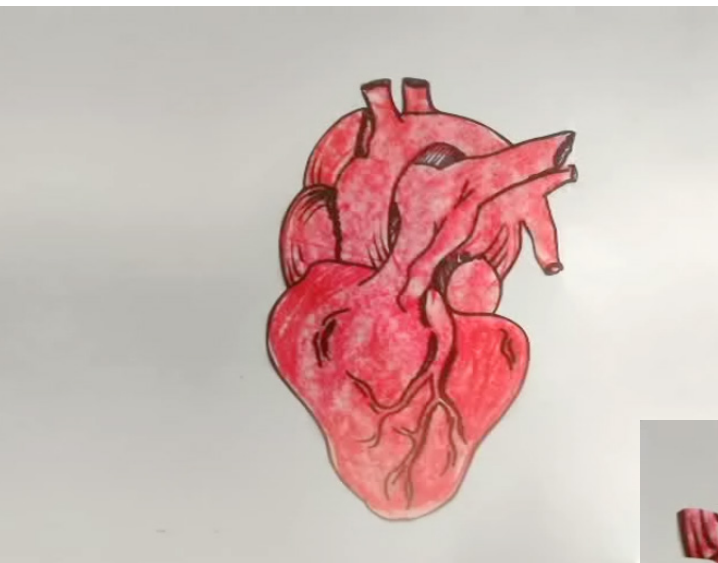






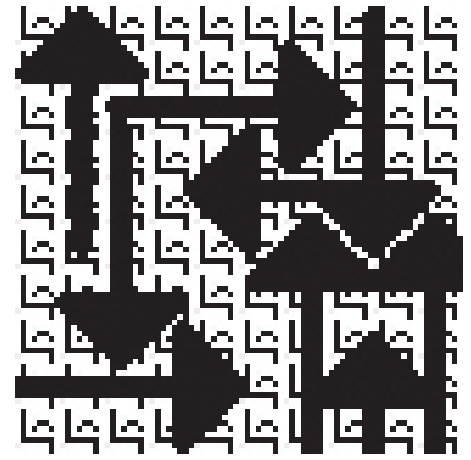
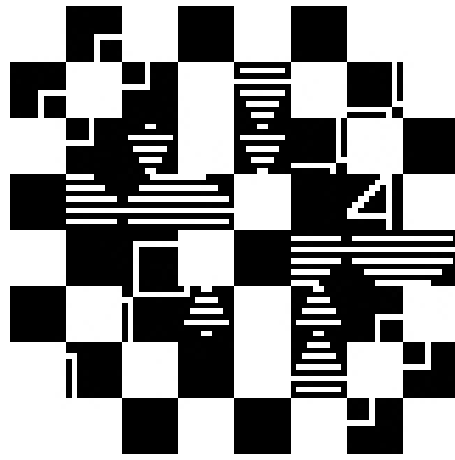
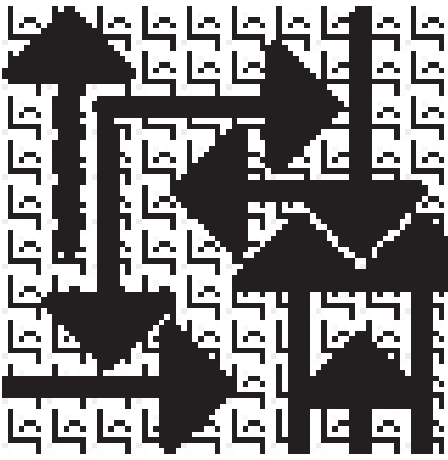


Profiling with RFID  
ik maakte het filmpje voor jessica



Als er een onbekende kaart kwam dan  
speelde we ook een filmpje af met een  
“erroro” en het hart ontploft





80x80 pixels (deze plaatjes zijn vergroot) deze plaatjes kan je vervolgens in processing zetten om zelf het patroon en op een simple wijzen aan te passen hoe jij het wilt, of juist de codes zo door elkaar heen gooien dat het een mix word van pixels.

