



IMPULS
PRO KARIÉRU
A PRAXI

PROGRAMOVÁNÍ MICRO:BIT PRO SŠ



EVROPSKÁ UNIE
Evropské strukturální a investiční fondy
Operační program Výzkum, vývoj a vzdělávání



MINISTERSTVO ŠKOLSTVÍ,
MLÁDEŽE A TĚLOVÝCHOVY

Jhk.cz



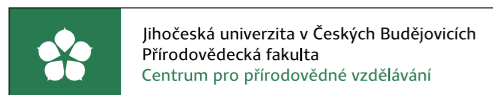
JIHOČESKÁ
HOSPODÁŘSKÁ
KOMORA


Jihočeský kraj

Pracovní postup „Programování Micro:bit pro SŠ“ je součástí publikace „Pracovní postupy pro workshopy digitalizace ve školách.“, která vznikla v rámci aktivity Asistenčního centra Impuls pro kariéru a praxi při Jihočeské hospodářské komoře díky realizaci projektu „Implementace Krajského akčního plánu Jihočeského kraje III“, který je spolufinancován Evropskou unií. Registrační číslo projektu CZ.02.3.68/0.0/0.0/19_078/0018246

Elektronická verze publikace je k dispozici na www.impulsprokarieru.cz

Publikaci připravila: Přírodovědecká fakulta Jihočeské univerzity
Autor: Mgr. Jiří Pech, Ph.D.



Grafický design: Čestmír Sukdol – www.brandi.cz

Vydala: Jihočeská hospodářská komora

2021

Obsah

Základní instrukce / **5**

Teoretická část k dané problematice / **7**

Příklady z praxe / **9**

Metodická a didaktická část / **10**

Doporučené pomůcky / **14**

Pracovní list 1 / Práce s výstupy micro:bitu / **15**

Pracovní list 2 / Práce se vstupy micro:bitu / **17**

Pracovní list 3 / Komunikace mezi dvěma micro:bity / **19**

Pracovní list 4 / Tvorba zabezpečovacího zařízení / **21**

Základní instrukce

Tento kurz je doporučován pro žáky prvního nebo druhého ročníku středních škol anebo vyšší ročníky osmiletých gymnázií. Je doporučen zejména pro studenty technických škol a gymnázií. Zejména u technických škol je dobré počítat ještě s nějakou složitější platformou, např. Arduinem.

Časová dotace tohoto kurzu nemůže být zcela jednoznačná. Závisí na tom, zda se studenti již seznámili s programováním a zda již mají nějaké zkušenosti s Micro:bitem. Učitel by měl nejprve prostudovat všechny materiály připravené k tomuto kurzu a pak se sám rozhodnout, kolik času kurzu věnovat.



Cílem workshopu je představit vývojovou destičku BBC Micro:bit primárně určenou pro výuku programování na základní i střední škole. V našem kurzu se zaměříme na programovací jazyk MicroPython jako podmnožinu jazyka Python. Začneme od úplných základů, ale postupně se přeneseme až k možnosti komunikace dvou Micro:bitů a ukážeme si práci ve dvojici anebo i ve větší skupině. Finálním produktem by mělo být vytvoření zabezpečovacího zařízení tvořeného dvěma Micro:bity, z nichž jeden má funkci sensoru a druhý hlásiče.

Doporučený průběh a časovou dotaci naleznete níže. Kurz lze rozčlenit do čtyř částí:

- 1. Práce s výstupy** – výstup na displej a přehrávání zvuku.
 - a. Pokud již proběhl nějaký kurz Micro:bitu, je možné tuto část přeskočit nebo minimalizovat.
 - b. Pokud žáci již měli nějaké jiné grafické programování např. Scratch, pak postačí jedna vyučovací hodina (45 minut)
 - c. Pokud žáci ještě neměli programování a neznají Micro:bit, pak doporučuji dvě vyučovací hodiny.
- 2. Práce se vstupy** – stisk tlačítka, detekce pohybu, magnetického pole, zvuku, světla, měření teploty.
 - a. Pokud žáci znají Micro:bit a programovali, postačí jedna vyučovací hodina. Pozor, pokud znají Microbit V1, u nového typu jsou některé věci navíc.
 - b. Ve všech ostatních případech doporučuji dvě vyučovací hodiny. Je nutné probrat podmíněné příkazy.
- 3. Komunikace mezi dvěma Micro:bity** – posílání zpráv mezi dvěma Micro:bity pomocí vestavěného radio modulu. Bezpečnost této komunikace.
 - a. Pokud s tímto žáci již pracovali, je možné tuto pasáž zcela přeskočit.
 - b. Jinak doporučuji jednu vyučovací hodinu.
- 4. Samotná tvorba zabezpečovacího zařízení** je už pro všechny stejná. Doporučuji nejméně dvě vyučovací hodiny. V první hodině učitel zvolí jednu z možností zabezpečení a společně s žáky jej zkonstruuje. Ve druhé a případně dalších hodinách žáci samostatně pracují na svých projektech. Je zde možnost přidání jedné nebo více hodin, kdy žáci budou své projekty představovat ostatním.

Minimální doba, za kterou lze tento výukový balík absolvovat, jsou tedy dvě vyučovací hodiny – první hodina: části 1 až 3, druhá hodina: část 4. Doporučený počet je osm hodin (2, 2, 1, 3) včetně představení projektů. Je třeba počítat s možností, že za běhu bude nutné přidat či ubrat hodiny.

Pro části 1 až 3 a první hodinu části 4 je vhodnou metodou výuky výklad spojený se samostatnou prací žáků v hodině. Učitel by měl vždy část látky vysvětlit a pak nechat žáky, aby si vše vyzkoušeli a přišli případně na další možnosti.

Je třeba počítat rovněž s tím, že od části 3 dále by žáci měli pracovat v ideálně dvoučlenných týmech, pokud je málo Micro:bitů tak vícečlenných tak, aby v každém týmu byly nejméně dva Micro:bity.

V závěrečných hodinách pak lze s úspěchem použít projektovou výuku, kdy žáci buď dostanou přidělené téma nebo si jej zcela sami zvolí. V poslední hodině by každý tým měl prezentovat své funkční zabezpečovací zařízení a předvést jeho přednosti a slabiny.

Pro výuku je doporučen Micro:bit verze 2 (Micro:bit V2). Nejlépe pro každého žáka jeden. Doporučuji vybavit se navíc USB kabely a držáky baterií pro běh Micro:bitu mimo počítač. To lze pořídit v kompletu za cca 550 Kč. Nakoupit lze například zde. Současně doporučuji vybavit se náhradními AAA bateriemi.

