



Kinderferienprogramm 2023

1. Einführung

Heute beschäftigen wir uns mit einem kleinen Mikroprozessor, dem D1 Mini, der schon seit Jahren in Elektrokleingeräten eingebaut wird (einige Funksteckdosen usw.). Er vereint einen Mikrocontroller und eine Funkeinheit in einem Bauteil.

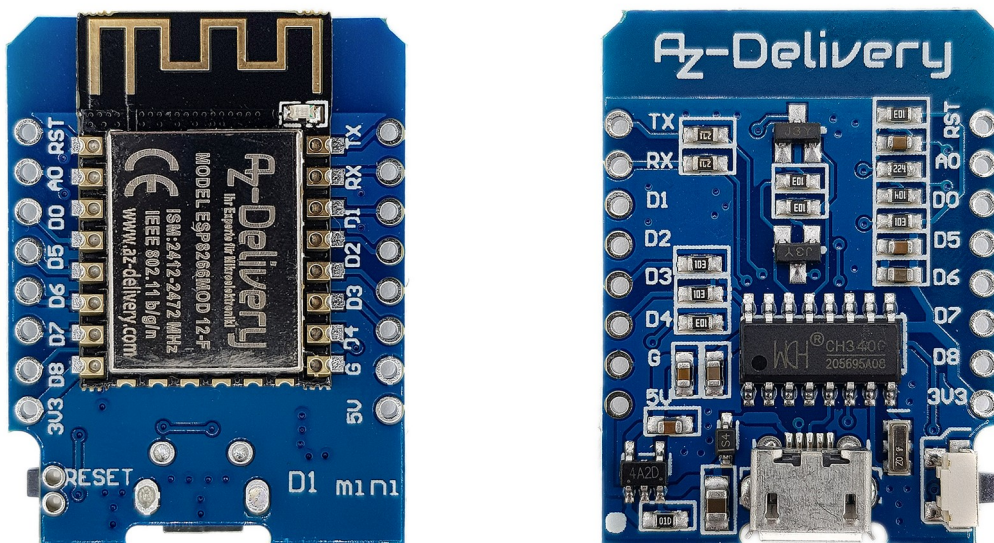


Abbildung 1: Vorder- und Rückseite des D1 Mini

Außerdem besitzt der D1 Mini einen USB-Stecker der das Bauteil mit Strom versorgt und es lässt sich über diese Schnittstelle auch problemlos programmieren.

2. Aufbau

Wie in Abb. 1 zu erkennen, besitzt der D1 Mini mehrere Anschlüsse, einen Reset-Schalter und einen USB-Anschluss.

Die Anschlüsse haben folgende Funktionen:

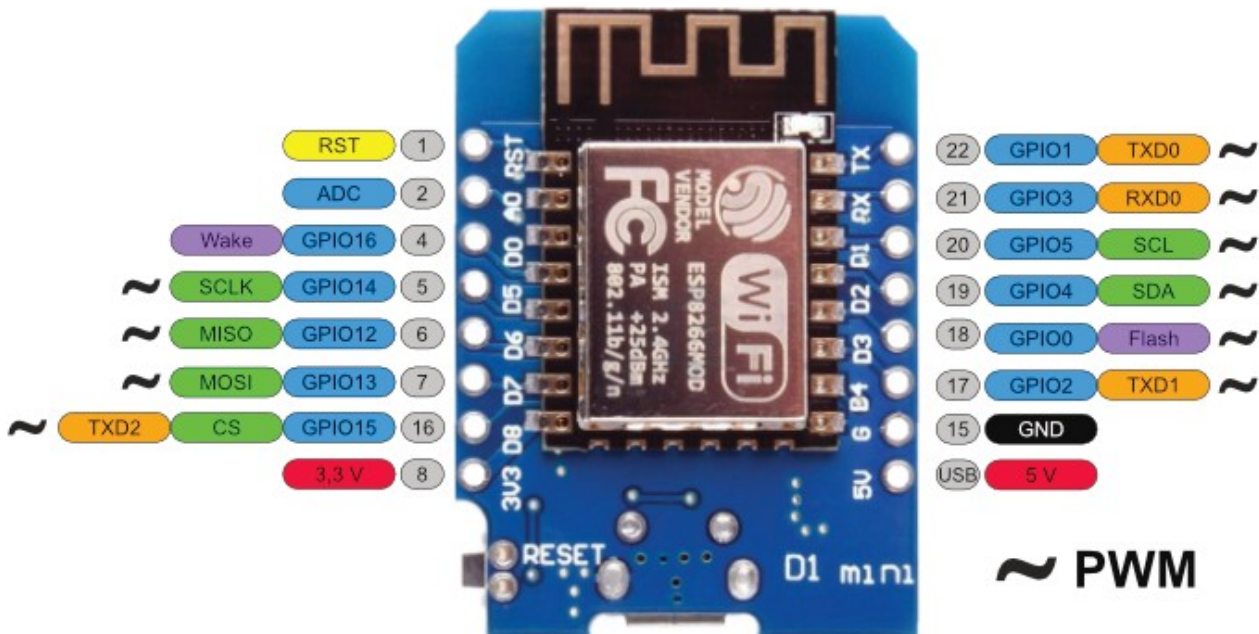


Abbildung 2: Anschlußbelegung des D1 Mini

Erklärung:

- 5V
- 3,3V
- G, GND -Ground oder auch Masse
- GPIO – universelle Ein- bzw. Ausgänge
- RST – Reset, Programm neu starten
- ADC – analoger Eingang
- TX, RX – serielle Schnittstellen
- SCL, SDA, - I2C-Bus
- Flash – Programmierleitung
- Wake -
- PWM – Pulsweitenmodulation
- SCLK, MISO, MOSI, CS – SPI-Bus

Das Ampelmodul besteht aus 3 LED und 330 Ohm Vorwiderständen und einem 4-poligen Stecker auf einer Platine.



3. Grundlagen

Bevor wir ein Programm schreiben müssen wir noch ein paar Dinge kennen lernen:

- Bit, ist die kleinste Informationseinheit z.B. An/Aus, 0/1, true/false usw.
- Byte, 8Bits zusammen ergeben ein Byte, -128 – 127, 0 – 255
- Integer
- Long Integer
- Float
- Double

4. Installation der Arduino Entwicklungsumgebung

5. Was ist ein Programm?

Alle Deklarationen (anlegen von Variablen und Konstanten) und Anweisungen für den Mikroprozessor bilden das Programm, hier verwenden wir die Programmiersprache C.

- `setup ()` - ist die Vorbereitung und läuft als erstes und nur einmal.
- `loop ()` - ist eine Dauerschleife, die anschließend läuft, bis zum Schlafenlegen des Prozessors oder der Strom abgeschaltet wird.

6. Einfache Ampel

```
// GPIO Pins festlegen
const int ledPin_gr = 14;
const int ledPin_ge = 12;
const int ledPin_rt = 13;
const int inPin1  = 2;
const int inPin2  = 16;

void setup ()
{
  pinMode ( inPin1, INPUT );
  pinMode ( inPin2, INPUT );
  pinMode ( ledPin_gr, OUTPUT );
  pinMode ( ledPin_ge, OUTPUT );
  pinMode ( ledPin_rt, OUTPUT );
}

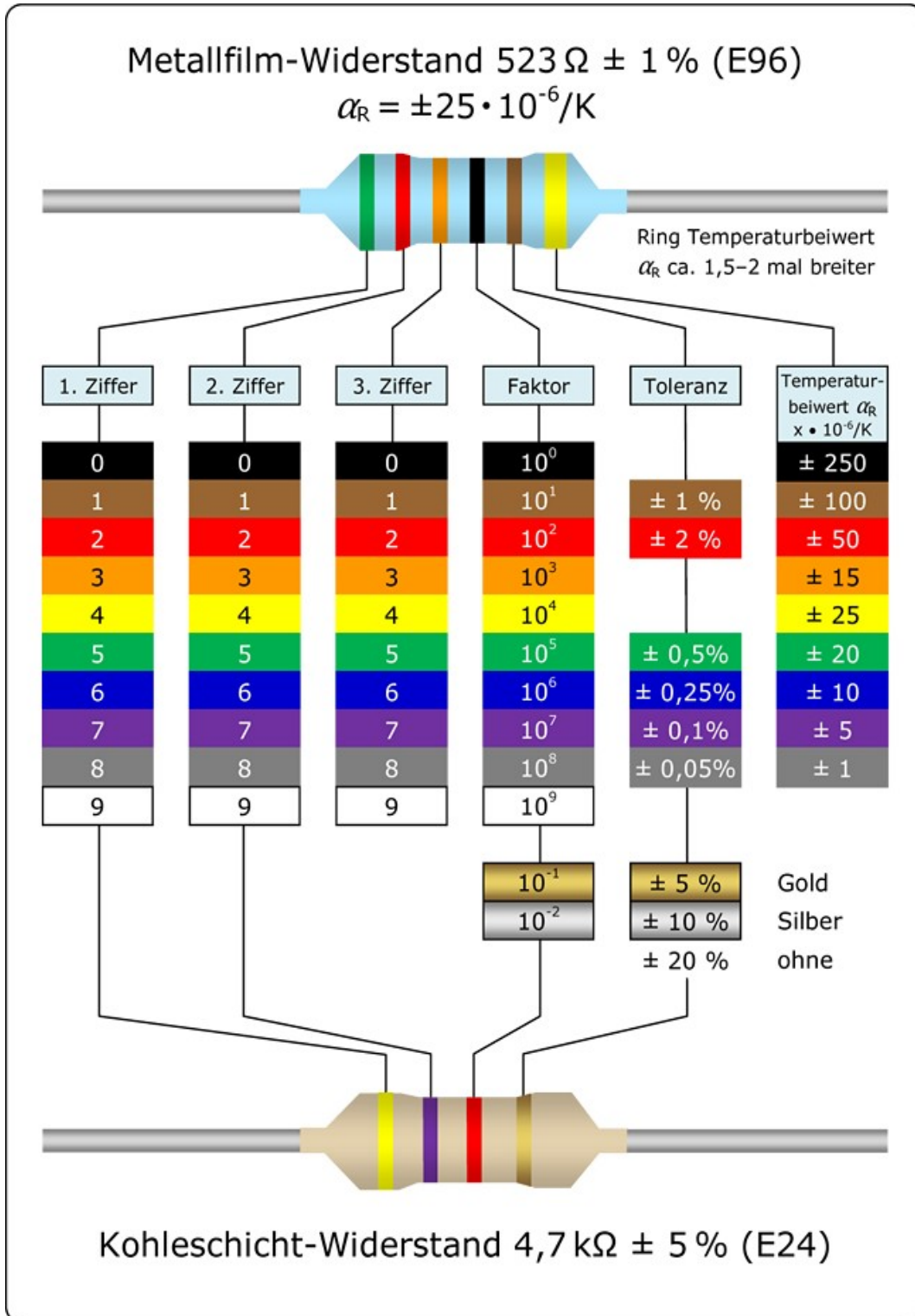
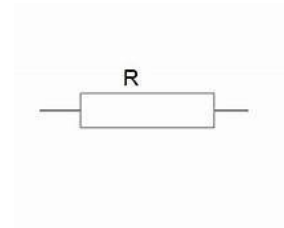
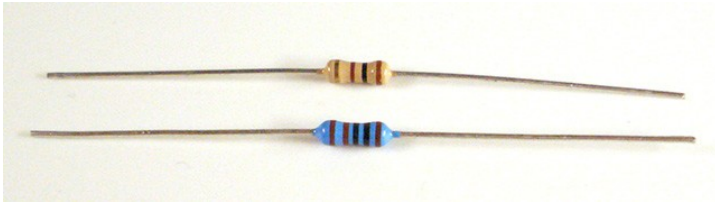
void loop ()
{
  if ( !digitalRead( inPin1 ) )
  {
    delay ( 1000 );
  }
}
```

```
digitalWrite ( ledPin_rt, HIGH );  
digitalWrite ( ledPin_ge, HIGH );  
digitalWrite ( ledPin_gr, LOW );  
delay ( 1000 );  
digitalWrite ( ledPin_rt, LOW );  
digitalWrite ( ledPin_ge, LOW );  
digitalWrite ( ledPin_gr, HIGH );  
delay ( 5000 );  
digitalWrite ( ledPin_rt, LOW );  
digitalWrite ( ledPin_ge, HIGH );  
digitalWrite ( ledPin_gr, LOW );  
delay ( 1000 );  
}
```

