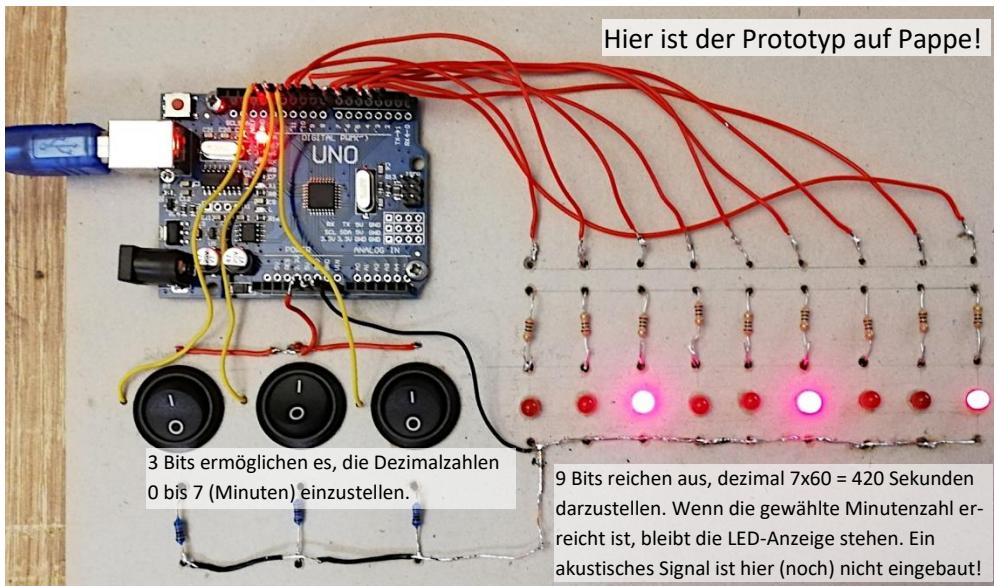


# Binäre Eieruhr – 7 Minuten sollten reichen!

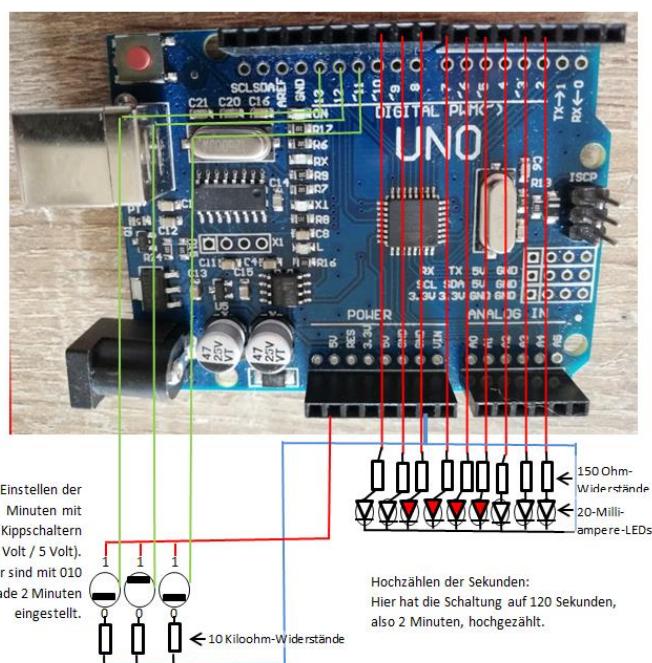


Wir denken dezimal, mit den Zahlen aus dem Zehnersystem, benutzen dabei die Ziffern 0 bis 9. Computer arbeiten binär, mit den Zahlen aus dem Zweiersystem, in dem es nur die Ziffern 0 und 1 gibt. Die hier vorgestellte Eieruhr funktioniert zum binären Abmessen von bis zu sieben Minuten. Sie zeigt im Sekundentakt an, wie die Zeit abläuft.

Dazu wird das Hochzählen der Sekunden mit LEDs im binären System dargestellt. Das Einprogrammieren der gewünschten Minutenzahl nehmen wir binär über drei Schalter vor. (Nach 7 Minuten sollte ein anständiges Ei hart sein.)