Máme-li Micro:bit V1, je nutné řešit audio výstup tak, jak je to popsáno v učebnici níže. V tom případě budete potřebovat dva kabely s krokodýlky pro každého a nějaký zdroj zvuku, např. sluchátka.

Pro výuku použijeme programovací jazyk MicroPython, který je podmnožinou jazyka Python. Pro Editor, ve kterém budeme psát programy, máme dvě možnosti:

1. **Thonny** – podle mě lepší volba. Umožňuje program pouze vzdáleně spustit na Micro:bitu bez ukládání na Micro:bit. Ukládání na micro:bit je ale možné též. To se hodí např. při testování zvuku – po stisku reset zvuk skončí. Na druhou stranu není zde kontrola syntaxe.
2. **Mu** – Umožňuje kontrolu syntaxe, ale program musí být před spuštěním na micro:bit uložen.

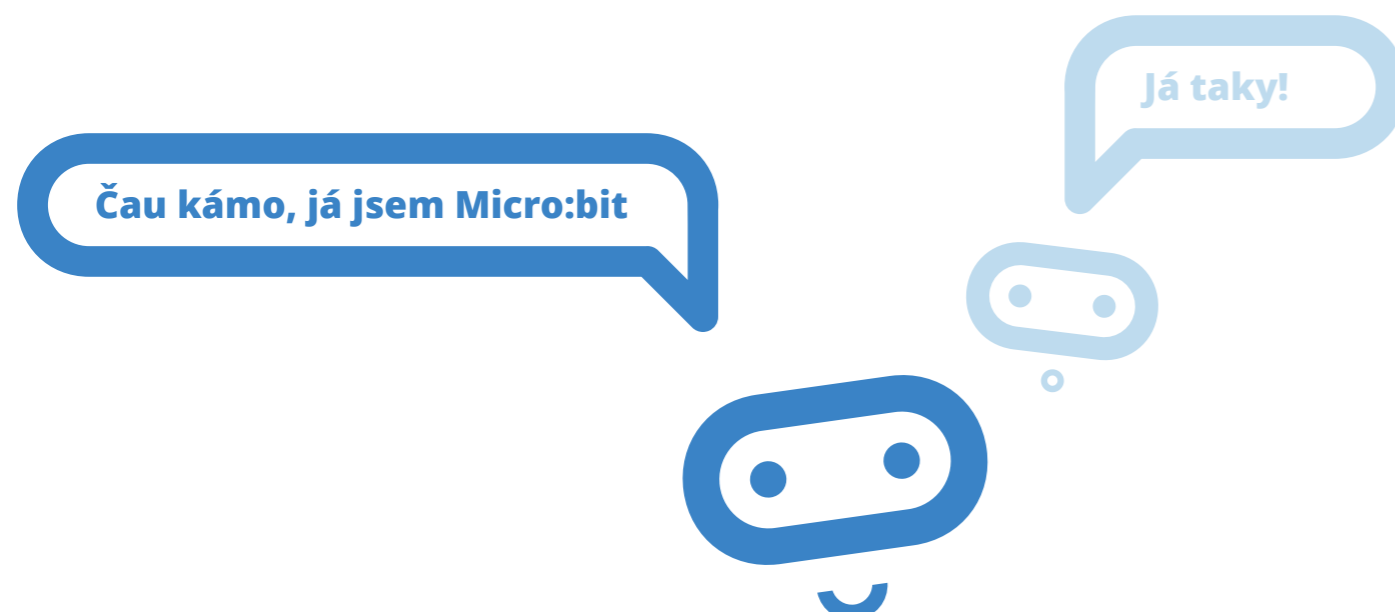
Z hlediska popsaných programů je zcela jedno, který z editorů si zvolíte.

Co se týče podkladů pro výuku, máme několik možností.

Česká učebnice vypracovaná v rámci projektu **iMyšlení** s názvem **Robotika pro střední školy: programujeme Micro:bit pomocí Pythonu** – to by měla být základní literatura pro tento projekt.

O MicroPythonu na micro:bitu jsou následující stránky: **BBC micro:bit MicroPython documentation**.

Mnoho dalších informací a návodů; včetně mnoha zajímavých projektů, lze rovněž najít na adrese **microbiti.cz**, anebo původní webové stránky projektu **BBC Micro:bit**.



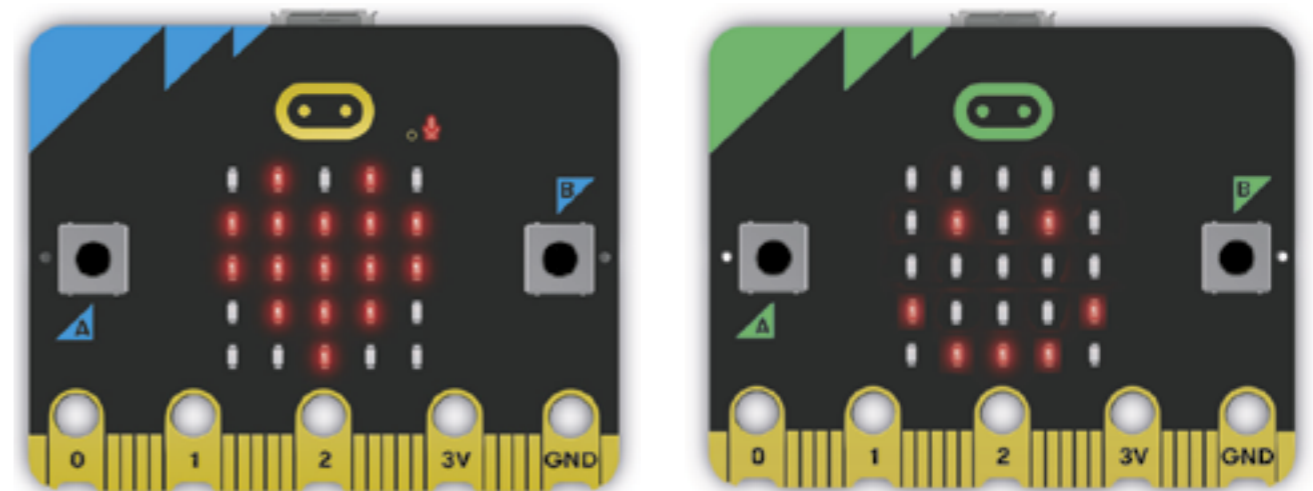
Teoretická část k dané problematice

Micro:bit je kapesní počítač, který vás uvede do světa software a hardware a jejich vzájemné spolupráce. Obsahuje LED displej, tlačítka, sensory a mnoho programovatelných funkcí. Nová verze micro:bitu obsahuje i mikrofon a buzzer. (<http://www.microbit.org>)