Door de getallen aan te passen achter DrawImage;..... en dan de getallen tussen de haakjes plaats je het patroon waar je het hebben wilt en hoe, dus je roteert het of spiegelt het en ook de positie bepaal je met de getallen.

The screenshot shows a Processing 2.2.1 IDE window with a sketch named 'sketch\_140909b'. The main canvas displays a 4x4 grid of images, each containing a stylized black silhouette of a tree or plant against a white background. The code editor shows the following code:

```
drawImage(2, 0, 0);
drawImage(3, 0, 90);
drawImage(3, 1, 180);
drawImage(2, 1, 270);
drawImage(4, 0, 0);
drawImage(5, 0, 90);
drawImage(5, 1, 180);
drawImage(4, 1, 270);

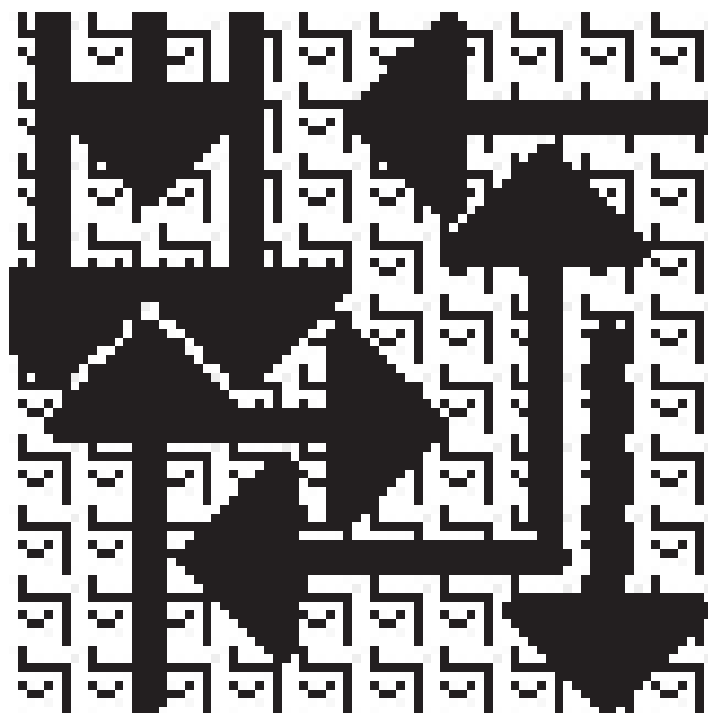
drawImage(0, 2, 0); //numbers: place x - place y - rotation in degrees
drawImage(1, 2, 90);
drawImage(1, 3, 180);
drawImage(0, 3, 270);
drawImage(2, 2, 0);
drawImage(3, 2, 90);
drawImage(3, 3, 180);
drawImage(2, 3, 270);
drawImage(4, 2, 0);
drawImage(5, 2, 90);
drawImage(5, 3, 180);
drawImage(4, 3, 270);
}

void drawImage(int xC, int yC, float myAngle) {
  pushMatrix();
```

