

3 HUDBA

Co se naučíte

- Připojit k micro:bitu reproduktor, buzzer nebo sluchátka
- Přehrát přednastavený zvuk
- Naučíte micro:bit mluvit
- Vytvořit vlastní melodii

Co budete potřebovat

- PC s nainstalovaným editorem mu
- Propojovací USB kabel micro USB koncovkou
- Micro:bit
- Dva vodiče nejlépe s krokodýlky na obou koncích
- Reproduktor nebo sluchátka s jackem, popřípadě piezzo buzzer.

1.PRŮVODCE HODINOU III-1

Studenti si připojí k micro:bitu hardware pro přehrání zvuku a naučí se na micro:bitu přehrávat zvuk.

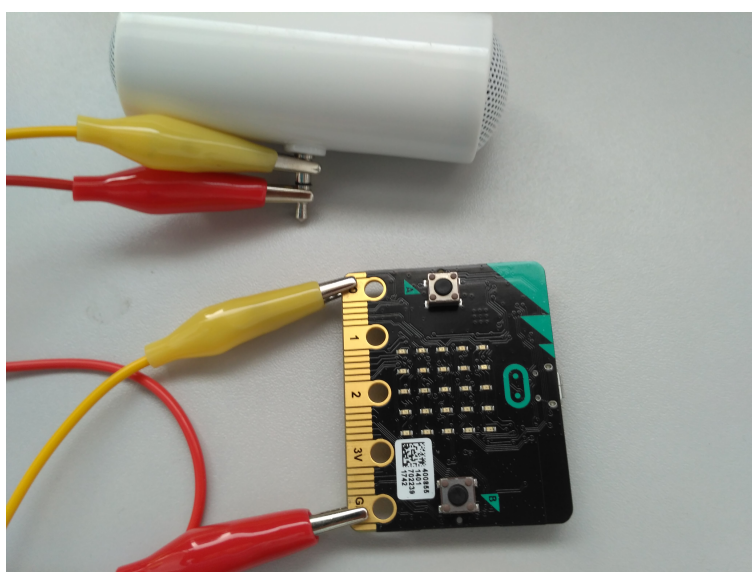
Co bude v této hodině potřeba:

- 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 15 minut

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

Micro:bit nemá přímý audio výstup, ale připojení externího reproduktoru je velmi snadné. Budete potřebovat dva vodiče nejlépe opatřené na koncích krokodýly. Ty na dolní straně micro:bitu připnete jeden na GND a druhý na 0. Druhý konec vodičů připojte na jack libovolného reproduktoru či sluchátek. Nezáleží na pořadí který vodič kam připojíte. Má-li váš jack tři vstupy, pak jeden z vodičů připojte na prostřední a druhý na libovolný z krajních vodičů. Má-li čtyři vstupy, pak by měly fungovat buď oba krajní nebo oba vnitřní (možná budete muset trochu experimentovat). Také můžete použít jako výstup piezzo buzzer, pak prostě připojíte každý vodič k jednomu z pinů. Viz následující obrázek.



Počítejte s tím, že v případě kvalitních reproduktorů může být výstup poměrně hlasitý a nastavte výstup na nižší úroveň.

2. krok 15 minut

Vyzkoušejte přehrávání na připravené melodii. Zapište následující kód, odlad'te a nahrajte do micro:bitu:

```
1. from microbit import *
2. import music
3.
4. music.play(music.NYAN)
```

Na řádce 2 je informace o přidání knihovny pro přehrávání zvuku. Na řádce 4 je příkaz pro přehrávání přednastavené melodie.

Seznam připravených melodií je na konci této kapitoly. Podobně jako u přednastavených obrázků jej vhodným způsobem poskytněte studentům.

Odlad'te u všech studentů přehrávání hudby. Pokud někomu hudba nehraje, zkuste jiný hardware. Tato melodie je vhodná pro testování – je dlouhá a výrazná.

3. krok 15 minut

Nyní se zkombinuje vše co již žáci znají. Zobrazení obrázku, práce s tlačítky a přehrávání melodie:

```
1. from microbit import *
2. import music
3.
4. while True:
5.     if button_a.is_pressed():
6.         display.show(Image.HAPPY)
7.         music.play(music.POWER_UP)
8.     if button_b.is_pressed():
9.         display.show(Image.SAD)
10.        music.play(music.POWER_DOWN)
11.        display.clear()
```

Tomuto příkladu by již žáci měli rozumět. Ověřte.

Pokud zbývá čas, nechte studenty upravit předchozí příklad dle nálady.

