

Príloha A – Dotazník

BBC micro:bit a Arduino

pre žiakov ZŠ,SŠ

* Povinné

Som *

chlapec

dievča

Stretol si sa už s edukačnou doskou BBC micro:bit?

áno

nie

Stretol si sa už s mikrokontrolérom Arduino?

áno

nie

Čo si robil s micro:bitom?

Vaša odpoveď

Čo si robil s arduinom?

Vaša odpoveď

Ako sa ti páčila hodina s micro:bitom?

- 1 - nepáčila
- 2 - skôr nepáčila
- 3 - neviem
- 4 - páčila
- 5 - veľmi páčila

Ako sa ti páčila hodina s mikrokontrolérom Arduino?

- 1 - nepáčila
- 2 - skôr nepáčila
- 3 - neviem
- 4 - páčila
- 5 - veľmi páčila

Aká bola hodina pre teba?

- 1 - veľmi ťažká
- 2 - ťažká
- 3 - neviem
- 4 - jednoduchá
- 5 - veľmi jednoduchá

Aké ťažké bolo použitie dosky BBC micro: bit?

- 1 - veľmi ťažké
- 2 - ťažké
- 3 - neviem
- 4 - veľmi ľahké
- 5 - ľahké

Aké ťažké bolo použitie mikrokontroléra Arduino?

- 1 - veľmi ťažké
- 2 - ťažké
- 3 - neviem
- 4 - veľmi ľahké
- 5 - ľahké

Chcel by si pokračovať s edukačnou doskou BBC micro:bit?

- 1 – nie
- 2 – skôr nie
- 3 – neviem
- 4 – áno
- 5 – skôr áno

Chcel by si pokračovať s mikrokontrolérom Arduino?

- 1 – nie
- 2 – skôr nie
- 3 – neviem
- 4 – áno
- 5 – skôr áno

Čo sa ti nepáčilo pri práci s BBC micro:bitom?

Vaša odpoveď

Čo sa ti páčilo pri práci s BBC micro:bitom?

Vaša odpoveď

Čo sa ti nepáčilo s prácou mikrokontroléra Arduino?

Vaša odpoveď

Čo sa ti páčilo s prácou mikrokontroléra Arduino?

Vaša odpoveď

Ako si sa cítil pri programovaní dosky BBC micro: bit?

- 1 - veľmi zle
- 2 - zle
- 3 - neviem
- 4 - dobre
- 5 - veľmi dobre

Ako si sa cítil pri programovaní dosky Arduino?

- 1 - veľmi zle
- 2 - zle
- 3 - neviem
- 4 - dobre
- 5 - veľmi dobre

Myslíš si, že sa tvoja úroveň programovania po týchto hodinách zlepšila?

Vaša odpoveď

Počul si už predtým o IoT - Internet vecí?

áno

nie

Ohodnot svojú úroveň znalostí z programovania - ako to cítiš Ty.

0 1 2 3 4 5

neviem programovať

viem programovať

Chcel by si niečo odkázať autorovi metodík?

Vaša odpoveď

Príloha B – Ovládanie LED diód pomocou dosky BBC micro:bit

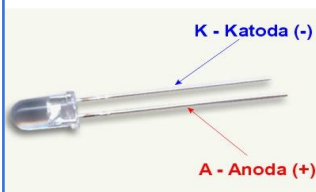
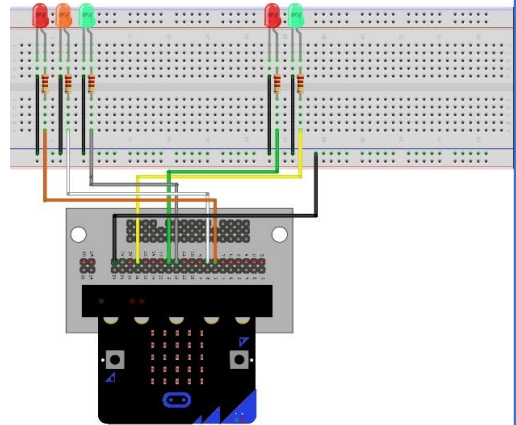


Príloha B. k metodickému listu č. 1 – Ovládanie LED diód pomocou dosky BBC micro:bit

Ovládanie semaforov na križovatke cez tlačidlo

Nasledujúca príloha k metodickému listu je venovaná napodobneniu ovládania semaforov pomocou tlačidla. Pre simuláciu semafora na križovatke použijeme tri LED diódy: červenú, žltú a zelenú. Ďalšie dve LED diódy: zelená a červená slúži pre signalizáciu semafora pre chodcov.

Luminiscenčná dióda alebo svetelná dióda (LED) je polovodičová elektronická súčiastka, ktorá vyžaruje úzko spektrálne svetlo len vtedy, keď ňou prechádza elektrický prúd v priepustom stave. V nepriepustnom stave LED dióda nevyžaruje svetlo. Farba vyžarovaného svetla závisí od štruktúry PN priechodu aj od použitého materiálu.



DAJ SI POZOR

- Dlhší vývod LED diódy (ANODA) musí byť pripojená cez rezistor k danému pinu. Kratší vývod LED diódy (KATODA) je pripojená na uzemnenie (pin GND).

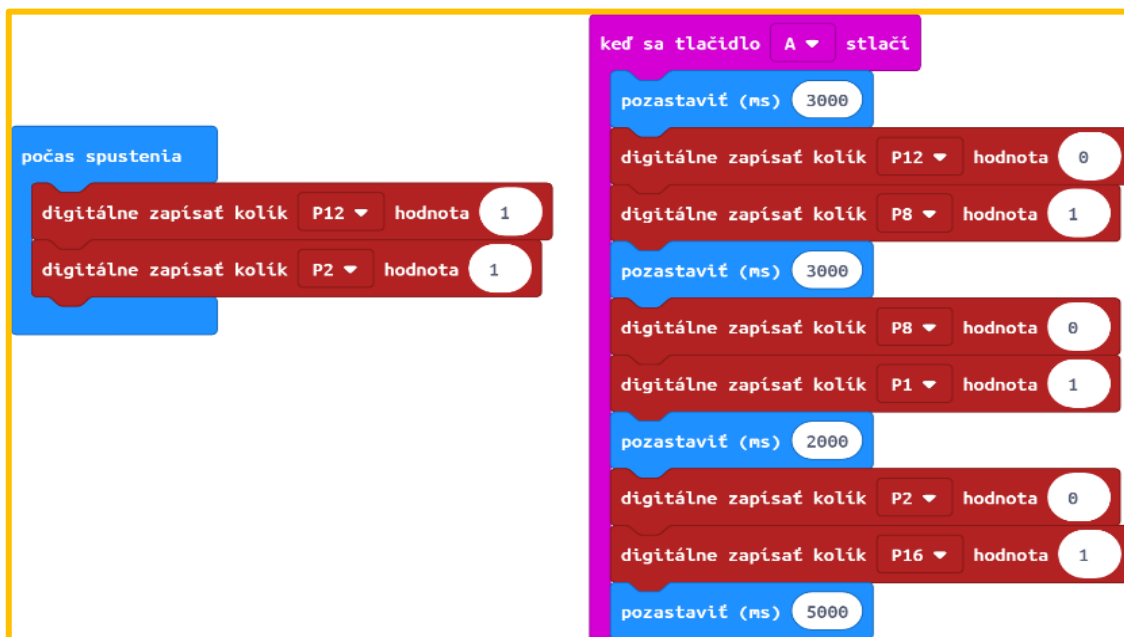


Tlačidlo alebo spínač je komponent, ktorý spája alebo rozopína dva body v obvode práve vtedy, keď ho stlačíte.



Rezistor je lineárny elektrický prvok, ktorého prevažujúca vlastnosť je jeho elektrický odpor. Jeho funkcia je obmedzenie pretekajúceho prúdu v obvode a na zníženie napätia pri záťaži.

Časť z programu:

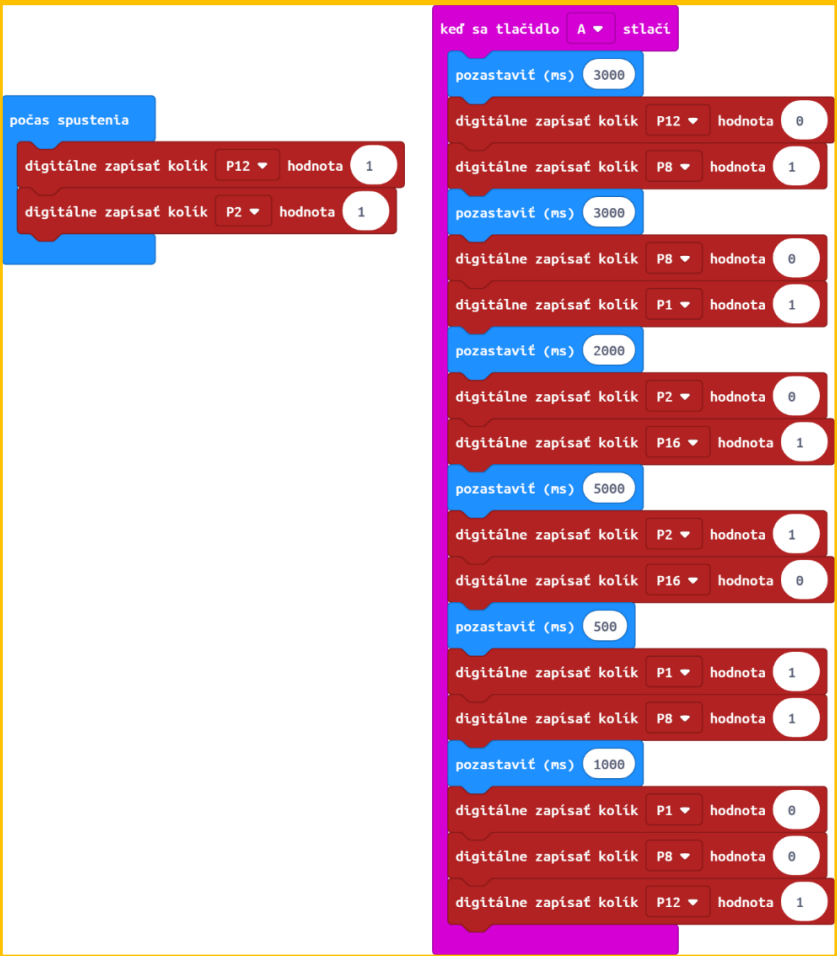


Metodický list číslo 1.


Názov témy:	Metodický list číslo 1. Ovládanie LED diód pomocou dosky BBC micro:bit
Popis témy:	Ovládanie semaforov na križovatke použitím tlačidla
Tematický celok:	Podľa ISCED
Trvanie hodiny:	1 vyučovacia hodina
Cieľová skupina:	SŠ
Didaktické prostriedky:	Vývojová doska BBC micro:bit, micro:bit GPIO Expansion Board, Nepájivé kontaktné pole, káblíky, USB kábel, 2x LED dióda červená, 2x LED dióda zelená, LED dióda oranžová, 5x Rezistor 220Ω, počítač, prístup na internet kvôli programovaciemu prostrediu
Vstupné znalosti:	<ul style="list-style-type: none"> – Študenti poznajú vývojovú dosku BBC micro:bit a jeho komponenty – Základy programovania v programovacom jazyka pre BBC micro:bit
Cieľ hodiny:	Kognitívne ciele: <ul style="list-style-type: none"> – Študent sa naučí programovať a pracovať s LED diódami,

	<ul style="list-style-type: none"> – študent sa vie orientovať v prostredí Online Blockly Editor, – študent dokáže vytvárať a používať premenné v programe, – študent vie vytvárať vlastné cykly a algoritmy riešiace problém. <p>Afektívne ciele:</p> <ul style="list-style-type: none"> – Študent vie rozpoznať, aká časť algoritmu sa má vykonať pred, počas a po skončení cyklu, – študent vie diskutovať o možnostiach použitia konkrétneho hardvéru.
Štruktúra vyučovacej hodiny:	<ol style="list-style-type: none"> 1. Úvod – oboznámenie sa s priebehom vyučovacej hodiny, rozdanie didaktických prostriedkov 2. Riešenie úlohy 3. Záverečná diskusia
Všeobecnodidaktické metódy:	<ul style="list-style-type: none"> • Informačno-Receptívna • Reproductívna • Problémový výklad • Heuristická • Výskumná
Priebeh vyučovacej hodiny:	<p>V úvode hodiny je potrebné zopakovať si pravidlá pri zapájaní a programovaní vývojovej dosky BBC micro:bit. Vysvetlíme si systém správaní semafora na križovatke. Objasníme si princíp fungovania použitých súčiastok. Pri dióde upozorníme žiakov na zapojenie, pri ktorej je dôležitá polarita. Riešime úlohu podľa zadania. Na konci vyučovacej hodiny zhodnotíme úspešnosť zadanej úlohy. Vytvoríme diskusiu o možno použití daného zapojenia v reálnom živote. Pozbierame a skontrolujeme stav didaktických pomôcok.</p> <p>Aplikované funkcie: on button A pressed, digital write, pause (ms)</p> <p>Zadanie úlohy: Vytvorte program, v ktorom budú LED diódy, modelom semafora a budú ovládané pomocou tlačidla. Na simuláciu semafora na hlavnej ceste použijete tri LED diódy:</p>

