

16 WebREPL

In dem Kapitel **Einführung** ist die Read-Evaluate-Print-Loop (kurz: **REPL**) bereits vorgestellt worden. Mit ihr werden Befehle von der Thonny-Shell an den ESP32 gesendet, dort verarbeitet und die Ergebnisse schließlich an die Shell zurückgesendet. Die Daten wurden dabei über das USB-Kabel übertragen. Die Übertragung kann allerdings auch per WLAN erfolgen. In diesem Fall spricht man dann von **WebREPL**. Wie man dazu vorgeht, wird in den folgenden Abschnitten dargestellt.



Abb. 1

1. Vorbereitung des TTGO

Zunächst verbinden Sie den TTGO wie üblich mit Ihrem PC und öffnen die Thonny-IDE. In der Shell gegen sie ein:

```
>>> import webrepl_setup
```

Folgender Dialog schließt sich an; wählen Sie dabei statt abc123 ein eigenes Passwort

```
WebREPL daemon auto-start status: disabled
```

```
Would you like to (E)nable or (D)isable it running on boot?
```

```
(Empty line to quit)
```

```
> E
```

```
To enable WebREPL, you must set password for it
```

```
New password (4-9 chars): abc123
```

```
Confirm password: abc123
```

```
Changes will be activated after reboot
```

```
Would you like to reboot now? (y/n)
```

Es soll jetzt noch nicht neu gebootet werden; antworten Sie also mit n.

Zuvor muss nämlich die Boot-Datei des ESP32 noch ergänzt werden. Dazu aktivieren Sie zunächst in der Thonny-IDE unter dem Menüpunkt **Ansicht** die Option **Dateien**. Unter der Lasche **Dateien** sehen Sie dann das Fenster **MicroPython Device**. Hier finden Sie die Boot-Datei unter dem Namen `boot.py`. Klicken sie doppelt auf diesen Eintrag; die Datei wird nun in den Editor von Thonny geladen. Nun fügen Sie die folgenden Zeilen in diese Boot-Datei ein:

```
from machine import Pin, SPI
import vga1_8x16 as font
import st7789
import webrepl

def do_connect(ssid, pwd):
    import network
    sta_if = network.WLAN(network.STA_IF)
    if not sta_if.isconnected():
        print('connecting to network...')
        sta_if.active(True)
        sta_if.connect(ssid, pwd)
        while not sta_if.isconnected():
            pass
    display.text(font, 'Network Address:', 30, 60)
    display.text(font, sta_if.ifconfig()[0], 30, 80)

spi = SPI(2, baudrate=20000000, polarity=1, sck=Pin(18), mosi=Pin(19))
display = st7789.ST7789(spi, 135, 240, reset=Pin(23, Pin.OUT), cs=Pin(5,
Pin.OUT), dc=Pin(16, Pin.OUT), backlight=Pin(4, Pin.OUT), rotation=3)

display.init()
display.fill(0)
display.text(font, 'WebRepl-Boot', 30, 20)

# Verbindung mit Ihrem Netzwerk herstellen:
do_connect('Ihre SSID', 'Ihr Passwort')

webrepl.start()
```

Speichern Sie nun die neue boot-Datei auf dem ESP32 ab (*Datei - Save* oder *Ctrl+S*).

Nur für Neugierige: Sie können mit Thonny auch einmal die Datei `webrepl_cfg.py` öffnen. Sie finden dort Ihr WebREPL-Passwort.

Schließen Sie nun die Thonny-IDE. Das USB-Kabel dient jetzt nur noch zur Stromversorgung.

2. WebRepl aktivieren

Führen Sie nun ein Boot-Vorgang durch, z. B. indem Sie den Reset-Taster am Platinenrand betätigen. Auf dem Bildschirm sollte nach kurzer Zeit die Meldung

WebRepl-Boot

Netzwerk-Address:
162.168.xxx.yyy

(mit konkreten Zahlen für xxx und yyy) erscheinen. Diese Netzwerk-Adresse werden Sie gleich benötigen.

