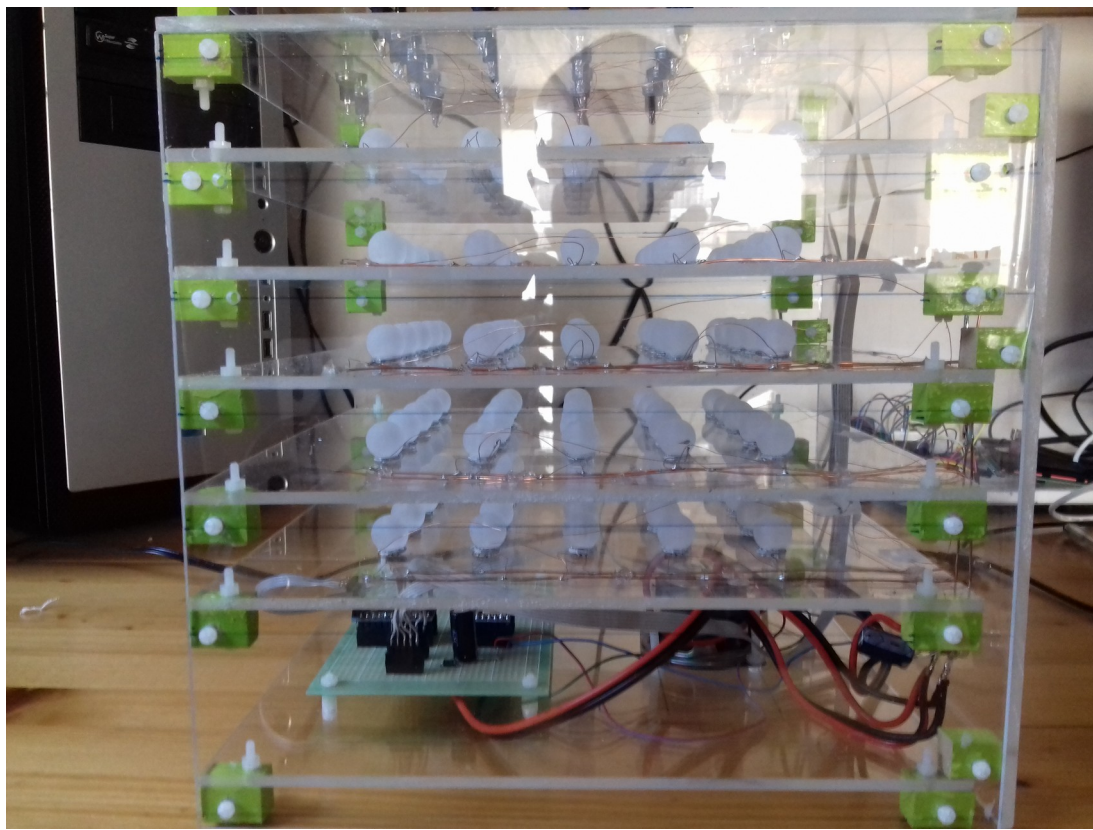


4 op 1 rij 3D



Onderdelen

1 x Arduino Mini Pro 5V

[https://nl.aliexpress.com/item/32821902128.html?
spm=a2g0o.productlist.main.5.43516e4536dFM5&algo_pvid=819301d5-e02d-4da7-9aab-7aae2a8d7724&algo_exp_id=819301d5-e02d-4da7-9aab-7aae2a8d7724-2&pdp_npi=3%40dis%21EUR%213.06%212.48%21%21%21%21%21%40211beca116822778264517938d0763%2167225925112%21sea%21BE%210&curPageLogUid=vw4chKLRJlzt](https://nl.aliexpress.com/item/32821902128.html?spm=a2g0o.productlist.main.5.43516e4536dFM5&algo_pvid=819301d5-e02d-4da7-9aab-7aae2a8d7724&algo_exp_id=819301d5-e02d-4da7-9aab-7aae2a8d7724-2&pdp_npi=3%40dis%21EUR%213.06%212.48%21%21%21%21%21%40211beca116822778264517938d0763%2167225925112%21sea%21BE%210&curPageLogUid=vw4chKLRJlzt)

1 x DF player mini

[https://nl.aliexpress.com/item/32427720062.html?
spm=a2g0o.productlist.main.5.6e7f3406P6oSd9&algo_pvid=79f5e113-1500-4a34-9519-a320a0e02dcc&algo_exp_id=79f5e113-1500-4a34-9519-a320a0e02dcc-2&pdp_npi=3%40dis%21EUR%211.13%210.91%21%21%21%21%21%40211bf3f116822779988055774d075c%2157508085057%21sea%21BE%210&curPageLogUid=mvRoMVyKh2ee](https://nl.aliexpress.com/item/32427720062.html?spm=a2g0o.productlist.main.5.6e7f3406P6oSd9&algo_pvid=79f5e113-1500-4a34-9519-a320a0e02dcc&algo_exp_id=79f5e113-1500-4a34-9519-a320a0e02dcc-2&pdp_npi=3%40dis%21EUR%211.13%210.91%21%21%21%21%21%40211bf3f116822779988055774d075c%2157508085057%21sea%21BE%210&curPageLogUid=mvRoMVyKh2ee)

1 x SD kaart

1 x luidspreker

27 x drukknop

[https://nl.aliexpress.com/item/1005003989505144.html?
spm=a2g0o.productlist.main.19.7ecc66f9v6k5sJ&algo_pvid=df9cc738-a22a-435b-a5b8-ceda301158b7&aem_p4p_detail=202304231227382502745775763520012749484&algo_exp_id=df9cc738-a22a-435b-a5b8-ceda301158b7-9&pdp_npi=3%40dis%21EUR%210.79%210.72%21%21%21%21%21%40211bea0816822780581346571d0760%2112000027671796726%21sea%21BE%210&curPageLogUid=b0iQAqMP56FT&ad_pvid=202304231227382502745775763520012749484_2&ad_pvid=202304231227382502745775763520012749484_2](https://nl.aliexpress.com/item/1005003989505144.html?spm=a2g0o.productlist.main.19.7ecc66f9v6k5sJ&algo_pvid=df9cc738-a22a-435b-a5b8-ceda301158b7&aem_p4p_detail=202304231227382502745775763520012749484&algo_exp_id=df9cc738-a22a-435b-a5b8-ceda301158b7-9&pdp_npi=3%40dis%21EUR%210.79%210.72%21%21%21%21%21%40211bea0816822780581346571d0760%2112000027671796726%21sea%21BE%210&curPageLogUid=b0iQAqMP56FT&ad_pvid=202304231227382502745775763520012749484_2&ad_pvid=202304231227382502745775763520012749484_2)

125 x WS2812B

[https://nl.aliexpress.com/item/33026835790.html?
spm=a2g0o.productlist.main.11.5de560d2ulnxRL&algo_pvid=3e6d91b1-d0ff-4bc9-9184-997662c4cfd1&algo_exp_id=3e6d91b1-d0ff-4bc9-9184-997662c4cfd1-5&pdp_npi=3%40dis%21EUR%212.9%212.44%21%21%21%21%21%40211be10916822781731326499d074a%2167286460865%21sea%21BE%210&curPageLogUid=oPtH2HA8Az83](https://nl.aliexpress.com/item/33026835790.html?spm=a2g0o.productlist.main.11.5de560d2ulnxRL&algo_pvid=3e6d91b1-d0ff-4bc9-9184-997662c4cfd1&algo_exp_id=3e6d91b1-d0ff-4bc9-9184-997662c4cfd1-5&pdp_npi=3%40dis%21EUR%212.9%212.44%21%21%21%21%21%40211be10916822781731326499d074a%2167286460865%21sea%21BE%210&curPageLogUid=oPtH2HA8Az83)

!!! LEDs eerst testen alvorens te lijmen !!!