Ehrlich gesagt: Diese Schaltung ist vor allem gedacht, um das Denken in binären Zahlen und das Umrechnen zwischen Binär und Dezimal zu üben – und natürlich das Bauen und Programmieren einer komplizierteren Arduino-Schaltung. Ob sie jemals beim Eierkochen zum Einsatz kommt, das ist für mich als Autor eine spannende Frage. An den Abbildungen solltest du erkennen können, wie die Bauteile verbunden werden müssen. Um den Sketch (das Programm) „BinaereEieruhr-3Bit7Minuten-9Bit420Sekunden.ino“ verstehen zu können, ist Übung mit der Arduino-Programmierung erforderlich. Als Einsteiger-Projekt ist es eher nicht geeignet.

Und du kannst dir vor, während oder nach dem Zusammenbau und der Programmierung Gedanken machen, wie Dezimalsystem und hier Binärsystem zusammenhängen. Dich dazu anzuregen, das sollte ja auch der Hauptzweck sein. Im Anhang findest du den Scan der Pappe, auf der die Schaltung oben aufgebaut worden ist.



BinaereEieruhr3Bit7Minuten9Bit420Sekunden.ino

```
int sekunde=0;int sekundenuebertrag1=0;int sekundenuebertrag2=0;int sekundenuebertrag3=0;
int sekundenuebertrag4=0;int sekundenuebertrag5=0;int sekundenuebertrag6=0;
int sekundenuebertrag7=0;int sekundenuebertrag8=0;
static unsigned long tick = 0;
int minuteEinerStelle = digitalRead(11); int minuteZweierStelle = digitalRead(12);
int minuteViererStelle = digitalRead(13);
int minutenWahl = minuteEinerStelle + 2*minuteZweierStelle + 4*minuteViererStelle;
void setup() {
pinMode(2, OUTPUT);pinMode(3, OUTPUT);pinMode(4, OUTPUT);pinMode(5, OUTPUT);
pinMode(6, OUTPUT);pinMode(7, OUTPUT);pinMode(8, OUTPUT);pinMode(9, OUTPUT);
pinMode(10, OUTPUT);pinMode(11, INPUT);pinMode(12, INPUT);pinMode(13, INPUT);
digitalWrite(11, HIGH);digitalWrite(12, HIGH);digitalWrite(13, HIGH);
Serial.begin (9600); delay(500); Serial.println(minutenWahl);
}
void loop() {
if (millis() - tick >= 1000) (tick = millis());sekunde++;Serial.print(sekunde);}
if ((sekunde-1)/60 == minutenWahl){exit(0);}
if (sekunde %2 ==1) { digitalWrite(2, HIGH);} else { digitalWrite(2, LOW);}
sekundenuebertrag1 = sekunde/2;
if (sekundenuebertrag1 %2 ==1) { digitalWrite(3, HIGH);} else { digitalWrite(3, LOW);}
sekundenuebertrag2 = sekundenuebertrag1/2;
if (sekundenuebertrag2 %2 ==1) { digitalWrite(4, HIGH);} else { digitalWrite(4, LOW);}
sekundenuebertrag3 = sekundenuebertrag2/2;
if (sekundenuebertrag3 %2 ==1) { digitalWrite(5, HIGH);} else { digitalWrite(5, LOW);}
sekundenuebertrag4 = sekundenuebertrag3/2;
if (sekundenuebertrag4 %2 ==1) { digitalWrite(6, HIGH);} else { digitalWrite(6, LOW);}
sekundenuebertrag5 = sekundenuebertrag4/2;
if (sekundenuebertrag5 %2 ==1) { digitalWrite(7, HIGH);} else { digitalWrite(7, LOW);}
sekundenuebertrag6 = sekundenuebertrag5/2;
if (sekundenuebertrag6 %2 ==1) { digitalWrite(8, HIGH);} else { digitalWrite(8, LOW);}
sekundenuebertrag7 = sekundenuebertrag6/2;
if (sekundenuebertrag7 %2 ==1) { digitalWrite(9, HIGH);} else { digitalWrite(9, LOW);}
sekundenuebertrag8 = sekundenuebertrag7/2;
if (sekundenuebertrag8 %2 ==1) { digitalWrite(10, HIGH);} else { digitalWrite(10, LOW);}
}
```

