

ferramentas

e inspirações

para projetos

de DMIS

filipe calegario

29.04.2014



**Centro
de Informática**

U F P E



U F P E



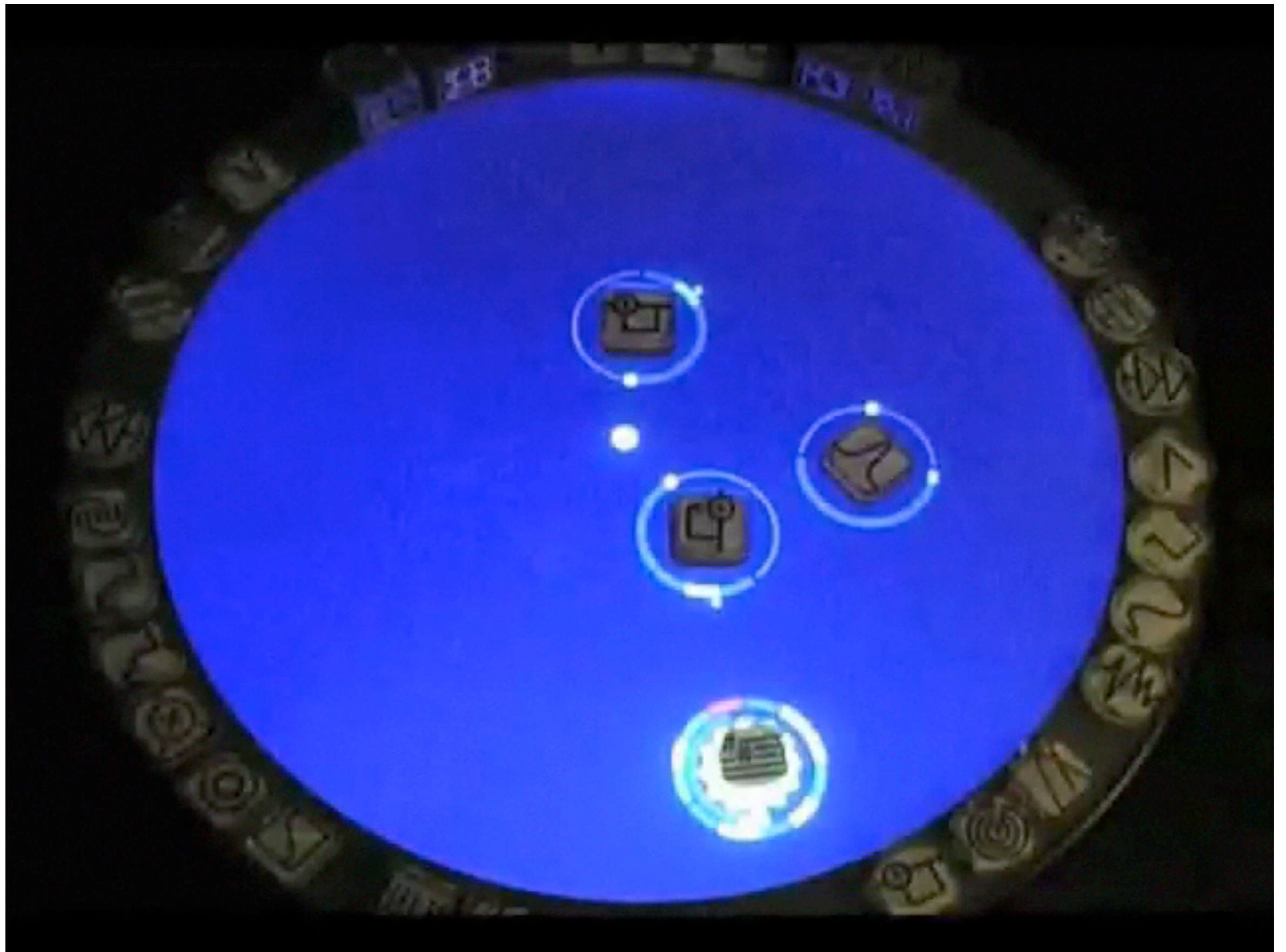
Graduação

Filipe Calegario

Akai Pro EWI USB

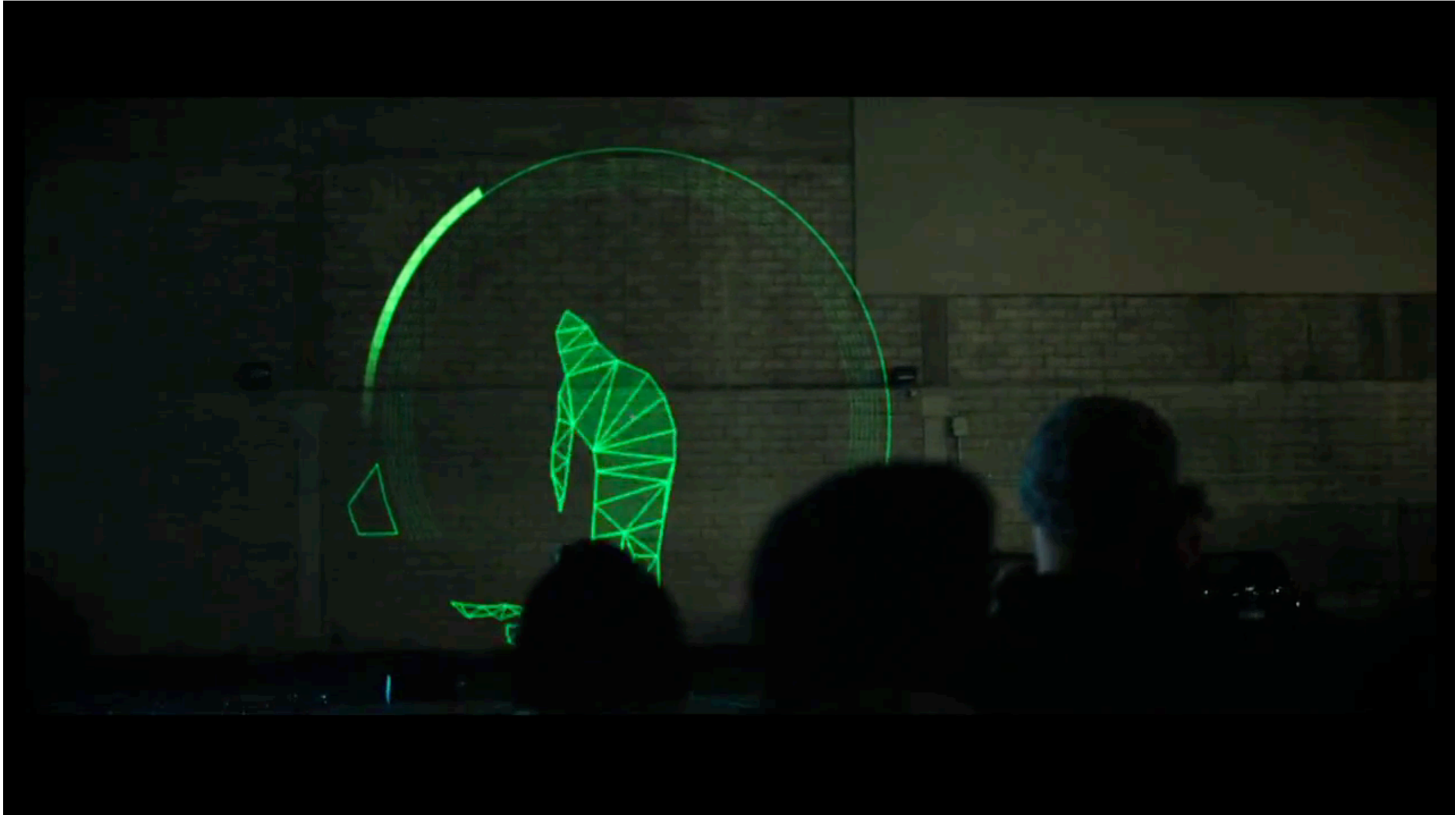


Akai Pro EWI USB



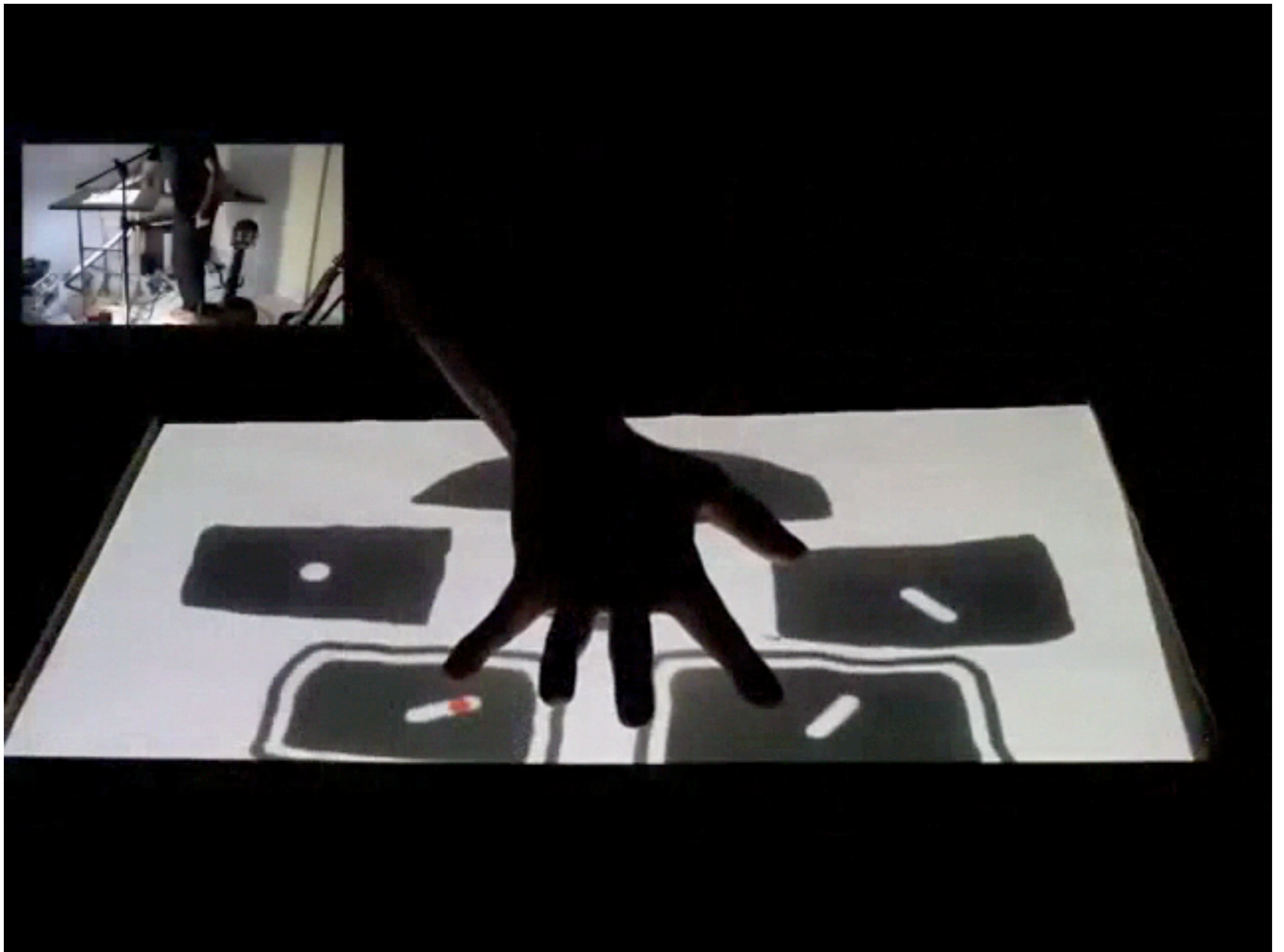
reactable

V Motion Project

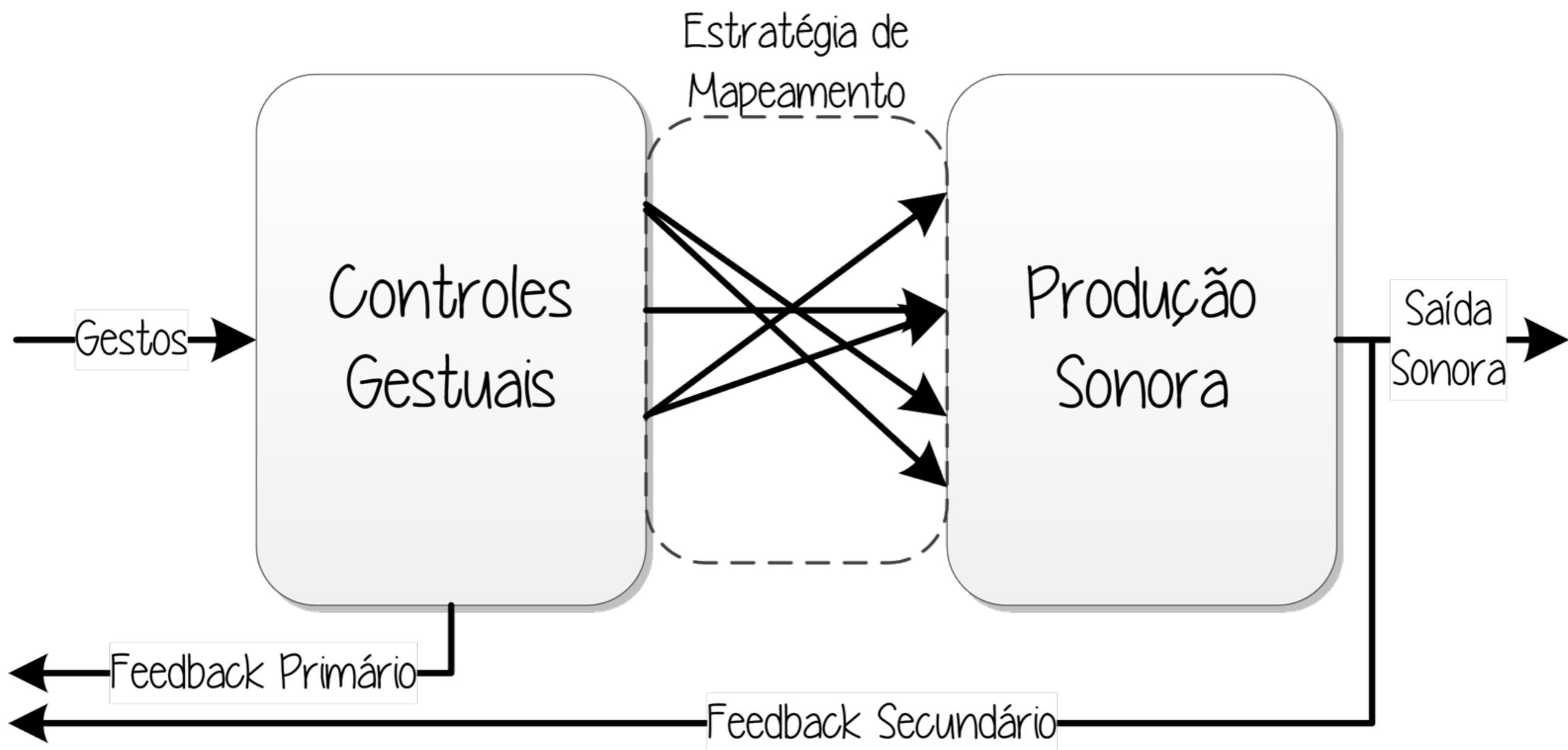


V Motion Project

illusio - jeraman



illusio - jeraman



instrumento musical digital

augmented flutes
(IRCAM, LMA, McGill flutes
hyper-flute)

