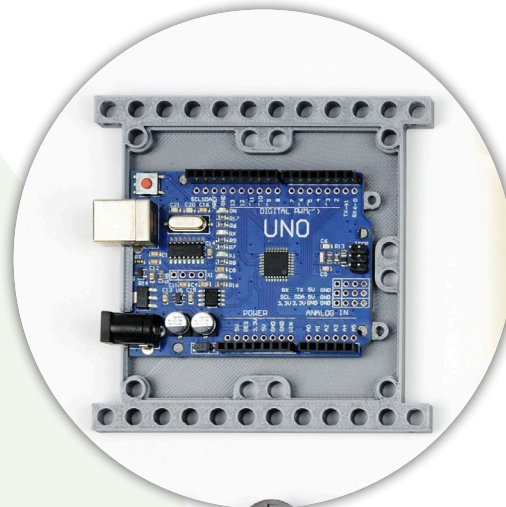
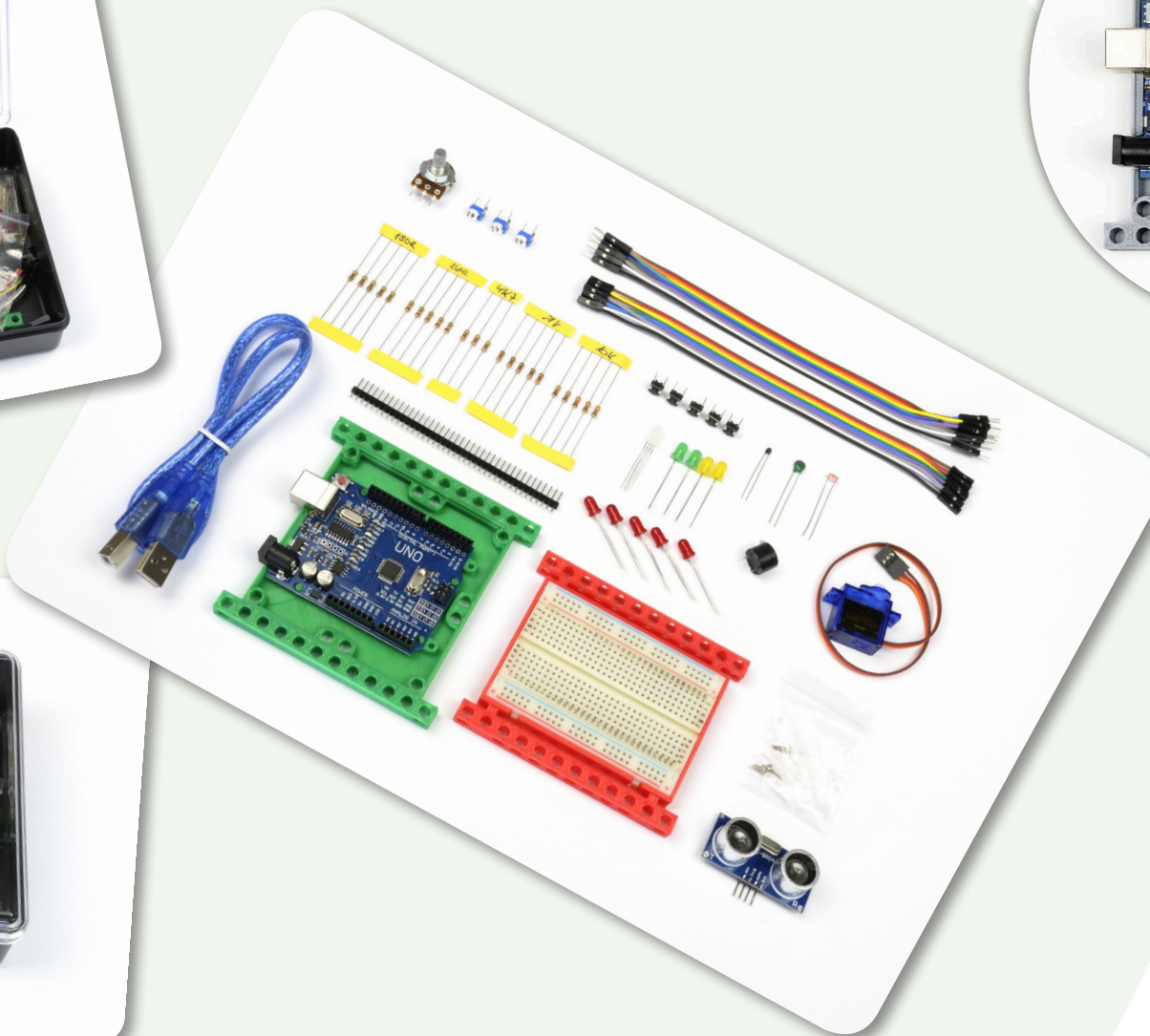


PRO PRÁCI S TĚMITO VÝUKOVÝMI KARTAMI
BUDEŠ POTŘEBOVAT ARDUINO UNO R3
(PŘÍPADNĚ NĚKTERÝ Z JEHO KLONŮ).

ARDUINO - VÝUKOVÉ KARTY VERZE 1.1



časopis o výuce
e-mole.cz
naučen s digitálními technologiemi

Sada e-Mole č. 1
základní výuková a experimentální Arduino sada

Obsah krabičky

Arduino UNO R3 klon + USB kabel	1 ks
Náhradní baterie 400	1 ks
Ultrazvukový senzor (HC-SR04)	1 ks
Servo SG90	1 ks
LED RGB	1 ks
LED R + G + Y	1 ks
Spínač	1 ks
Termistor NTC-1K, NTC-10K	5 + 2 = 7 ks
Pozistor	1 ks
Rezistory 150, 240, 1K, 4,7K, 10K	5 + 8 + 3 = 16 ks
Piezo-Drumák pasivní	1 + 3 ks
Propagační notička	1 ks
Krabičky 3D tiskné	10 + 10 ks
	2 ks

www.e-mole.cz & www.interactivelarning.cz



ZÁKLADNÍ VÝUKOVÁ A EXPERIMENTÁLNÍ ARDUINO SADA

ŽÁK + UČITEL

TENTO PDF SOUBOR NENÍ ZADARMO!

Materiál, který se ti dostal do rukou, vznikl na základě zkušeností z výuky a je součástí **Základní výukové a experimentální Arduino sady** (www.e-mole.cz/diy).

Rádi bychom, aby se tento materiál dostal ke všem, pro které může být něčím přínosný. Nemusí to být pouze učitelé a vedoucí zájmových vzdělávacích kroužků, může to být kdokoli, kdo se snaží seznámit s **Arduinem**...

Soubor PDF můžeš šířit přes internet nebo na libovolném datovém nosiči. Můžeš ho také dát k dispozici ke stažení ze svého serveru nebo webu. Jedinou podmínkou je, že šíření PDF souboru musí být vždy v kompletní podobě a zcela zdarma. Stažení souboru přitom nesmí být podmíněno přihlášením/registrací uživatele.

UŽÍVÁNÍ TOHOTO MATERIÁLU NENÍ ZDARMA, JE TŘEBA UHRADIT POPLATEK...

Návrh a příprava materiálu si vyžádala mnoho hodin práce. Pokud budeš materiál prakticky používat, prosíme, uhrad' za užívání tohoto digitálního materiálu poplatek **69 Kč**. Tvůj finanční příspěvek bude využit pro přípravu dalších výukových materiálů, které pak budou podobným způsobem opět k dispozici všem zájemcům.

Pokud chceš materiál používat jako učební materiál pro své žáky/studenty ve škole nebo v kroužku, prosíme, uhrad' uvedenou částku za každého žáka/studenta.

Pokud jsi ještě dítě a nemůžeš uvedenou částku přes internet uhradit, požádej své rodiče, aby platbu provedli. Pokud z nějakého důvodu nemůžeš platbu zařídit, nic si z toho nedělej. Klidně materiál používej. Věříme, že až budeš starší, částku nám dodatečně uhradíš.

V PŘÍPADĚ, ŽE JSI ZAKOUPIL(A) ZÁKLADNÍ VÝUKOVOU A EXPERIMENTÁLNÍ ARDUINO SADU PŘÍMO OD NÁS, NEMUSÍŠ ZA TENTO VÝUKOVÝ MATERIÁL NIC PLATIT!

Více informací o **Základní výukové a experimentální Arduino sadě** (Sada e-Mole č. 1), včetně informací pro provedení platby, najdeš na:

<http://www.e-mole.cz/diy/vyukova-experimentalni-arduino-sada>

Na stejné adrese najdeš také přehled všech připravených materiálů, včetně rozšiřujících materiálů, které nejsou volně k dispozici a mohou je získat pouze předplatitelé našich výukových materiálů.

- Pokud umístíš soubor ke stažení na svůj web/server, prosíme, napiš nám o tom. Dáme ti vědět, když bude k dispozici nová verze.
- I když je soubor připraven primárně pro zobrazení na displeji, dá se vytisknout na papír formátu A4. Při tisku PDF na papír jiné velikosti nezapomeň nastavit korektní přízpusobené velikosti (např. v tiskovém dialogu Adobe Readeru: Size Options → Fit).
- Prosíme, nezasahuj do PDF souboru. Pokud narazíš na nějakou chybu, napiš nám o ní a my ji opravíme.

e-mole.cz časopis o výuce
nejen s digitálními technologiemi

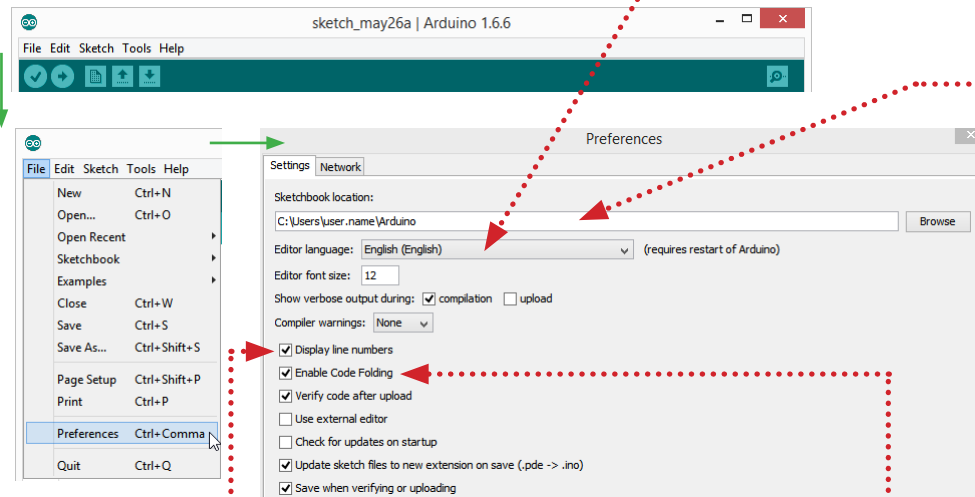
Naším hlavním cílem je ukázat, že digitální technologie nejsou nám učitelům na obtíž, ale že je to přesně naopak, přináší nové možnosti, jak dětem přiblížit problematiku našich oborů a jak z někdy nudné výuky udělat dobrodružnou cestu za poznáním. Na rozdíl od různých školicích akcí a konferencí vidíme v časopisu příležitost přinést zajímavé nápady a ukázky z výuky většímu množství učitelů.



