1.1 SOS

```
#include "RGB_LED.h"
int kurz=500;
int lang=2000;
int pause1=500;
int pause2=3000;
RGB_LED rgbLed;

void setup() {
}

void loop() {
  for (int i=0;i<3;i++){
     rgbLed.setColor(255, 0, 0);
     delay(kurz);
     rgbLed.turnOff();
     delay(pause1);
  }
  for (int i=0;i<3;i++){
     rgbLed.setColor(255, 0, 0);
     delay(lang);
     rgbLed.turnOff();
     delay(pause1);
  }
  for (int i=0;i<3;i++){
     rgbLed.setColor(255, 0, 0);
     delay(kurz);
     rgbLed.turnOff();
     delay(pause1);
  }
  delay(pause2);
}
```

1.2 Treppenhauslicht (input alle 50ms)

```
#include "RGB_LED.h"
RGB_LED rgbLed;
int zeit;

void setup() {
  Serial.begin(9600);
  pinMode(USER_BUTTON_A,INPUT);
  pinMode(USER_BUTTON_B,INPUT);
  zeit=0;
}

void loop() {
  int inputA=digitalRead(USER_BUTTON_A);
  int inputB=digitalRead(USER_BUTTON_B);

  Serial.printf("A: %d B: %d\n", inputA,inputB);
  if (zeit==5000) rgbLed.turnOff();
  if (inputB==LOW) rgbLed.turnOff();
  if (inputA==LOW)
  {
    rgbLed.setColor(0, 0, 255);
    zeit=0;
   }
  delay(50);
  zeit=zeit+50;
}
```

1.3 LED Steuerung

```
#include "RGB_LED.h"
RGB_LED rgbLed;
char incomingByte = 0;
void setup() {
   Serial.begin(9600);
}

void loop() {
  if (Serial.available() > 0) {
    // read the incoming byte:
    incomingByte = Serial.read();
    switch (incomingByte){
      case 'r':case 'R': rgbLed.setColor(255,0,0);break;
      case 'g':case 'G': rgbLed.setColor(0,255,0);break;
      case 'b':case 'B':rgbLed.setColor(0,0,255);break;
      case 'c':case 'C':rgbLed.turnOff();break;
    }
    //Serial.print("I received: ");
    //Serial.println(incomingByte);
  }
}
```

2.1

```
#include <OledDisplay.h>
String s;
void setup() {
    Serial.begin(9600);
}

void loop() {
 if (Serial.available() > 0) {
    s=Serial.readString();
    Serial.print(s);

    char buf[20];
    s.toCharArray(buf,20);
    Screen.print(buf);
 }
}
```

2.2 Quiz

```
#include <OledDisplay.h>
String s;
int countTrue=0;
int countFalse=0;

void setup() {
  Serial.begin(9600);
  pinMode(USER_BUTTON_A,INPUT);
  pinMode(USER_BUTTON_B,INPUT);
}

void loop() {
  if (Serial.available() > 0) {
    s=Serial.readString();
    //Serial.print(s);
    char buf[2];
    s.toCharArray(buf,2);
```

```
    String sub=s.substring(2);
    char bufAuf[20];
    sub.toCharArray(bufAuf,20);

    Screen.print(bufAuf);

    char c=buf[0];
    //Serial.println(c);

    int inputA;
    int inputB;
    do{
      inputA=digitalRead(USER_BUTTON_A);
      inputB=digitalRead(USER_BUTTON_B);
    //Serial.printf("A: %d B: %d\n", inputA,inputB);
    }while (inputA!=LOW && inputB!=LOW);

    if ((c=='A' && inputA==LOW) || (c=='B' && inputB==LOW)) countTrue++;
    else countFalse++;

    Serial.println(countTrue);
    Serial.println(countFalse);

    String str = String(String(countTrue)+" "+String(countFalse));
    str.toCharArray(buf,20);
    Serial.println(buf);

    Screen.print(2,buf);
    //Screen.clean();
  }
}
```

2.3 Ampel

```
#include <OledDisplay.h>
String phasen[]={"Gruen","Gelb","Rot"};
int ms=1000;
void setup() {
   pinMode(USER_BUTTON_A,INPUT);
   pinMode(USER_BUTTON_B,INPUT);
   while (digitalRead(USER_BUTTON_A)==HIGH && digitalRead(USER_BUTTON_B)==HIGH);
}

void loop() {
      char buf[8];
      for (int i=0;i<3;i++){
        phasen[i].toCharArray(buf,8);
        Screen.print(i,buf,true);
        delay(ms);
        Screen.clean();
        }
      phasen[1].toCharArray(buf,8);
      Screen.print(1,buf);
      phasen[2].toCharArray(buf,8);
      Screen.print(2,buf);
      delay(ms);
      Screen.clean();
}
```

3.1

```
#include "HTS221Sensor.h"
#include "LPS22HBSensor.h"

DevI2C *i2c;
HTS221Sensor *sensor;
LPS22HBSensor *psensor;
float temperature = 0;
float pressure = 0;

void setup() {
    //Baudrate
    //Serial.begin(9600);
    //Serial.begin(115200);

    i2c = new DevI2C(D14, D15);
    //humidity & temperature
    sensor = new HTS221Sensor(*i2c);
    // init the sensor
    sensor -> init(NULL);
    //pressure
    psensor = new LPS22HBSensor(*i2c);
    psensor->init(NULL);
}
void loop() {
    // enable
    sensor -> enable();
    // get temperature
    sensor -> getTemperature(&temperature);
    Serial.print(temperature);
    // disable the sensor
    sensor -> disable();
    sensor -> reset();
    // get pressure
    psensor->getPressure(&pressure);
    Serial.print(",");
    Serial.println(pressure);
    delay(1000);
}
```

3.2

```
#include <OledDisplay.h>
#include "HTS221Sensor.h"

DevI2C i2c(D14, D15);
HTS221Sensor sensor(i2c);
float temperatur=0;
char buf[32];
String einheit="Celsius";

void setup() {
  Screen.init();
  sensor.init(NULL);
  pinMode(USER_BUTTON_A,INPUT);
  pinMode(USER_BUTTON_B,INPUT);
}

void loop() {
  if (digitalRead(USER_BUTTON_A)==LOW)
  {
    einheit="Celsius";
```

```
  }
  if (digitalRead(USER_BUTTON_B)==LOW)
  {
    einheit="Fahrenheit";
  }
  sensor.enable();
  sensor.getTemperature(&temperatur);
  sensor.disable();
  sensor.reset();

  if (einheit.equals("Fahrenheit")) temperatur=temperatur*9/5+32;
  String s=String(temperatur)+" "+einheit;
  s.toCharArray(buf,32);
  Screen.print(buf);
  delay(1000);
}
```