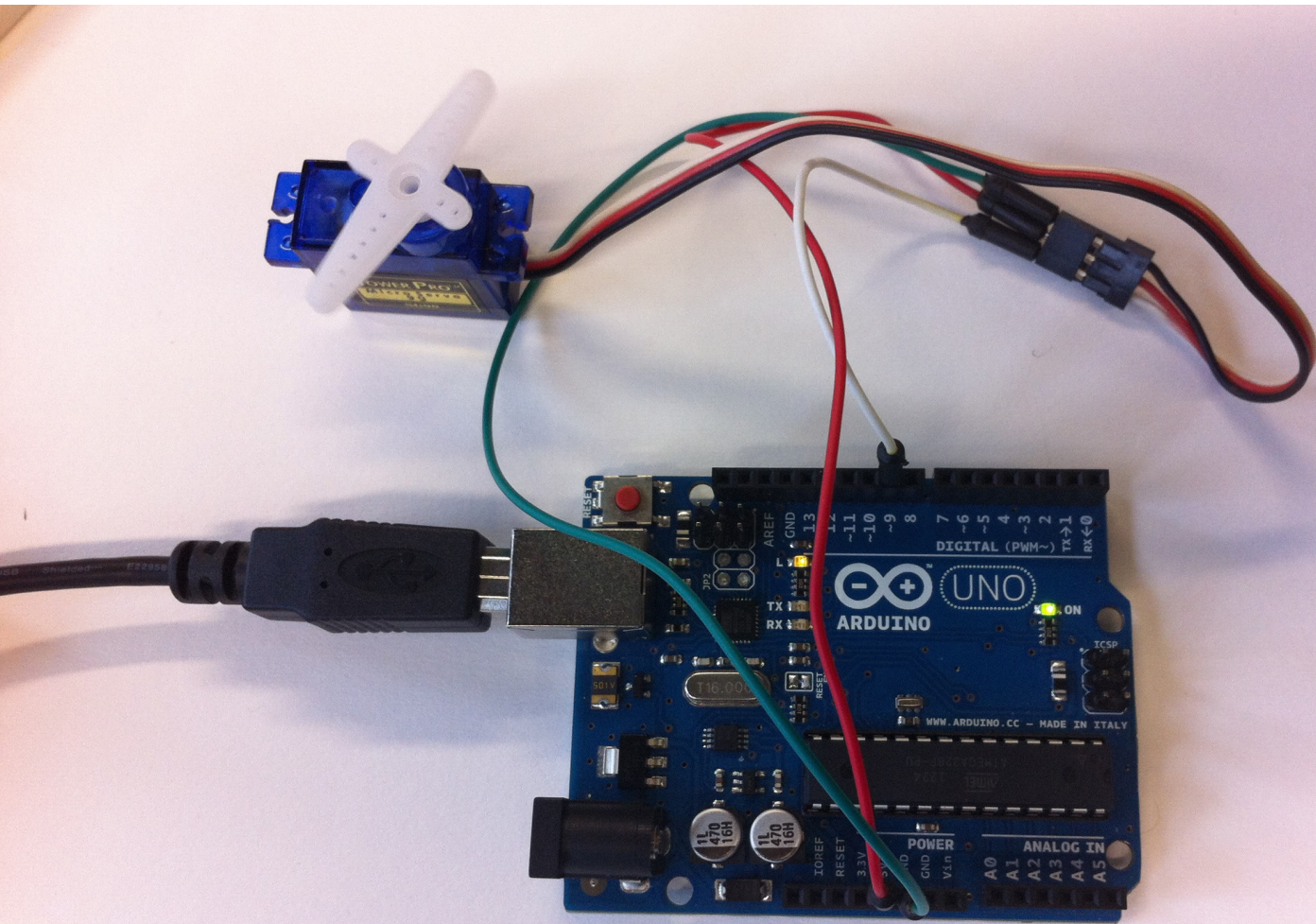
The IDE interface includes a toolbar with icons for play, stop, and other functions. The status bar at the bottom shows 'ImagePatterns' and 'tweaks (0)'. The window title is 'sketch\_140909b | Processing 2.2.1'.



Daarna hebben we deze print in de computer gezet en een bestand van gemaakt die de brei machine kan lezen, daarna door gestuurd naar de breimachine zelf, 2 kleuren in de machine spannen en breien maar.