2.PRACOVNÍ LIST III-1

Naučíte se k micro:bitu připojit sluchátka a přehrát na nich melodii.

Co se naučíte

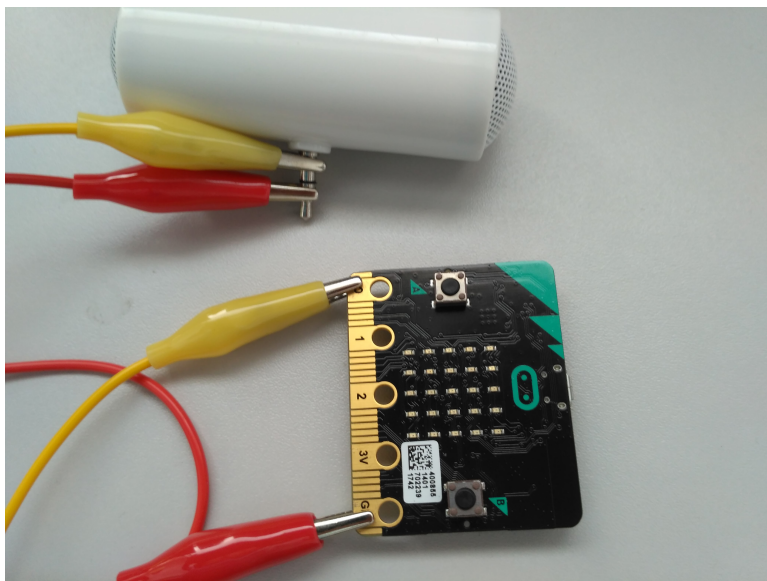
- Připojit k micro:bitu hardware na výstup zvuku
- Přehrát předpřipravenou melodii a zkombinovat jí se zobrazením obrázku

Co budete potřebovat

- PC s nainstalovaným editorem mu
- Propojovací USB kabel micro USB koncovkou
- Micro:bit
- Dva vodiče nejlépe s krokodýlky na obou koncích
- Reproduktor nebo sluchátka s jackem, popřípadě piezzo buzzer.

A jděte na to ...

Připojte k micro:bitu sluchátka nebo repráček dle následujícího obrázku:



Připojení sluchátek či reproduktoru je velmi snadné. Budete nyní potřebovat dva vodiče nejlépe opatřené na koncích krokodýly. Ty na dolní straně micro:bitu připněte jeden na GND a druhý na 0. Druhý konec vodičů připojte na jack libovolného reproduktoru či sluchátek. Nezáleží na pořadí, který vodič kam připojíte. Má-li váš jack tři vstupy, pak jeden z vodičů připojte na prostřední a druhý na libovolný z krajních vodičů. Má-li čtyři vstupy, pak by měly fungovat buď oba krajní nebo oba vnitřní (možná budete muset trochu experimentovat).

Nyní nahrajte do micro:bitu následující program:

```
1. from microbit import *
2. import music
3.
4. music.play(music.NYAN)
5. Příkaz na řádku 2 zavádí knihovnu pro práci se zvukem a na
   řádku 4 se přehraje připravený zvuk. Tento zvuk je celkem
   dlouhý a tak je vhodný pro testování.
```

Seznam všech připravených melodií vám poskytne vyučující.

Pokud máte program v pořádku nahrán na micro:bitu nasadte si sluchátka. Pokud neslyšíte tón stiskněte tlačítko reset na micro:bitu. Pokud ani nic nyní neslyšíte zkuste jiné konektory na jacku sluchátek. Pokud to nepomůže, zkuste jiná sluchátka či jiný micro:bit.

Nyní si zkombinujeme vše co už znáte. Zobrazení obrázku, stisk tlačítek a přehrání zvuku. Nahrajte následující kód do micro:bitu a vyzkoušejte:

```
1. from microbit import *
2. import music
3.
4. while True:
5.     if button_a.is_pressed():
6.         display.show(Image.HAPPY)
7.         music.play(music.POWER_UP)
8.     if button_b.is_pressed():
9.         display.show(Image.SAD)
10.        music.play(music.POWER_DOWN)
11.        display.clear()
```

Jaký je význam jednotlivých řádků?

Zkuste si program upravit s jinými obrázky a melodiemi.

3.PRŮVODCE HODINOU III-2

Studenti si na micro:bitu připraví vlastní melodii a naučí jej mluvit.

