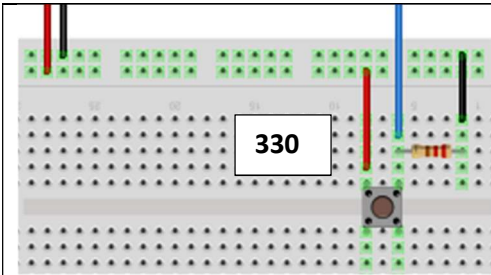
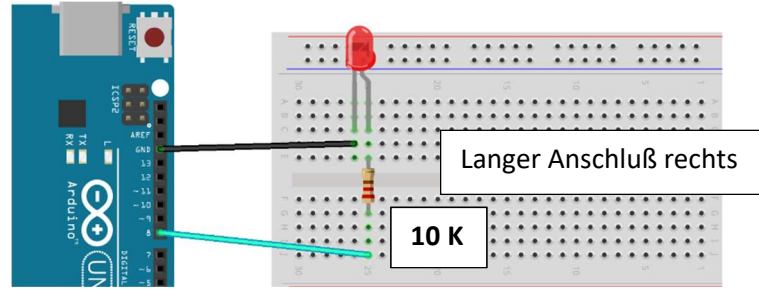


Arduino Kurzreferenz

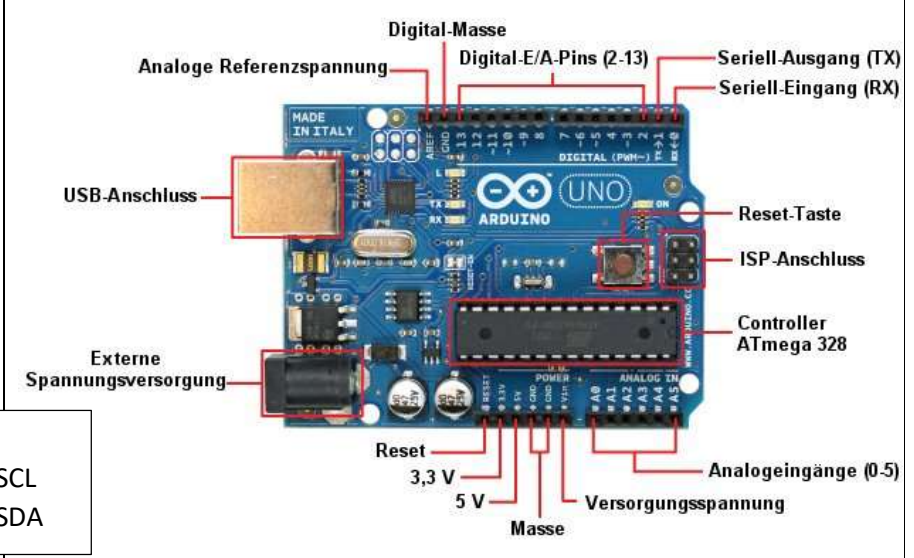
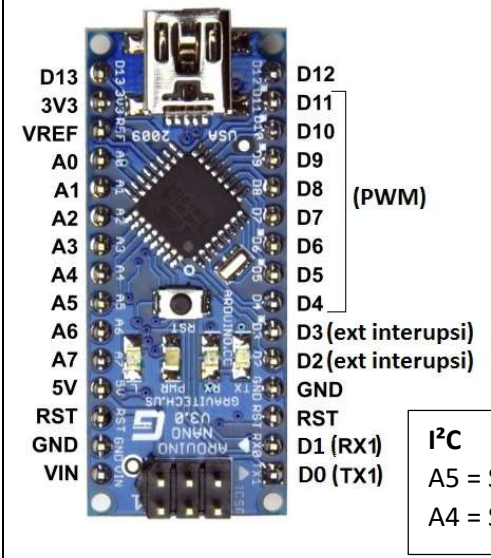
Grundstruktur			
#include <IRremote.h>; include „IRremote“		Einbinden einer Bibliothek (hier für Infrarot)	
<ul style="list-style-type: none"> - mit Anführungszeichen muss die Headerdatei im Sketchordner liegen. Z.B.#include "LedBlinker.h" Farbe= Blaugrün - in Spitzklammern werden diese im Library Ordner gesucht z.B. #include <DHT.h> Farbe =Orange(Standard) sonst schwarz - Bibliothek aus Netz laden und im Ordner User/.../Documents/Arduino/libraries 1.Bibliothek laden 2.entpacken der zip-Datei unter /Users/.../Documents/Arduino/libraries/IRremote 3.Arduino DIE öffnen 4.Sketch/ Bibliothek einbinden/ zu Zip-Bibliothek hinzufügen/ Ordner IRremote auswählen			
int sensorwert; od. int t=50;	Variablen initialisieren / kann auch in Loop angelegt werden		
Const int digitalPin = 8	Variable initialisieren, deren Wert sich nicht ändern kann bzw. darf		
void setup()	Für Grundeinstellungen (z.B. Kanal In- oder Output) wird einmal durchlaufen		
void loop()	Methode wird ständig wiederholt für der eigentliche Programmablauf		
Variablen			
int -32.768 bis 32.767 z.B. int meinWert=10;	long -2 bis 2 Milliarden z.B. long meinWert = 10;		
float Kommazahlen z.B. float meinWert = 2.5;	char Buchstaben z.B. char meinBuchstabe = 'a';		
array Gruppierung von Variablen eines Typs z.B. meinWert[3]={10,12,14}; analogWrite(ledPin , meinWert[0]);			
Befehle			
pinMode(3,OUTPUT); oder INPUT	in setup()	setzt den digitalen Kanal 3 als Ausgang	
digitalWrite(3,HIGH); bzw. (3,LOW) für 0 Volt	Schaltet 5V+ auf den digitalen Kanal 3 (zB. Für LED)		
digitalWrite(3, INPUT_PULLUP) od. digitalWrite(3,HIGH);	Pull-Up Widerstand wurde zugeschaltet		
digitalRead(4);	liefert HIGH oder LOW (z.B. für Schalterabfragen)		
analogWrite(3,255) Achtung: nur an PWN-Kanal (~)	am digitalen Kanal 3 wird werden 5V+ angelegt (0-255)		
analogRead(1);	liefert den anliegenden Wert vom analogen Kanal 1		
delay(1000);	der Programmablauf wird eine Sekunde verzögert		
Serial.begin(9600); für virtuellen Monitor	in setup()	Startet die Datenübertragung mit 9600 Baud	
Serial.println(analogRead(1)); od. Serial.print(„Hallo Welt“);	Sendet den Wert vom Kanal 1 an den seriellen Monitor		
Serial.read()	Liest Daten vom seriellen Monitor (serielle Schnittstelle)		
Serial.flush();	Leert den Empfangsbuffer der seriellen Schnittstelle		
map(wert,0,1023,0,3) z.B. temp=map(sensor,0,410,-50,150);	Teilt den Bereich von 0 bis 1023 auf den Bereich von 0 bis 3		
pulseIn(6,HIGH);	Zeit in	Mikrosek z.B. bis Schall zum Ultraschallsensor zurückkehrt	
tone(8,200);	Passivlautsprecher an Pin 8 mit Frequenz 200 (100-5000)		
tone(8, NOTE_C4, 1000); (Bibliothek pitches.h erforderlich)	Tonausgabe Note C4 für 1000 ms (Ende nicht erforderlich)		
noTone(8);	schaltet Ton aus		
random(6) od. Random(10-20)	Zufallszahl zw. 0 – 5 oder zw. 10-19		
randomSeed(analogRead(5));	in setup()	verbessert Zufallszahl (nicht verbundenen Analogeingang)	
available();	prüft ob Daten an der seriellen Schnittstelle anliegen		
min(x,y); bzw max(x,y);	gibt den kleineren bzw. größeren Wert zurück;		
constrain(wert,10,20);	gibt Wert zwischen 10 und 20 aus (z.B. bei Wert 30 -> 20)		
Methoden			
Definition	Definition mit Parameterübergabe	Definition für Rückgabe	
void blinken(){ digitalWrite(ledPin, HIGH); delay(500); digitalWrite(ledPin, LOW); delay(500); }	void blinken(int thePin, int dauer){ digitalWrite(thePin, HIGH); delay(500); digitalWrite(thePin, LOW); delay(500); }	float quadrat(float x){ float ergebnis = x*x; return ergebnis; }	
Aufruf: blinken();	Aufruf : blinken(3,1000);	Aufruf: wert = quadrat(12.3);	
Abfragen		Schleifen	
if-Abfrage	Switch-case-Abfrage	for-Schleife	do-while-Schleife
if (digitalRead(btnPin)==HIGH) { digitalWrite(ledPin,HIGH); } else { digitalWrite(ledPin,LOW); }	switch (meineVariable) { case 1: befehl1; break; case 2: befehl2; default: befehl3; break;}	for (int i=0; i<10; i++){ // Anweisungen } while-Schleife while (Bedingung) (z.B.i<5) { Anweisung; i++; }	do { delay(50); x = analogRead(3); // prüft den Sensorwert am Pin 3 } while (x < 100); (wird wiederholt bis Abbruchbedingung)



