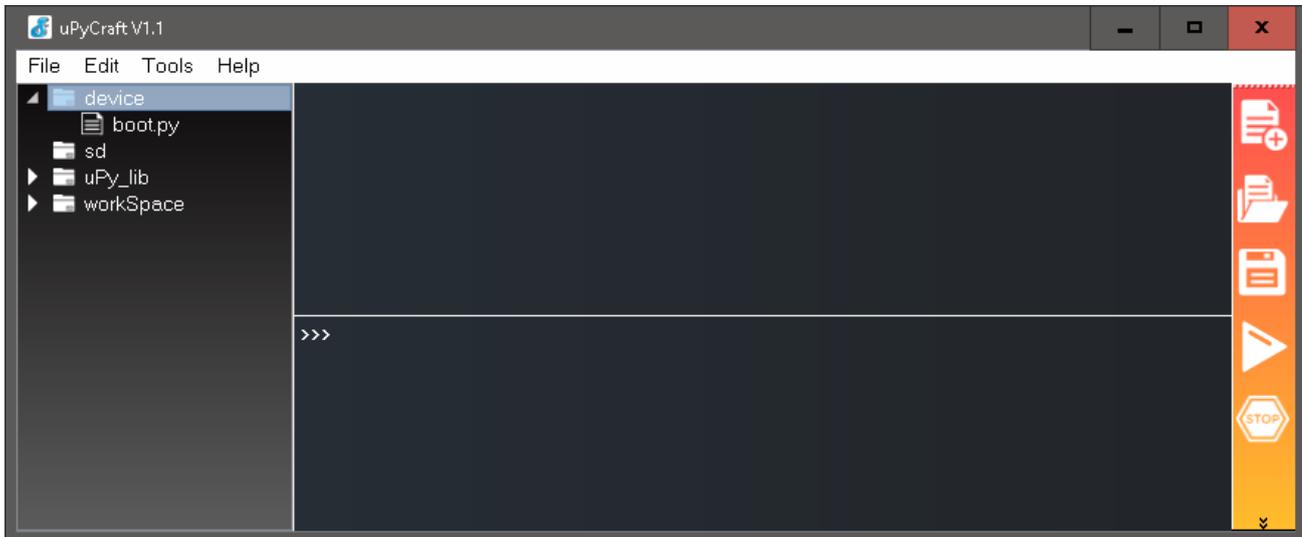


## Webrepl auf dem ESP aktivieren und konfigurieren

Das ESP-Modul wurde mit MicroPython geflasht.

( „MicoPython wurde darauf installiert.“ , „MicroPython-Firmware ist auf dem Modul.“ )

Auf dem Modul ist eine Datei boot.py zu finden.

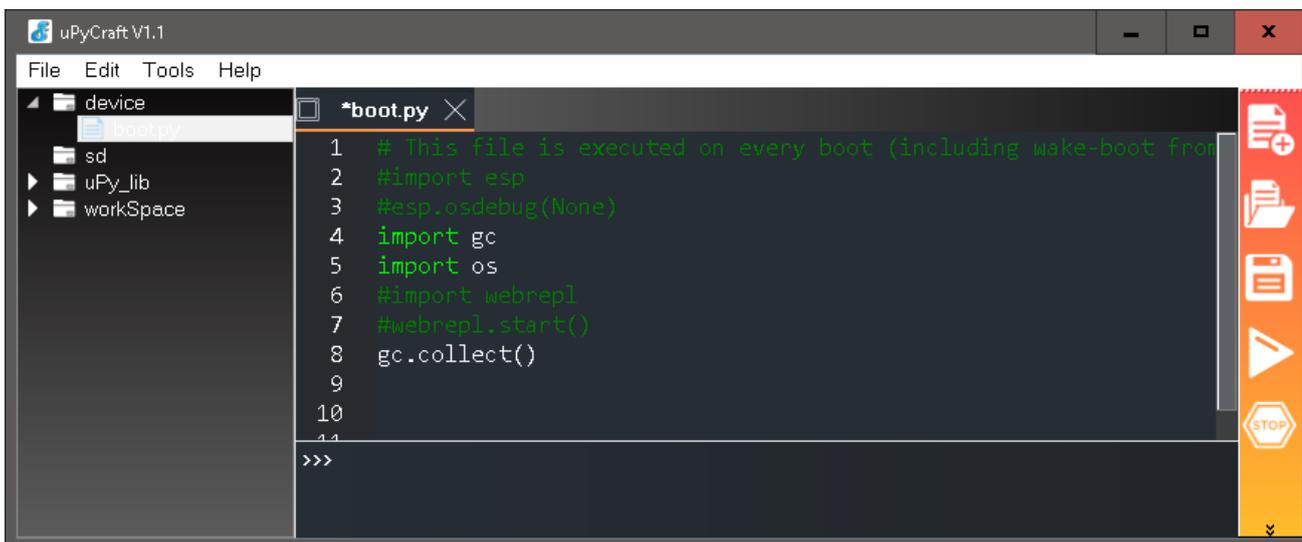


Diese Datei wird bei jedem Boot-Vorgang ausgeführt.