3. TTGO mit der WebRepl-Terminal verbinden

Wir öffnen nun die WebRepl-Seite <http://micropython.org/webrepl/>:

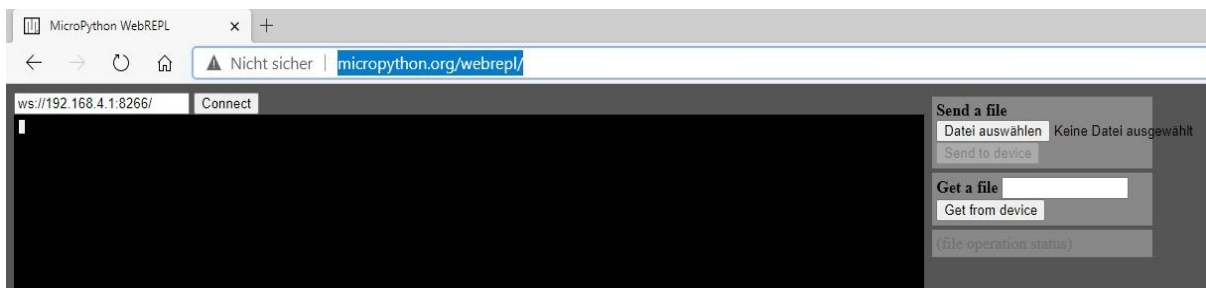


Abb. 2

Hier müssen Sie zuerst die IP in der oberen linken Ecke ersetzen durch die Netzwerk-Adresse, die auf dem Display des TTGO angezeigt wird; die Portnummer hinter dem Doppelpunkt wird nicht verändert.

Dann betätigen Sie die Schaltfläche *Connect*. Es erscheint die Meldung "Welcome to MicroPython!" und Sie werden aufgefordert, Ihr Passwort einzugeben (vgl. Abb. 3). Ihre Eingabe wird in keiner Weise angezeigt. Schließen Sie Ihre Eingabe mit der Enter-Taste ab. Wenn jetzt die Anzeige

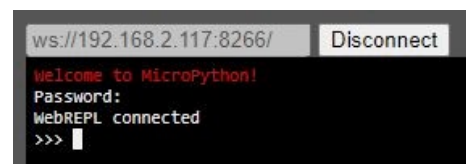


Abb. 3

```
WebREPL connected
>>>
```

erscheint, ist die Verbindung zustanden gekommen.

Geben Sie nun zum Testen hinter dem Python-Prompt `1+1` ein und schließen Sie die Eingabe mit der Enter-Taste ab, dann sollte in der nächsten Zeile das Ergebnis `2` angezeigt werden.

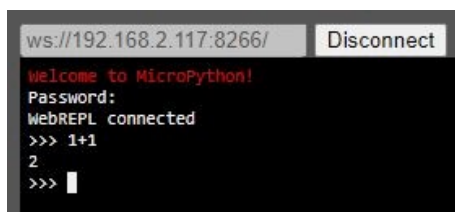


Abb. 4

Sie können nun mit dem Terminal-Fenster dieser Webseite ganz ähnlich verfahren wie mit der Thonny-Shell; allerdings ist dieses Fenster nicht ganz so komfortabel wie die Thonny-Shell.

4. Ergänzungen

Mit Hilfe von WebREPL können Dateien auch vom PC in den ESP32 geladen und umgekehrt auch vom ESP32 auf den PC geholt werden. Dies geschieht über die beiden Formulare `Send a file` und `Get a file` am rechten Rand des Browser-Fensters.

So können Sie z. B. das Programm `animation_0.py` aus dem Programm-Ordner der ESP32-Materialien auf den ESP32 laden. Dieses Programm wird gestartet, wenn man im Terminal-Fenster der WebREPL-Seite `import animation_0` eingibt. Die Animation lässt sich durch Strg-C abbrechen.

Wem die Schrift im Terminal-Fenster zu klein ist, der sollte den Zoom-Faktor des Browsers vergrößern, z. B. auf 150%.