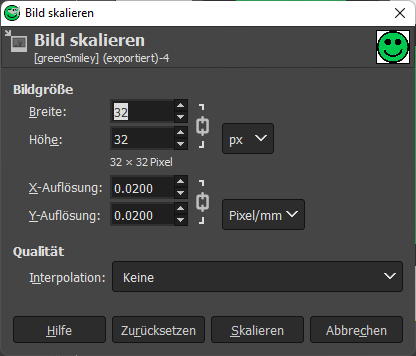
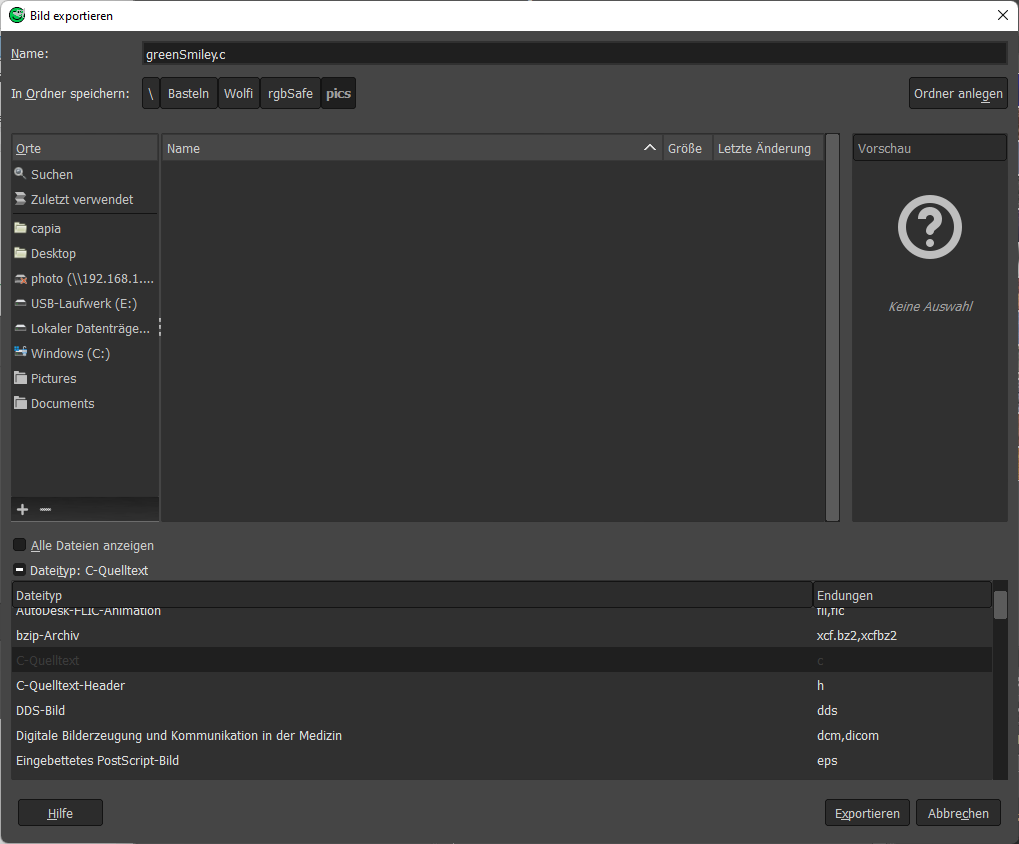
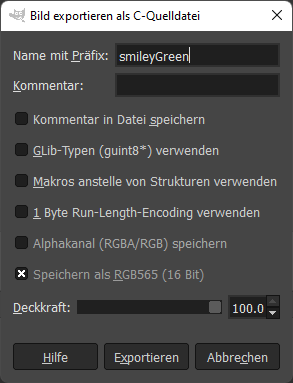
1. Bild in gimp importieren (bmp Bild, alternativ mit paint als bmp abspeichern)
2. Bild skalieren: Bild -> Bild skalieren
3. Einstellungen:
   1. Breite / Höhe 32x32
   2. x/y Auflösung 0.02 Pixel/mm (5 mm Raster, RGB Matrix)
   3. Interpolation: Keine



1. Datei -> Exportieren nach
2. Dateityp: Nach Endung, C-Quelltext



1. Speichern als RGB565 (16 Bit), alles andere deaktivieren



1. C- File anpassen (am Schluss «;» anstelle von «,};» )
2. /\* GIMP RGB C-Source image dump (greenSmiley.c) \*/
3. #ifdef \_\_AVR\_\_
4. #include <avr/io.h>
5. #include <avr/pgmspace.h>
6. #elif defined(ESP8266)
7. #include <pgmspace.h>
8. #elif defined(\_\_IMXRT1052\_\_) || defined(\_\_IMXRT1062\_\_)
9. // PROGMEM is defefind for T4 to place data in specific memory section
10. #undef PROGMEM
11. #define PROGMEM
12. #else
13. #define PROGMEM
14. #endif
15. static const unsigned char greenSmiley[] PROGMEM =

* matrix.drawRGBBitmap(0, 0, (const uint16\_t\*)greenSmiley, 32, 32);
* matrix.show();

<https://learn.adafruit.com/adafruit-matrixportal-m4>

# Passthrough für z.B. hexos

void setup() {

Serial.begin(9600);

Serial1.begin(9600);

}

void loop() {

if (Serial.available()) { // If anything comes in Serial (USB),

Serial1.write(Serial.read()); // read it and send it out Serial1 (pins 0 & 1)

}

if (Serial1.available()) { // If anything comes in Serial1 (pins 0 & 1)

Serial.write(Serial1.read()); // read it and send it out Serial (USB)

}

}