

Georg Nees

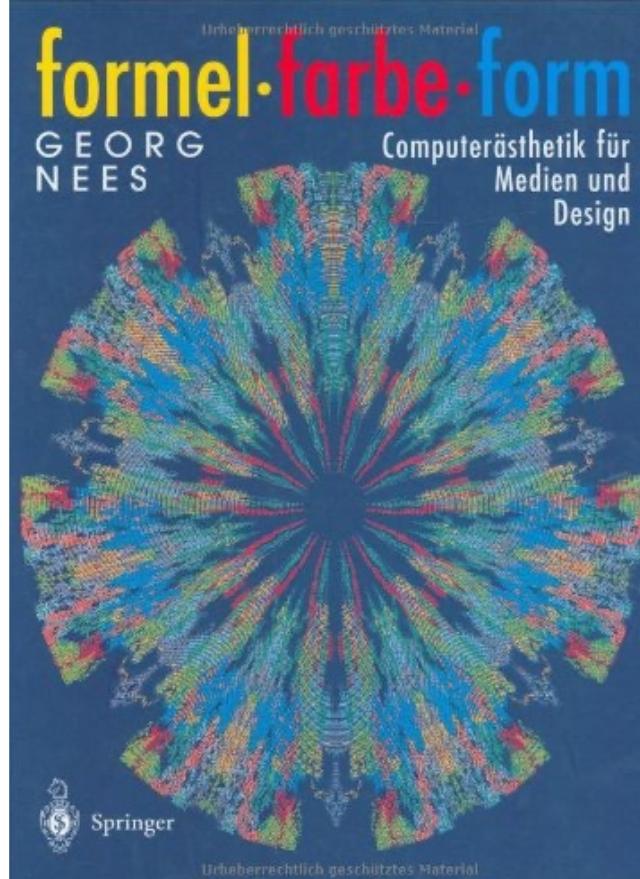
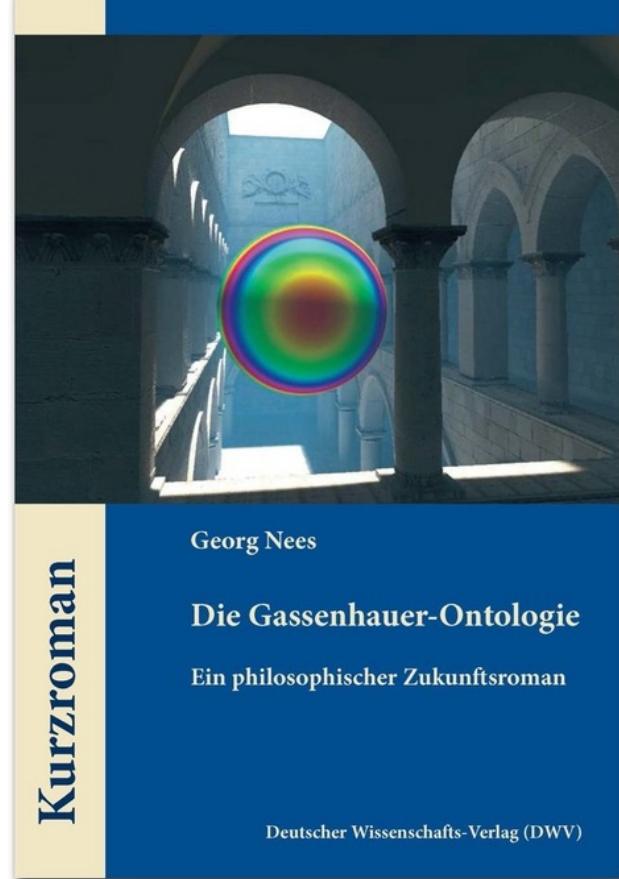
