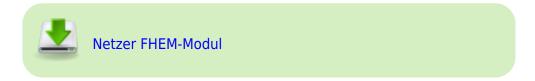
Zugriff auf die Portpins des Netzers (für IO_base Version 1.5) über die Haussteuersoftware FHEM

FHEM ist ein in Perl geschriebener Server für Hausautomatisierung. Er lässt sich auf einigen FritzBox Modellen, Mini-computern, NAS-Systemen etc. betreiben. Zu Testzwecken ist auch ein normaler PC gut geeignet. Alles was benötigt wird, ist eine Perl-Umgebung. Beispielsweise StrawberryPerl, welches auch ohne Installation genutzt werden kann. Das Einrichten von FHEM ist auf der Homepage gut erläutert.



Um den Netzer in Form von Schaltern, Schiebereglern,... in FHEM einzubinden, wird das Modul netzer_fhem.zip benötigt. Das Modul wird derzeit nicht mit FHEM ausgeliefert. Nachdem die Datei entpackt und ins Modulverzeichnis kopiert wurde, ist sie nach einem Serverneustart nutzbar.

Vorbereitung des Netzers

Zuerst müssen über das Webinterface (io) des Netzers die Ports so konfiguriert werden, wie man sie später nutzen möchte. Anschließend wird der gpioserver aktiviert. Über diesen kommunizieren FHEM und der Netzer miteinander. Wenn ein Port mit Ereignistrigger konfiguriert ist, dann werden bei jedem Ereignis alle Portwerte in FHEM aktualisiert.

Konfigurieren von FHEM

Folgende Zeilen werden in die *fhem.cfg* eingetragen und nach dem Speichern über die FHEM Website sofort übernommen. Als erstes wird einer neues Netzermodul definiert:

define <name> Netzer <host:port>

Bereits jetzt sind alle Ports als Eingang verfügbar. Für eine andere Funktion werden sie entsprechend der Einstellung der Netzer Weboberfläche in der *fhem.cfg* eingetragen:

attr <name> Port_a <out|PWM|ADC|cnt|in>

- out Ausgang
- **PWM** PWM Ausgang (die set Funktion wird mit einem Schieberegler versehen)
- ADC A/D Konverter Eingang
- **cnt** digitaler Eingang, es wird ein zusätzliches Reading mit dem Zählerwert angelegt. Dieser lässt ich lesen und schreiben.
- in Eingang (das Attribut kann weggelassen werden)

Jetzt ist die Konfiguration fertig. Für eine bessere Darstellung lassen sich die einzelnen Readings der Ports mit der "readingsProxy" Funktion in FHEM auch einzeln darstellen.

Beispielkonfiguration

```
define netzer01 Netzer 192.168.178.33:65000
attr netzer01 Port a out
attr netzer01 Port d PWM
attr netzer01 Port e ADC
attr netzer01 Port f in
attr netzer01 room Netzer
define netzerla readingsProxy netzer01:Port_a
attr netzerla room Netzer
attr netzerla setFn {($CMD eq "off")?"a 0":"a 1"}
attr netzerla setList on off
attr netzerla valueFn {($VALUE == 0)?"off":"on"}
define netzer1b readingsProxy netzer01:Port_b
attr netzer1b room Netzer
attr netzer1b setFn {($CMD eg "off")?"b 0":"b 1"}
attr netzer1b setList on off
attr netzer1b valueFn {($VALUE == 0)?"off":"on"}
define act netzer1b notify netzer1b {if ("%" eq "off") {fhem ("get
test03");;}}
define netzer1d readingsProxy netzer01:Port d
attr netzerld devStateIcon 0:off 100:on \d:dim06% 1\d.*:dim12% 2\d.*:dim25%
3\d.*:dim37% 4\d.*:dim43% 5\d.*:dim50% 6\d.*:dim62% 7\d.*:dim75%
8\d.*:dim87% 9\d.*:dim93%
attr netzer1d room Netzer
attr netzer1d setFn {($CMD eg "off")?"d 0":($CMD eg "on")?"d 1000":"d " .
$CMD *10}
attr netzerld setList off:noArg on:noArg state:slider,0,1,100
attr netzer1d valueFn {($VALUE == 0)?"off":($VALUE / 10)}
attr netzerld webCmd off:state:on
```

From:

http://www.mobacon.de/dokuwiki/ - MoBaCon

Permanent link:

http://www.mobacon.de/dokuwiki/doku.php?id=de:netzer:fhem

Last update: 2025/06/11 20:42

