

```

//+++++
//+ Wortuhr testen
//+
//+ FACTORY>
//+ <SPACE
//+ Hochschule Pforzheim
//+
//+ www.hs-pforzheim.de/factory-space
//+++++

//*****INFO*****
// Taster lange drücken zum hoch und runterschalten der LEDs
//Ihr könnt den Code einfach kopieren und diesen in Tinkercad einfügen,
//wenn ihr euch nicht sicher seid wo, fragt mal in der Softwareentwicklung nach :)
//testet eure Verkabelung indem ihr die Taster verwendet :)
//*****

//Kompatibilität zu stdint.h herstellen
#define uint32_t unsigned long int
#define uint8_t unsigned char

//+ Taster ++++++
#define SW1 1
#define SW2 0
#define SW1_get() !digitalRead(SW1)
#define SW2_get() !digitalRead(SW2)

int LEDNo=0;
void LED_increment(void);
void LED_decrement(void);

//+ RGB LEDs WS2812B ++++++
#include <Adafruit_NeoPixel.h>
#define WS28PIN 13
#define WS28LINE 27
Adafruit_NeoPixel pixels(WS28LINE, WS28PIN, NEO_GRB + NEO_KHZ800);
#define WS2812_MAXBRIGHTNESS 255
uint32_t PixelLine[27]={0}; //Datenfeld LEDs
void WS2812_setPixelinLine(uint32_t ui32M[],uint32_t ui8Nr, uint32_t uiR, uint32_t uiG, uint32_t uiB );
void WS2812_write(uint32_t ui32M[],uint32_t ui32Size);
bool DisplayLEDS=1;

// Main Loop
void loop()
{
//+++++
//LEDs prüfen
//+++++

//LEDs mit Taster SW2 stellen
if(SW2_get()==1)
{
LED_increment(); //Minute erhohen
//Time_print(1,0); //Zeit anzeigen
DisplayLEDS=1; //Ausgabeanforderung(Flag) fuer LEDs setzen
}
//ToDo: LEDs mit Taster SW1 stellen
if(SW1_get()==1)
{
LED_decrement(); //Minute erhohen
//Time_print(1,0); //Zeit anzeigen
DisplayLEDS=1; //Ausgabeanforderung(Flag) fuer LEDs setzen
}

//+++++
//+Teil2 Ausgabe der LEDs
//+++++

//Ausgabeanforderung abarbeiten
if(DisplayLEDS==1)
{
//Anforderung aufheben
DisplayLEDS=0;

//Vorbereitung alle LEDs ausschalten
for(int i=0; i<27; i++)
{
WS2812_setPixelinLine(PixelLine,i,0,0,0);
}
}
}

```

```

    }

    WS2812_setPixelinLine(PixelLine,LEDNo,255,30,20);

    WS2812_write(PixelLine,27);
}

}

void LED_increment(void)
{
    LEDNo++;
    if(LEDNo>=27)
    {
        LEDNo=0;
    }
}

void LED_decrement(void)
{
    LEDNo--;
    if(LEDNo<0)
    {
        LEDNo=26;
    }
}

void setup()
{
    pinMode(SW1, INPUT);
    pinMode(SW2, INPUT);
    digitalWrite(SW1, HIGH);
    digitalWrite(SW2, HIGH);
    pixels.begin();
}

void WS2812_setPixelinLine(uint32_t ui32M[],uint32_t ui8Nr, uint32_t uiR, uint32_t uiG, uint32_t uiB )
{
    uint32_t ui32Pixel=0;
    uint32_t uiMax=0;
    // (R+G+B) *255 (R+G+B)
    uiMax=uiR+uiG+uiB;
    if(uiMax>WS2812_MAXBRIGHTNESS)
    {
        uiR=(uiR*255)/uiMax;
        uiG=(uiG*255)/uiMax;
        uiB=(uiB*255)/uiMax;
    }
    if(uiR>WS2812_MAXBRIGHTNESS) uiR=WS2812_MAXBRIGHTNESS;
    if(uiG>WS2812_MAXBRIGHTNESS) uiG=WS2812_MAXBRIGHTNESS;
    if(uiB>WS2812_MAXBRIGHTNESS) uiB=WS2812_MAXBRIGHTNESS;
    //ui32Pixel=uiG*256*256+uiR*256+uiB;
    ui32Pixel=uiR*256*256+uiG*256+uiB;
    ui32M[ui8Nr]=ui32Pixel;
}

void WS2812_write(uint32_t ui32M[],uint32_t ui32Size)
{
    uint32_t ui32Loop=0;
    for(ui32Loop=0; ui32Loop<ui32Size; ui32Loop++)
        pixels.setPixelColor(ui32Loop, ui32M[ui32Loop]);
    pixels.show();
}

```