redakce časopisu e-Mole
casopis.e.mole@gmail.com

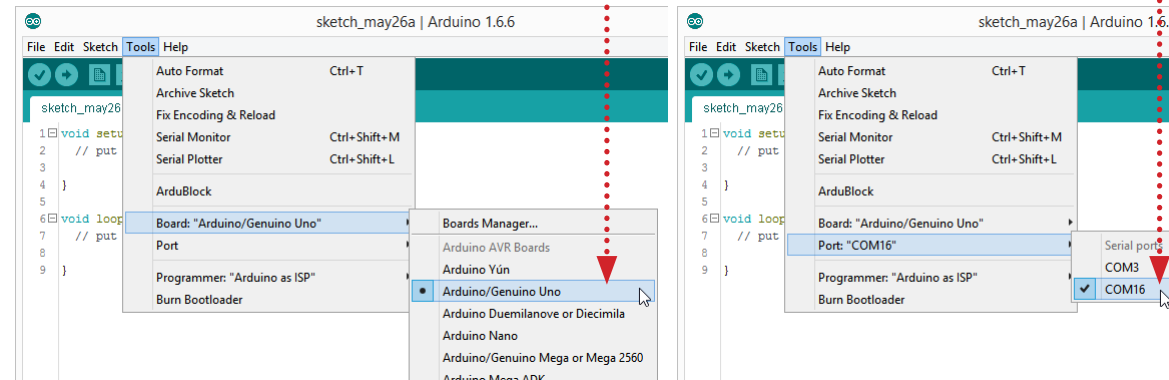
ARDUINO IDE (WWW.ARDUINO.CC) = MULTIPLATFORMNÍ TEXTOVÉ VÝVOJOVÉ PROSTŘEDÍ PRO ARDUINO

1. Po instalaci doporučujeme jako jazyk IDE nastavit angličtinu



2. Vhodné je aktivovat číslování řádků (line numbers) a možnost „sbalení“ kódu (Code Folding)

4. Po připojení Arduina (Arduino klonu) k počítači pomocí USB kabelu je třeba v menu **Tools** vybrat odpovídající typ Arduino **desky** (v našem případě Arduino UNO) a nastavit správný komunikační **port** (číslo portu bude na vašem počítači pravděpodobně jiné).



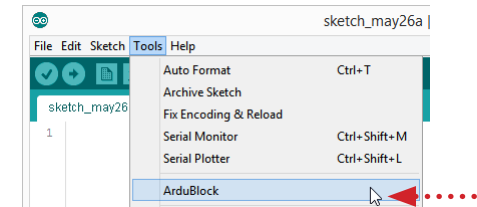
Tip: Pokud se vám po propojení Arduina s počítačem USB kabelem neobjeví nový identifikovaný COM port, je možné, že nebylo vaše Arduino (klon) rozpoznáno automaticky a je nutné doinstalovat odpovídající ovladače. Ovladače k Arduino klonu v naší sadě můžete stáhnout z našich stránek (www.e-mole.cz/system/files/arduino/ch341ser.zip).

3. Doinstalování vizuálního rozhraní **ArduBlock** (<https://sourceforge.net/projects/ardublock>) provedete zkopírováním JAR souboru do příslušného adresáře: Soubor si můžete stáhnout také z našich stránek (www.e-mole.cz/system/files/arduino/ardublock.zip).

a) v adresáři (*Sketchbook location*) vytvořte podadresáře **/tools/ArduBlockTool/tool**, b) do posledního uvedeného nakopírujte JAR soubor (**ardublock-beta-20140828.jar**), c) po spuštění Arduino IDE se v menu **Tools** objeví nová položka – **ArduBlock**.

Více informací o ArduBlocku naleznete také na www.ardublock.com.

Tip: Pokud chcete ArduBlock instalovat pro všechny uživatele určitého počítače a přitom zachovat individuální cestu *Sketchbook location*, vytvořte adresáře z bodu a) přímo v instalačním adresáři Arduino IDE.



5. Samotnou kompilaci kódu můžeme provést stisknutím tlačítka se **zatržítkem**. Kompilaci kódu a jeho následné nahrání do Arduina provedeme pomocí tlačítka s **šipkou**.

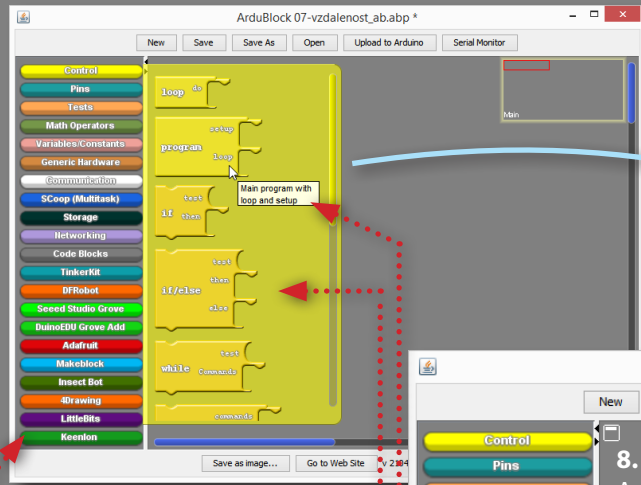


Informace o úspěšném nahrání zkompilevaného kódu je ve spodní části okna (v zeleném pruhu).

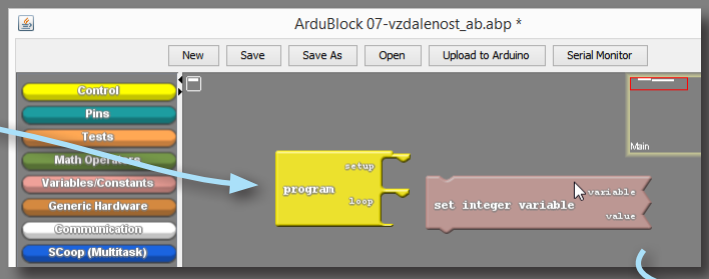
POKUD BUDEŠ TYTO MATERIÁLY POUŽÍVAT, UHRAD ČÁSTKU 69 Kč. Více najdeš na straně č. 1 nebo na www.e-mole.cz/diy/vyukova-experimentalni-arduino-sada



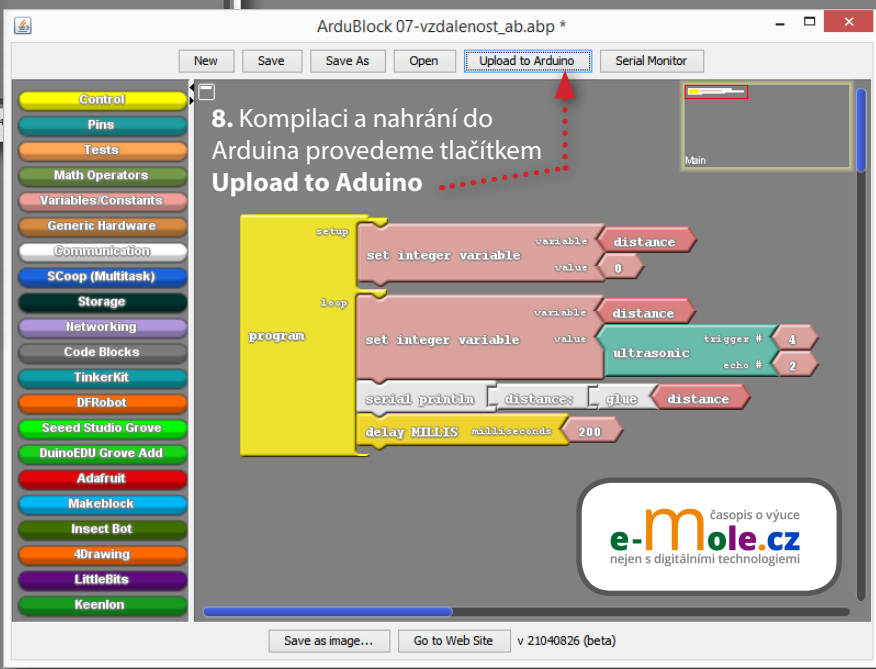
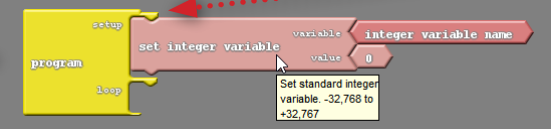
ARDUBLOCK (WWW.ARDUBLOCK.COM) = GRAFICKÉ VÝVOJOVÉ PROSTŘEDÍ PRO ARDUINO



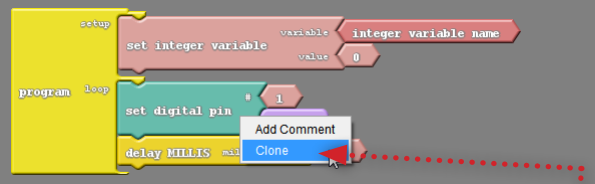
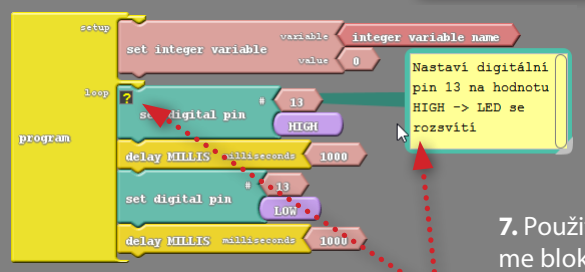
1. Z levého panelu rozvineme nabídku jednotlivých **bloků** – kliknutím na odpovídající sekci (jednotlivá barevná tlačítka).
2. Pokud se nad blokem zastavíme, zobrazí se **nápovědný text** s popisem typického použití bloku.
3. Vybraný blok uchopíme myší (LT) a přesuneme nad šedou plochu vpravo.



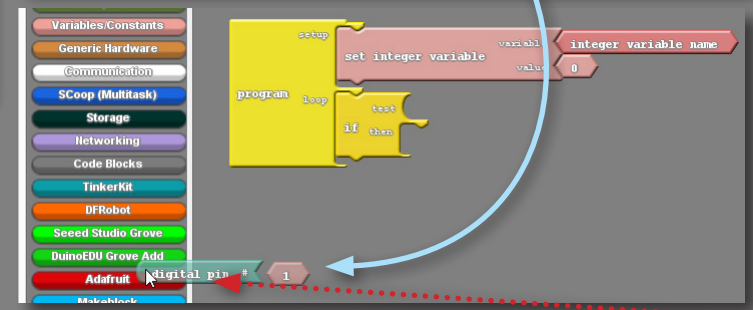
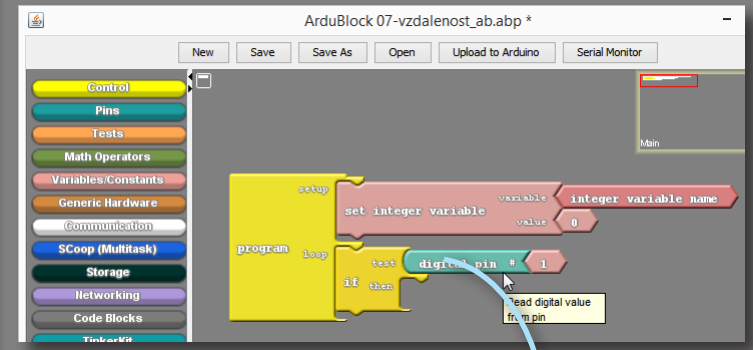
4. Bloky mají schopnost „zavaknout“ do sebe. Blok, který držíme myší, přesuneme na odpovídající místo a tlačítko myši uvolníme. Pokud si oblasti „tvarově“ odpovídají, ozve se zvuk „cvaknutí“ a uvolněný blok se **přímkne** ke spodnímu bloku, nebo se včlení mezi stávající bloky. I v okně s poskládaným kódem se nám po najetí myší nad blok zobrazí nápovědný text.