125 x clear matte parels 16 mm

[https://nl.aliexpress.com/item/32982212917.html?
spm=a2g0o.store_pc_groupList.8148356.6.3e1c5697MTDgvI&pdp_npi=2%40dis%21USD%21US%20%2416.34%21US%20%2413.07%21%21%21%21%21%402103205316823631924997449e72d6%2166799261893%21sh](https://nl.aliexpress.com/item/32982212917.html?spm=a2g0o.store_pc_groupList.8148356.6.3e1c5697MTDgvI&pdp_npi=2%40dis%21USD%21US%20%2416.34%21US%20%2413.07%21%21%21%21%21%402103205316823631924997449e72d6%2166799261893%21sh)

koper wikkeldraad

<https://www.conrad.be/nl/p/block-koperdraad-gelakt-buitendiameter-excl-isolatielak-0-22-mm-571-m-0-20-kg-605311.html>

Transparante lijm

<https://www.conrad.com/p/pattex-multi-purpose-glue-100-p1bc6-50-g-892573>

houtboor 8mm

6 x plexiglas 245 x 245 x 4 (leds)

1 x plexiglas 255 x 245 x 4 (pushbuttons)

2 x plexiglas 255 x 280 x 4 (zijanten / sides)

2 x plexiglas 253 x 280 x 4 (voor en achterkant /front and back)

veel nylon bouten m3 x 20 en moeren m3

1 x veel tijd en geduld

BOUWEN

Ground Floor

Monteer hier de Arduino mini pro en DF player mini.
SD kaart opbouw

mp3 folder

- 0001_Alweer een winnaar.mp3
- 0002_5 4 3 2 are you readyyyy.mp3
- 0003_yabbadabbadoo.mp3
- 0004_stoomboot.mp3
- 0005_Tot ziens.mp3
- 0006_oh lala.mp3
- 0007_Nieuwe ronde.mp3
- 0008_Lage hoorn.mp3
- 0009_langzaam_lachen.mp3
- 0010_Ja wat is dat hier allemaal.mp3
- 0011_Vogeltjesdans.mp3
- 0012_tuut tuut.mp3
- 0012_Scream_1.mp3
- 0013_Opgepast voor de start.mp3

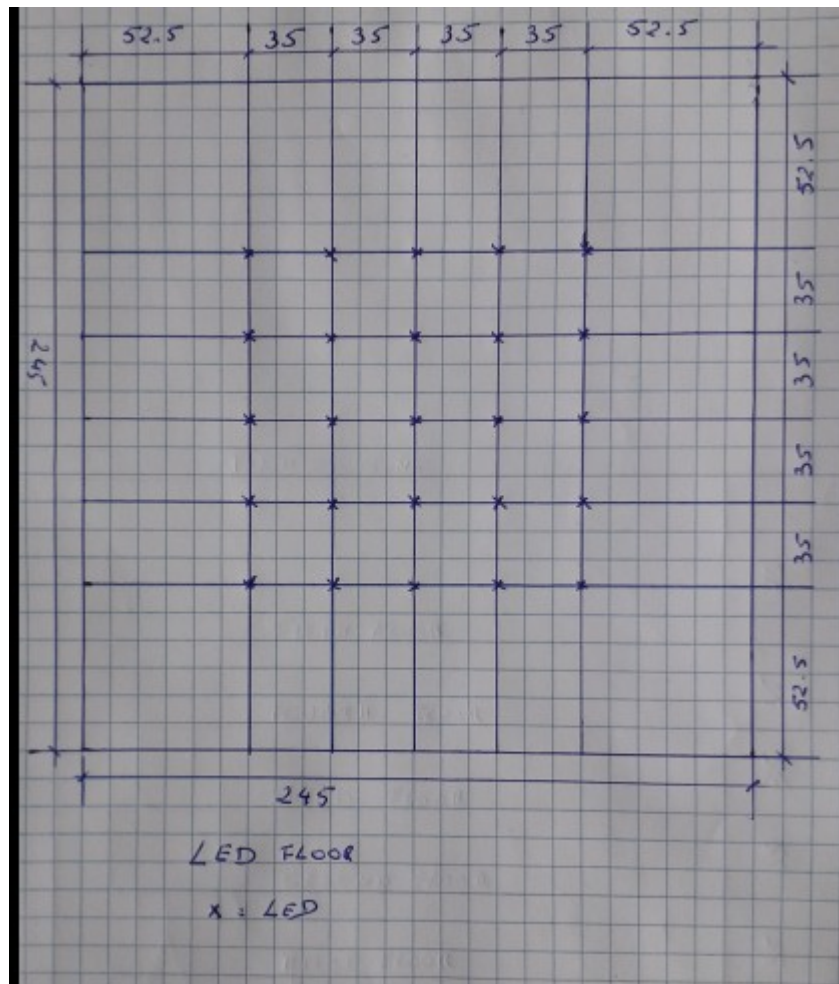
De geluidsbestanden vind je hier:

<https://peterdoina.nl/KERMIS-GELUIDEN.html>

Audacity kan helpen om te downloaden en op te slaan.

Foto's zijn duidelijker dan uitleg

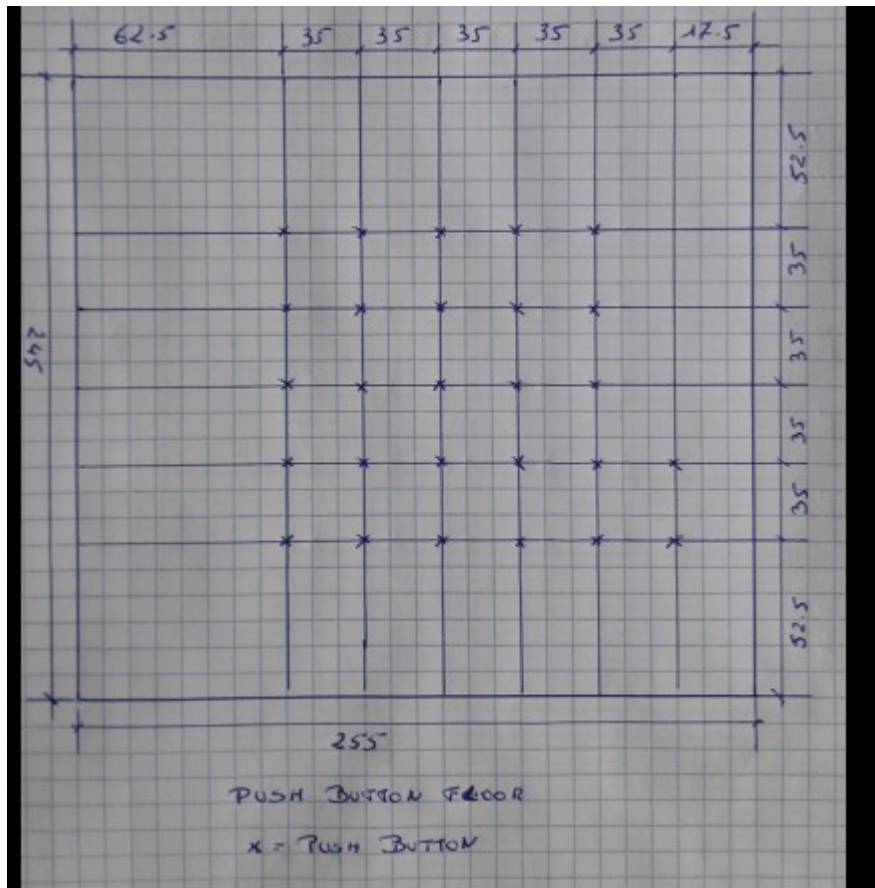
Tekeningen over de afstand tussen de verschillende leds, drukknoppen en verdiepingen



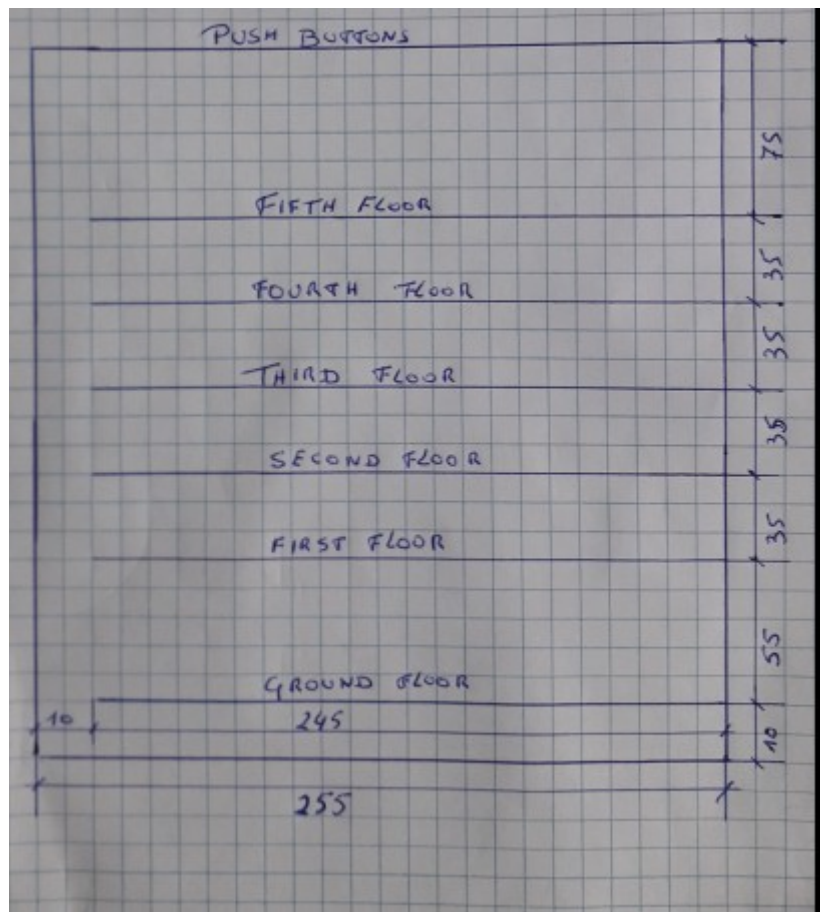
Tekening van een led vloer. In totaal heb je er zo 5 nodig. Afmetingen in millimeters.

Plexiplaat 245 x 245 x 4

Onderlinge afstand tussen de leds 35mm.



Tekening van drukknop vloer. Aangezien deze aan de bovenzijde ook het geheel afsluit is deze plexiplaat 10 mm langer.
Afmeting 255 x 245 x 4



Zijaanzicht van alle vloeren samengevoegd.