synthophone

disklavier

meta-saxophone

meta-trumpet

IR violin

hyper-instruments

SBass

keyboards

Akai EWI

Yamaha WXS

sequential drum

VideoHarp

Accordiatron

SuperPalm

multiply-touch-
sensitive clavier

BoSSA

Hands

Continuum

Radio Baton

Buchla Lightning

Gyrotyre

SillyTone Squish

Data Glove

Lady's Glove

Yamaha Miburi

Rolky Asproyd

Matrix

Lemur

Instrumentos
Acústicos
Aumentados

Simuladores de
Instrumentos
Acústicos

Instrumentos
inspirados em
Instrumentos
Acústicos

Instrumentos
Alternativos

classificação DMI

componentes e ferramentas

projetos e inspirações

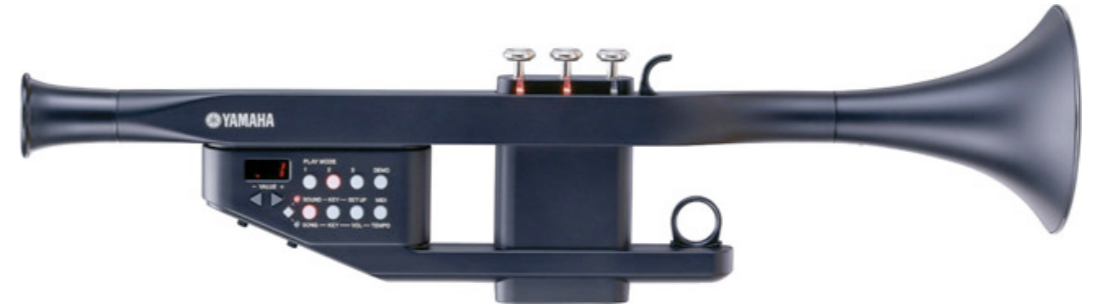
pesquisa

conceitos e referências

componentes

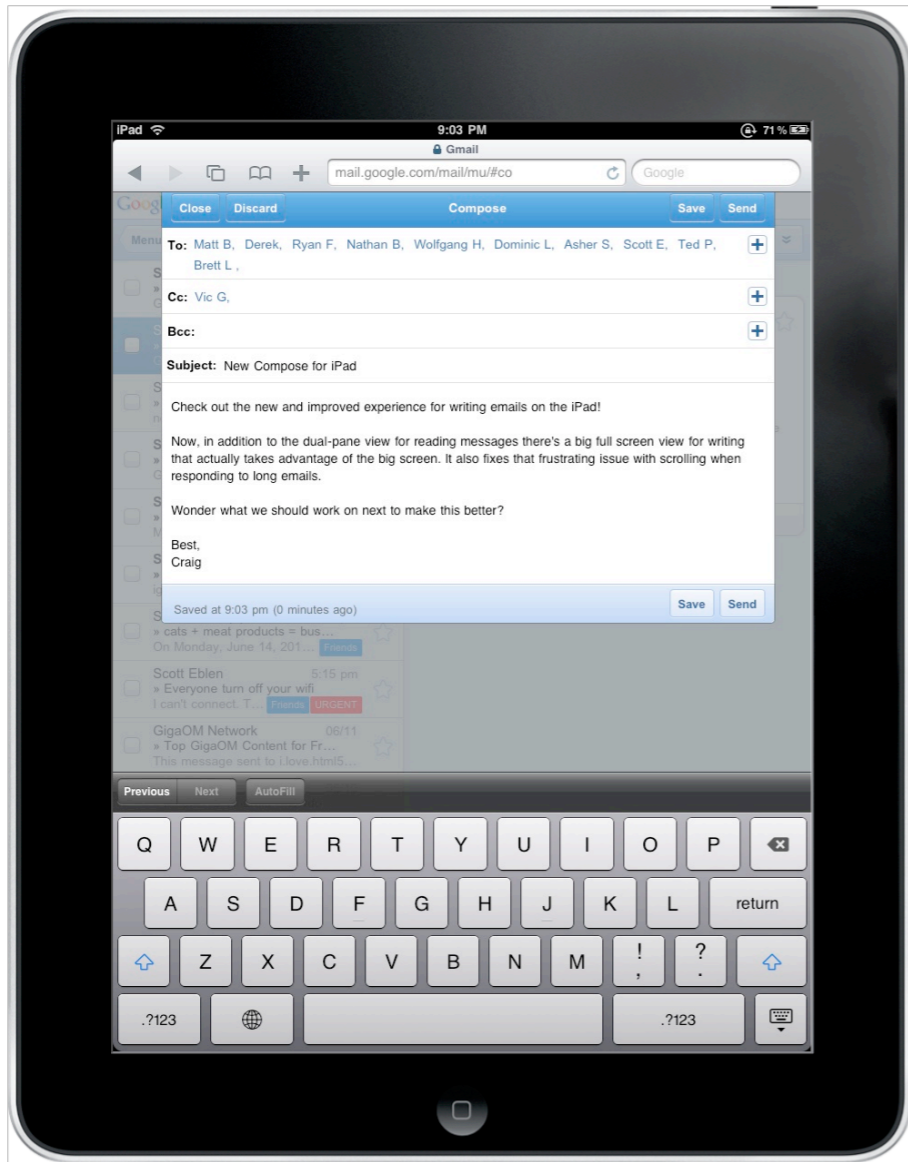
ferramentas

controladores MIDI



novas formas de interação

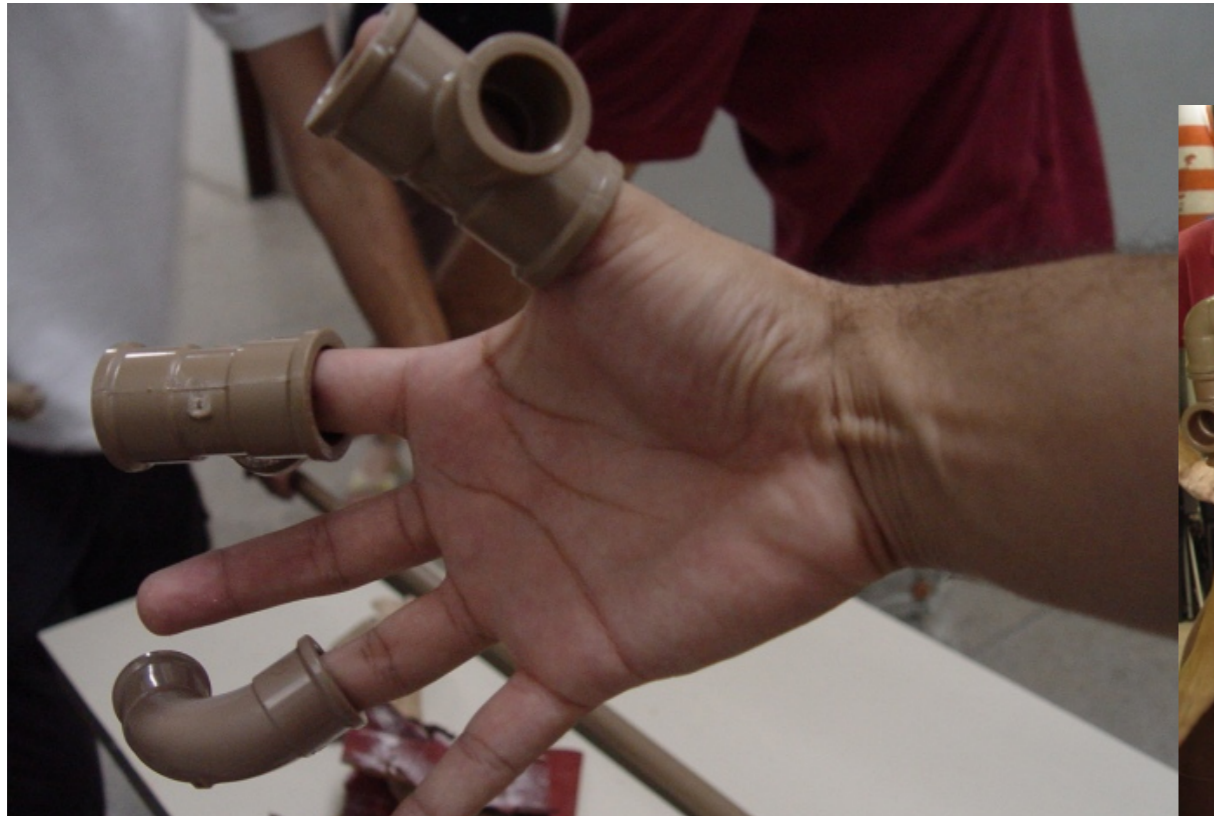


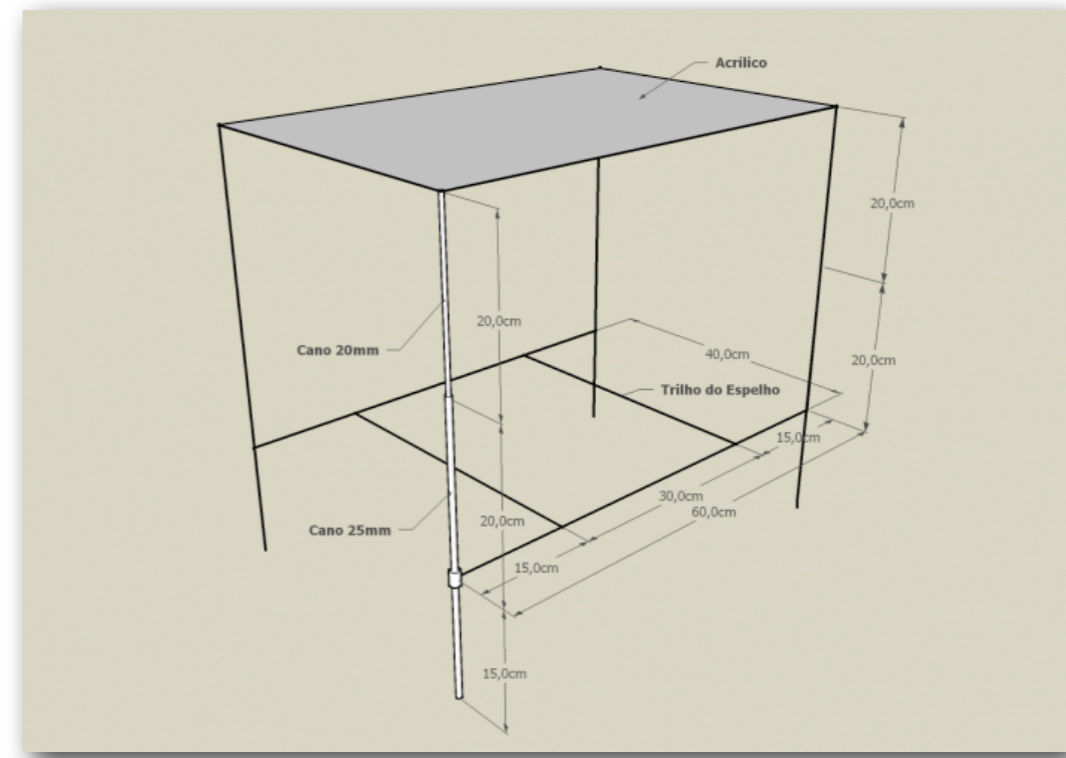
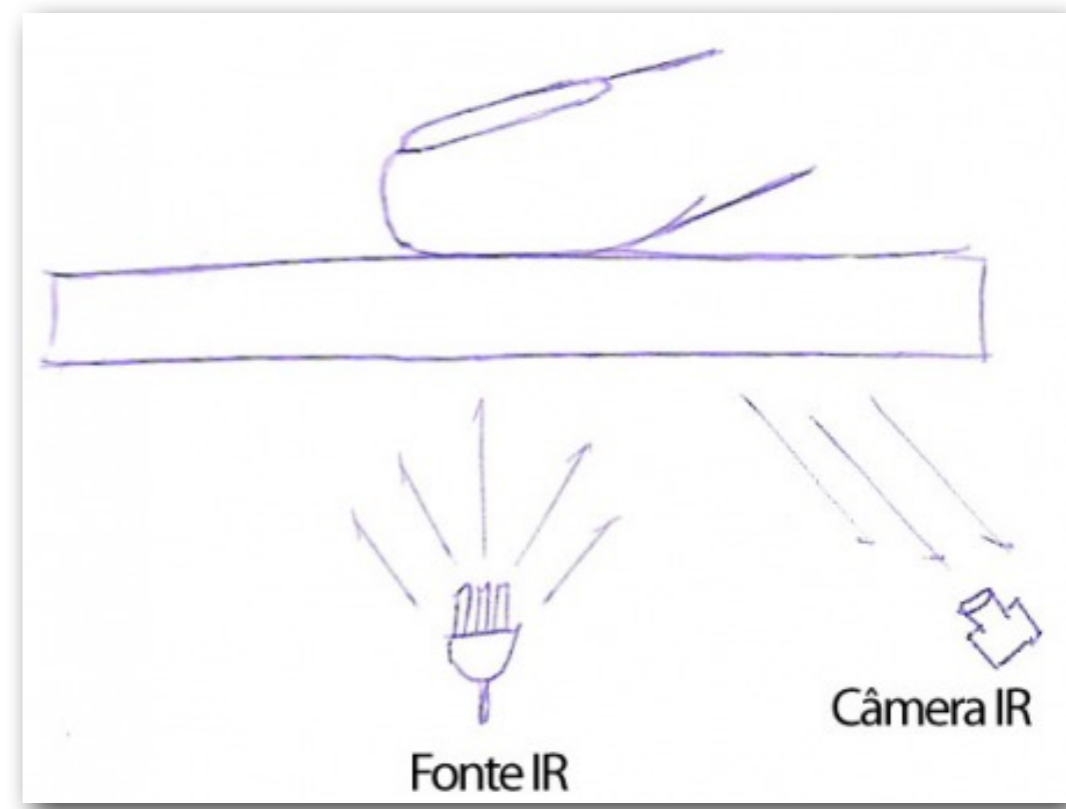
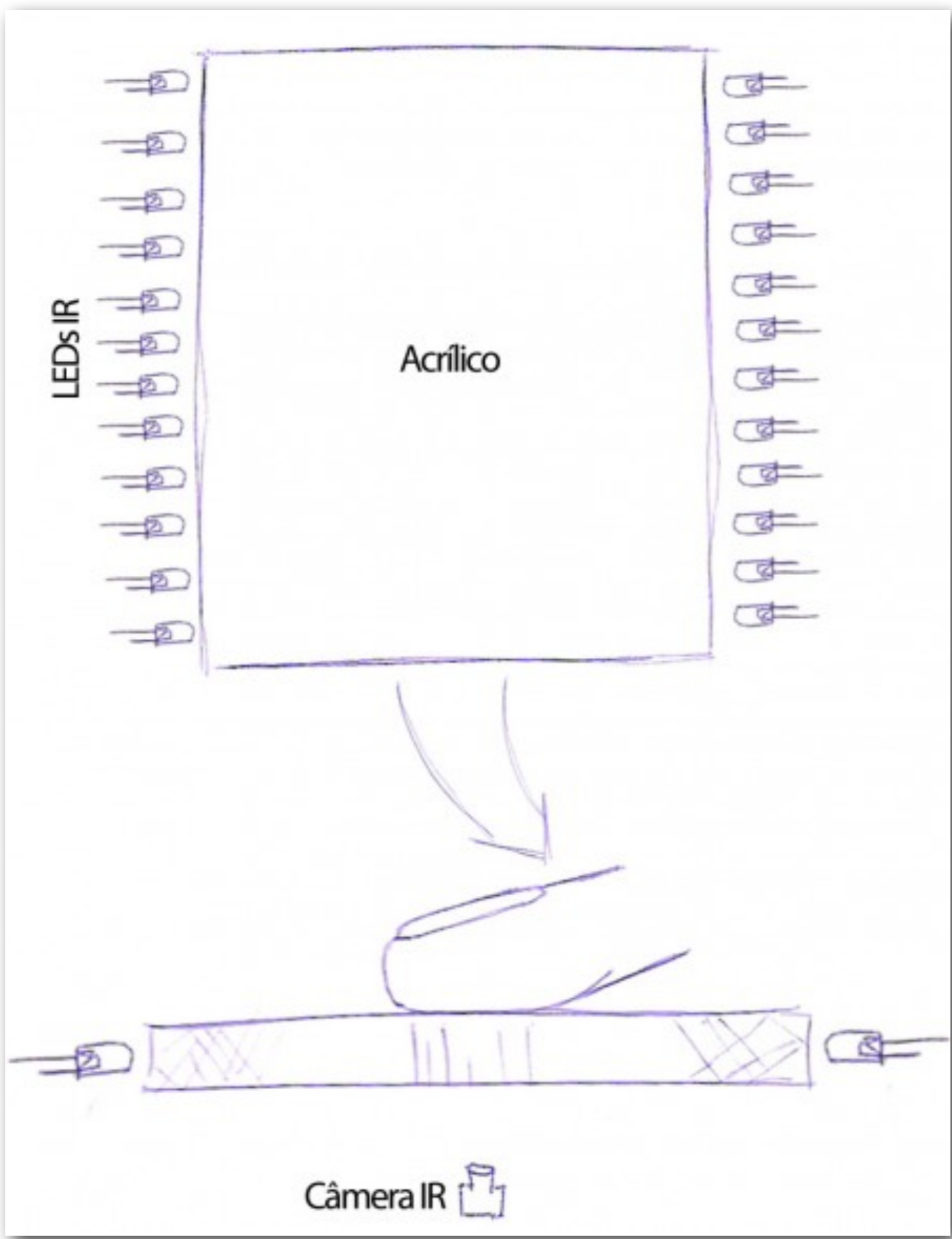


M.E.S.A.



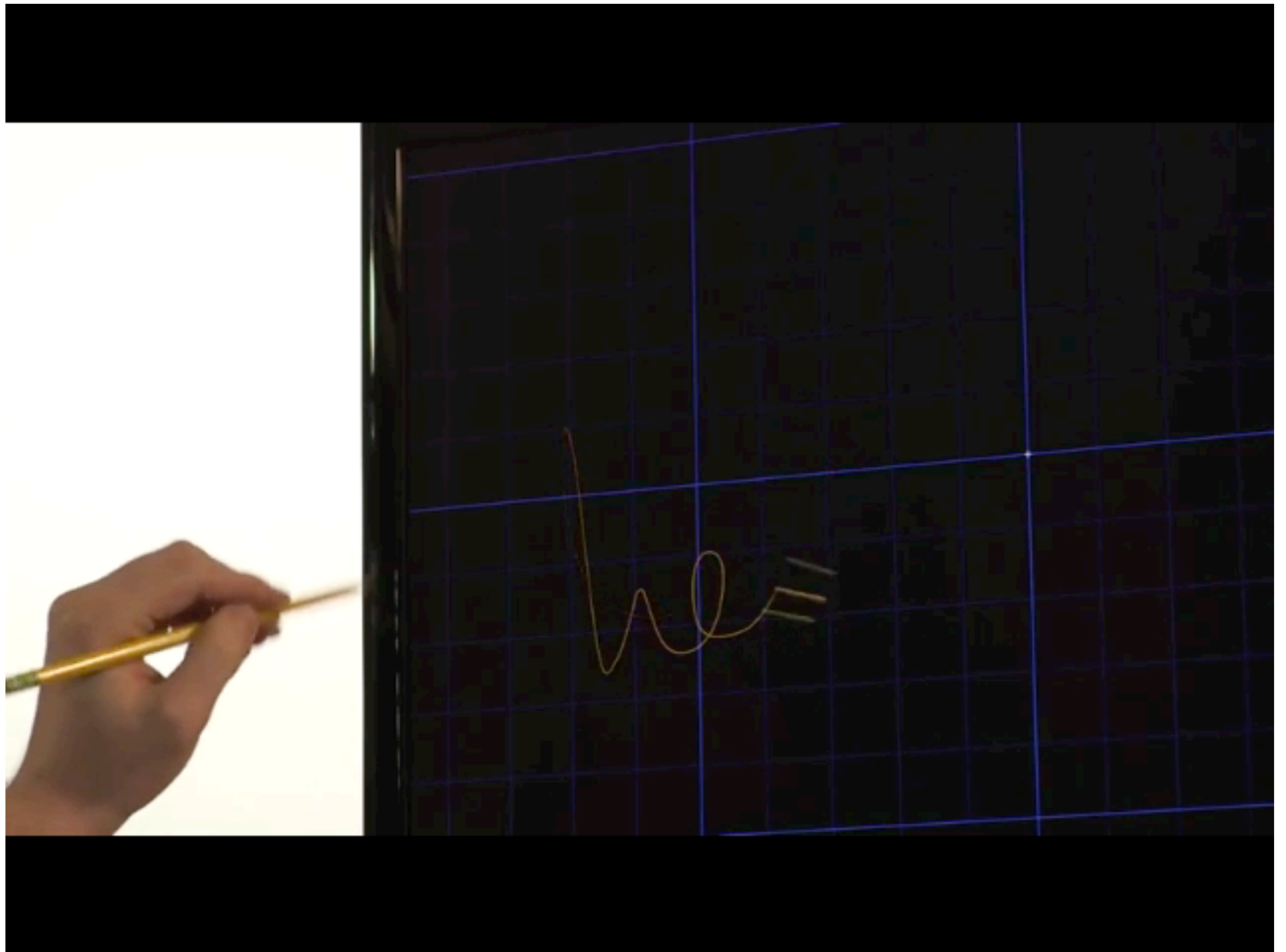
multitâtxi eintxertxeimentu sârfeice apliqueixõns