8. Kompilaci a nahrání do Arduina provedeme tlačítkem **Upload to Arduino**



7. Použitím pravého tlačítka myši nad libovolným blokem můžeme blok (a současně bloky následující) **kopírovat**. Další užitečnou funkcí je přidávání textových **komentářů**.

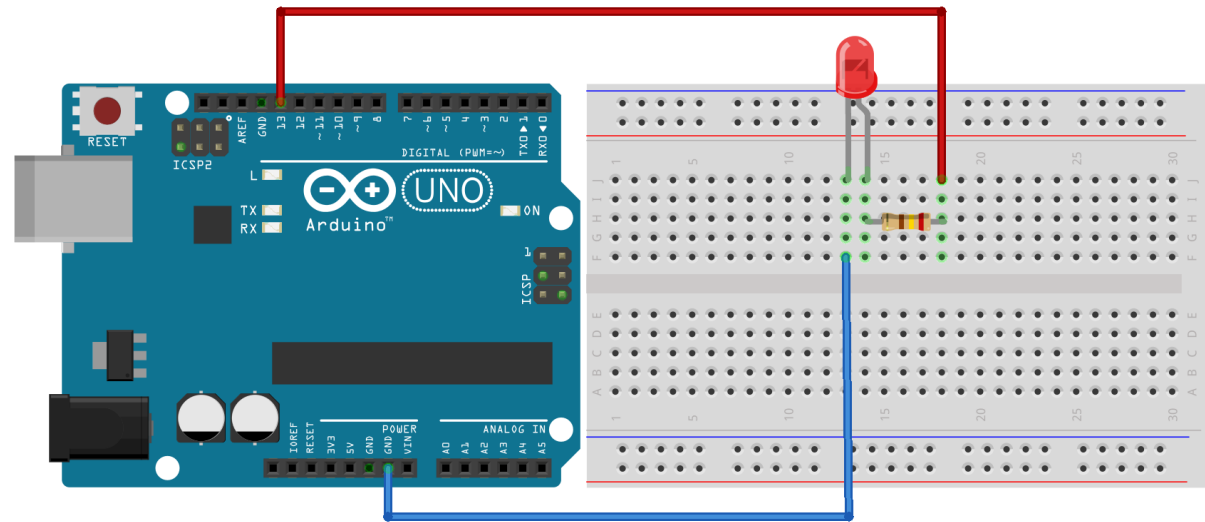


6. Uvolněný blok si můžeme nechat ležet na ploše pro pozdější použití. Úplné smazání bloku provedeme přesunem a **puštěním** bloku nad levým panelem.

POKUD BUDEŠ TYTO MATERIÁLY POUŽÍVAT, UHRAĎ ČÁSTKU 69 Kč. Více najdeš na straně č. 1 nebo na www.e-mole.cz/diy/vyukova-experimentalni-arduino-sada

01 Blikání LED

Dioda připojená anodou (delší nožka) přes rezistor 240 Ω k digitálnímu pinu 13 vždy půl sekundy svítí a půl sekundy je zhasnutá (stejnou funkci – diody připojené k pinu 13 – má malá LED integrovaná přímo na desce Arduino UNO).

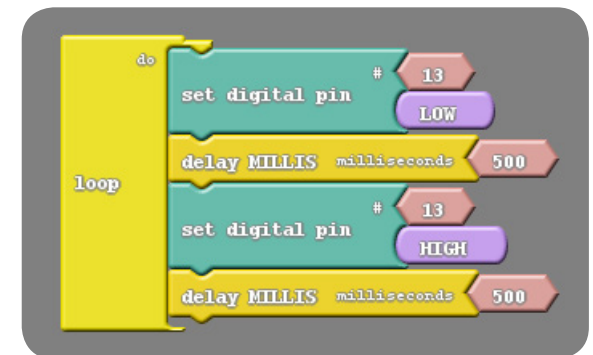


```
void setup()
{
  pinMode( 13 , OUTPUT);
}

void loop()
{
  digitalWrite( 13 , LOW );
  delay( 500 );
  digitalWrite( 13 , HIGH );
  delay( 500 );
}
```

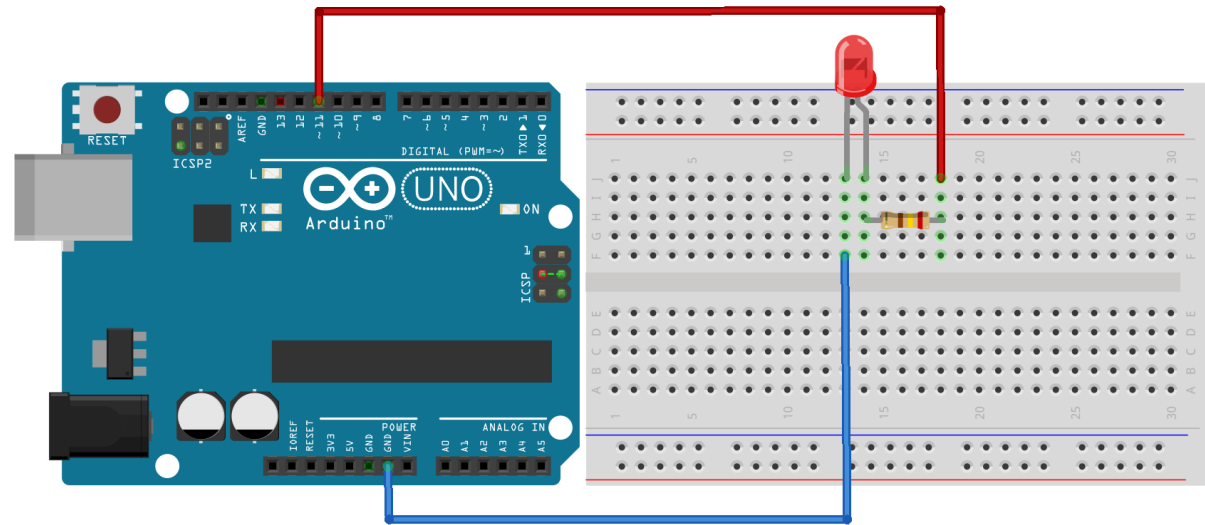
Tip:

- 1) Zkuste měnit rychlost blikání změnou časové prodlevy (delay).
- 2) Zapojte stejným způsobem dvě diody ke dvěma různým pinům a naprogramujte blikáč (střídavě svítí jedna nebo druhá dioda).



02 Rozsvěcení LED

Dioda je připojena k pinu 11, který díky technologii PWM (Pulse-width modulation) umožňuje nastavovat výstupní výkon pinu v 256 krocích (takto vybavené digitální piny jsou na desce označeny vlnovkou). Jednoduchý cyklus s postupně se zvětšující úrovní výkonu pinu tak způsobuje rozsvěcení LED.



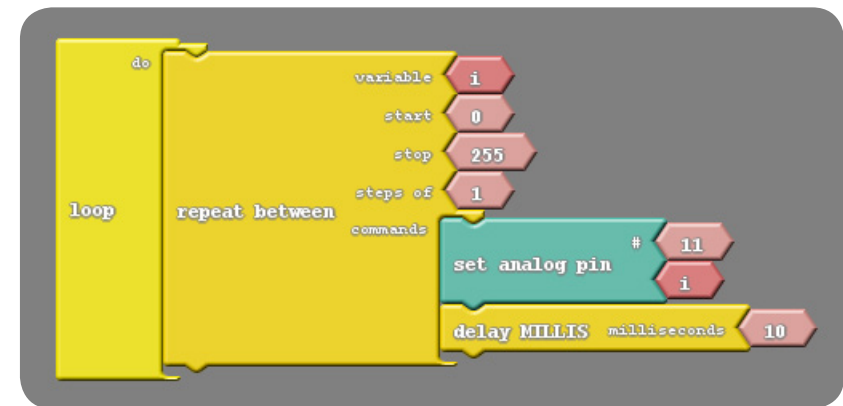
```
int _ABVAR_1_i = 0 ;

void setup()
{
  pinMode( 11 , OUTPUT);
}

void loop()
{
  for( _ABVAR_1_i = 0;
  0<=255?_ABVAR_1_i <= 255:_ABVAR_1_i >= 255;
  0<=255?_ABVAR_1_i = _ABVAR_1_i + 1:_
  ABVAR_1_i = _ABVAR_1_i - 1)
  {
    analogWrite(11 , _ABVAR_1_i);
    delay( 10 );
  }
}
```

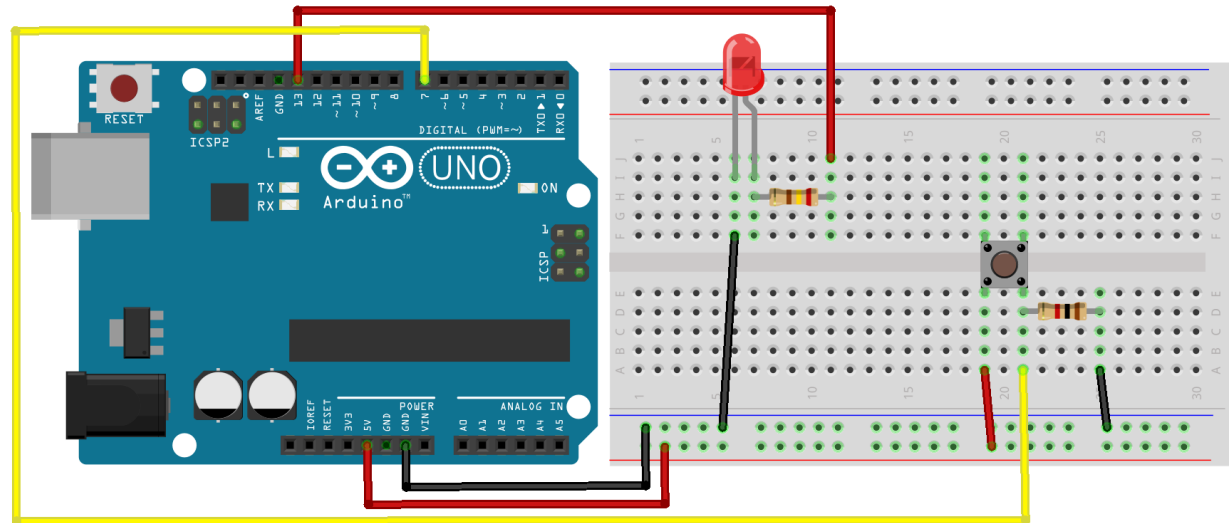
Tip:

Zkuste měnit rychlost rozsvěcení a dále naprogramovat postupné rozsvěcení a následné zhašení LED.