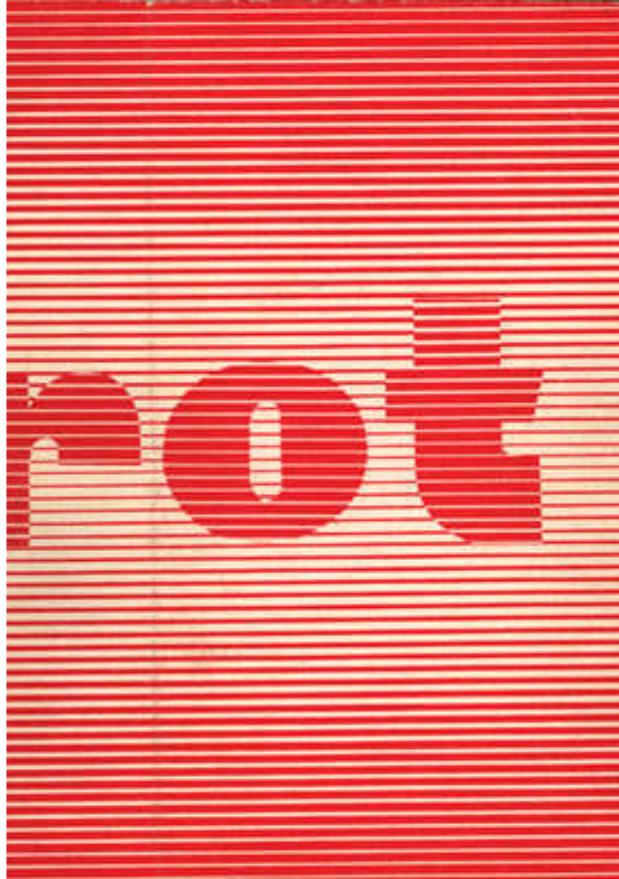
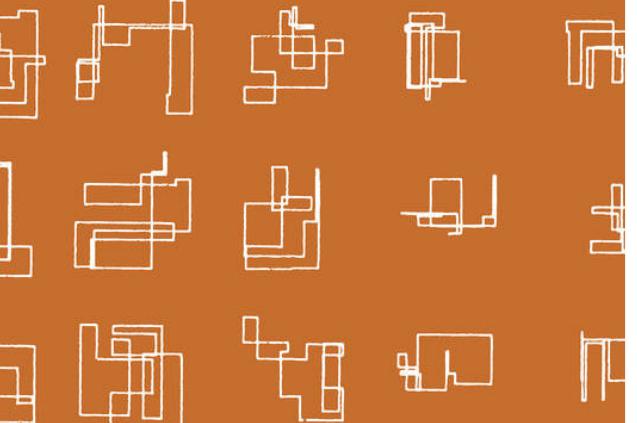
Recreating the Past