multitâtxi eintxertxeimenti sârfeice apliqueixôns

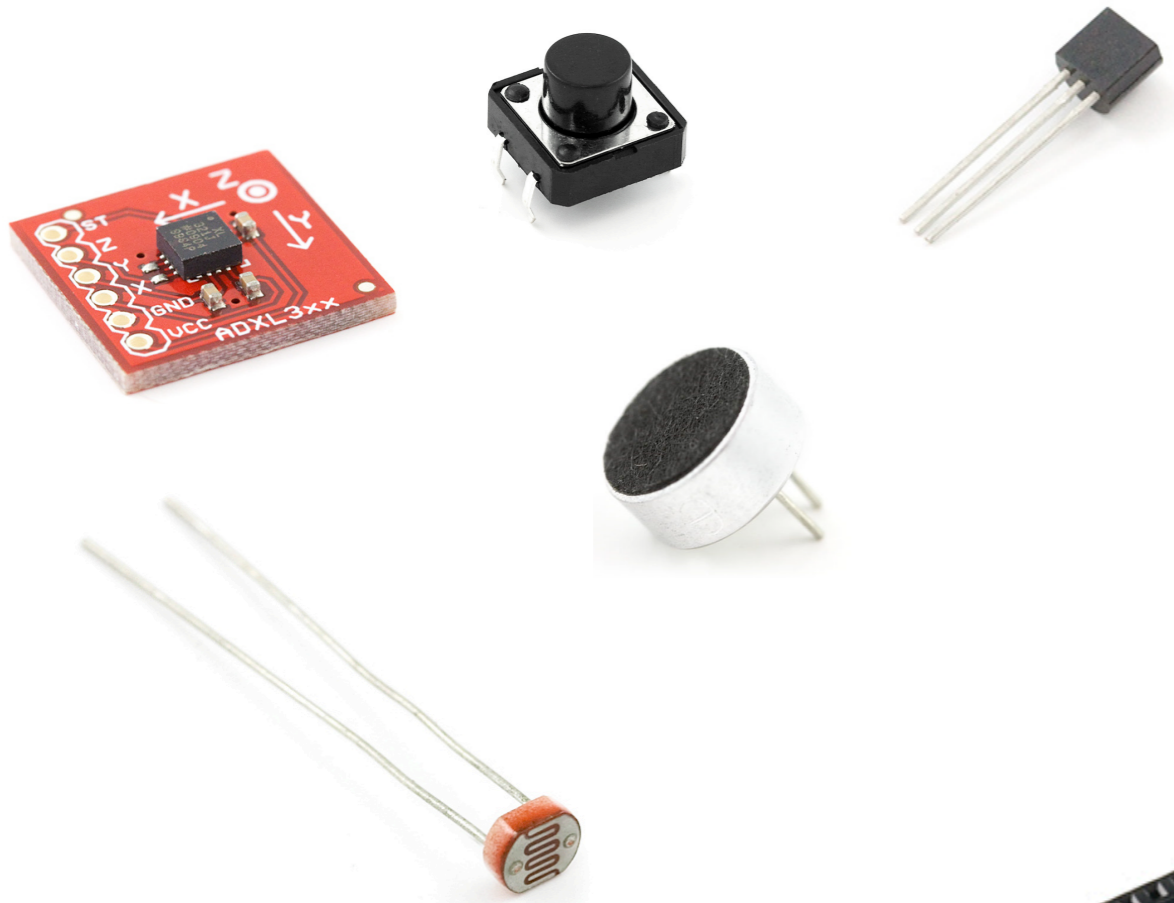


leap motion



myo

arduino



```
BlinkSimples | Arduino 0021
BlinkSimples

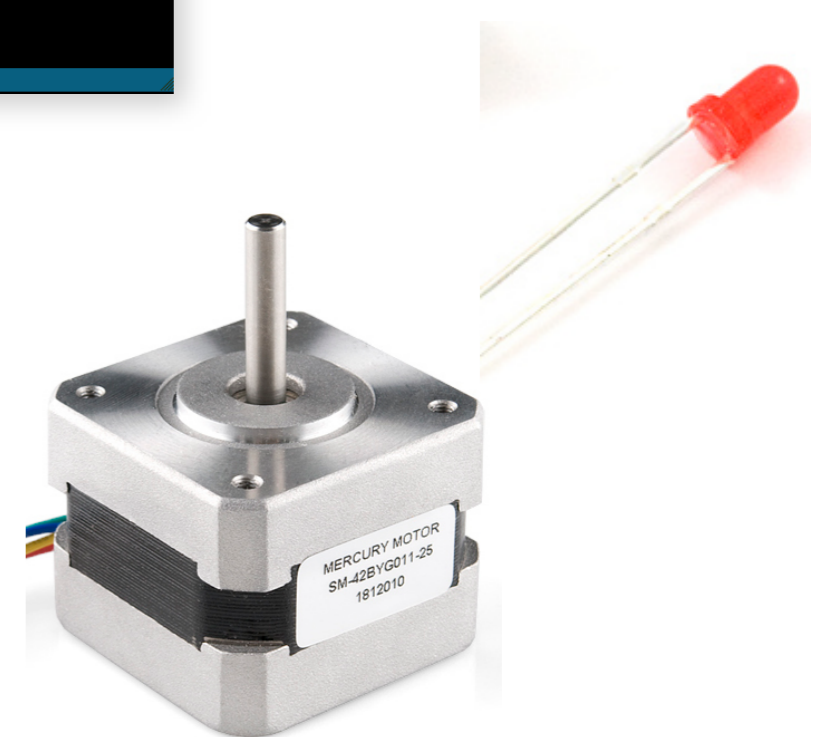
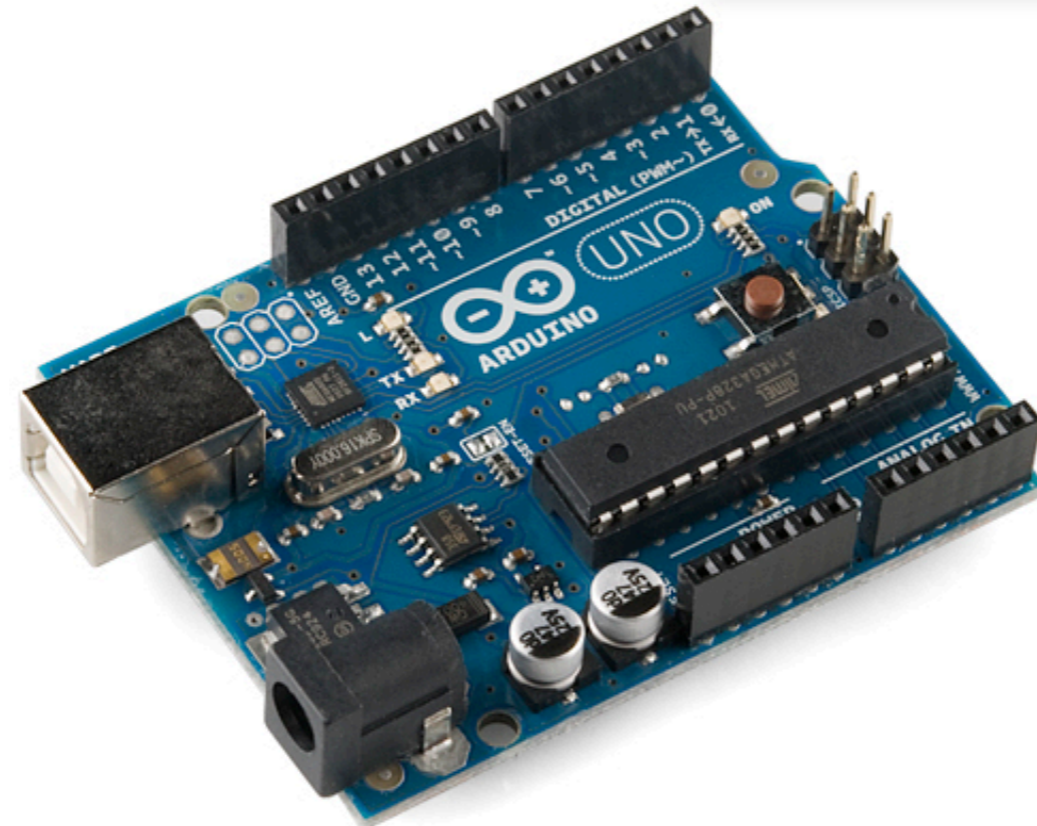
void setup() {
  // initialize the digital pin as an output:
  pinMode(ledPin, OUTPUT);
  pinMode(potPin, INPUT);
}

// the loop() method runs over and over again,
// as long as the Arduino has power

void loop()
{
  int potValor = analogRead(potPin);
  digitalWrite(ledPin, HIGH); // set the LED on
  delay(potValor);           // wait for a second
  digitalWrite(ledPin, LOW); // set the LED off
  delay(potValor);         // wait for a second
}
```

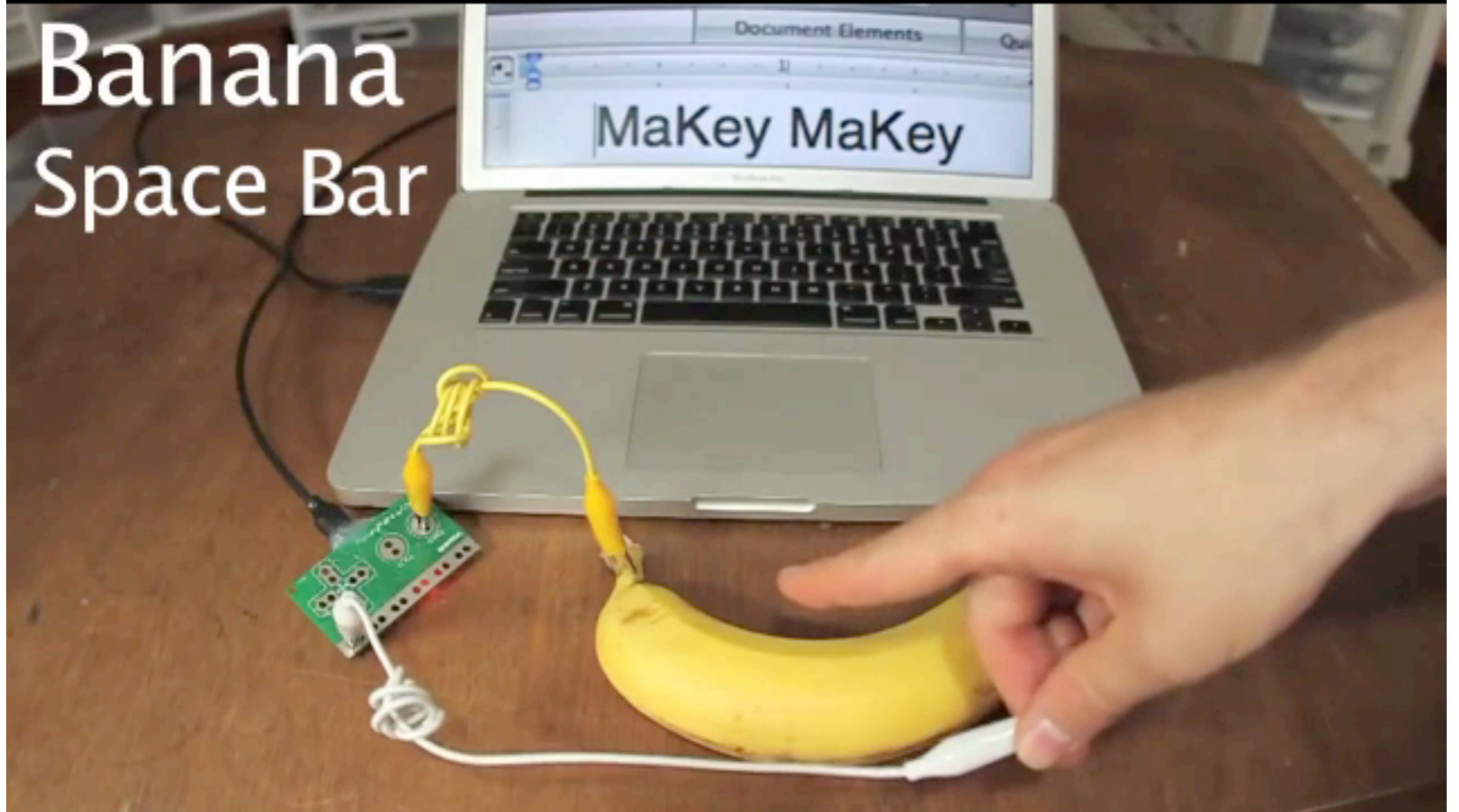


sensores

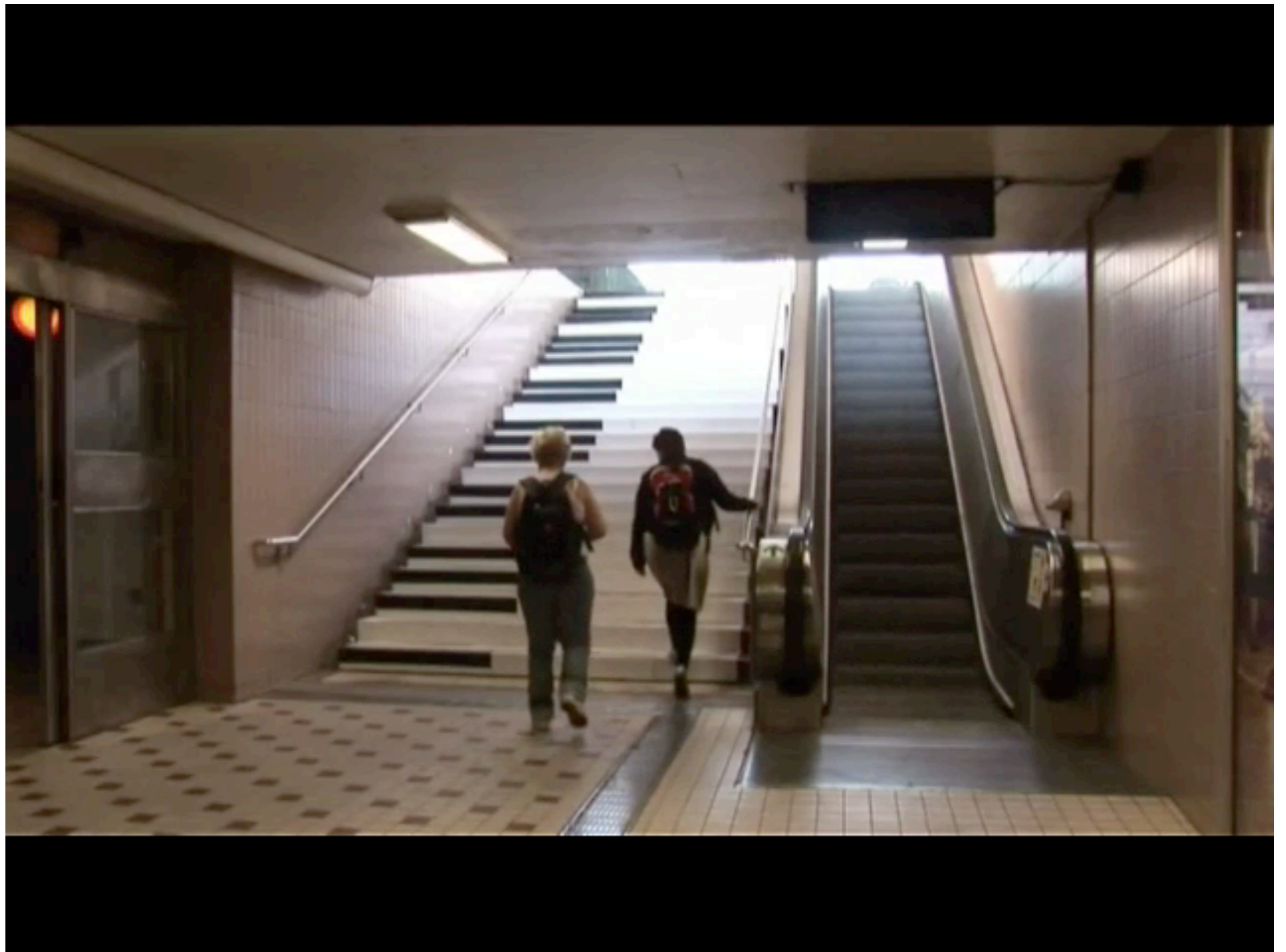


atuadores

Banana Space Bar



makey makey

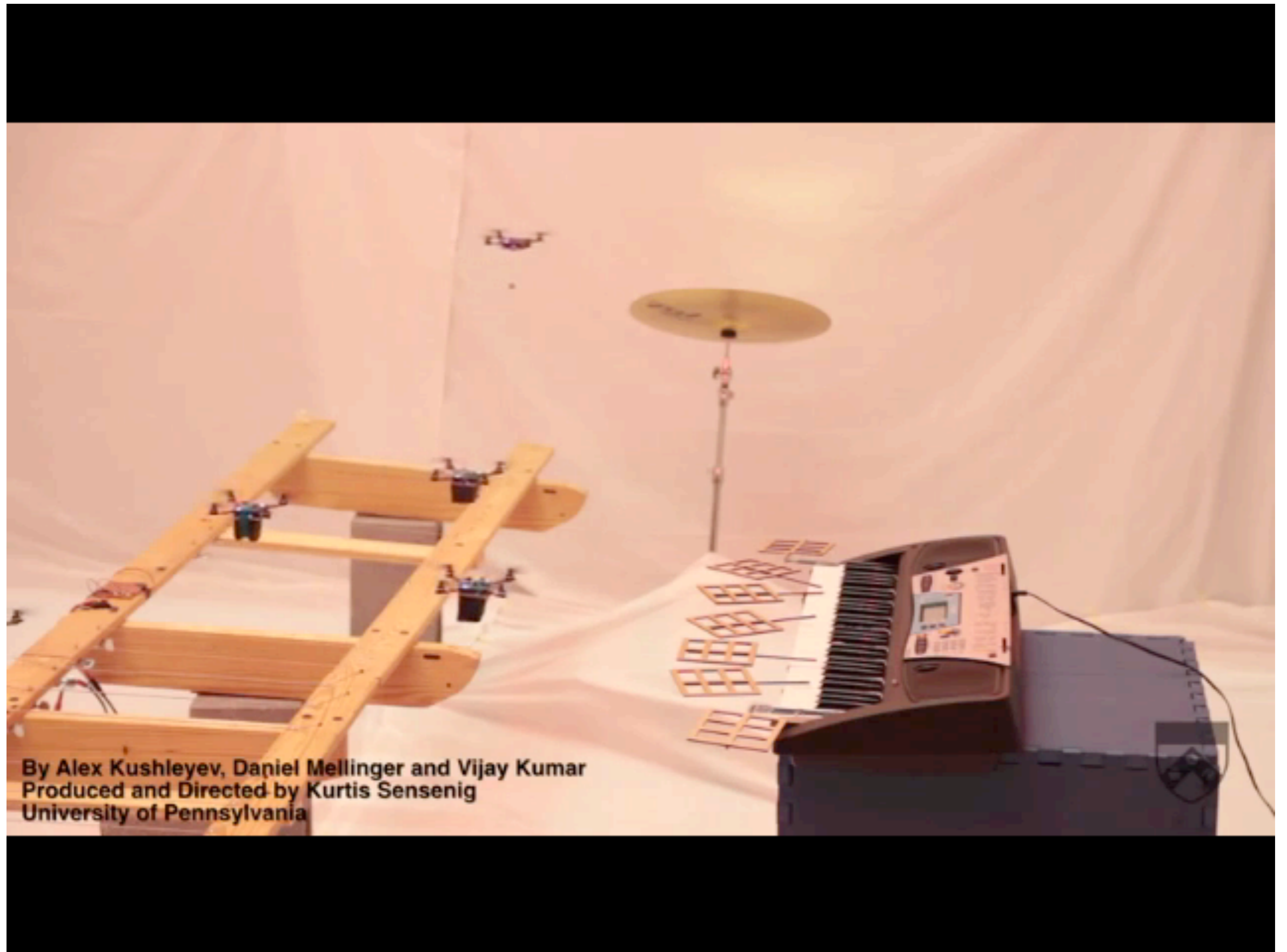


piano stairs - thefuntheory.com

the world's deepest bin - thefuntheory.com



the world's deepest bin - thefuntheory.com

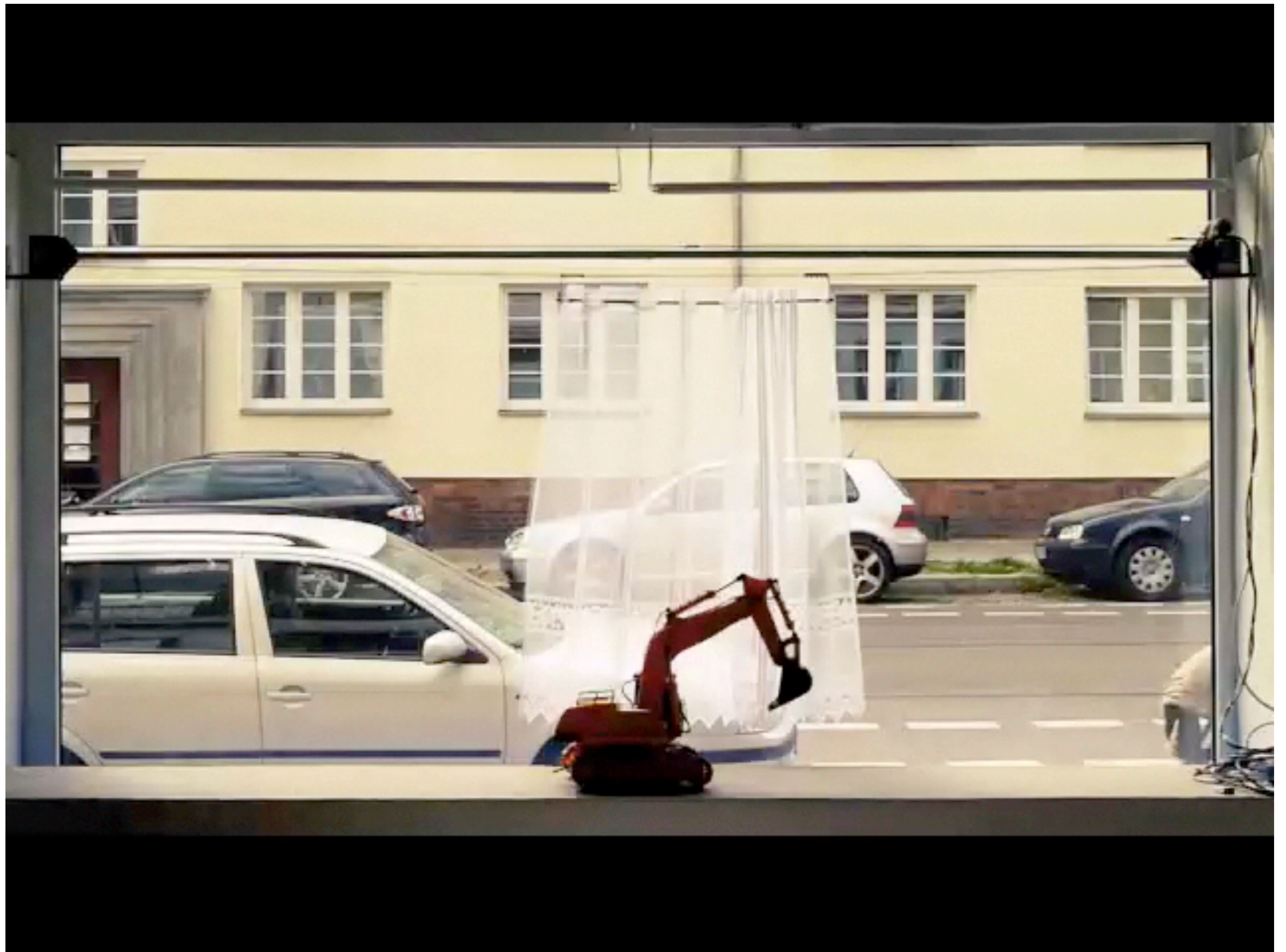


By Alex Kushleyev, Daniel Mellinger and Vijay Kumar
Produced and Directed by Kurtis Sensenig
University of Pennsylvania