	<p>červenú, žltú a zelenú. Ďalšie dve LED diódy zelenú a červenú aplikujte pre signalizáciu semafora pre chodcov. Naprogramujte program s nasledujúcimi požiadavkami:</p> <ol style="list-style-type: none"><li data-bbox="632 353 1430 667">1. Červená LED dióda na križovatke pripojená na PIN 1, oranžová LED dióda na križovatke pripojená na PIN 8, zelená LED dióda na križovatke pripojená na PIN 12, zelená LED dióda na prechode pre chodcov pripojená na PIN 16, červená LED dióda na prechode pre chodcov pripojená na PIN 2<li data-bbox="632 745 1430 1541">2. Ak sa stlačí tlačidlo na prechode pre chodcov - program počká 3 sekundy, vypne zelenú LED diódu na križovatke, zapne oranžovú LED diódu na križovatke, program počká 3 sekundy, vypne oranžovú LED diódu na križovatke, zapne červenú LED diódu na križovatke, program počká 2 sekundy, vypne červenú LED diódu na prechode pre chodcov, zapne zelenú LED diódu na prechode pre chodcov, program čaká 5 sekúnd pokým prejdú chodci cez zelenú, následne vypne zelenú LED diódu na prechode pre chodcov, program počká 0,5 sekundy, zapne červenú LED diódu na prechode pre chodcov, zapne oranžovú LED diódu na križovatke, program počká 1 sekundu, vypne červenú LED diódu na križovatke, vypne oranžovú LED diódu na križovatke, zapne zelenú LED diódu na križovatke<li data-bbox="632 1574 1430 1821">3. Ak sa nestlačí tlačidlo na prechode pre chodcov - zapne zelenú LED diódu na križovatke, zapne červenú LED diódu na prechode pre chodcov, vypne zelenú LED diódu na prechode pre chodcov, vypne oranžovú LED diódu na križovatke, vypne červenú LED diódu na križovatke
--	--

	<p>Riešenie:</p>  <pre> when started digitálne zapísať kolík P12 hodnota 1 digitálne zapísať kolík P8 hodnota 1 digitálne zapísať kolík P2 hodnota 1 when button A is pressed pozastaviť (ms) 3000 digitálne zapísať kolík P12 hodnota 0 digitálne zapísať kolík P8 hodnota 1 pozastaviť (ms) 3000 digitálne zapísať kolík P8 hodnota 0 digitálne zapísať kolík P1 hodnota 1 pozastaviť (ms) 2000 digitálne zapísať kolík P2 hodnota 0 digitálne zapísať kolík P16 hodnota 1 pozastaviť (ms) 5000 digitálne zapísať kolík P2 hodnota 1 digitálne zapísať kolík P16 hodnota 0 pozastaviť (ms) 500 digitálne zapísať kolík P1 hodnota 1 digitálne zapísať kolík P8 hodnota 1 pozastaviť (ms) 1000 digitálne zapísať kolík P1 hodnota 0 digitálne zapísať kolík P8 hodnota 0 digitálne zapísať kolík P12 hodnota 1 </pre>
<p>Hodnotenie (Spätá väzba):</p>	<p>Učiteľ vyhodnotí aktivitu žiakov počas vyučovacej hodiny.</p>
<p>Prílohy:</p>	<p>Príloha B. k metodickému listu č. 1</p>

Príloha C – Senzor plameňa naprogramovaný pomocou dosky BBC micro:bit





Príloha C. k metodickému listu č. 2 – Senzor plameňa naprogramovaný pomocou dosky BBC micro:bit

Detekcia plameňa


Infračervený senzor plameňa KY-026 deteguje plameň alebo podobné svetlo s vlnovou dĺžkou od 760 nm do 1100 nm. Rozsah detekčného uhla je približne 60 stupňov. Na module je možné ladiť citlivosť zabudovaným potenciometrom. Modul je schopný detekovať plameň zapalovača až na vzdialenosť 80cm.

Senzor obsahuje 4 pripojovacie piny:


- Micro:bit GND -> Modul G
- Micro:bit + 3,3V -> Modul +
- Modul D0 -vysoký / nízky elektrický výstup signálu úrovne
- Micro:bit PIN 0-2 -> modul A0 - výstupný signál termistorového napätia v reálnom čase



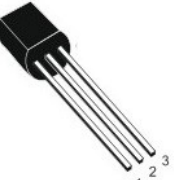
Pre viac informácií pozri Prílohu B. k metodickému listu č. 1



Aktívny buzzer pre alarm alebo signalizáciu. Pri napájaní buzzer “pípa” frekvenciou približne 2300 Hz. V prípade, že hľadáte buzzer vytvárajúci jednoduchú melódiu, prejdite na pasívny buzzer.

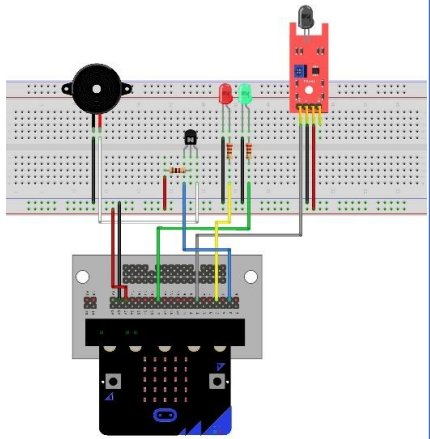


Pre viac informácií pozri Prílohu B. k metodickému listu č. 1

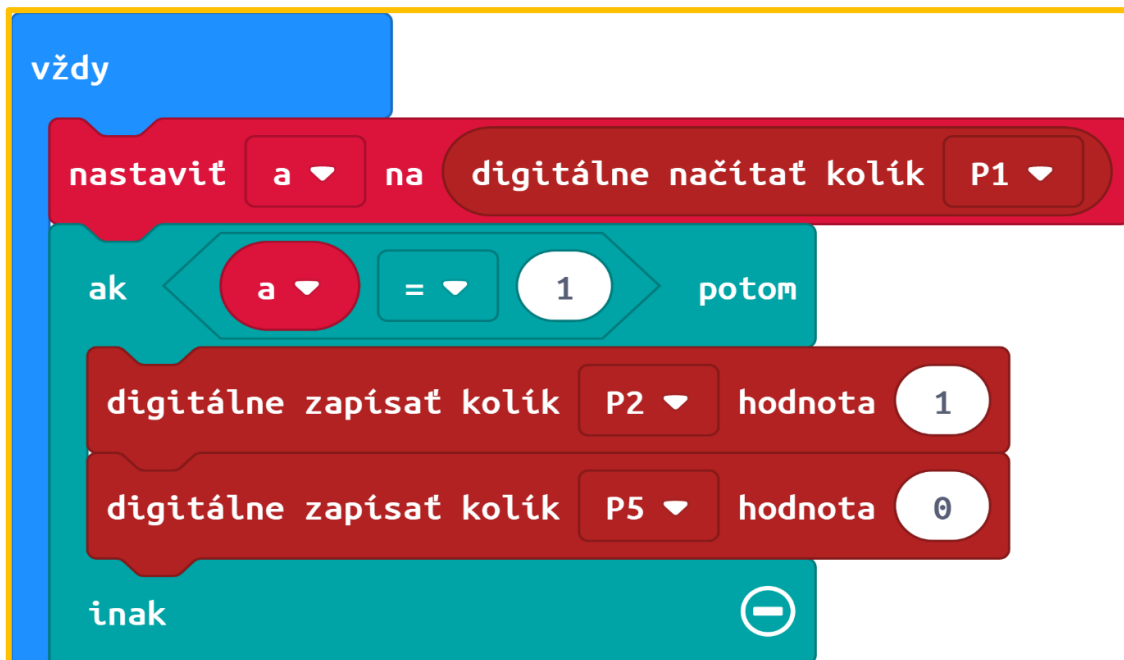


1. Emitter
2. Base
3. Collector

Tranzistor alebo zriedkavo polovodičová trióda je polovodičová súčiastka, používaná ako zosilňovač, spínač, stabilizátor a modulátor elektrického napätia alebo prúdu.



Časť z programu:



Metodický list číslo 2.

Názov témy:	Metodický list číslo 2. Senzor plameňa naprogramovaný pomocou dosky BBC micro:bit
Popis témy:	Detekcia plameňa pomocou senzora
Tematický celok	Podľa ISCED
Trvanie hodiny:	1 vyučovacia hodina
Cieľová skupina:	SŠ
Didaktické prostriedky:	Vývojová doska BBC micro:bit, micro:bit GPIO Expansion Board, Nepájivé kontaktné pole, káblíky, USB kábel, Senzor plameňa KY-026, Buzzer, LED dióda červená, LED dióda zelená, 2x Rezistor 220Ω, Rezistor 1k Ω, Tranzistor S9013, počítač, prístup na internet kvôli programovaciemu prostrediu
Vstupné znalosti:	<ul style="list-style-type: none">– Študenti poznajú vývojovú dosku BBC micro:bit a jeho komponenty– Základy programovania v programovacom jazyka pre BBC micro:bit

Cieľ hodiny:	<p>Kognitívne ciele:</p> <ul style="list-style-type: none"> – Študent sa naučí programovať a pracovať so senzorom plameňa, – študent sa vie orientovať v prostredí Online Blockly Editor, – študent dokáže vytvárať a používať premenné v programe, – študent vie vytvárať vlastné cykly a algoritmy riešiace problém. <p>Afektívne ciele:</p> <ul style="list-style-type: none"> – Študent vie rozpoznať, aká časť algoritmu sa má vykonať pred, počas a po skončení cyklu, – študent vie diskutovať o možnostiach použitia konkrétneho hardvéru.
Štruktúra vyučovacej hodiny:	<ol style="list-style-type: none"> 4. Úvod – oboznámenie sa s priebehom vyučovacej hodiny, rozdanie didaktických prostriedkov 5. Riešenie úlohy 6. Záverečná diskusia
Všeobecnodidaktické metódy:	<ul style="list-style-type: none"> • Informačno-Receptívna • Reprodukčná • Problémový výklad • Heuristická • Výskumná
Priebeh vyučovacej hodiny:	<p>Zopakujeme si pravidlá pri zapájaní a programovaní vývojovej dosky BBC micro:bit. Vysvetlíme si princíp detegovania plameňa pomocou infračerveného senzora. Na senzore si ukážeme, kde je možné ladiť jeho citlivosť pomocou potenciometra a oboznámime sa s jeho zapojením. Zopakujeme si princíp fungovania použitých súčiastok. Riešime úlohu podľa zadania. Na konci hodiny zhodnotíme úspešnosť zadanej úlohy. Vytvoríme diskusiu o možno využití daného senzora v praxi. Pozbierame a skontrolujeme stav didaktických pomôcok.</p>

	<p>Aplikované funkcie: forever, set to, if, else, digital write, digital read, pause (ms)</p> <p>Zadanie úlohy: Vytvorte program v ktorom bude senzor ohňa detegovať plameň alebo podobné svetlo s vlnovou dĺžkou od 760 nm do 1100 nm. Naprogramujte program s nasledujúcimi požiadavkami:</p> <ol style="list-style-type: none">1. Červená LED dióda pripojená na PIN 5, Zelená LED dióda pripojená na PIN 2, Buzzer pripojený na Emitor tranzistora, Kolektor tranzistora pripojený na PIN 0, Senzor plameňa modul A0 pripojený na PIN 12. Ak senzor deteguje plameň vo svojej blízkosti, buzzer začne oznamovať prítomnosť ohňa alebo podobného svetla s vlnovou dĺžkou od 760 nm do 1100 nm zvukovou signalizáciu (pípanie) a svetelnou signalizáciu (svieti červená LED dióda).3. Nedetegovania ohňa alebo podobného svetla s vlnovou dĺžkou od 760 nm do 1100 nm je signalizované svetelnou signalizáciu (svieti zelená LED dióda).
--	--

Riešenie:

```
vždy
  nastaviť a na digitálne načítať kolík P1
  ak a = 1 potom
    digitálne zapísať kolík P2 hodnota 1
    digitálne zapísať kolík P5 hodnota 0
  inak -
    digitálne zapísať kolík P2 hodnota 0
    digitálne zapísať kolík P5 hodnota 1
    digitálne zapísať kolík P0 hodnota 1
  pozastaviť (ms) 1000
  digitálne zapísať kolík P0 hodnota 0
  +
```

Hodnotenie (Spätňá väzba):	Učiteľ vyhodnotí aktivitu žiakov počas vyučovacej hodiny.
Prílohy:	Príloha C. k metodickému listu č. 2

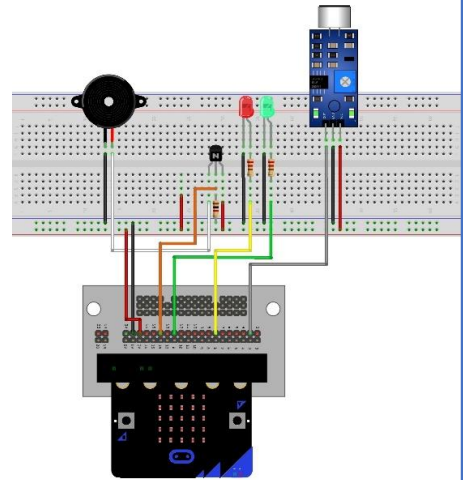
Príloha D - Senzor zvuku naprogramovaní pomocou dosky BBC micro:bit



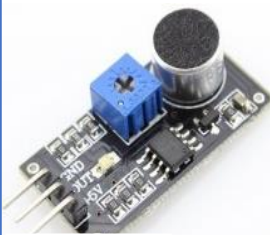
Príloha D. k metodickému listu č.3 – Senzor zvuku naprogramovaní pomocou dosky BBC micro:bit

Detekcia zvuku

Zvukový senzor ER-SPM22101S obsahuje mikrofón s výkonomým zosilňovačom LM386 3-5V. Na základe výkonomého zosilňovača a elekterového mikrofónu sa môže použiť na zistenie, či existuje nejaký zvuk, ktorý presahuje prahovú hodnotu. Prahová hodnota sa dá nastaviť pomocou potenciometra. Doska pre zvukový senzor obsahuje indikačnú červenú LED diódu, ktorá sa rozsvieti pri hluku , ktorá presahuje nastavenú hranicu.



