

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE

FAKULTA CHEMICKEJ A POTRAVINÁRSKEJ TECHNOLOGIE

Ústav informatizácie, automatizácie a matematiky

Oddelenie informatizácie a riadenia procesov



RIADENIE ROBOTOV LEGO MINDSTORMS

Vypracoval: Robert Taraba

Školiteľ: Ing. Michal Kvasnica, PhD.

Bratislava 2008

ZADANIE PRÁCE

Ďakujem vedúcemu projektu Ing.
Michalovi Kvasnicovi, PhD. za jeho
čas, pripomienky a pomoc pri
vypracovaní tohto projektu.

Abstrakt

Táto práca rozoberá problematiku riadenia robotov stavebnice LEGO Mindstorms. Hlavnými cieľmi tejto práce bolo preskúmanie možností riadenia robotov, ich využitie pri výučbe a taktiež vytvorenie niekoľkých samostatných úloh, ktorých výsledným prepojením vznikla hlavná úloha. Počas plnenia jednotlivých úloh využívajú roboti všetky dostupné senzory, akčné členy a vzájomnú koordináciu. Práca taktiež analyzuje činnosti svetelných, tlakových, ultrazvukových a zvukových senzorov ako aj činnosti servomotorov a hlavnej riadiacej kocky, ktorá slúži na riadenie robotov. Časť práce sa zaoberá programovaním robotov pomocou programu MINDSTORMS NXT. Výsledkom práce je podrobný popis vypracovania jednotlivých úloh, ktoré môžu slúžiť ako návod na laboratórne cvičenia k predmetom spätých s témou práce.

Abstract

This work deals with control of LEGO Mindstorms robots. The main goal were to investigate, whether this technology is suitable to be used in the teaching processes, and if so, to create a set of independent exercises involving construction and programming of the robots. The robots can use sensors, actuators, and communication channels to fulfill individual tasks. This work also provides the description of individual sensors (e.g. light, pressure, ultrasonic, and sound sensors), as well as the description of available actuators. In addition, this work can also serve as a guide to programming of LEGO robots. The outcome is a detailed description of individual exercises, along with the solutions.

OBSAH

ÚVOD.....	7
1. TECHNICKÉ INFORMÁCIE.....	8
1.1 Riadiaca kocka NXT.....	8
1.2 Senzory.....	10
1.2.1 Dotykový senzor.....	10
1.2.2 Senzor zvuku.....	10
1.2.3 Svetelný senzor.....	11
1.2.4 Ultrazvukový senzor.....	11
1.3 Interaktívny servomotor.....	12
1.4 Bluetooth komunikácia.....	13
2. SOFTWARE.....	14
2.1 Programovacia paleta.....	15
2.1.1 Obecná paleta.....	15
2.1.2 Kompletná paleta.....	16
2.1.3 Vlastná paleta.....	18
2.2 Konfiguračný panel.....	18
2.3 Panel riadenia.....	19
3. PRAKTICKÁ ČASŤ.....	20
3.1 Dopravný vozík (podávač loptičiek).....	21
3.1.1 Úloha 1 pre dopravný vozík.....	22
3.1.2 Úloha 2 pre dopravný vozík.....	24
3.1.3 Úloha 3 pre dopravný vozík.....	28
3.1.4 Úloha 4 pre dopravný vozík.....	31
3.1.5 Úloha 5 pre dopravný vozík.....	36
3.2 Robotické rameno (triedič loptičiek).....	42
3.2.1 Úloha 1 pre robotické rameno.....	43
3.2.2 Úloha 2 pre robotické rameno.....	47
3.2.3 Úloha 3 pre robotické rameno.....	50
3.2.4 Úloha 4 pre robotické rameno.....	55

3.2.5 Úloha 5 pre robotické rameno.....	61
3.2.5.1 Úloha 5.1 pre robotické rameno.....	70
3.2.5.2 Úloha 5.2 pre robotické rameno.....	71
3.2.5.3 Úloha 5.3 pre robotické rameno.....	72
3.3 Hlavná úloha (Triediaca linka).....	75
3.3.1 Program pre dopravný vozík (podávač loptičiek).....	77
3.3.2 Program pre robotické rameno (triedič loptičiek).....	83
ZÁVER.....	93
LITERATÚRA.....	94

ÚVOD

Lego Mindstorms je novou generáciou robotických systémov Lego, určených k podpore vzdelávania. Žiakov a študentov uvádza do sveta vedy a techniky, pomáha im v štúdiu základných zákonitostí prírodných vied vrátane matematiky, informatiky, fyziky a to formou zábavných činností.

Lego systém umožňuje študentom navrhovať, modelovať a testovať robotov na základe ich predstavivosti, šikovnosti a vedomostí získaných z rozličných predmetov. Kolektívna práca pri riešení rozmanitých úloh podnecuje študentov k tvorivosti, zdokonaľuje ich komunikačné schopnosti, zlepšuje ich schopnosť orientovať sa v probléme a plánovať postupy pri riešení výskumných úloh.

Stavebnica Lego Mindstorms obsahuje diely na zostavenie rozličných robotov. Roboti majú možnosť využívať rôzne senzory pre získavanie informácií z okolia. Na základe týchto informácií sa môžu rozhodovať akú činnosť, respektíve časť programu vykonajú. V programe MINDSTORMS NXT sa pomocou programovacích blokov dajú naprogramovať rôzne programy, vďaka ktorým je robot schopný vykonávať rozličné činnosti a plniť úlohy. Hlavnou riadiacou jednotkou robota je Lego Mindstorms NXT kocka, do ktorej sa nahrá program z počítača cez USB kábel, alebo bluetooth spojenie. Senzory ako aj akčné členy sú prepojené s NXT kockou pomocou špeciálnych káblov, ktoré sa pripájajú do jednotlivých portov. Časť portov označená písmenami A,B,C,D je určená pre pripojenie senzorov a druhá časť s označením 1,2,3 je určená pre pripojenie akčných členov. Roboti môžu taktiež využívať pre vzájomnú koordináciu a komunikáciu technológiu bluetooth, vďaka ktorej si môžu posielat' rôzne informácie pomocou správ.

Cieľom bakalárskej práce je preskúmanie možností riadenia robotov Lego Mindstorms, popísať technické informácie ohľadom týchto robotov, preskúmať ich možné využitie na účely výučby a v neposlednom rade, napísať podrobný popis vypracovania jednotlivých úloh, ktoré môžu slúžiť ako návod na laboratórne cvičenia k predmetom spätým s témou práce.

1. TECHNICKÉ INFORMÁCIE [1]

Lego Mindstorms NXT pozostáva z hlavnej riadiacej kocky NXT, zo štyroch senzorov (dotykový, zvukový, svetelný, ultrazvukový), troch interaktívnych servomotorov. K baleniu sa ešte okrem samotných lego častí priložený aj programovací program MINDSTORMS NXT.



Obr. 1. Riadiaca kocka NXT so zapojenými senzormi a akčnými členmi

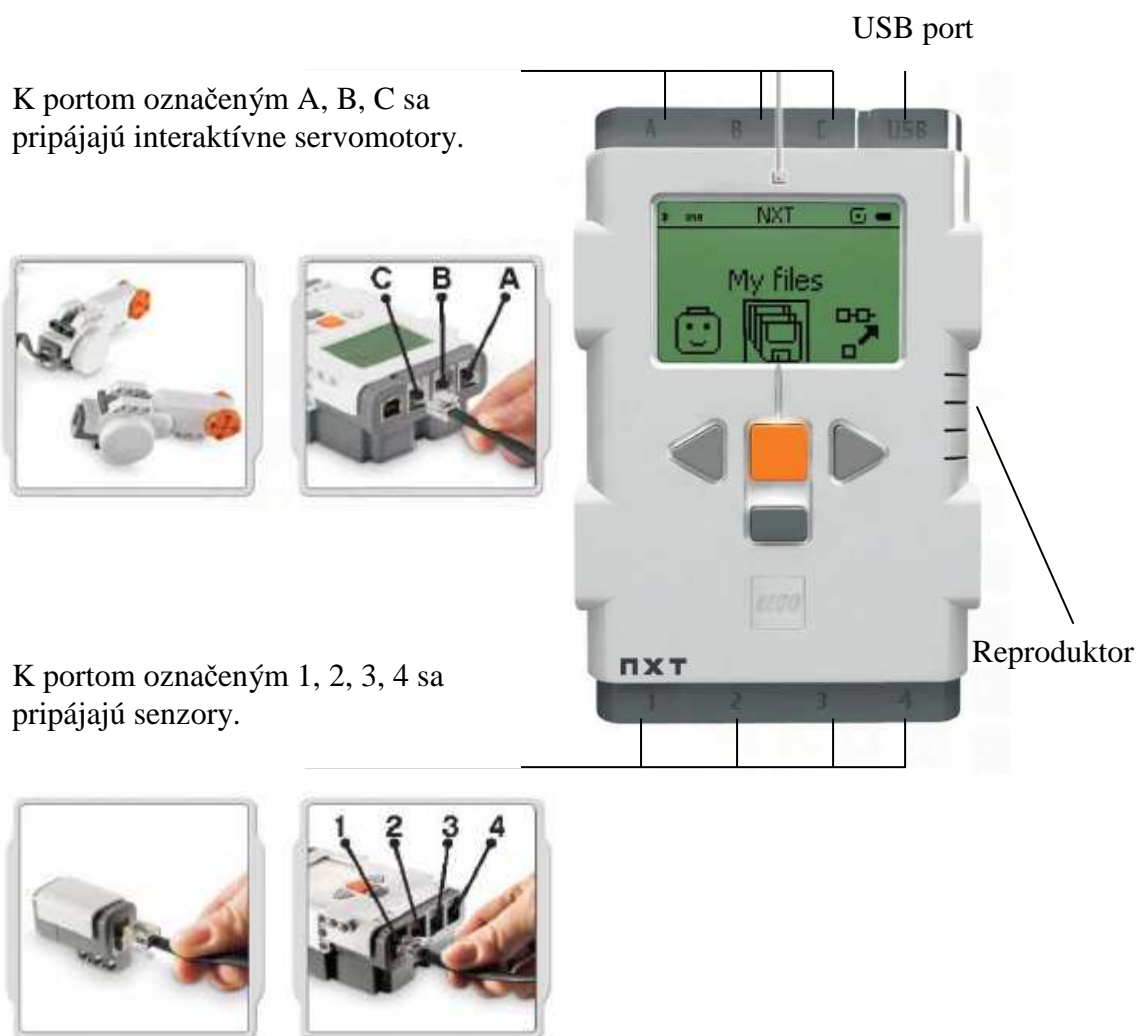
1.1 Riadiaca kocka NXT

Kocka NXT je mozgom robota Lego Mindstorms. Je to inteligentná kocka riadená počítačom, ktorá umožňuje oživenie robota, jeho inteligentné správanie a vykonávanie rôznych činností.

Pozostáva z display, tlačidiel na ovládanie, reproduktora, bluetooth, USB portu, vstupných portov pre motory a senzory. Motory a senzory sa k NXT pripájajú pomocou káblov. Kocka NXT je napájaná šiestimi tužkovými batériami typu AA/LR6 alebo akumulátorom. Kvôli veľmi rýchlemu vybíjaniu batérii je výhodnejšie zakúpiť akumulátor. Pripojenie k PC sa realizuje buď cez USB kábel alebo cez bluetooth.