Georg Nees

- 1926 – 2016
- German Academic
- Mathematician
- *Generative* computer graphics
- Pioneer in computer art
- Applied computer science professor at University of Erlangen



Generative Computergraphik Georg Nees

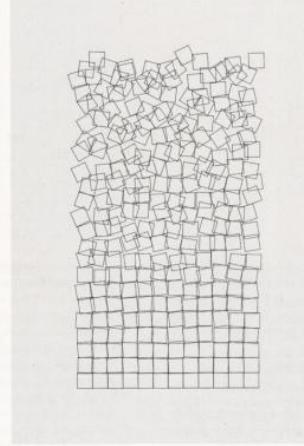


Publications

ZKM Collection

KUNSTHALLE BREMEN

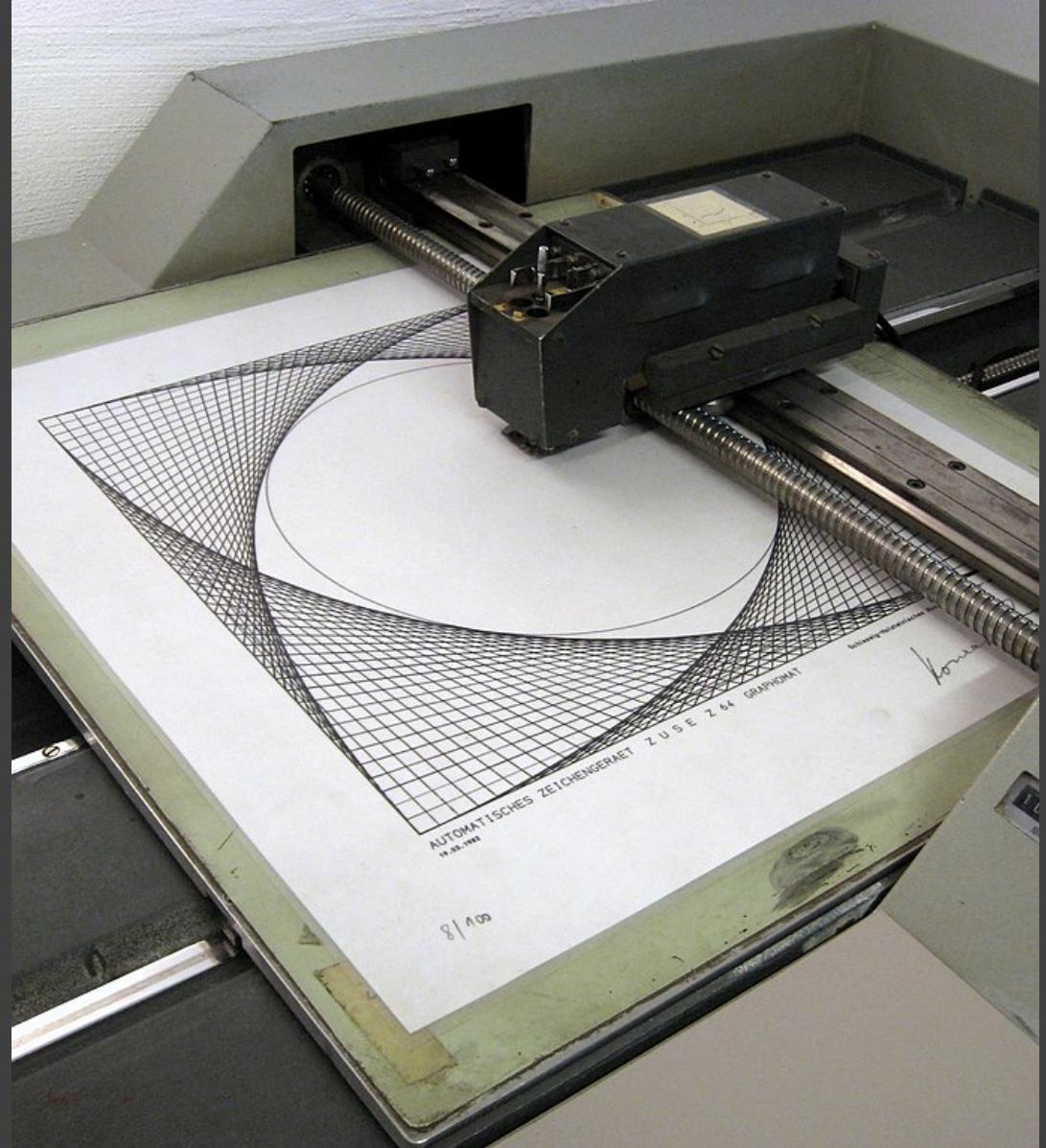
Kupferstichkabinett

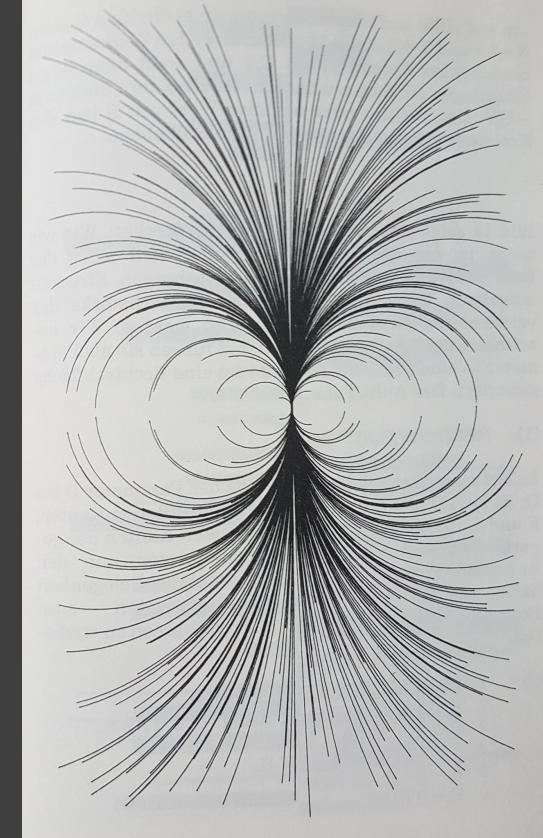


**Georg Nees. Künstliche Kunst:
Die Anfänge**
30.8. – 2.10.2005

ZUSE Z64

- ZUSE Graphomat Z64 flatbed drawing machine of high precision
- Presented in 1961
- Code that had to be input on punch tape or punch cards