03 LED a tlačítko

Zapojení představuje obecné použití tlačítka, po jehož stisknutí je na digitálním pinu 7 zaznamenána logická 1 (hodnota high), na kterou rozhodovací podmínka zareaguje rozsvícením diody (obecně můžeme takto spustit jakýkoli proces). Při vypnutém tlačítku je na pinu 7 logická 0 a dioda podle zadané podmínky nesvítí.

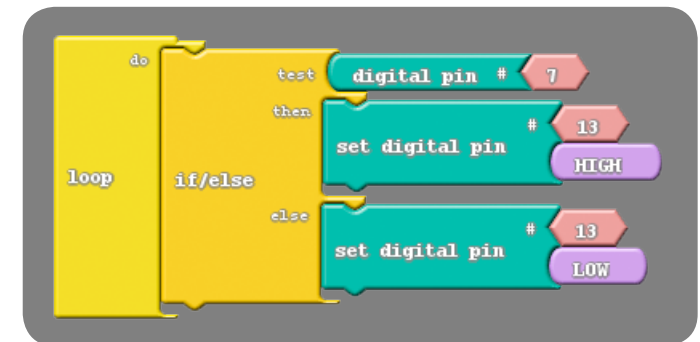


```
void setup()
{
  pinMode( 7 , INPUT);
  pinMode( 13 , OUTPUT);
}

void loop()
{
  if (digitalRead(7))
  {
    digitalWrite( 13 , HIGH );
  }
  else
  {
    digitalWrite( 13 , LOW );
  }
}
```

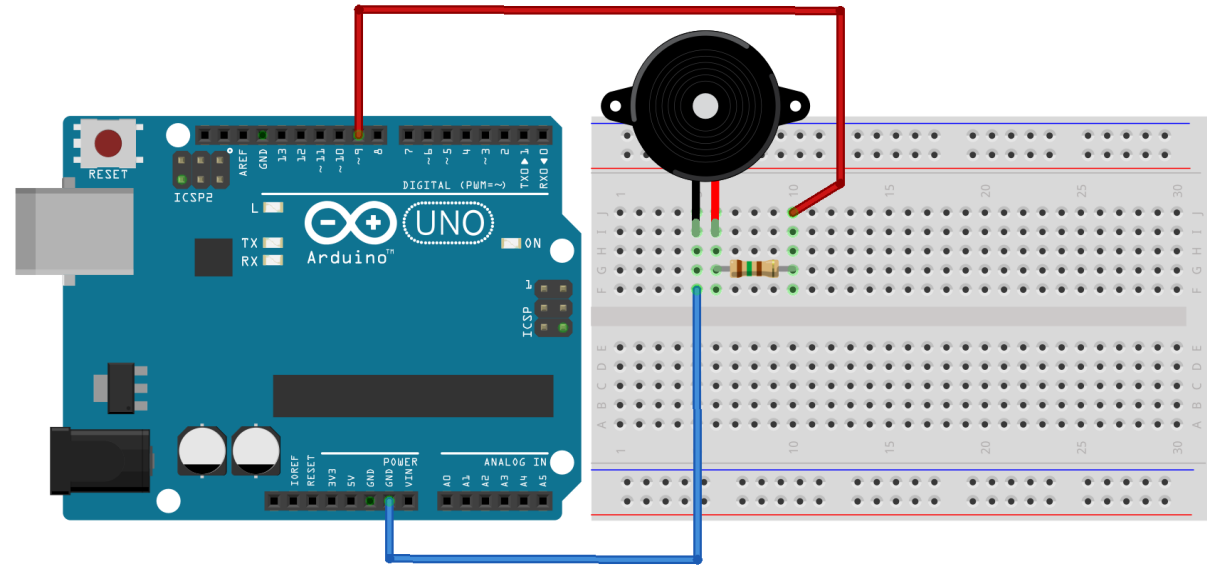
Tip:

Zkuste zapojit tlačítko sériově přímo mezi jeden z přírodních vodičů k diodě (červený nebo černý) a ověřit, že pro pouhé zapínání a vypínání jedné LED by stačilo takové jednoduché zapojení jako v klasickém analogovém obvodu (v programu musí být přítom samozřejmě na pinu 13 nastavena logická 1 – parametr „HIGH“).

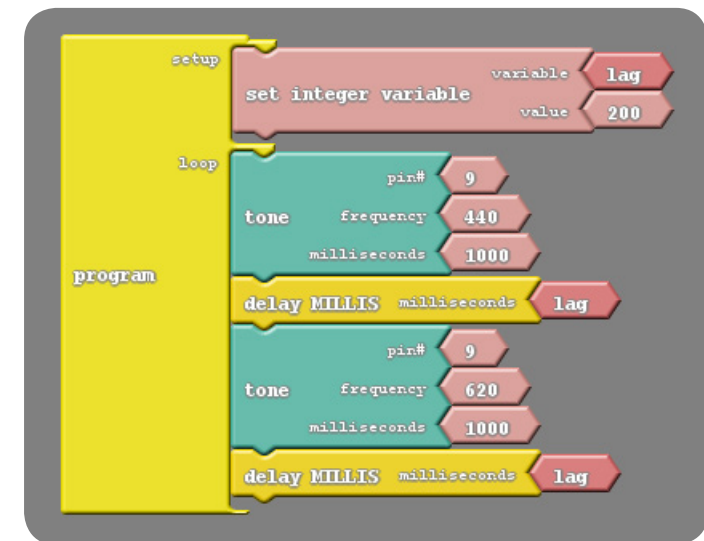


04 Tónový generátor

Jednoduchý piezoelektrický bzučák má označený kladný vstup (červený vodič), který je připojen přes rezistor 150 Ω k pinu 9. Velikost odporu rezistoru reguluje hlasitost bzučáku. V uvedeném kódu je použita proměnná „prodleva“ pro dobu mezi střídáním dvou odlišných tónů.



```
int _ABVAR_1_lag = 0 ;
void setup()
{
  _ABVAR_1_lag = 200 ;
}
void loop()
{
  tone(9, 440, 1000);
  delay( _ABVAR_1_lag );
  tone(9, 620, 1000);
  delay( _ABVAR_1_lag );
}
```

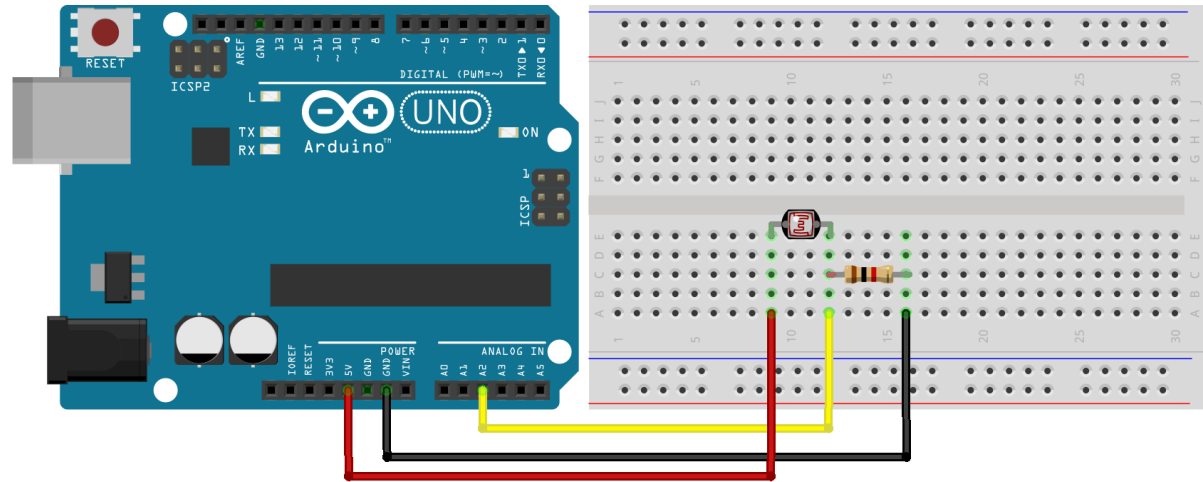


Tip:

- 1) Pokuste se vytvořit kód pro zahrání nějakého hudebního intervalu (podle příslušného poměru frekvencí), nebo jednoduché melodie.
- 2) Použijte několik proměnných pro vytvoření různě dlouhých tónů.

05 Detekce osvětlení

Fotorezistor je propojen sériově s rezistorem 1 kΩ (možno vyzkoušet i jiné hodnoty – například 10 kΩ) k napájecímu (5V) a zemnicímu pinu (GND). Na analogovém pinu A2 je pak vyčítána hodnota napětí na rezistoru v hodnotách 0 - 1023, které odpovídají lineárně napětí 0 V – 5 V (hodnota je dána poměrem odporu rezistoru a proměnného odporu fotorezistoru). Vyčtené hodnoty na A2 se zobrazují na sériovém monitoru, který lze zapnout v editoru Arduino IDE (ikonka v pravém horním rohu) nebo v okně Ardublock.