# Servo motor



```
Arduino Bestand Bewerken Sketch Extra Help
sketch_oct02a | Arduino 1.0.5
sketch_oct02a §
#include <Servo.h>

Servo myservo; // create servo object to control a servo
               // a maximum of eight servo objects can be created

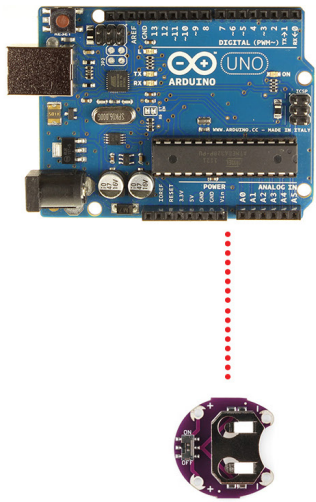
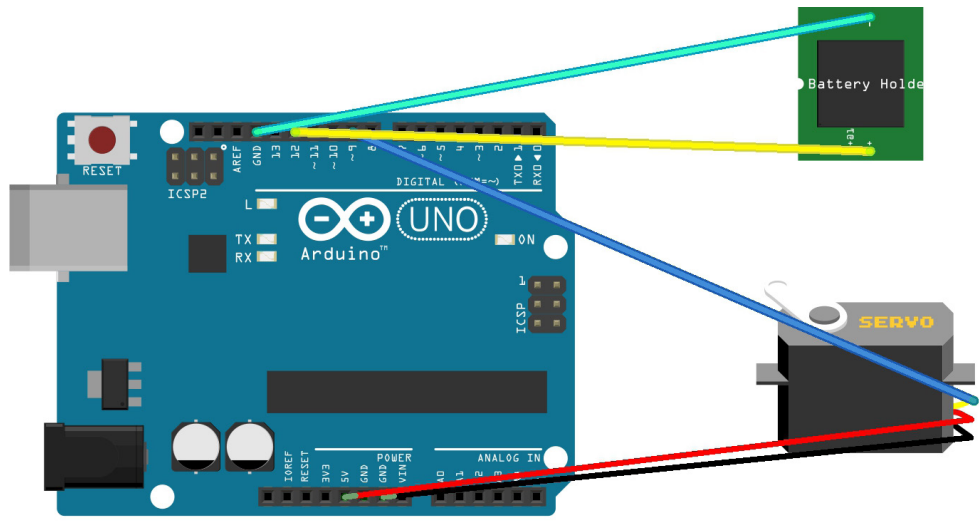
int pos = 0; // variable to store the servo position

void setup()
{
  myservo.attach(9); // attaches the servo on pin 9 to the servo object
}

void loop()
{
  for(pos = 0; pos < 180; pos += 1) // goes from 0 degrees to 180 degrees
  {
    // in steps of 1 degree