Technická špecifikácia:

- 32 bitový mikroprocesor ARM7
- 256 KB FLASH, 64 KB RAM
- 8 - bitový mikroprocesor
- 4KB FLASH, 512 B RAM
- Bluetooth® komunikácia, Bluetooth triedy II V2.0 compliant
- USB 2.0 port (12 Mbit/s)
- 4 vstupné porty, 6 vodičová digitálna platforma
- 3 výstupné porty, 6 vodičová digitálna platforma
- maticový displej, 60 x 100 pixlov
- reproduktor, 8 kHz kvalita zvuku
- elektrické zdroje: nabíjacia lithiová batéria, alebo 6 AA článkov



Obr. 2. Riadiaca kocka NXT – spôsob pripájania senzorov a akčných členov

1.2 Senzory

1.2.1 Dotykový senzor



Dotykový senzor je ako vypínač, môže byť zapnutý, alebo vypnutý. Dáva robotovi zmysel pre dotyk. Dotykový senzor je možné pridať do NXT modelu a naprogramovať chovanie modelu tak, aby reagoval na stav stlačenia či uvoľnenia (zapnuté - vypnuté).



zatlačený



uvoľnený



kliknutie

1.2.2 Senzor zvuku



Zvukový senzor deteguje intenzitu zvuku v decibeloch (dB) od jemných tichých zvukov až po hlasné. Zvukový senzor pracuje s dB, ako aj s dBA.

- dB – všetky snímané zvuky vrátane vysokých, alebo nízkych frekvencií, ktoré ľudské ucho nepočuje.
- dBA – len zvuky, ktoré počuje ľudské ucho.

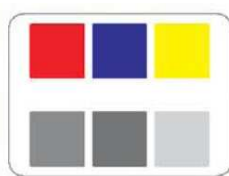
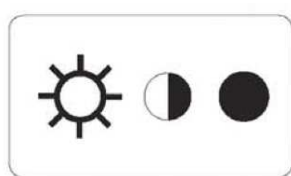
Zvukový senzor meria akustický tlak do úrovne približne 90 dB, čo odpovedá hluku bežiaccej sekačky na trávu. Údaje o zvuku, ktorý je načítaný, je Lego MINDSTORMS NXT zobrazovaný v percentuálnom rozsahu. Napr. 4-5 % zodpovedá tichu v izbe, 5-10 % odpovedá vzdialenému rozhovoru, 10-30 % je rozhovor v blízkosti senzoru, 30-100 % odpovedá hlučnosti v prostredí s hlasnou hudbou. Tieto rozsahy zodpovedajú zvukom vo vzdialenosti 1 m od senzoru.

Po pripojení zvukového senzoru k NXT modelu je možné programovať jeho chovanie podľa hodnôt tohto senzoru.

1.2.3 Svetelný senzor



Svetelný senzor je jedným z dvoch senzorov, ktoré umožňujú robotovi vidieť. Svetelný senzor umožňuje robotovi rozlíšiť svetlo a tmu. Dokáže rozpoznať intenzitu svetla v miestnosti a zmerať intenzitu svetla farebných povrchov na základe odrazeného svetla.

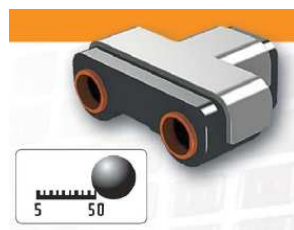


Toto vidia ľudské oči.

Toto vidí robot prostredníctvom svetelného senzora.

Pridaním svetelného senzora k NXT modelu môžeme programovať jeho chovanie v závislosti na zmene intenzity svetla.

1.2.4 Ultrazvukový senzor

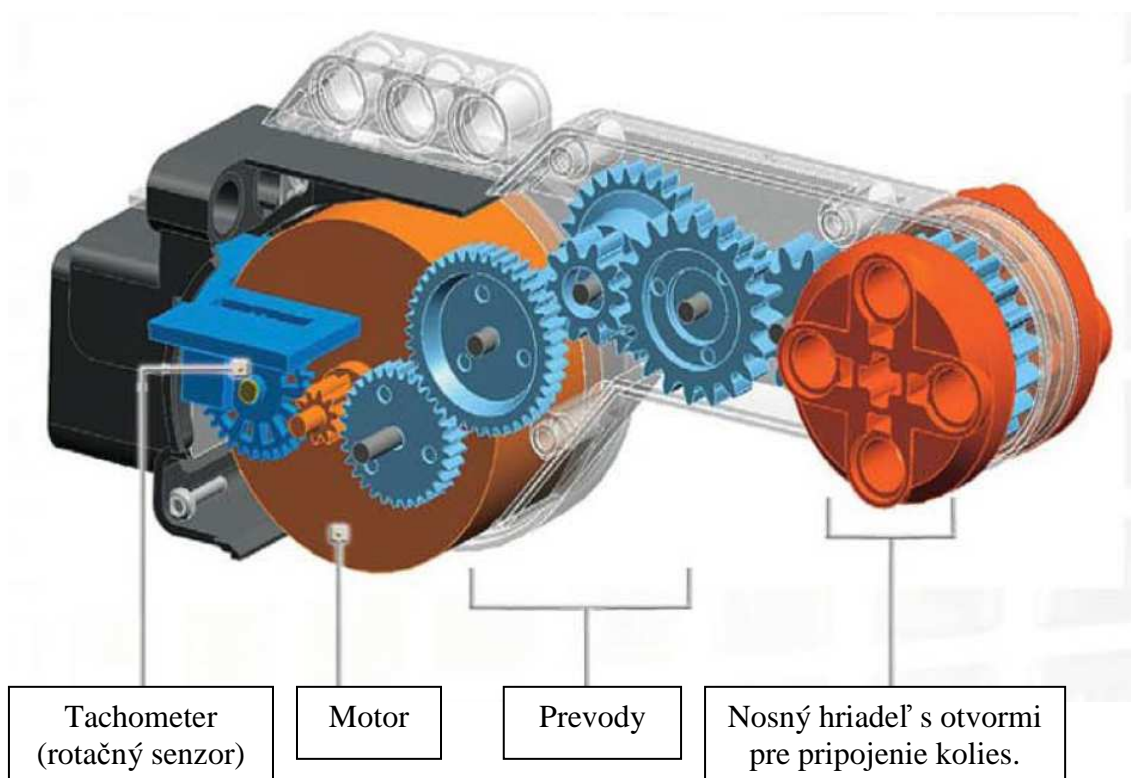


Ultrazvukový senzor umožňuje robotovi vidieť, hľadať predmety, vyhýbať sa prekážkam, merať vzdialenosť a zaznamenávať pohyb. Ultrazvukový senzor využíva rovnakých vedeckých princípov ako netopieri: meria vzdialenosť na základe výpočtu času, behom ktorého dorazí k predmetu zvuková vlna a znovu sa vráti – rovnako ako ozvena. Ultrazvukový senzor meria vzdialenosť v centimetroch aj v palcoch a zobrazuje ich na displeji. Dokáže zmerať vzdialenosť od 0 do 255 cm s presnosťou ± 3 cm. Najlepšie sa získavajú dáta o predmetoch väčších rozmerov. Predmety vyrobené z mäkkých materiálov a zaoblených tvarov (napr. loptičky) alebo predmety, ktoré sú príliš tenké alebo malé, hľadá senzor ťažšie. Pripojením ultrazvukového senzora k NXT modelu môžeme chovanie modelu programovať na základe údajov z ultrazvukového senzora. Ak sú v miestnosti dva či viacej ultrazvukových senzorov, môžu sa vzájomne rušiť.

1.3 Interaktívny servomotor



Pohyb robota umožňujú tri servomotory. Každý motor má vstavaný rotačný senzor, čo umožňuje presnejšie ovládanie robota. Rotačný senzor meria otáčanie motora v stupňoch alebo celkové otáčanie (s presnosťou \pm jeden stupeň). Jedno otočenie zodpovedá 360 stupňom, takže pokiaľ nastavíme motor na otáčanie o 180 stupňoch, motor sa otočí o pol otáčky. Vstavaný rotačný senzor v každom motore umožňuje nastavenie rôznych rýchlostí motorov s rôznymi výkonnosťnými parametrami.



Obr. 3. Interaktívny servomotor

1.4 Bluetooth komunikácia



Bluetooth je technológia, vďaka ktorej sa dajú posielat' a získavat' dáta bez použitia drôtov a káblov. Vďaka tomuto zariadeniu sa dajú vymieňať programy a správy medzi kockami NXT. Dá sa taktiež zariadiť bezdrôtové spojenie medzi počítačom a robotom a vyskúšať tak programy okamžite, aj keď je robot na opačnej strane miestnosti.

Bluetooth komunikáciu je vhodné použiť v nasledujúcich prípadoch:

- Pre prenos programov z počítača do NXT kocky bez použitia USB kábla.
- Pre prenos programov z iného zariadenia ako je počítač (napr. mobil) vrátane NXT kocky.
- Pre prenos programov do ďalšieho NXT zariadenia, alebo skupiny zariadení. Skupinu tvoria viacej ako tri NXT zariadenia.
- Pre komunikáciu medzi dvoma robotmi pomocou posielania a prijímania správ.

Telefón s Bluetooth sa dá využiť na ovládanie robota. Dá sa dokonca použiť ako kvalitný senzor – napr. senzor fotoaparátu.

Nadviazanie spojenia medzi NXT kockami sa realizuje pomocou menu Bluetooth v nastaveniach kocky NXT. Pri voľbe nastavenia volíme kanál cez ktorý budú NXT kocky spolu komunikovať. Máme možnosť zvoliť jeden zo štyroch kanálov. Prvý, druhý a tretí kanál slúžia na komunikáciu medzi robotom, ktorý je zvolený ako **Master** s robotmi, ktorým bola pridelená hodnosť **Slave**. **Slave** má možnosť s robotom ktorý je pripojený ako **Master** komunikovať iba cez nultý kanál a to za podmienky, že už prijal nejakú správu od robota s hodnosťou **Master**.

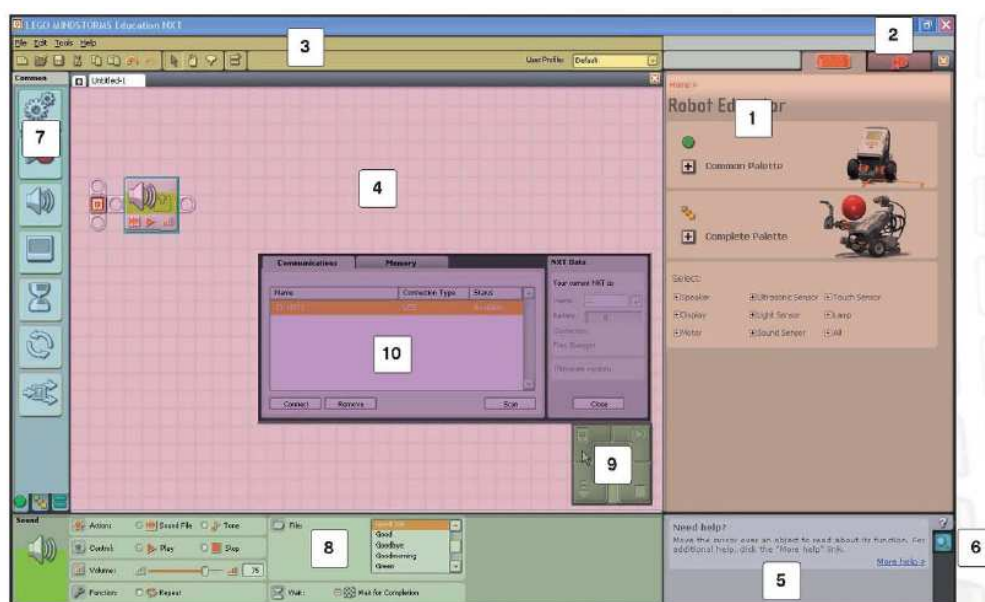


Obr. 4. Bluetooth

2. SOFTWARE

LEGO Education a National Instruments spoločne vytvorili LEGO MINDSTORMS Education NXT software, ktorý je založený na intuitívnej práci v ikonografickom prostredí, vhodnom ako pre úplných začiatočníkov, tak pre expertov na programovanie. LEGO MINDSTORMS Education software je optimalizovanou verziou profesionálneho grafického programovacieho nástroja LabVIEW, ktorý je používaný technikmi a vedcami celého sveta k návrhom konštrukciám a ovládaniam produktov ako sú MP3 prehrávače, DVD prehrávače, mobilné telefóny, bezpečnostný air bag v autách, atď.