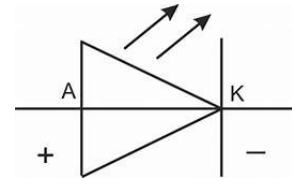
```
digitalWrite ( ledPin_rt, HIGH );  
digitalWrite ( ledPin_ge, LOW );  
digitalWrite ( ledPin_gr, LOW );  
  
}
```

7. zusätzliche Bauteile erklärt

- Widerstände



- Leuchtdioden



8. Ein wenig rechnen (ich weis Mathe ist doof)

Das Ohmsche-Gesetz, beschreibt den Zusammenhang zwischen Strom, Spannung und Widerstand:

Hier die Formel in 3 Varianten:

$$U = R * I$$

$$R = U / I$$

$$I = U / R$$

U - Spannung (V)

I - Strom (A)

R - Widerstand (Ω)

9. Anhang

komische Buchstaben vor Ampere, Volt und Ohm

Wie Euch bereits aufgefallen stehen manchmal so Buchstaben vor unseren Einheiten, was bedeuten die? Die Antwort: wir sind zu faul zum zählen. 1000000 Ω , da habe ich keine Lust die Nullen zu zählen dafür schreibt man dann 1 M Ω .

Hier die Liste:

Zahl	Mathem.	Nullen	Kürzel	Name
1.000.000.000.000.000.000	10^{18}	18	E	Exa
1.000.000.000.000.000	10^{15}	15	P	Peta
1.000.000.000.000	10^{12}	12	T	Tera
1.000.000.000	10^9	9	G	Giga
1.000.000	10^6	6	M	Mega
1.000	10^3	3	k	kilo
1	10^0	0		
0,001	10^{-3}	-2	m	milli
0,000001	10^{-6}	-5	μ	mikro
0,000000001	10^{-9}	-8	n	nano
0,0000000000001	10^{-12}	-11	p	piko
0,0000000000000001	10^{-15}	-14	f	femto
0,000000000000000001	10^{-18}	-17	a	atto

10. Weitere Informationen zum nachbauen

Die Teile wurden größtenteils von AZ-Delivery bezogen

Bei Rückfragen stehe ich unter folgender Adresse zur Verfügung

Computrverein Rotenburg e.V.

Gerd Helbig

Vor dem Glumm 7

27356 Rotenburg

Tel.: 04261 83248

eMail: KFP@CVR.de

Viel Spaß beim Basteln wünschen Euch

Gerd Helbig