robot quadrotors perform james bond theme

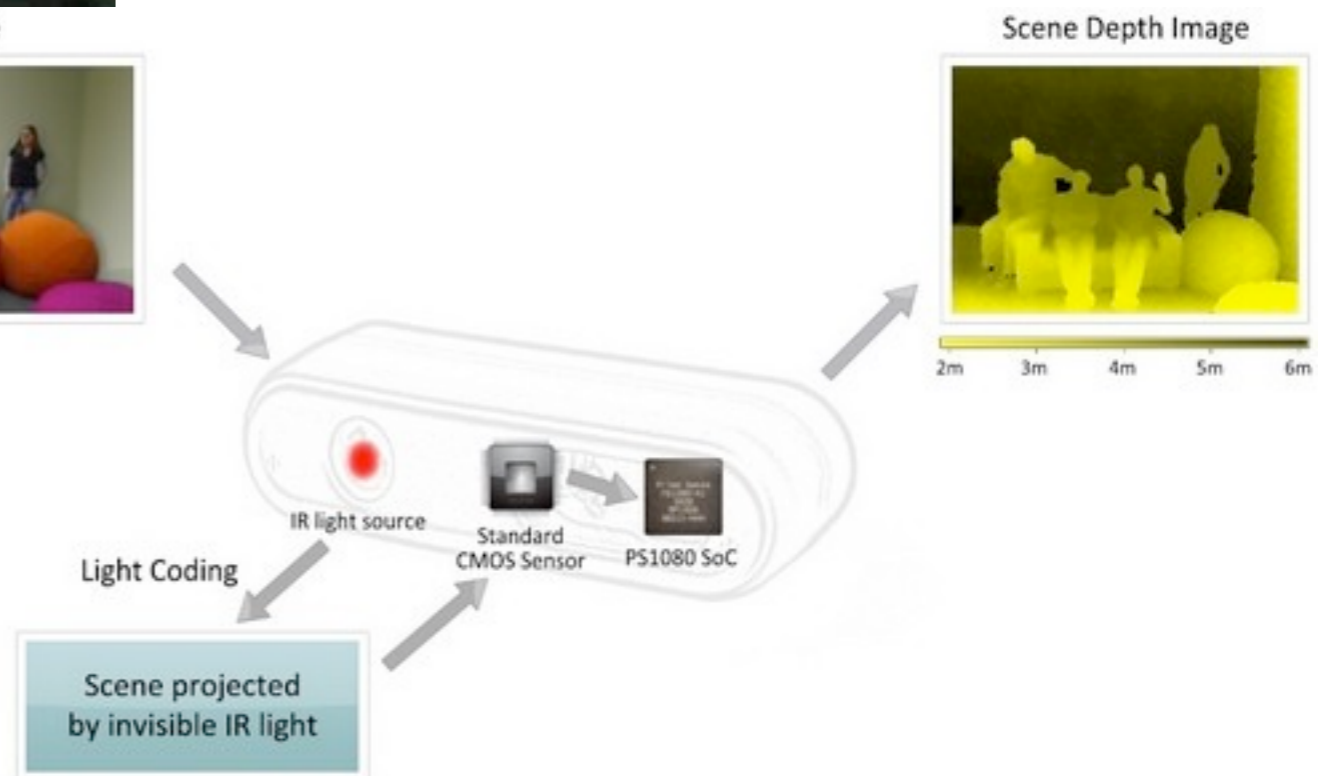
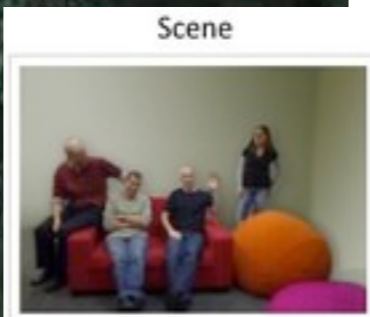
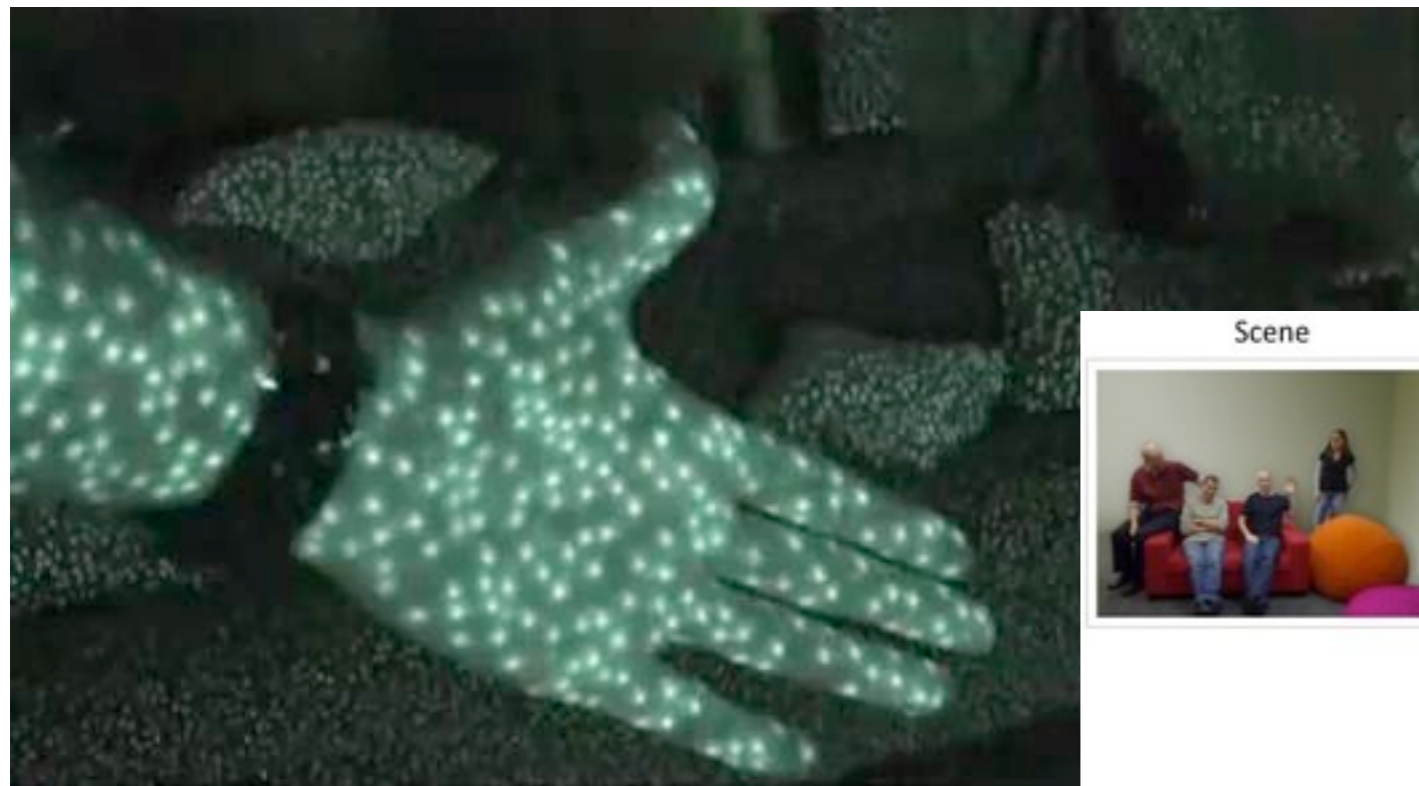


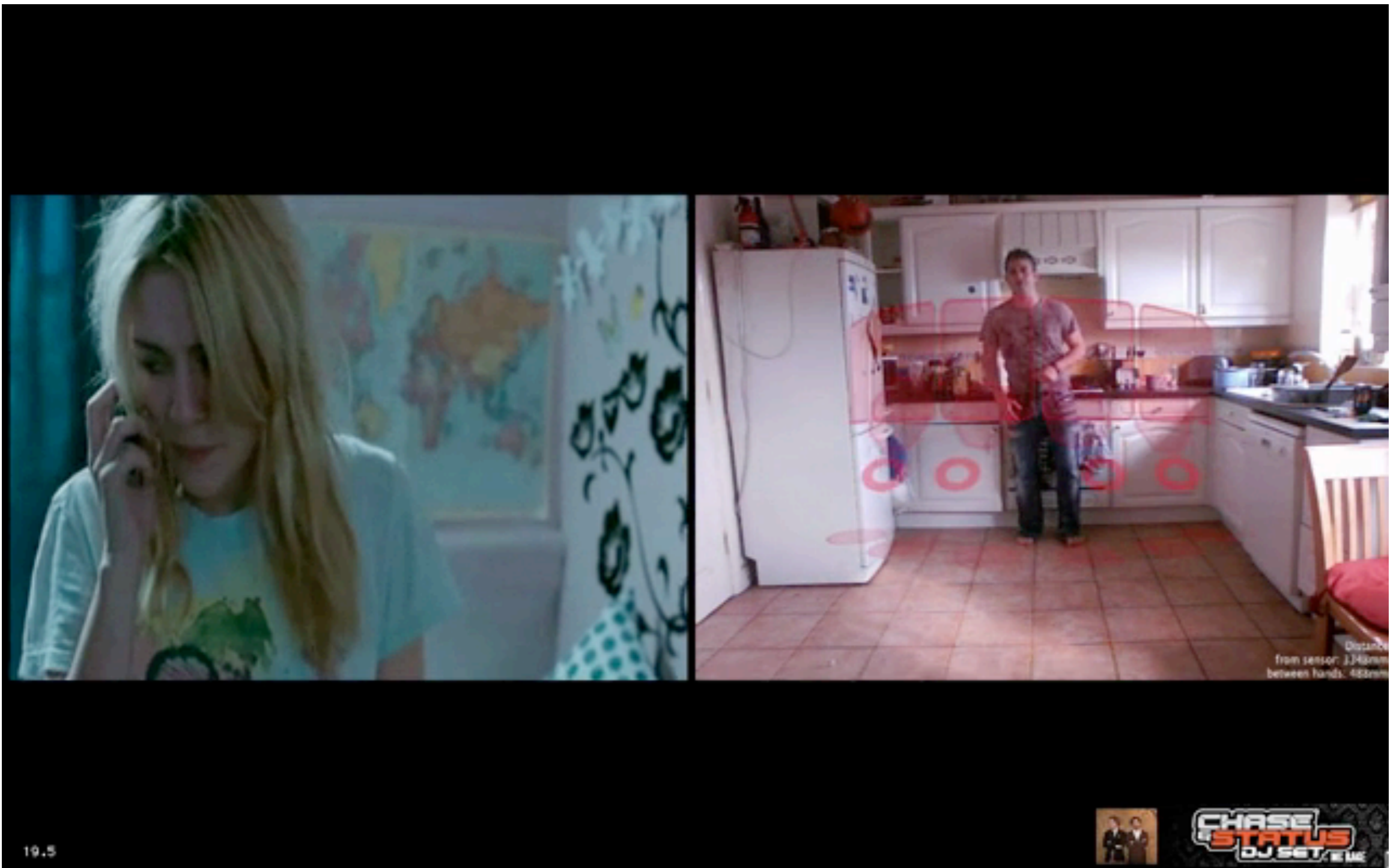
useless machine



my little piece of privacy - niklas roy

recife: the playable city - press play/toca aí





DJ Set

johannes
kreidler

kinect 3D
sensor studies

Johannes Kreidler

kinect 3D sensor studies

(2011)

kinect 3D
sensor studies

johannes
kreidler



Johannes Kreidler

kinect 3D sensor studies

(2011)

kinect 3D
sensor studies

johannes
kreidler

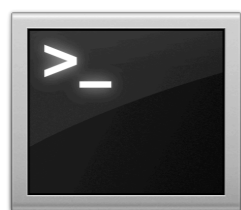




holofunk



só testando...