Stručný prehľad:



Obr. 5. Pracovná plocha programu Lego Mindstorms NXT

1. Robot Educator – Časť v ktorej sa nachádzajú návody na zostavenie a naprogramovanie niekoľkých druhov modelov robotov.
2. Môj portál – Poskytuje prístup k www.MINDSTORMSeducation.com behom programovania robotov . Na stránke sa dajú získať rôzne nápady, tipy a sťahovať programy, zvukové efekty a ďalšie skvelé možnosti.
3. Panel nástrojov – Obsahuje najčastejšie užívané príkazy z programovej ponuky.
4. Pracovná plocha – Priestor na obrazovke, kde prebieha programovanie. Programovanie na pracovnej ploche prebieha pomocou presúvania

programovacích príkazov z programovacej palety na programovaciu plochu a ich následné spájanie do sekvenčných zväzkov.

5. Malé pomocné okno s nápovedou
6. Mapa pracovnej plochy – Nástroj k jednoduchému posúvaniu po pracovnej ploche.
7. Programovacia paleta – Obsahuje všetky potrebné príkazy (bloky) potrebné pri programovaní. Tabulátory v spodnej časti palety umožňujú prepínať medzi obecnou paletou (obsahuje najčastejšie používané príkazy), úplnou paletou (obsahuje všetky príkazy) a upravenou paletou (obsahuje príkazy, ktoré si užívateľ môže stiahnuť z internetu alebo sám vytvoriť).
8. Konfiguračný panel – Každý programovací príkaz má konfiguračný panel, ktorý umožňuje upraviť tento príkaz z hľadiska špecifických výstupov a vstupov.
9. Panel riadenia – Päť tlačidiel riadiacej jednotky umožňuje sťahovať programy (alebo ich časti) z počítača do NXT kocky. Pomocou riadiacej jednotky sa dajú taktiež zmeniť nastavenia NXT.
10. Okno NXT – Toto dialógové okno poskytuje informácie o pamäti NXT a komunikačných nastaveniach.

2.1 Programovacia paleta

Programovacia paleta je rozdelená na tri rôzne skupiny.

2.1.1 Obecná paleta



Blok pohybu (Move block)

Pomocou tohto príkazu sa ovládajú motory robota.



Blok záznamu/prehrávania (Record/Play block)

Pomocou tohto príkazu sa dá zaznamenať fyzický pohyb robota a ten zopakovať v ktoromkoľvek mieste programu.



Blok zvuku (Sound block)

Blok zvuku umožňuje robotovi vydávať zvuky vrátane slov.



Blok display (Display block)

Tento príkaz sa používa na zobrazenie obrázku, napísaniu textu alebo nakresleniu ľubovoľného tvaru na display NXT.



Blok čakania (Wait block)

Tento príkaz umožňuje robotovi zistiť podmienky, než bude program pokračovať. Tento príkaz sa dá použiť vtedy, keď chceme aby robot počkal na špecifický zvuk, dotyk alebo určitý čas a až potom pokročil v programe ďalej.



Blok cyklu (Loop block)

Tento blok použijeme vtedy, ak chceme aby robot opakoval určitú časť programu stále dokola.



Blok prepínania (Switch blok)

Tento príkaz umožní robotovi uskutočniť vlastné rozhodnutie, napríklad môže ísť vľavo keď započuje hlasný zvuk, alebo vpravo, keď započuje tichý zvuk.

2.1.2 Kompletná paleta

Obecné bloky:



Obecné bloky obsahujú také isté ikony aké sú dostupné v Obecnej palete.

Akčné bloky:



Akčné bloky sú príkazovými ikonami určené k riadeniu výstupných zariadení ako sú: interaktívny servomotor, zvuky NXT, display NXT, Bluetooth odosielanie, motory, lampy.

Bloky senzorov:



Použitím týchto príkazových ikon v kombinácii so senzormi je možné riadiť chovanie robota. Obsahuje ikony senzorov dotyku, zvuku, svetla a vzdialenosti (ultrazvukový senzor), tlačidlá NXT, polohy interaktívneho servomotoru, časovače, Bluetooth príjem, a starších senzorov na dotyk, polohu, svetlo a teplotu.

Bloky priebehov:



Obsahuje príkazové ikony umožňujúce komplexnejšie chovanie robota. Medzi týmito nástrojmi riadenia sú čakanie na, opakovanie, logické premenné, zastavenie bežiaceho programu. Programovacie bloky umožňujú lepšie využitie špecifických podmienok.

Bloky s dátami:



Príkazové ikony k nastavení booleovských operácií, náhodných podmienok, rozsahu a prahu pre senzory.

Zmeny:



Príkazové ikony pre úpravu textu, transformovanie číselného formátu na textový, nastavenie funkcie spánku NXT, ukladanie dátových súborov NXT, kalibrovanie senzorov a resetovanie motorov.

2.1.3 Vlastná paleta

Moje bloky:



Do tohto bloku sa dá uložiť celý program, ktorý možno ako samostatnú príkazovú ikonu použiť kdekoľvek v inom programe

Web prenos:



Blok webového prenosu sa dá použiť v prípade ukladania programov poslaných e-mailom, alebo z webových stránok.

2.2 Konfiguračný panel

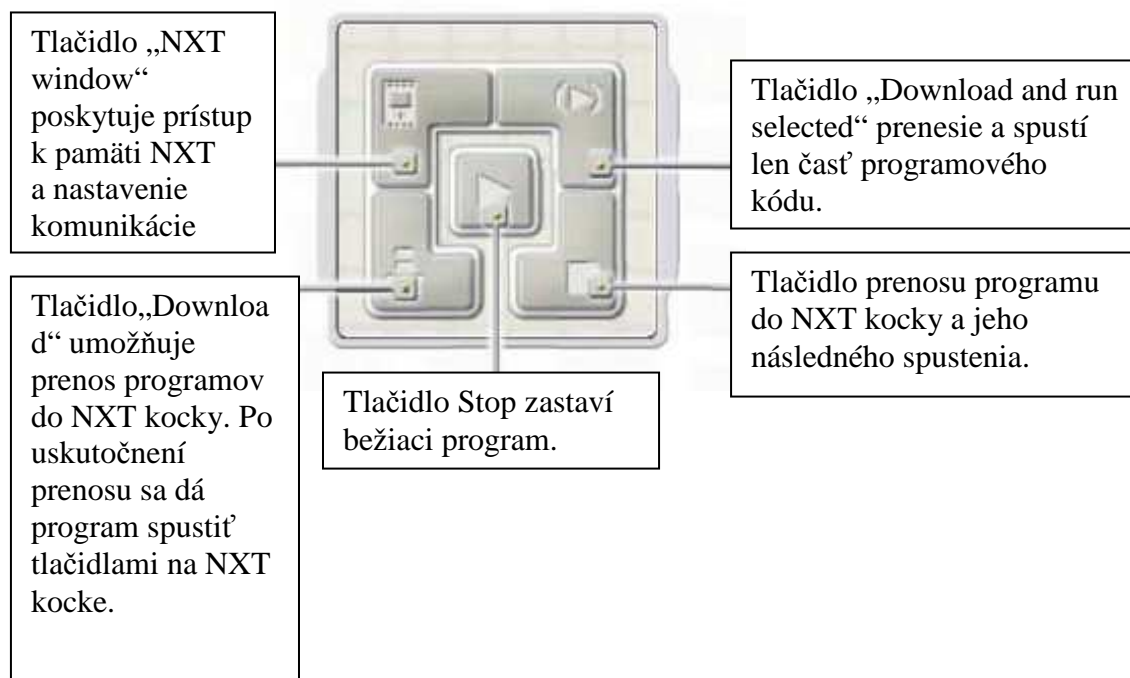
Každý programovací blok obsahuje svoj konfiguračný panel, ktorý umožňuje upraviť nastavenia príkazu na špecifické hodnoty. Po zvolení príkazu na pracovnej ploche sa zobrazí jeho konfiguračný panel v spodnej časti obrazovky.

Zmenou parametrov na každom konfiguračnom paneli možno zmeniť reagovanie na jednotlivé príkazy. Pokiaľ napríklad chceme aby sa robot pohyboval rýchlejšie, môžeme zmeniť výkonnosť (Power property) na konfiguračnom paneli príkazu pohybu (Move).



2.4 Panel riadenia

Päť tlačidiel panelu riadenia umožňuje komunikáciu medzi počítačom a NXT kockou.



3. PRAKTICKÁ ČASŤ

Zostavenie robotov a úlohy:

Po naštudovaní základov týkajúcich sa konštruovania, ovládania a programovania robotov stavebnice Lego Mindstorms som poskladal dvoch robotov. Keďže zmyslom mojej práce nebolo navrhovať konštrukciu, respektíve dizajn robota, tak som využil návody na zostavenie základných typov robotov, ktoré sa nachádzajú v softvéri Lego Mindstorms NXT v časti s názvom Robo Center. K dispozícii som mal štyri rôzne druhy robotov. Prvým bolo vozidlo, druhým robotické rameno, tretím bol robot, ktorý pripomína škorpióna a posledným robotom je humanoid s názvom Alpha Rex.



Mojím výberom bolo vozidlo a robotické rameno, ktoré som vylepšil a prispôbil pre svoju prácu. Počas mojej práce som si kládol rôzne otázky a úlohy na ktoré som sa snažil nájsť odpovede. Postupne som sa dostával cez jednoduché úlohy k zložitejším, čím sa moja práca stávala zaujímavejšou. Výsledkom je úloha na ktorej spolupracujú obidvaja roboti, využívajúc všetky dostupné senzory a komunikáciu prostredníctvom Bluetooth. V nasledujúcej kapitole, by som chcel spomenúť a vysvetliť všetky úlohy, cez ktoré som musel prejsť, než som sa dopracoval k samotnému splneniu hlavnej úlohy.

3.1 Dopravný vozík (podávač loptičiek)

- **Konštrukcia, akčné členy a senzory:**

Pomocou návodu som poskladal robota, ktorý v mojej záverečnej úlohe slúži ako dopravný vozík (podávač loptičiek).

Akčnými členmi tohto robota sú tri nezávislé servomotory, z ktorých dva slúžia na pohyb dopravného vozíka a tretí slúži na ovládanie klieští pomocou ktorých je robot schopný uchopiť loptičku. Všetky motory sú spojené s NXT kockou cez svoj samostatný kábel, ktorý sa zapojí do portu na NXT kocke s označením A, B alebo C. Je dôležité vedieť cez ktorý kábel a port je motor pripojený, pretože pri programovaní cez software Lego MINDSTORMS treba zadať číslo, resp. označenie portu do ktorého sme príslušný akčný člen alebo aj senzor zapojili. Nesprávne označenie portu v programe má za následok nefunkčnosť programu a robot nebude vykonávať to čo by sme chceli.

Na dopravný vozík som umiestnil senzory, ktoré som postupne všetky pri svojich úlohách využil. Robot využíva tieto senzory: dotykový senzor, senzor zvuku, svetelný senzor a ultrazvukový senzor. Každý z týchto senzorov získava informácie pre robota na základe ktorých sa potom robot rozhoduje. Taktiež ako pri akčných členoch aj u senzor je dôležité ich správne zapojenie do príslušného portu s označením: 1,2,3 a 4. Pri programovaní treba v každom bloku senzora zvoliť ten port ktorým je senzor pripojený k NXT kocke.