Vlastnosti Micro:bitu (V2 znamená pouze verze V2):

- ✓ Matice 5x5 LED diod
- ✓ Dvě programovatelná tlačítka
- ✓ Jedno dotekové tlačítko (V2)
- ✓ Akcelerátor, umožňující reagovat na pohyb
- ✓ Čidlo světla, magnetického pole a teploměr
- ✓ Mikrofon (V2)
- ✓ Buzzer (primitivní reproduktor) (V2)
- ✓ Možnost vzájemné komunikace pomocí radia
- ✓ Bluetooth
- ✓ Konektory pro připojení dalších zařízení, z nich tři pro pohodlné připojení pomocí kabelů s krokodýlky
- ✓ Možnost snadného připojení k rozšiřujícím zařízením pomocí pásu konektorů

Existují dvě verze micro:bitu. Starší verze prodáváná od roku 2016, nyní označovaná jako V1 a novější od října 2020 označovaná jako V2.



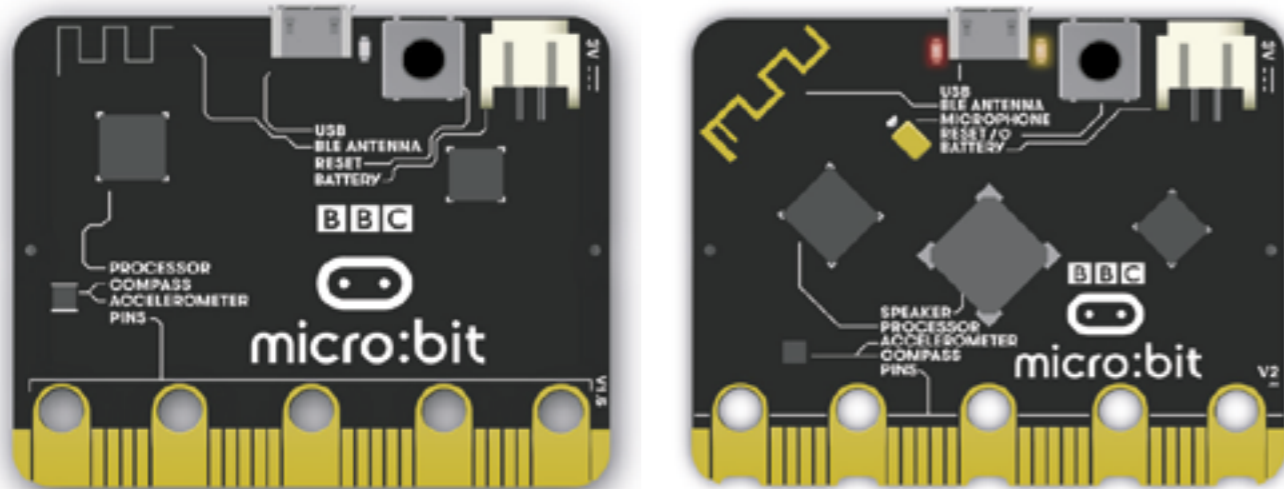
Obrázek 1: čelní strana micro:bitu, vlevo V1, vpravo V2

Rozhodně si nyní pořizujte V2, je lépe vybavená a má větší paměť pro programy při de facto stejné pořizovací ceně. Její výhodou je integrovaný buzzer a mikrofon, takže pro práci s tímto textem nepotřebujete nic jiného než dva micro:bity.

Micro:bit lze programovat jak pomocí grafického programování, tak pomocí klasických textových jazyků.

Grafické programování – takové programování, kdy program netvoříme psaním kódu na klávesnici, ale posouváním grafických bloků po obrazovce. Mezi tyto nástroje patří například Scratch, MakeBlock nebo Open Roberta Lab.

Microsoft MakeCode – grafický programovací jazyk vyvinutý firmou Microsoft pro programování Micro:bitu.



Obrázek 2: Zadní strana micro:bitu, vlevo V1, vpravo V2

V případě micro:bitu lze použít následující „klasické“ textové jazyky JavaScript a MicroPython.

- ✓ JavaScript můžeme použít v online editoru na stránkách microbit.org, pokud se do něj přepneme v programovém prostředí grafického programovacího jazyka MakeCode.
- ✓ MicroPython můžeme opět programovat v online prostředí domovské stránky micro:bitu anebo pomocí již zmíněných editorů Mu a Thonny.



Příklady z praxe

Autora k vytvoření tohoto materiálu přivedla příhoda z praxe, která se stala, když prvním rokem ověřoval svou učebnici na základní škole v rámci testování.

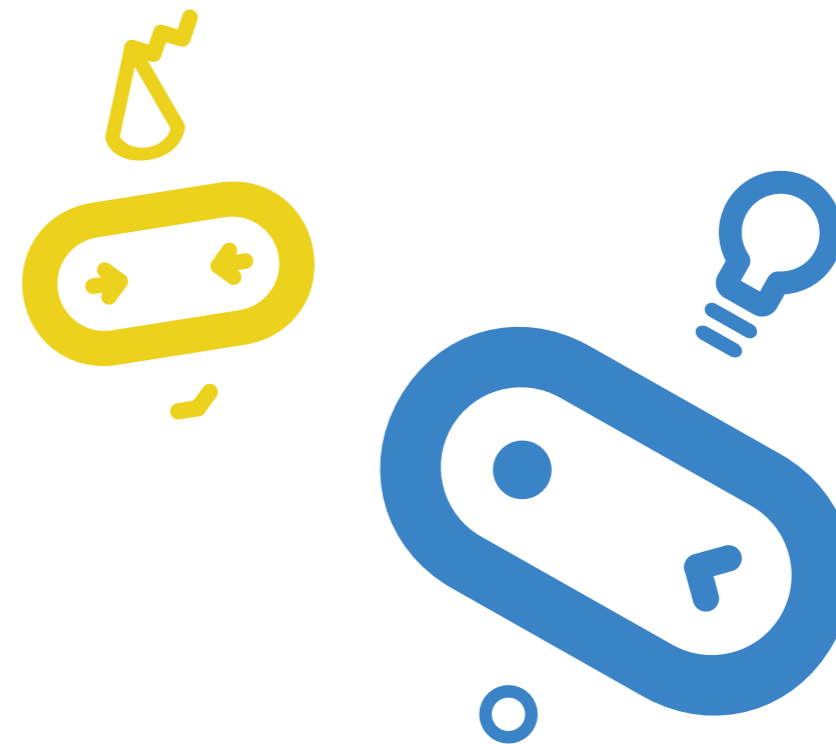
Dva bratři vlastníci dva micro:bity si doma vytvořili primitivní zabezpečovací zařízení. Jeden microbit ukryli na verandě domu a naprogramovali jej, aby reagoval na změnu světla posláním signálu druhému micro:bitu umístěnému v jejich pokoji. Ten měli připojený k reproduktoru, aby je na tuto skutečnost upozornil. Když přišel domů někdo z rodičů, byli o tom včas informováni a mohli odložit tablety, knihy atd. a vzít si do ruky učebnice.