Tip:

Sledujte, jak se mění hodnota na Sériovém monitoru při různé míře zastínění fotorezistoru. Vyzkoušejte, co se změní, pokud prohodíme konce napájecího a zemnicího vodiče (hodnota A2 nyní určuje velikost napětí na fotorezistoru).

```
int _ABVAR_1_lag = 0 ;
int _ABVAR_2_my_A2_value = 0 ;

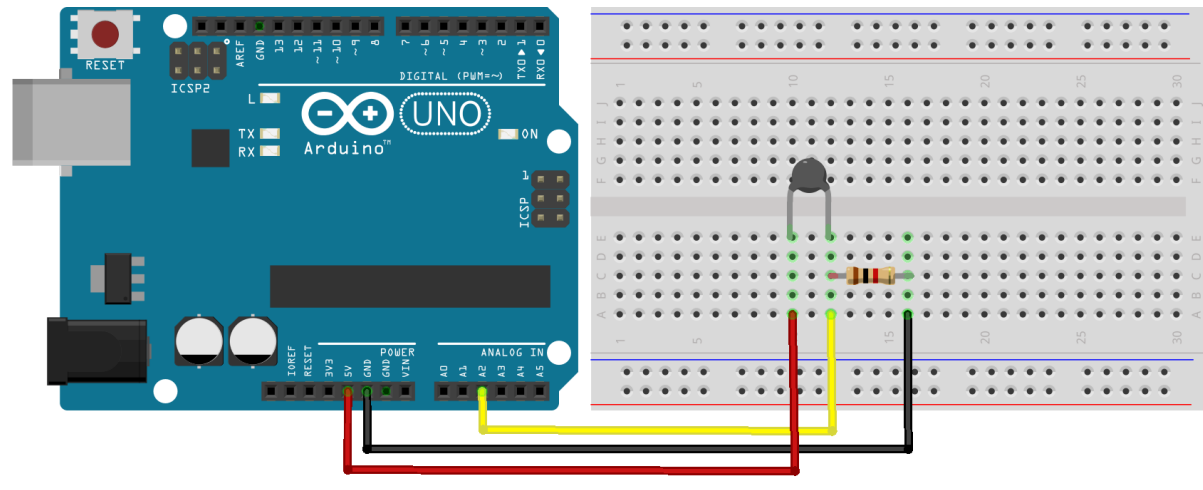
void setup()
{
  Serial.begin(9600);
  _ABVAR_1_lag = 500 ;
}

void loop()
{
  _ABVAR_2_my_A2_value = analogRead(2) ;
  Serial.print(„my value: “);
  Serial.print(_ABVAR_2_my_A2_value);
  Serial.println();
  delay( _ABVAR_1_lag );
}
```

POKUD BUDEŠ TYTO MATERIÁLY POUŽÍVAT, UHRAĎ ČÁSTKU 69 Kč. Více najdeš na straně č. 1 nebo na www.e-mole.cz/diy/vyukova-experimentalni-arduino-sada

06 Zapojení termistoru

Zaměníme-li fotorezistor z předchozího příkladu (Detekce osvětlení) za termistor, můžeme pomocí zcela stejného kódu detekovat hodnotu na pinu A2, která se mění se změnou teploty (a tím i odporu) termistoru. Změnou velikosti odporu rezistoru nebo použitím jiného termistoru dosáhneme různé citlivosti v různých teplotních intervalech.



```
int _ABVAR_1_lag = 0 ;
int _ABVAR_2_my_A2_value = 0 ;

void setup()
{
  Serial.begin(9600);
  _ABVAR_1_lag = 500 ;
}

void loop()
{
  _ABVAR_2_my_A2_value = analogRead(2) ;
  Serial.print(„my value: “);
  Serial.print(_ABVAR_2_my_A2_value);
  Serial.println();
  delay(_ABVAR_1_lag );
}
```

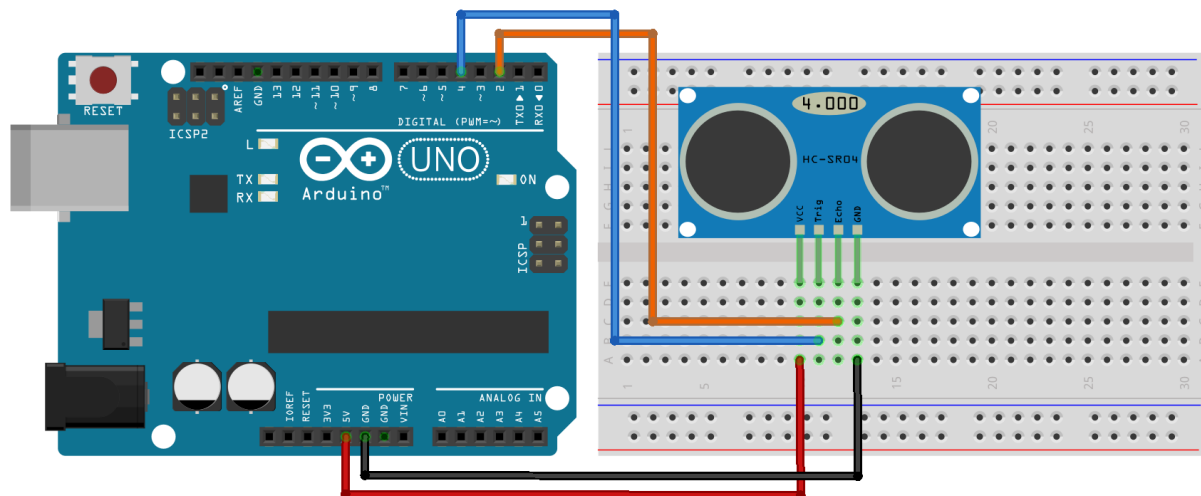


Tip:

Můžete se pokusit zaznamenat hodnotu na A2 při různých teplotách termistoru (například prodloužením přívodů termistoru pomocí kablíčků a zahříváním vzduchu v uzavřené nádobě s termistorem a teploměrem), vytvořit si graf závislosti hodnoty A2 na teplotě (kalibrační křivku) a získat tak jednoduchý teploměr.

07 Měření vzdálenosti

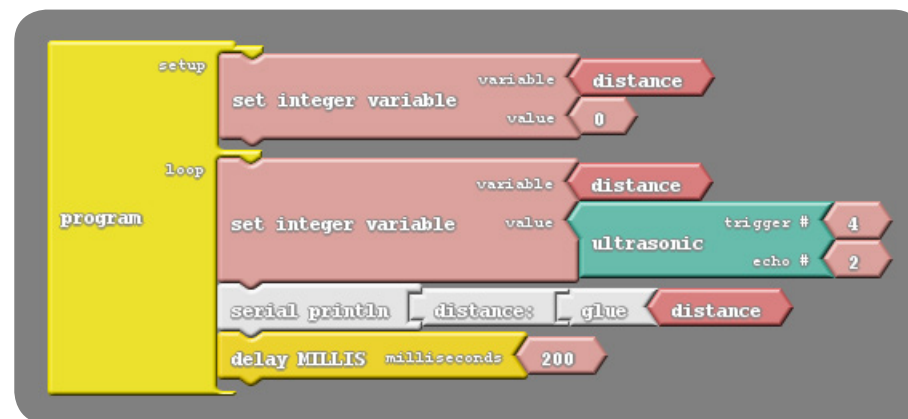
Ultrazvukový senzor polohy má čtyři vývody. Kromě napájení VCC a uzemnění GND je to Trig, který je připojen k pinu řídicímu vysílání krátkého ultrazvukového pulsu, a Echo vracející informaci o přijetí tohoto pulsu po odrazu od překážky. V programu použitý blok „ultrasonic“ vrací přímo hodnotu vzdálenosti předmětu v centimetrech. Tato hodnota je potom zobrazována v okně sériového monitoru.



```
int _ABVAR_1_distance = 0 ;
int ardublockUltrasonicSensorCodeAutoGeneratedReturnCM(int trigPin, int echoPin)
{
    long duration;
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(20);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    duration = duration / 59;
    if ((duration < 2) || (duration > 300))
        return false;
    return duration;
}

void setup()
{
    digitalWrite( 4 , LOW );
    Serial.begin(9600);
    _ABVAR_1_distance = 0 ;
}

void loop()
{
    _ABVAR_1_distance = ardublockUltrasonicSensorCodeAutoGeneratedReturnCM( 4 , 2 ) ;
    Serial.print („distance: “);
    Serial.print(_ABVAR_1_distance);
    Serial.println();
    delay( 200 );
}
```



Tip:

Ve vygenerovaném kódu programu Arduino IDE je vidět konstanta 59, kterou je vydělen čas od vyslání po návrat ultrazvukového pulsu v mikrosekundách. Pokuste se odvodit, jaký je význam této konstanty, a jak bychom ji měli upravit, kdybychom potřebovali například kvůli změně teploty uvažovat jinou rychlost zvuku.

POKUD BUDEŠ TYTO MATERIÁLY POUŽÍVAT, UHRAĎ ČÁSTKU 69 Kč. Více najdeš na straně č. 1 nebo na www.e-mole.cz/diy/vyukova-experimentalni-arduino-sada

08 Servo

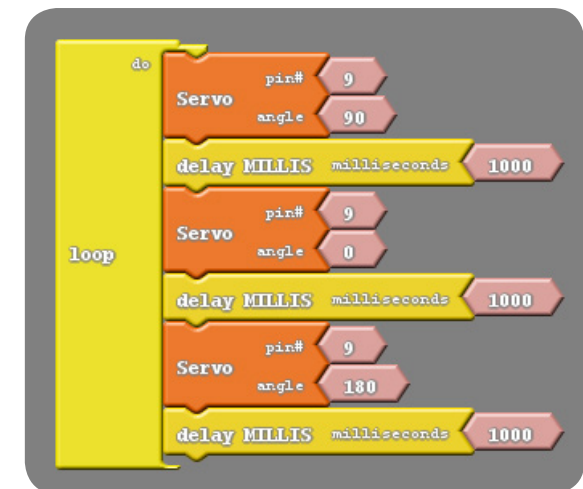
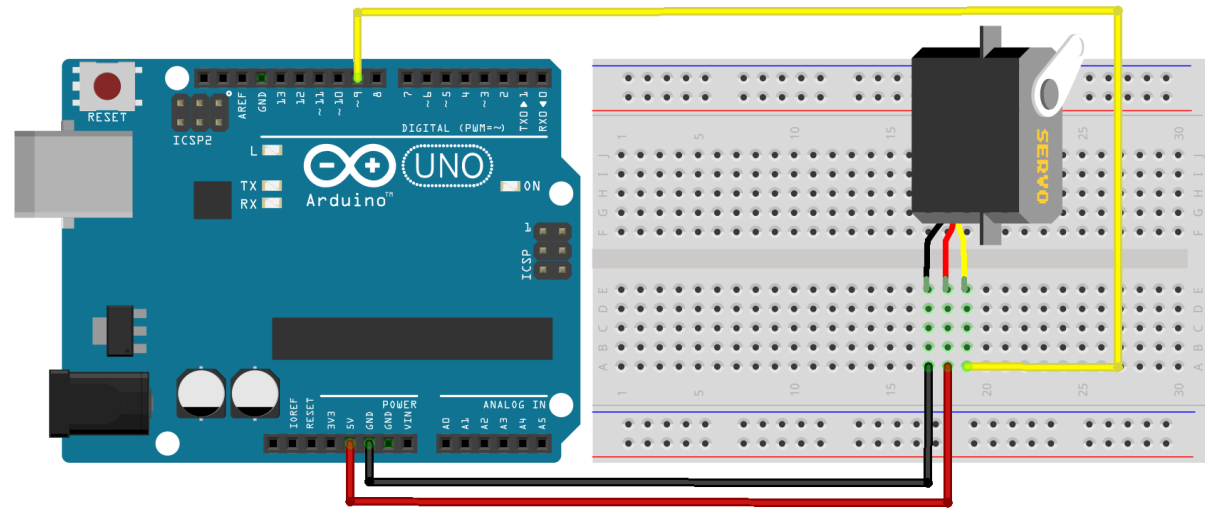
Servomotor, zkráceně servo, je speciální motor, který umožňuje přesné nastavení úhlu (polohy natočení osy). Naše servo je možné nastavit v rozsahu 0-180°. Barevné označení připojovacích kabelů od serva se může lišit od vyobrazení ve schématu. Často se používá kombinace červeného kabelu pro napájení, hnědého pro uzemnění a oranžového pro signál – v našem případě připojeného k pinu 9. Znázorněný program natočí osu serva postupně na úhel 90°, následně na 0°, nakonec na 180° a poté celý cyklus opakuje znovu.

```
#include <Servo.h>

Servo servo_pin_9;

void setup()
{
  servo_pin_9.attach(9);
}

void loop()
{
  servo_pin_9.write( 90 );
  delay( 1000 );
  servo_pin_9.write( 0 );
  delay( 1000 );
  servo_pin_9.write( 180 );
  delay( 1000 );
}
```

