

# PRŮVODCE HODINOU IV-1

Studenti se v této hodině naučí, jakým způsobem zjistit orientaci micro:bitu v prostoru a jak tuto informaci využít.

## Co bude v této hodině potřeba:

PC s editorem Mu.

Micro:bit s USB kabelem

Dva vodiče nejlépe s krokodýlky na obou koncích

Reproduktor nebo sluchátka s jackem, popřípadě piezzo buzzer.

Pokud je k dispozici, tak dataprojektor

Prezentaci k této lekci

Pracovní listy pro studenty

## 1. krok 25 minut

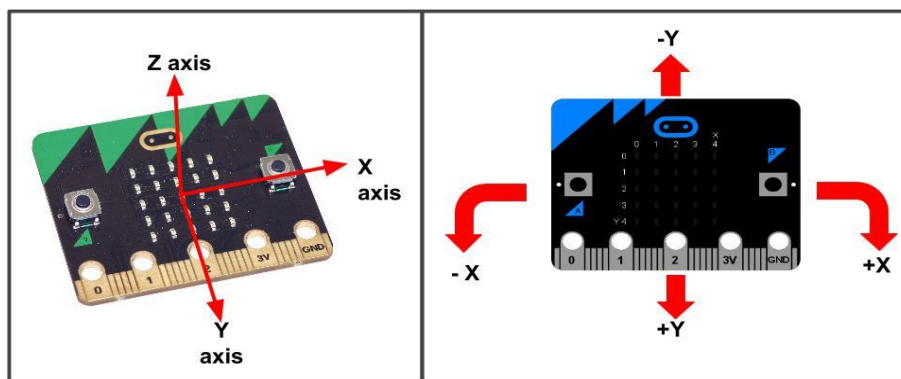
Rozdejte studentům micro:bity a kabely. Řekněte jim ať si připraví sluchátka. Raději mějte připravená sluchátka pro ty, kteří si je zapomenou.

Programy v této lekci jsou "dlouhé". Doporučujeme poskytnout studentům zdrojový kód, a tak vše urychlit.

Vysvětlíte studentům pojem akcelerometr – zařízení pro zjištění aktuální orientace, směru pohybu, zrychlení, popř. volného pádu nebo zrychlení.

Zeptejte se studentů na využití akcelerometru – herní ovladač, generátor událostí. Program reagující na zatřesení, změnu orientace.

Vysvětlíte studentům, jak fungují osy x, y a z u micro:bitu. Je-li položen vodorovně, piny k vám, je osa x rovnoběžně s vámi (a sleduje tedy náklon vlevo a vpravo), osa y pak směrem k vám a sleduje náklon od vás k vám. Osa z v tomto případě není sledovatelná. Použijete jí, pouze pokud je micro:bit „postaven“ na konektorech a naklání se k vám a od vás.



Poloha os a význam plus a minus ve směru os.

Zdroj <http://fibasile.fabcloud.io/microbit-notebook/accelerometer/>

Zapište a odlaďte následující program, ukazující pohyb ve směru osy x (vlevo a vpravo):

```
from microbit import *
mez = 400
while True:
    naklon = accelerometer.get_x()
    if naklon > mez:
        display.show("P")
    elif naklon < -mez:
        display.show("L")
    else:
        display.show("-")
```

Tento program ukazuje princip sledování pohybu micro:bitu ve směru osy x (náklon vlevo a vpravo). Proměnná mez určuje hodnotu, od které považujeme micro:bit za nakloněný vpravo či vlevo.

Nechte studenty sledovat, jak program funguje vzhledem k pozici micro:bitu v prostoru. Nechte je měnit hodnotu proměnné mez. Poté změňte osy. Zcela vždy postačí nahradit `get_x` za `get_y` nebo `get_z`. Toto je vlastně způsob, jak můžete micro:bit použít jako dálkový ovladač, až se naučíte v některé z příštích hodin, přenášet data mezi micro:bity.

## 2. krok 20 minut

Nyní si postavíte jednoduchý simulátor Thereminu. Theremin je nástroj, který ovládáme pohybem rukou bez dotyku nástroje. Pohybem jedné ruky určujeme výšku tónu a pohybem druhé, pak jeho délku.

Zapište a odlaďte následující program:

```

from microbit import *
import music
while True:
    x = accelerometer.get_x()
    y = accelerometer.get_y()
    if (x < -1000):
        ton = "C4"
    elif (x < -700):
        ton = "D4"
    elif (x < -400):
        ton = "E4"
    elif (x < -100):
        ton = "F4"
    elif (x < 200):
        ton = "G4"
    elif (x < 500):
        ton = "A4"
    elif (x < 800):
        ton = "B4"
    else:
        ton = "C5"
    if (y < -500):
        nota = ton
    elif (y < 0):
        nota = ton + ":2"
    elif (y < 500):
        nota = ton + ":4"
    else:
        nota = ton + ":8"
    music.play(nota)

```

Připojte k micro:bitu repráčky (sluchátka), dle postupu z minulých lekcí (mezi piny 0 a GND). Natáčením micro:bitu vpravo a vlevo regulujete výšku tónu, dopředu dozadu jeho délku. Ponechte studenty upravovat rozsah tónů, citlivost atd.