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 10 minut

Napište a odlaďte následující program:

```
1. from microbit import *
2. import speech
3.
4. speech.say("Hello", speed=100)
```

Na řádce 2 se zavádí knihovna pro hovor a na řádce 4 je zadán příkaz pro mluvení. Zde micro:bit pozdraví. Parametr `speed=100` je nepovinný a je možné je vynechat včetně čárky. Defaultní hodnota je 72, ale přijde nám, že při této hodnotě mluví micro:bit příliš rychle. Čím vyšší číslo, tím je řeč pomalejší a naopak.

Pozor micro:bit mluví pouze anglicky a nelze použít české znaky. Např. „Josef“ je třeba napsat jako „Yoseph“ atd.

Dokumentace doporučuje zapojit pro hovor sluchátka (repráky) mezi porty 0 a 1 (a ne 0 a GND jako u hudby). A skutečně zvuk je v tomto případě silnější a čistší než mezi 0 a GND. Řekněte to žákům. Nezapomeňte na další části vodiče navrátit mezi 0 a GND.

Řekněte studentům, ať zkusí naučit micro:bit říkat jejich jméno a příjmení (bez háčeků a čárek).

2. krok 20 minut

Rozdejte studentům micro:bity a kabely. Nechte je připojit sluchátka. Pak je nechte napsat a odladit následující program, který přehraje melodii ovčáci čtveráci. V tomto případě je možné pro zmenšení počtu chyb tento program vhodným způsobem studentům vysdílet. Melodie je poměrně neumělá, pokud máte mezi studenty hudebníky, určitě jí upraví:

```
1. from microbit import *
2. import music
3.
4. nota = ["C4:4", "R:1", "E4:4", "R:1", "G4:4", "R:4", "C4:4",
5.         "R:1", "E4:4", "R:1", "G4:4", "R:4",
6.         "E4:2", "R:1", "E4:2", "R:1", "D4:2", "R:1", "E4:2",
7.         "R:2", "F4:2", "R:1", "D4:2", "R:1",
8.         "E4:2", "R:1", "E4:2", "R:1", "D4:2", "R:1", "E4:2",
9.         "R:2", "F4:2", "R:1", "D4:2", "R:1",
10.        "E4:4", "R:1", "D4:4", "R:1", "C4:4"]
11. music.play(nota)
```

Datová struktura `nota` je **seznam**, který by již měli studenti znát. Zkuste se jich zeptat.

Význam jednotlivých tónů je: C4:4 znamená nota C ve čtvrté oktávě (0 – nejnížší, 8 – nejvyšší) o délce 4. Nota R znamená pauzu (rest). Příkaz `music.play(nota)` pak daný záznam přehraje.

3. krok 15 minut

Vyzvěte studenty ať si sestaví vlastní melodii nebo ať naprogramují přehrání nějaké známé melodie.

Doporučení

Touto hodinou končí úvodní část seznamování s micro:bitem. Nyní se nabízí možnost zadání nějaké samostatné nebo týmové práce.

Navrhujeme, abyste nyní studentům zadali po dvojicích (nebo i větších skupinách) následující úlohu: Vytvořte pomocí dvou nebo tří micro:bitů animaci s melodií. Jeden micro:bit se bude starat o animaci a druhý k tomu bude hrát melodii. Popřípadě na třetím micro:bitu může probíhat nějaký hovor. Upozorněte studenty, že je třeba se nějak synchronizovat, např. současně stisknout tlačítka na obou micro:bitech. Později se studenti naučí též synchronizaci pomocí rádia, která by byla vhodnější, ale zatím jí nemůžete použít.

PRACOVNÍ LIST III-2

Naučíte se na micro:bitu přehrát vlastní melodii a naučíte jej mluvit.

Co se naučíte

- Naučíte micro:bit mluvit
- Naeditovat vlastní melodii pomocí not a přehrát jí.

Co budete potřebovat

- PC s nainstalovaným editorem mu
- Propojovací USB kabel micro USB koncovkou
- Micro:bit
- Dva vodiče nejlépe s krokodýlky na obou koncích
- Reproduktor nebo sluchátka s jackem, popřípadě piezzo buzzer.

A jděte na to ...

Napište a odlaďte následující program:

```
1. from microbit import *
2. import speech
3.
4. speech.say("Hello", speed=100)
```

Na řádku 2 se zavádí knihovna pro hovor a na řádku 4 je zadán příkaz pro mluvení. Zde micro:bit pozdraví. Parametr speed=100 je nepovinný a je možné je vynechat včetně čárky. (Defaultní hodnota je 72, ale při této hodnotě mluví micro:bit příliš rychle. Čím vyšší číslo, tím je řeč pomalejší a naopak.)

Pozor micro:bit mluví pouze anglicky a nelze použít české znaky. Např. „Josef“ je třeba napsat jako „Yoseph“ atd.