    myservo.write(pos); // tell servo to go to position in variable 'pos'
    delay(7);           // waits 15ms for the servo to reach the position
  }
  for(pos = 180; pos>=1; pos-=1) // goes from 180 degrees to 0 degrees
  {
    // tell servo to go to position in variable 'pos'
    myservo.write(pos); // tell servo to go to position in variable 'pos'
    delay(7);           // waits 15ms for the servo to reach the position
  }
}

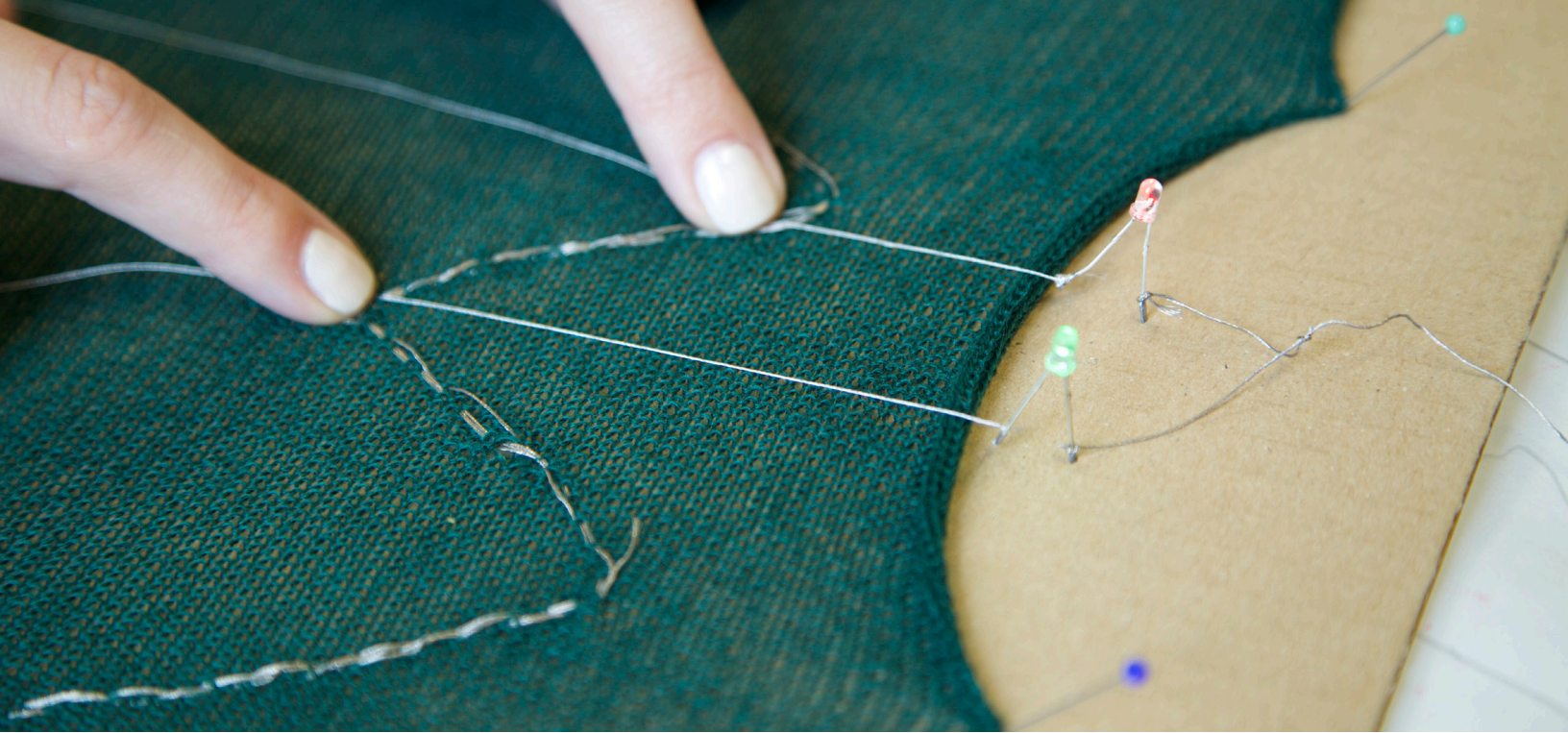
Upload voltooid.
Binaire sketch-grootte: 2.666 bytes (van een 32.256-byte maximum)
```