Senzor obsahuje 3 pripojovacie piny:



- Micro:bit GND -> Modul GND
- Micro:bit + 3,3V -> Modul +
- Micro:bit PIN 0-2 -> Modul OUT -digitálny výstup signálu



Pre viac informácií pozri Príloha B. k metodickému listu č. 1



Pre viac informácií pozri Príloha C. k metodickému listu č. 2



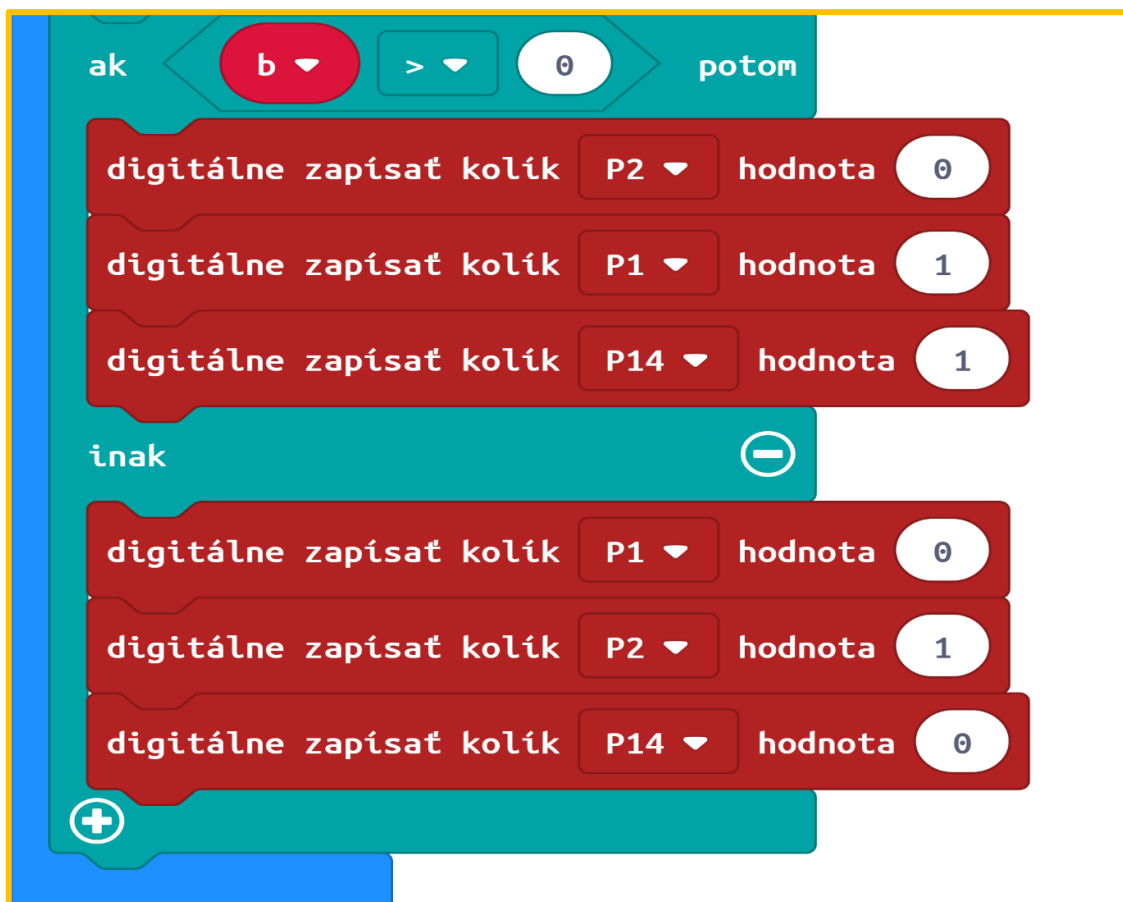
Pre viac informácií pozri Príloha B. k metodickému listu č. 1



Pre viac informácií pozri Príloha C. k metodickému listu č. 2

1. Emitter
2. Base
3. Collector

Časť z programu:

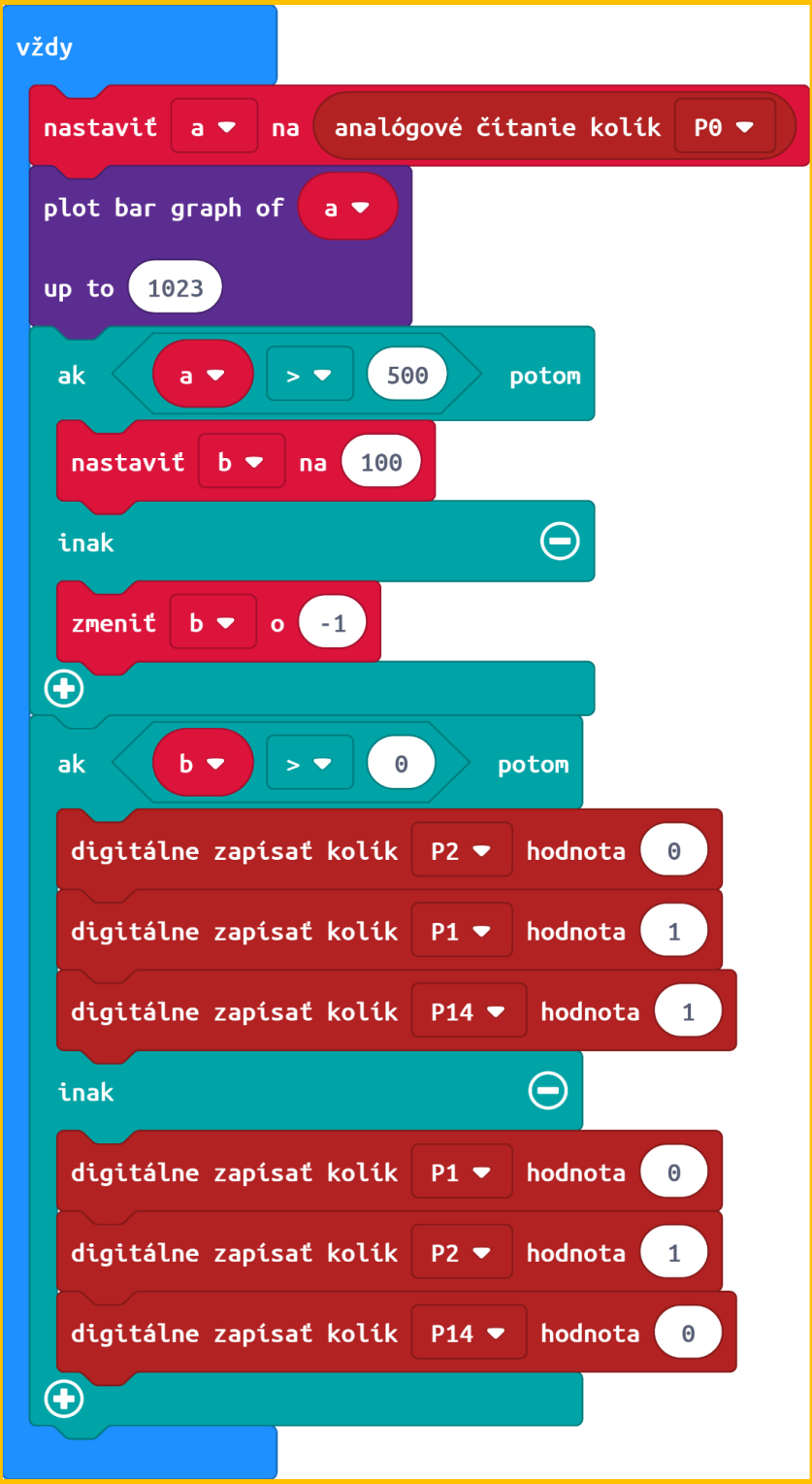


Metodický list číslo 3.

Názov témy:	Metodický list číslo 3. Senzor zvuku naprogramovaný pomocou dosky BBC micro:bit
Popis témy:	Detekcia zvuku pomocou senzora
Tematický celok	Podľa ISCED
Trvanie hodiny:	1 vyučovacia hodina
Cieľová skupina:	SŠ
Didaktické prostriedky:	Vývojová doska BBC micro:bit, micro:bit GPIO Expansion Board, Nepájivé kontaktné pole, káblíky, USB kábel, Zvukový senzor ER-SPM22101S, LED dióda červená, LED dióda zelená, 2x Rezistor 220Ω, Buzzer, Rezistor 1k Ω, Tranzistor S9013, počítač, prístup na internet kvôli programovaciemu prostrediu
Vstupné znalosti:	– Študenti poznajú vývojovú dosku BBC micro:bit a jeho komponenty

	<ul style="list-style-type: none"> – Základy programovania v programovacom jazyka pre BBC micro:bit
Cieľ hodiny:	<p>Kognitívne ciele:</p> <ul style="list-style-type: none"> – Študent sa naučí programovať a pracovať so senzorom zvuku, – študent sa vie orientovať v prostredí Online Blockly Editor,, – študent dokáže vytvárať a používať premenné v programe, – študent vie vytvárať vlastné cykly a algoritmy riešiace problém. <p>Afektívne ciele:</p> <ul style="list-style-type: none"> – Študent vie rozpoznať, aká časť algoritmu sa má vykonať pred, počas a po skončení cyklu, – študent vie diskutovať o možnostiach použitia konkrétneho hardvéru.
Štruktúra vyučovacej hodiny:	<ol style="list-style-type: none"> 1. Úvod – oboznámenie sa s priebehom vyučovacej hodiny, rozdanie didaktických prostriedkov 2. Riešenie úlohy 3. Záverečná diskusia
Všeobecnodidaktické metódy:	<ul style="list-style-type: none"> • Informačno-Receptívna • Reprodukčívna • Problémový výklad • Heuristická • Výskumná
Priebeh vyučovacej hodiny:	<p>Zopakujeme si pravidlá pri zapájaní a programovaní vývojovej dosky BBC micro:bit. Vysvetlíme si princíp detegovania zvuku pomocou zvukového senzora. Na senzore si ukážeme, kde je možné ladiť jeho citlivosť pomocou potenciometra a oboznámime sa s jeho zapojením. Zopakujeme si princíp fungovania použitých súčiastok. Riešime úlohu podľa zadania. Na konci hodiny zhodnotíme úspešnosť zadanej úlohy. Vytvoríme diskusiu o možno využití daného senzora v praxi. Pozbierame a skontrolujeme stav didaktických pomôcok.</p>

	<p>Aplikované funkcie: forever, set to, change by, if, then, else, digital write, analog read</p> <p>Zadanie úlohy: Vytvorte program, v ktorom bude senzor zvuku detegovať zvukové hodnoty. Naprogramujte program s nasledujúcimi požiadavkami:</p> <ol style="list-style-type: none">1. Červená LED dióda pripojená na PIN 1, Zelená LED dióda pripojená na PIN 2, Buzzer pripojený na Emitter tranzistora, Báza tranzistora pripojený na PIN 14, Senzor zvuku modul OUT pripojený na PIN 02. Ak senzor deteguje zvuk vo svojej blízkosti, buzzer začne oznamovať prítomnosť hluku zvukovou signalizáciou (pípanie) a svetelnou signalizáciou (svieti červená LED dióda).3. V prípade nedetegovania zvuku je vysielaný signál svetelnou signalizáciou (svieti zelená LED dióda).
--	---

	<p>Riešenie:</p>  <pre> vždy nastaviť a na analógové čítanie kolík P0 plot bar graph of a up to 1023 ak a > 500 potom nastaviť b na 100 inak zmeniť b o -1 ak b > 0 potom digitálne zapísať kolík P2 hodnota 0 digitálne zapísať kolík P1 hodnota 1 digitálne zapísať kolík P14 hodnota 1 inak digitálne zapísať kolík P1 hodnota 0 digitálne zapísať kolík P2 hodnota 1 digitálne zapísať kolík P14 hodnota 0 </pre>
<p>Hodnotenie (Spätná väzba):</p>	<p>Učiteľ vyhodnotí aktivitu žiakov počas vyučovacej hodiny.</p>
<p>Prílohy:</p>	<p>Príloha D. k metodickému listu č. 3</p>

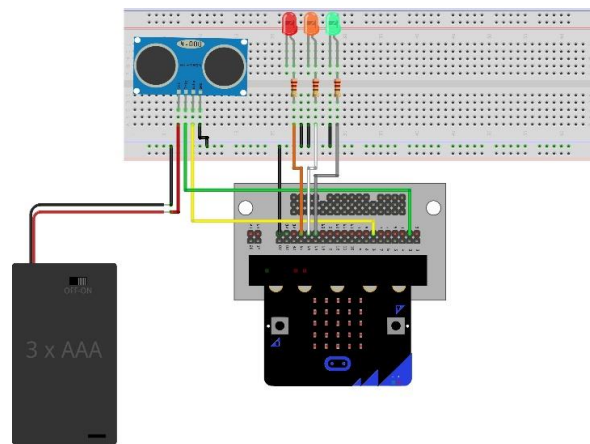
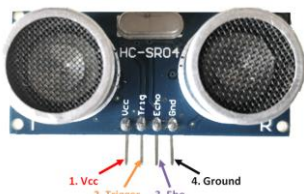
Príloha E – Parkovací asistent naprogramovaný pomocou dosky BBC micro:bit



Príloha E. k metodickému listu č. 4 - Parkovací asistent naprogramovaný pomocou dosky BBC micro:bit

Detekcia vzdialenosti pomocou ultrazvukového senzora

Ultrazvukový senzor HC-SR04 slúži na meranie vzdialenosti. Detekčná vzdialenosť je od 1 cm do 4,5 m s presnosťou na 0,3 cm. Pozorovací uhol je do 15°.



Senzor obsahuje 4 pripojovacie piny:

- Batéria + 4,5V -> Modul Vcc
- Micro:bit GND -> Modul Gnd
- Micro:bit PIN 0-2 -> Modul Trigger - vysiela signál (vysokofrekvenčný zvuk)
- Micro:bit PIN 0-2 -> Modul Echo - produkuje impulz len vtedy keď je prijatý odrazený signál



Pre viac informácií pozri Príloha B. k metodickému listu č. 1



Pre viac informácií pozri Príloha B. k metodickému listu č. 1

Časť z programu:



Metodický list číslo 4.