Ačkoliv z pedagogického hlediska s tím samozřejmě nesouhlasím, z didaktického hlediska se jedná o zajímavý případ spojený se spontánní týmovou spoluprací a využitím znalostí získaných v kroužku.

Dalo by se tedy shrnout, že absolvováním tohoto kurzu žáci získají nebo si prohloubí základní programátorské návyky. Dále se dozví informace ze světa hardware a integrace software a hardware. Zde záleží pouze na učiteli, který musí odhadnout úroveň svých žáků a jak hluboko si může dovést ponořit se do dané problematiky.

Dále lze na této problematice dobře vysvětlit možné bezpečnostní problémy bezdrátové komunikace – např. odposlouchávání, man in the middle attack atd.

V neposlední řadě lze tímto úkolem rozvíjet práci ve skupině – pro správnou funkci naprosté většiny případů jsou nutné dva micro:bity. S úspěchem lze na závěr zadat žákům projekt ve skupině, který budou pak muset před ostatními představit a obhájit.



Metodická a didaktická část

Jak bylo vysvětleno v první kapitole, kurz je rozdělen do čtyř částí. Postupně se zastavíme u všech čtyř a řekneme si co v nich učít.

Práce s výstupy

V této části je třeba zvládnout následující úkoly:

- ✓ Seznámit se s micro:bitem
- ✓ Seznámit se s prostředím editoru Mu nebo Thonny
- ✓ Naučit se nahrát vytvořený kód do micro:bitu
- ✓ Práce s displejem – zobrazení ikonky a textu
- ✓ Práce se zvukem

Na úvod rozdává učitel žákům micro:bity a v krátkosti jim je představí. Upozorníte žáky na skutečnost, že micro:bit má na sobě popisky jednotlivých součástí. Vysvětlíte význam jednotlivých tlačítek – A, B programovatelná tlačítka, Reset slouží k opětovnému spuštění programu od začátku anebo k vypnutí micro:bitu při podržení 3 sekundy.

Nyní si studenti spustí zvolený editor. Seznamte je s prostředím a základními principy používání zvoleného editoru. V dalším průběhu budu předpokládat použití mnou doporučeného editoru Thonny.

Zkuste vytvořit jednoduchý program a nahrát jej do micro:bitu. Ukažte žákům rozdíl mezi možnostmi Uložit (program se uloží na disk) a Spustit (program se spustí rovnou do micro:bitu). Vysvětlíte, že po připojení je micro:bit přístupný podobně jako flash disk a uložený program do něj lze nahrát i tak, že jej na tento disk přetáhneme. Ukažte žákům, že během nahrávání bliká žlutá dioda na zadní straně. Také jim ukažte, že ukončený program mohou opět spustit stiskem tlačítka Reset.



```
from microbit import *
display.scroll("Ahoj svete")
```

U dalších programů v této části nebudou již většinou vkládány zdrojové kódy. Naleznete je v příložených pracovních listech.

Nechte žáky upravit program tak, že přidáte blok pro nekonečné opakování. Zkuste přidat časovou prodlevu. Vyzkoušejte zobrazení připravených obrázků, opět s vhodnými časovými prodlevami. Vyzkoušejte případně i cyklus pro pevný počet opakování. Vysvětlíte žákům rozdíly mezi cykly while a for.

Vyzkoušejte si přehrávání zvuku. Pokud máte micro:bit V1, musíte prvně připojit výstupní zařízení. Vyzkoušejte bloky „hraj tón“, „hraj melodii“ a „play sound“ (pouze V2). Můžete nechat žáky složit vlastní melodii, ale v tom případě doporučuji vybavit se klapkami na uši a třídu zvukově izolovat.

Práce se vstupy

- ✓ Kromě pomůcek z minulé kapitoly doporučuji jeden nebo více silných magnetů na třídu.

Nejjednodušším typem vstupu jsou dvě tlačítka značená A a B. Jejich nejjednodušší použití je pomocí bloku „Po stisknutí tlačítka A(B)“ v kategorii vstup. Naprogramujte akci (výstup na displej, zahrání tónu“ po stisku zvoleného tlačítka). Přiřaďte aktivitu k oběma tlačítkům. Např. takto:



```
from microbit import *
while True:
    if button_a.is_pressed():
        display.show(Image.HAPPY)
    if button_b.is_pressed():
        display.show(Image.SAD)
    sleep(100)
display.clear()
```

Vyzkoušejte si nyní další možnosti vstupů.

- ✓ microphone.sound_level() – test na úroveň zvuku u micro:bitu V2
- ✓ pin_logo.is_touched() – stisknuto logo na micro:bitu V2
- ✓ accelerometer.was_gesture("shake") – zatřesení micro:bitem
- ✓ temperature() – teplota
- ✓ display.read_light_level() – úroveň osvětlení
- ✓ compass.get_field_strength() – intenzita magnetického pole
- ✓ compass.heading() – azimut ve stupních, nutná inicializace pomocí compass.calibrate()

Napište program, který po stisku tlačítka A zobrazí teplotu, po stisku tlačítka B intenzitu světla a současném stisku obou tlačítek zobrazí intenzitu magnetického pole. Experimentujte se zhasínáním a rozsvícením a osvětlením a dále s přiblížením magnetu. Napište program, který po stisku tlačítka A zobrazí teplotu, po stisku tlačítka B intenzitu světla a současném stisku obou tlačítek zobrazí intenzitu magnetického pole. Experimentujte se zhasínáním a rozsvícením a osvětlením a dále s přiblížením magnetu.

Můžete na neznalcích vyzkoušet magii. Přiblížíte-li ruku zatátou v pěst k micro:bitu, zobrazí se smajlík. Neříkejte, že máte v té pěstě silný magnet a vyzvěte je, ať to zkusí po vás. Můžete úkol vylepšit tím, že řeknete, že obrázek zobrazí jen tomu, kdo má magický potenciál. Slovo magický pak můžete nahradit slovem magnetický a sledujte kdy si toho dotyčný všimne. Můžete vyzkoušet ještě další magický program, při zatřesení vám dá micro:bit náhodnou odpověď ano/ne. Můžete se ho tak dotazovat na mám/nemám.