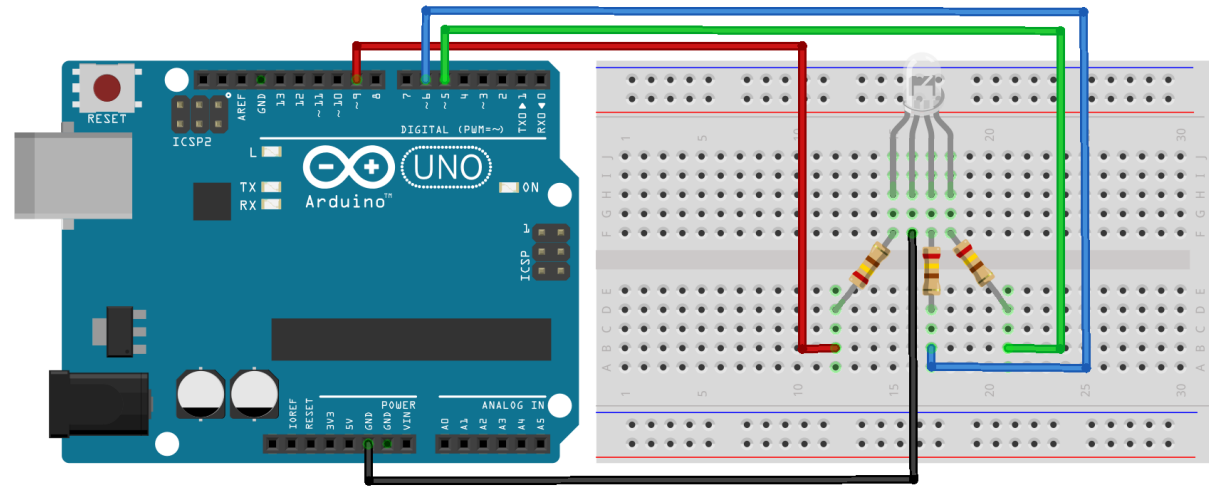


Tip:

Zkuste naprogramovat servo tak, aby se pomalu otáčelo od 0° do 120° a potom rychleji zpět. Můžete využít například podobný program jako u rozsvícení LED.

09 RGB LED

Při zapojení RGB LED se společnou katodou je potřeba katodu (bývá to nejdelší nožička) připojit k zemnímu pinu GND, ostatní nožičky odpovídají anodám jednotlivých barev uspořádaných podle typu diody (lze nalézt v technické dokumentaci, nebo jednoduše vyzkoušet). Změnou hodnoty (0 – 255) na pinech jednotlivých anod lze měnit výslednou barvu LED.



```
int _ABVAR_1_red = 0 ;
int _ABVAR_2_green = 0 ;
int _ABVAR_3_blue = 0 ;

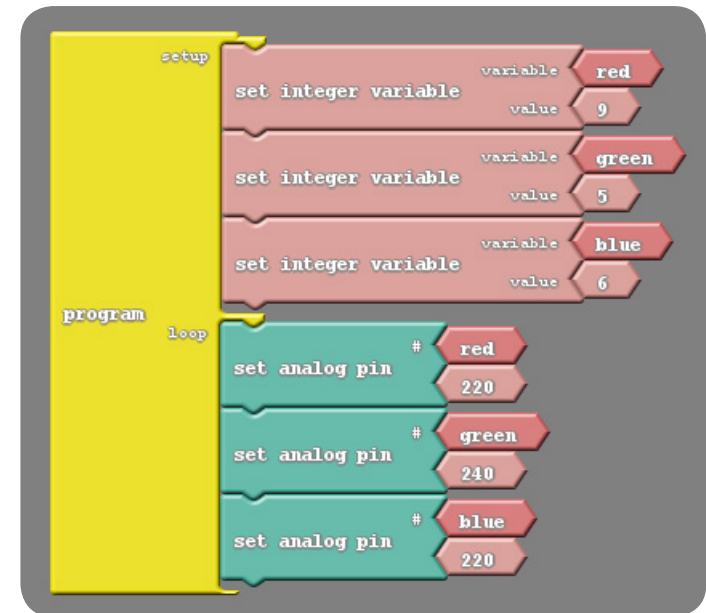
void setup()
{
  pinMode( _ABVAR_1_red , OUTPUT);
  pinMode( _ABVAR_2_green , OUTPUT);
  pinMode( _ABVAR_3_blue , OUTPUT);

  _ABVAR_1_red = 9 ;
  _ABVAR_2_green = 5 ;
  _ABVAR_3_blue = 6 ;
}

void loop()
{
  analogWrite( _ABVAR_1_red , 220);
  analogWrite( _ABVAR_2_green , 240);
  analogWrite( _ABVAR_3_blue , 220);
}
```

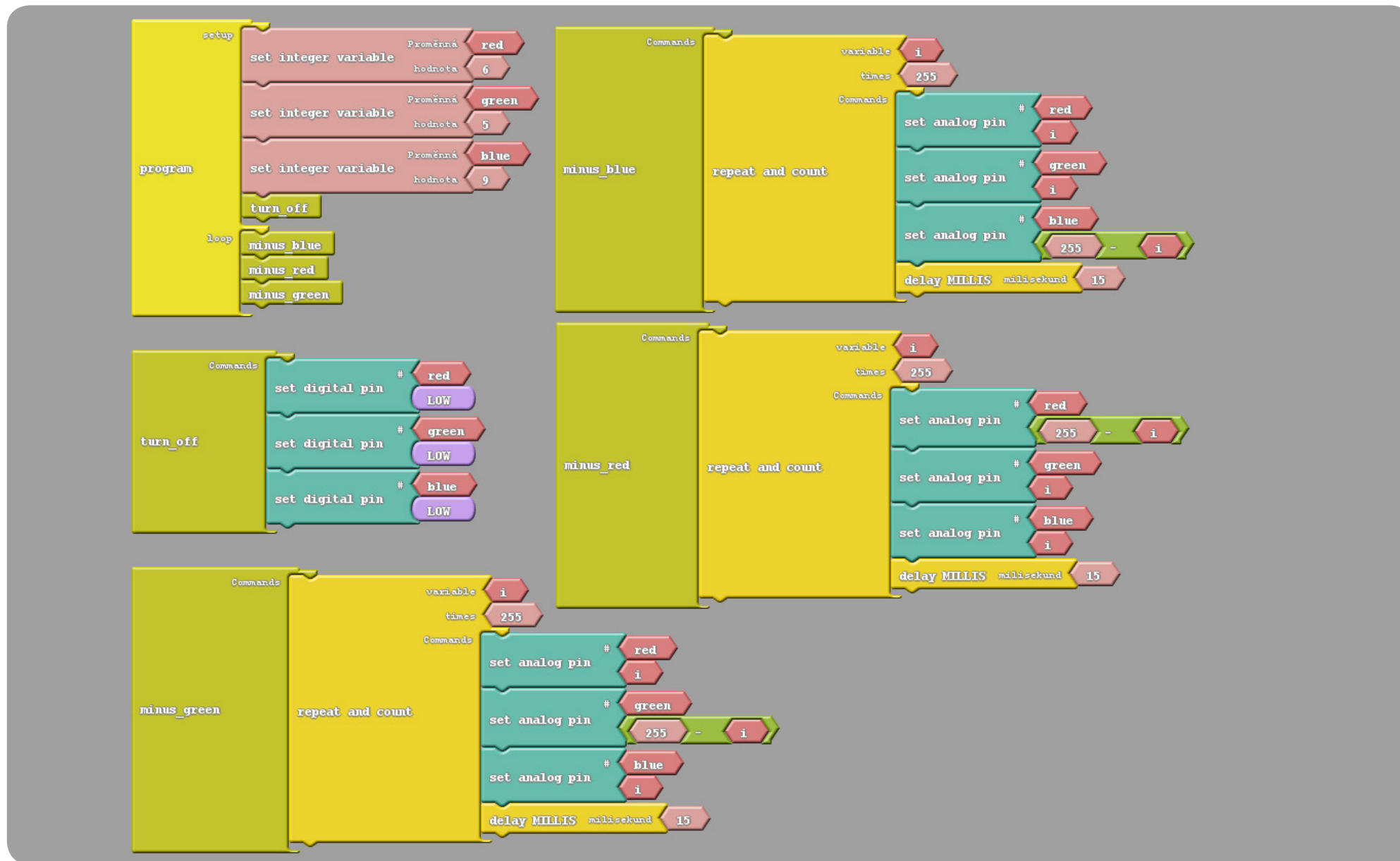
Tip:

- 1) Pro lepší míchání jednotlivých barev ve výslednou je vhodné nasadit na RGB LED ping-pongový míček s vyvrtaným otvorem o průměru 5 mm.
- 2) Pozorujte, jaké barvy získáte, pokud necháte hodnotu vždy jedné proměnné nulovou a budete nastavovat pouze zbylé dvě.



10 RGB LED – změna barev

Následující program bude při stejném zapojení jako v předchozím příkladu opakovaně měnit výslednou barvu postupným zeslabováním vždy jedné ze tří základních barev.



