

Lektion 6 : Verwendung der Module ti_hub & ti_rover

Übung 2 : Ein- und Ausgabegeräte

In der zweiten Übung der Lektion 6 wird gezeigt, wie man ein Ein- und Ausgabegerät am **TI-Innovator™** unter Zuhilfenahme des Moduls **ti_hub** verwendet.

Objectifs :

- Entdecken des Moduls **ti_hub**.
- Schreiben eines Programmes, das einen « Grove »-Sensor verwendet.

Als Beispiel für einen Sensor soll ein wichtiges Gerät dienen, das **Potentiometer**.

Ein Potentiometer ist ein veränderlicher Widerstand. Es besteht aus einer an beiden Seiten mit Kontakten versehenen kreisförmigen Widerstandsbahn, auf der ein drehbarer Kontakt schleift. Schließt man an die Kontakte der Schleifbahn eine Spannung an, so wird durch den Schleifkontakt eine Teilspannung eingestellt.

Potentiometer werden z.B. für die Lautstärkeregelung bei Verstärkern und Radios verwendet, da sie die variable Einstellung einer Spannung ermöglichen. Man kann sie auch als Winkelgeber verwenden, wenn man den eingestellten Widerstand in eine Drehbewegung umwandelt.

Im Beispielprogramm soll die durch den Schleifkontakt eingestellte Spannung gemessen und angezeigt werden.

Anmerkung: Der Sinn dieser Lektion besteht nicht darin, den Sensor selbst zu studieren, sondern ihn in ein Python-Programm zu integrieren, um die Informationen zu erhalten, die er liefern sollte. Das Programm, das erstellt wird, lässt sich also leicht auf jeden anderen Sensor übertragen.



Das Programm:

- Es wird ein neues Programm mit dem Namen **U6SB2** als **Hub Project** angelegt.
- Danach sieht das Programm wie abgebildet aus.
- Das Modul **ti_system** sollte man etwas allgemeiner halten, indem man **get_key** löscht und durch ***** ersetzt.

```

1.1 1.2 *Dok RAD 1/10
*U6SB2.py
# Hub Project
#-----
from ti_hub import *
from math import *
from time import sleep
from ti_plotlib import text_at,cls
from ti_system import *
#-----

```

- Säubern des Bildschirmes durch **clear_history()** aus dem Modul **ti_system**.
- Anlegen der Funktion **poti()**.
- Einfügen des Potentiometers als externen Sensor an Port **IN1**.

```

1.1 1.2 *Dok RAD 11/13
*U6SB2.py
# Hub Project
#-----
from ti_hub import *
from math import *
from time import sleep
from ti_plotlib import text_at,cls
from ti_system import *
#-----
clear_history()
def poti():
    Var=potentiometer("Port")

```



10 Minuten Coding

TI-NSPIRE CX II-T

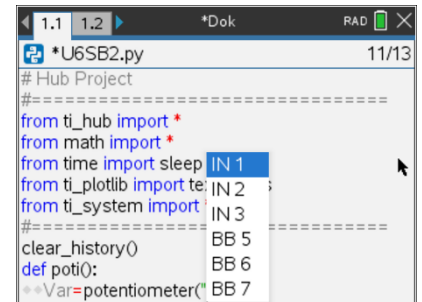
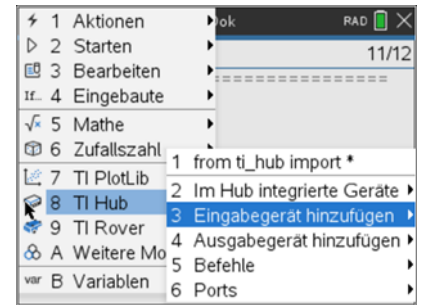
LEKTION 6 : ÜBUNG 2