Afmetingen zijplaat (2x)

280 x 255 x 4

De ledvloeren worden aan de voorkant gelijk gemonteerd met het begin van de zijplaat, zo blijft er aan de achterkant 10 mm vrij voor de bedrading.

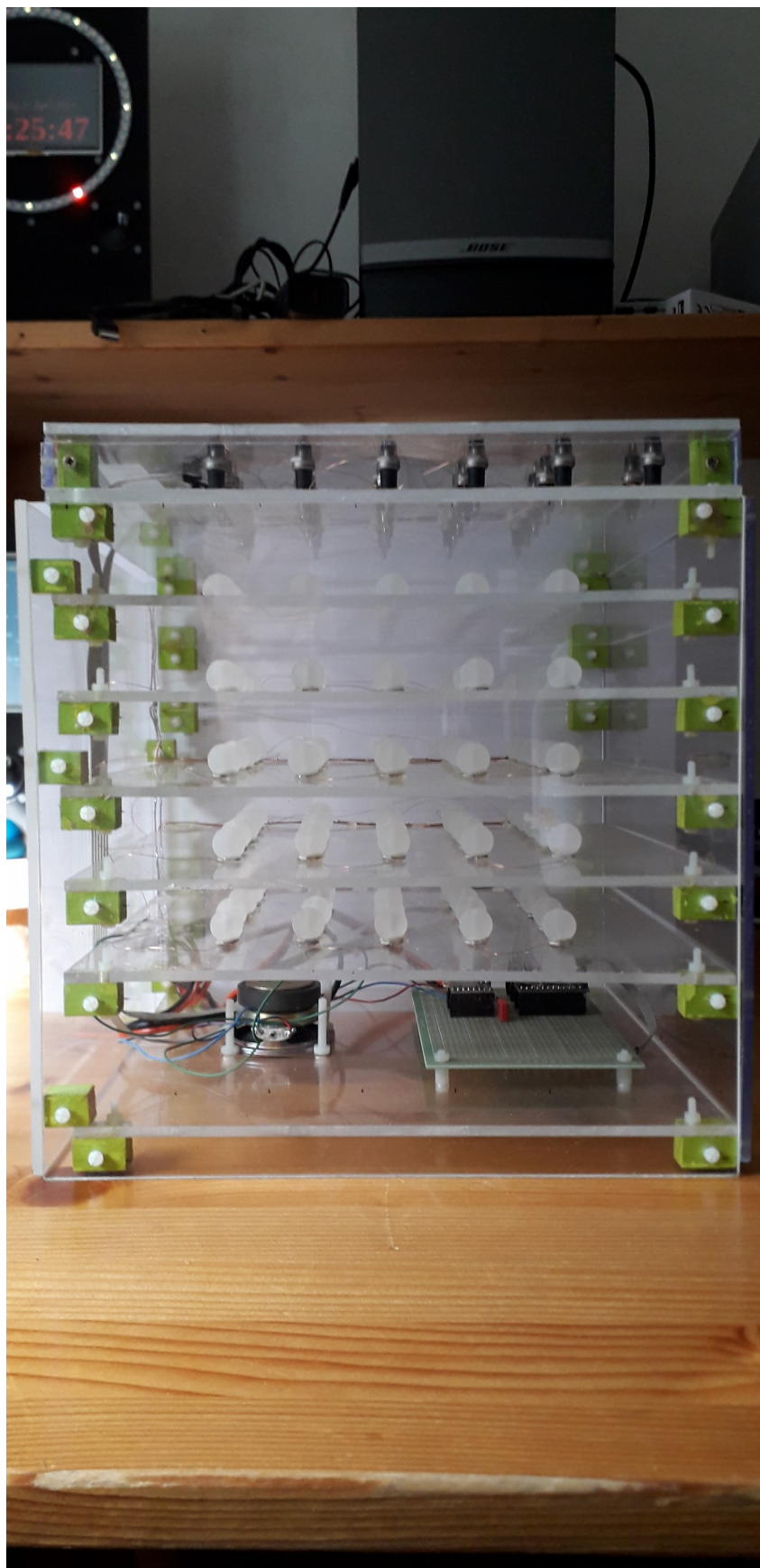
Drukknop vloer komt boven op de zijplaten.

Afbeeldingen van het geheel samengebouwd.









Plaats de leds volgens de opgegeven afstanden op een plexiplaat.

Verbind de verschillende leds met elkaar met behulp van de koperen wikkeldraad.

De vernis aan het begin van een draad is gemakkelijk te verwijderen door een klein drupje soldeertin aan de punt van de soldeerbout te doen en hierin het begin van een koperdraad te steken. Alleen dit puntje van de vernis smelt weg.

Het bedradingsschema vind je terug verderop in de handleiding.

Alvorens de leds vast te lijmen op een plexiplaat controleer eerst of ze branden.

Parels,

boor met behulp van een houtboor van 8mm een 3 mm diep gat in de parel. Gebruik een doorvoeropening als middelpunt.

Gebruik hiervoor geschikt gereedschap en houd zeker de parel niet met de hand vast tijdens het boren.

Eens de leds gelijmd zijn en het geheel nogmaals getest is, lijm dan de parels op de leds. Gebruik hiervoor transparante lijm, zoals aangegeven bij de onderdelen. Lijm ook verkrijgbaar in de Action.

Monteren van de verschillende vloeren volgens die aangegeven tussenruimtes.

De groene blokjes die je op foto ziet zijn geverfde houten blokjes van 20 x 10 x10.

Gezaagd van een lengte van 1000 x 10 x 10 die je in elke doe-het-zelf-zaak kan vinden.

Drukknoppen.

Plaats deze volgens de opgegeven afstanden in de daarvoor bestemde plexiplaat. De 2 extra drukknoppen komen aan de voorkant.

Bedrading van de drukknoppen zie schema's verderop.

Voor de rest valt er over de montage weinig te vertellen, bij twijfel kijk naar de foto's, dan zou het duidelijk moeten worden.

Programmeren

Laad het programma **vier_op_een_rij_1.ino** met behulp van de Arduino IDE in een arduino mini pro (16MHz, 5V).

Als de leds per groep van 4 beginnen te branden kan het spel beginnen.

Startknop	links voor
Resestknop	rechts voor

Elke drukknop boven een verticale rij indrukken laat in die rij een led vallen.