POKUD BUDEŠ TYTO MATERIÁLY POUŽÍVAT, UHRAĎ ČÁSTKU 69 Kč. Více najdeš na straně č. 1 nebo na www.e-mole.cz/diy/vyukova-experimentalni-arduino-sada

10 RGB LED – změna barev (VLASTNÍ KÓD GENEROVANÝ ARDUBLOCKEM)

```

int _ABVAR_1_red = 0 ;
int _ABVAR_2_green = 0 ;
int _ABVAR_3_blue = 0 ;

void __ardublockDigitalWrite(int pinNumber, boolean status)
{
    pinMode(pinNumber, OUTPUT);
    digitalWrite(pinNumber, status);
}

int _ABVAR_4_i = 0 ;

void minus_red();
void turn_off();
void minus_green();
void minus_blue();

void setup()
{
    pinMode( _ABVAR_1_red , OUTPUT);
    pinMode( _ABVAR_2_green , OUTPUT);
    pinMode( _ABVAR_3_blue , OUTPUT);

    _ABVAR_1_red = 6 ;

    _ABVAR_2_green = 5 ;

    _ABVAR_3_blue = 9 ;

    turn_off();
}

void loop()
{
    minus_blue();
    minus_red();
    minus_green();
}

```

```

void turn_off()
{
    __ardublockDigitalWrite( _ABVAR_1_red, LOW);
    __ardublockDigitalWrite( _ABVAR_2_green, LOW);
    __ardublockDigitalWrite( _ABVAR_3_blue, LOW);
}

void minus_red()
{
    for ( _ABVAR_4_i= 1; _ABVAR_4_i<= ( 255 ); _ABVAR_4_i++ )
    {
        analogWrite( _ABVAR_1_red , ( 255 - _ABVAR_4_i ));
        analogWrite( _ABVAR_2_green , _ABVAR_4_i);
        analogWrite( _ABVAR_3_blue , _ABVAR_4_i);
        delay( 15 );
    }
}

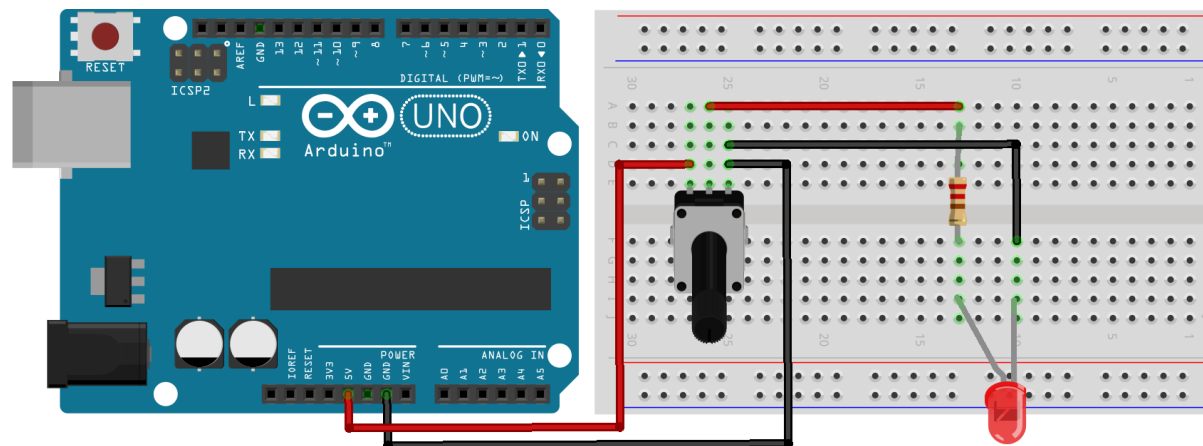
void minus_green()
{
    for ( _ABVAR_4_i= 1; _ABVAR_4_i<= ( 255 ); _ABVAR_4_i++ )
    {
        analogWrite( _ABVAR_1_red , _ABVAR_4_i);
        analogWrite( _ABVAR_2_green , ( 255 - _ABVAR_4_i ));
        analogWrite( _ABVAR_3_blue , _ABVAR_4_i);
        delay( 15 );
    }
}

void minus_blue()
{
    for ( _ABVAR_4_i= 1; _ABVAR_4_i<= ( 255 ); _ABVAR_4_i++ )
    {
        analogWrite( _ABVAR_1_red , _ABVAR_4_i);
        analogWrite( _ABVAR_2_green , _ABVAR_4_i);
        analogWrite( _ABVAR_3_blue , ( 255 - _ABVAR_4_i ));
        delay( 15 );
    }
}

```

11 Řízení jasu LED potenciometrem

Znázorněné schéma představuje klasické analogové zapojení potenciometru připojeného ke zdroji napětí 5V. K ovládání LED není potřeba žádný kód, stačí, aby byla deska Arduino připojena ke zdroji napětí. Proč se dioda začne v tomto případě rozsvěcet až zhruba od poloviny rozsahu potenciometru?

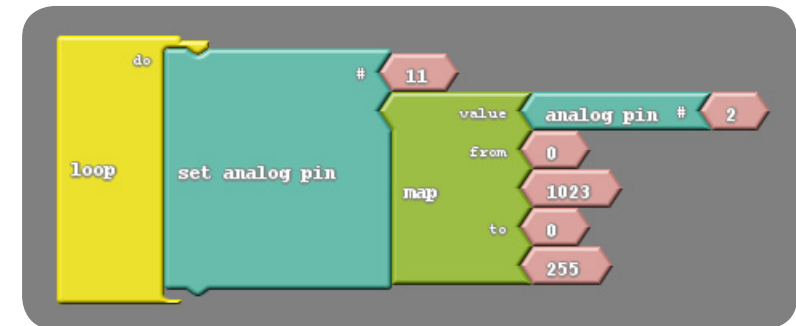
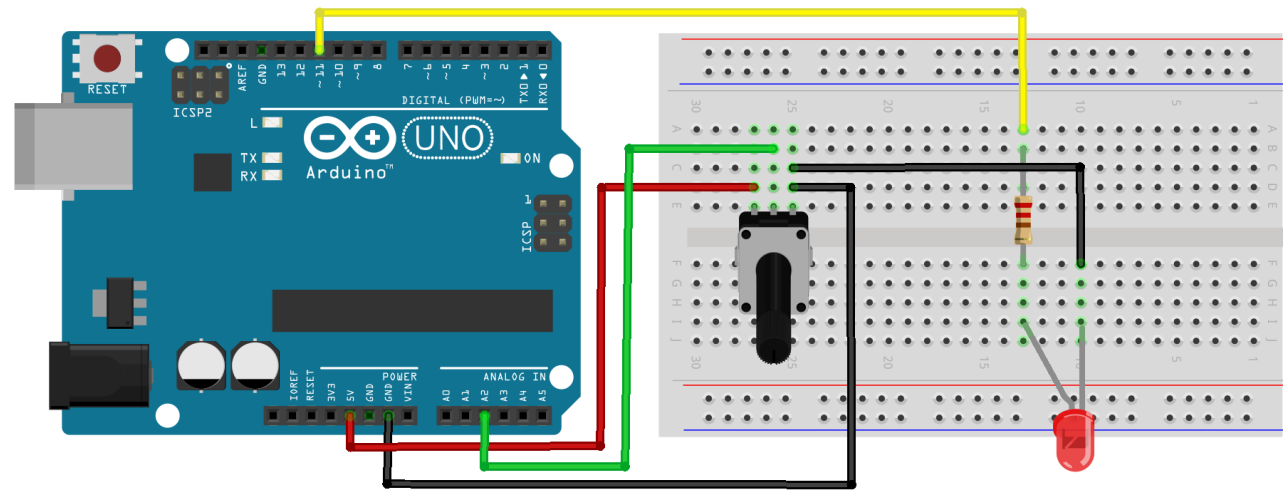


Tip:

Zapojte obdobným způsobem RGB diodu tak, aby každá anoda byla řízena jedním potenciometrem. Otáčením jednotlivých potenciometrů můžete analogově nastavovat výslednou barvu diody. Jaké jsou výhody a nevýhody tohoto řešení v porovnání se zapojením 09?

12 Řízení jasu LED potenciometrem II

V tomto zapojení vyčítáme pomocí analogového pinu A2 hodnotu napětí na jezdcí potenciometru a podle ní řídíme hodnotu digitálního pinu 11, který napájí diodu prostřednictvím technologie PWM. Program má za úkol pouze převést hodnotu z A2 (0 – 1023) na hodnotu na digitálním výstupu 11 (0 – 255). Na rozdíl od předchozího příkladu tak řídíme jas LED v celém rozsahu otáčení potenciometru.



```
void setup()
{
  pinMode( 11 , OUTPUT);
}

void loop()
{
  analogWrite(11 , map ( analogRead(2) , 0 ,
  1023 , 0 , 255 ) );
}
```

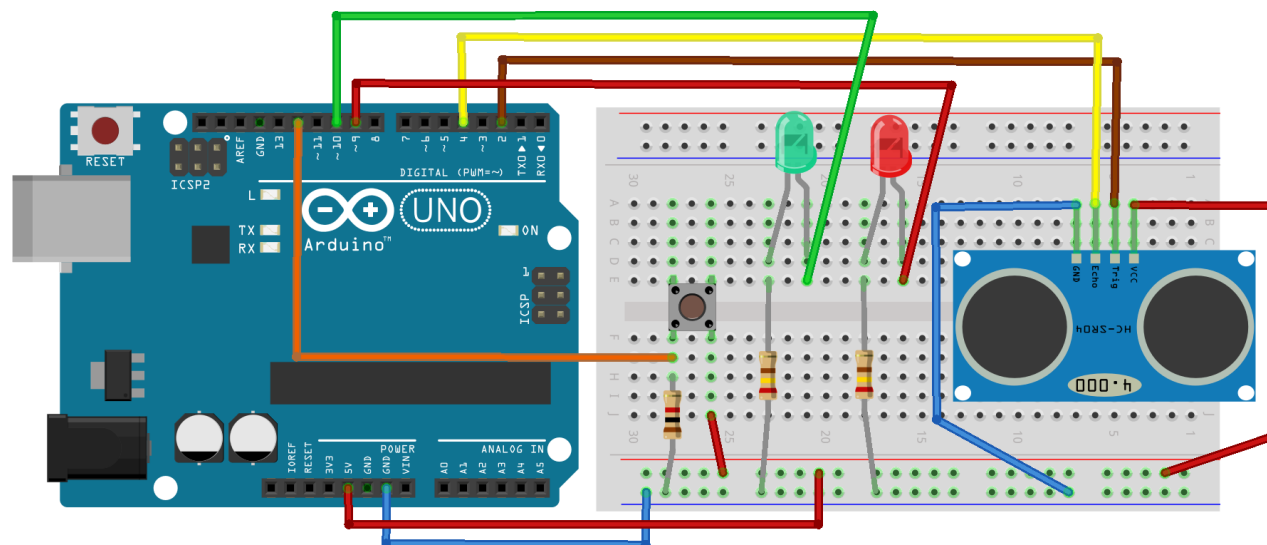
Tip:

Pokuste se naprogramovat blikající LED, jejíž rychlost blikání bude nastavována polohou potenciometru.

13 Odhad vzdálenosti

Odhadnete správně vzdálenost?

Po zapojení a naprogramování je funkce následující: Když stisknu tlačítko a budu vzdálen(a) 15 – 20 cm, rozsvítí se zelená LED, jinak červená.