Můžete vyzkoušet ještě další magický program, při zatřesení vám dá micro:bit náhodnou odpověď ano/ne. Můžete se ho tak dotazovat na otázky na které je odpověď ano/ne. Můžete přidat i možnosti nevím, nelze rozhodnout atd.

Komunikace mezi dvěma micro:bity

- ✓ Pro splnění této hodiny musí již být vytvořeny nejméně dvoučlenné skupiny vybavené dvěma micro:bity.

Na úvod naprogramujte takovýto jednoduchý programek na obou micro:bitech ve skupině. >>

Je důležité, aby měly různé skupiny různou skupinu rádia (zde 23), jinak se budou vzájemně rušit. Skupinu volíme z intervalu <0,83>. (Nechte nejdříve žáky, aby se přesvědčili, co se stane, když to neudělají. Bude se vám pak dobře vykládat bezpečnost bezdrátového spojení.)

Poté co si žáci rozdělí skupiny rádia, je nechte zkusit si vzájemně posílat signál. Je-li to možné (a bezpečné), vypusťte je mimo učebnu, ať si otestují dosah micro:bitů ve volném prostoru i přes zdi. Bude se to hodit při tvorbě zabezpečovacího zařízení.

Naeditujte složitější případ, kdy se posílá jiné číslo při stisku A a jiné při stisku B. Máte-li V2, tak třetí se pošle při stisku loga. Přijímající micro:bit pak podle přijatého čísla rozhodne o své činnosti.

Žáci mohou kód upravit tak, že význam při poslání signálu z jednoho micro:bitu může být např. „Půjdeme na hřiště?“, „Půjdeme se učit“, „Máme úkoly“ a odpovědi zpět např. „Ano“, „Ne“ a „Nevím“. Ponechte žákům, ať si sami vyberou a řešení otestují. Micro:bity lze nyní označit jako tázající a odpovídající.

Vysvětlete žákům nebezpečí této komunikace. Pokud útočník zná použitou skupinu a význam kódů může konverzaci odposlouchávat. Navíc může i do komunikace vstoupit a způsobit zmatení účastníků.

```
from microbit import *
import radio
kanal = 23
radio.on()
radio.config(channel = kanal)
display.clear()
while True:
    if button_a.is_pressed():
        radio.send("Zprava")
    zprava = radio.receive()
    if (zprava):
        display.scroll(zprava)
        zprava = ""
    sleep(100)
radio.off()
```

Tvorba zabezpečovacího zařízení

Vytvoříme jednoduché zabezpečovací zařízení, které bude reagovat na intenzitu světla.

Vyzkoušejte si nejprve na jednoduchém programu rozdíl v hodnotě intenzity světla mezi zhasnutím a rozsvícením (zavřenými a otevřenými žaluziemi). >>

Potřebujeme dva micro:bity:

- ✓ Vysílač – hlídá intenzitu světla a při rozsvícení (zhasnutí) vyšle signál. Kód by mohl vypadat např. takto: >>

```
from microbit import *
while True:
    display.scroll(display.read_light_level())
```

```
from microbit import *
import radio
kanal = 23
radio.on()
radio.config(channel = kanal)
while True:
    display.scroll(display.read_light_level())
    if (display.read_light_level()>127):
        radio.send("Poplach")
```

- ✓ Přijímač – dostane info, že došlo k incidentu a vyhlásí poplach. Kód například takto: >>

Nyní zadejte žákům projekt na vytvoření zabezpečovacího zařízení pomocí dvou micro:bitů. Napovězte jim, co mohou použít – stisk tlačítka (naopak i uvolnění tlačítka), změna intenzity světla, teploty, magnetického pole, audio signál, pohyb.

```
from microbit import *
import music
import radio
kanal = 23
radio.on()
radio.config(channel = kanal)
display.clear()
while True:
    zprava = radio.receive()
    if (zprava):
        display.scroll(zprava)
        zprava = ""
        music.play(music.NYAN)
    sleep(100)
radio.off()
```

Doporučené pomůcky

V první řadě potřebujeme dostatek micro:bitů. Ideální stav je, když má každý žák svůj micro:bit. Ideálnější je pak, když má svůj osobní micro:bit, pokud si to může škola anebo rodiče žáků dovolit.

Ke každému micro:bitu potřebujeme připojovací USB kabel a doporučuji i přenosný držák na baterie. Ideální je koupit vše dohromady jako výhodný balíček, jak je popsáno v úvodu textu.

Dále potřebujeme počítač, který má nainstalovaný editor Mu nebo Thonny, anebo asi nejlépe oba. Určitě doporučuji druhou možnost, nebudeme tolik ovlivněni vrtochy internetového připojení.

Pro vysvětlení reakce na magnetické pole potřebujeme také silnější magnet (např. „vypůjčený“ z nástěnky).

Pokud máme starší micro:bit verze jedna, pak ještě potřebujeme externí repráčky anebo sluchátka.

Pracovní list

Přílohou tohoto materiálu jsou pracovní listy. Tyto pracovní listy jsou k dispozici v editovatelné elektronické formě, aby si je každý učitel mohl upravit, např. dle toho, co má již s žáky probráno.

Pracovní listy jsou čtyři, pro každé z výše uvedených témat jeden. V pracovních listech počítám s tím, že škola má k dispozici Micro:bit V2. Pokud má k dispozici pouze starší verzi V1, je nutné tyto listy upravit. Rozdíly jsou zejména v tom, že u V1 je nutno pro přehrání zvuku připojit externí zařízení a dále V1 nemá mikrofon a dotekové tlačítko, takže tyto dvě zařízení nelze použít jako čidlo pro vyhlášení poplachu.

Pracovní list 1 Práce s výstupy micro:bitu

Co budeme potřebovat

- ✓ Počítač s nainstalovaným editorem Thonny nebo Mu
- ✓ Micro:bit V2
- ✓ Propojovací USB kabel

Ajdeme na to

Vezměte si svůj micro:bit a pořádně si jej prohlédněte. Na přední straně máte pole 5x5 LED diod, které umí svítit různou intenzitou červené barvy. Po jejich stranách máte dvě programovatelná tlačítka označení A a B. Nad diodami se ještě nachází programovatelné dotekové tlačítko a vpravo nahoře nad LED je malá tečka, což je vlastně mikrofon. Otočme nyní micro:bit a podívejme se na zadní stranu. Nahoře máme dva porty microUSB pro připojení k PC a konektor pro připojení baterie packu. Mezi nimi je tlačítko RESET pro opětovné spuštění programu od začátku. Všimněte si, že další části jsou zde popsány. Pro úplnost ještě dodejme, že z obou stran dole vidíme piny pro připojení dalších periférií. Celý micro:bit je například možné zasunout do nějakého externího zařízení.