Názov témy:	Metodický list číslo 4. Parkovací asistent naprogramovaný pomocou dosky BBC micro:bit
Popis témy:	Detekcia vzdialenosti pomocou ultrazvukového senzora
Tematický celok	Podľa ISCED
Trvanie hodiny:	1 vyučovací hodina
Cieľová skupina:	SŠ
Didaktické prostriedky:	Vývojová doska BBC micro:bit , BBC micro:bit GPIO Expansion Board, Nepájivé kontaktné pole, káblíky, USB kábel, LED dióda červená. LED dióda zelená, LED dióda oranžová, 3x Rezistor 220Ω, Ultrazvukový senzor vzdialenosti HC-SR04, batéria 4,5V, počítač, prístup na internet kvôli programovaciemu prostrediu
Vstupné znalosti:	– Študenti poznajú vývojovú dosku BBC Micro:bit a jeho komponenty

	<ul style="list-style-type: none"> – Základy programovania v programovacom jazyka pre BBC Micro:bit
Cieľ hodiny:	<p>Kognitívne ciele:</p> <ul style="list-style-type: none"> – Študent sa naučí programovať a pracovať so senzorom vzdialenosti, LCD displejom, – študent sa vie orientovať v prostredí Online Blockly Editor, – študent dokáže vytvárať a používať premenné v programe, – študent vie vytvárať vlastné cykly a algoritmy riešiace problém. <p>Afektívne ciele:</p> <ul style="list-style-type: none"> – Študent vie rozpoznať, aká časť algoritmu sa má vykonať pred, počas a po skončení cyklu, – študent vie diskutovať o možnostiach použitia konkrétneho hardvéru.
Štruktúra vyučovacej hodiny:	<ol style="list-style-type: none"> 4. Úvod – oboznámenie sa s priebehom vyučovacej hodiny, rozdanie didaktických prostriedkov 5. Riešenie úlohy 6. Záverečná diskusia
Všeobecnodidaktické metódy:	<ul style="list-style-type: none"> • Informačno-Receptívna • Reprodukčná • Problémový výklad • Heuristická • Výskumná
Priebeh vyučovacej hodiny:	<p>Zopakujeme si pravidlá pri zapájaní a programovaní vývojovej dosky BBC micro:bit. Vysvetlíme si systém správanie sa parkovacieho asistenta. Objasníme si princíp fungovania použitých súčiastok.</p> <p>Riešime úlohu podľa zadania. Na konci hodiny zhodnotíme úspešnosť zadanej úlohy. Vytvoríme diskusiu o možno využití daného zapojenia v praxi. Pozbierame a skontrolujeme stav didaktických pomôcok.</p>

	<p>Aplikované funkcie: forever, set to, show number, if, then, else, digital write</p> <p>Zadanie úlohy: Vytvorte program, v ktorom budú LED diódy, ultrazvukový senzor vzdialenosti modelovať parkovací asistent. Naprogramujte program s nasledujúcimi požiadavkami:</p> <ol style="list-style-type: none">1. Červená LED dióda pripojená na PIN 16, Oranžová LED dióda pripojená na PIN 15, Zelená LED dióda pripojená na PIN 14, Trigger senzora pripojený na PIN 0, Echo senzora pripojený na PIN 12. Ak je vzdialenosť senzora od objektu väčšia alebo rovná ako 10 cm tak:<ol style="list-style-type: none">a) Zobrazuj na displeji vzdialenosť senzora od objektu v cm,b) Zapni zelenú LED diódu, vypni červenú a oranžovú LED diódu,c) Opakuj cyklus3. Ak je vzdialenosť senzora od objektu väčšia alebo rovná 5cm:<ol style="list-style-type: none">a) Zobrazuj na displeji vzdialenosť senzora od objektu v cm,b) Zapni oranžovú LED diódu, vypni zelenú a červenú LED diódu,c) Opakuj cyklus4. Ak je vzdialenosť senzora od objektu menšia ako 5cm:<ol style="list-style-type: none">a) Zobrazuj na displeji vzdialenosť senzora od objektu v cm,b) Zapni červenú LED diódu, vypni oranžovú a zelenú LED diódu,c) Opakuj cyklus
--	--

Riešenie:

The image shows a Scratch script for controlling an ultrasonic sensor. It starts with a 'vždy' (forever) loop. Inside the loop, there is a 'ping trig' block with 'P0' selected, followed by an 'echo' block with 'P1' selected, and a 'unit' block with 'cm' selected. These three blocks are grouped together in a red oval. Below this is a 'nastaviť vzdialenosť na' block with a dropdown menu set to 'vzdialenosť'. This is followed by a 'zobrazí číslo' block with a dropdown menu set to 'vzdialenosť'. Then, there is an 'ak' (if) block with a condition 'vzdialenosť >= 10'. Inside this 'if' block, there are three 'digitálne zapísať kolík' blocks: P14 with value 1, P15 with value 0, and P16 with value 0. Below the 'if' block is an 'inak' (else) block. Inside the 'else' block, there is another 'ak' (if) block with a condition 'vzdialenosť >= 5'. Inside this 'if' block, there are three 'digitálne zapísať kolík' blocks: P14 with value 0, P15 with value 1, and P16 with value 0. Below this 'if' block is another 'inak' (else) block. Inside this 'else' block, there are three 'digitálne zapísať kolík' blocks: P14 with value 0, P15 with value 0, and P16 with value 1. The script ends with two empty 'digitálne zapísať kolík' blocks at the bottom of the 'else' block.

Hodnotenie (Spätňá väzba):	Učiteľ vyhodnotí aktivitu žiakov počas vyučovacej hodiny.
Prílohy:	Príloha E. k metodickému listu č. 4

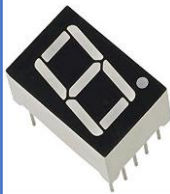
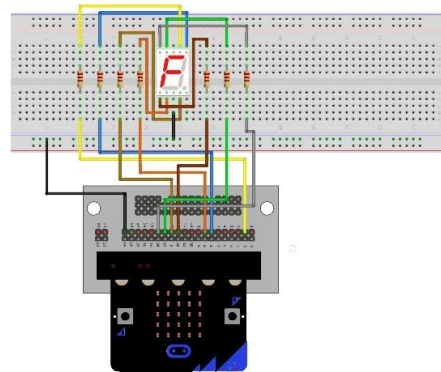
Príloha F – 7 – segmentový displej naprogramovaný pomocou dosky BBC micro:bit



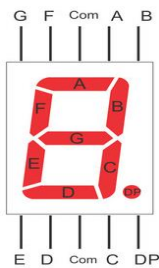
Príloha F. k metodickému listu č. 5 - 7- segmentový displej naprogramovaný pomocou dosky BBC micro:bit

Počítadlo cez 7-segmentový displej

Cieľom tohto pracovného listu bude vytvoriť odpočítavací časovač riadený pomocou BBC micro:bitu, ktorý bude zobrazovať čísla na obrazovke pomocou 7segmentového displeja.



Sedem segmentový displejov je bežný spôsob zobrazovania čísel a často používa pre: hodiny, kalkulačky a elektronických meracích prístrojoch.

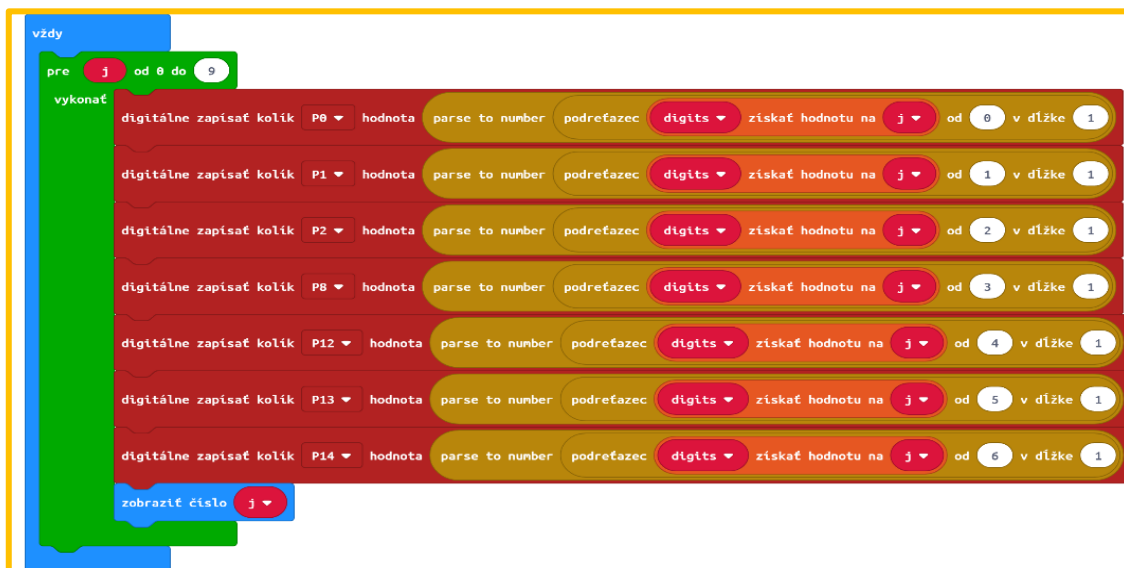


7-segmentové displeje sa zvyčajne skladajú zo siedmich jednotlivých farebných LED nazývaných (segmenty). Každý segment je možné zapnúť alebo vypnúť a vytvoriť tak jedinečný vzor. Každý segment je označený písmenom od A až po G.



Pre viac informácií pozri Príloha B. k metodickému listu č. 1

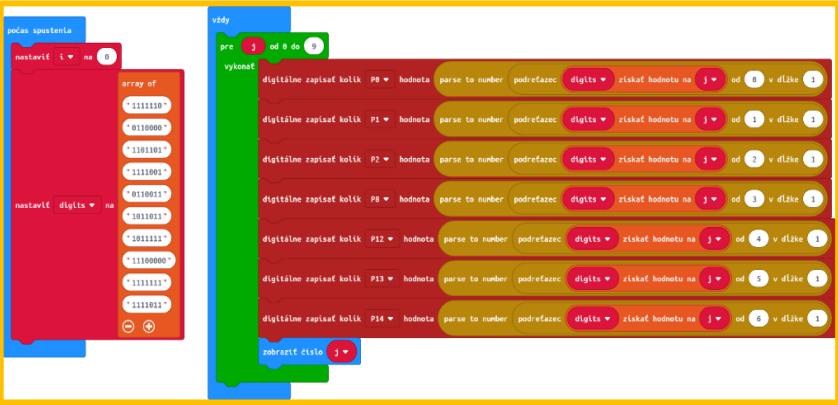
Časť z programu:



Metodický list číslo 5.

Názov témy:	Metodický list číslo 5. 7-segmentový displej naprogramovaný pomocou dosky BBC micro:bit
Popis témy:	Počítadlo cez 7-segmentový displej
Tematický celok	Podľa ISCED
Trvanie hodiny:	1 vyučovacia hodina
Cieľová skupina:	SŠ
Didaktické prostriedky:	Vývojová doska BBC micro:bit , micro:bit GPIO Expansion Board, Nepájivé kontaktné pole, káblíky, USB kábel, 7-segmentový displej, 7x Rezistor 220Ω, počítač, prístup na internet kvôli programovaciemu prostrediu
Vstupné znalosti:	<ul style="list-style-type: none"> – Študenti poznajú vývojovú dosku BBC micro:bit a jeho komponenty – Základy programovania v programovacom jazyka pre BBC micro:bit
Cieľ hodiny:	Kognitívne ciele: <ul style="list-style-type: none"> – Študent sa naučí programovať a pracovať so 7-segmentovým displejom, – študent sa vie orientovať v prostredí Online Blockly Editor,

	<ul style="list-style-type: none"> – študent dokáže vytvárať a používať premenné v programe, – študent vie vytvárať vlastné cykly a algoritmy riešiace problém. <p>Afektívne ciele:</p> <ul style="list-style-type: none"> – Študent vie rozpoznať, aká časť algoritmu sa má vykonať pred, počas a po skončení cyklu, – študent vie diskutovať o možnostiach použitia konkrétneho hardvéru.
Štruktúra vyučovacej hodiny:	<p>7. Úvod – oboznámenie sa s priebehom vyučovacej hodiny, rozdanie didaktických prostriedkov</p> <p>8. Riešenie úlohy</p> <p>9. Záverečná diskusia</p>
Všeobecnodidaktické metódy:	<ul style="list-style-type: none"> • Informačno-Receptívna • Reprodukčná • Problémový výklad • Heuristická • Výskumná
Priebeh vyučovacej hodiny:	<p>Zopakujeme si pravidlá pri zapájaní a programovaní vývojovej dosky BBC micro:bit. Vysvetlíme si princíp zobrazovania segmentov na displeji. Oboznámime sa s jeho zapojením. Zopakujeme si princíp fungovania použitých súčiastok. Riešime úlohu podľa zadania. Na konci hodiny zhodnotíme úspešnosť zadanej úlohy. Vytvoríme diskusiu o možno využití daného senzora v praxi. Pozbierame a skontrolujeme stav didaktických pomôcok.</p> <p>Aplikované funkcie: on start, forever, set to, for, do, digital write, show number</p> <p>Zadanie úlohy: Vytvorte program, v ktorom bude 7-segmentový displej postupne zobrazovať číslice od 0 po 9. Program sa bude opakovať n-krát. Naprogramujte program s nasledujúcimi požiadavkami:</p>

	<p>Segment A pripojený na PIN 0, Segment B pripojený na PIN 1, Segment C pripojený na PIN 2, Segment D pripojený na PIN 8, Segment E pripojený na PIN 12, Segment F pripojený na PIN 13, Segment G pripojený na PIN 14, Segment COM pripojený na PIN GND</p> <p>Riešenie:</p> 
<p>Hodnotenie (Spätná väzba):</p>	<p>Učiteľ vyhodnotí aktivitu žiakov počas vyučovacej hodiny.</p>
<p>Prílohy:</p>	<p>Príloha F. k metodickému listu č. 5</p>

Príloha G – Ovládanie LED diód pomocou dosky Arduino



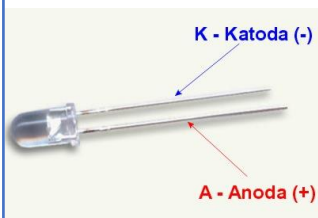
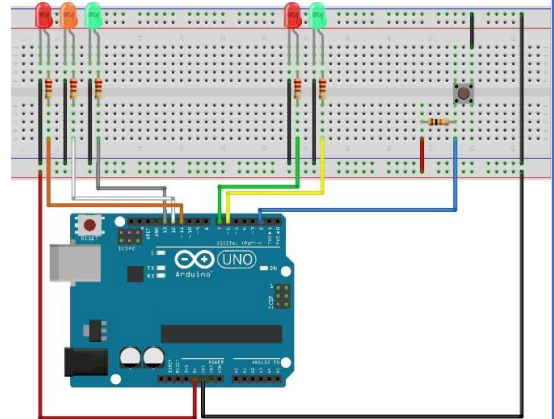
Príloha G. k metodickému listu č. 6 – Ovládanie LED diód pomocou dosky Arduino

Ovládanie semaforov na križovatke cez tlačidlo

Nasledujúca príloha k metodickému listu je venovaná napodobneniu ovládania semaforov pomocou tlačidla. Pre simuláciu semafora na križovatke použijeme tri LED diódy: červenú, žltú a zelenú. Ďalšie dve LED diódy: zelená a červená slúžia pre signalizáciu semafora pre chodcov.