(Verbinden mit dem PC per USB-Kabel, Drücken des RESET-Tasters, ...)

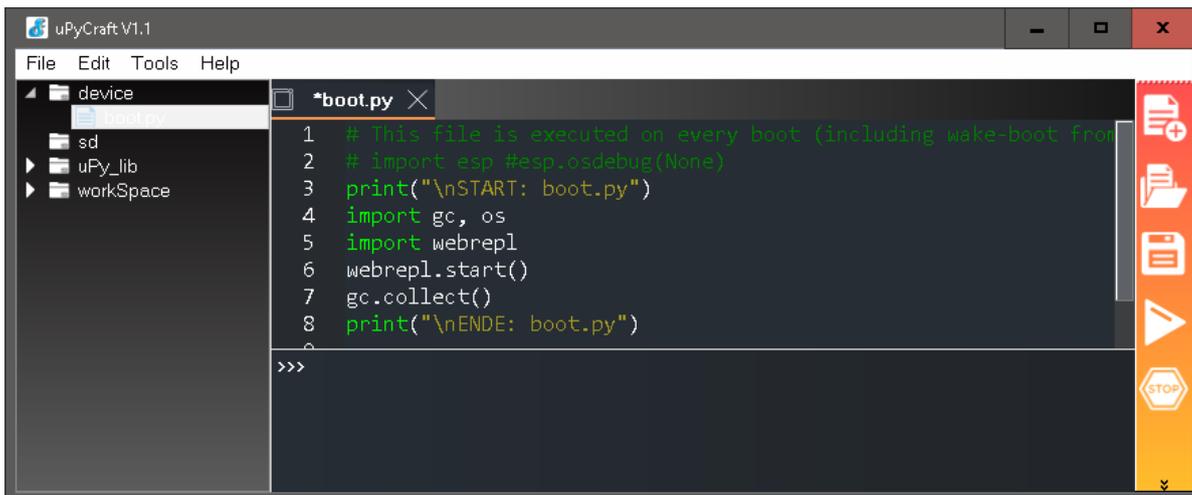
Der Inhalt der Datei wird von uns betrachtet.

`gc.collect` dient zur Reorganisation des Arbeitsspeichers (RAM).



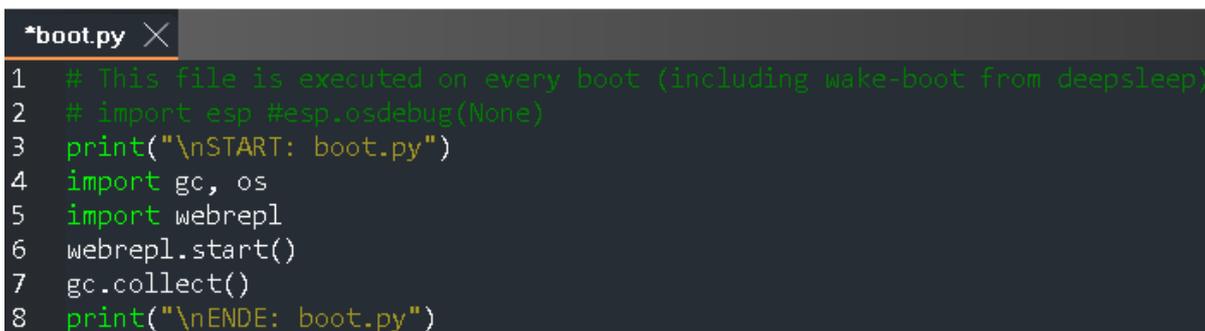
Die folgende kleinen Änderungen :

- informieren uns über die Ausführung der `boot.py`.
- aktivieren WEBRepl auf dem Modul

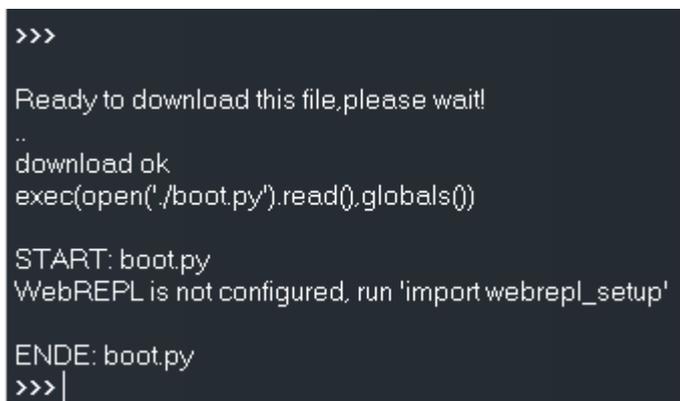


```
uPyCraft V1.1
File Edit Tools Help
device
├── sd
├── uPy_lib
└── workSpace
*boot.py
1 # This file is executed on every boot (including wake-boot from deepsleep)
2 # import esp #esp.osdebug(None)
3 print("\nSTART: boot.py")
4 import gc, os
5 import webrepl
6 webrepl.start()
7 gc.collect()
8 print("\nENDE: boot.py")
>>>
```