Spusťte editor Thonny anebo Mu a seznamte se s ním.

Zkuste vytvořit jednoduchý program a nahrát jej do micro:bitu. Zkuste si vysvětlit mezi možnostmi spustit a uložit jako u Thonny. Všimněte si, že během nahrávání bliká žlutá dioda na zadní straně. Vyzkoušejte si i to, že program lze spustit opakovaně pomocí tlačítka RESET (anebo odpojením a připojením micro:bitu ke zdroji energie).

Zkuste přidat blok while True (pozor u True musí být velké písmeno) a přidat časovou prodlevu. Vyzkoušejte zobrazení připravených obrázků, opět s vhodnými časovými prodlevami.

Upravíme program, tak aby text zobrazil třikrát:

Pozor - pro tři zobrazení opravdu musí rozsah být 1 až 4. Je to vlastně polozavřený (zleva) interval.

```
from microbit import*
display.scroll("Ahoj svete")
```

```
from microbit import*
while True:
    display.scroll("Ahoj svete")
    sleep(100)
```

```
from microbit import*
for i in range(1, 4):
    display.scroll("Ahoj svete")
sleep(1000)
display.clear()
```


Vyzkoušejte si přehrávání zvuku:



```
from microbit import *
import music
display.show(Image.HAPPY)
while True:
    music.play(music.POWER_UP)
    sleep(1000)
```

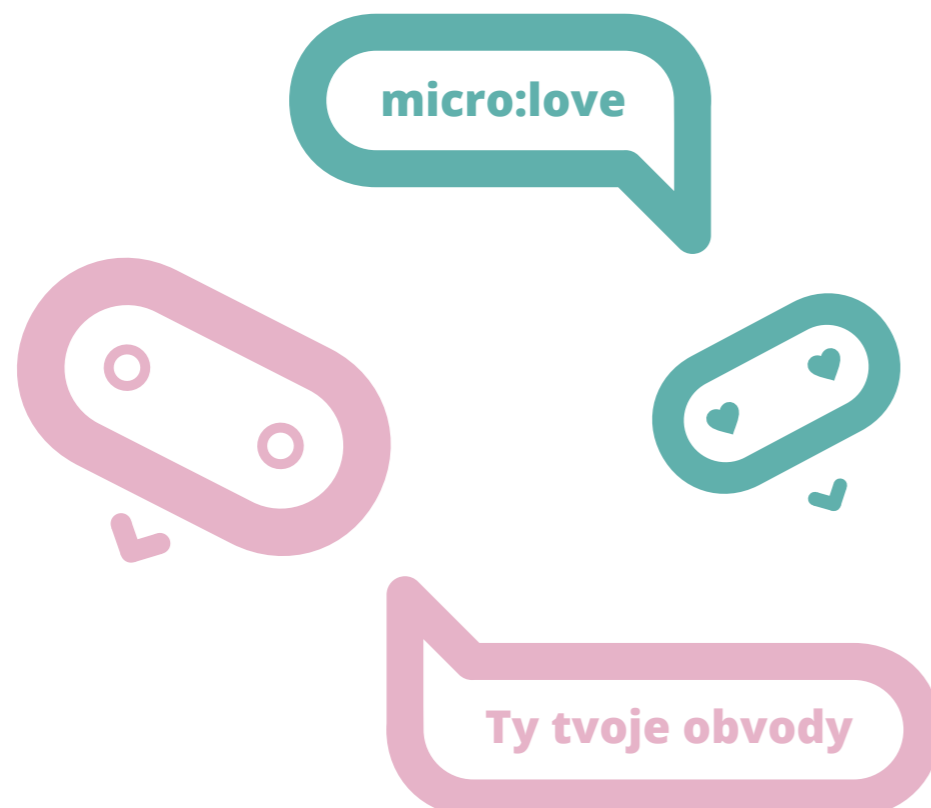
Až vás začnou bolet uši, stiskněte tlačítko reset na micro:bitu (pouze u Thonny). Můžete nechat micro:bit i promluvit:



```
from microbit import *
import speech
speech.say("Attention please", speed=100)
```

Co jste se naučili

Zobrazovat obrázky a text na displeji micro:bitu. Víte, jak část kódu spustit jednou, opakovaně, několikrát. Dále umíte na micro:bitu přehrát tón a nechat jej promluvit.



Pracovní list 2 Práce se vstupy micro:bitu

Co budeme potřebovat

- ✓ Počítač s nainstalovaným editorem Thonny nebo Mu
- ✓ Micro:bit V2
- ✓ Propojovací USB kabel

A jdeme na to

Nejjednodušším typem vstupu jsou dvě tlačítka značená A a B. Micro:bit V2 navíc dotykové logo. Použití dobře demonstruje následující příklad:



```
from microbit import *
import music
while True:
    if button_a.is_pressed():
        display.show(Image.HAPPY)
    elif button_b.is_pressed():
        display.show(Image.SAD)
    elif pin_logo.is_touched():
        music.play(music.DADADADUM)
    sleep(100)
    display.clear()
```

Jedná se o podmíněný příkaz. Pokud platí první podmínka, provede se a skočí se na konec bloku, a tak dále. Část jinak (else) zde chybí.

Nyní vyzkoušejte měřit další veličiny a zobrazit jejich hodnotu: intenzitu světla, magnetického pole a teplotu. Můžete vyzkoušet i azimut. Nezapomeňte, že před použitím se musí micro:bit zkalibrovat (micro:bit napíše výzvu, pak se objeví uprostřed displeje bod a natáčením micro:bitu musíme celý displej zaplnit).

Použijte následující příkazy:

- ✓ microphone.sound_level() – test na zvuk u micro:bitu V 2
- ✓ pin_logo.is_touched() – stisknuto logo namicro:bitu V 2
- ✓ accelerometer.was_gesture("shake") – zatřesení
- ✓ temperature() – teplota
- ✓ display.read_light_level() – úroveň osvětlení
- ✓ compass.get_field_strength() – intenzita magnetického pole
- ✓ compass.heading() – azimut ve stupních, nutná inicializace pomocí compass.calibrate()

Napište program, který po stisku tlačítka A zobrazí teplotu, po stisku tlačítka B intenzitu světla a současném stisku obou tlačítek zobrazí intenzitu magnetického pole. Experimentujte se zhasínáním a rozsvícením a osvětlením a dále s přiblížením magnetu.