OSCeleton



/joint/r_hand/1 0.7 0.1 5.9



OSCulator

/parameter1 0.7 0.1



re.scalla



note on



note off



SimpleSynth

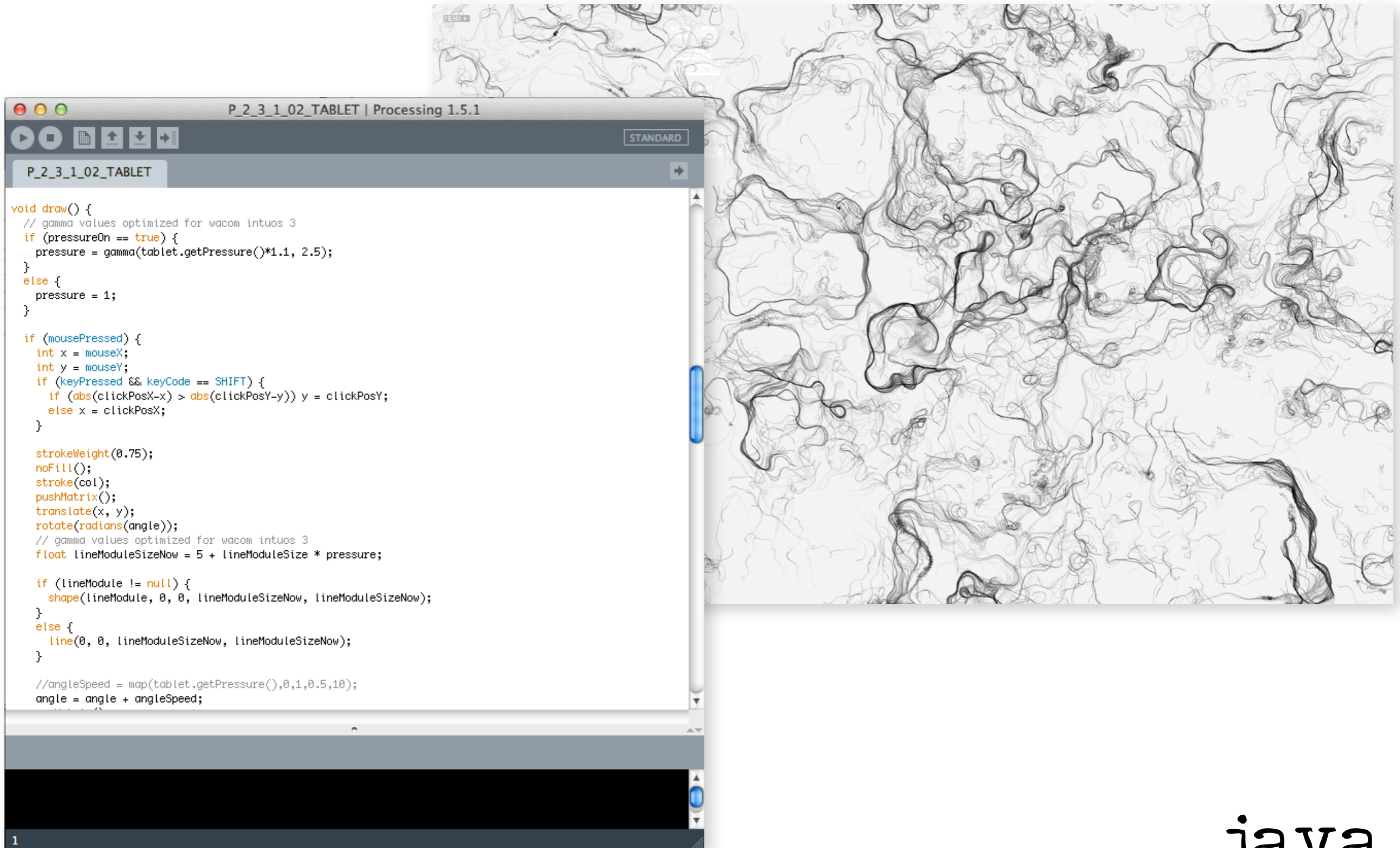
kimusik



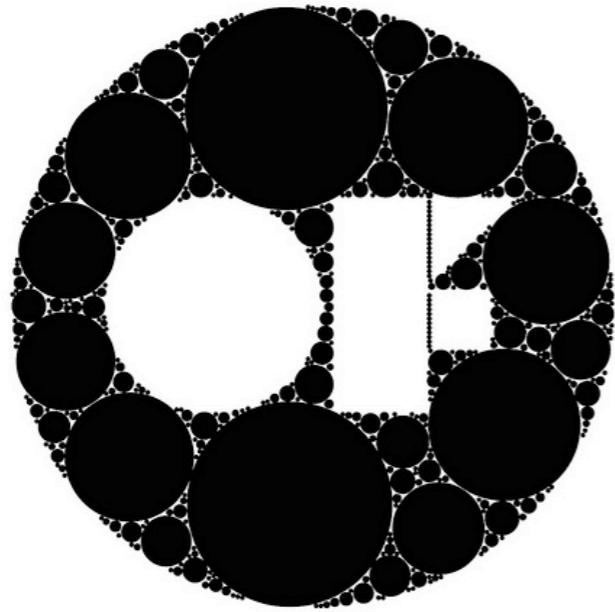
kimusik



processing



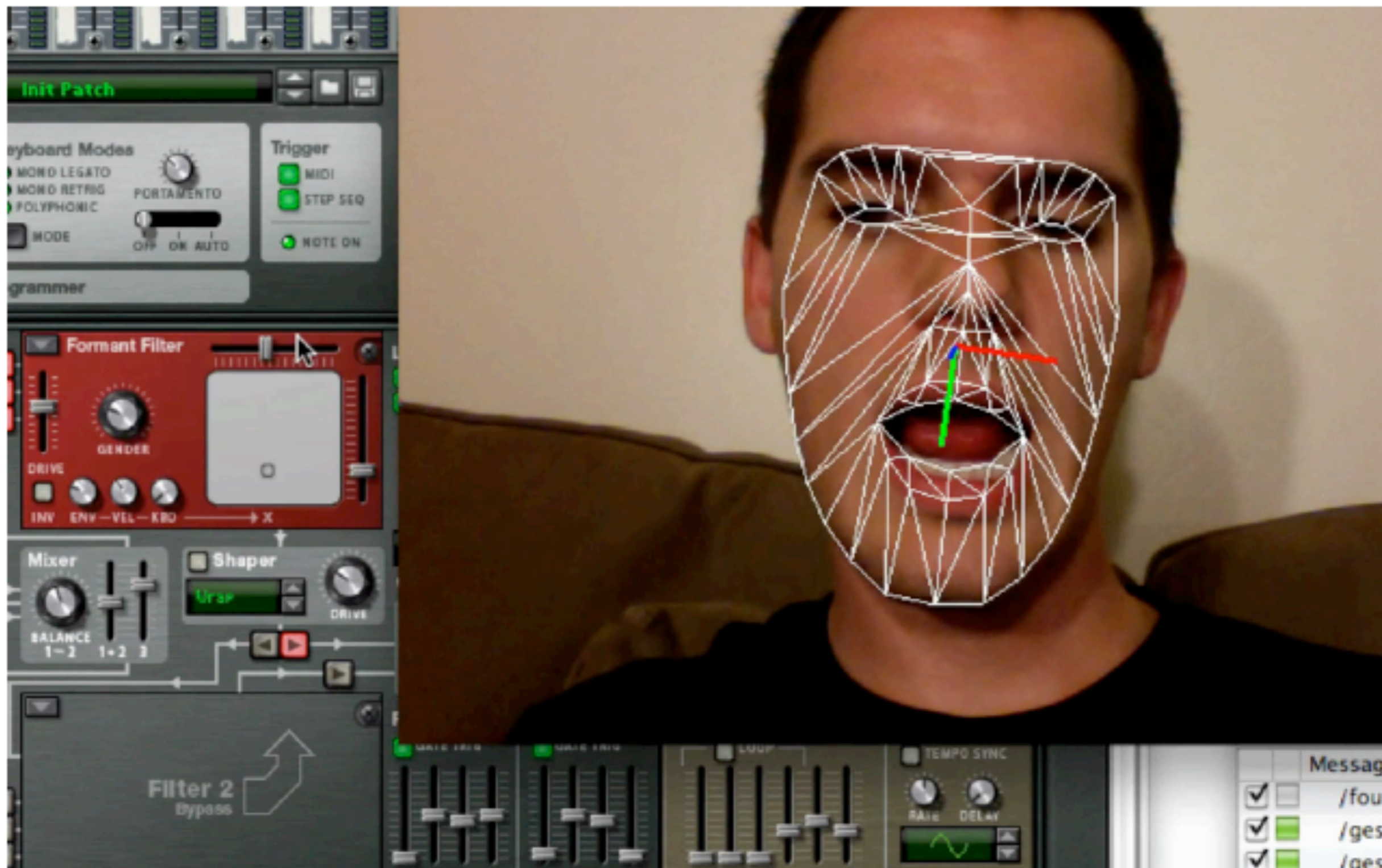
java



openframeworks

cinder





faceOSC

faceOSC



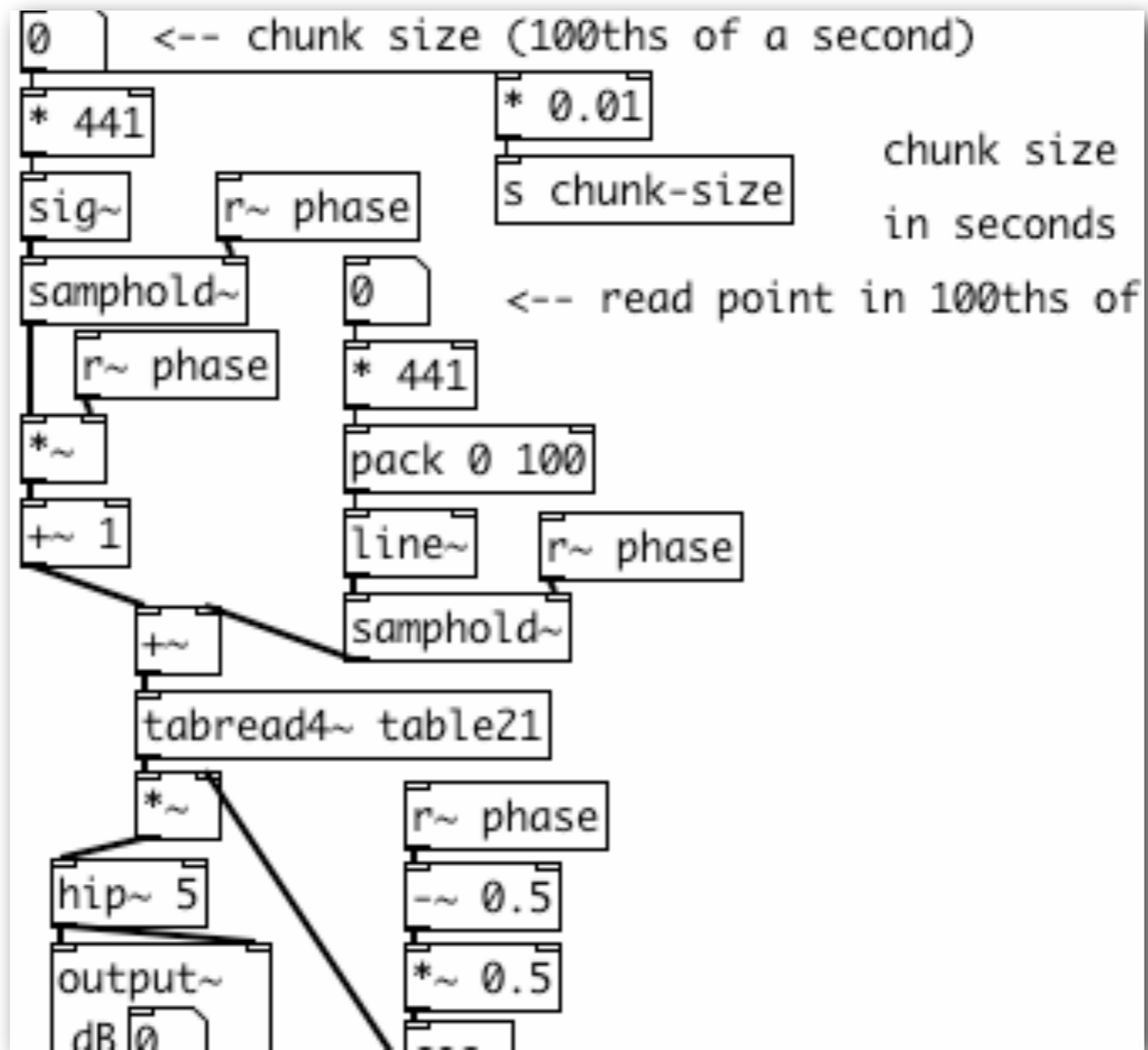
faceOSC

```
Player player = new Player();  
player.play("C");  
player.play("C7h");  
player.play("C5maj7w");  
player.play("G5h+B5h+C6q_D6q");  
player.play("G5q G5q F5q E5q D5h");  
player.play("T[Allegro] V0 I0 G6q A5q V1 A5q G6q");  
player.play("V0 Cmajw V1 I[Flute] G4q E4q C4q E4q");  
player.play("T120 V0 I[Piano] G5q G5q V9 [Hand_Clap]q Rq");
```

JFugue

```
Player player = new Player();  
player.play("C");  
player.play("C7h");  
player.play("C5maj7w");  
player.play("G5h+B5h+C6q_D6q");  
player.play("G5q G5q F5q E5q D5h");  
player.play("T[Allegro] V0 I0 G6q A5q V1 A5q G6q");  
player.play("V0 Cmajw V1 I[Flute] G4q E4q C4q E4q");  
player.play("T120 V0 I[Piano] G5q G5q V9 [Hand_Clap]q Rq");
```

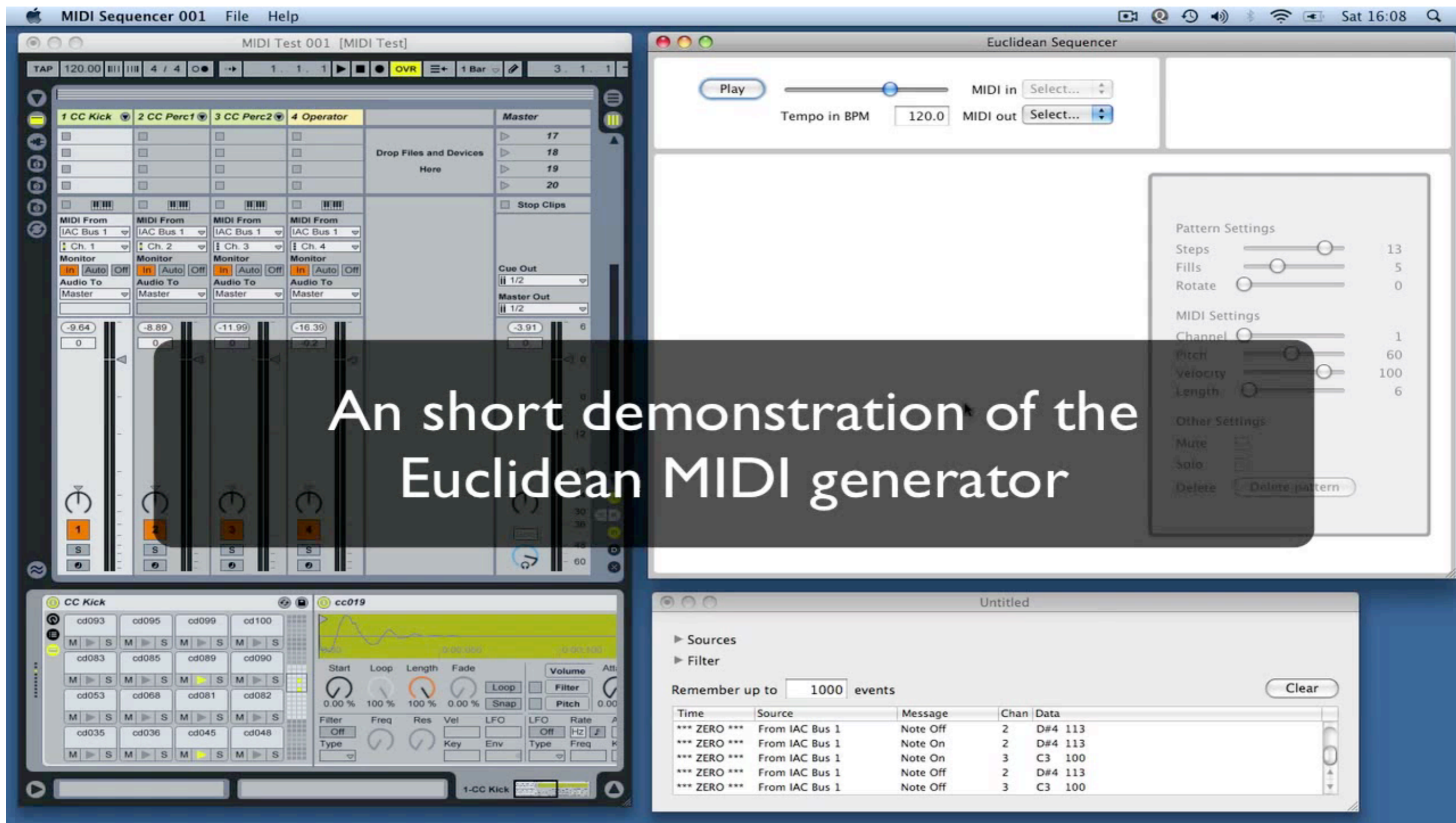
JFugue



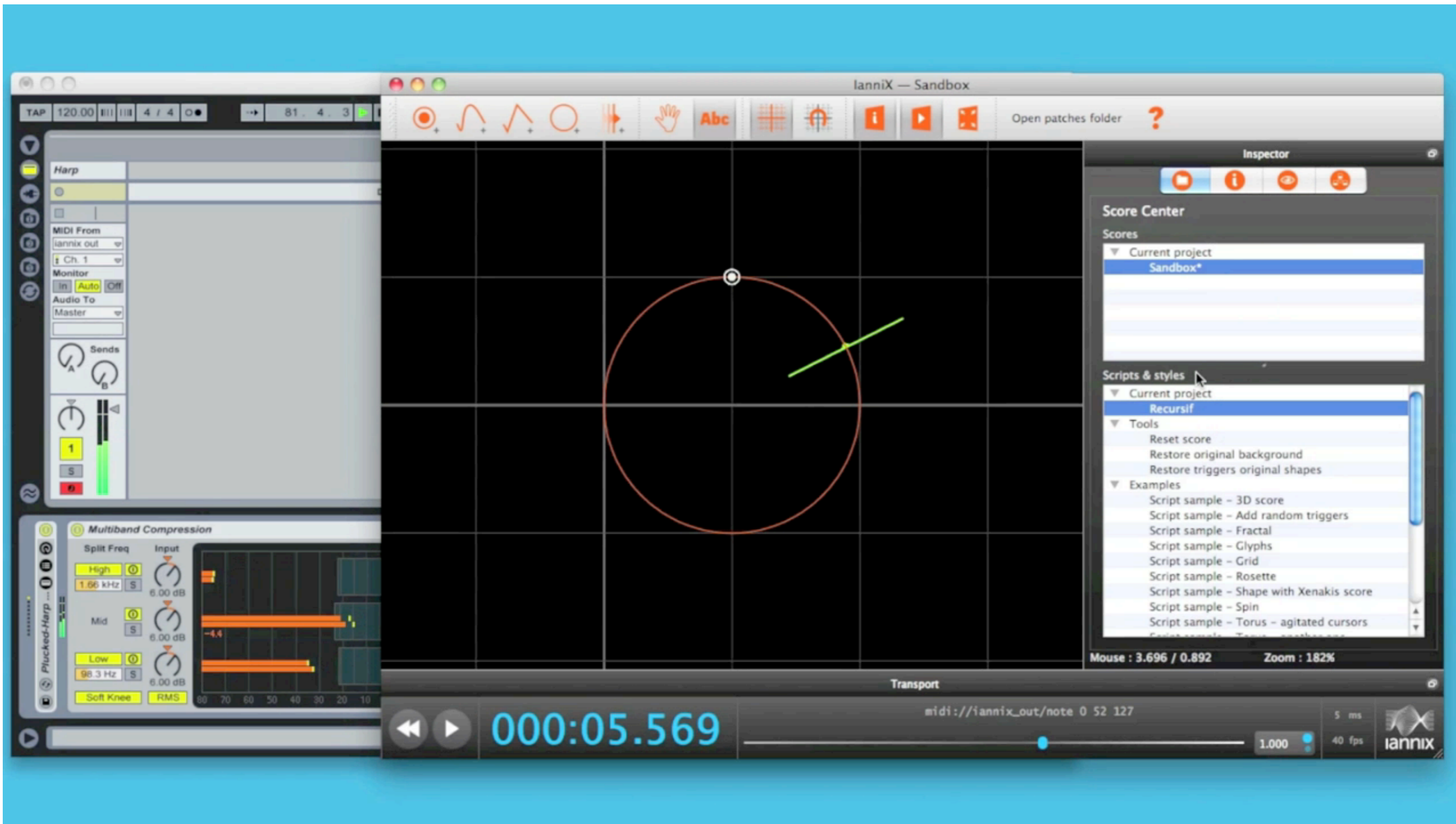
pure data



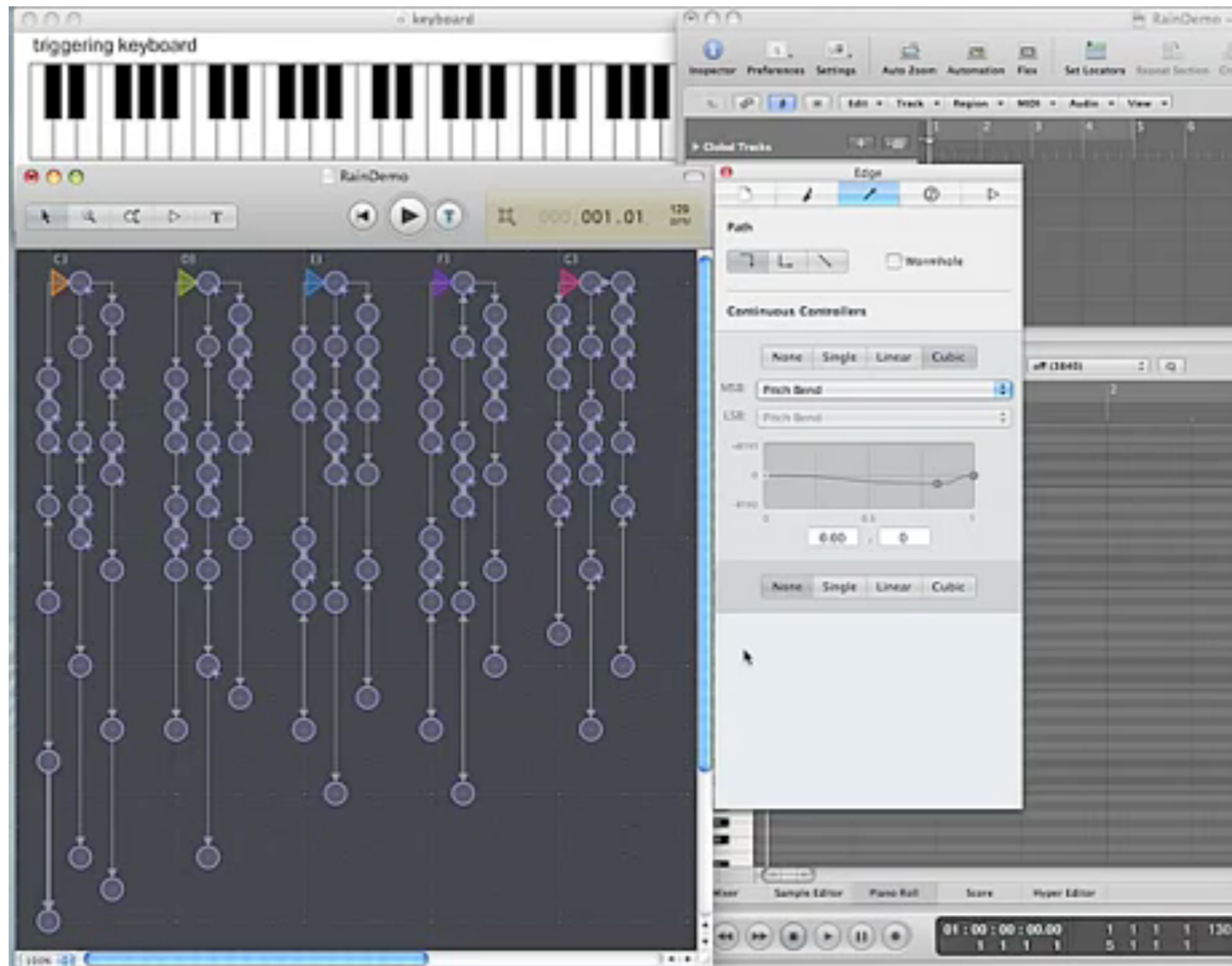
ableton live



euclidean MIDI patterns



iannix



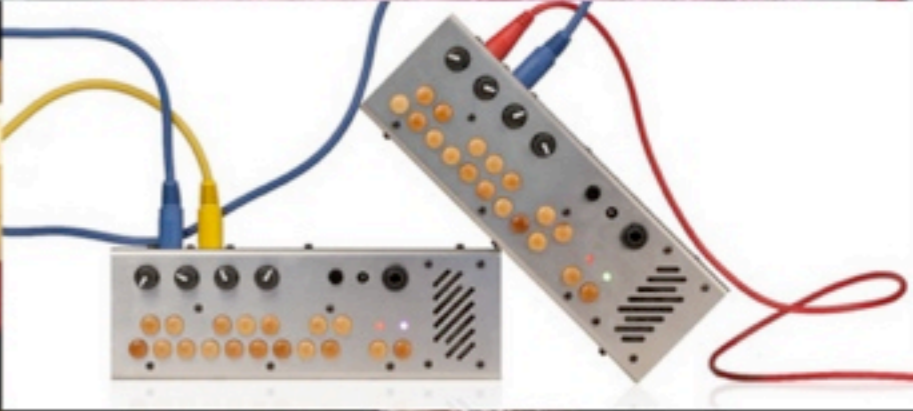
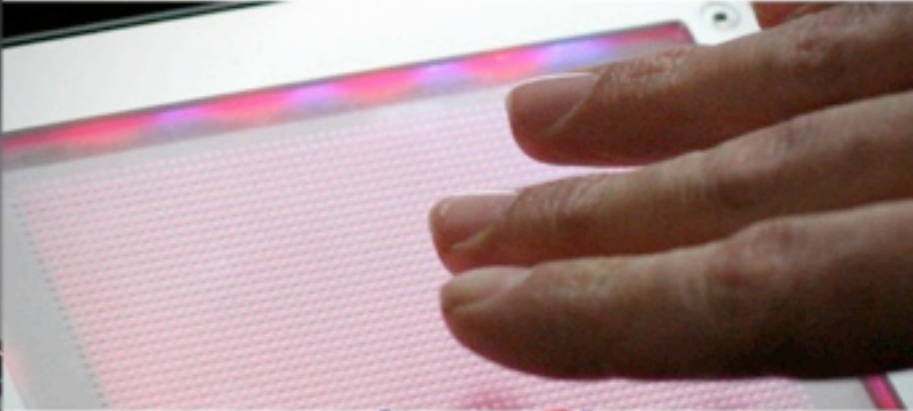
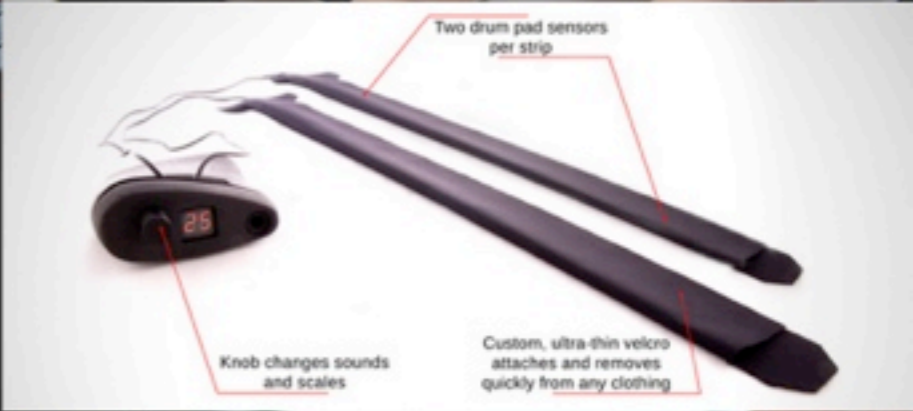
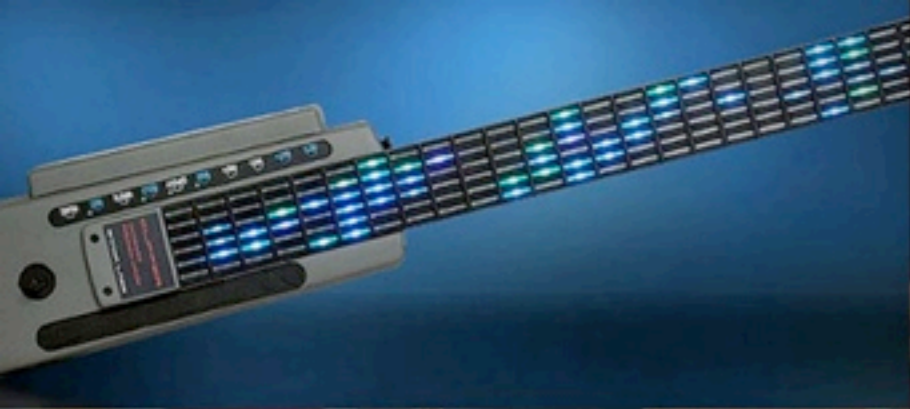
nodal