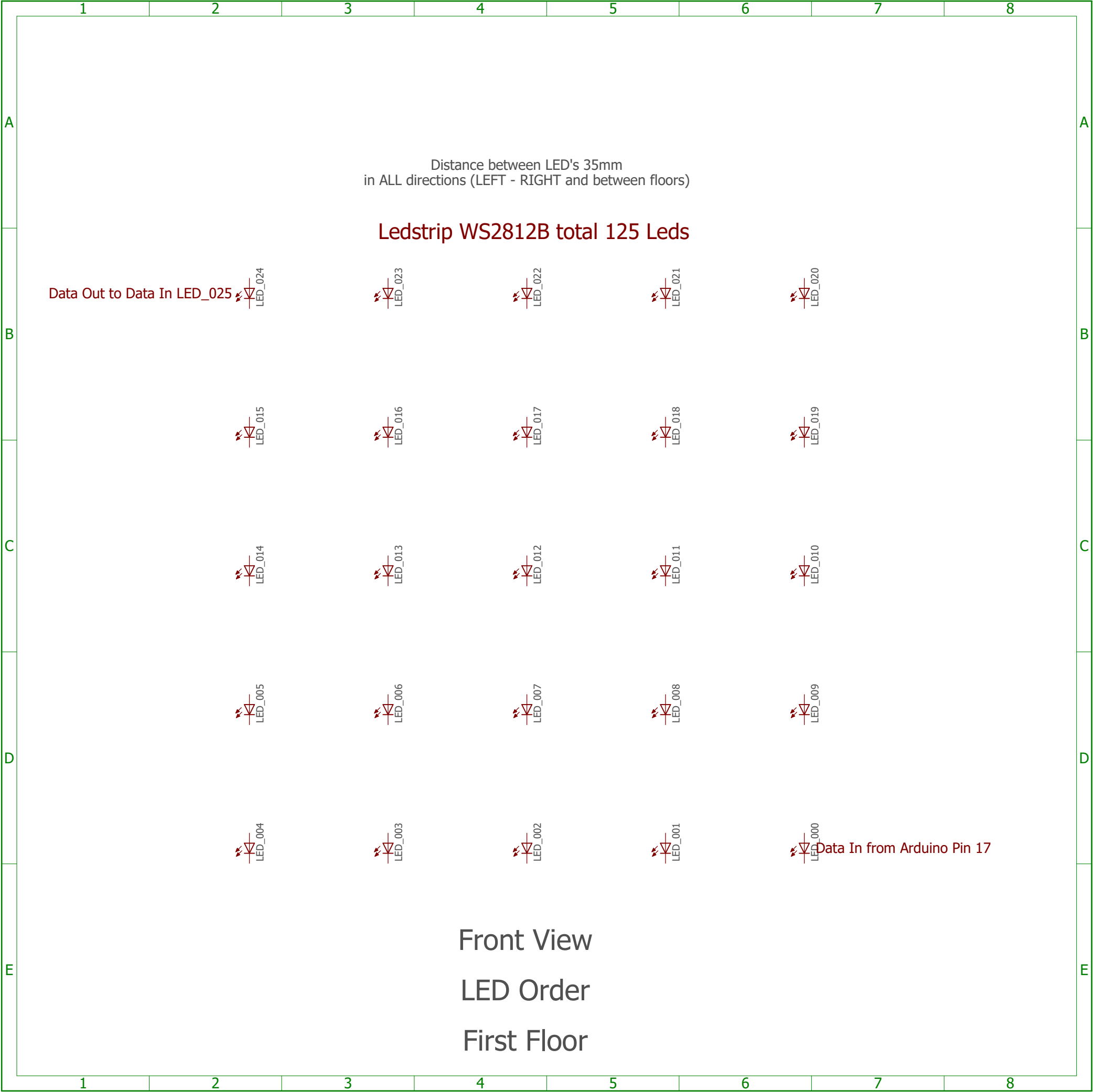
Speler wissel gebeurt automatisch

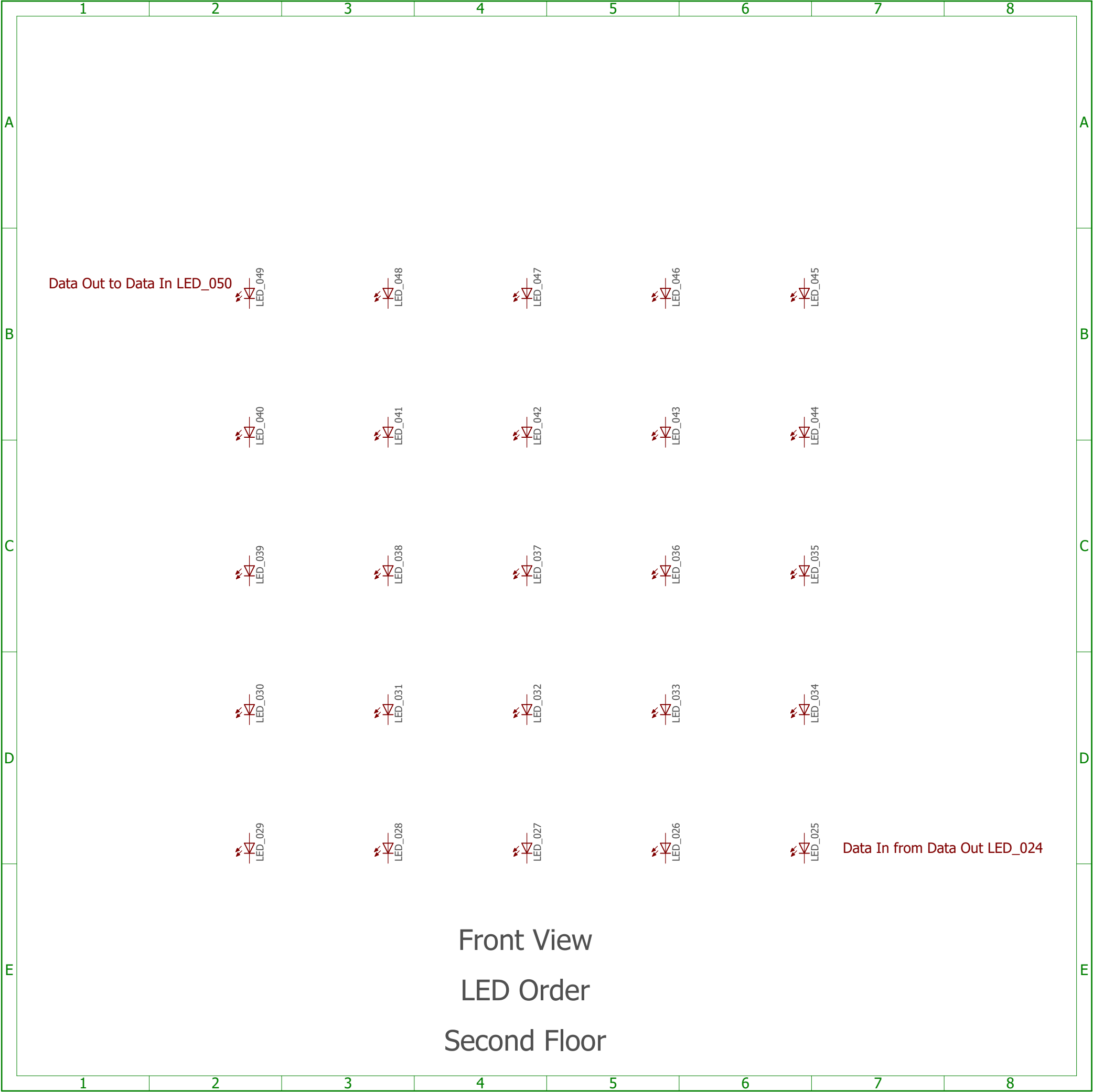
Controleren op 4 op een rij gebeurt ook automatisch.

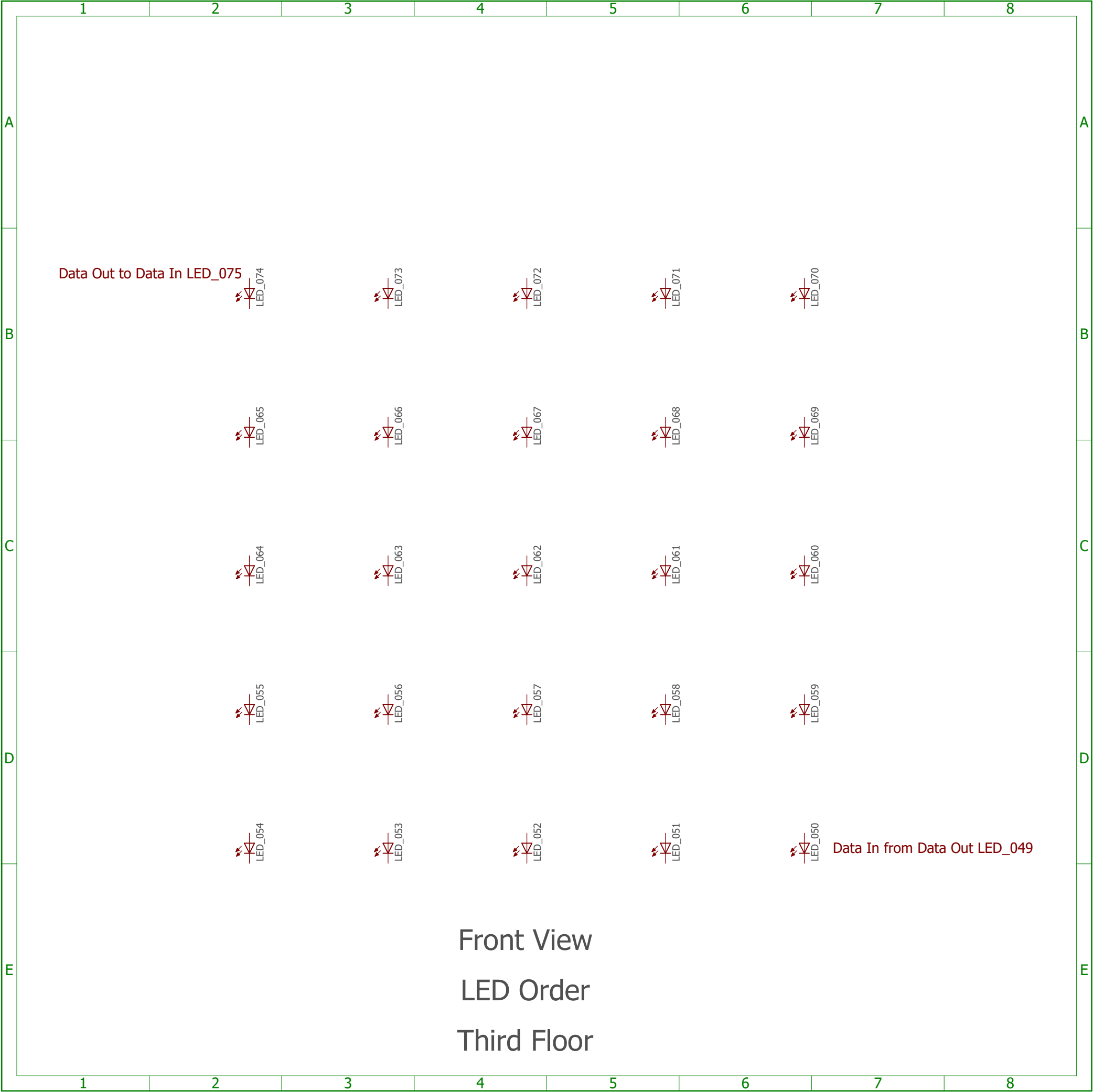
Spel herstarten met resetknop.