LEHRERMATERIAL

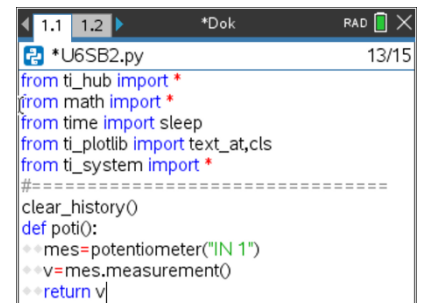
- Die Variable soll **mes** heißen.
- Ein Auswahlménú stellt die möglichen Ports für den Input bereit.
- Es wird **IN 1** gewählt. Die Variable **mes** ist damit mit dem Port **IN 1** verbunden. Alle Aktionen laufen nun über **mes**.

Hinweis: Alle Sensoren (**input devices**) benötigen mindestens zwei Variablen :

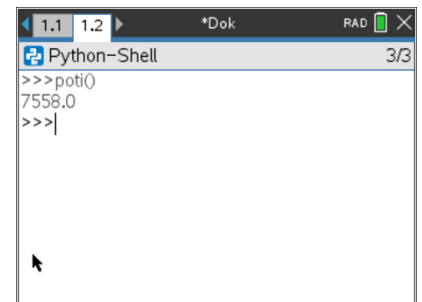
1. `var1 = sensor(« port »)` ordnet `var1` einem Eingang `port` zu
2. `var2 = var1.measurement()` löst die Messung aus und überträgt den Messwert in `var2`



- Die Variable **v** nimmt nun die Messwerte auf, nachdem **mes** zur Durchführung einer Messung aufgefordert wurde..



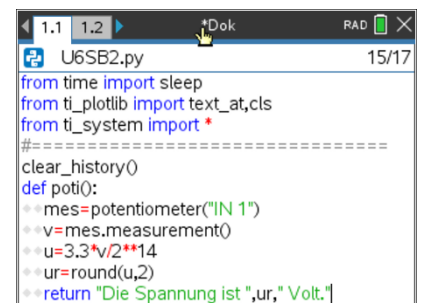
- Nun kann man überprüfen, ob der Sensor funktioniert.
- Dazu muss der TI-Innovator™ mit dem Taschenrechner verbunden werden. Der Sensor wird mit dem Port **IN1** verbunden und die Achse des Potentiometers etwa in eine Mittenstellung gedreht.
- Aufrufen der Funktion **poti()**.
- Man erhält ein Ergebnis ähnlich wie das abgebildete. Dabei handelt es sich nicht um eine Spannung, denn die Betriebsspannung des Potentiometers ist 3,3 V. Es müsste sich eine Spannung im Intervall 0 V bis 3,3 V einstellen.



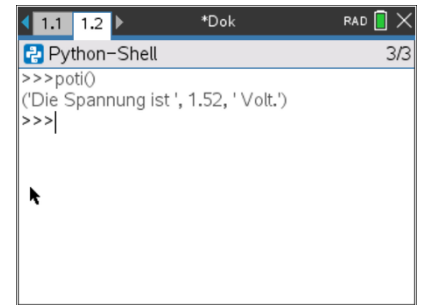
- Man muss noch die Auflösung des Analogwandlers berücksichtigen. Sie beträgt 14 Bit ; also erhält man die Spannung *u* durch

$$u = U_0 \times \frac{v}{2^{14}}$$

- Die Spannung wird auf 2 Stellen gerundet..



- Nun kann man die Spannung direkt ablesen.



```
1.1 1.2 *Dok RAD 3/3
Python-Shell
>>> poti()
('Die Spannung ist ', 1.52, ' Volt.')
>>>|
```

Ein paar weiterführende Ideen :

- Das Potentiometer kann dazu verwendet werden, um einen Drehwinkel anzuzeigen. Der Zusammenhang $\alpha = f(u)$ kann dann auch grafisch veranschaulicht werden.
- Man kann auch einen Punkt auf dem Bildschirm auf und ab wandern lassen, ähnlich wie bei einem Joystick..