Luminiscenčná dióda alebo svetelná dióda (LED) je polovodičová elektronická súčiastka, ktorá vyžaruje úzko spektrálne svetlo len vtedy,

keď ňou prechádza elektrický prúd v priepustom stave. V nepriepustnom stave LED dióda nevyžaruje svetlo. Farba vyžarovaného svetla závisí od štruktúry PN priechodu aj od použitého materiálu.



DAJ SI POZOR

- Dlhší vývod LED diódy (ANODA) musí byť pripojená cez rezistor k danému pinu. Kratší vývod LED diódy (KATODA) je pripojená na uzemnenie (pin GND).



Tlačidlo alebo spínač je komponent, ktorý spája alebo rozopína dva body v obvode práve vtedy, keď ho stlačíte.



Rezistor je lineárny elektrický prvok, ktorého prevažujúca vlastnosť je jeho elektrický odpor. Jeho funkcia je obmedzenie pretekajúceho prúdu v obvode a zníženie napätia pri záťaži.

Časť z programu:

```
void loop() {  
  
  tlacidlo_chodci=digitalRead(2); //Zaznamenáva či je tlačidlo pre chodcov stlačené alebo nie je stlačené  
  
  if (tlacidlo_chodci == 0) //Ak sa stlačí tlačidlo na prechode pre chodcov tak:  
  {  
    delay(3000); //Program čaká 3 sekundy  
    digitalWrite(9,LOW); //Vypne zelenú LED diódu na križovatke  
    digitalWrite(8,HIGH); //Zapne oranžovú LED diódu na križovatke  
    delay(3000); //Program čaká 3 sekundy  
    digitalWrite(8,LOW); //Vypne oranžovú LED diódu na križovatke  
    digitalWrite(7,HIGH); //Zapne červenú LED diódu na križovatke  
    delay(2000); //Program čaká 2 sekundy  
    digitalWrite(5,LOW); //Vypne červenú LED diódu na prechode pre chodcov  
    digitalWrite(4,HIGH); //Zapne zelenú LED diódu na prechode pre chodcov  
    delay(5000); //Program čaká 8 sekundy pokým prejdú chodci cez zelenú LED diódu  
  
    {  
      digitalWrite(4,LOW); //Vypne zelenú LED diódu na prechode pre chodcov  
      digitalWrite(5,HIGH); //Zapne červenú LED diódu na prechode pre chodcov  
      delay(500);  
      digitalWrite(8,HIGH); //Zapne oranžovú LED diódu na križovatke  
      delay(1000);  
      digitalWrite(5,HIGH); //Zapne červenú LED diódu na prechode pre chodcov  
    }  
  }  
}
```

Metodický list číslo 6.

Názov témy:	Metodický list číslo 6. Ovládanie LED diód pomocou dosky Arduino
Popis témy:	Ovládanie semaforov na križovatke použitím tlačidla
Tematický celok	Podľa ISCED
Trvanie hodiny:	1 vyučovací hodina
Cieľová skupina:	SŠ
Didaktické prostriedky:	Vývojová doska Arduino, Nepájivé kontaktné pole, káblíky, USB kábel, Tlačidlo, 2x LED dióda červená, 2x LED dióda zelená, LED dióda oranžová, 5x Rezistor 220Ω, Rezistor 1kΩ, počítač, nainštalované programovacie prostredie Arduino IDE
Vstupné znalosti:	<ul style="list-style-type: none">– Študenti poznajú vývojovú dosku Arduino a jeho komponenty– Základy programovania v programovacom jazyka pre Arduino
Cieľ hodiny:	Kognitívne ciele: <ul style="list-style-type: none">– Študent sa naučí programovať a pracovať s LED diódami,– študent sa vie orientovať v prostredí Arduino IDE,– študent dokáže vytvárať a používať premenné v programe,

	<p>– študent vie vytvárať vlastné cykly a algoritmy riešiacie problém.</p> <p>Afektívne ciele:</p> <p>– Študent vie rozpoznať, aká časť algoritmu sa má vykonať pred, počas a po skončení cyklu,</p> <p>– študent vie diskutovať o možnostiach použitia konkrétneho hardvéru.</p>
Štruktúra vyučovacej hodiny:	<p>10. Úvod – oboznámenie sa s priebehom vyučovacej hodiny, rozdanie didaktických prostriedkov</p> <p>11. Riešenie úlohy</p> <p>12. Záverečná diskusia</p>
Všeobecnodidaktické metódy:	<ul style="list-style-type: none"> • Informačno-Receptívna • Reprodukčívna • Problémový výklad • Heuristická • Výskumná
Priebeh vyučovacej hodiny:	<p>V úvode hodiny je potrebné zopakovať si pravidlá pri zapájaní a programovaní vývojovej dosky Arduino. Vysvetlíme si systém správania sa semafora na križovatke. Objasníme si princíp fungovania použitých súčiastok. Pri dióde upozorníme žiakov na zapojenie diódy, pri ktorej je dôležitá polarita. Riešime úlohu podľa zadania. Na konci vyučovacej hodiny zhodnotíme úspešnosť zadanej úlohy. Vytvoríme diskusiu o možno použití daného zapojenia v reálnom živote. Pozbierame a skontrolujeme stav didaktických pomôcok.</p> <p>Aplikované funkcie: setup, loop, if, else, digitalWrite, digitalRead, pinMode, delay</p> <p>Zadanie úlohy: Vytvorte program, v ktorom budú LED diódy, modelom semafora a budú ovládané pomocou tlačidla. Na simuláciu semafora na hlavnej ceste použijete tri LED diódy: červenú, žltú a zelenú. Ďalšie dve LED diódy zelenú a červenú aplikujte pre signalizáciu semafora pre chodcov. Naprogramujte program s nasledujúcimi požiadavkami:</p>

	<ol style="list-style-type: none">1. Červená LED dióda na križovatke pripojená na PIN 7, oranžová LED dióda na križovatke pripojená na PIN 8, zelená LED dióda na križovatke pripojená na PIN 9, tlačidlo pripojené na PIN 2, zelená LED dióda na prechode pre chodcov pripojená na PIN 4, červená LED dióda na prechode pre chodcov pripojená na PIN 5, 2. Ak sa stlačí tlačidlo na prechode pre chodcov - program počká 3 sekundy, vypne zelenú LED diódu na križovatke, zapne oranžovú LED diódu na križovatke, program počká 3 sekundy, vypne oranžovú LED diódu na križovatke, zapne červenú LED diódu na križovatke, program počká 2 sekundy, vypne červenú LED diódu na prechode pre chodcov, zapne zelenú LED diódu na prechode pre chodcov, program čaká 5 sekúnd pokým prejdú chodci cez zelenú, následne vypne zelenú LED diódu na prechode pre chodcov, zapne červenú LED diódu na prechode pre chodcov, program počká 0,5 sekundy, zapne oranžovú LED diódu na križovatke, program počká 1 sekundu, zapne červenú LED diódu na prechode pre chodcov, program čaká 2 sekundy potom zapne zelenú LED diódu na križovatke, zapne červenú LED dióda na prechode pre chodcov 3. Ak sa nestlačí tlačidlo na prechode pre chodcov - zapne zelenú LED diódu na križovatke, zapne červenú LED diódu na prechode pre chodcov, vypne zelenú LED diódu na prechode pre chodcov, vypne oranžovú LED diódu na križovatke, vypne červenú LED diódu na križovatke
--	--

	<p>Riešenie:</p> <pre> byte tlacidlo_chodci=0; //Zapamätáva si stav stlačeného tlačidla na prechode pre chodcov void setup() { pinMode(7, OUTPUT); //Definuje červenú LED diódu na križovatke ako výstupné rozhranie na PIN 7 pinMode(8, OUTPUT); //Definuje oranžovú LED diódu na križovatke ako výstupné rozhranie na PIN 8 pinMode(9, OUTPUT); //Definuje zelenú LED diódu na križovatke ako výstupné rozhranie na PIN 9 pinMode(2, INPUT); //Definuje tlačidlo ako vstupné rozhranie na PIN 2 pinMode(4, OUTPUT); //Definuje zelenú LED diódu na prechode pre chodcov ako výstupné rozhranie na PIN 4 pinMode(5, OUTPUT); //Definuje červenú LED diódu na prechode pre chodcov ako výstupné rozhranie na PIN 5 } void loop() { tlacidlo_chodci=digitalRead(2); //Zaznamenáva či je tlačidlo pre chodcov stlačené alebo nie je stlačené if (tlacidlo_chodci == 0) //Ak sa stlačí tlačidlo na prechode pre chodcov tak: { delay(3000); //Program čaká 3 sekundy digitalWrite(9,LOW); //Vypne zelenú LED diódu na križovatke digitalWrite(8,HIGH); //Zapne oranžovú LED diódu na križovatke delay(3000); //Program čaká 3 sekundy digitalWrite(8,LOW); //Vypne oranžovú LED diódu na križovatke digitalWrite(7,HIGH); //Zapne červenú LED diódu na križovatke delay(2000); //Program čaká 2 sekundy digitalWrite(5,LOW); //Vypne červenú LED diódu na prechode pre chodcov digitalWrite(4,HIGH); //Zapne zelenú LED diódu na prechode pre chodcov delay(5000); //Program čaká 5 sekúnd pokiaľ prejdú chodci cez zelenú { digitalWrite(4,LOW); //Vypne zelenú LED diódu na prechode pre chodcov digitalWrite(5,HIGH); //Zapne červenú LED diódu na prechode pre chodcov delay(500); //Program čaká 0,5 sekundy digitalWrite(8,HIGH); //Zapne oranžovú LED diódu na križovatke delay(1000); //Program čaká 1 sekundu digitalWrite(5,HIGH); //Zapne červenú LED diódu na prechode pre chodcov } delay(2000); //Program čaká 2 sekundy, potom zapne zelenú LED diódu na križovatke, zapne červenú LED diódu na prechode pre chodcov } else //Inak zapne zelenú LED diódu na križovatke a zapne aj červenú LED diódu na prechode pre chodcov a ostatné LED diódy vypne { digitalWrite(9, HIGH); //Zapne zelenú LED diódu na križovatke digitalWrite(5,HIGH); //Zapne červenú LED diódu na prechode pre chodcov digitalWrite(4,LOW); //Vypne zelenú LED diódu na prechode pre chodcov digitalWrite(8,LOW); //Vypne oranžovú LED diódu na križovatke digitalWrite(7,LOW); //Vypne červenú LED diódu na križovatke } } </pre>
<p>Hodnotenie (Spätná väzba):</p>	<p>Učiteľ vyhodnotí aktivitu žiakov počas vyučovacej hodiny.</p>
<p>Prílohy:</p>	<p>Príloha G. k metodickému listu č. 6</p>

Príloha H - Senzor plameňa naprogramovaný pomocou dosky Arduino



Príloha H. k metodickému listu č. 7 – Senzor plameňa naprogramovaný pomocou dosky Arduino

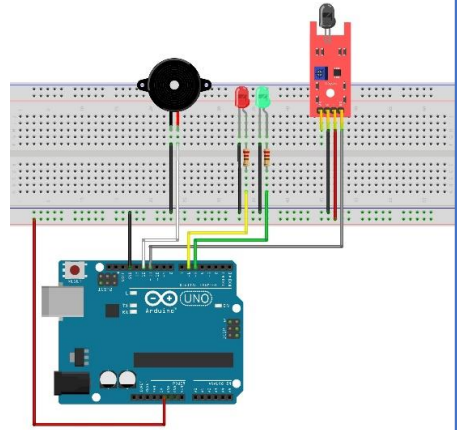
Detekcia plameňa

Infračervený senzor plameňa KY-026 deteguje plameň alebo podobné svetlo s vlnovou dĺžkou od 760 nm do 1100 nm. Rozsah detekčného uhla je približne 60 stupňov. Na module je možné ladiť citlivosť zabudovaným potenciometrom. Modul je schopný detekovať plameň zapalovača až na vzdialenosť 80cm.

Senzor obsahuje 4 pripojovacie piny:



- Arduino GND -> Modul G
- Arduino + 5V -> Modul +
- Arduino digital (D2-D13) -> Modul D0 -vysoký / nízky elektrický výstup signálu úrovne
- Arduino (A0-A6) -> modul A0 - výstupný signál termistorového napätia v reálnom čase



Pre viac informácií pozri Príloha B. k metodickému listu č. 1



Aktívny buzzer pre alarm alebo signalizáciu. Pri napájaní buzzer “pípa” frekvenciou približne 2300 Hz. V prípade, že hľadáte buzzer vytvárajúci jednoduchú melódiu, prejdite na pasívny buzzer.