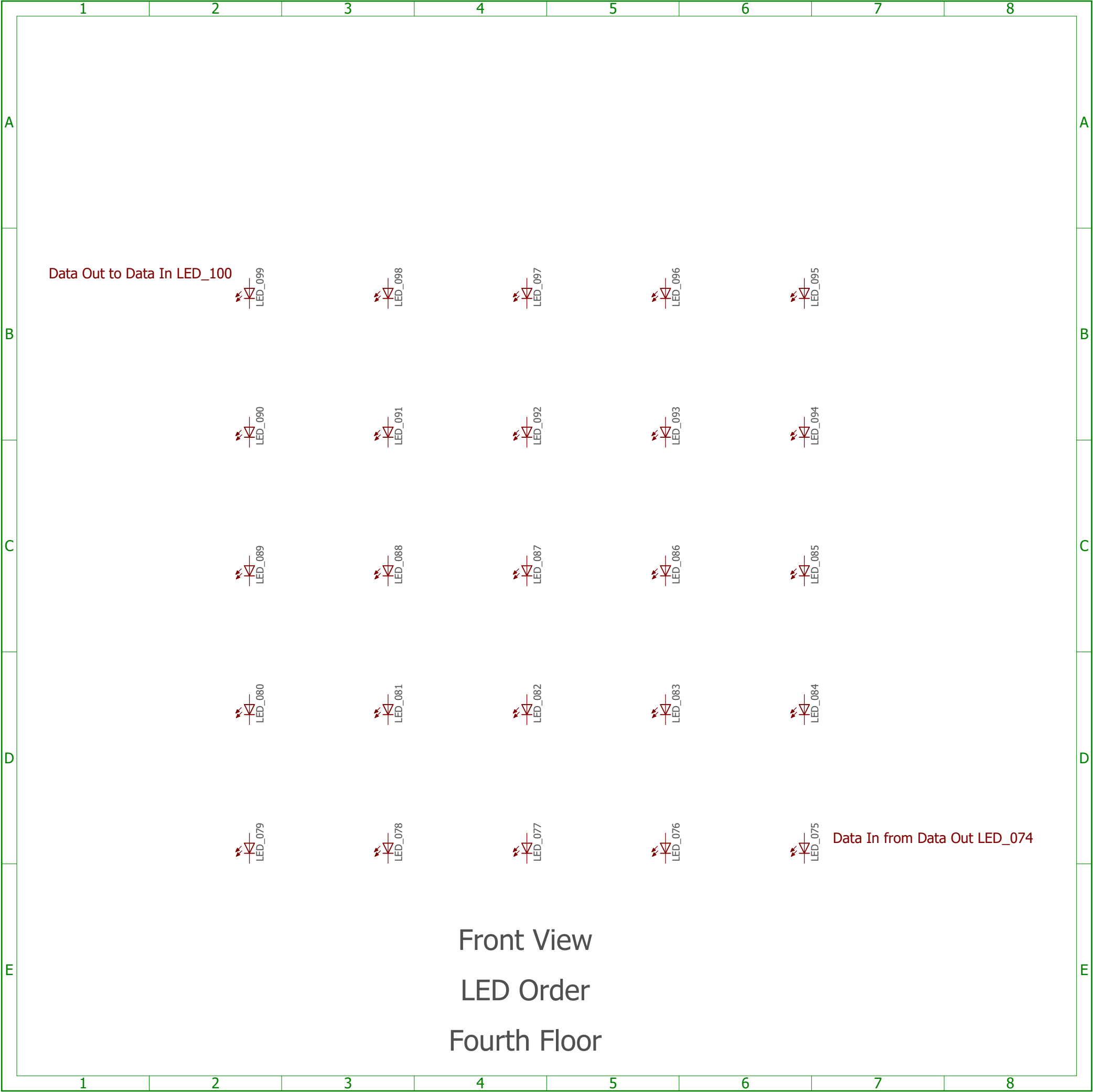
Veel plezier,

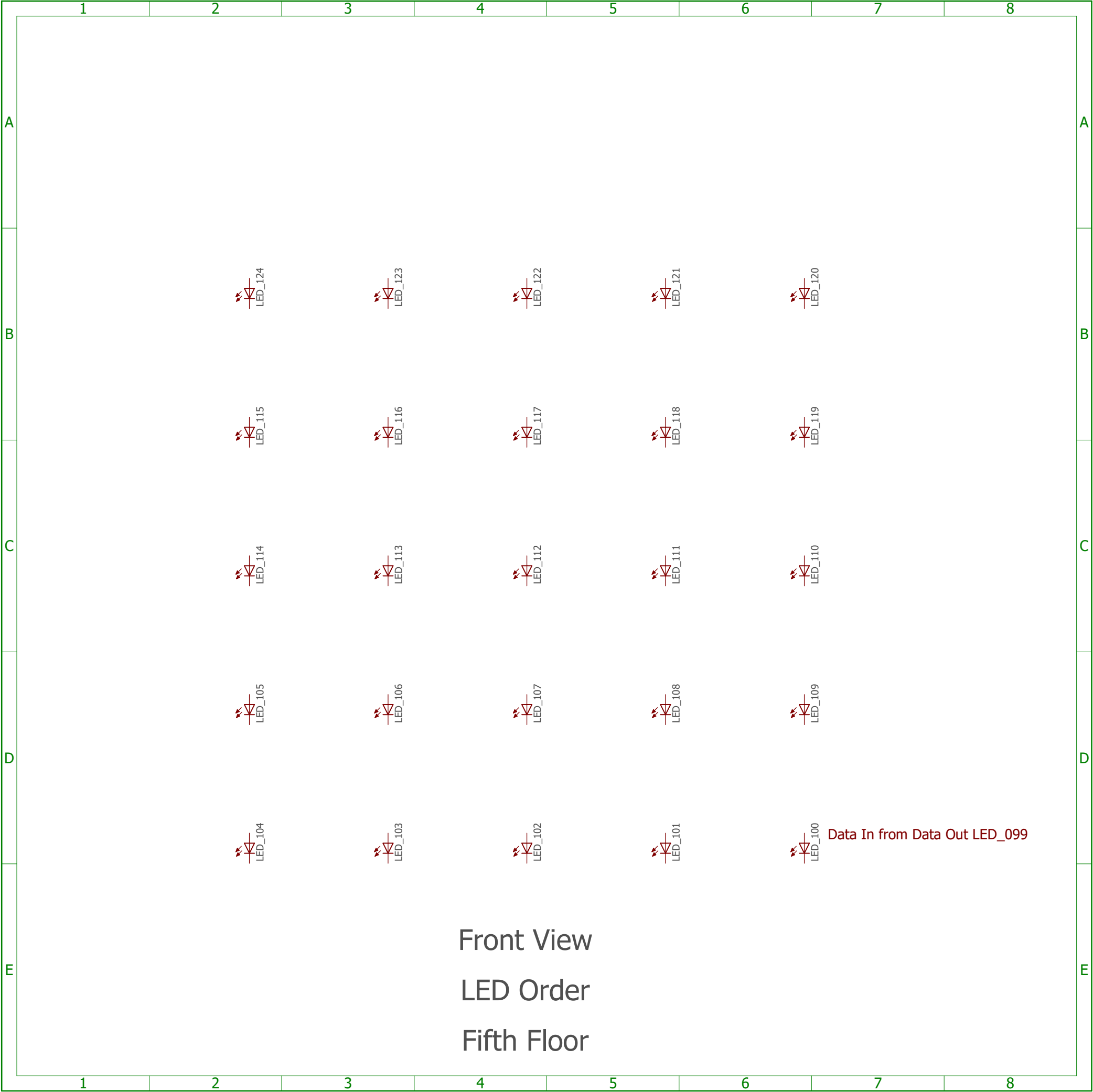
groeten,
thieu-b55

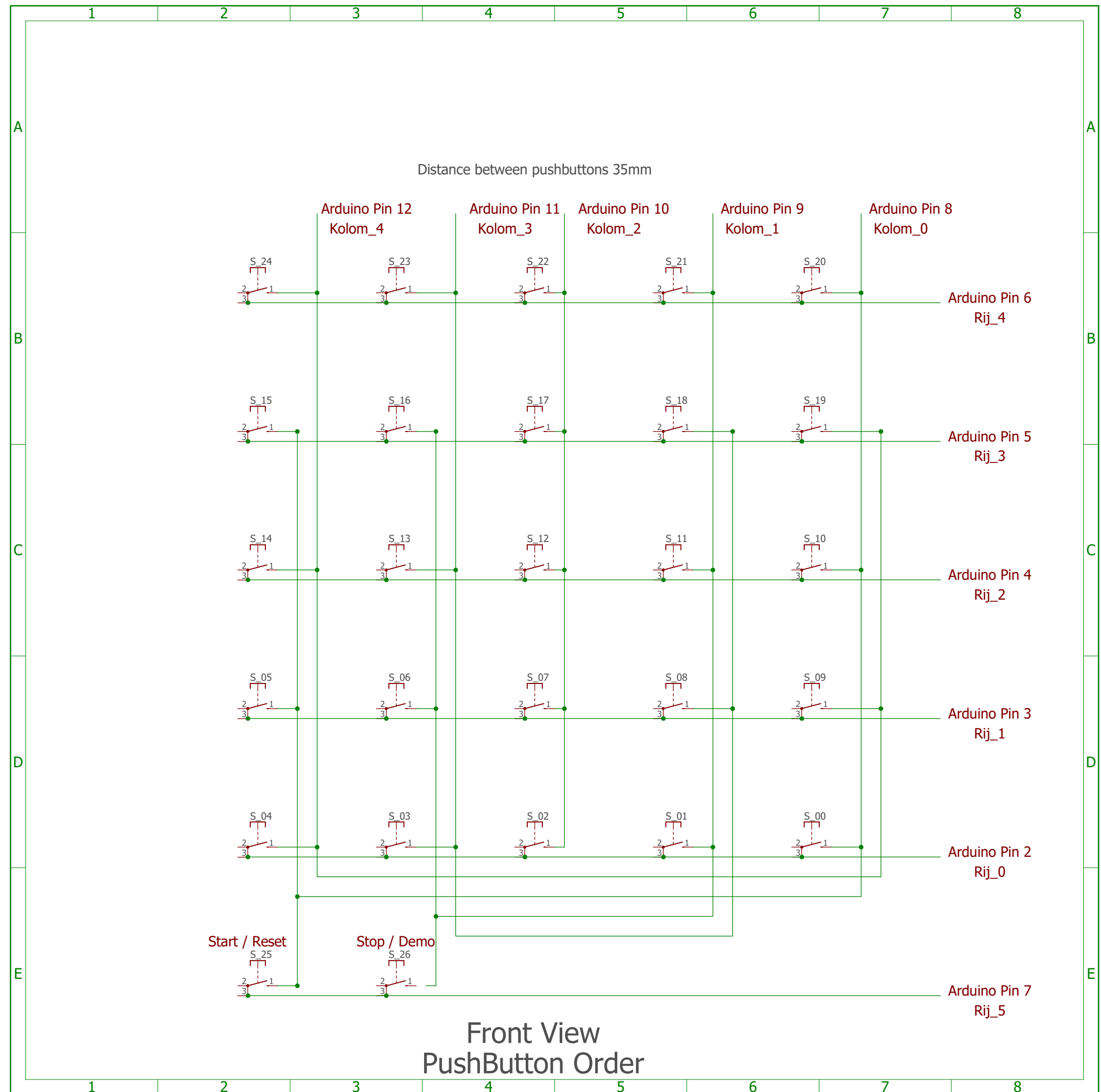
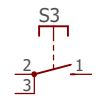
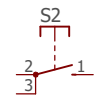
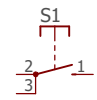