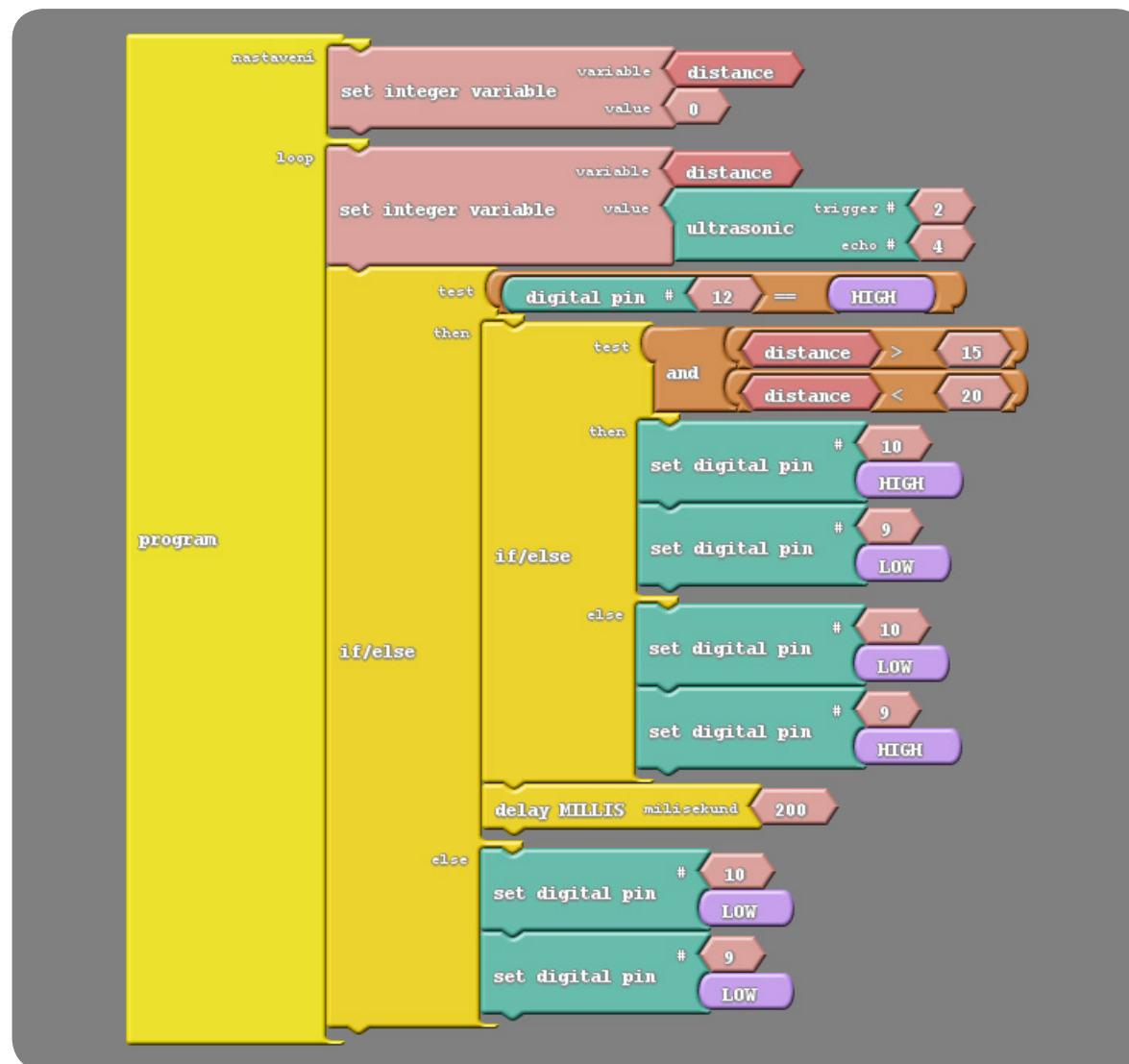


Tip:

- 1) Pro přesnější měření nedávejte před ultrazvukové čidlo pouze ruku, ale nějaký větší plochý předmět (např. postavený penál, kus kartonu, ap.).
- 2) Zkuste upravit zdrojový kód (je na další stránce) pro různé vzdálenosti.

13 Odhad vzdálenosti

Řešení na úrovni programového kódu.



13 Odhad vzdálenosti

(VLASTNÍ KÓD GENEROVANÝ
ARDUBLOCKEM)

```
int _ABVAR_1_distance = 0 ;
int arduinoUltrasonicSensorCodeAutoGeneratedReturnCM(int trigPin, int echoPin)
{
    long duration;
    pinMode(trigPin, OUTPUT);
    pinMode(echoPin, INPUT);
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(20);
    digitalWrite(trigPin, LOW);
    duration = pulseIn(echoPin, HIGH);
    duration = duration / 59;
    if ((duration < 2) || (duration > 300)) return false;
    return duration;
}

void setup()
{
    pinMode( 12 , INPUT);
    digitalWrite( 2 , LOW );

    pinMode( 10 , OUTPUT);
    pinMode( 9 , OUTPUT);
    _ABVAR_1_distance = 0 ;
}

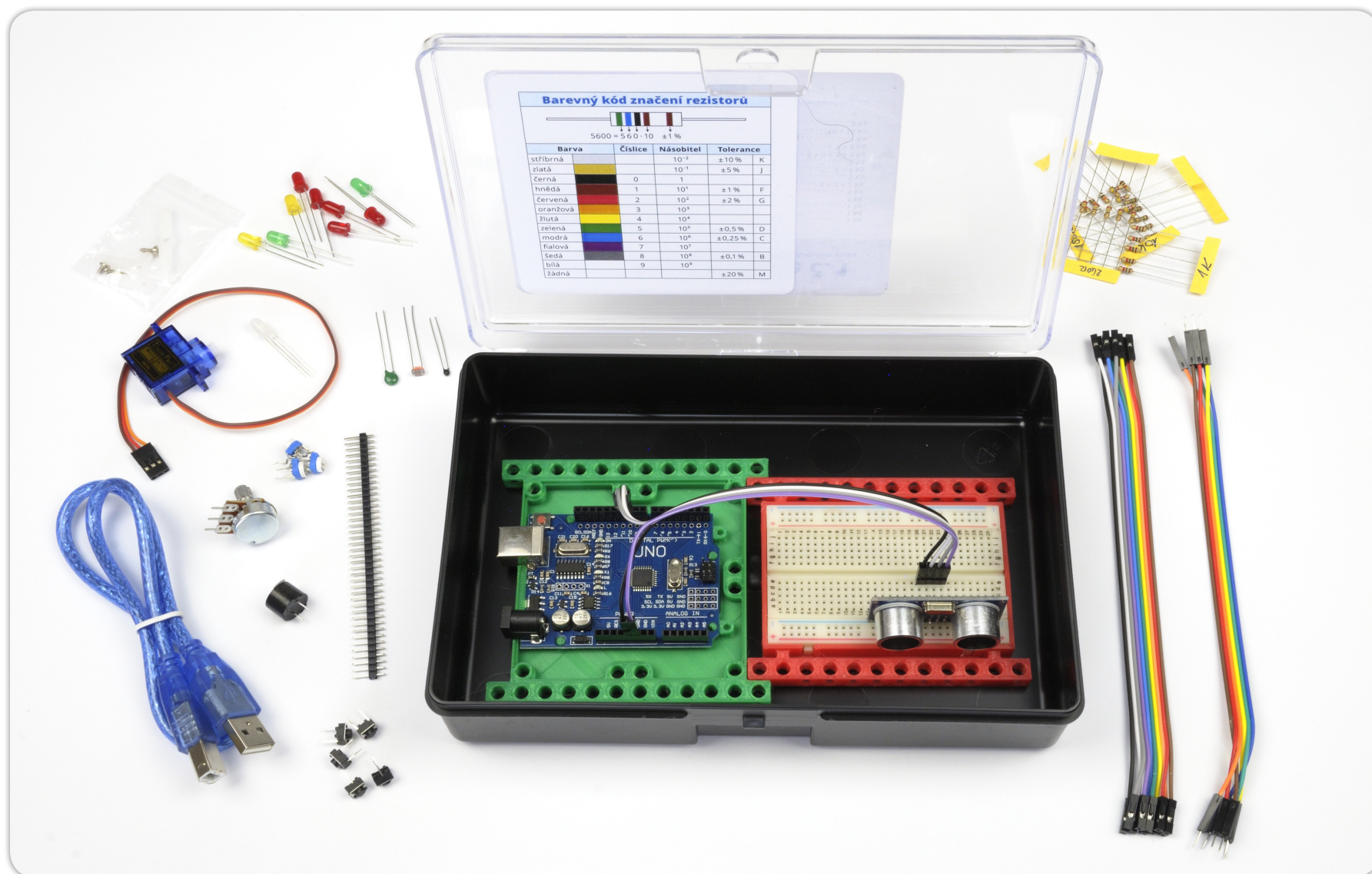
void loop()
{
    _ABVAR_1_distance = arduinoUltrasonicSensorCodeAutoGeneratedReturnCM( 2 , 4 ) ;
    if (( ( digitalRead(12) ) == ( HIGH ) ))
    {
        if (( ( ( _ABVAR_1_distance ) > ( 15 ) ) && ( ( _ABVAR_1_distance ) < ( 20 ) ) ))
        {
            digitalWrite( 10 , HIGH );
            digitalWrite( 9 , LOW );
        }
        else
        {
            digitalWrite( 10 , LOW );
            digitalWrite( 9 , HIGH );
        }
        delay( 200 );
    }
    else
    {
        digitalWrite( 10 , LOW );
        digitalWrite( 9 , LOW );
    }
}
```

Několik dalších námětů na samostatnou činnost

- U zapojení 03 naprogramujte tlačítko tak, aby fungovalo jako vypínač – při jednom stisknutí se dioda rozsvítí, při dalším zhasne. (nápověda: aktuální stav zapnuto/vypnuto si uložte do proměnné)
- U zapojení 05 se matematickou úpravou vyčítaných hodnot v programu pokuste nakalibrovat fotorezistor tak, aby se v sériovém monitoru zobrazovala přibližná hodnota osvětlení v luxech. Pro kalibraci můžete použít například mobilní telefon s vhodnou aplikací obsahující luxmetr.
- U zapojení 06 se pokuste podobným způsobem nakalibrovat termistor tak, aby zobrazoval v sériovém monitoru teplotu ve °C.
- Když se přiblížím na méně než 50 cm, rozsvítí se dioda (využijte zkušenosti ze zapojení 07 a 13).
- Když se přiblížím na méně než 30 cm, spustí se zvukový alarm (využijte zkušenosti ze zapojení 04 a 07).
- Když se přiblížím na méně než 10 cm, zavře se závora ovládaná servem (využijte zkušenosti ze zapojení 07 a 08).
- Když se teplota dostane nad určitou hodnotu, spustí se zvukový alarm (využijte zkušenosti ze zapojení 04 a 06).
- Když osvětlení poklesne pod určitou hodnotu, začnou střídavě blikat dvě diody (využijte zkušenosti ze zapojení 01 a 05).
- Když se budu rukou přibližovat k ultrazvukovému čidlu, bude se příslušně měnit výška tónu bzučáku (využijte zkušenosti ze zapojení 06 a 07).
- Světelný had tvořený pěti LED řízený potenciometrem, vzdáleností od ultrazvukového čidla, teplotou...
- Servo ovládané termistorem – ručičkový teploměr.
- Změna barvy RGB LED v závislosti na teplotě – barevný teploměr.
- Ovládání natočení serva pomocí potenciometru – model dálkového ovládání.
- Jas LED v závislosti na velikosti osvětlení fotorezistoru – model regulace podsvícení displeje.

ARDUINO - VÝUKOVÉ KARTY

Seznam potřebných součástek odpovídá obsahu DIY Arduino sady dostupné na stránkách časopisu e-Mole (Výuková a experimentální Arduino sada).



Tyto karty jsou vytvořeny s využitím volně dostupných SW nástrojů: **Arduino IDE** (<http://www.arduino.cc>), **ArduBlock** (<http://blog.ardublock.com>) a **Fritzing** (<http://fritzing.org>).
Pokud máte o tyto karty zájem a nemáte zakoupenou naši Arduino sadu, kontaktujte nás na adrese casopis.e.mole@gmail.com.