Zkuste podobně experimentovat s úrovní hlasitosti v okolí.

Můžete na neznalcích vyzkoušet magii:

Přiblížíte-li ruku zaťatou v pěst k micro:bitu, zobrazí se smajlík. Neříkejte, že máte v té pěsti silný magnet a vyzvěte je, ať to zkusí po vás. Můžete úkol vylepšit, tím že řeknete, že obrázek zobrazí jen tomu, kdo má magický potenciál. Slovo magický pak můžete nahradit slovem magnetický a sledujte kdy si toho dotýčný všimne.

Můžete vyzkoušet ještě další magický program, při zatřesení vám dá micro:bit náhodnou odpověď ano/ne. Můžete se ho tak dotazovat na mám/nemám.

Co jste se naučili

Programovat vstupní události, např. stisk tlačítka nebo zatřesení microbitem. Naučili jste se odečítat fyzikální veličiny z reálného světa – teplotu, světlo, hluk a magnetické pole. Vše toto později využijete v tvorbě zabezpečovacích zařízení.

```
from microbit import *
hodnota = 5000
compass.calibrate()
pocatek = compass.get_field_strength()
while True:
    sleep(100)
    sila = compass.get_field_strength()
    if abs(sila - pocatek) > hodnota:
        display.show(Image.HAPPY)
        sleep(3000)
        display.clear()
```

Pracovní list 3 Komunikace mezi dvěma micro:bity

Co budeme potřebovat

- ✓ Počítač s nainstalovaným editorem MakeCode anebo s přístupem na internet
- ✓ 2 Micro:bity nejlépe V2
- ✓ Propojovací USB kabel
- ✓ Doporučuji pro oba Micro:bity použít Baterý Box – pro testování spojení v „terénu“

A jdeme na to

V této a následující hodině musí být vytvořeny nejméně dvoučlenné skupiny vybavené dvěma micro:bity.

Na úvod naprogramujte takovýto jednoduchý prográmk na obou micro:bitech ve skupině.

Je důležité, aby měly různé skupiny různou skupinu rádia (zde 23) z intervalu <0,83>, jinak si vzájemně polezete do zelí. Signál by měly obdržet všechny micro:bity se stejnou skupinou.

Zkoušejte si posílat vzájemně signál. Připojte micro:bity na baterý pack a naopak je odpojte z USB kabelu a vezměte je na procházku ven a vyzkoušejte dosah signálu.

Pro testování si také můžete nahrát na jeden micro:bit, který ponecháte připojený u počítače následující program a jít „na procházku“ jen s jedním micro:bitem, na kterém ponecháte předchozí program a zjistit tak dosah. Nebo upravte program na micro:bitu, aby posílal automaticky zprávy po dvou vteřinách a vyzkoumáte tak dosah micro:bitu z učebny.

```
from microbit import *
import radio
kanal = 23
radio.on()
radio.config(channel = kanal)
display.clear()
while True:
    if button_a.is_pressed():
        radio.send("Zprava")
        zprava = radio.receive()
        if (zprava):
            display.scroll(zprava)
            zprava = ""
        sleep(100)
radio.off()
```

Pozor - pokud připojíte Micro:bit k počítači, vždy nejdříve odpojte Battery Pack, aby nedošlo k poškození Micro:bitu vyšším napětím.

Naeditujte složitější případ, kdy se posílá jiné číslo při stisku A a jiné při stisku B. Přijímající micro:bit pak podle přijatého čísla rozhodne o své činnosti (např., kterou zprávu zobrazí).

Máte-li V2, upravte příklad tak, že třetí číslo se pošle při stisku loga.

Upravte kód tak, že význam při poslání signálu z jednoho micro:bitu může být např. „Půjdem na hřiště?“, „Půjdeme se učit?“, „Máme úkoly?“ a odpovědi zpět např. „Ano“, „Ne“ a „Nevím“. V takovémto případě už bude nutné mít v micro:bitu různý kód.

Uvědomte si nebezpečí této komunikace. Pokud útočník zná použitou skupinu a význam kódů, může konverzaci odposlouchávat. Navíc může i do komunikace vstoupit a způsobit zmatení účastníků.

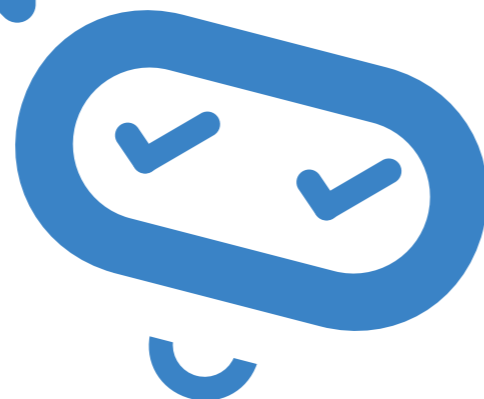
Co jste se naučili

Jak komunikují dva micro:bity.

Poslat zprávu z jednoho micro:bitu na druhý a dešifrovat její význam.

Rovněž jste se dozvěděli něco o bezpečnosti bezdrátové komunikace.

Easy! Nebo ne?



Pracovní list 4

Tvorba zabezpečovacího zařízení

Co budeme potřebovat

- ✓ Počítač s nainstalovaným editorem Thonny nebo Mu
- ✓ 2 Micro:bity nejlépe V2
- ✓ Propojovací USB kabel
- ✓ Doporučuji pro oba Micro:bity použít Batery Box – možnost umístění i mimo počítač

A jdeme na to

Vytvoříme jednoduché zabezpečovací zařízení, které bude reagovat na intenzitu světla. Budeme opět pracovat ve dvojicích.

Vyzkoušejte si nejprve na jednoduchém programu rozdíl v hodnotě intenzity světla mezi zhasnutím a rozsvícením (zavřenými a otevřenými žaluziemi).



```
from microbit import *
while True:
    display.scroll(display.read_light_level())
```

Stanovte si hodnotu, která pro vás bude znamenat dělicí hodnotu mezi světlem a tmou pro vyhlášení poplachu.

Nyní již vytvoříme jednoduché zabezpečovací zařízení. Hodnotu 128 můžete nahradit odpozorovanou hodnotou z předchozího programu. Tentokrát budeme nahrávat rozdílné programy na každý Micro:bit.