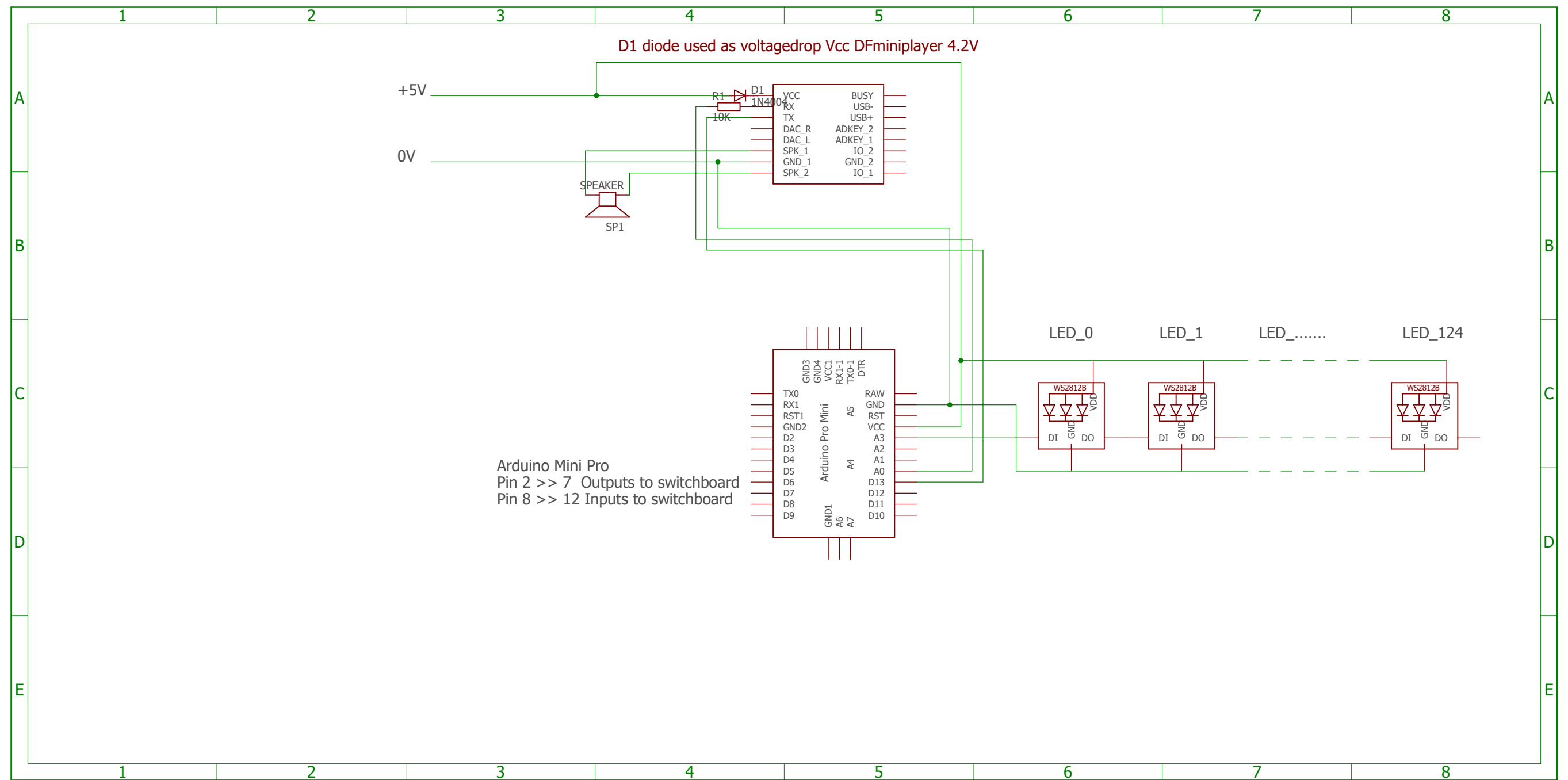












Programma

```
/*
 * 0001_Alweer een winnaar
 * 0002_5 4 3 2 are you readyyyy
 * 0003_yabbadabbadoo
 * 0004_stoomboot
 * 0005_Tot ziens
 * 0006_oh lala
 * 0007_Nieuwe ronde
 * 0008_Lage hoorn
 * 0009_langzaam lachen
 * 0010_Ja wat is dat hier allemaal
 * 0011_Vogeltjesdans
 * 0012_tuut tuut
 * 0012_Scream_1
 * 0013_Opgepast voor de start
 */
#include <avr/pgmspace.h>
#include <FastLED.h>
#include <DFPlayer_Mini_Mp3.h>
#include <SoftwareSerial.h>

#define NUM_LEDS 125
#define DATA_PIN 17
#define LED_TYPE WS2812B
#define COLOR_ORDER GRB
#define BRIGHTNESS 200
CRGB leds[NUM_LEDS];

#define RIJ_0 2
#define RIJ_1 3
#define RIJ_2 4
#define RIJ_3 5
#define RIJ_4 6
#define RIJ_5 7
#define KOLOM_0 8
#define KOLOM_1 9
#define KOLOM_2 10
#define KOLOM_3 11
#define KOLOM_4 12

PROGMEM const uint8_t tabel[] = {0, 9, 10, 19,      1, 8, 11, 18,      2, 7, 12, 17,      3, 6, 13, 16,      4, 5, 14, 15,
    25, 34, 35, 44,      26, 33, 36, 43,      27, 32, 37, 42,      28, 31, 38, 41,      29, 30, 39, 40,
    50, 59, 60, 69,      51, 58, 61, 68,      52, 57, 62, 67,      53, 56, 63, 66,      54, 55, 64, 65,
    75, 84, 85, 94,      76, 83, 86, 93,      77, 82, 87, 92,      78, 81, 88, 91,      79, 80, 89, 90,
    100, 109, 110, 119, 101, 108, 111, 118, 102, 107, 112, 117, 103, 106, 113, 116, 104, 105, 114, 115,
    120, 95, 70, 45,      121, 96, 71, 46,      122, 97, 72, 47,      123, 98, 73, 48,      124, 99, 74, 49,
    119, 94, 69, 44,      118, 93, 68, 43,      117, 92, 67, 42,      116, 91, 66, 41,      115, 90, 65, 40,
    110, 85, 60, 35,      111, 86, 61, 36,      112, 87, 62, 37,      113, 88, 63, 38,      114, 89, 64, 39,
    109, 84, 59, 34,      108, 83, 58, 33,      107, 82, 57, 32,      106, 81, 56, 31,      105, 80, 55, 30,
    100, 75, 50, 25,      101, 76, 51, 26,      102, 77, 52, 27,      103, 78, 53, 28,      104, 79, 54, 29,

    9, 10, 19, 20,      8, 11, 18, 21,      7, 12, 17, 22,      6, 13, 16, 23,      5, 14, 15, 24,
    34, 35, 44, 45,      33, 36, 43, 46,      32, 37, 42, 47,      31, 38, 41, 48,      30, 39, 40, 49,
    59, 60, 69, 70,      58, 61, 68, 71,      57, 62, 67, 72,      56, 63, 66, 73,      55, 64, 65, 74,
    84, 85, 94, 95,      83, 86, 93, 96,      82, 87, 92, 97,      81, 88, 91, 98,      80, 89, 90, 99,
    109, 110, 119, 120, 108, 111, 118, 121, 107, 112, 117, 122, 106, 113, 116, 123, 105, 114, 115, 124,
    95, 70, 45, 20,      96, 71, 46, 21,      97, 72, 47, 22,      98, 73, 48, 23,      99, 74, 49, 24,
    94, 69, 44, 19,      93, 68, 43, 18,      92, 67, 42, 17,      91, 66, 41, 16,      90, 65, 40, 15,
    85, 60, 35, 10,      86, 61, 36, 11,      87, 62, 37, 12,      88, 63, 38, 13,      89, 64, 39, 14,
    84, 59, 34, 9,      83, 58, 33, 8,      82, 57, 32, 7,      81, 56, 31, 6,      80, 55, 30, 5,
    75, 50, 25, 0,      76, 51, 26, 1,      77, 52, 27, 2,      78, 53, 28, 3,      79, 54, 29, 4,

    0, 1, 2, 3,      9, 8, 7, 6,      10, 11, 12, 13,      19, 18, 17, 16,      20, 21, 22, 23,
    25, 26, 27, 28,      34, 33, 32, 31,      35, 36, 37, 38,      44, 43, 42, 41,      45, 46, 47, 48,
    50, 51, 52, 53,      59, 58, 57, 56,      60, 61, 62, 63,      69, 68, 67, 66,      70, 71, 72, 73,
