

## Elektronický obvod? To není žádná duchařina ;)



Představte si potok, na kterém je mlýnek.

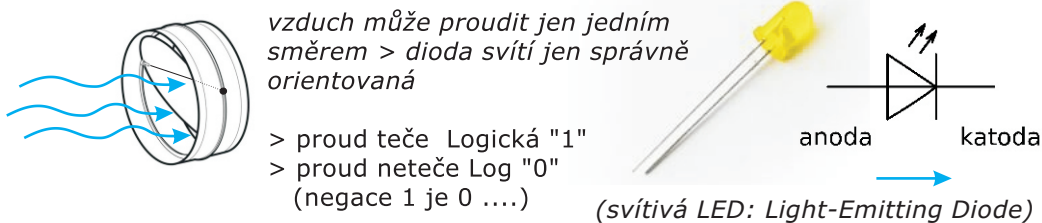
*Mlýnek se točí, když je dostatečný proud vody, která teče korytem potoka (nebo zde: dřevěné vantroky)*

Podobně, když vodičem teče elektrický **proud**, může roztočit motor nebo rozsvítit žárovku.

V obou případech je potřeba nějaký zdroj (tekoucí vodě dodává energii gravitace - voda teče "dolů", elektřina proudí díky rozdílu potenciálů, které nazýváme **napětí**), dále potřebujeme "cestu", po které je energie unášena/přenášena (koryto potoka nebo **vodič**-drát) a potom něco, co energii přijímá/spotřebovává (mlýnek, motor nebo žárovka/ LEDka)

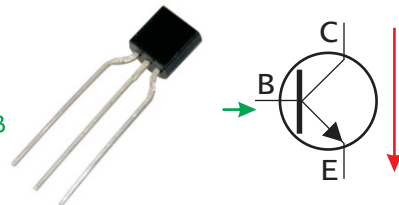
Zkusíme to s využitím analogií > bez vzorečků, bez složitých definic a schémat.

Vodič elektřinu vede, kdežto **polovodič** ji vede jen za určitých podmínek. Například **dioda** - vede proud pouze jedním směrem (podobně jako zpětná klapka u digestoře propustí vzduch jen jedním směrem).



**Tranzistor** - jako elektronický spínač - je podobný jednoduchému mechanickému stroji (páka/kladka), kdy malým proudem je spínán větší proud - a díky malým rozměrům je základním stavebním prvkem integrovaných obvodů.

*působení 3x menší silou?  
lze pomocí páky*



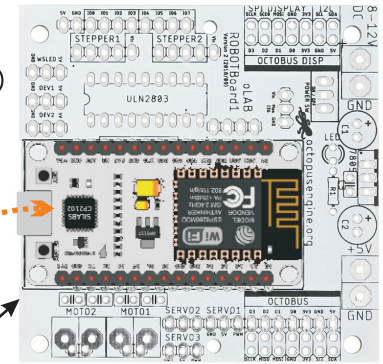
## Práce s ESP32 - a úvod do Micropythonu

inicializace: ESPTOOL > erase, load Micropython  
spojení pomocí sériové linky (HW) a terminálu (SW)  
nastavení wifi a stažení základního FW (knihovny)

```
[Ctrl+C] přerušení běhu programu
>>> promt Micropythonu
a co nám píše

>>> a = 60
>>> print(a/12)
5.0
>>> def suma(a,b):
...     return a+b
...
>>> suma(1,2)
3
>>> octopus() inicializace hlavní knihovny
>>> h() nápověda - help
>>> w() připojení k síti

setup() nastavení (metoda)
```



**ESP32**  
a vývojový modul  
octopusLAB **ROBOTboard**

> opětovné spojení pomocí wifi/BT nebo sériové linky a terminálu například Putty (Win) nebo Screen (Linux)

> přenastavení wifi, konfigurace periférií  
> pokročilejší: nastavení MQTT, úvod do influx DB + Grafana...

Pro výuku programování doporučujeme  
<https://naucese.python.cz/>

**MQTT:** Internet věcí vyžaduje obrovskou škálovatelnost v síťovém prostoru, aby zvládl nárůst zařízení. S miliardami zařízení, které jsou přidávány do internetového prostoru, hraje IPv6 hlavní roli při řešení škálovatelnosti síťové vrstvy. „MQ“ v „MQTT“ pochází z produktové řady IBM MQ řady zpráv: (**Message Queuing Telemetry Transport**) @wiki

MQTT je tedy jednoduchý **centralizovaný protokol** sloužící nejčastěji pro použití s nejrůznějšími senzory IoT (Internetu věcí). Lze jej však využít i pro přenos mnoha jiných, například telemetrických dat. Základem je princip typu **zveřejnit/odebírat (publish/subscribe)**.

Zařízení s funkcí zveřejnit odesílají zprávy zprostředkovateli (**broker**), který na základě přihlášených odběrů provede třídění a přeposlání uživatelům. Klient-uživatel může zároveň publikovat i odebírat.