Generative Computergraphik

Ein Anwendungsbeispiel zeigt Bild 13. der Programmablauf, durch den Bild 13 generiert wird, lautet folgendermaßen:

(2) 1 'BEGIN'
2 'COMMENT'
DEMONSTRATION KREISBOGEN.,
3 'INTEGER' I, 'REAL' U, V, R.,
4 OPEN(X,Y), JI1.=JS1., JI2.=JS2.,
5 JA1.=0., JE1.=130., JA2.=-80., JE2.=80.,
6 'FOR' I.=1 'STEP' 1 'UNTIL' 200 'DO'
7 'BEGIN' ANF..
8 U.=J1., V.=J2.,
9 'IF' ABS(V)'LESS' 1.0 'THEN' 'GOTO' ANF.,
10 R.=.5*(U*U+V*V)/V.,
11 LEER(U,V), ZIRK(0.0,R,R,-U,V)
12 'END',
13 CLOSE
14 'END' DEMONSTRATION KREISBOGEN.,

Wie Zeile 11 des Programms zeigt, befindet sich der Aufruf von ZIRK im Innern der von Zeile 6 bis Zeile 12 sich erstreckenden Laufanweisung. Das bedeutet, daß insgesamt 200 Kreishäfen generiert werden (siehe Zeile 6).

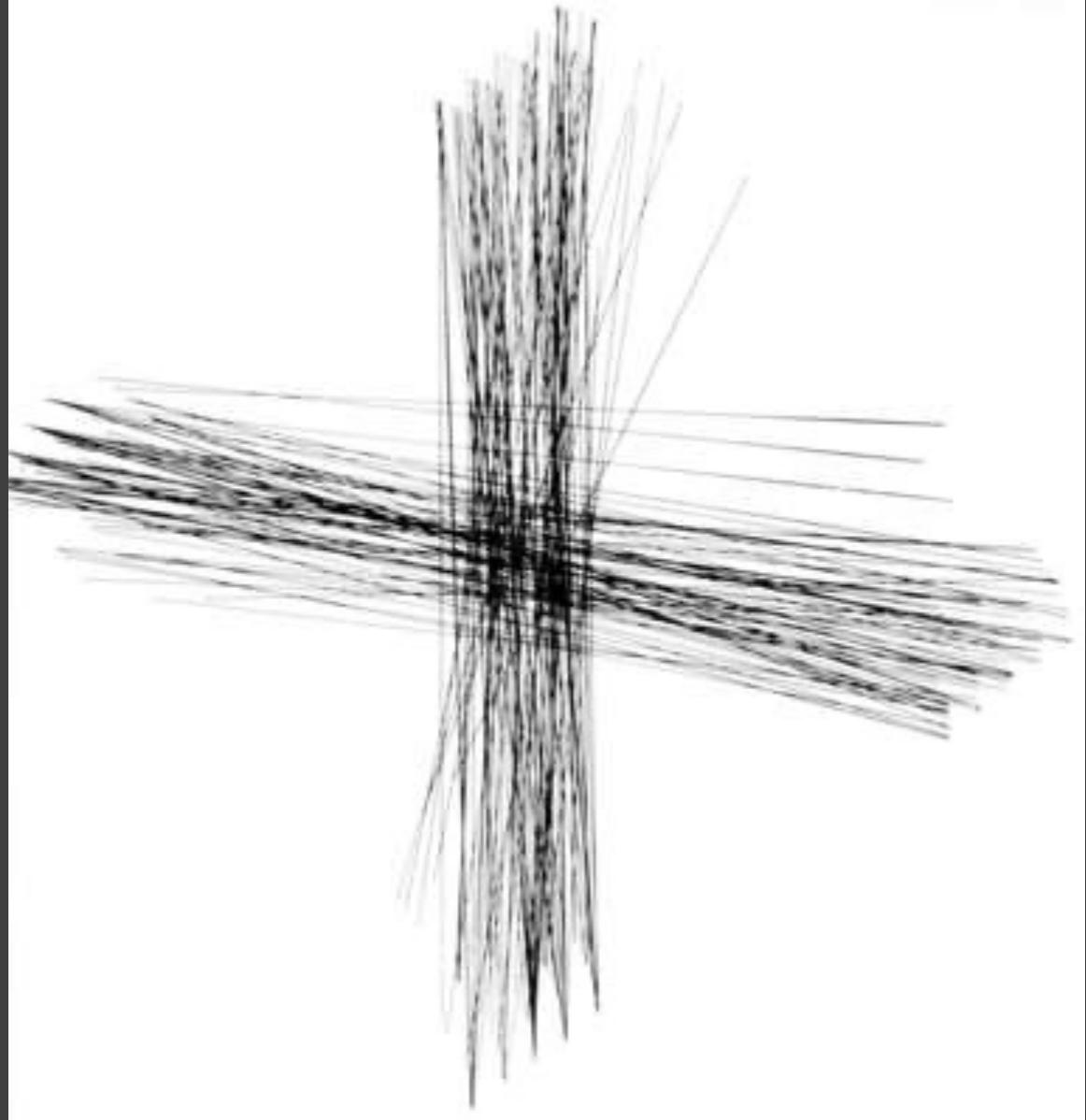
durchaus brauchbar, wenn wir die Zweierpotenzen 128, 256, 512 durch die Potenzen 2 hoch 32 bis 2 hoch 34 ersetzen. In die Programmiersprache ALGOL übersetzt, wird der Zufallsgenerator durch die Zeilen 6 bis 15 des folgenden Programms wiedergegeben:

(1) 2 'BEGIN' 'COMMENT' ZUFALLSFOLGE 1,
3 'INTEGER' JI, JS,,
4 'REAL' JA, JE,
5 'REAL' 'PROCEDURE' J,,
6 'BEGIN' 'COMMENT' ZUFALLS-
GENERATOR J,,
7 JI=5*JI,,
8 'IF' JI 'NOTLESS' 8589934592,
9 'THEN' JI.=JI-8589934592,,
10 'IF' JI 'NOTLESS' 4294967296
11 'THEN' JI.=JI-4294967296,,
12 'IF' JI 'NOTLESS' 2147483648
13 'THEN' JI.=JI-2147483648,,
14 J.=JI/2147483648*(JE-JA)+JA
15 'END' ZUFALLSGENERATOR J,,
16 JS.=1306859721,,
17 'BEGIN' 'REAL' U,,
18 JI.=JS,, OPEN(0,0),
19 JA.=0., JE.=90.,
20 'FOR' U.=0 'STEP' 2 'UNTIL' 258 'DO'
21 'BEGIN' LEER(U+.5,0),
LINE(U+.5,J),
22 LEER(U+1.5,J),
LINE(U+1.5,0)
23 'END', CLOSE
24 'END'
25 'END' ZUFALLSFOLGE 1,