- **Úlohy pre dopravný vozík:**

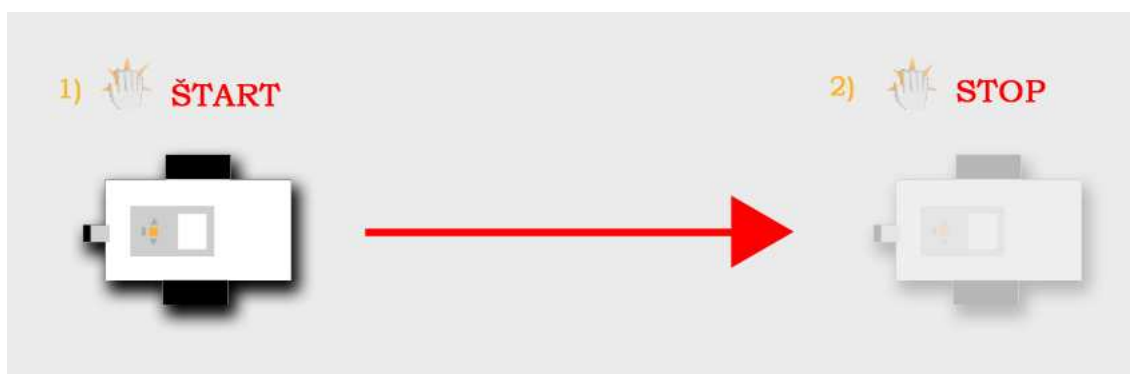
3.1.1 Úloha 1 pre dopravný vozík

Vytvoriť robota, ktorý sa vie pohybovať a reagovať na povel, ktoré príma pomocou zvukového senzora. Po tlesknutí sa robot dá do pohybu smerom dopredu, keď sa tleskne ešte raz robot sa zastaví a zostane stáť.

Potrebné akčné členy: dva servomotory.

Potrebné senzory: zvukový senzor.

Schéma č. 1



Po spustení programu v NXT kocke robot čaká na povel. Keď zvukový senzor zaznamená zvuk, ktorý prekročí určitú hodnotu v decibeloch program sa posunie o krok dopredu čo znamená spustenie obidvoch servomotorov s rovnakým výkonom a otáčkami. Servomotory sa točia dovtedy pokiaľ zvukový senzor opäť nezaznamená zvuk čím sa splní podmienka, servomotory sa vypnú a program skončí.

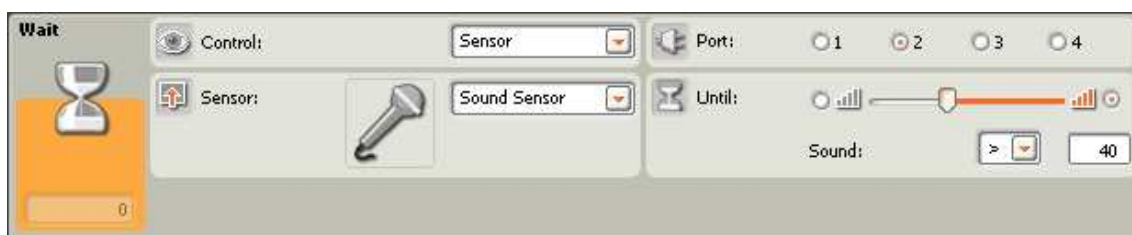
Program 1



Na zostavenie programu som použil dva typy blokov:

- a) blok Wait so zvukovým senzorom
- b) blok na ovládanie servomotorov

1. blok



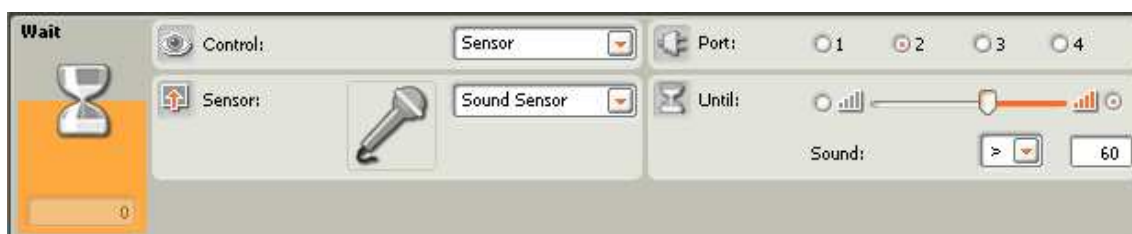
Po vybratí bloku Wait so zvukovým senzorom sa nám automaticky aktivuje hodnota **Senzor** v ponuke **Control** a hodnota **Sound Sensor** v ponuke **Sensor**. V ponuke **Port** je dôležité zvoliť správne číslo portu do ktorého sme pripojili zvukový senzor. Pomocou ponuky **Until** môžeme nastaviť rozmedzie na akú hlasitosť zvuku senzor zareaguje a na akú nie. V tomto prípade senzor reaguje iba na zvuky, ktoré majú vyššiu hodnotu ako 40.

2.blok



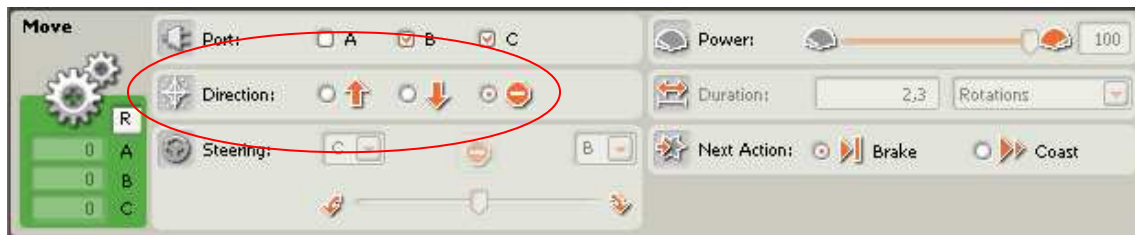
Keďže pomocou tohto bloku chceme ovládať obidva servomotory súčasne, v ponuke **Port** zvolíme tie porty, do ktorých sme zapojili servomotory umožňujúce pohyb robota. **Direction** nám slúži na nastavenie smeru pohybu robota: dopredu, dozadu alebo či má robot zastaviť. Pomocou voľby **Steering** sa dá nastaviť zatáčanie robota. Hodnotu pod akou chceme aby sa robot točil nastavujeme pomocou pohyblivého bežca. V tomto prípade chceme aby robot išiel priamo. Ponuka **Power** nám umožňuje nastaviť výkon motorov. **Duration** nastavíme na **Unlimited**, pretože chceme aby sa motory točili neobmedzene.

3.blok



V treťom bloku nastavíme tie isté hodnoty ako prvom až na hodnotu v ponuke **Until**, ktorú zvýšime na 60 z dôvodu hluku motorov, ktorý by mohol zvukový senzor zaznamenať ako povel.

4. blok



V štvrtom bloku nastavíme **Direction** na **STOP**, čo zabezpečí zastavenie motorov a tým aj robota.

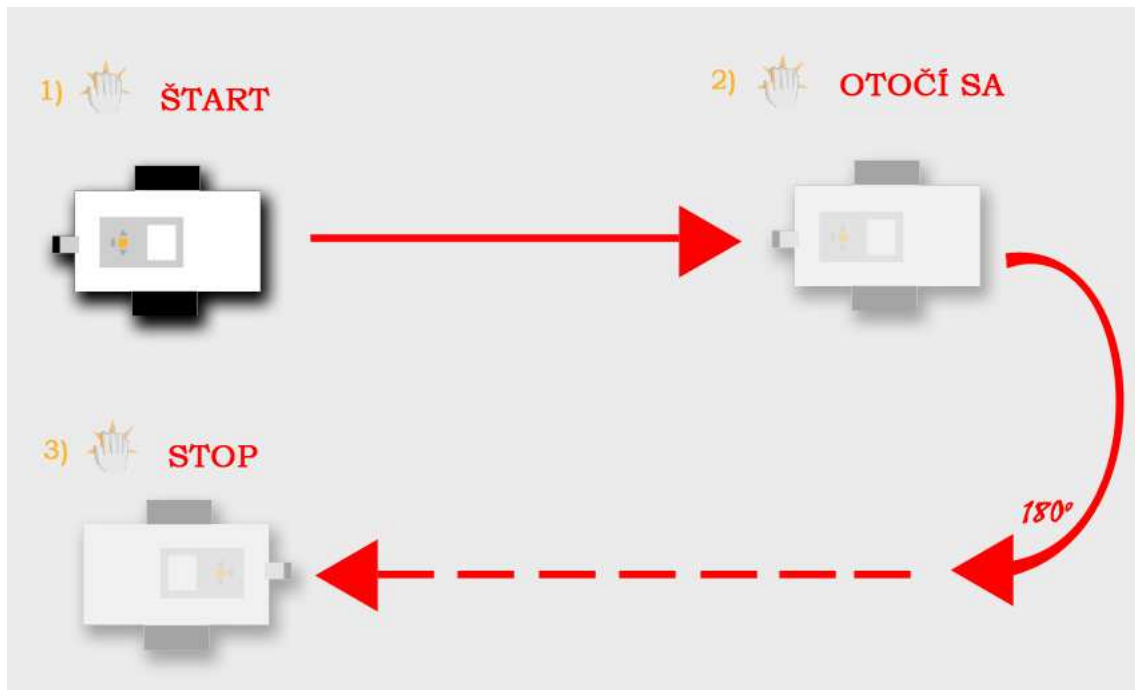
3.1.2 Úloha 2 pre dopravný vozík

Vytvoriť robota, ktorý je schopný pohybu a reagovať na povely, ktoré príma pomocou zvukového senzora. Po tlesknutí sa robot dá do pohybu smerom dopredu, keď sa tleskne druhý krát robot sa otočí na mieste o 180° a keď sa tleskne tretí krát robot zastane.

Potrebné akčné členy: dva servomotory.

Potrebné senzory: zvukový senzor.

Schéma č. 2



Po spustení programu v NXT kocke robot čaká na povel. Keď zvukový senzor zaznamená zvuk, ktorý prekročí určitú hodnotu v decibeloch program sa posunie o krok dopredu čo znamená spustenie obidvoch servomotorov s rovnakým výkonom a

otáčkami. Servomotory sa točia dovtedy pokiaľ zvukový senzor opäť nezaznamená zvuk čím sa splní podmienka, a robot sa otočí o 180° a ide dopredu dovtedy, pokiaľ zvukový senzor nezaznamená posledné tlesknutie, servomotory sa vypnú a program sa ukončí.

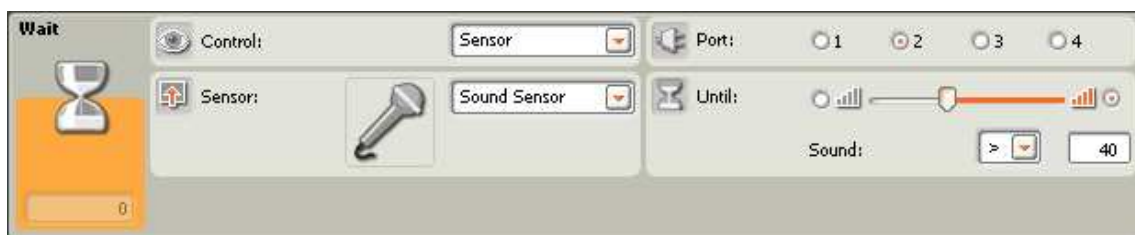
Program 2



Na zostavenie programu som použil dva typy blokov:

- blok Wait so zvukovým senzorom
- blok na ovládanie servomotorov

1. blok



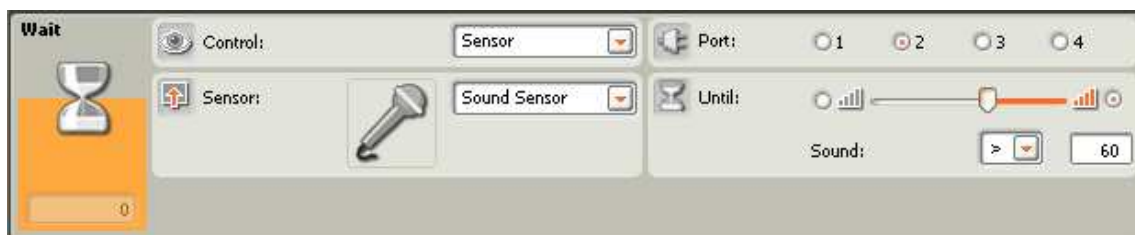
Blok 1 nastavíme rovnako ako blok 1 v úlohe č. 1 (kap. 3.1.1).