    75, 76, 77, 78,      84, 83, 82, 81,      85, 86, 87, 88,      94, 93, 92, 91,      95, 96, 97, 98,
    100, 101, 102, 103, 109, 108, 107, 106, 110, 111, 112, 113, 119, 118, 117, 116, 120, 121, 122, 123,
    1, 2, 3, 4,      8, 7, 6, 5,      11, 12, 13, 14,      18, 17, 16, 15,      21, 22, 23, 24,
    26, 27, 28, 29,      33, 32, 31, 30,      36, 37, 38, 39,      43, 42, 41, 40,      46, 47, 48, 49,
    51, 52, 53, 54,      58, 57, 56, 55,      61, 62, 63, 64,      68, 67, 66, 65,      71, 72, 73, 74,
    76, 77, 78, 79,      83, 82, 81, 80,      86, 87, 88, 89,      93, 92, 91, 90,      96, 97, 98, 99,
    101, 102, 103, 104, 108, 107, 106, 105, 111, 112, 113, 114, 118, 117, 116, 115, 121, 122, 123, 124,

    19, 11, 7, 3,      21, 17, 13, 5,      44, 36, 32, 28,      46, 42, 38, 30,      69, 61, 57, 53,
```

71, 67, 63, 55,	94, 86, 82, 78,	96, 92, 88, 80,	119, 111, 107, 103,	121, 117, 113, 105,
45, 71, 97, 123,	21, 47, 73, 99,	44, 68, 92, 116,	18, 42, 66, 90,	35, 61, 87, 113,
11, 37, 63, 89,	34, 58, 82, 106,	8, 32, 56, 80,	25, 51, 77, 103,	1, 27, 53, 79,
25, 59, 85, 119,	9, 35, 69, 95,	26, 58, 86, 118,	8, 36, 68, 96,	27, 57, 87, 117,
7, 37, 67, 97,	28, 56, 88, 116,	6, 38, 66, 98,	29, 55, 89, 115,	5, 39, 65, 99,
18, 12, 6, 4,	43, 37, 31, 29,	68, 62, 56, 54,	93, 87, 81, 79,	118, 112, 106, 104,
46, 72, 98, 124,	43, 67, 91, 115,	36, 62, 88, 114,	33, 57, 81, 105,	26, 52, 78, 104,
34, 60, 94, 120,	33, 61, 93, 121,	32, 62, 92, 122,	31, 63, 91, 123,	30, 64, 90, 124,
20, 18, 12, 6,	45, 43, 37, 31,	70, 68, 62, 56,	95, 93, 87, 81,	120, 118, 112, 106,
20, 46, 72, 98,	19, 43, 67, 91,	10, 36, 62, 88,	9, 33, 57, 81,	0, 26, 52, 78,
0, 34, 60, 94,	1, 33, 61, 93,	2, 32, 62, 92,	3, 31, 63, 91,	4, 30, 64, 90,
9, 11, 17, 23,	1, 7, 13, 15,	34, 36, 42, 48,	26, 32, 38, 40,	59, 61, 67, 73,
51, 57, 63, 65,	84, 86, 92, 98,	76, 82, 88, 90,	109, 111, 117, 123,	101, 107, 113, 115,
95, 71, 47, 23,	121, 97, 73, 49,	94, 68, 42, 16,	118, 92, 66, 40,	85, 61, 37, 13,
111, 87, 63, 39,	84, 58, 32, 6,	108, 82, 56, 30,	75, 51, 27, 3,	101, 77, 53, 29,
75, 59, 35, 19,	109, 85, 69, 45,	76, 58, 36, 18,	108, 86, 68, 46,	77, 57, 37, 17,
107, 87, 67, 47,	78, 56, 38, 16,	106, 88, 66, 48,	79, 55, 39, 15,	105, 89, 65, 49,
0, 8, 12, 16,	25, 33, 37, 41,	50, 58, 62, 66,	75, 83, 87, 91,	100, 108, 112, 116,
120, 96, 72, 48,	119, 93, 67, 41,	110, 86, 62, 38,	109, 83, 57, 31,	100, 76, 52, 28,
100, 84, 60, 44,	101, 83, 61, 43,	102, 82, 62, 42,	103, 81, 63, 41,	104, 80, 64, 40,
8, 12, 16, 24,	33, 37, 41, 49,	58, 62, 66, 74,	83, 87, 91, 99,	108, 112, 116, 124,
96, 72, 48, 24,	93, 67, 41, 15,	86, 62, 38, 14,	83, 57, 31, 5,	76, 52, 28, 4,
84, 60, 44, 20,	83, 61, 43, 21,	82, 62, 42, 22,	81, 63, 41, 23,	80, 64, 40, 24,
0, 33, 62, 91,	8, 37, 66, 99,	25, 58, 87, 116,	33, 62, 91, 124,	
20, 43, 62, 81,	18, 37, 56, 79,	45, 68, 87, 106,	43, 62, 81, 104,	
24, 41, 62, 83,	16, 37, 58, 75,	49, 66, 87, 108,	41, 62, 83, 100,	
4, 31, 62, 93,	6, 37, 68, 95,	29, 56, 87, 118,	31, 62, 93, 120,	
9, 36, 67, 98,	1, 32, 63, 90,	19, 36, 57, 78,	21, 42, 63, 80,	
23, 42, 61, 84,	15, 38, 57, 76,	3, 32, 61, 94,	5, 38, 67, 96,	
34, 61, 92, 123,	26, 57, 88, 115,	44, 61, 82, 103,	46, 67, 88, 105,	
48, 67, 86, 109,	40, 63, 82, 101,	28, 57, 86, 119,	30, 63, 92, 121,	