In Programm (1) wird der Zufallsgenerator J durch eine

Generative Computergraphik

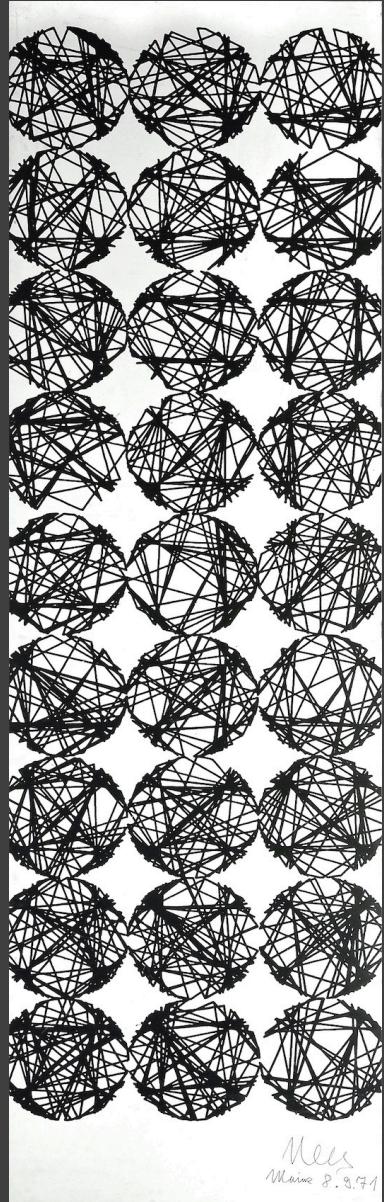
Andreaskreuz



Silkscreen

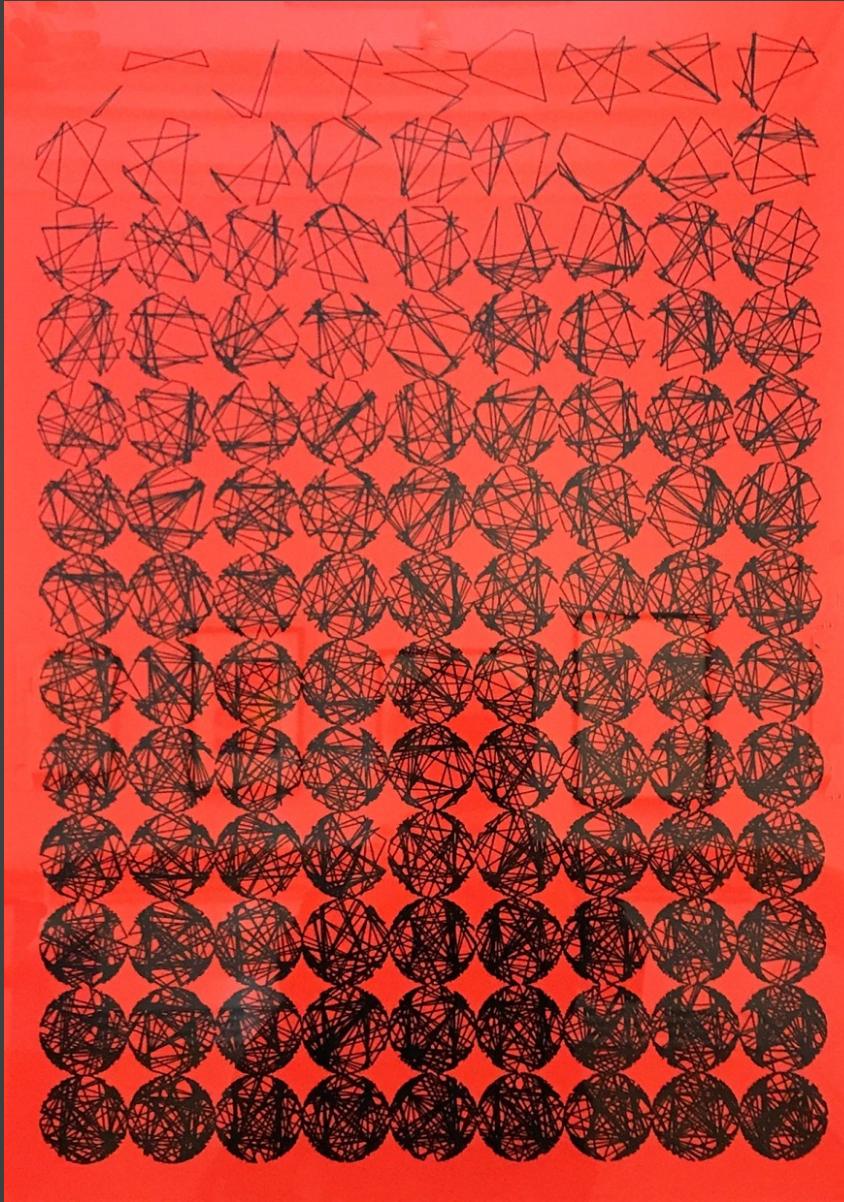


COMPUTERGRAFIK MIT SIEMENS-SYSTEM 4004
gezeichnet mit ZUSE-GRAFOMAT

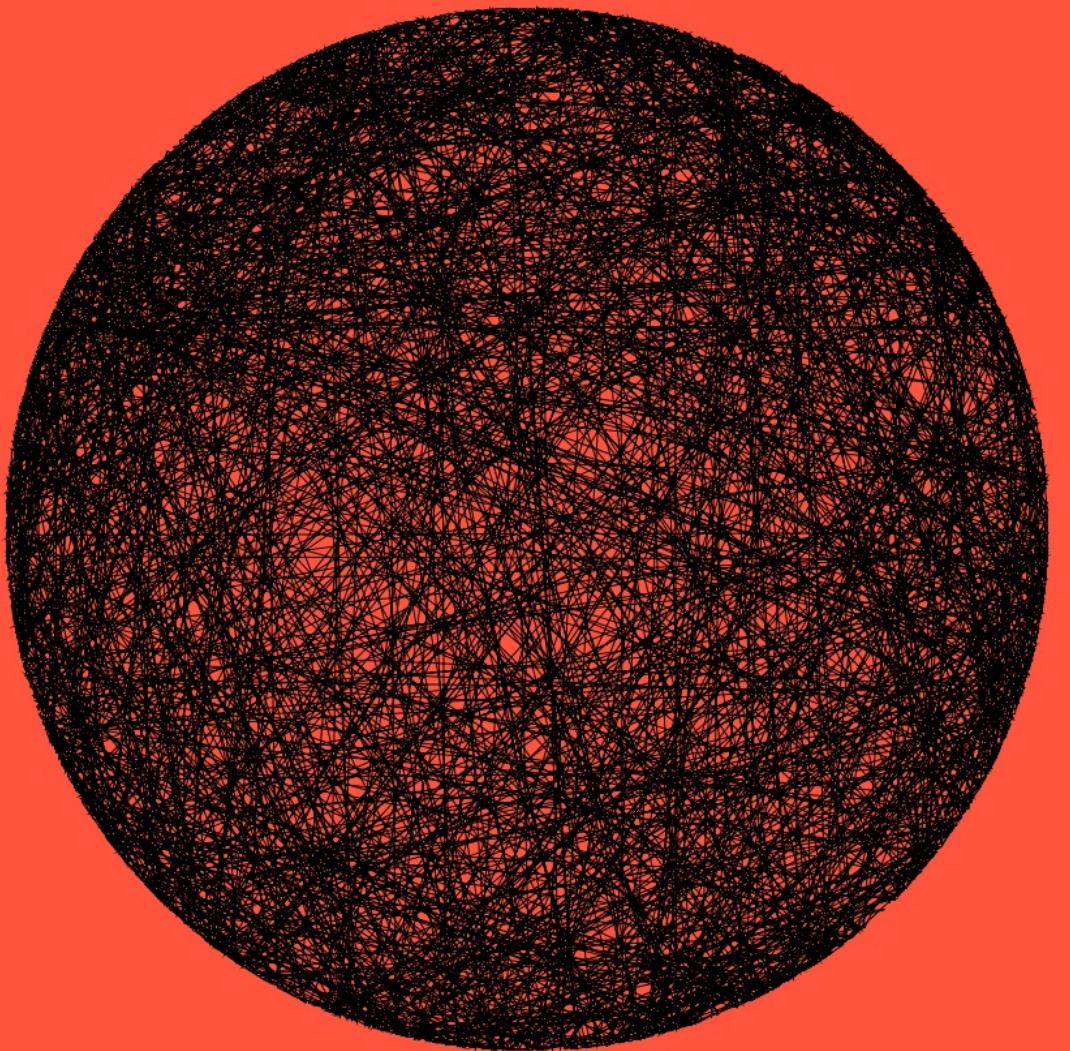


Untitled

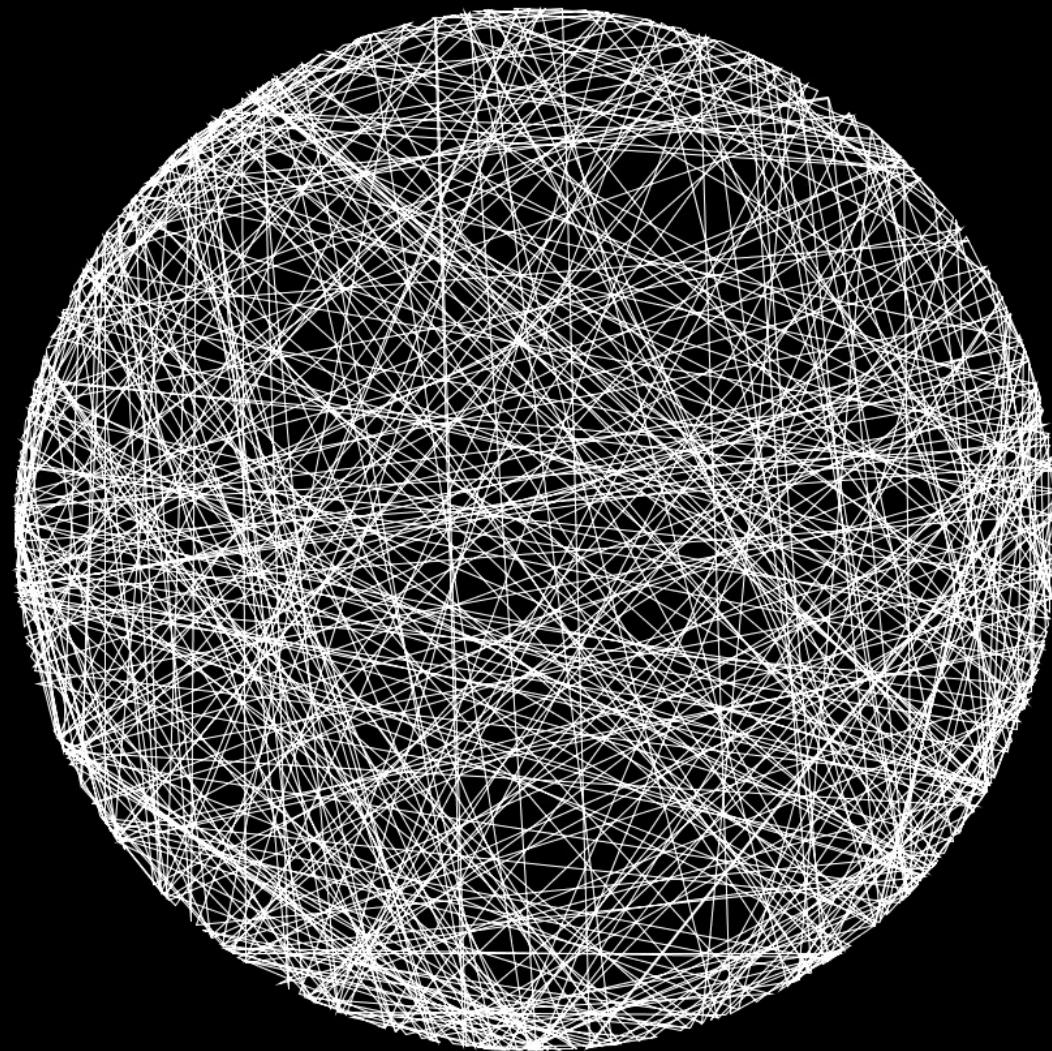
Untitled



Untitled_01_Georg_Ness_Audio_Reactive



Untitled_02_Georg_Ness_Audio_Reactive



Untitled_01_Georg_Ness_Audio_Reactive

```
import netP5.*;
import oscP5.*;

ArrayList<PVector> pts;
boolean stop = true;
int speed = 1;
int play = 1;
int direction = 1;
float m;

OscP5 oscP5;
NetAddress myRemoteLocation;