2.blok



Blok 2 nastavíme rovnako ako blok 2 v úlohe č. 1 (kap. 3.1.1).

3.blok



Blok 3 nastavíme rovnako ako blok 3 v úlohe č. 1 (kap. 3.1.1).

4.blok



V štvrtom bloku nastavíme **Direction** na **STOP**, čo zabezpečí zastavenie motorov a vďaka nastaveniu **Next Action** na **Coast** program plynule prejde na nasledujúci blok.

5. blok



V piatom bloku máme zvolený iba **Port C**, čo spôsobí otáčanie iba jedného motora. V **Direction** nastavíme aby sa motor otáčal dozadu. **Duration** nastavíme na **370 Degrees** (stupňov). Power nastavíme na požadovanú hodnotu. Nastavenie **Next Action** na **Coast** zaručí aby program vykonal plynule nasledujúci krok

6.blok



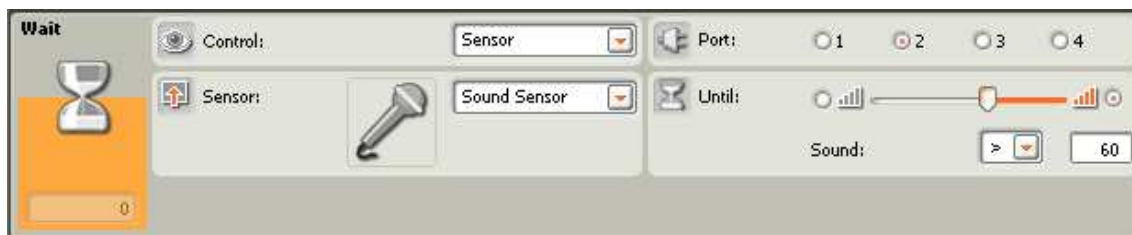
V šiestom bloku máme zvolený iba port B, čo spôsobí otáčanie iba jedného motora. V **Direction** nastavíme aby sa motor otáčal dopredu. **Duration** nastavíme na **370 Degrees** (stupňov). Power nastavím na požadovanú hodnotu. Vďaka **Next Action** nastavenému na **Coast** program prejde plynule na ďalší blok.

7.blok



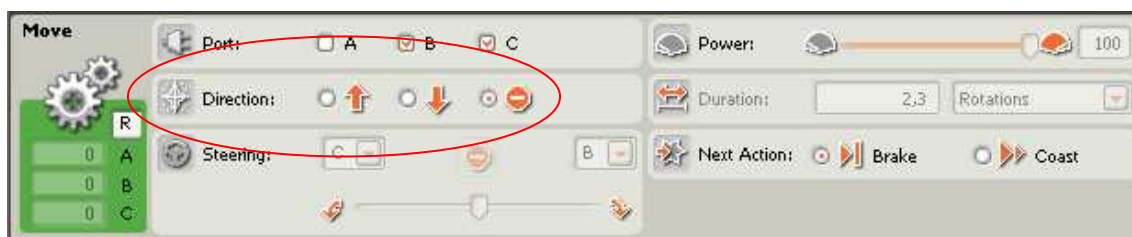
Blok 7 nastavíme rovnako ako blok 2 v úlohe č. 1 (kap. 3.1.1), respektíve blok č. 2 v úlohe č. 2 (kap. 3.1.2).

8.blok



Blok 8 nastavíme rovnako ako blok 3 v úlohe č. 1 (kap. 3.1.1), respektíve blok 3 v úlohe č. 2 (kap. 3.1.2).

9. blok



Blok 9 nastavíme rovnako ako blok 4 v úlohe č. 1 (kap. 3.1.1).

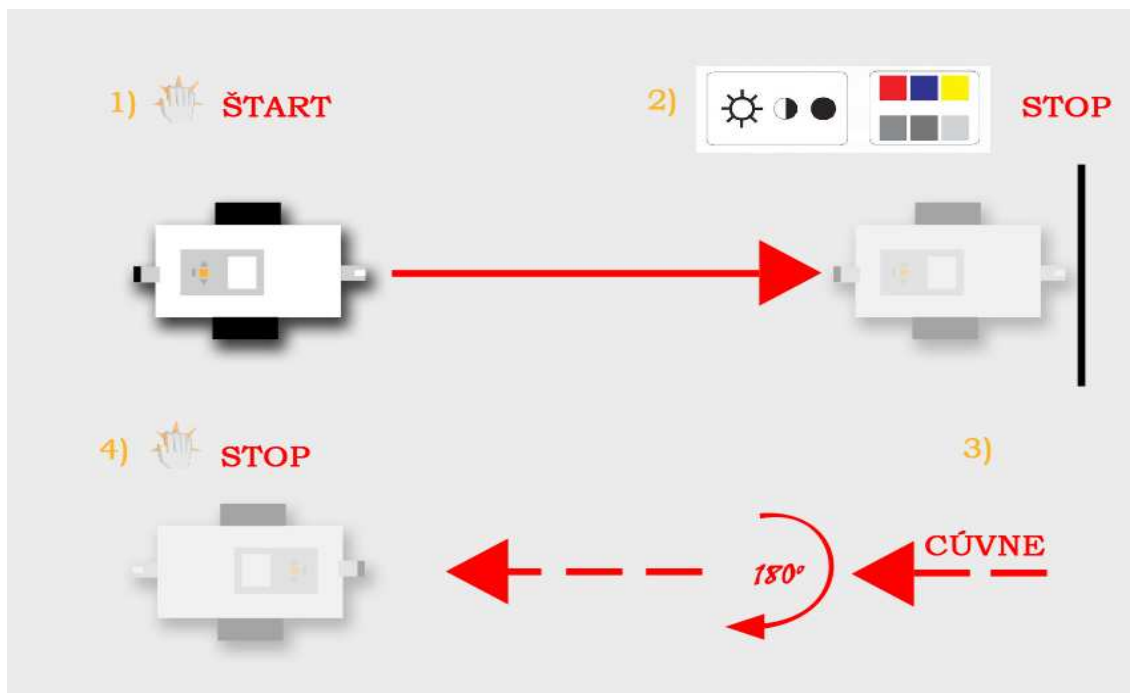
3.1.3 Úloha 3 pre dopravný vozík

Vytvoriť robota, ktorý je schopný sa pohybovať, reagovať na povel, ktoré práma pomocou zvukového senzora a taktiež využíva svetelný senzor na získanie informácie kde má zastaviť a otočiť sa. Po otočení sa vracia späť pokiaľ nedostane povel tlesknutím na to aby zastavil.

Potrebné akčné členy: dva servomotory.

Potrebné senzory: zvukový senzor a svetelný senzor.

Schéma č. 3



Po spustení programu v NXT kocke robot čaká na povel. Keď zvukový senzor zaznamená zvuk, ktorý prekročí určitú hodnotu v decibeloch program sa posunie o krok dopredu čo znamená spustenie obidvoch servomotorov s rovnakým výkonom a otáčkami. Servomotory sa točia dovtedy pokiaľ svetelný senzor nezaznamená čiernu čiaru na zemi, čo pre neho znamená väčší kontrastný rozdiel, po zaznamenaní zastanú. Hneď na to sa robot na mieste otočí o 180° a pokračuje smerom dopredu, pokiaľ zvukový senzor nezaznamená povel tlesknutím, servomotory sa zastavia a program skončí.

Program 3



Na zostavenie programu som použil tri typy blokov:

- a) blok Wait so zvukovým senzorom
- b) blok Wait so svetelným senzorom
- c) blok na ovládanie servomotorov

1. blok



Blok 1 nastavíme rovnako ako blok 1 v úlohe č. 1 (kap. 3.1.1), respektíve ako blok 1 v úlohe č. 2 (kap. 3.1.2).

2.blok



Blok 2 nastavíme rovnako ako blok 2 v úlohe č. 1 (kap. 3.1.1), respektíve blok 2 v úlohe č. 2 (kap. 3.1.2).

3.blok



V bloku **Wait** so svetelným senzorom sa nám automaticky aktivuje hodnota **Senzor** v ponuke **Control** a hodnota **Light Sensor** v ponuke **Sensor**. V ponuke **Port** je opäť veľmi dôležité nastaviť správne číslo portu cez ktorý sme pripojili svetelný senzor s NXT kockou. V **Until** pomocou **Light** a voľby > a < či nám senzor bude reagovať na tmavé alebo svetlé miesta. Posúvaním bežca nastavíme citlivosť na kontrast, to

znamená čo senzor bude vnímať ako tmavé a čo ako svetlé. V ponuke **Function** vyberieme hodnotu **Generate light**, čo znamená že senzor generuje svetlo, ktoré sa potom odráža od meraného objektu a senzor následne po prijatí odrazeného svetla vyhodnotí farbu, resp. svetlosť objektu. Pokiaľ by sme neznačili **Generate light** senzor by meral intenzitu okolitého svetla.

4.blok



Blok 4 nastavíme rovnako ako blok 4 v úlohe č. 2 (kap. 3.1.2).

5.blok



V bloku štyri nastavíme **Port** na hodnoty B a C, pretože chceme ovládať obidva motory. **Direction** nastavíme dozadu, pretože chceme aby robot cúval. **Steering** nemeníme. **Power** nastavíme na požadovaný výkon. **Duration** nastavíme na 2 **Rotations**, čo znamená, že motor vykoná dve otáčky. A **Next Action** nastavíme na **Brake**.

6. blok



Blok 6 nastavíme rovnako ako blok 5 v úlohe č. 2 (kap. 3.1.2).

7.blok



Blok 7 nastavíme rovnako ako blok 6 v úlohe č. 2 (kap. 3.1.2).

8.blok



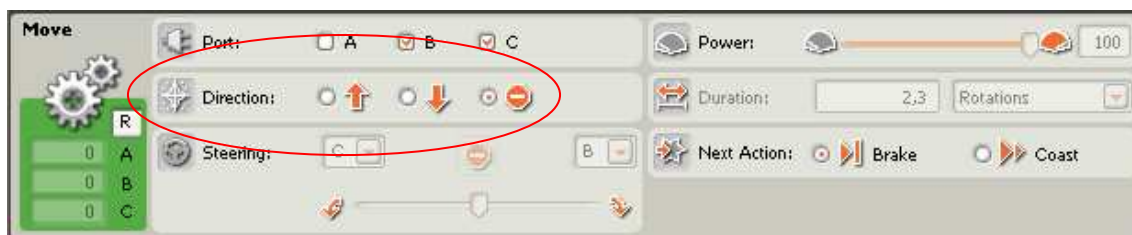
Blok 8 nastavíme rovnako ako blok 7 v úlohe č. 2 (kap. 3.1.1).

9.blok



Blok 9 nastavíme rovnako ako blok 8 v úlohe č. 2 (kap. 3.1.2).

10. blok



Blok 10 nastavíme rovnako ako blok 9 v úlohe č. 2 (kap. 3.1.2).