// kleur_1[0] en kleur_2[0] mogen niet hetzelfde zijn !!!!

```
uint8_t kleur_1[] = {0xff, 0x00, 0xff};
uint8_t kleur_2[] = {0xfe, 0xff, 0x00};
uint8_t kleurBuffer[3];
int pauze = 175;
uint16_t controle_getal_1;
uint16_t controle_getal_2;
int positie;
int gekozen;
bool kleur = 0;
bool demo = 1;
int demoPauze = 125;
int teller = 0;
bool eenmalig;
unsigned long millisVorig;
unsigned long millisVorig1;
```

```
void setup() {
  delay(2000);
  Serial.begin(115200);
  SoftwareSerial mySerial(13, 14);
  mySerial.begin(9600);
  FastLED.addLeds<NEOPIXEL, DATA_PIN>(leds, NUM_LEDS);
  FastLED.setBrightness(BRIGHTNESS);
  pinMode(RIJ_0, OUTPUT);
  pinMode(RIJ_1, OUTPUT);
  pinMode(RIJ_2, OUTPUT);
  pinMode(RIJ_3, OUTPUT);
  pinMode(RIJ_4, OUTPUT);
  pinMode(RIJ_5, OUTPUT);
  pinMode(KOLOM_0, INPUT_PULLUP);
  pinMode(KOLOM_1, INPUT_PULLUP);
  pinMode(KOLOM_2, INPUT_PULLUP);
  pinMode(KOLOM_3, INPUT_PULLUP);
  pinMode(KOLOM_4, INPUT_PULLUP);
  digitalWrite(RIJ_0, 1);
  digitalWrite(RIJ_1, 1);
  digitalWrite(RIJ_2, 1);
  digitalWrite(RIJ_3, 1);
  digitalWrite(RIJ_4, 1);
  digitalWrite(RIJ_5, 1);
```

```

controle_getal_1 = kleur_1[0] * 4;
controle_getal_2 = kleur_2[0] * 4;
randomSeed(analogRead(A0));
mp3_set_serial(mySerial);
mp3_set_volume(20);

}

void loop() {
  eenmalig = 0;
  millisVorig1 = millis();
  while(demo == 1){
    if(((millis() - millisVorig1) > 20000) && (eenmalig == 0)){
      eenmalig = 1;
      mp3_play(11);
    }
    wisLeds();
    demo_display();
    millisVorig = millis();
    while((millis() - millisVorig) < demoPauze){
      if(digitalRead(KOLOM_0) == 0){
        delay(5);
        if(digitalRead(KOLOM_0) == 0){
          demo = 0;
          wisLeds();
          teller = 0;
          mp3_play(13);
          while(digitalRead(KOLOM_0) == 0){
            delay(1);
          }
          digitalWrite(RIJ_5, 1);
        }
      }
    }
  }
}

gekozen = 0;
while(gekozen == 0){
  positie = 99;
  digitalWrite(RIJ_0, 0);
  leesInput();
  digitalWrite(RIJ_0, 1);
  digitalWrite(RIJ_1, 0);
  leesInput();
  digitalWrite(RIJ_1, 1);
  digitalWrite(RIJ_2, 0);
  leesInput();
  digitalWrite(RIJ_2, 1);
  digitalWrite(RIJ_3, 0);
  leesInput();
  digitalWrite(RIJ_3, 1);
  digitalWrite(RIJ_4, 0);
  leesInput();
  digitalWrite(RIJ_4, 1);
  digitalWrite(RIJ_5, 0);
  leesInput();
  digitalWrite(RIJ_5, 1);
}
}

void demo_display(){
  uint8_t demokleur[4];
  digitalWrite(RIJ_5, 0);
  demokleur[0] = random(255);
  demokleur[1] = random(25);
  demokleur[2] = random(255);
  leds[pgm_read_byte_near(tabel + teller)].setRGB(demokleur[0], demokleur[1], demokleur[2]);
  leds[pgm_read_byte_near(tabel + teller + 1)].setRGB(demokleur[0], demokleur[1], demokleur[2]);
  leds[pgm_read_byte_near(tabel + teller + 2)].setRGB(demokleur[0], demokleur[1], demokleur[2]);
  leds[pgm_read_byte_near(tabel + teller + 3)].setRGB(demokleur[0], demokleur[1], demokleur[2]);
  FastLED.show();
  teller = teller + 4;
  if(teller > 1079){
    teller = 0;
  }
}

void leesInput(){
  if(gekozen == 0){

```

```

    positie++;
    if(digitalRead(KOLOM_0) == 0){
        delay(5);
        if(digitalRead(KOLOM_0) == 0){
            plaatsKeuze();
            while(digitalRead(KOLOM_0) == 0){
                delay(1);
            }
        }
    }
}
if(gekozen == 0){
    positie++;
    if(digitalRead(KOLOM_1) == 0){
        delay(5);
        if(digitalRead(KOLOM_1) == 0){
            plaatsKeuze();
            while(digitalRead(KOLOM_1) == 0){
                delay(1);
            }
        }
    }
}
if(gekozen == 0){
    positie++;
    if(digitalRead(KOLOM_2) == 0){
        delay(5);
        if(digitalRead(KOLOM_2) == 0){
            plaatsKeuze();
            while(digitalRead(KOLOM_2) == 0){
                delay(1);
            }
        }
    }
}
if(gekozen == 0){
    positie++;
    if(digitalRead(KOLOM_3) == 0){
        delay(5);
        if(digitalRead(KOLOM_3) == 0){
            plaatsKeuze();
            while(digitalRead(KOLOM_3) == 0){
                delay(1);
            }
        }
    }
}
if(gekozen == 0){
    positie++;
    if(digitalRead(KOLOM_4) == 0){
        delay(5);
        if(digitalRead(KOLOM_4) == 0){
            plaatsKeuze();
            while(digitalRead(KOLOM_4) == 0){
                delay(1);
            }
        }
    }
}
}

void plaatsKeuze(){
    if(positie < 125){
        if(leds[positie].r == 0){
            if(kleur == LOW){
                leds[positie].setRGB(kleur_1[0], kleur_1[1], kleur_1[2]);
            }
            if(kleur == HIGH){
                leds[positie].setRGB(kleur_2[0], kleur_2[1], kleur_2[2]);
            }
            FastLED.show();
            if(leds[positie - 25].r == 0){
                leds[positie - 25].setRGB(leds[positie].r, leds[positie].g, leds[positie].b);
                leds[positie].setRGB(0, 0, 0);
                delay(pauze);
                FastLED.show();
            }
            if(leds[positie - 50].r == 0){
                leds[positie - 50].setRGB(leds[positie - 25].r, leds[positie - 25].g, leds[positie - 25].b);
                leds[positie - 25].setRGB(0, 0, 0);
            }
        }
    }
}

```



```

    delay(pauze);
    FastLED.show();
    if(leds[positie - 75].r == 0){
        leds[positie - 75].setRGB(leds[positie - 50].r, leds[positie - 50].g, leds[positie - 50].b);
        leds[positie - 50].setRGB(0, 0, 0);
        delay(pauze);
        FastLED.show();
        if(leds[positie - 100].r == 0){
            leds[positie - 100].setRGB(leds[positie - 75].r, leds[positie - 75].g, leds[positie - 75].b);
            leds[positie - 75].setRGB(0, 0, 0);
            delay(pauze);
            FastLED.show();
        }
    }
}
}
}
gekozen = 1;
kleur = kleur ^ 1;
vier_op_een_rij();
}
else{
    mp3_play(10);
    kleurBuffer[0] = leds[positie].r;
    kleurBuffer[1] = leds[positie].g;
    kleurBuffer[2] = leds[positie].b;
    for(int x = 0; x < 5; x++){
        leds[positie] = CRGB::Black;
        FastLED.show();
        delay(500);
        leds[positie] = CRGB::Red;
        FastLED.show();
        delay(500);
    }
    leds[positie] = CRGB::Black;
    FastLED.show();
    delay(500);
    leds[positie].setRGB(kleurBuffer[0], kleurBuffer[1], kleurBuffer[2]);
    FastLED.show();
}
}
else if(positie == 125){
    mp3_play(6);
    wisLeds();
    teller = 0;
}
else if(positie == 126){
    mp3_play(5);
    gekozen = 1;
    wisLeds();
    demo = 1;
    teller = 0;
}
}
}

void vier_op_een_rij(){
    uint8_t led1;
    uint8_t led2;
    uint8_t led3;
    uint8_t led4;
    uint8_t rbuf[4];
    uint8_t gbuf[4];
    uint8_t bbuf[4];
    int tt = 0;
    int punt = 0;
    uint16_t som;
    while((tt < 1192) && (punt == 0)){
        led1 = pgm_read_byte_near(tabel + tt);
        led2 = pgm_read_byte_near(tabel + tt + 1);
        led3 = pgm_read_byte_near(tabel + tt + 2);
        led4 = pgm_read_byte_near(tabel + tt + 3);
        uint8_t k = leds[led1].r;
        uint8_t l = leds[led2].r;
        uint8_t m = leds[led3].r;
        uint8_t n = leds[led4].r;
        som = k + l + m + n;
        if((som == controle_getal_1) || (som == controle_getal_2)){
            punt = 1;
        }
        tt = tt + 4;
    }
}

```

```

}
if(punt == 1){
  mp3_play(1);
  punt = 0;
  rbuf[0] = leds[led1].r;
  gbuf[0] = leds[led1].g;
  bbuf[0] = leds[led1].b;
  rbuf[1] = leds[led2].r;
  gbuf[1] = leds[led2].g;
  bbuf[1] = leds[led2].b;
  rbuf[2] = leds[led3].r;
  gbuf[2] = leds[led3].g;
  bbuf[2] = leds[led3].b;
  rbuf[3] = leds[led4].r;
  gbuf[3] = leds[led4].g;
  bbuf[3] = leds[led4].b;
  for(int x = 0; x < 5; x++){
    leds[led1] = CRGB::Blue;
    leds[led2] = CRGB::Blue;
    leds[led3] = CRGB::Blue;
    leds[led4] = CRGB::Blue;
    FastLED.show();
    delay(500);
    leds[led1].setRGB(rbuf[0], gbuf[0], bbuf[0]);
    leds[led2].setRGB(rbuf[1], gbuf[1], bbuf[1]);
    leds[led3].setRGB(rbuf[2], gbuf[2], bbuf[2]);
    leds[led4].setRGB(rbuf[3], gbuf[3], bbuf[3]);
    FastLED.show();
    delay(500);
  }
  wisLeds();
  mp3_play(4);
}
}

void wisLeds(){
  for(int wis = 0; wis < NUM_LEDS; wis++){
    leds[wis] = CRGB::Black;
  }
  FastLED.show();
  kleur = 0;
}

```