

4.1. Aufgaben zu Ampelschaltungen

Aufgabe 1

Baue die Schaltung auf, teste sie und speichere den Sketch unter **Ampel1.ino**. Drehe ein kurzes Video von der blinkenden LED.

Achtung:

- **Ohne Vorwiderstand brennt die Platine durch!**
- **Eine falsch gepolte LED leuchtet nicht! Erinnerung: Langes Bein an + und kurzes Bein an –!**

Aufgabe 2

- a) Bestimme die **Spannungen** U_L an der LED und U_R an dem Widerstand. Verwende dazu ein **Multimeter**. Überlege vorher, in welchem Bereich die beiden Spannungen liegen müssen und stelle den Messbereich entsprechend ein. Überlege nun, wie das Multimeter geschaltet werden muss, um die beiden Spannungen zu messen und zeichne es in die Schaltskizze aus 4.1.1.4. ein.
- b) Bestimme die **Stromstärke** I durch die LED. Überlege wieder, in welchem Bereich die Stromstärke liegen muss und stelle den Messbereich entsprechend ein! Wie muss das Multimeter diesmal geschaltet werden? Zeichne es ein drittes Mal in die Schaltskizze aus 4.1.1.4. ein.
- c) Bestimme nun zur Kontrolle direkt den **Widerstand** der LED. Das Multimeter hat für diesen Zweck eine eigene kleine Batterie, mit deren Hilfe es einen kleinen Messstromkreis aufbauen kann, wenn man den Messbereich auf Ω stellt und die LED direkt in die Buchsen des Multimeters steckt.
- d) Berechne den Widerstand der LED aus den Messungen von a) sowie b) und vergleiche mit c).
- e) **Analog-Multimeter** können eigentlich nur **Ströme** messen, deren magnetische Wirkung dann den Zeiger in Bewegung setzt. Wie müssen solche Ampèremeter geschaltet werden, um auch **Spannungen** erfassen zu können?
- f) **Digital-Multimeter** können eigentlich nur **Spannungen** messen. Die Spannung an den Messelektroden wird mit einer stufenweise ansteigenden Vergleichsspannung kompensiert, bis (nahezu) kein Strom mehr fließt und die Vergleichsspannung den gleichen Wert wie die zu messende Spannung hat. Die Vergleichsspannung bzw. die Zahl der Spannungsstufen wird dann als Messwert digital angezeigt. Wie muss ein solches Voltmeter geschaltet werden, um auch **Ströme** erfassen zu können?

Aufgabe 3

- a) Verändere den Sketch und die Schaltung so, dass die LED jetzt an Pin 10 sitzt.
- b) Erweitere den Sketch so, dass zu Beginn des Sketches die Leuchtdauer in % festgelegt wird. Durch die Zeile `int Leuchtdauer = 27` wird dann festgelegt, dass die LED immer abwechselnd 27 ms an- und 63 ms ausgeschaltet ist.
- c) Teste die Schaltung und speichere den neuen Sketch unter **Ampel2.ino**. Drehe ein kurzes Video von der schnell blinkenden LED.