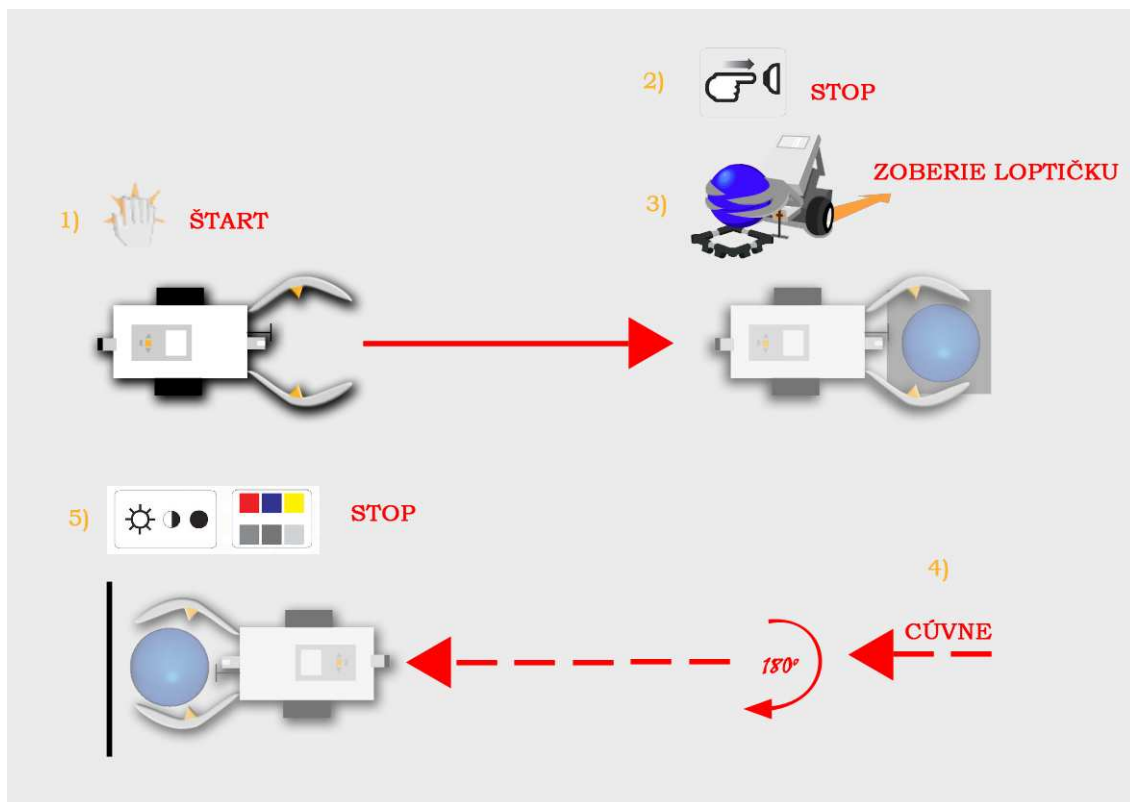
3.1.4 Úloha 4 pre dopravný vozík

Vytvoriť robota, ktorý je schopný sa pohybovať, reagovať na povely, ktoré priamo pomocou zvukového senzora, využíva svetelný senzor na získanie informácie kde má zastaviť a otočiť sa a používa tlakový senzor na to aby zistil že má pred sebou loptičku, ktorú vie chytiť a preniesť na iné miesto. Po tlesknutí sa robot dá do pohybu smerom dopredu a ide, pokiaľ nezaznamená tlakovým senzorom pred sebou loptičku. Keď ju zaznamená zoberie ju, cúvne, otočí sa o 180° a ide dopredu až dovtedy pokiaľ na zemi nezaznamená čiernu čiaru. Potom zastaví a loptičku pustí.

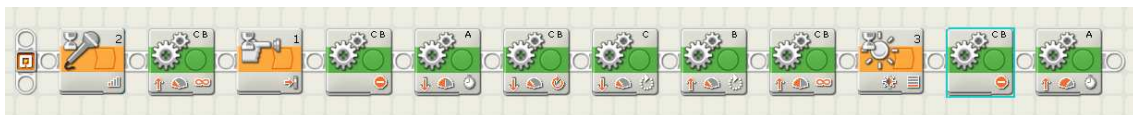
Potrebné akčné členy: dva servomotory na ovládanie pohybu robota a jeden servomotor na ovládanie robotických klieští.

Potrebné senzory: zvukový senzor, svetelný senzor a tlakový senzor.

Schéma č. 4



Program 4



Na zostavenie programu som použil tieto typy blokov:

- blok Wait so zvukovým senzorom
- blok Wait so svetelným senzorom
- blok Wait s tlakovým senzorom
- blok na ovládanie servomotorov

1. blok



Blok 1 nastavíme rovnako ako blok 1 v úlohách č. 1,2,3 (kap. 3.1.1, 3.1.2, 3.1.3.).

2.blok



Blok 2 nastavíme rovnako ako blok 2 v úlohách č. 1,2,3 (kap. 3.1.1, 3.1.2, 3.1.3.).

3.blok



Po vybratí bloku **Wait** s tlakovým senzorom sa nám automaticky aktivuje hodnota **Senzor** v ponuke **Control** a hodnota **Touch Sensor** v ponuke **Sensor**. Zvolíme si správny port cez ktorý sme pripojili tlakový senzor s NXT kockou, v tomto prípade je to port č. 1. V ponuke Action máme tri možnosti:

- **Pressed** – senzor zareaguje, keď sa stlačí,
- **Released** – senzor zareaguje keď sa uvoľní,
- **Bumped** – zatlačí a pustí

Z týchto troch možností si vyberieme prvú **Pressed**, lebo chceme aby sa program posunul na ďalší blok, keď senzor zaznamená, že je zatlačený.

4.blok



Blok 4 nastavíme rovnako ako blok 4 v úlohách č. 1,2,3 (kap. 3.1.1, 3.1.2, 3.1.3.).

5.blok



Piaty blok nám slúži na uzavretie ramena a tých uchopenie loptičky. Servomotor, ktorý ovláda kliešte je spojený s NXT kockou cez A port. Preto v ponuke **Port** vyberieme voľbu A. Vieme že keď sa motor hýbe dopredu, tak sa rameno otvára a keď sa hýbe dozadu tak sa zatvára, preto nastavíme **Direction** na dozadu. **Power** nám stačí nastaviť na polovičnú hodnotu, čiže 50 a otáčanie motoru sa bude vykonávať 0,5 sekundy, takže **Duration** bude 0,5 Seconds. **Next action** ponecháme na **Brake**.

6.blok



Blok 6 nastavíme rovnako ako blok 5 v úlohe č. 3 (kap. 3.1.3).

7. blok



Blok 7 nastavíme rovnako ako blok 5 v úlohe č. 2 (kap. 3.1.2), respektíve blok 6 v úlohe č. 3 (kap. 3.1.3).

8.blok



Blok 8 nastavíme rovnako ako blok 6 v úlohe č. 2 (kap. 3.1.2), respektíve blok 7 v úlohe č. 3 (kap. 3.1.3).

9.blok



Blok 9 nastavíme rovnako ako blok 7 v úlohe č. 2 (kap. 3.1.2), respektíve blok 8 v úlohe č. 3 (kap. 3.1.3).

10.blok



Blok 10 nastavíme rovnako ako blok 3 v úlohe č. 3 (kap. 3.1.3).

11.blok



Blok 11 nastavíme rovnako ako blok 4 v úlohách č. 1,2,3 (kap. 3.1.1, 3.1.2, 3.1.3.).

12.blok



Posledným krokom je pustenie loptičky, ktoré sa vykoná pomocou bloku **Move**, kde si zvolíme **Port A**, **Direction** nastavíme smerom dopredu, čím sa otvoria kliešte a robot pustí loptičku. **Power** nastavíme na vyšší výkon 75, aby robot pustil loptičku rýchlo a **Duration** na 0,3 **Seconds**. Toto je zároveň posledný blok po ktorom program skončí.

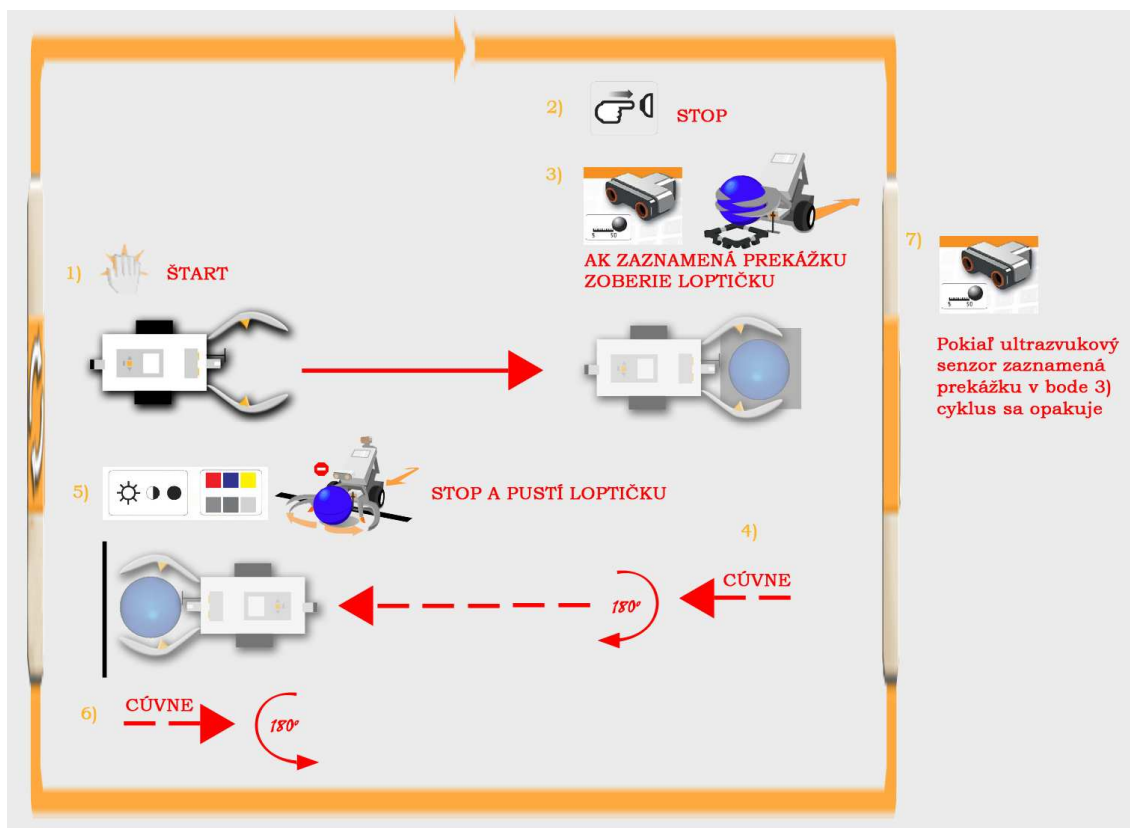
3.1.5 Úloha 5 pre dopravný vozík

Vytvoriť robota, ktorý je schopný sa pohybovať, reagovať na povely, ktoré priamo pomocou zvukového senzora, využíva svetelný senzor na získanie informácie kde má zastaviť a otočiť sa, používa tlakový senzor na to aby zistil že má pred sebou loptičku, ktorú vie chytiť a preniesť na iné miesto a využíva ultrazvukový senzor na zistenie dokedy má vykonávať cyklus.

Potrebné akčné členy: dva servomotory na ovládanie pohybu robota a jeden servomotor na ovládanie robotických klieští.

Potrebné senzory: zvukový senzor, svetelný senzor, tlakový senzor a ultrazvukový senzor.