Pre viac informácií pozri Príloha B. k metodickému listu č. 1

Časť z programu:

```
void loop()
{
  ohen = digitalRead(senzor_ohna);
  if (ohen== HIGH)           //Ak senzor detekuje oheň:

  {
    digitalWrite(bzučiak, HIGH); //Bzučiak pípa
    digitalWrite(led_cervena, HIGH); //Červená LED dióda svieti
    digitalWrite(led_zelena, LOW); //Zelená LED dióda nesvieti
  }
  else                       //Ak senzor nedetekuje oheň:
  {
    digitalWrite(bzučiak, LOW); //Bzučiak nepípa
    digitalWrite(led_zelena, HIGH); //Zelená LED dióda svieti
    digitalWrite(led_cervena, LOW); //Červená LED dióda nesvieti
  }
}
```

Metodický list číslo 7.

Názov témy:	Metodický list číslo 7. Senzor plameňa naprogramovaný pomocou dosky Arduino
Popis témy:	Detekcia plameňa pomocou senzora
Tematický celok	Podľa ISCED
Trvanie hodiny:	1 vyučovacia hodina
Cieľová skupina:	SŠ
Didaktické prostriedky:	Vývojová doska Arduino, Nepájivé kontaktné pole, káblíky, USB kábel, Senzor plameňa KY-026, Buzzer, LED dióda červená, LED dióda zelená, 2x Rezistor 220Ω, počítač, nainštalované programovacie prostredie ARDUINO IDE.
Vstupné znalosti:	<ul style="list-style-type: none">– Študenti poznajú vývojovú dosku Arduino a jeho komponenty– Základy programovania v programovacom jazyka pre Arduino
Cieľ hodiny:	Kognitívne ciele: <ul style="list-style-type: none">– Študent sa naučí programovať a pracovať so senzorom plameňa,– študent sa vie orientovať v prostredí Arduino IDE,

	<ul style="list-style-type: none"> – študent dokáže vytvárať a používať premenné v programe, – študent vie vytvárať vlastné cykly a algoritmy riešiacie problém. <p>Afektívne ciele:</p> <ul style="list-style-type: none"> – Študent vie rozpoznať, aká časť algoritmu sa má vykonať pred, počas a po skončení cyklu, – študent vie diskutovať o možnostiach použitia konkrétneho hardvéru.
Štruktúra vyučovacej hodiny:	<p>13. Úvod – oboznámenie sa s priebehom vyučovacej hodiny, rozdanie didaktických prostriedkov</p> <p>14. Riešenie úlohy</p> <p>15. Záverečná diskusia</p>
Všeobecnodidaktické metódy:	<ul style="list-style-type: none"> • Informačno-Receptívna • Reprodukčívna • Problémový výklad • Heuristická • Výskumná
Priebeh vyučovacej hodiny:	<p>V úvode hodiny je potrebné zopakovať si pravidlá pri zapájaní a programovaní vývojovej dosky Arduino. Vysvetlíme si princíp detegovania plameňa pomocou infračerveného senzora. Na senzore si ukážeme, kde je možné ladiť jeho citlivosť pomocou potenciometra a oboznámime sa s jeho zapojením. Zopakujeme si princíp fungovania použitých súčiastok. Riešime úlohu podľa zadania. Na konci hodiny zhodnotíme úspešnosť zadanej úlohy. Vytvoríme diskusiu o možno využití daného senzora v praxi. Pozbierame a skontrolujeme stav didaktických pomôcok.</p> <p>Aplikované funkcie: setup, loop, if, else, digitalWrite, digitalRead, pinMode, delay</p> <p>Zadanie úlohy: Vytvorte program v ktorom bude senzor ohňa detekovať plameň alebo podobné svetlo s vlnovou dĺžkou od 760 nm do 1100 nm. Naprogramujte program s nasledujúcimi požiadavkami:</p>

	<p>4. Červená LED dióda pripojená na PIN 6, Zelená LED dióda pripojená na PIN 5, Buzzer pripojený na PIN 12, Senzor plameňa modul D0 pripojený na PIN 11</p> <p>5. Ak senzor deteguje plameň vo svojej blízkosti, buzzer začne oznamovať prítomnosť ohňa alebo podobného svetla s vlnovou dĺžkou od 760 nm do 1100 nm zvukovou signalizáciu (pípanie) a svetelnou signalizáciu (svieti červená LED dióda).</p> <p>6. Nedetegovania ohňa alebo podobného svetla s vlnovou dĺžkou od 760 nm do 1100 nm je signalizované svetelnou signalizáciu (svieti zelená LED dióda).</p> <p>Riešenie:</p> <pre> const int bzučiak = 12; //Inicializácia bzučiaka na PIN 12 const int senzor_ohna = 11; //Inicializácia senzora na PIN 11 int led_cervena = 5; //Inicializácia červenej LED diódy na PIN 5 int led_zelena = 6; //Inicializácia zelenej LED diódy na PIN 6 int ohen = LOW; //Nastavenie senzora na stav - vypnutý void setup() { pinMode(bzučiak, OUTPUT); //Definuje bzučiak ako výstupné rozhranie pinMode(led_cervena, OUTPUT); //Definuje červenú LED diódu ako výstupné rozhranie pinMode(led_zelena, OUTPUT); //Definuje zelenú LED diódu ako výstupné rozhranie pinMode(senzor_ohna, INPUT); //Definuje senzor ako vstupné rozhranie Serial.begin(9600); } void loop() { ohen = digitalRead(senzor_ohna); if (ohen== HIGH) //Ak senzor detekuje oheň: { digitalWrite(bzučiak, HIGH); //Bzučiak pípa digitalWrite(led_cervena, HIGH); //Červená LED dióda svieti digitalWrite(led_zelena, LOW); //Zelená LED dióda nesvieti } else //Ak senzor nedetekuje oheň: { digitalWrite(bzučiak, LOW); //Bzučiak nepípa digitalWrite(led_zelena, HIGH); //Zelená LED dióda svieti digitalWrite(led_cervena, LOW); //Červená LED dióda nesvieti } } </pre>
Hodnotenie (Spätná väzba):	Učiteľ vyhodnotí aktivitu žiakov počas vyučovacej hodiny.
Prílohy:	Príloha H. k metodickému listu č. 7

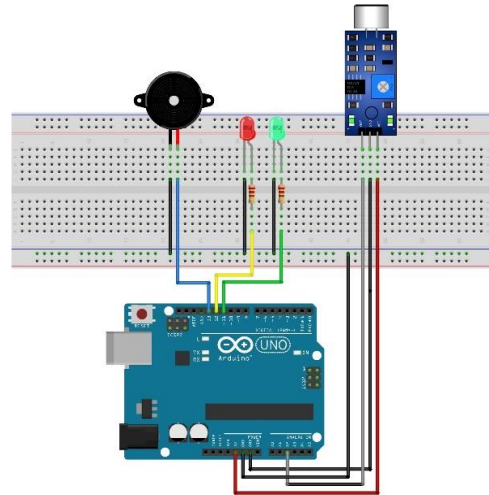
Príloha I – Senzor zvuku naprogramovaný pomocou dosky Arduino



Príloha I. k metodickému listu č. 8 – Senzor zvuku naprogramovaný pomocou dosky Arduino

Detekcia zvuku

Zvukový senzor ER-SPM22101S obsahuje mikrofón s výkonovým zosilňovačom LM386 3-5V. Na základe výkonového zosilňovača a elektretového mikrofónu sa môže použiť na zistenie, či existuje nejaký zvuk, ktorý presahuje prahovú hodnotu. Prahová hodnota sa dá nastaviť pomocou potenciometra. Doska pre zvukový senzor obsahuje indikačnú červenú LED diódu, ktorá sa rozsvieti pri hluku, ktorá presahuje nastavenú hranicu.



Senzor obsahuje 3 pripojovacie piny:



- Arduino GND -> Modul GND
- Arduino + 5V -> Modul +
- Arduino digital (D2-D13) -> Modul OUT -digitálny výstup signálu



Pre viac informácií pozri Príloha B. k metodickému listu č. 1



Pre viac informácií pozri Príloha C. k metodickému listu č. 2



Pre viac informácií pozri Príloha B. k metodickému listu č. 1

Časť z programu:

```
void loop() {
  int zvukove_hodnoty=analogRead(senzor_zvuku); //číta zvukové hodnoty zo senzora
  if (zvukove_hodnoty>prahova_hodnota) //Ak sú zvukové hodnoty väčšie alebo rovné prahovej hodnote:
  {
    digitalWrite(bzuciak,HIGH); //Bzučiak pípa
    digitalWrite(led_cervena,HIGH); //červená LED dióda svieti
    digitalWrite(led_zelena,LOW); //Zelená LED dióda nesvieti
    delay(1000); //Program čaká 1 sekundu
  }
  else //Ak nie sú zvukové hodnoty väčšie alebo rovné prahovej hodnote:
  {
    digitalWrite(bzuciak,LOW); //Bzučiak nepípa
    digitalWrite(led_cervena,LOW); //červená LED dióda nesvieti
    digitalWrite(led_zelena,HIGH); //Zelená LED dióda svieti
  }
}
```

Metodický list číslo 8.

Názov témy:	Metodický list číslo 8. Senzor zvuku
Popis témy:	Detekcia zvuku pomocou senzora
Tematický celok	Podľa ISCED
Trvanie hodiny:	1 vyučovací hodina
Cieľová skupina:	SŠ
Didaktické prostriedky	Vývojová doska Arduino, Nepájivé kontaktné pole, káblíky, USB kábel, LED dióda červená, LED dióda zelená, 2x Rezistor 220Ω, Buzzer, počítač, nainštalované programovacie prostredie Arduino IDE.
Vstupné znalosti:	<ul style="list-style-type: none">– Študenti poznajú vývojovú dosku Arduino a jeho komponenty– Základy programovania v programovacom jazyka pre Arduino
Cieľ hodiny:	Kognitívne ciele: <ul style="list-style-type: none">– Študent sa naučí programovať a pracovať so senzorom zvuku,– študent sa vie orientovať v prostredí Arduino IDE,– študent dokáže vytvárať a používať premenné v programe,– študent vie vytvárať vlastné cykly a algoritmy riešiacie problém.

	<p>Afektívne ciele:</p> <ul style="list-style-type: none"> – Študent vie rozpoznať, aká časť algoritmu sa má vykonať pred, počas a po skončení cyklu, – študent vie diskutovať o možnostiach použitia konkrétneho hardvéru.
Štruktúra vyučovacej hodiny	<p>16. Úvod – oboznámenie sa s priebehom vyučovacej hodiny, rozdanie didaktických prostriedkov</p> <p>17. Riešenie úlohy</p> <p>18. Záverečná diskusia</p>
Všeobecnodidaktické metódy	<ul style="list-style-type: none"> • Informačno-Receptívna • Reproductívna • Problémový výklad • Heuristická • Výskumná
Priebeh vyučovacej hodiny:	<p>V úvode hodiny je potrebné zopakovať si pravidlá pri zapájaní a programovaní vývojovej dosky Arduino. Vysvetlíme si princíp detegovania zvuku pomocou zvukového senzora. Na senzore si ukážeme, kde je možné ladiť jeho citlivosť pomocou potenciometra a oboznámime sa s jeho zapojením. Zopakujeme si princíp fungovania použitých súčiastok. Riešime úlohu podľa zadania. Na konci hodiny zhodnotíme úspešnosť zadanej úlohy. Vytvoríme diskusiu o možno využití daného senzora v praxi. Pozbierame a skontrolujeme stav didaktických pomôcok.</p> <p>Aplikované funkcie: setup, loop, if, else, digitalWrite, analogRead, pinMode, delay</p> <p>Zadanie úlohy: Vytvorte program, v ktorom bude senzor zvuku detegovať zvukové hodnoty. Naprogramujte program s nasledujúcimi požiadavkami:</p> <ol style="list-style-type: none"> 1. Červená LED dióda pripojená na PIN 12, Zelená LED dióda pripojená na PIN 11, Buzzer pripojený na PIN 13, Senzor zvuku modul OUT pripojený na PIN A2

	<p>2. Ak senzor deteguje zvuk vo svojej blízkosti, buzzer začne oznamovať prítomnosť hluku zvukovou signalizáciou (pípanie) a svetelnou signalizáciou (svieti červená LED dióda).</p> <p>3. V prípade nedetegovania zvuku je vysielaný signál svetelnou signalizáciou (svieti zelená LED dióda).</p> <p>Riešenie:</p> <pre> const int bzuciak = 13; //inicializácia bzučiaka na PIN 13 const int senzor_zvuku = A2; // inicializácia senzora na PIN A2 const int prahova_hodnota = 600; // definovanie prahovej hodnoty int led_cervena = 12; //inicializácia červenej LED diódy na PIN 12 int led_zelena = 11; //inicializácia zelenej LED diódy na PIN 11 void setup() { pinMode(led_cervena,OUTPUT); //definuje červenú LED diódu ako výstupné rozhranie pinMode(led_zelena,OUTPUT); //definuje zelenú LED diódu ako výstupné rozhranie pinMode(senzor_zvuku,INPUT); //definuje senzor ako vstupné rozhranie pinMode(bzuciak, OUTPUT); //definuje bzučiak ako výstupné rozhranie } void loop() { int zvukove_hodnoty=analogRead(senzor_zvuku); //číta zvukové hodnoty zo senzora if (zvukove_hodnoty>=prahova_hodnota) //Ak sú zvukové hodnoty väčšie alebo rovné prahovej hodnote: { digitalWrite(bzuciak,HIGH); //Bzučiak pípa digitalWrite(led_cervena,HIGH); //Červená LED dióda svieti digitalWrite(led_zelena,LOW); //Zelená LED dióda nesvieti delay(1000); //Program čaká 1 sekundu } else //Ak nie sú zvukové hodnoty väčšie alebo rovné prahovej hodnote: { digitalWrite(bzuciak,LOW); //Bzučiak nepípa digitalWrite(led_cervena,LOW); //Červená LED dióda nesvieti digitalWrite(led_zelena,HIGH); //Zelená LED dióda svieti } } </pre>
Hodnotenie (Spätná väzba):	Učiteľ vyhodnotí aktivitu žiakov počas vyučovacej hodiny.
Prílohy:	Príloha I. k metodickému listu č. 8

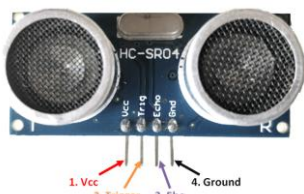
Príloha J – Parkovací asistent naprogramovaný pomocou dosky Arduino



Príloha J. k metodickému listu č. 9 – Parkovací asistent naprogramovaný pomocou dosky Arduino

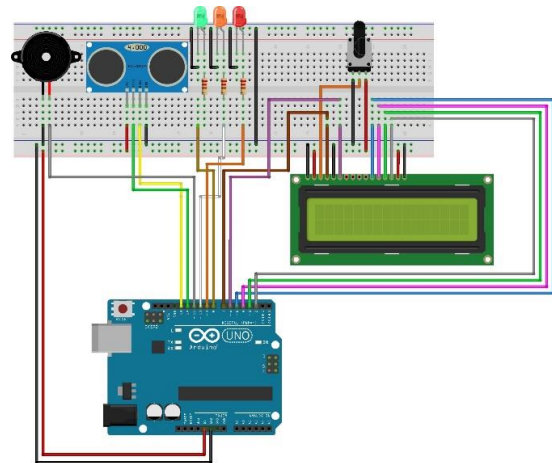
Detekcia vzdialenosti pomocou ultrazvukového senzora

Ultrazvukový senzor HC-SR04 slúži na meranie vzdialenosti. Detekčná vzdialenosť je od 1 cm do 4,5 m s presnosťou na 0,3 cm. Pozorovací uhol je do 15°.