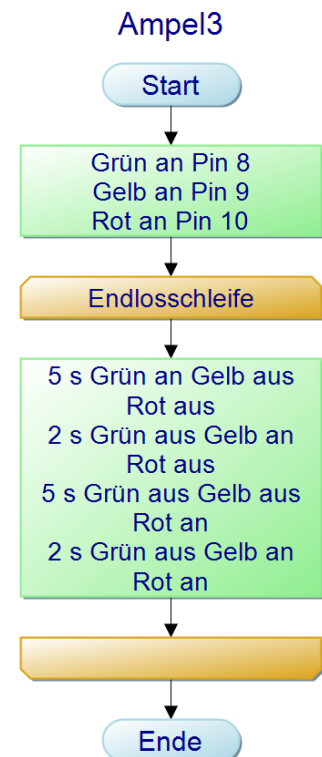
Aufgabe 4

- a) Zum Schluss soll eine Verkehrsampel simuliert werden. Verändere den Sketch und die Schaltung so, dass
 - eine rote LED **mit Vorwiderstand** an Pin 10
 - eine gelbe LED **mit Vorwiderstand** an Pin 9
 - eine grüne LED **mit Vorwiderstand** an Pin 8 sitzt.
- b) Erweitere den Sketch so, dass die Ampel nach dem rechts abgebildeten Ablaufdiagramm geschaltet ist.
- c) Teste die Schaltung und speichere den neuen Sketch unter **Ampel3.ino**. Drehe ein kurzes Video von einem vollständigen Durchlauf.

Aufgabe 5

Die Ampel aus Aufgabe 4 soll mit einer **Fußgängerampel** kombiniert werden. Erstelle zunächst den Ablaufplan in der folgenden Tabelle, erweitere dann die Schaltung und den Sketch aus Aufgabe 4 entsprechend, speichere den Sketch unter **Ampel4.ino** und drehe eine Video über die vollständige Sequenz.

Zustand	Zeit	Fußgänger		Autos		
		grün/6	rot/7	grün/8	gelb/9	rot/10
1	5 s	aus	an	an	aus	aus
2	1 s					
3						
4						
5						
6						



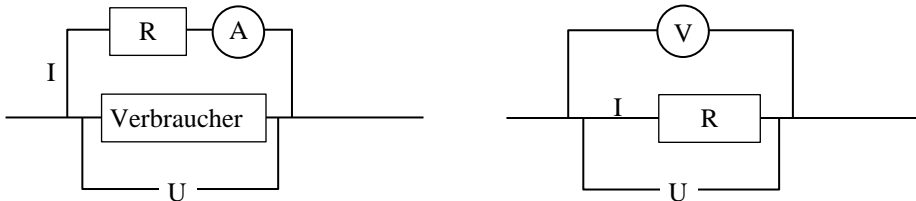
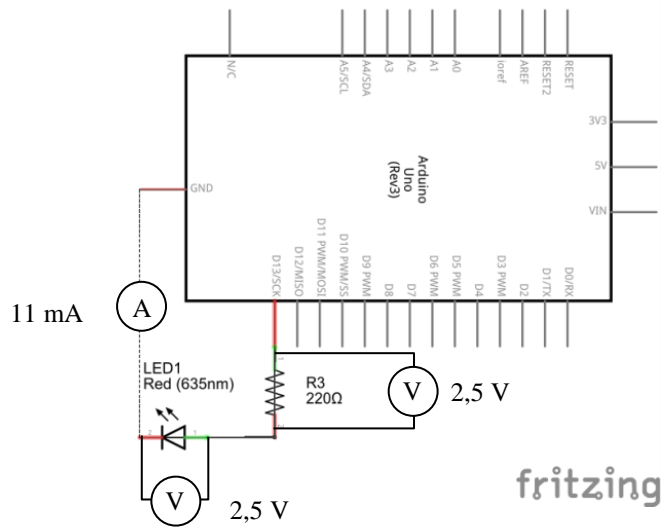
4.1. Lösungen zu den Aufgaben zu Ampelschaltungen

Aufgabe 1

<http://www.poenitz-net.de/Informatik/4.Mikrocontroller/4.1.Ampel1.mp4>

Aufgabe 2

- siehe rechts
- siehe rechts
- siehe rechts
- Die LED hat ebenfalls einen Widerstand von ca. 220 Ω.
- Das Ampèremeter A hat einen sehr geringen Innenwiderstand und wird daher mit einem hohen Vorwiderstand R parallel zum Verbraucher geschaltet. Dann wird nur ein geringer Strom abgezweigt, dessen Stärke $I = \frac{U}{R}$ aber proportional zur abfallenden Spannung U ist. (siehe unten links)
- Das Voltmeter V hat bereits einen sehr hohen Innenwiderstand und wird daher wie bei der normalen Spannungsmessung parallel zum Verbraucher mit dem Widerstand R geschaltet. Die gemessene Spannung $U = R \cdot I$ ist proportional zum Stromstärke I durch den Verbraucher. (siehe unten rechts)