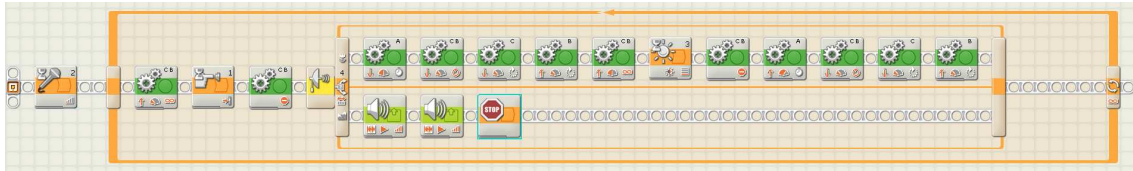
Schéma č. 5



Po tlesknutí sa robot dá do pohybu smerom dopredu a ide, pokiaľ nezaznamená tlakovým senzorom pred sebou loptičku. Keď ju zaznamená zoberie ju, cúvne, otočí sa o 180° a ide dopredu až dovtedy pokiaľ na zemi nezaznamená čiernu čiaru. Potom zastaví a loptičku pustí. Následne cúvne otočí sa o 180° a cyklus sa opakuje bez počiatočného tlesknutia. Robot pracuje dovtedy, pokiaľ sa splní podmienka

zaznamenania objektu ultrazvukovým senzorom. Keď ho nezaznamená povie „No object“ a program sa ukončí.

Program 5



Na zostavenie programu som použil tieto typy blokov:

- blok Wait so zvukovým senzorom
- blok Wait so svetelným senzorom
- blok Wait s tlakovým senzorom
- blok Wait s ultrazvukovým senzorom
- blok na ovládanie servomotorov
- blok STOP

1. blok



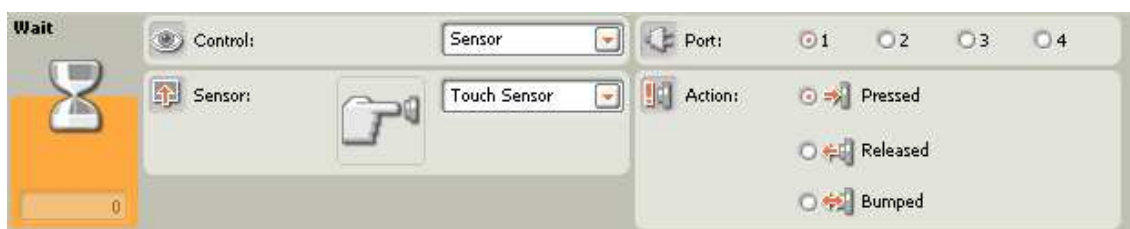
Blok 1 nastavíme rovnako ako blok 1 v úlohách č. 1,2,3,4 (kap. 3.1.1, 3.1.2, 3.1.3, 3.1.4).

2.blok



Blok 2 nastavíme rovnako ako blok 2 v úlohách č. 1,2,3,4 (kap. 3.1.1, 3.1.2, 3.1.3, 3.1.4).

3.blok



Blok 3 nastavíme rovnako ako blok 3 v úlohe č. 4 (kap. 3.1.4).

4.blok



Blok 4 nastavíme rovnako ako blok 4 v úlohách č. 1,2,3,4 (kap. 3.1.1, 3.1.2, 3.1.3, 3.1.4).

5.blok



Piaty blok si zvolíme blok **Switch**, ktorý nám slúži ako podmienka if. Z ponuky **Control** vyberieme **Senzor** a v ponuke **Senzor** vyberieme **Ultrasonic Sensor** – ultrazvukový senzor, pretože chceme aby na základe informácii z neho robil robot rozhodnutie. Zvolíme správny port v ponuke **Port**. V ponuke **Compare** nastavíme vzdialenosť menšiu ako 15 a v ponuke **Show** centimetre. Znamená to že senzor vykoná určitú časť programu pokiaľ zaznamená objekt vo vzdialenosti 15 a menej centimetrov a inú časť programu, pokiaľ objekt nezaznamená. Bloky s označením a) vykoná pokiaľ sa splní podmienka a bloky s označením b) pokiaľ sa podmienka nesplní.

6. a) blok



Blok 6 a) nastavíme rovnako ako blok 5 v úlohe č. 4 (kap. 3.1.4).

7. a) blok



Blok 7 a) nastavíme rovnako ako blok 6 v úlohe č. 4 (kap. 3.1.4), respektíve blok 5 v úlohe č. 3 (kap. 3.1.3).

8. a) blok



Blok 8 a) nastavíme rovnako ako blok 7 v úlohe č. 4 (kap. 3.1.4), blok 6 v úlohe č. 3 (kap. 3.1.3), blok 5 v úlohe č. 2 (kap. 3.1.2).

9. a) blok



Blok 9 a) nastavíme rovnako ako blok 8 v úlohe č. 4 (kap. 3.1.4), blok 7 v úlohe č. 3 (kap. 3.1.3), blok 6 v úlohe č. 2 (kap. 3.1.2).

10. a) blok



Blok 10 a) nastavíme rovnako ako blok 9 v úlohe č. 4 (kap. 3.1.4), blok 8 v úlohe č. 3 (kap. 3.1.3), blok 7 v úlohe č. 2 (kap. 3.1.2).

11. a) blok



Blok 11 a) nastavíme rovnako ako blok 10 v úlohe č. 4 (kap. 3.1.4), blok 3 v úlohe č. 3 (kap. 3.1.3).

12. a) blok



Blok 12 a) nastavíme rovnako ako blok 11 v úlohe č. 4 (kap. 3.1.4), respektíve blok 4 v úlohách č. 1,2,3,4 (kap. 3.1.1, 3.1.2, 3.1.3, 3.1.4).

13. a) blok



Blok 13 a) nastavíme rovnako ako blok 12 v úlohe č. 4 (kap. 3.1.4).



Keďže sme použili cyklus, tak všetky bloky a) sa vykonávajú dovtedy, pokiaľ je splnená podmienka v 5 bloku!

Ak podmienka nie je splnená, program pokračuje blokmi:

6. b) blok



Blok **Sound** nám slúži na prehratie zvuku, resp. tónu. V ponuke **Action** zvolím **Sound File**, pretože chceme aby nám robot prehral zvukový súbor. V ponuke **Control** nastavíme **Play**, lebo chceme zvuk prehrať. **Volume** nastavíme na ľubovoľnú hlasitosť. V ponuke **File** vyberieme zvuk **No** a v ponuke **Wait** zaškrtneme **Wait for completion**, čo znamená, že program počká kým sa prehrá zvukový súbor a až potom prejde na ďalší blok.

7. b) blok



V bloku 7. b) nastavíme tie isté hodnoty ako v bloku 6. b), len v ponuke File vyberieme iný zvukový súbor – Object.

Výsledkom blokov 6. b) a 7. b) je, že robot povie No Object.

8. b) blok



Blok **Stop** zastaví program.

3.2 Robotické rameno (triedič loptičiek)

- **Konštrukcia, akčné členy a senzory:**

Robotické rameno som poskladal na základe návodu, ktorý je v softvéri Lego Mindsotrms NXT v časti s názvom Robo Center. Toto robotické rameno slúži ako triedič loptičiek.

Akčnými členmi robota sú tri servomotory. Každý servomotor má špeciálnu úlohu, prvý umožňuje robotovi na základe prevodov otáčať sa do strán, druhý motor umožňuje naklápanie ramena vo vertikálnom smere - smerom hore a dole, posledný motor slúži na ovládanie robotických klieští pomocou ktorých je robot schopný chytiť loptičku. Všetky motory sú spojené s NXT kockou cez svoj samostatný kábel, ktorý sa zapojí do portu na NXT kocke s označením A, B alebo C. Je dôležité vedieť cez ktorý kábel a port je motor pripojený, pretože pri programovaní cez software Lego MINDSTORMS treba zadať číslo, resp. označenie portu do ktorého sme príslušný akčný člen alebo aj senzor zapojili. Nesprávne označenie portu v programe má za následok nefunkčnosť programu a robot nebude vykonávať to čo by sme chceli.

Robotické rameno využíva všetky dostupné senzory, ktoré som postupne využil pri úlohách. Podrobnejšie informácie o využití senzorov na robotickom ramene som popísal v jednotlivých častiach úloh.

Robot využíva nasledovné senzory: dotykový senzor, senzor zvuku, svetelný senzor a ultrazvukový senzor. Každý z týchto senzorov získava informácie pre robota na základe ktorých sa potom robot rozhoduje. Taktiež ako pri akčných členoch aj u senzor je dôležité ich správne zapojenie do príslušného portu s označením: 1,2,3 a 4. Pri programovaní treba v každom bloku senzora zvoliť ten port ktorým je senzor pripojený k NXT kocke.

- Úlohy pre robotické rameno:

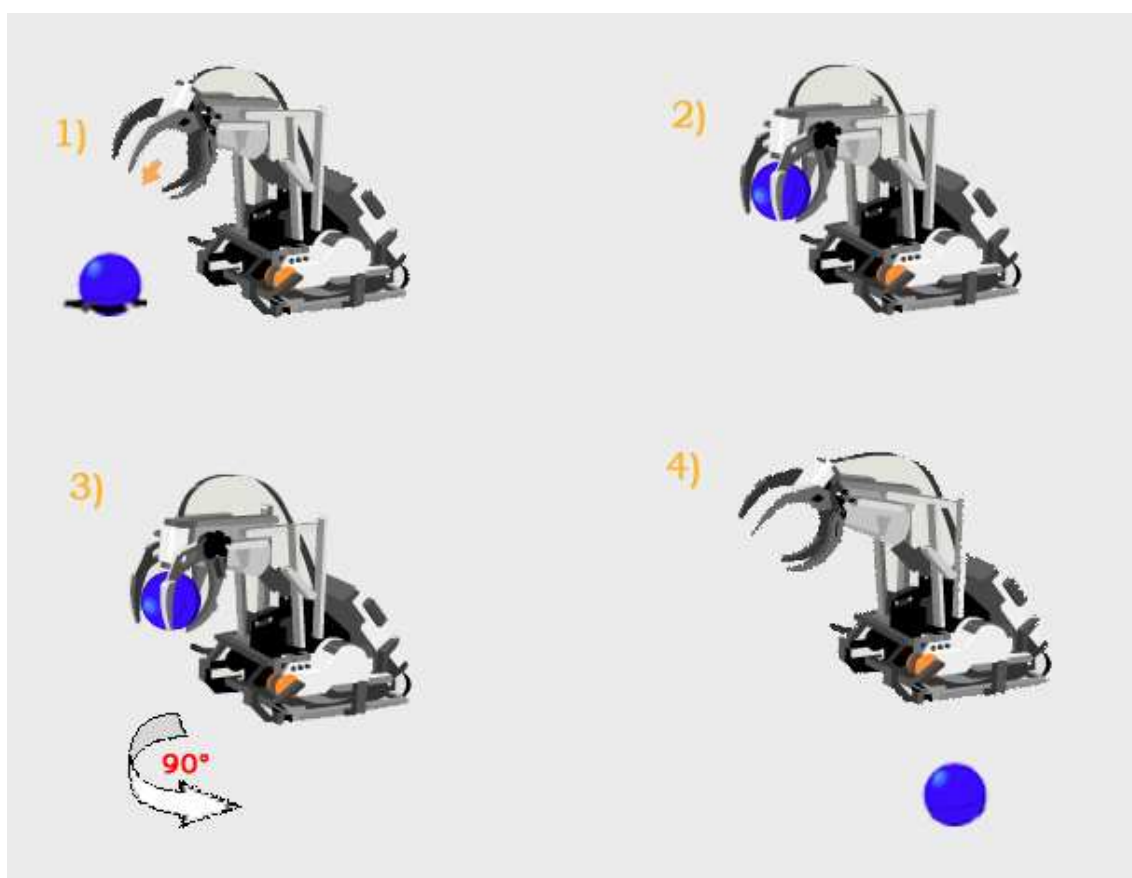
3.2.1 Úloha 1 pre robotické rameno

Vytvoriť robotické rameno, ktoré bude schopné prenášať loptičku. Po zdvihnutí loptičky sa rameno otočí o 90° a loptičku pustí.

Potrebné akčné členy: tri servomotory.

