik thuis nog wel had liggen. Maar ik wist niet waar ik moest beginnen, uiteindelijk vond ik een online pdf: "Practical Arduino- Cool Projects for Open Source Hardware (Learn in-depth arduino techniques using real world projects)" waarin stond aangeduidt hoe je een ps/2 toetsenbord en muis moet aansluiten op een prototype shield en dat verbinden met je Arduino.

Hiervoor had je ook een ps/2 verlengstuk nodig (of een 6pin mini Din socket) en een heleboel draadjes.



De uiteindelijke opstelling

ARDUINO



```
sketch_mar04a | Arduino 0017
File Edit Sketch Tools Help
sketch_mar04a ps2keyboard.h
#include <PS2Keyboard.h>
#define DATA_PIN 4

PS2Keyboard keyboard;

void setup()
{
  keyboard.begin(DATA_PIN);
  Serial.begin(9600);
  Serial.println("Initialized PS/2 reader");
  delay(1000);
}

void loop()
{
  if(keyboard.available()) {
    byte data = keyboard.read();
    if(data == PS2_KC_BREAK) {
      Serial.println("BREAK");
    } else if(data == PS2_KC_ENTER) {
      Serial.println("");
    } else if(data == PS2_KC_ESC) {
      Serial.println("ESC");
    } else if(data == PS2_KC_FPLUS) {
      Serial.println("F+");
    } else if(data == PS2_KC_FMINUS) {
      Serial.println("F-");
    } else if(data == PS2_KC_MULTI) {
      Serial.println("M+");
    } else if(data == PS2_KC_NUM) {
      Serial.println("NUMLOCK");
    } else if(data == PS2_KC_BESP) {
      Serial.println("BACKSPACE");
    } else {
      Serial.println(data, HEX);
    }
  }
}
```

De laatste stap was het aansluiten van het klavier op de Arduino en een programma schrijven die het klavier zou herkennen en zodoende ook de data doorsturen naar de PC, om zo deze data te verwerken in Pd. Hiervoor maak ik gebruik van Pduino, om de communicatie te laten verlopen tussen mijn Arduino en Pd.

Ook waren er libraries “PS2keyboard.h” nodig voor het toetsenbord, zodat de Arduino het vreemde toestel zeker zou herkennen.

PROBLEMEN:

Het zou natuurlijk mooi meegenomen zijn als alles naar behoren werkt, maar jammer genoeg is de verbinding tussen het extern toetsenbord en de Arduino niet gelukt, waardoor ik het niet heb kunnen testen. Ik vermoed dat de rede licht bij een defect in het toetsenbord, ik heb jammer genoeg ook geen tussenstuk om na te gaan, of het degelijk aan het klavier ligt.



Waarom ik het klavier verdenk? Omdat ik stroom binnen krijg en als ik met een tussenstuk (ps/2 naar usb) de muis aansluit ondervind ik weinig tot geen problemen (na veel geprul).

Daarom heb ik nog geen verbinding kunnen maken tussen Arduino en Pd, noch heb ik kunnen testen of de signalen die het toetsenbord doorstuurt overeenkomen met diegene die mijn klavier van de laptop doorstuurt.

BRONNEN

Practical Arduino- Cool Projects for Open Source Hardware (Learn in-depth arduino techniques using real world projects); Jonathan Oxer en Hugh Blemings (2009)

Loadbang - Programming Electronic Music in Pd; Johannes Kreidler (2009)

<http://en.flossmanuals.net/PureData/Introduction>

<http://puredata.info/>