<https://github.com/micropython/micropython-esp32/issues/90>



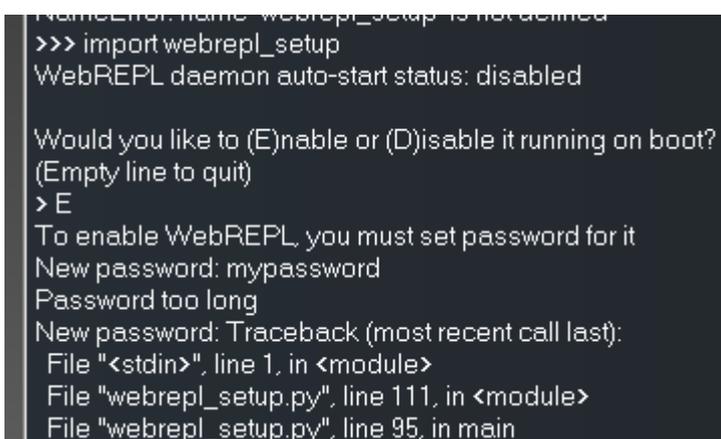
```
*boot.py
1 # This file is executed on every boot (including wake-boot from deepsleep)
2 # import esp #esp.osdebug(None)
3 print("\nSTART: boot.py")
4 import gc, os
5 import webrepl
6 webrepl.start()
7 gc.collect()
8 print("\nENDE: boot.py")
```



```
>>>
Ready to download this file, please wait!
..
download ok
exec(open('./boot.py').read(),globals())

START: boot.py
WebREPL is not configured, run 'import webrepl_setup'

ENDE: boot.py
>>> |
```



```
NameError: name 'webrepl_setup' is not defined
>>> import webrepl_setup
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: mypassword
Password too long
New password: Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
  File "webrepl_setup.py", line 111, in <module>
  File "webrepl_setup.py", line 95, in main
```

```
>>> dir(webrepl_setup)
['main', 'get_daemon_status', 'change_daemon', 'exists', 'getpass', 'sys', 'input_choice', 'RC', 'CONFIG',
'os', 'machine', '__name__', 'copy_stream', 'add_daemon', 'input_pass']
>>> dir(webrepl_setup)
>>>
```

Das Passwort war zu lang,

```
>>> import webrepl_setup
>>> webrepl_setup.main()
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: test4me
Confirm password: test4me
Changes will be activated after reboot
Would you like to reboot now? (y/n) y
bcn 0
del if1
usl
>>>
ets Jan 8 2013,rst cause:2, boot mode:(3,6)

load 0x40100000, len 32108, room 16
tail 12
checksum 0x76
ho 0 tail 12 room 4
load 0x3ffe8000, len 1096, room 12
tail 12
checksum 0x21
ho 0 tail 12 room 4
load 0x3ffe8450, len 3000, room 12
tail 12
checksum 0xa3
csum 0xa3
$| {r'c | bl c$| c ${ $ | d| ssocld c c c |{ | d| s{ocl c c c $c | 'sg| | $|d' {
Traceback (most recent call last):
  File "boot.py", line 1
IndentationError: unexpected indent
OSError: [Errno 2] ENOENT

MicroPython v1.9-6-g821dc27e-dirty on 2017-07-21; ESP module with ESP8266
Type "help()" for more information.
>>>|
```

webrepl\_setup.get\_daemon\_status()

```
>>> webrepl_setup.get_daemon_status()
True
```

```
>>> dir(network)
['__name__', 'WLAN', 'phy_mode', 'STA_IF', 'AP_IF', 'STAT_IDLE', 'STAT_CONNECTING', 'STAT_WRONG_PASSWORD', 'STAT_NO_AP_FOUND', 'STAT_CONNECT_FAI',
'AUTH_OPEN', 'AUTH_WEP', 'AUTH_WPA_PSK', 'AUTH_WPA2_PSK', 'AUTH_WPA_WPA2_PSK']
```

[https://git.card10.badge.events.ccc.de/annejan/micropython/blob/02afc0d241dbd0efb985ba5a7ded9cea616ffcd7/ports/esp8266/modules/webrepl\\_setup.py](https://git.card10.badge.events.ccc.de/annejan/micropython/blob/02afc0d241dbd0efb985ba5a7ded9cea616ffcd7/ports/esp8266/modules/webrepl_setup.py)