Pokud se vám zdá, že micro:bit mluví potichu, zapojte sluchátka mezi 0 a 1.

Zkuste naučit micro:bit říkat své jméno a příjmení (bez háčeků a čárek).

Připojte si opět sluchátka k micro:bitu mezi 0 a GND a pak přeložte a odlaďte následující program:

```
1. from microbit import *
2.
3. import music
4.
5. nota = ["C4:4", "R:1", "E4:4", "R:1", "G4:4", "R:4", "C4:4",
6.         "R:1", "E4:4", "R:1", "G4:4", "R:4",
7.         "E4:2", "R:1", "E4:2", "R:1", "D4:2", "R:1", "E4:2",
8.         "R:2", "F4:2", "R:1", "D4:2", "R:1",
9.         "E4:2", "R:1", "E4:2", "R:1", "D4:2", "R:1", "E4:2",
10.        "R:2", "F4:2", "R:1", "D4:2", "R:1",
11.        "E4:4", "R:1", "D4:4", "R:1", "C4:4"]
12. music.play(nota)
```

Program by měl hrát melodii „Ovčáci čtveráci“. Pokud máte hudební sluch a vyznáte se v notách, můžete melodii zkusit upravit. Význam jednotlivých tónů je: C4:4 znamená nota C ve čtvrté oktávě (0 – nejnižší, 8 – nejvyšší) o délce 4. Nota R znamená pauzu (rest). Příkaz `music.play(nota)` pak daný záznam přehraje.

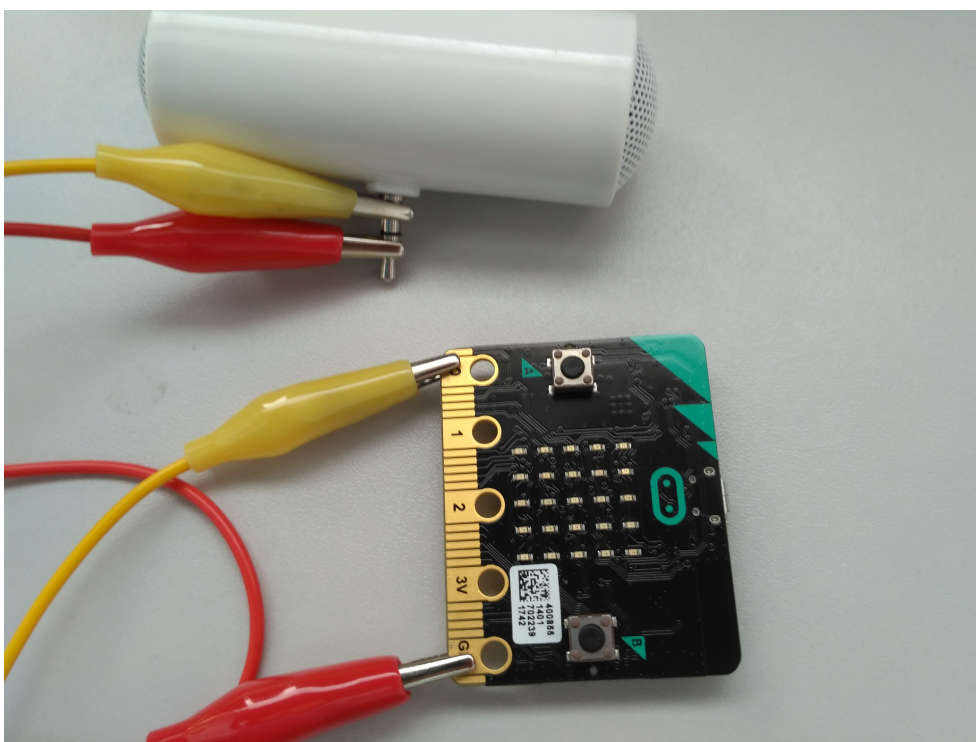
Otázka: Co je za strukturu `nota`?

Zkuste si naprogramovat vlastní melodii nebo nějakou známou skladbu.

PRŮVODCE TEORIÍ

Připojení audio výstupu

Micro:bit nemá přímý audio výstup, ale připojení externího reproduktoru je velmi snadné. Budete nyní potřebovat dva vodiče nejlépe opatřené na koncích krokodýly. Ty na dolní straně micro:bitu připnete jeden na GND a druhý na 0. Druhý konec vodičů připojte na jack libovolného reproduktoru či sluchátek. Nezáleží na tom který vodič kam připojíte. Má-li váš jack tři vstupy, pak jeden z vodičů připojte na prostřední a druhý na libovolný z krajních vodičů. Má-li čtyři vstupy, pak by měly fungovat buď oba krajní nebo oba vnitřní (možná budete muset trochu experimentovat). Také můžete použít jako výstup piezzo buzzer, pak prostě připojíte každý vodič k jednomu z pinů. Vše popisuje následující obrázek.