Joeke van der Veen, Megan Mills, Alex Jacque, Lizet van der Knaap

# CURRENT

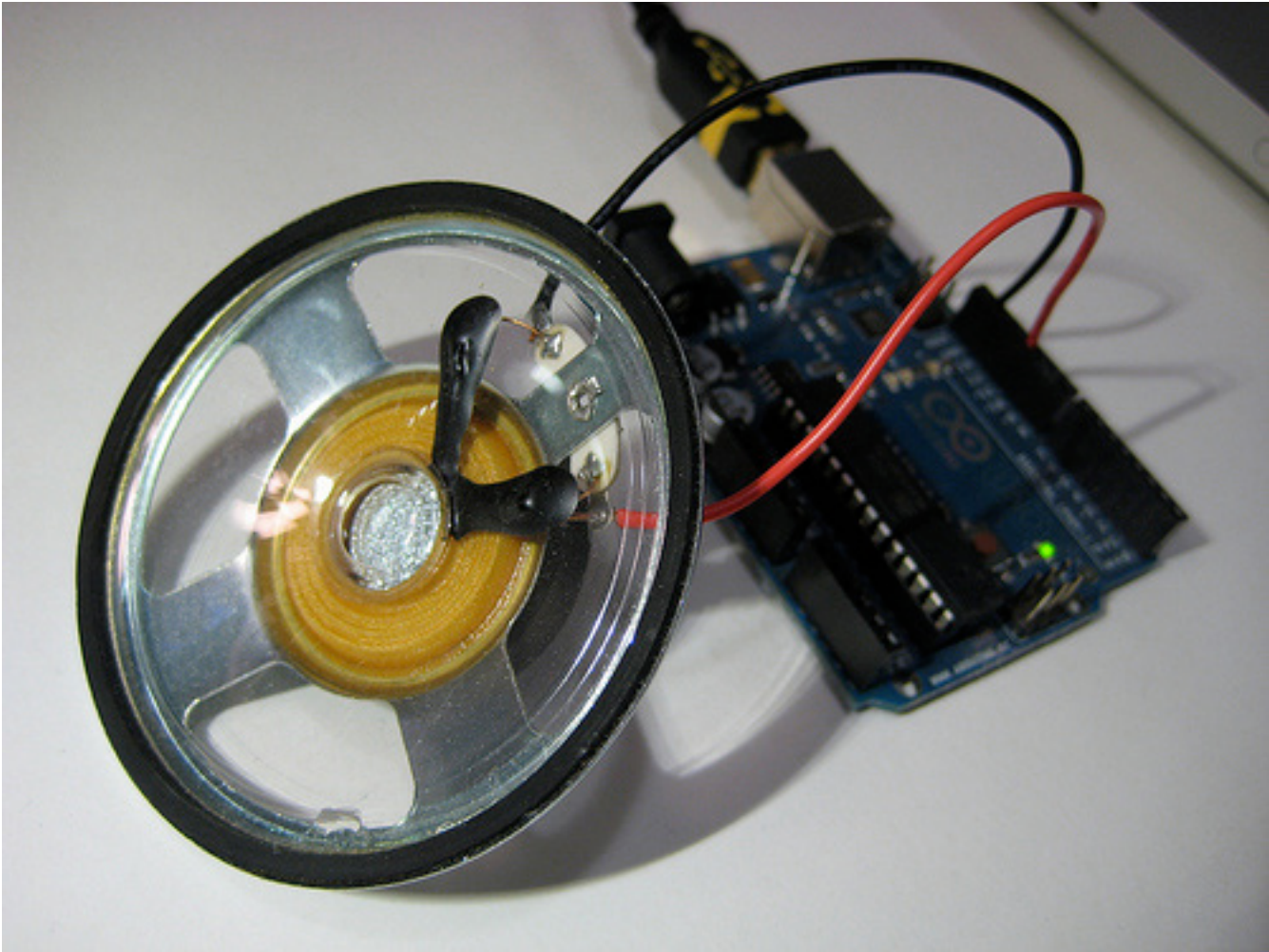
Current is a visualization of data, knitted in real time. The installation is a collaboration of students from Williem de Kooning of Rotterdam, The Netherlands and Maryland Institute College of Art of Baltimore, Maryland.

Through the use of a speculative automated knitting machine along with materials consisting of yarn, conductive thread, resistive thread, LED lights, and small power source, we are able to transmit the amount of sunlight from one city and kit a complex visualization in another.





geluidsirene



```
for ( int i = 500; i <2000 ; i = i - 50 )
```

als je deze codes aanpast, de getallen hoger of lager, kun je de snelheid en de toonhoogte veranderen waardoor het hoger of lager gaat,



## geluidsirene

```
int beats = 50;

void setup() {

  pinMode(8, OUTPUT);
  pinMode(9, OUTPUT);
  digitalWrite(9, LOW);

  pinMode(13, OUTPUT);

}

void loop() {
  beats = beats + 5;

  digitalWrite(8, HIGH);
  digitalWrite(13, HIGH);

  //delay(1000);

  for( int i = 0; i <beats ; i = i + 1 )
    __asm__("nop\n\t");

  digitalWrite(8, LOW);
  digitalWrite(13, LOW);

  //delay(1000);

  for( int i = 0; i <beats ; i = i + 1 )
    __asm__("nop\n\t");

  if (beats > 5000 ) beats = 200;
}
```