

Das magische Auge

Beschreibung einer Nachbildung

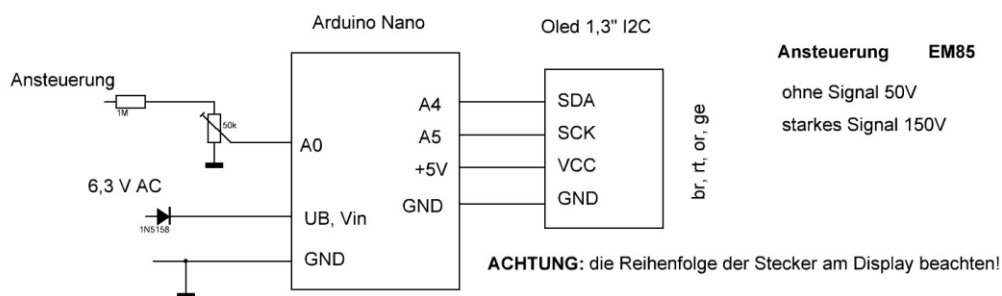
Gute Originale sind inzwischen kaum noch zu beschaffen. Will man ein altes Röhren-Radio reparieren, dann kann das magische Auge oft nicht ersetzt werden. Abhilfe gibt es jedoch:

Die Nachbildung mit einem Arduino Nano und einem I2C-Oled-Display kostet nicht viel. Man kann unter 10 EUR bleiben. Dagegen werden noch halbwegs brauchbare Ersatzröhren erst ab 40 EUR aufwärts angeboten.

Viele verzichten auf dieses wunderbare Teil, das zur korrekten Abstimmung auf einen Sender dienlich ist. Vor sechzig Jahren hatten fast alle Radios so eine grün leuchtende Anzeige, die durch ihre Leuchtkraft wie ein magisches Auge aussieht. Diese runden oder länglichen Elektronkröhren strahlten ein geheimnisvolles Licht aus. Wenn man einen Sender suchte, öffnete sich ein grüner Fächer oder eine Balkenanzeige. Die Leute waren davon fasziniert. Viel später wurden sie in der Audiotechnik auch als Pegelanzeige angewendet. Diese Augen folgen der Lautstärke fast trägheitslos. Allerdings werden die Leuchtschirme ständig von Elektronen bombardiert und ihre Leuchtkraft lässt nach einigen Jahren rapide nach, sodass man kaum noch die Konturen der Fächer oder Leucht balken erkennen kann. Solche magischen Augen bleiben dunkel und sind fortan keine Hilfe mehr für den Radiohörer.

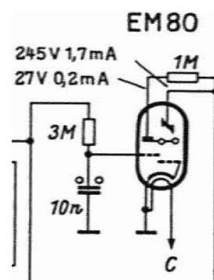
Hier stelle ich nun eine Ersatzlösung vor. Dieses kleine Bastelprojekt kann von jedem nachgebaut werden, der einen

LötKolben in der Werkzeugkiste hat und der sich nicht scheut, einfache Software mit dem PC zu erstellen. Die Entwicklungsumgebung, die IDE, kann man kostenfrei aus dem Netz herunterladen. Nach der Einbindung der Library wird kompiliert und auf den Mikroprozessor hochgeladen. Ein wenig Hardware drumherum ist simpel. Die Bauteile hat fast jeder Bastler vorrätig. Und schon kommt das Display zu einem sinnvollen Einsatz und zeigt nicht nur die Begrüßung: „Hello world“. So ein Arduino Nano kostet bei uns 3,50 EUR. Das Display etwa gleich viel. Man braucht noch einen Widerstand und ein Trimpoti. Hier die Schaltung:

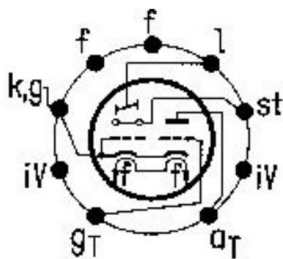


Magisches Auge

Wir nehmen die Steuerspannung bei(27V...) ab:

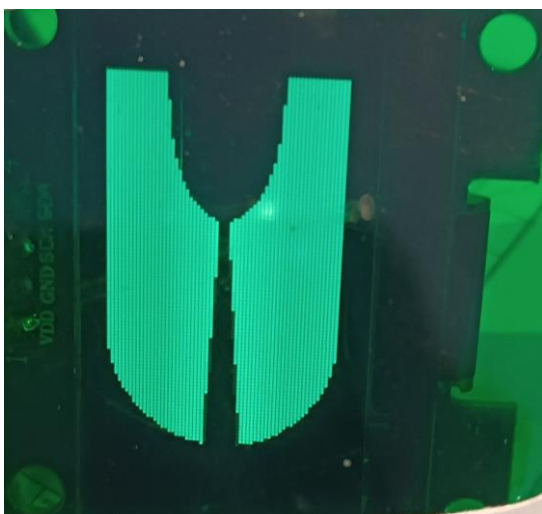


Bei der EM85 ist es der Anschluss st:



Die hohen Spannungen werden durch den Spannungsteiler 1M / 50k auf ca. 0...5V gebracht. Dadurch ist der Arduino nicht überlastet. Wer ganz sicher gehen will, schließt parallel zum Eingang A0 noch eine Zenerdiode(6,3V) an. Ein 10n kann Spannungsspitzen dämpfen. Aber bei mir funktionierten bisher auch so alle Schaltungen ohne Probleme.

Und so sieht das bei der **EM85** aus:



Der Fächer öffnet sich bei geringen Feldstärken. Er wird mit Grafikbefehlen für Rechteck, Ellipse, Kreis und Dreieck programmiert. Der Sketch wurde von dem Funkamateurl DH1DA erstellt. Die Rechte sind bei ihm. Man kann hier den Code kopieren und eine Simulation genießen.

Auf einem 1,3“-Oled wird eine grüne Folie geklebt. Die gibt es z.B. hier:

<https://plotterfolie.de/de/oracal-8300-transparent-cal-transparente-vinylfolie.html>

Die Folie Blau/Grün 097 ist besonders schön. Mit Versand kostet sie < 5 EUR. Und sie kann für viele „Augen“ verwendet werden, denn sie ist im DIN A4-Format.

Ersatz für die EM84

Hier werden Balken angezeigt. Von oben und unten kommen zwei aufeinander zu. Treffen die sich in der Mitte, dann ist der Sender richtig abgestimmt. Dabei kann es sein, dass eine Lücke bleibt. Die zeigt eine geringere Feldstärke an, als für den störungsfreien Empfang notwendig wäre.

Das Programm(der Sketch) ist vergleichsweise einfach. Zuerst werden Rechtecke für einen Rahmen gezeichnet. Die Bahn für die Balkenhälften wird ohne Signal schwarz(dunkel) dargestellt. Zwei helle Balken wachsen aufeinander zu, wenn das Signal zunimmt. Bei einem optimalen Empfang treffen die sich in der Mitte. Die Schaltung und die Ansteuerung sind wie zuvor.

Sketch

```

/* OLED Magisches Auge als Ersatz für eine EM84
berndg42 - Oktober 2024
Hardware:
  Arduino nano V3 clone
  1,3" OLED mit SH1106 Chip, I2C-Interface: A4 - SDA, A5 - SCK
  Analogeingang an A0
Software
  Arduino IDE V 2.2.3
  Grafiktreiber: https://github.com/olikraus/U8g2\_Arduino
*/

#include <Arduino.h>
#include <U8g2lib.h>
#include <Wire.h>

U8G2_SH1106_128X64_NONAME_F_HW_I2C u8g2(U8G2_R0, /* reset=*/
U8X8_PIN_NONE);          // 1.3" Display Test

int wert = 0;
int wertalt = 0;
int diff = 0;

void setup(void) {
  u8g2.begin();
  anzeige();
}

void loop(void) {
  wert = 5*wert + analogRead(A0)/7;
  wert = wert / 6;
  if(wert < 1) {
    wert = 1;
  }
  if (wert > 100) {
    wert = 100;
  }
  diff = abs(wert - wertalt);          // nur anzeigen
  wenn sich etwas ändert
  if (diff >1) {
    wertalt = wert;
    anzeige();
  }
}

void anzeige() {
  //zunächst ein helles Rechteck für den Rahmen zeichnen
  u8g2.clearBuffer();
}