Aufgabe 3

```
int ledPin = 10;           // Variable ledPin initialisieren
int Leuchtdauer = 27;     // Leuchtdauer initialisieren
void setup()
{
  pinMode(ledPin,OUTPUT); // Pin 10 als Ausgang deklarieren
}
void loop()
{
  digitalWrite(ledPin,HIGH); // LED anschalten
  delay(Leuchtdauer);        // 5 Sekunden warten
  digitalWrite(ledPin,LOW);  // LED ausschalten
  delay(100-Leuchtdauer);    // 5 Sekunden warten
}
```

<http://www.poenitz-net.de/Informatik/4.Mikrocontroller/4.1.Ampel2.mp4>

Aufgabe 4

```
int gruen = 8;           // Variable gruen initialisieren
int gelb = 9;           // Variable gelb initialisieren
int rot = 10;           // Variable rot initialisieren

void setup()
{
  pinMode(gruen, OUTPUT); // Pin gruen als Ausgang festlegen
  pinMode(gelb, OUTPUT); // Pin gelb als Ausgang festlegen
  pinMode(rot, OUTPUT);  // Pin rot als Ausgang festlegen
}
```

```

void loop()
{
  digitalWrite(gruen, HIGH); // gruen anschalten
  delay(5000); // 5 Sekunden warten
  digitalWrite(gruen, LOW); // gruen ausschalten
  digitalWrite(gelb, HIGH); // gelb anschalten
  delay (2000); // 2 Sekunden warten
  digitalWrite(gelb, LOW); // gelb ausschalten
  digitalWrite(rot, HIGH); // rot anschalten
  delay (5000); // 5 Sekunden warten
  digitalWrite(gelb, HIGH); // gelb anschalten
  delay(2000); // 2 Sekunden warten
  digitalWrite(gelb, LOW); // gelb ausschalten
  digitalWrite(rot, LOW); // rot ausschalten
}

```

<http://www.poenitz-net.de/Informatik/4.Mikrocontroller/4.1.Ampel3.mp4>

Aufgabe 5

Zustand	Zeit	Fußgänger		Autos		
		grün/6	rot/7	grün/8	gelb/9	rot/10
1	5 s	aus	an	an	aus	aus
2	1 s	aus	an	aus	an	aus
3	1	aus	an	aus	aus	an
4	5	an	aus	aus	aus	an
5	2	aus	an	aus	aus	an
6	1	aus	an	aus	an	aus

```

void setup(){
  pinMode(10,OUTPUT); // Kraftfahrzeug rot
  pinMode(9,OUTPUT); // Kraftfahrzeug gelb
  pinMode(8,OUTPUT); // Kraftfahrzeug grün
  pinMode(7,OUTPUT); // Fußgänger rot
  pinMode(6,OUTPUT); // Fußgänger grün
}

```

```

void loop(){
  digitalWrite(6,LOW); //Zustand 1
  digitalWrite(7,HIGH);
  digitalWrite(8,HIGH);
  digitalWrite(9,LOW);
  digitalWrite(10,LOW);
  delay(5000);
  digitalWrite(8,LOW); //Zustand 2
  digitalWrite(9,HIGH);
  delay(1000);
  digitalWrite(9,LOW); //Zustand 3
  digitalWrite(10,HIGH);
  delay(1000);
  digitalWrite(7,LOW); //Zustand 4
  digitalWrite(6,HIGH);
  delay(5000);
  digitalWrite(6,LOW); //Zustand 5
  digitalWrite(7,HIGH);
  delay(2000);
  digitalWrite(9,HIGH); //Zustand 6
  delay(1000);
}

```

<http://www.poenitz-net.de/Informatik/4.Mikrocontroller/4.1.Ampel4.mp4>