cartas de sensores e atuadores

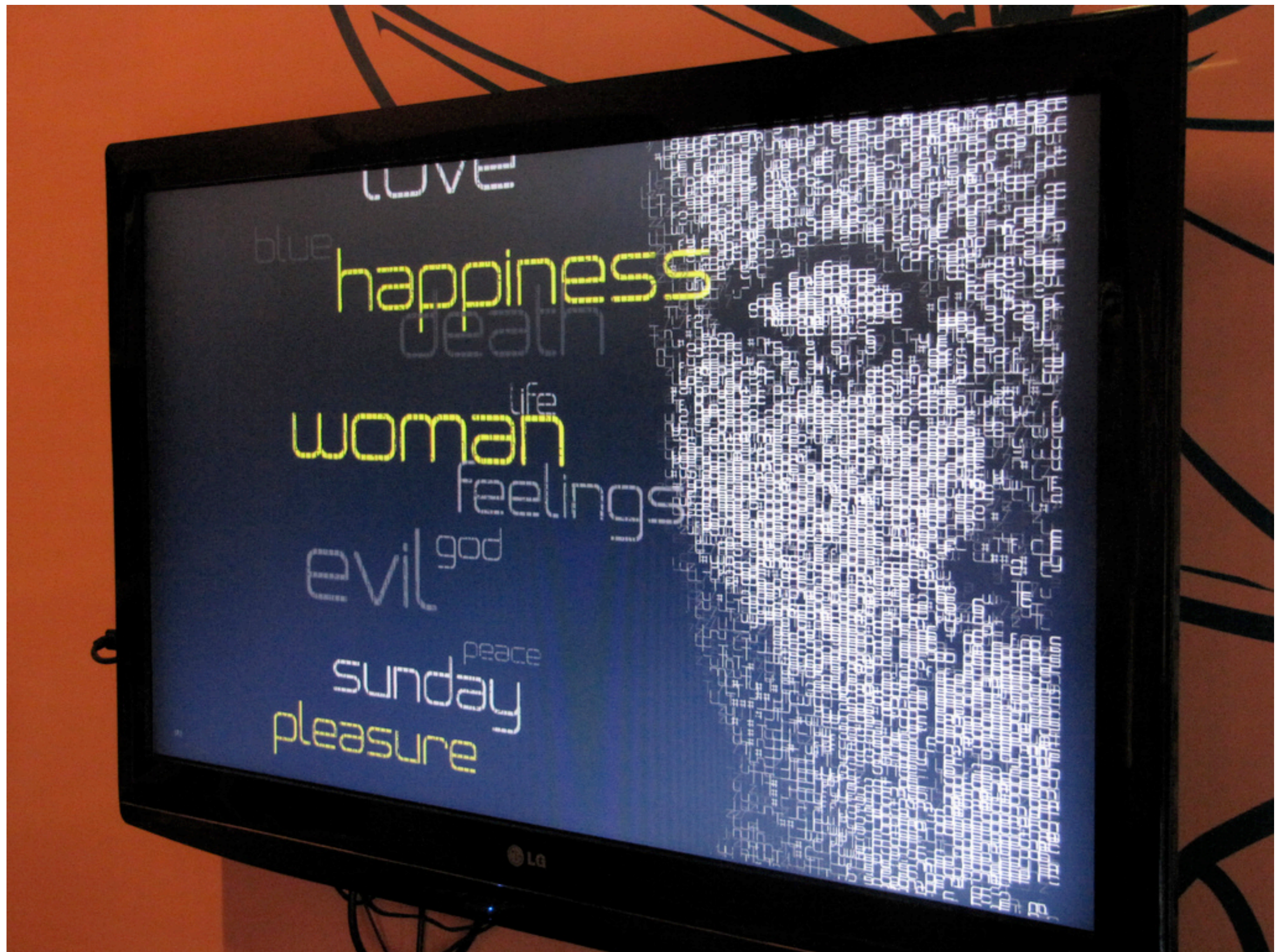


projetos

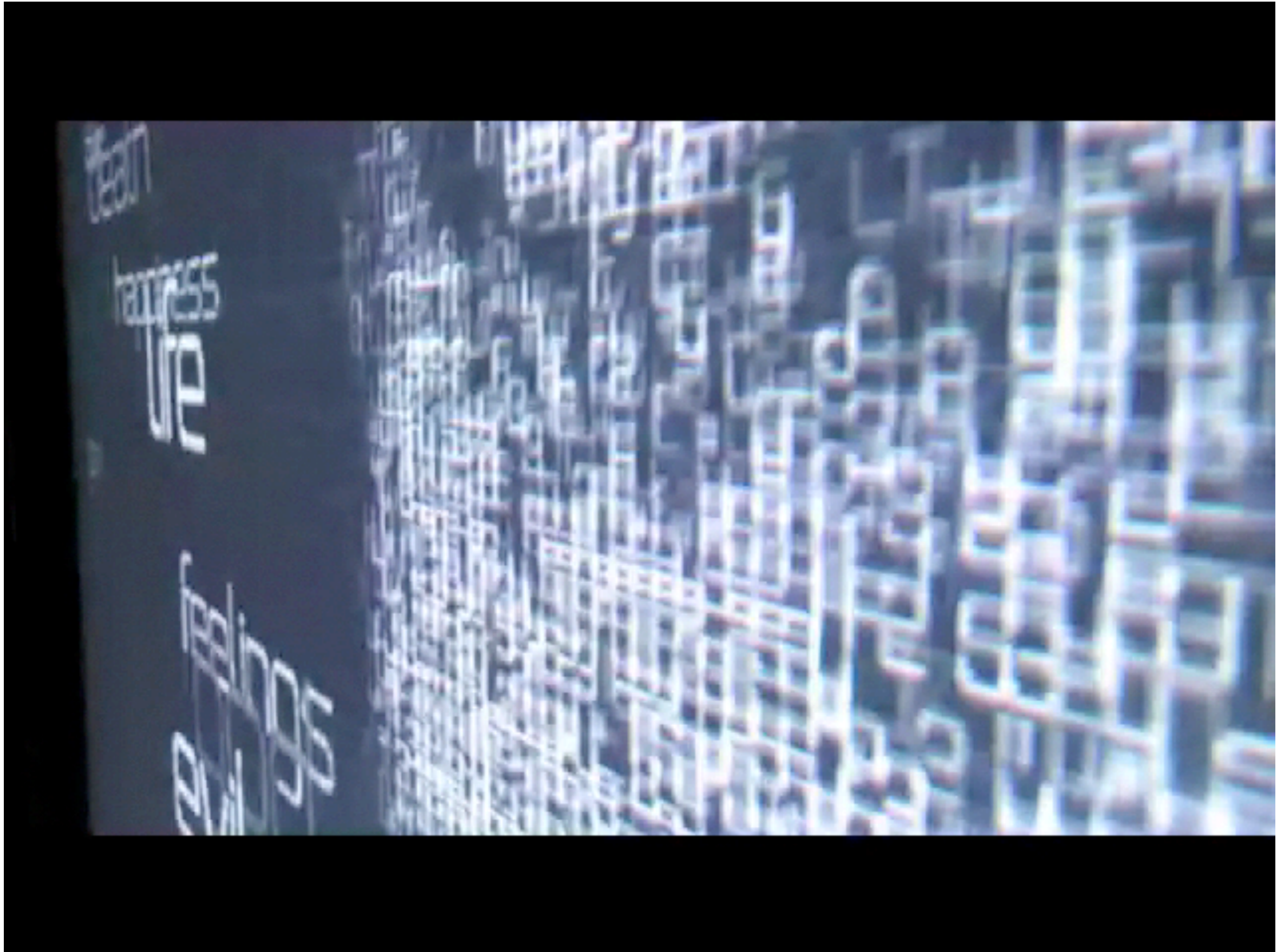
inspirações



bota os vídeos,
mago véi!



marvim gainsbug



marvim gainsbug

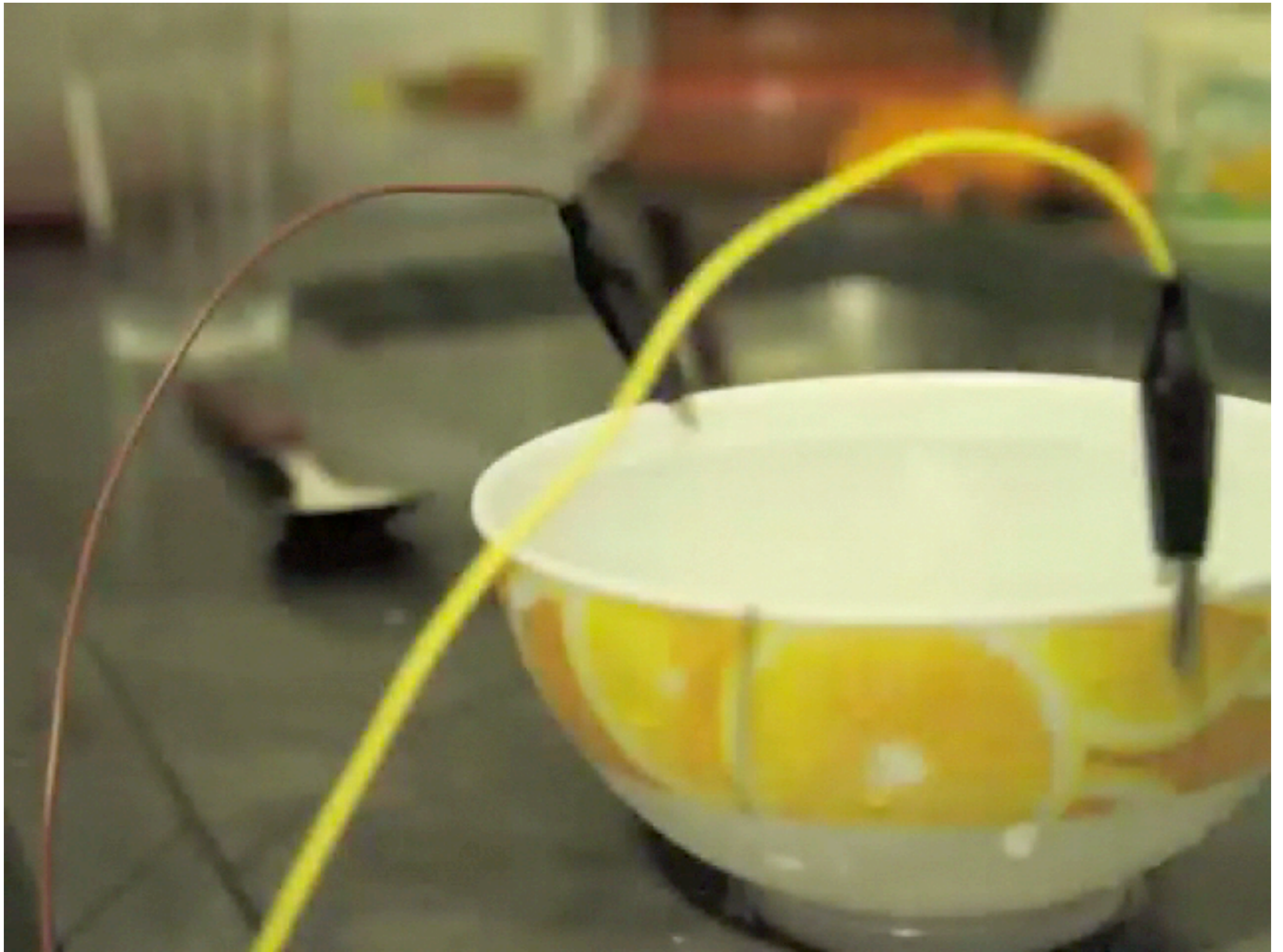


artistas

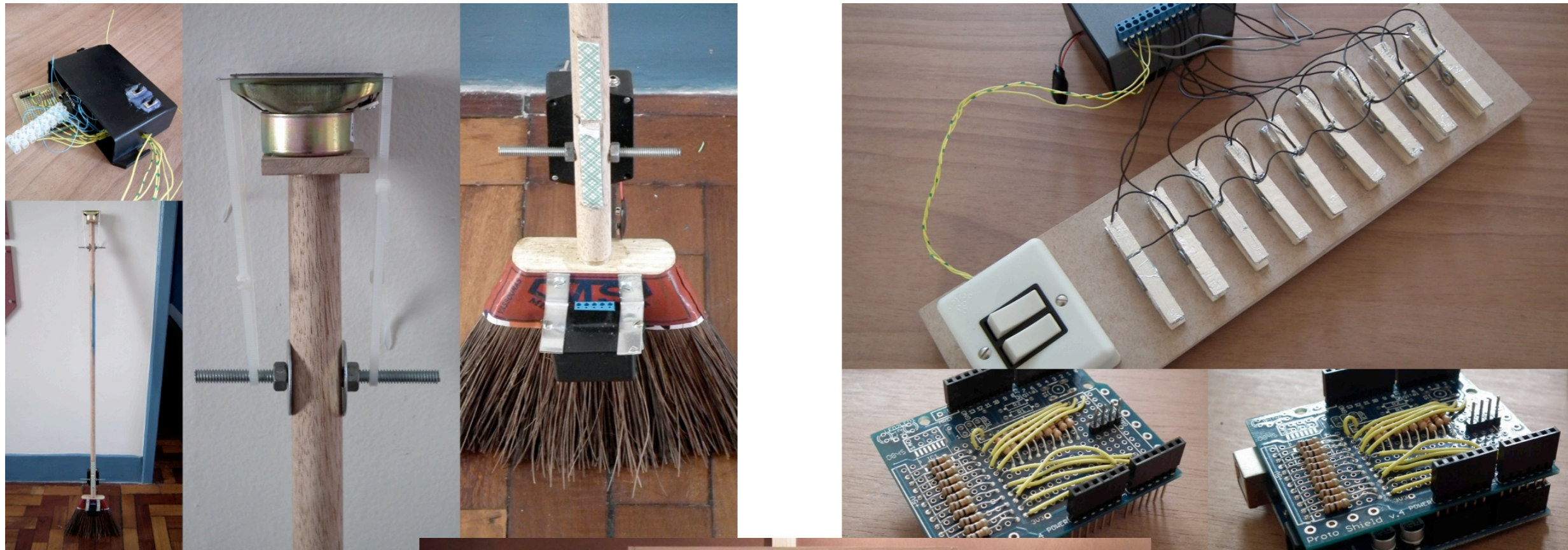
laboratório

instrutores

colaboradores



testes, testes, testes..



simples coisas sonoras



simples coisas sonoras

pesquisa



música, tecnologia, interatividade e criatividade

<http://mustic.info>

linhas de pesquisa

digital musical instruments

automatic accompaniment systems

sound and music analysis

creativity support systems

educational systems

challenges in designing
new interfaces for
musical expression

challenges: digital

the mapping problem

audience understanding

sensor limitations

embodied relationship

challenges: musical

expressivity and virtuosity

context of use

evolution and success criteria

beyond the artifact



sketchument

abstração

abstração

```
SL      W3, #1, W
SL      W2, #2, W
LSR     W4, #14,
ADD     W2, W6,
SL      W4, #2, W
INC     W3
CPSGT   W3, W2
GOTO    $+10
DEC     W3
GOTO    $+8
SUB     W3, W2,
INC     W3
```

abstração

```
SL      W3, #1, W
SL      W2, #2, W
LSR
ADD
SL
INC
CPSGT
GOTO
DEC
GOTO
SUB
INC
```

```
#include <stdio.h>
#include <conio.h>
#define TRUE 1
#define PI 3.141593
#define floatingpointno float
void main()
{
    floatingpointno a;
    clrscr();
    if(TRUE)
    {
        a=PI;
        printf("a=%f\nPI=%f", a, PI);
    }
    getch();
}
```

abstração

```
SL    W3, #1, W
SL    W2, #2, W
LSR
ADD
SL
INC
CPSGT
GOTO
DEC
GOTO
SUB
INC
```

```
#include <stdio.h>
#include <conio.h>
#define TRUE 1
#define PI 3.141592653589793238462643383279502884197169399375105820974944597408
#define floati
void main()
{
    floatingpo
    clrscr();
    if(TRUE)
    {
        a=PI;
        printf
    }
    getch();
}
```

```
class Apple {
    public String color="red";
}

public class Main {
    public static void main(String[] args) {
        Apple apple = new Apple();
        System.out.println(apple.color);

        changeApple(apple);
        System.out.println(apple.color);
    }

    public static void changeApple(Apple apple){
        apple.color = "green";
    }
}
```


abstração

```
SL    W3, #1, W
SL    W2, #2, W
LSR
ADD
SL
INC
CPSGT
GOTO
DEC
GOTO
SUB
INC
```

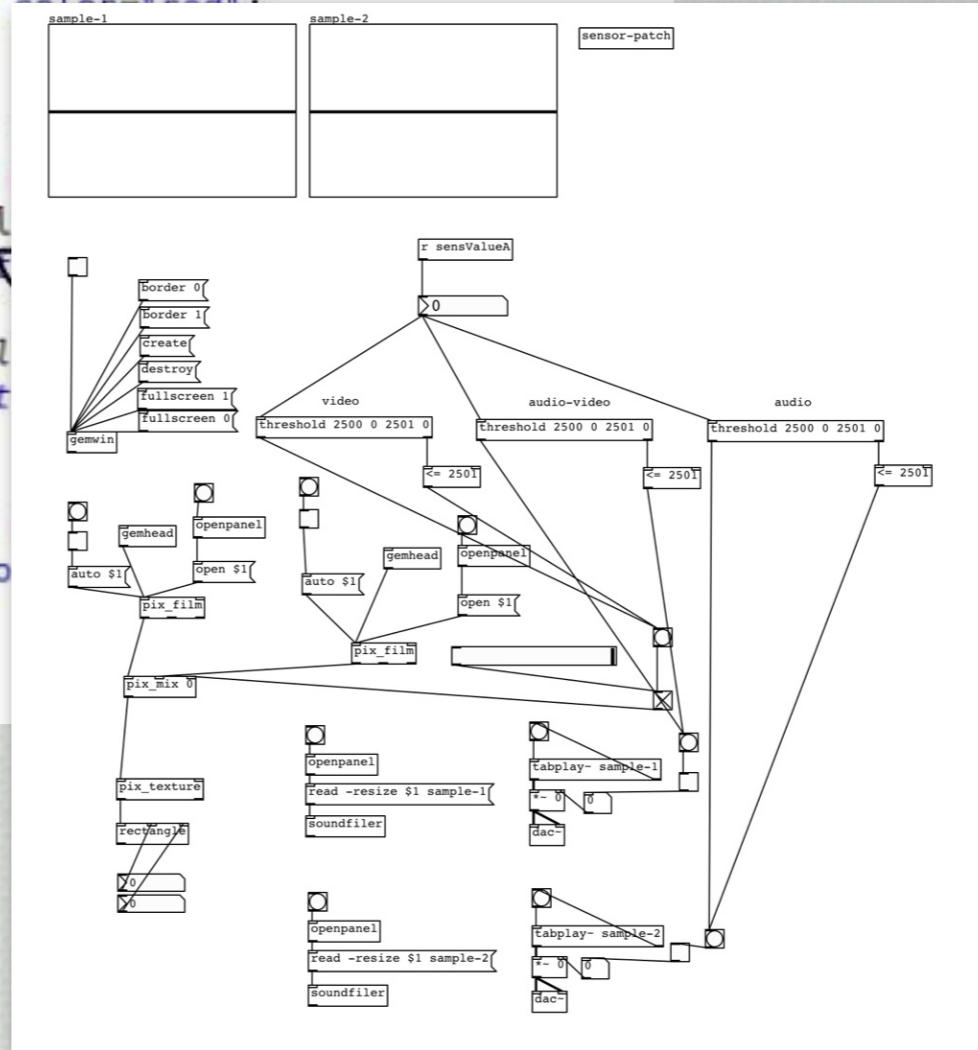
```
#include <stdio.h>
#include <conio.h>
#define TRUE 1
#define PI 3.141592653589793238462643383279502884197169399375105820974944597408
#define floati
void main()
{
    floatingpoint
    clrscr();
    if(TRUE)
    {
        a=PI;
        printf
    }
    getch();
}
```

```
class Apple {
public String color;
}

public class Main
public static
Apple appl
System.out

changeAppl
System.out

public static
apple.color
}
```