Senzor obsahuje 4 pripojovacie piny:

- Arduino +5V -> Modul Vcc
- Arduino GND -> Modul Gnd
- Arduino digital (D2-D13) -> Modul Trigger - vysiela signál (vysokofrekvenčný zvuk)
- Arduino (A0-A6) -> Modul Echo - produkuje impulz len vtedy keď je prijatý odrazený signál



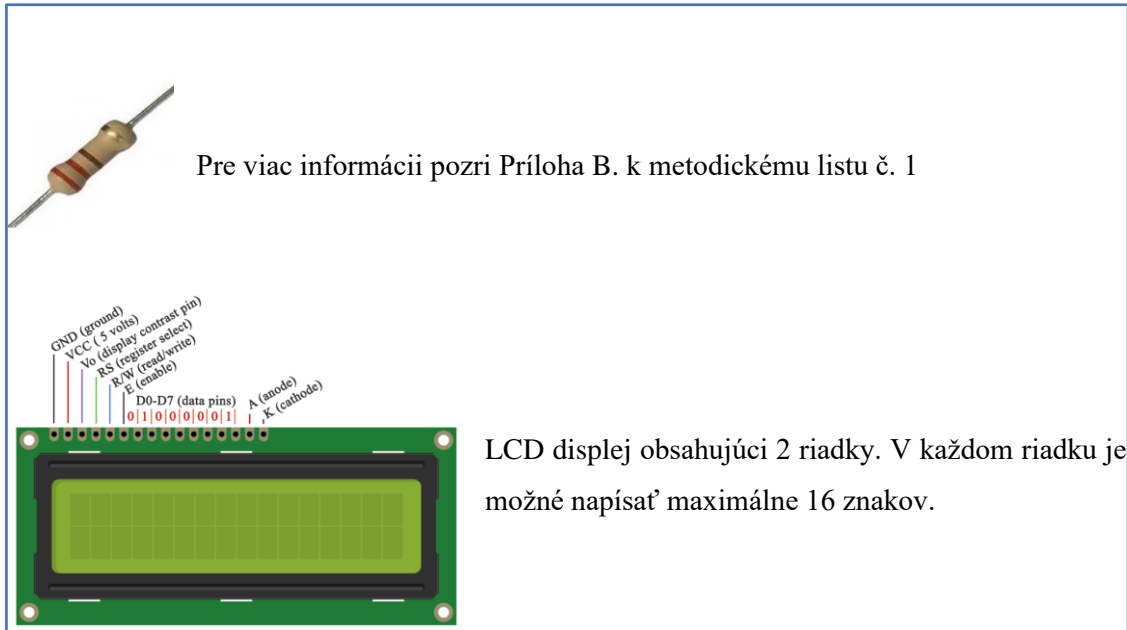
Potenciometer je elektrická súčiastka - regulovateľný delič napätia.



Pre viac informácií pozri Príloha C. k metodickému listu č. 2



Pre viac informácií pozri Príloha B. k metodickému listu č. 1



Časť z programu:

```

if (vzdialenost < 10) //Ak vzdialenost je mensia ako 10
{
  Serial.print("OHROZENIE ");
  Serial.print(vzdialenost); //Zobrazuje na LCD displeji spravu OHROZENIE + vzdialenost v cm
  Serial.println(" cm");
  digitalWrite(led_cervena, HIGH); //Cervená LED dióda svieti
  digitalWrite(led_zelena, LOW); //Zelená LED dióda nesvieti
  digitalWrite(led_oranzova, LOW); //Oranžová LED dióda nesvieti
  digitalWrite(bzuciak, HIGH); //Bzučiak pípa
  delay(30); //Pauza 30ms
  digitalWrite(bzuciak, LOW); //Bzučiak nepípa
  delay(30); //Pauza 30ms
  lcd.clear(); //Vymaže obrazovku LCD displeja a umiestni kurzor do ľavého horného rohu
  lcd.print("OHROZENIE ");
  lcd.print(vzdialenost); //Zobrazuje na LCD displeji spravu OHROZENIE + vzdialenost v cm
  lcd.print(" cm");
  delay(100); //Pauza 100ms
}

```

Metodický list číslo 9.

Názov témy:	Metodický list číslo 9. Parkovací asistent naprogramovaný pomocou dosky Arduino
Popis témy:	Detekcia vzdialenosti pomocou ultrazvukového senzora
Tematický celok	Podľa ISCED
Trvanie hodiny:	1 vyučovacia hodina
Cieľová skupina:	SŠ
Didaktické prostriedky	Vývojová doska Arduino, Nepájivé kontaktné pole, káblíky, USB kábel, LCD displej 16x2, Potenciometer 10kΩ, LED dióda červená, LED dióda zelená, LED dióda oranžová, 3x Rezistor

	220Ω, Buzzer, Ultrazvukový senzor vzdialenosti HC-SR04, počítač, nainštalované programovacie prostredie Arduino IDE.
Vstupné znalosti:	<ul style="list-style-type: none"> – Študenti poznajú vývojovú dosku Arduino a jeho komponenty – Základy programovania v programovacom jazyka pre Arduino
Cieľ hodiny:	<p>Kognitívne ciele:</p> <ul style="list-style-type: none"> – Študent sa naučí programovať a pracovať so senzorom vzdialenosti, LCD displejom, – študent sa vie orientovať v prostredí Arduino IDE, – študent dokáže vytvárať a používať premenné v programe, – študent vie vytvárať vlastné cykly a algoritmy riešiace problém. <p>Afektívne ciele:</p> <ul style="list-style-type: none"> – Študent vie rozpoznať, aká časť algoritmu sa má vykonať pred, počas a po skončení cyklu, – študent vie diskutovať o možnostiach použitia konkrétneho hardvéru.
Štruktúra vyučovacej hodiny	<p>19. Úvod – oboznámenie sa s priebehom vyučovacej hodiny, rozdanie didaktických prostriedkov</p> <p>20. Riešenie úlohy</p> <p>21. Záverečná diskusia</p>
Všeobecnodidaktické metódy	<ul style="list-style-type: none"> • Informačno-Receptívna • Reprodukčná • Problémový výklad • Heuristická • Výskumná
Priebeh vyučovacej hodiny:	V úvode hodiny je potrebné zopakovať si pravidlá pri zapájaní a programovaní vývojovej dosky Arduino. Vysvetlíme si systém správy parkovacieho asistenta. Objasníme si princíp fungovania použitých súčiastok. Riešime úlohu podľa zadania. Na konci hodiny zhodnotíme úspešnosť zadanej úlohy. Vytvoríme

	<p>diskusiu o možno využití daného zapojenia v praxi. Pozbierame a skontrolujeme stav didaktických pomôcok.</p> <p>Aplikované funkcie: setup, loop, Serial.begin, lcd.begin, lcd.clear, pulseIn, Serial.print, Serial.println, if, else, else if, digitalWrite, pinMode, delay</p> <p>Zadanie úlohy: Vytvorte program, v ktorom budú LED diódy, buzzer, potenciometer, ultrazvukový senzor vzdialenosti a LCD displej (16x2) modelovať parkovací asistent. Naprogramujte program s nasledujúcimi požiadavkami:</p> <p>5. LCD displej je pripojená na PIN s číslami: 7, 6, 5, 4, 3, 2. Trigger senzora pripojený na PIN 12, Echo senzora pripojený na PIN 13, Buzzer pripojený na PIN 11, zelená LED dióda pripojená na PIN 8, červená LED dióda pripojená na PIN 9, oranžová LED dióda pripojená na PIN 10</p> <p>6. Ak je vzdialenosť senzora od objektu menšia ako 10 cm tak:</p> <ol style="list-style-type: none"> Zobraz na LCD displeji správu "OHROZENIE" + vzdialenosť senzora od objektu v cm, Zapni červenú LED diódu, vypni zelenú a oranžovú LED diódu, Zvuková signalizácia je striedavo zapínaná a vypínaná v intervale 30ms, Opakuj cyklus každých 100ms. <p>7. Ak je vzdialenosť senzora od objektu väčšia alebo rovná ako 10cm a zároveň menšia alebo rovná ako 20cm:</p> <ol style="list-style-type: none"> Zobraz na LCD displeji správu "SPOMAL" + vzdialenosť senzora od objektu v cm, Zapni oranžovú LED diódu, vypni zelenú a červenú LED diódu, Zvuková signalizácia je striedavo zapínaná a vypínaná v intervale 60ms, Opakuj cyklus každých 100ms.
--	--

8. Ak je vzdialenosť senzora od objektu väčšia ako 20cm:

- Zobraz na LCD displeji správu "OK" + vzdialenosť senzora od objektu v cm,
- Zapni zelenú LED diódu, vypni oranžovú a červenú LED diódu,
- Zvuková signalizácia nevydáva žiaden zvukový signál,
- Opakuj cyklus každých 100ms.

Riešenie:

```
#include<LiquidCrystal.h> //Knižnica pre LCD displej
LiquidCrystal lcd(7, 6, 5, 4, 3, 2); //Inicializácia LCD displeja na PINY: 7, 6, 5, 4, 3, 2

int trigger = 12; //Inicializácia triggera na PIN 12
int echo = 13; //Inicializácia echo na PIN 13

int buzciak = 11; //Inicializácia bzučička na PIN 11

int led_zelena = 8; //Inicializácia zelenej LED diódy na PIN 8
int led_cervena = 9; //Inicializácia červenej LED diódy na PIN 9
int led_oranzova = 10; //Inicializácia oranžovej LED diódy na PIN 10

void setup()
{
  pinMode(trigger, OUTPUT); //Definuje trigger ako výstupné rozhranie
  pinMode(echo, INPUT); //Definuje echo ako vstupné rozhranie
  pinMode(buzciak, OUTPUT); //Definuje bzučičak ako výstupné rozhranie

  pinMode(led_zelena, OUTPUT); //Definuje zelenú LED diódu ako výstupné rozhranie
  pinMode(led_cervena, OUTPUT); //Definuje červenú LED diódu ako výstupné rozhranie
  pinMode(led_oranzova, OUTPUT); //Definuje oranžovú LED diódu ako výstupné rozhranie

  Serial.begin(9600);
  lcd.begin(16, 2); //Nastavenie LCD displeja (počet stĺpcov a riadkov)
  lcd.clear(); //Vymaže obrazovku LCD displeja a umiestni kurzor do ľavého horného rohu
}

void loop()
{
  long trvanie, vzdialenost; //Premenné typu long
  digitalWrite(trigger, LOW); //Nevyslela signál (vysokofrekvenčný zvuk)
  delayMicroseconds(10); //Pauza 10ms
  digitalWrite(trigger, HIGH); //Vyslela signál (vysokofrekvenčný zvuk)
  delayMicroseconds(10); //Pauza 10ms
  digitalWrite(trigger, LOW); //Nevyslela signál (vysokofrekvenčný zvuk)
  trvanie = pulseIn(echo, HIGH); //Produkuje impulz len vtedy keď je prijatý odrazený signál a vloží ho do premennej trvanie
  vzdialenost = (trvanie / 58); //Vzorec na výpočet vzdialenosti

  if (vzdialenost < 10) //Ak vzdialenost je menšia ako 10
  {
    Serial.print("OHROZENIE ");
    Serial.print(vzdialenost); //Zobrazuje na LCD displeji správu OHROZENIE + vzdialenost v cm
    Serial.println(" cm");
    digitalWrite(led_cervena, HIGH); //Červená LED dióda svieti
    digitalWrite(led_zelena, LOW); //Zelená LED dióda nesvieti
    digitalWrite(led_oranzova, LOW); //Oranžová LED dióda nesvieti
    digitalWrite(buzciak, HIGH); //Bzučičak pípa
    delay(30); //Pauza 30ms
    digitalWrite(buzciak, LOW); //Bzučičak nepípa
    delay(30); //Pauza 30ms
    lcd.clear(); //Vymaže obrazovku LCD displeja a umiestni kurzor do ľavého horného rohu
    lcd.print("OHROZENIE ");
    lcd.print(vzdialenost); //Zobrazuje na LCD displeji správu OHROZENIE + vzdialenost v cm
    lcd.print(" cm");
    delay(100); //Pauza 100ms
  }
  else if ((vzdialenost >= 10) && (vzdialenost <= 20)) //Ak vzdialenost je väčšia alebo rovná ako 10 a zároveň menšia alebo rovná ako 20
  {
    Serial.print("SPOMAL ");
    Serial.print(vzdialenost); //Zobrazuje na LCD displeji správu SPOMAL + vzdialenost v cm
    Serial.println(" cm");
    digitalWrite(led_oranzova, HIGH); //Oranžová LED dióda svieti
    digitalWrite(led_zelena, LOW); //Zelená LED dióda nesvieti
    digitalWrite(led_cervena, LOW); //Červená LED dióda nesvieti
    digitalWrite(buzciak, HIGH); //Bzučičak pípa
    delay(60); //Pauza 60ms
    digitalWrite(buzciak, LOW); //Bzučičak nepípa
    delay(60); //Pauza 60ms
    lcd.clear(); //Vymaže obrazovku LCD displeja a umiestni kurzor do ľavého horného rohu
    lcd.print("SPOMAL ");
    lcd.print(vzdialenost); //Zobrazuje na LCD displeji správu SPOMAL + vzdialenost v cm
    lcd.print(" cm");
    delay(100); //Pauza 100ms
  }
  else
  {
    Serial.print("OK ");
    Serial.print(vzdialenost); //Zobrazuje na LCD displeji správu OK + vzdialenost v cm
    Serial.println(" cm");
    digitalWrite(led_zelena, HIGH); //Zelená LED dióda svieti
    digitalWrite(led_cervena, LOW); //Červená LED dióda nesvieti
    digitalWrite(led_oranzova, LOW); //Oranžová LED dióda nesvieti
    digitalWrite(buzciak, LOW); //Bzučičak nepípa
    lcd.clear(); //Vymaže obrazovku LCD displeja a umiestni kurzor do ľavého horného rohu
    lcd.print("OK ");
    lcd.print(vzdialenost); //Zobrazuje na LCD displeji správu OK + vzdialenost v cm
    lcd.print(" cm");
    delay(100); //Pauza 100ms
  }
}
```


Hodnotenie (Spätná väzba):

Učiteľ vyhodnotí aktivitu žiakov počas vyučovacej hodiny.