- ✓ Vysílač – hlídá intenzitu světla a při rozsvícení (zhasnutí) vyšle signál – poplach. Kód by mohl vypadat např. takto:



```
from microbit import *
import radio
kanal = 23
radio.on()
radio.config(channel = kanal)
while True:
    display.scroll(display.read_light_level())
    if (display.read_light_level()>127) :
        radio.send("Poplach")
```

- ✓ Přijímač – dostane info, že došlo k incidentu a vyhlásí poplach. Kód by mohl vypadat například takto:



```
from microbit import *
import music
import radio
kanal = 23
radio.on()
radio.config(channel = kanal)
display.clear()
while True:
    zprava = radio.receive()
    if (zprava):
        display.scroll(zprava)
        zprava = ""
        music.play(music.NYAN)
        sleep(100)
radio.off()
```

Zkuste upravit kód přijímače, tak aby opakoval varování, dokud není např. stisknuto tlačítko. (potvrzení poplachu)

Nyní vytvořte libovolné zabezpečovací zařízení pomocí dvou micro:bitů.

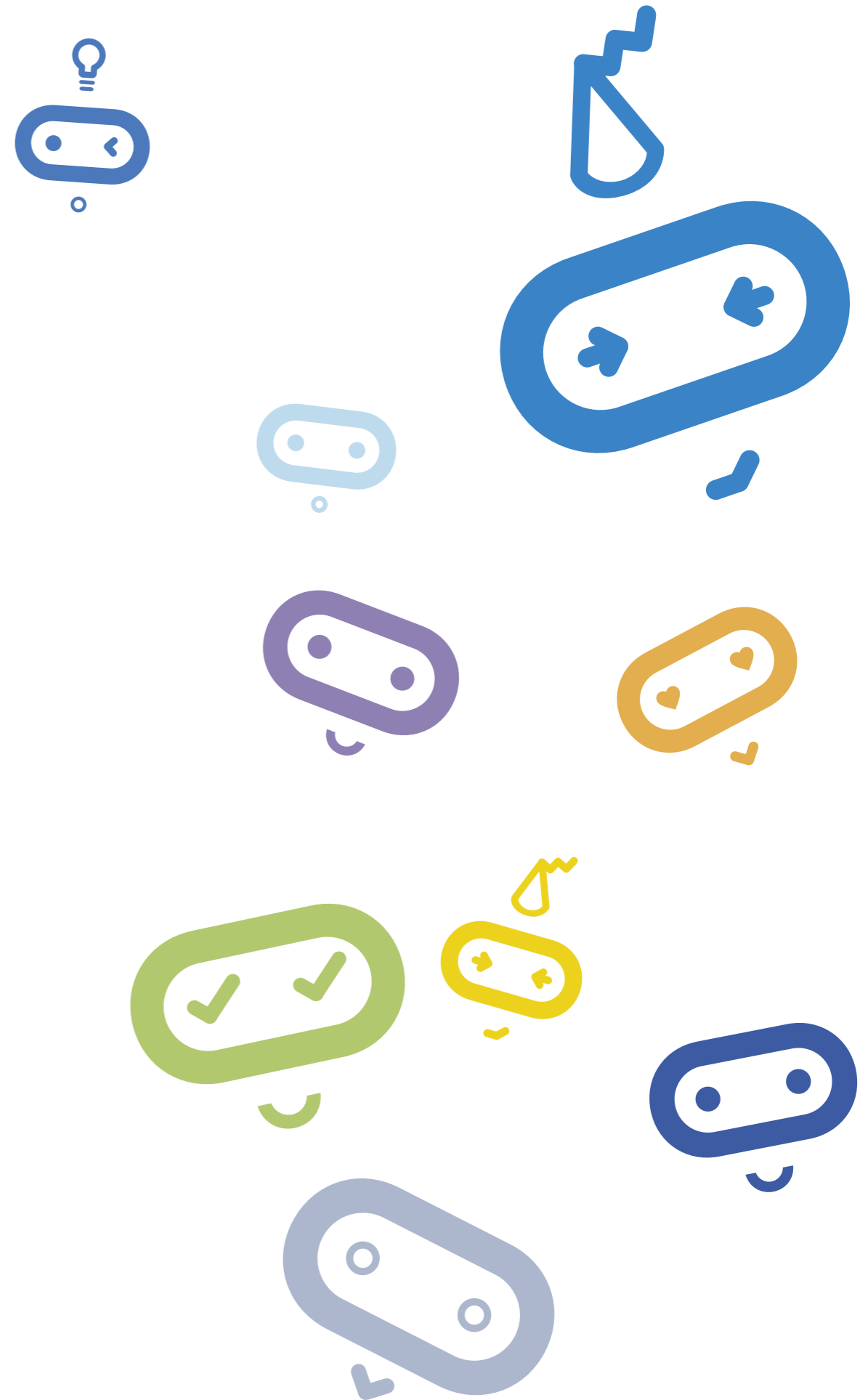
Můžete vyzkoušet různé typy událostí:

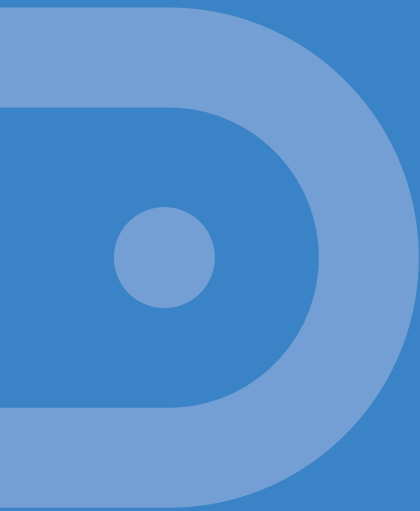
- ✓ stisk tlačítka (naopak i uvolnění tlačítka)
- ✓ změna intenzity světla
- ✓ změna teploty
- ✓ změna intenzity magnetického pole
- ✓ audio signál (zvuk)
- ✓ pohyb micro:bitu (zatřesení)

Co jste se naučili

Znalosti získané v tomto kurzu jsou vlastně znalostmi ze světa robotiky. Umíte nyní sledovat své okolí a zjišťovat jeho změny – zde se jedná o analogii s lidskými smysly. Na jednu stranu sice nemáte smysly „chuť a zrak“, ale máte zde naopak smysl intenzita magnetického pole. Na tyto události nyní umíte reagovat zprávami na displej či zvukem. Umíte rovněž komunikovat s jiným micro:bitem na dálku.

Je třeba si uvědomit, že ačkoliv tento kurz vás mnoho naučil, záleží pouze na vás, jak se získanými dovednostmi naložíte a zda je dále budete prohlubovat pomocí dalších kurzů či další literatury.





IMPULS
PRO KARIÉRU
A PRAXI