Počítejte s tím, že v případě kvalitních reproduktorů může být výstup poměrně hlasitý a nastavte výstup na nižší úroveň.

Přehrávání připravených melodií

MicroPython obsahuje asi dvacet předem připravených melodií, jejichž seznam najdete v dokumentaci. Ukázka použití je v následujícím příkladu:

```
1. from microbit import *
2. import music
3.
4. music.play(music.NYAN)
```

Tento zvuk je poměrně dlouhý, takže budete mít čas správně nastavit reproduktor, nasadit si sluchátka apod.

Všimněte si, že na řádce 2 je nutné zavést knihovnu pro přehrávání hudby. Samotný příkaz pro přehrání melodie je pak na řádce 4.

Seznam všech připravených melodií naleznete v příloze B nebo na stránkách dokumentace MicroPythonu pro micro:bit.

Připravené melodie lze dobře kombinovat s připravenými obrázky, jak ukazuje další příklad:

```
1. from microbit import *
2. import music
3. while True:
4.     if button_a.is_pressed():
5.         display.show(Image.HAPPY)
6.         music.play(music.POWER_UP)
7.     if button_b.is_pressed():
8.         display.show(Image.SAD)
9.         music.play(music.POWER_DOWN)
10. display.clear()
```

Význam jednotlivých příkazů už by vám měl být jasný a proto neuvádíme žádný další popis..

Micro:bit mluví

Microbit umí i mluvit. Naneštěstí pro nás pouze anglicky. Knihovna pro mluvení je navíc zatím označena jako vývojová, takže se můžete potkat s chybami. Mluvení je velmi jednoduché:

```
1. from microbit import *
2. import speech
3.
4. speech.say("Hello", speed=100)
```

Na řádce 2 se zavádí knihovna pro hovor a na řádce 4 je zadán příkaz pro mluvení. Zde micro:bit pozdraví. Parametr `speed=100` je nepovinný. Defaultní hodnota je 72, ale přijde nám, že při této hodnotě mluví micro:bit příliš rychle. Čím vyšší číslo, tím je řeč pomalejší a naopak.

Dokumentace doporučuje zapojit pro hovor sluchátka (repráky) mezi porty 0 a 1 (a ne 0 a GND jako u hudby). A skutečně zvuk je v tomto případě silnější a čistší než mezi 0 a GND.

Přehrání not

Micro:bit dovede přehrát noty. Následující program přehraje melodii „Ovčáci, čtveráci“. Zápis programu trochu připomíná vytváření animovaných obrázků.

```
1. from microbit import *
2. import music
3. nota = ["C4:4", "R:1", "E4:4", "R:1", "F4:4", "R:4", "C4:4",
4.         "R:1", "E4:4", "R:1", "F4:4", "R:4",
5.         "E4:2", "R:1", "E4,2", "R:1", "D4:2", "R:1", "E4:2",
6.         "R:2", "F4:2", "R:1", "D4:2", "R:1",
7.         "E4:2", "R:1", "E4,2", "R:1", "D4:2", "R:1", "E4:2",
8.         "R:2", "F4:2", "R:1", "D4:2", "R:1",
9.         "E4:4", "R:1", "D4:4", "R:1", "C4:4"]
10. music.play(nota)
```

Struktura seznam (list) nota je vlastně zápis jednotlivých tónů. Např. C4:4 znamená nota C ve čtvrté oktávě (0 – nejnižší, 8 – nejvyšší) o délce 4. Nota R znamená pauzu (rest). Příkaz `music.play(nota)` pak daný záznam přehraje.

PŘÍLOHA – SEZNAM PŘIPRAVENÝCH MELODIÍ

- `music.DADADADUM`
- `music.ENTERTAINER`
- `music.PRELUDE`
- `music.ODE`
- `music.NYAN`
- `music.RINGTONE`
- `music.FUNK`
- `music.BLUES`
- `music.BIRTHDAY`
- `music.WEDDING`
- `music.FUNERAL`
- `music.PUNCHLINE`
- `music.PYTHON`
- `music.BADDY`
- `music.CHASE`
- `music.BA_DING`
- `music.WAWAWAWAA`
- `music.JUMP_UP`
- `music.JUMP_DOWN`
- `music.POWER_UP`
- `music.POWER_DOWN`