```

```

u8g2.setDrawColor(1); // hell

u8g2.drawBox(0,20, 118,30); //helles Rechteck

u8g2.setDrawColor(0); //dunkel
u8g2.drawBox(2,22, 114,26);

u8g2.setDrawColor(1); //hell
u8g2.drawBox(4,24, 110,22);

//jetzt den aktiven Balken zeichnen

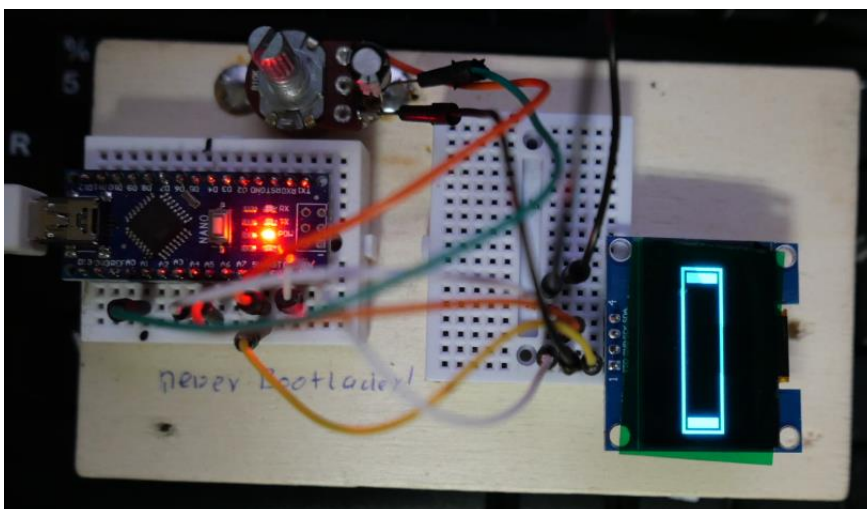
int L = 100 - wert;//Länge des hellen Balkens
L = L/2;
L = L + 4;
//Beginn Mitte für die obere Hälfte
u8g2.setDrawColor(0); //dunkel
u8g2.drawBox(54,24, L,22);

int x; //Beginn untere Hälfte
x = 64 - L;
  u8g2.drawBox(x,24, L,22);
  u8g2.sendBuffer();
}

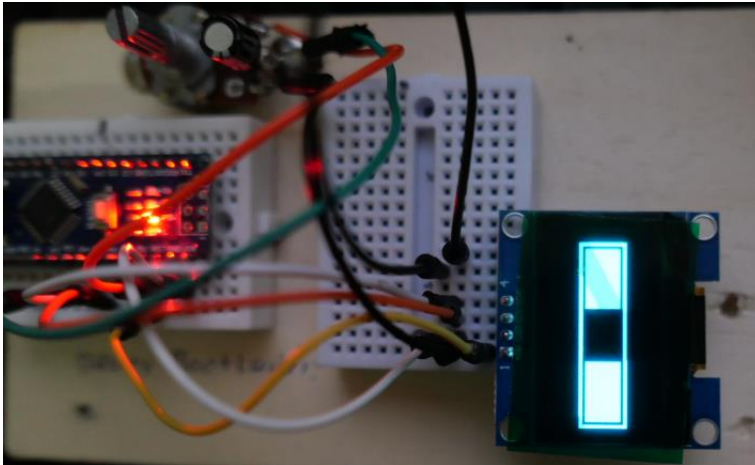
```

Diese Software ist Freeware! Sie darf allgemein für private Zwecke verwendet werden, nicht für kommerzielle.

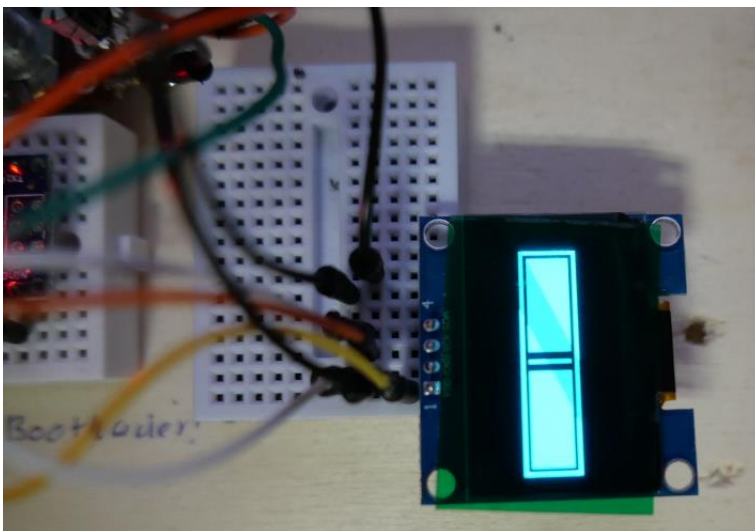
Signal schwach, kein Sender:



Signal mittelmäßig:



Signal optimal:



Der schräge Schatten ist ein Artefakt(Kamera) und real nicht zu sehen!

Fazit

Ich hoffe mit diesem Bericht eine Anregung zum Selbstbau zu geben. Viel Spaß beim Basteln!

DF8ZR; im Oktober 2024