abstração

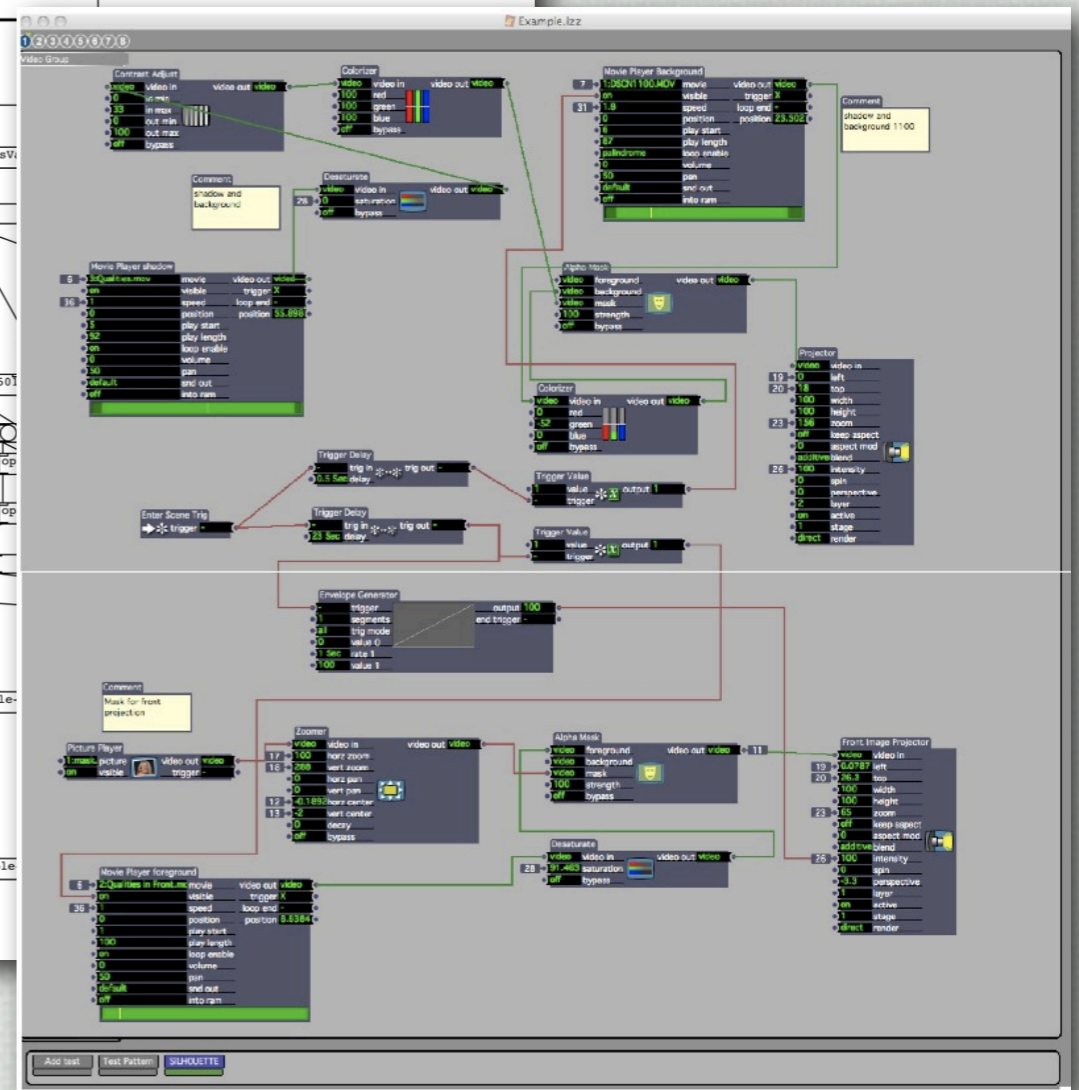
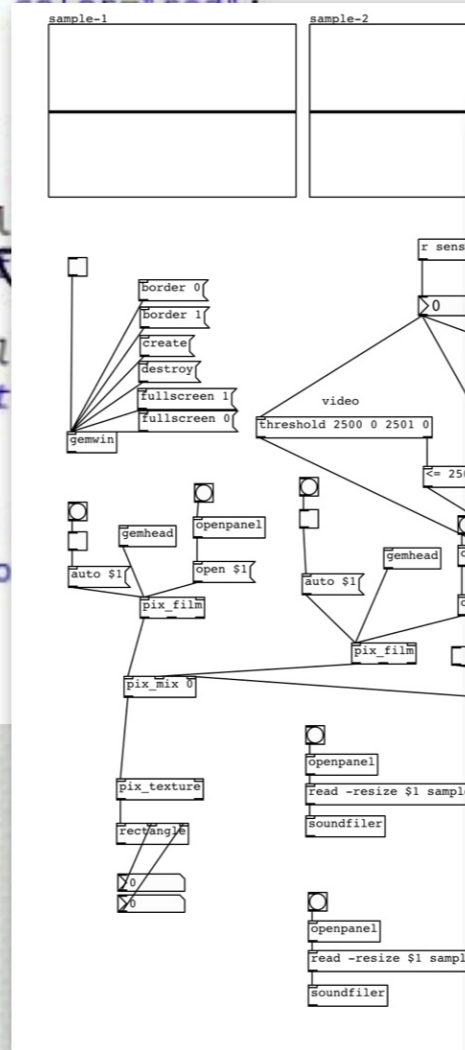
```
SL      W3, #1, W
SL      W2, #2, W
LSR
ADD
SL      #include <stdio.h>
        #include <conio.h>
        #define TRUE 1
        #define PI 3.141592653589793238462643383279502884197169399375105820974944597
        #define floati
void main()
{
    floatingpoint
    clrscr();
    if(TRUE)
    {
        a=PI;
        printf
    }
    getch();
}
```

```
class Apple {
public String color;
}

public class Main
public static
Apple appl
System.out

changeAppl
System.out

public static
apple.color
}
```



abstração

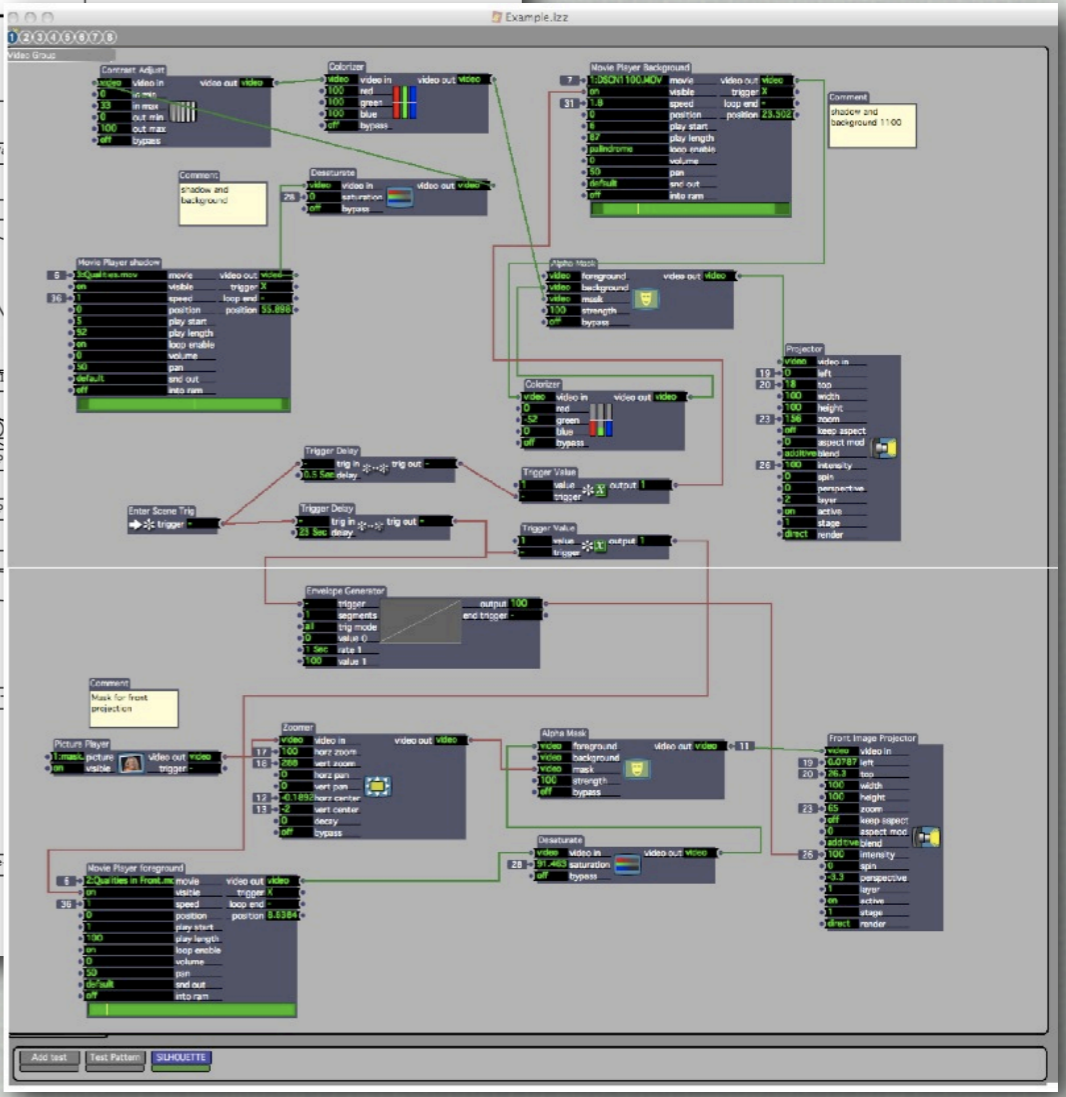
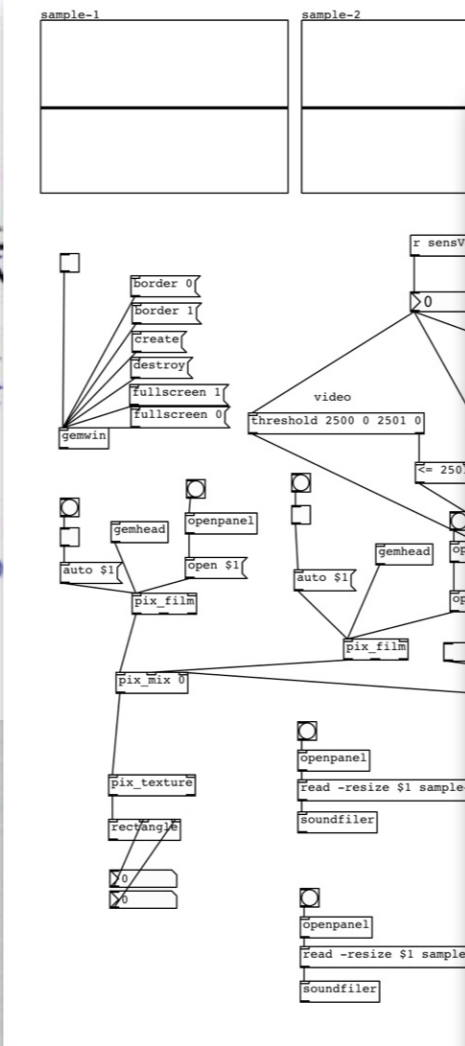
```
SL W3, #1, W
SL W2, #2, W
LSR
ADD
SL
INC
CPSGT
GOTO
DEC
GOTO
SUB
INC
```

```
#include <stdio.h>
#include <conio.h>
#define TRUE 1
#define PI 3.141592653589793238462643383279502884197169399375105820974944597408
void main()
{
    floatingpoint_t a;
    clrscr();
    if(TRUE)
    {
        a=PI;
        printf("a = %f\n", a);
    }
    getch();
}
```

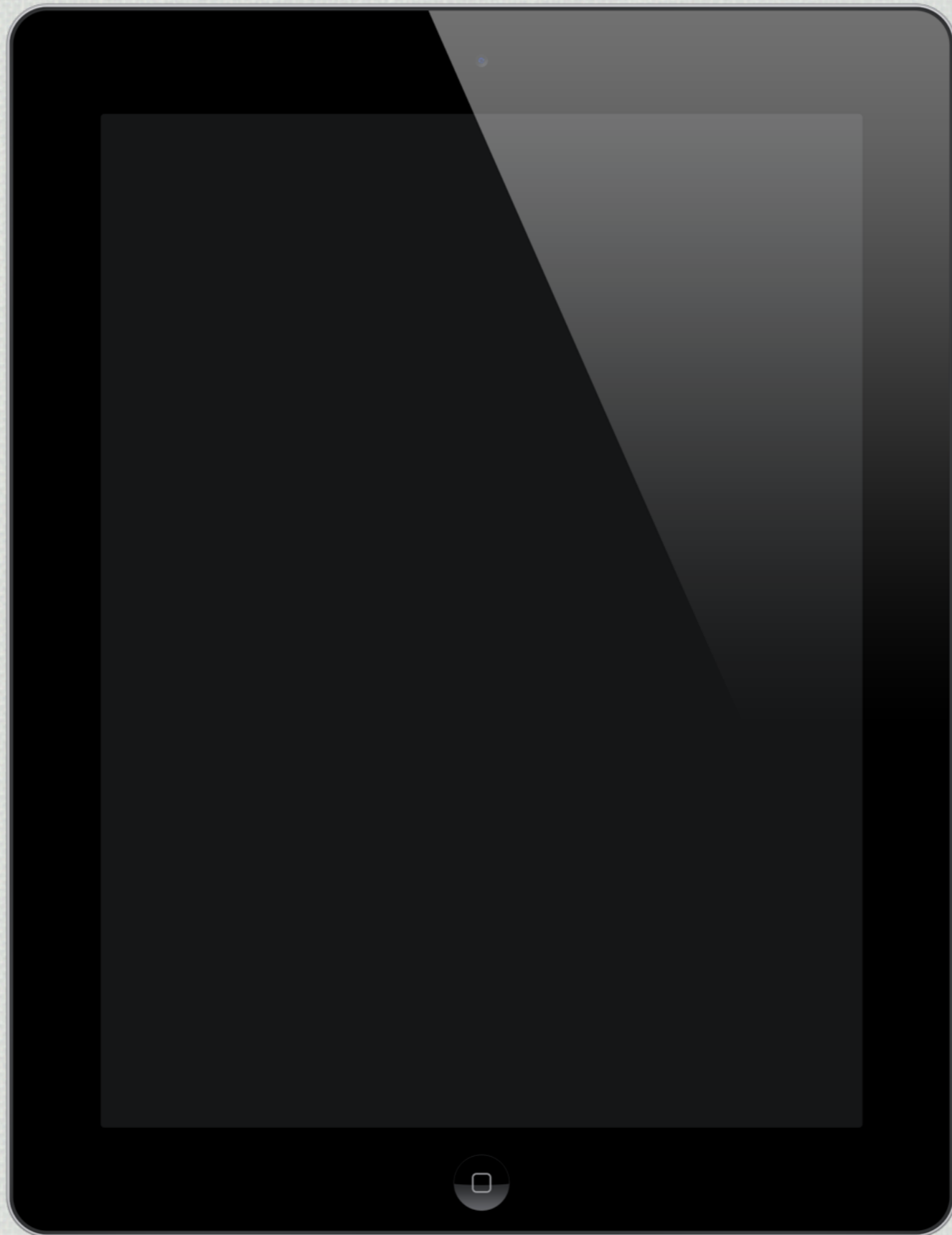
```
class Apple {
public String color;
}

public class Main
public static
Apple appl;
System.out.println("changeApple");
System.out.println("apple color");

public static
apple.color;
```