void setup() {
    size(1000,1000);
    smooth();
    noFill();
    pts = new ArrayList<PVector>();

    //use [prepend /test] in Max //
    //With "result" you define a name to construct a function inside Processing //
    oscP5 = new OscP5(this,12000);
    oscP5 plug(this,"result","/test");

}

void draw() {
    background(253,69,59,255);

    //frame rate controled by max//

    float m = map(speed, 0, 30, 1, 30);
    //println(m);
    frameRate(m);

    //Draws if playback speed <0, Erases if playbacks speed < 0//

    if (direction == -1){
        if (pts.size() > 0) pts.remove(pts.size()-1);
    }else{
        float radius = 400;
        float a = random(TWO_PI);
        float x = width/2+cos(a)*radius;
        float y = height/2+sin(a)*radius;
        pts.add(new PVector(x,y));
    }
}
```

```
beginShape();
for (int i = 0; i < pts.size(); i++){
    vertex(pts.get(i).x, pts.get(i).y);
}
endShape();

// The function must be public
public void result(int valueA, int valueB, int valueC) {
    //#[prepend /test] sending 3 messages from max via OSC//
    println("The three result will be: " + valueA + ", " + valueB + ", " + valueC);

    speed = valueA;
    play = valueB;
    direction = valueC;

    //Lets you play and pause drawing from max//
    if (play == 0){
        noLoop();
    }else{
        loop();
    }
}
```