Prílohy:

Príloha J. k metodickému listu č. 9

Príloha K – 7 – segmentový displej naprogramovaný pomocou dosky Arduino

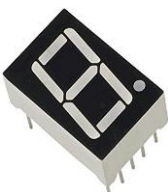


ARDUINO

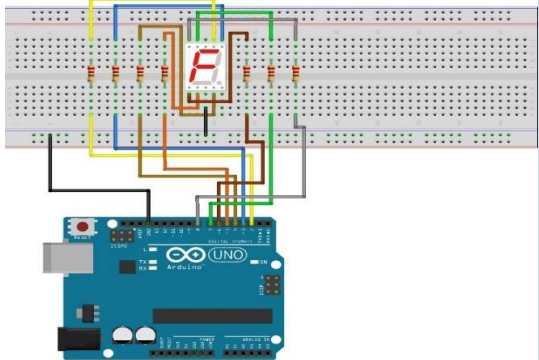
Príloha K. k metodickému listu č. 10 – 7- segmentový displej naprogramovaný pomocou dosky Arduino

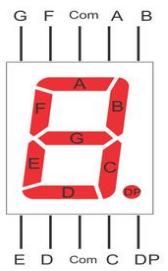
Počítadlo cez 7-segmentový displej

Cieľom tohto pracovného listu bude vytvoriť odpočítavací časovač riadený pomocou BBC Micro:bitu, ktorý bude zobrazovať čísla na obrazovke pomocou 7-segmentového displeja.




Sedem segmentový displejov je bežný spôsob zobrazovania čísel a často používa pre: hodiny, kalkulačky a elektronických meracích prístrojoch.





7-segmentové displeje sa zvyčajne skladajú zo siedmich jednotlivých farebných LED nazývaných (segmenty). Každý segment je možné zapnúť alebo vypnúť a vytvoriť tak jedinečný vzor. Každý segment je označený písmenom od A až po G.



Pre viac informácií pozri Príloha B. k metodickému listu č. 1

Časť z programu:

```
void loop() {
for (int i = 0; i < 10; i++) { //cyklus na výpis čísiel od 0 po 9
switch(i) { //podmienka

case 0: //hodnota 0
digitalWrite(segment_a,HIGH); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteInosti daných čísel
digitalWrite(segment_b,HIGH);
digitalWrite(segment_c,HIGH);
digitalWrite(segment_d,HIGH);
digitalWrite(segment_e,HIGH);
digitalWrite(segment_f,HIGH);
digitalWrite(segment_g,LOW);
break;

case 1: //hodnota 1
digitalWrite(segment_a,LOW); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteInosti daných čísel
digitalWrite(segment_b,HIGH);
digitalWrite(segment_c,HIGH);
digitalWrite(segment_d,LOW); //celý case 1 slúži pre vysvietenie číslice 1
digitalWrite(segment_e,LOW);
digitalWrite(segment_f,LOW);
digitalWrite(segment_g,LOW);
break;
}
```

Metodický list číslo 10.

Názov témy:	Metodický list číslo 10. 7-segmentový displej naprogramovaný pomocou dosky Arduino
Popis témy:	Počítadlo cez 7-segmentový displej
Tematický celok	Podľa ISCED
Trvanie hodiny:	1 vyučovacia hodina
Cieľová skupina:	SŠ
Didaktické prostriedky	Vývojová doska Arduino, Nepájivé kontaktné pole, káblíky, USB kábel, 7-segmentový displej, 7x Rezistor 220Ω, počítač, nainštalované programovacie prostredie Arduino IDE.
Vstupné znalosti:	<ul style="list-style-type: none">– Študenti poznajú vývojovú dosku Arduino a jeho komponenty– Základy programovania v programovacom jazyka pre Arduino
Cieľ hodiny:	Kognitívne ciele: <ul style="list-style-type: none">– Študent sa naučí programovať a pracovať so 7-segmentovým displejom,– študent sa vie orientovať v prostredí Arduino IDE,– študent dokáže vytvárať a používať premenné v programe,

	<p>– študent vie vytvárať vlastné cykly a algoritmy riešiacie problém.</p> <p>Afektívne ciele:</p> <p>– Študent vie rozpoznať, aká časť algoritmu sa má vykonať pred, počas a po skončení cyklu,</p> <p>– študent vie diskutovať o možnostiach použitia konkrétneho hardvéru.</p>
Štruktúra vyučovacej hodiny	<p>22. Úvod – oboznámenie sa s priebehom vyučovacej hodiny, rozdanie didaktických prostriedkov</p> <p>23. Riešenie úlohy</p> <p>24. Záverečná diskusia</p>
Všeobecnodidaktické metódy	<ul style="list-style-type: none"> • Informačno-Receptívna • Reprodukčívna • Problémový výklad • Heuristická • Výskumná
Priebeh vyučovacej hodiny:	<p>V úvode hodiny je potrebné zopakovať si pravidlá pri zapájaní a programovaní vývojovej dosky Arduino. Vysvetlíme si princíp zobrazovania segmentov na displeji. Oboznámime sa s jeho zapojením. Zopakujeme si princíp fungovania použitých súčiastok. Riešime úlohu podľa zadania. Na konci hodiny zhodnotíme úspešnosť zadanej úlohy. Vytvoríme diskusiu o možno využití daného senzora v praxi. Pozbierame a skontrolujeme stav didaktických pomôcok.</p> <p>Aplikované funkcie: setup, loop, for, switch, case, digitalWrite, pinMode, delay</p> <p>Zadanie úlohy: Vytvorte program, v ktorom bude 7-segmentový displej postupne zobrazovať číslice od 0 po 9. Program sa bude opakovať n-krát. Naprogramujte program s nasledujúcimi požiadavkami:</p> <ol style="list-style-type: none"> 1. Segment A pripojený na PIN 2, Segment B pripojený na PIN 3, Segment C pripojený na PIN 4, Segment D pripojený na PIN 5, Segment E pripojený na PIN 6,

Segment F pripojený na PIN 7, Segment G pripojený na PIN 8, Segment COM pripojený na PIN GND

Riešenie:

```
int segment_a = 2; //inicializácia segmentu A
int segment_b = 3; //inicializácia segmentu B
int segment_c = 4; //inicializácia segmentu C
int segment_d = 5; //inicializácia segmentu D
int segment_e = 6; //inicializácia segmentu E
int segment_f = 7; //inicializácia segmentu F
int segment_g = 8; //inicializácia segmentu G

void setup() {
  pinMode(segment_a,OUTPUT); //definuje segment A ako výstupné rozhranie
  pinMode(segment_b,OUTPUT); //definuje segment B ako výstupné rozhranie
  pinMode(segment_c,OUTPUT); //definuje segment C ako výstupné rozhranie
  pinMode(segment_d,OUTPUT); //definuje segment D ako výstupné rozhranie
  pinMode(segment_e,OUTPUT); //definuje segment E ako výstupné rozhranie
  pinMode(segment_f,OUTPUT); //definuje segment F ako výstupné rozhranie
  pinMode(segment_g,OUTPUT); //definuje segment G ako výstupné rozhranie
}

void loop() {
  for (int i = 0; i < 10; i++) { //cyklus na výpis číslíc od 0 po 9
    switch(i) { //podmienka
      case 0: //hodnota 0
        digitalWrite(segment_a,HIGH); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteľnosti daných čísel
        digitalWrite(segment_b,HIGH);
        digitalWrite(segment_c,HIGH);
        digitalWrite(segment_d,HIGH);
        digitalWrite(segment_e,HIGH);
        digitalWrite(segment_f,HIGH);
        digitalWrite(segment_g,LOW);
        break;
      case 1: //hodnota 1
        digitalWrite(segment_a,LOW); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteľnosti daných čísel
        digitalWrite(segment_b,HIGH);
        digitalWrite(segment_c,HIGH);
        digitalWrite(segment_d,LOW); //celý case 1 slúži pre vysvietenie čísllice 1
        digitalWrite(segment_e,LOW);
        digitalWrite(segment_f,LOW);
        digitalWrite(segment_g,LOW);
        break;
      case 2: //hodnota 2
        digitalWrite(segment_a,HIGH); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteľnosti daných čísel
        digitalWrite(segment_b,HIGH);
        digitalWrite(segment_c,LOW);
        digitalWrite(segment_d,HIGH); //celý case 2 slúži pre vysvietenie čísllice 1
        digitalWrite(segment_e,HIGH);
        digitalWrite(segment_f,LOW);
        digitalWrite(segment_g,HIGH);
        break;
      case 3: //hodnota 3
        digitalWrite(segment_a,HIGH); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteľnosti daných čísel
        digitalWrite(segment_b,HIGH);
        digitalWrite(segment_c,HIGH);
        digitalWrite(segment_d,HIGH); //celý case 3 slúži pre vysvietenie čísllice 3
        digitalWrite(segment_e,LOW);
        digitalWrite(segment_f,LOW);
        digitalWrite(segment_g,HIGH);
        break;
      case 4: //hodnota 4
        digitalWrite(segment_a,LOW); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteľnosti daných čísel
        digitalWrite(segment_b,HIGH);
        digitalWrite(segment_c,HIGH);
        digitalWrite(segment_d,LOW); //celý case 4 slúži pre vysvietenie čísllice 4
        digitalWrite(segment_e,LOW);
        digitalWrite(segment_f,HIGH);
        digitalWrite(segment_g,HIGH);
        break;
      case 5: //hodnota 5
        digitalWrite(segment_a,HIGH); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteľnosti daných čísel
        digitalWrite(segment_b,LOW);
        digitalWrite(segment_c,HIGH);
        digitalWrite(segment_d,HIGH);
        digitalWrite(segment_e,LOW); //celý case 5 slúži pre vysvietenie čísllice 5
        digitalWrite(segment_f,HIGH);
        digitalWrite(segment_g,HIGH);
        break;
      case 6: //hodnota 6
        digitalWrite(segment_a,HIGH); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteľnosti daných čísel
        digitalWrite(segment_b,HIGH);
        digitalWrite(segment_c,HIGH);
        digitalWrite(segment_d,HIGH); //celý case 6 slúži pre vysvietenie čísllice 6
        digitalWrite(segment_e,HIGH);
        digitalWrite(segment_f,HIGH);
        digitalWrite(segment_g,HIGH);
        break;
      case 7: //hodnota 7
        digitalWrite(segment_a,HIGH); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteľnosti daných čísel
        digitalWrite(segment_b,HIGH);
        digitalWrite(segment_c,HIGH);
        digitalWrite(segment_d,LOW);
        digitalWrite(segment_e,LOW); //celý case 7 slúži pre vysvietenie čísllice 7
        digitalWrite(segment_f,LOW);
        digitalWrite(segment_g,LOW);
        break;
      case 8: //hodnota 8
        digitalWrite(segment_a,HIGH); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteľnosti daných čísel
        digitalWrite(segment_b,HIGH);
        digitalWrite(segment_c,HIGH);
        digitalWrite(segment_d,HIGH);
        digitalWrite(segment_e,HIGH); //celý case 8 slúži pre vysvietenie čísllice 8
        digitalWrite(segment_f,HIGH);
        digitalWrite(segment_g,HIGH);
        break;
      case 9: //hodnota 9
        digitalWrite(segment_a,HIGH); //nastavenie určitých segmentov na HIGH a LOW, kvôli zobraziteľnosti daných čísel
        digitalWrite(segment_b,HIGH);
        digitalWrite(segment_c,HIGH);
        digitalWrite(segment_d,HIGH);
        digitalWrite(segment_e,LOW); //celý case 9 slúži pre vysvietenie čísllice 9
        digitalWrite(segment_f,HIGH);
        digitalWrite(segment_g,HIGH);
        break;
    }
    delay(1000); //oneskorenie pre vykonanie ďalšieho príkazu - 1sekunda
  }
}
```

Hodnotenie (Spätná väzba):	Učiteľ vyhodnotí aktivitu žiakov počas vyučovacej hodiny.
Prílohy:	Príloha K. k metodickému listu č. 10