ainda há espaço para mais camadas de abstração

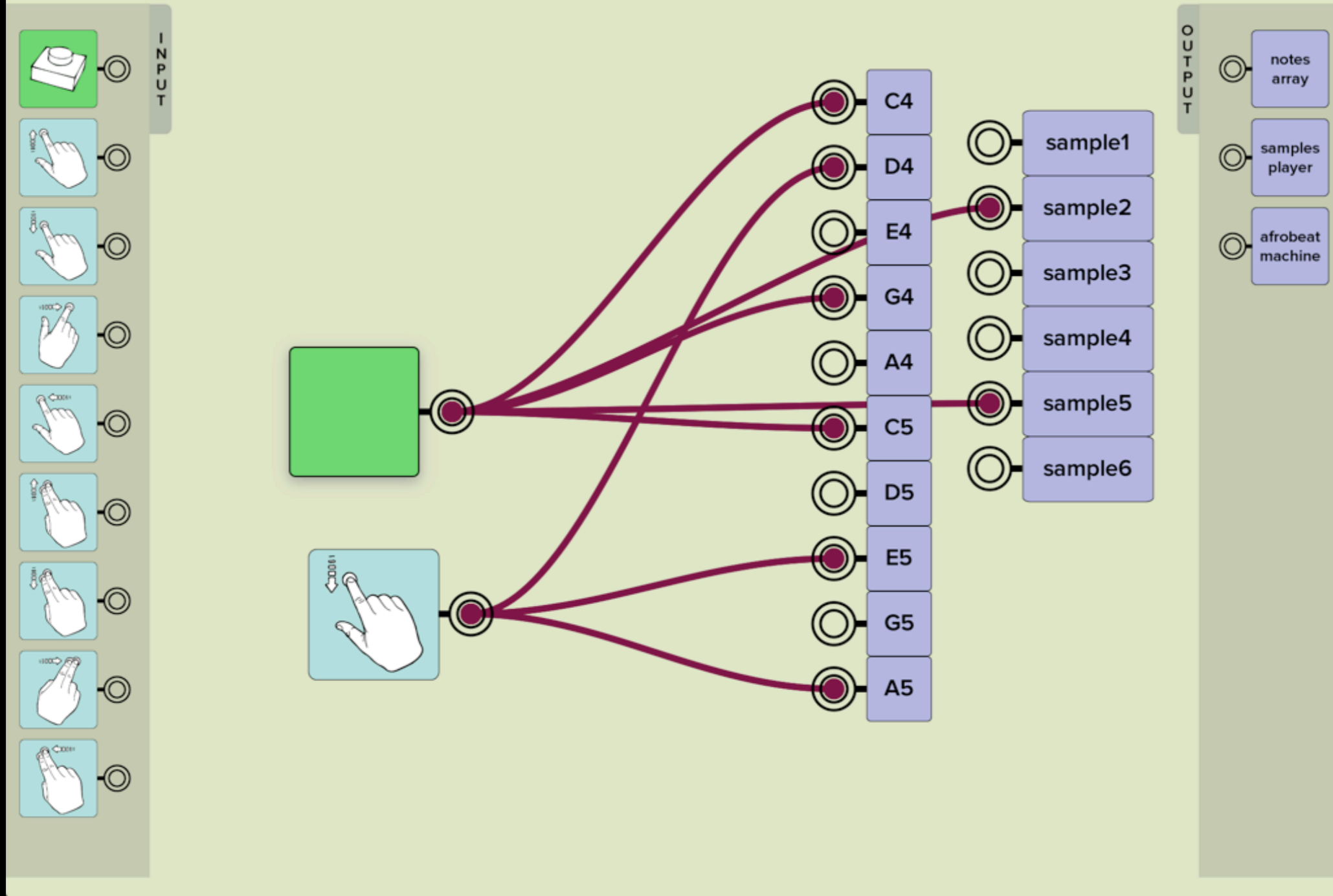


Feedback

Help

Edit Mode

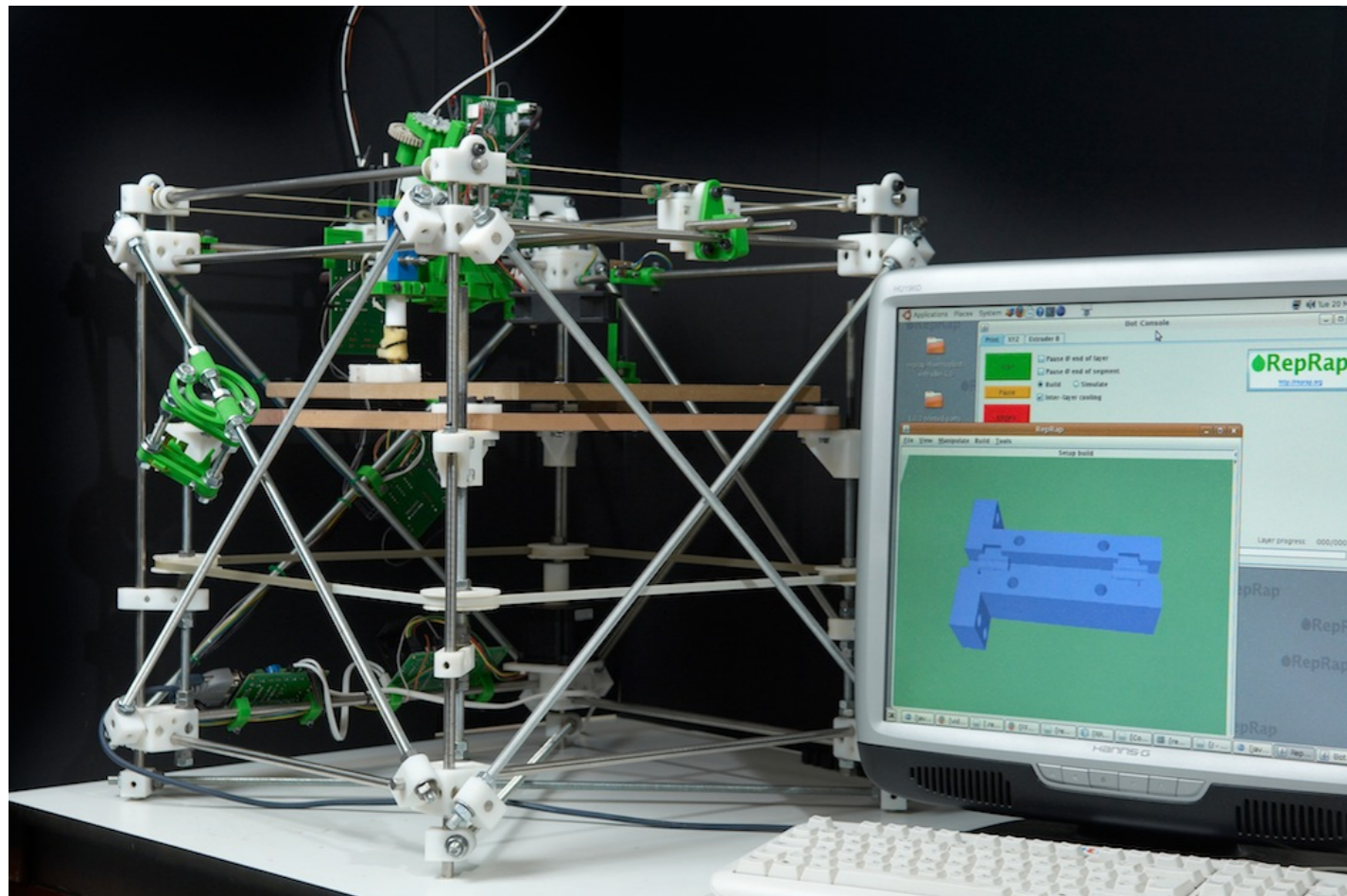
Go to Play Mode



conceitos

referências

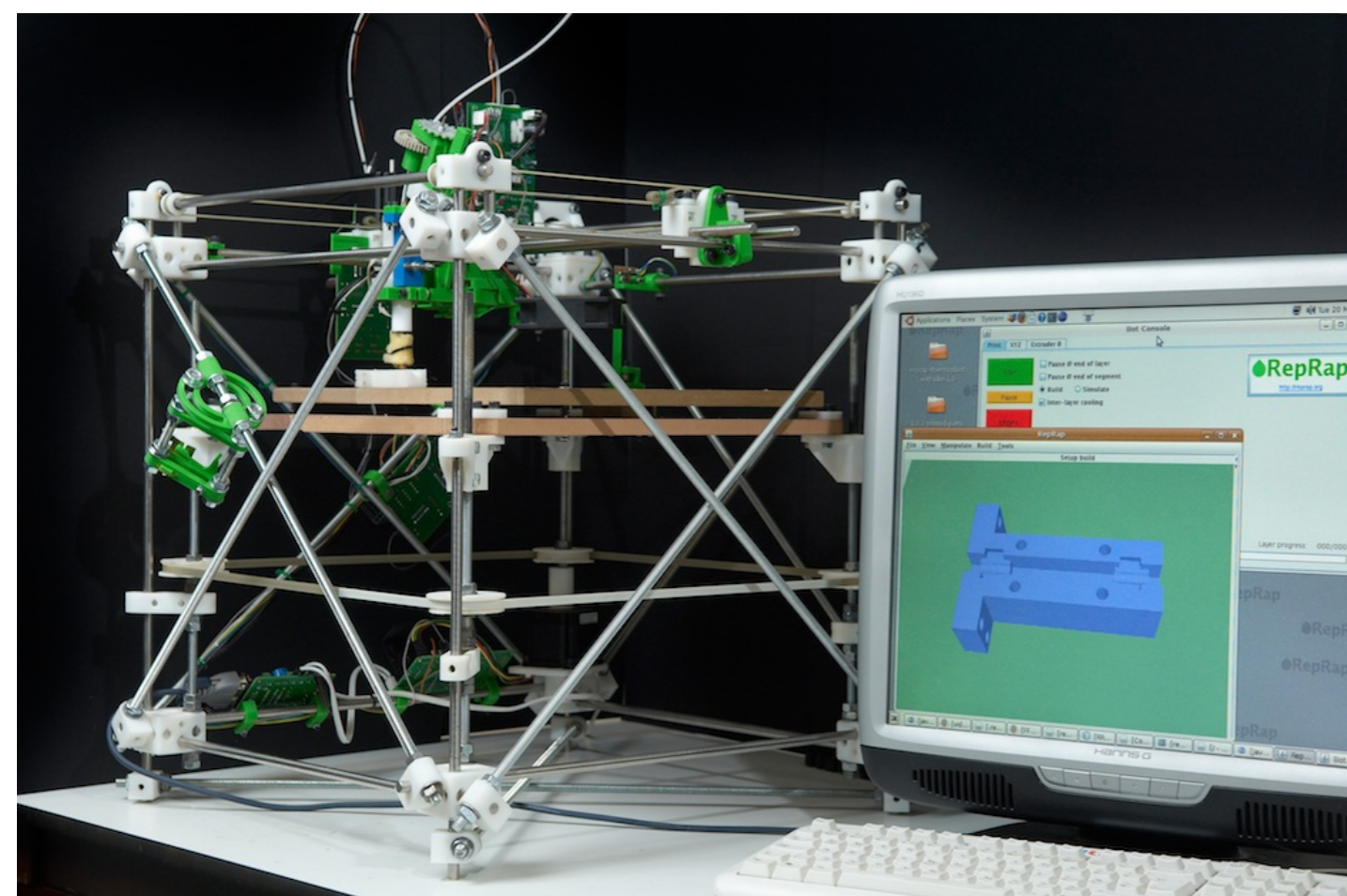
More than mere consumers of technology, we are *makers*, adapting technology to our needs and integrating it into our lives. Some of us are born *makers* and others, like me, become *makers* almost without realizing it.



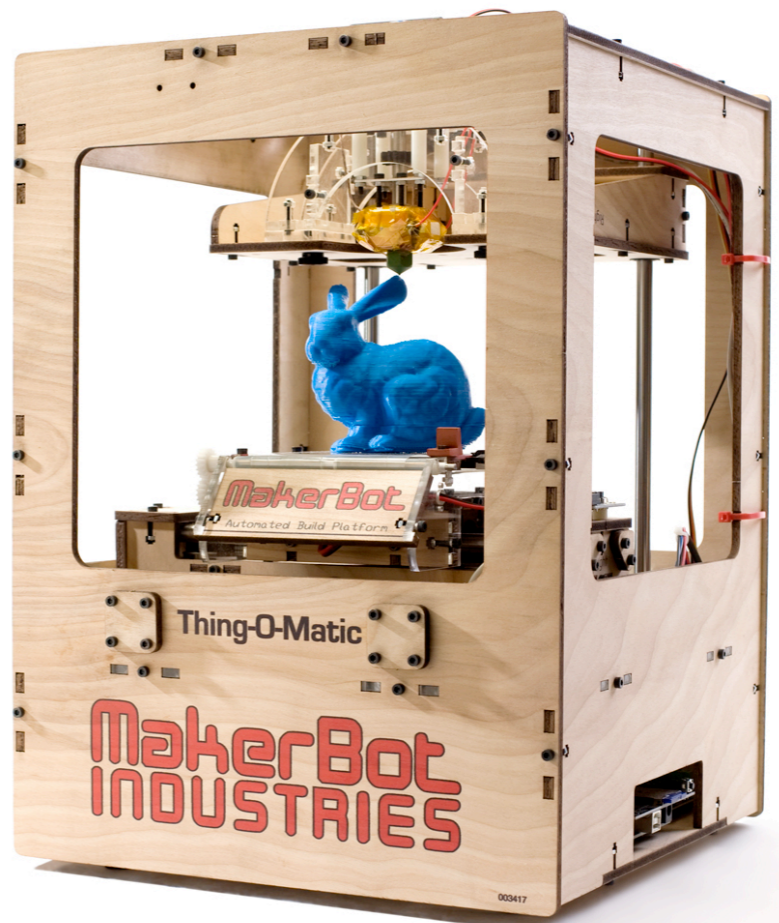
without realizing it.

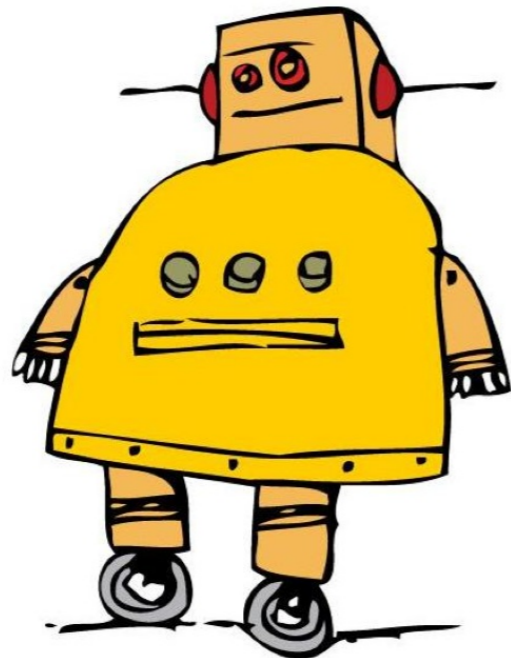
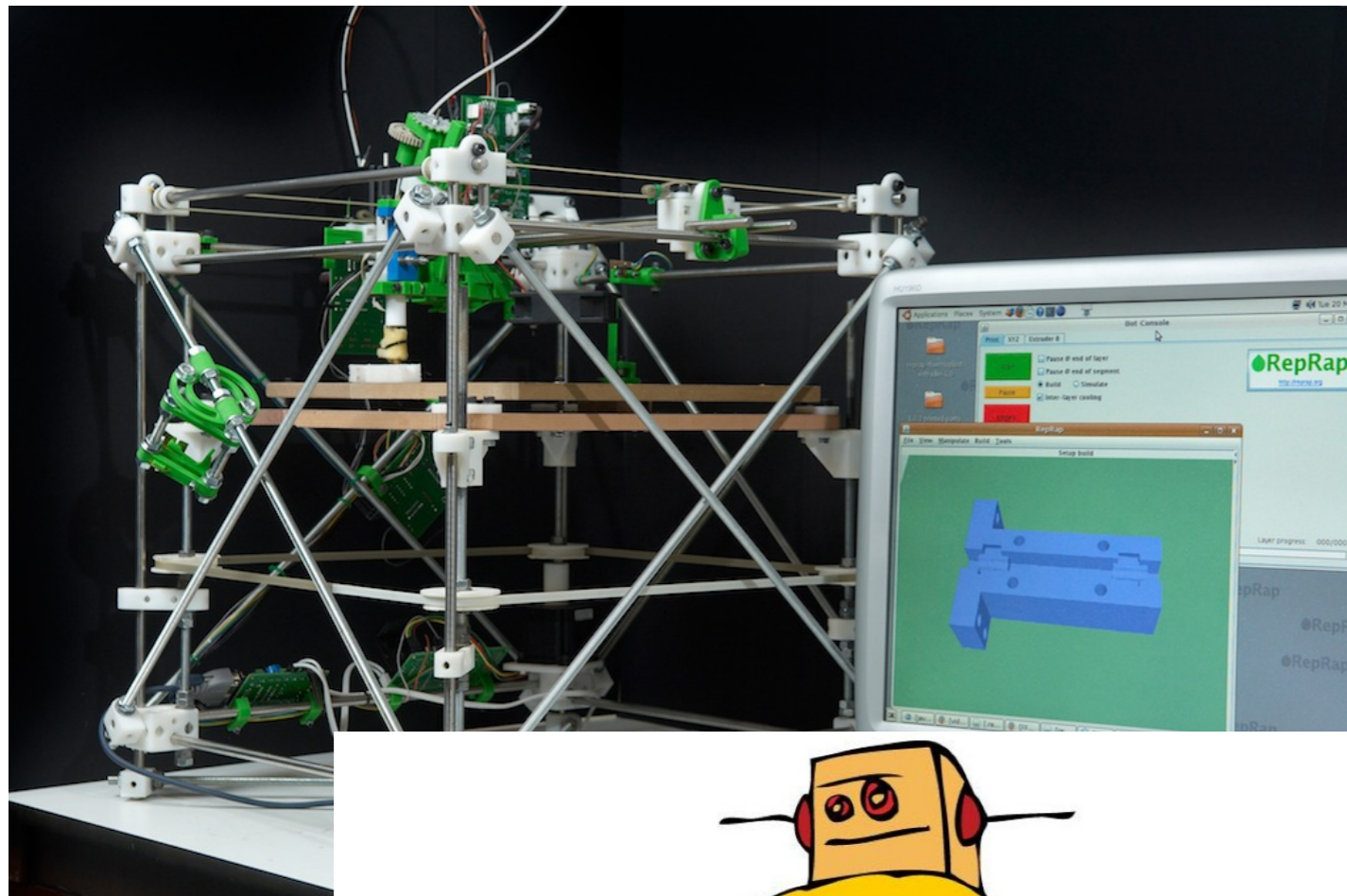


some of us are both
become *makers* almost



without realizing it.





instructables.com
THE WORLD'S BIGGEST SHOW & TELL



"I don't want to buy new stuff all the time. I want to hack the stuff I already have so it works better for me."

Jane Ní Dhulchaointigh



3 estágios: não saber,
ação e realização

tudo é um rascunho

não existe estágio
de edição

fingir que sabe = saber

se procrastinar uma ideia por
mais de uma semana, abandone-a

ficar pronto não é terminar,
é fazer outras coisas

tarefa terminada, descarte-a

ria da perfeição

peças sem as mãos
sujas estão erradas

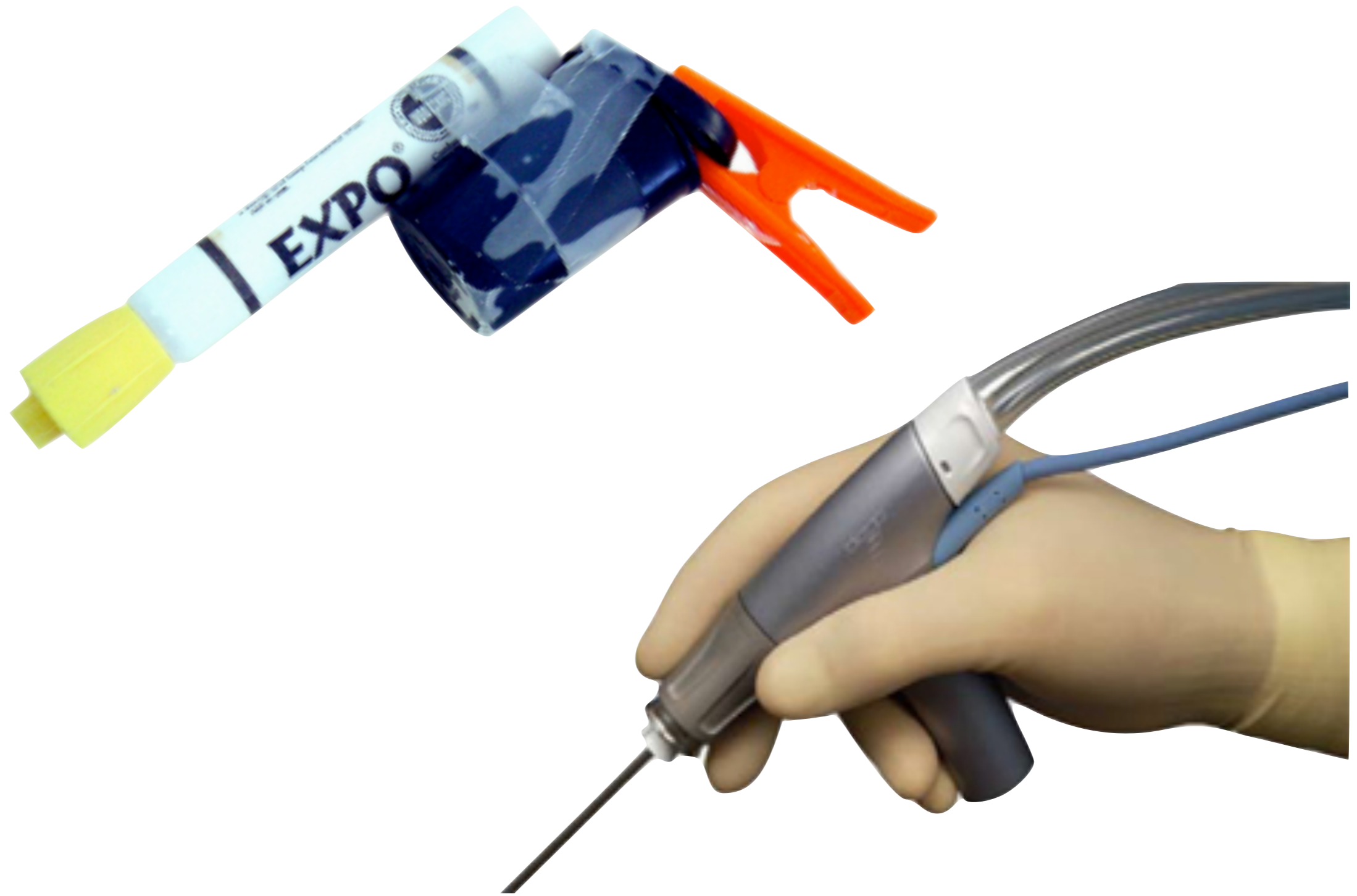
falhas contam como
"feito", cometa erros

destruição é uma
variação do "feito".

publicar uma ideia sem ações,
é um fantasma do "feito"

o "feito" é o motor do mais





Diego Powered Dissector System
desenvolvido por Gyrus ENT. e IDEO

não criar do zero

não reinventar a roda

combinar coisas que já existem

feio e imperfeito

validar logo a ideia

partir para a próxima etapa



lojas para comprar

sparkfun

dx.com

soldafria

seeedstudio



sugestão de blogs

creativeapplications.net

createdigitalmusic.com

synthopia.com

djtechtools.com

filipecalegario.net