Taster mit Pull-Down-Widerstand (10K Ohm)

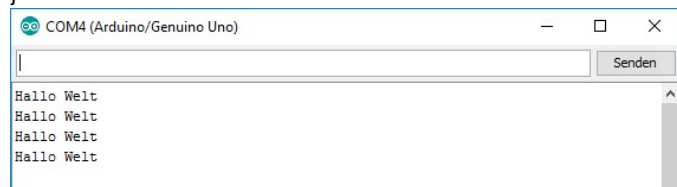


LED mit Vorwiderstand (330 Ohm)



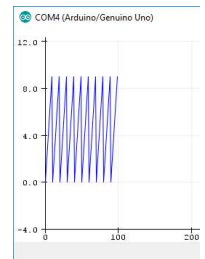
Serieller Monitor: Ausgabe

```
void setup()
{ Serial.begin(9600); } // Konfiguration der Schnittstelle
void loop()
{ Serial.println("Hallo Welt"); // Ausgabe
  delay(1000);
}
```



Serieller Plotter: Ausgabe

```
void setup() {
  Serial.begin(9600);
}
void loop() {
  for (int i = 0; i < 10; i++) {
    Serial.println(i);
  }
  for (int i = 0; i > 10; i--) {
    Serial.println(i);
  }
  delay(1000);
}
```



Serieller Monitor: Daten für Auswertung senden

```
int gesendeterWert=0;
void setup()
{ Serial.begin(9600); } // Konfiguration der Schnittstelle
void loop()
{ if (Serial.available() > 0) // prüft ob Daten anliegen
  {gesendeterWert = Serial.read();} // Daten werden übergeben
  if (gesendeterWert == '1')
  { Serial.println("Wert war 1"); } // Ausgabe, falls Wert = 1
  else
  { Serial.println("Wert war nicht 1"); } // Ausgabe, falls Wert <> 1
  delay(500);
}
```



Serieller Monitor: Daten für Anzeige senden

```
char gesendeterWert; // Variable als char
void setup() {
  Serial.begin(9600);
}
void loop() {
  if (Serial.available() > 0)
  {
    gesendeterWert = Serial.read();
    Serial.print("Eingabe war ");
    Serial.println(gesendeterWert);
  }
  delay(500);
}
```