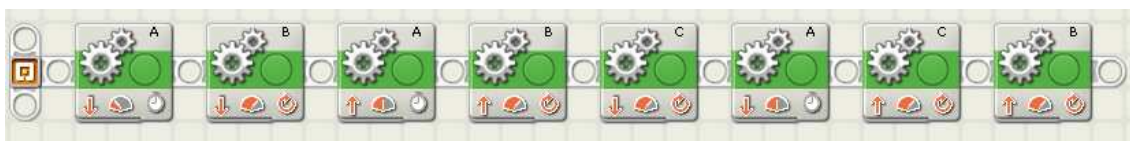
Potrebné senzory: žiadne

Schéma č. 1



Robotické rameno sa zohne po loptičku a uchopí ju do klieští. Následne sa zdvihne a otočí sa o 90° . Po otočení, loptičku pustí a vráti sa späť do pôvodnej pozície.

Program 1



Na zostavenie programu som použil jeden typ blokov:

- a) blok na ovládanie servomotorov

1.blok



Prvý blok v programe nám slúži na otvorenie klieští. Keďže robotické kliešte sú ovládané motorom, ktorý je pripojený do portu A, tak v ponuke **Port** zvolíme A. Keď sa servomotor točí dopredu – kliešte sa zatvárajú, keďže chceme kliešte otvoriť, tak v ponuke **Direction** nastavíme smer otáčania motora na dozadu. **Power** nastavíme na malý výkon 25 a **Duration** na 0,5 **Seconds**, čo je dostatočný čas na otvorenie klieští. **Next Action** ponecháme na **Brake**.

2.blok



V bloku číslo 2 spustíme rameno smerom dole po loptičku. Motor, ktorý nám slúži na vertikálne pohybovanie robotického ramena je pripojený k NXT kocke cez port B, preto v ponuke **Port** vyberieme B. **Direction** nastavíme smerom dole, pretože chceme aby sa rameno posunulo smerom nadol. **Power** nastavíme na žiadanú hodnotu, v tomto prípade postačí 75. **Duration** nastavíme na 23 **Rotations** - otáčok. Počet otáčok závisí od toho v akej bolo rameno v pôvodnej polohe. **Next Action** ponecháme na **Brake**.

3.blok



Tretím blokom docielim zavretie robotických klieští a tým aj uchopenie loptičky. **Port** zvolíme A. Pri nastavení **Direction** dopredu sa nám rameno zavrie. **Power** nastavíme na požadovanú hodnotu. Na uchopenie loptičky stačí **Power** nastaviť na 50. Dostatočným časom na uzavretie klieští je 0,5 sekundy, preto **Duration** nastavíme na 0,5 **Seconds**. **Next Action** ponecháme na **Brake**.

4.blok



Pomocou štvrtého bloku rameno o niečo zdvihneme. **Port** nastavíme na B a **Direction** dopredu, čo nám zaistí zdvihnutie ramena. **Power** nastavíme na 75 a **Duration** na 18 **Rotations** - otáčok. **Next Action** ponecháme na **Brake**.

5.blok



Piaty blok nám zaistí pootočenie ramena o 90° v horizontálnom smere. Motor ktorý umožňuje otáčanie ramena v horizontálnom smere je pripojený ku kocke NXT prostredníctvom portu C, preto **Port** nastavíme na C. **Direction** nastavíme dozadu, pretože keď sa motor točí dozadu – rameno sa otáča doprava. **Power** nastavíme na 75 a otočenie o 90° nám zaistí 45 otáčok motora, preto **Duration** nastavíme na 45 **Rotations**. **Next Action** ponecháme na **Brake**.

6. blok



Blok 6 nastavíme rovnako ako blok 1 v úlohe č. 4.2.1 .

7.blok



Pomocou bloku 7 sa rameno pootočí späť o 90°. **Port** zvolíme C, **Direction** dopredu, **Power** nastavíme na 75 a **Duration** na 45 **Rotations**. **Next action** ponecháme na **brake**.

8.blok



Posledný blok nám zaistí vrátenie ramena do pôvodnej polohy, **Port** nastavíme na B, **Direction** dopredu. **Power** nám postačuje nastaviť na 75. Keďže v druhom bloku sme posunuli rameno dole o 23 otáčok a vo štvrtom o 18 otáčok hore musíme rameno zdvihnúť ešte o 5 otáčok. Preto nastavíme **Duration** na 5 **Rotations**. **Next Action** ponecháme na **Brake**. Po vykonaní tohto bloku program skončí.

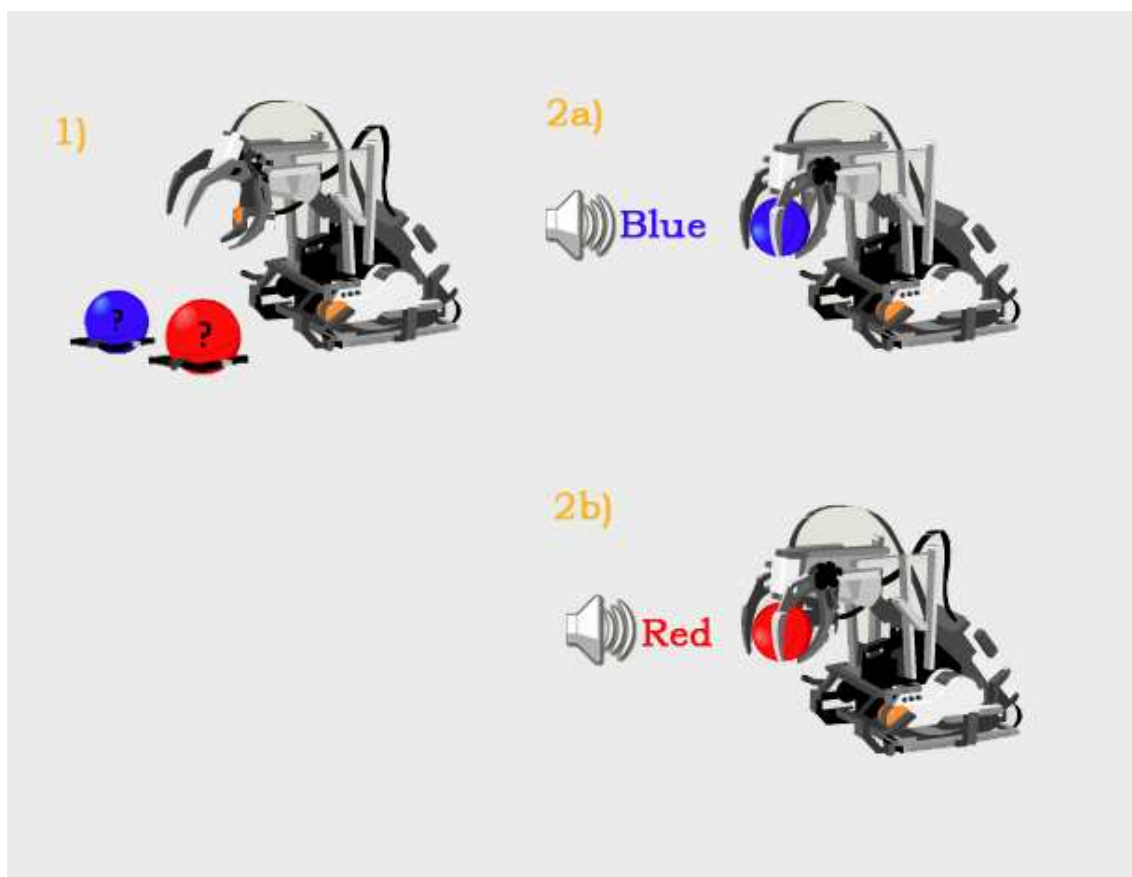
3.2.2 Úloha 2 pre robotické rameno

Vytvoriť robotické rameno, ktoré vie rozoznávať farbu loptičiek pomocou svetelného senzora.

Potrebné akčné členy: dva servomotory, reproduktor

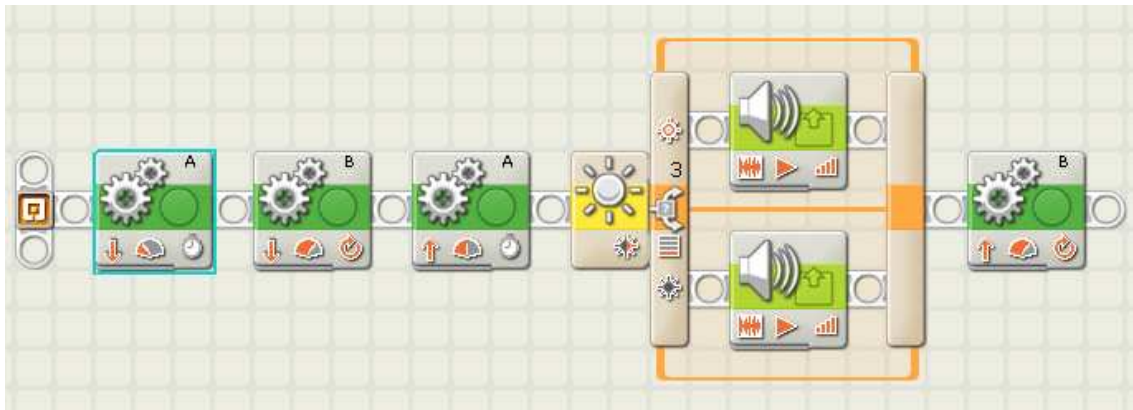
Potrebné senzory: svetelný senzor

Schéma č. 2



Robotické rameno sa zohne po loptičku, uchopí ju do klieští a pomocou svetelného senzora určí či je loptička červená alebo modrá. Ak je loptička modrá povie „Blue“, ak je loptička červená povie „Red“.

Program 2



Na zostavenie programu som použil tieto typy blokov:

- a) blok na ovládanie servomotorov
- b) blok s využitím svetelného senzora
- c) zvukový blok

1.blok



Blok 1 nastavíme rovnako ako blok 1 v úlohe č. 1 (kap. 3.2.1).

2.blok



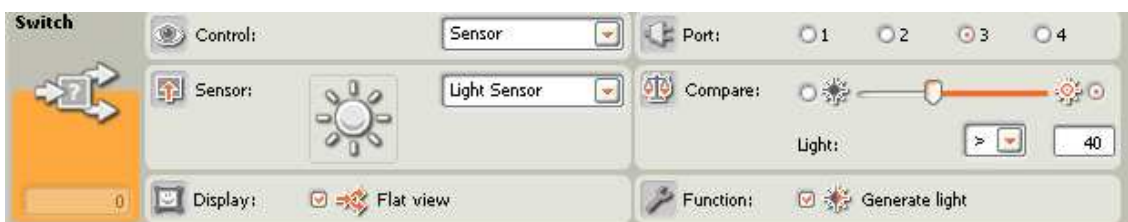
Blok 2 nastavíme rovnako ako blok 2 v úlohe č. 1 (kap. 3.2.1).

3.blok



Blok 3 nastavíme rovnako ako blok 3 v úlohe č. 1 (kap. 3.2.1).

4. blok



Štvrtý blok **Switch** nám slúži ako podmienka if. Pokiaľ sa splní podmienka, program pokračuje jednou cestou programu a pokiaľ nie, pokračuje inou cestou programu.

V ponuke **Control** si zvolíme, že chceme využívať senzor. Z ďalšej ponuky senzorov si vyberieme **Light Sensor** (svetelný senzor), pretože chceme aby sa na základe neho robot rozhodoval. Vieme, že svetelný senzor máme pripojený k NXT konce cez port č. 3, preto **Port** nastavíme na 3. V **Compare** máme možnosť nastaviť citlivosť, to znamená čo bude robot vnímať pomocou senzora ako svetlé a čo ako tmavé. Pre dobré rozlíšenie modrej a červenej farby je dobré nastaviť Light na > ako 40. V ponuke **Function** zvolíme **Generate Light**, pretože chceme merať odrazené svetlo.

5. a) blok



Piaty a) blok nám slúži na to aby robot povedal, že je loptička červená, pokiaľ je splnená podmienka v bloku 4. V **Action** vyberieme **Sound File** – prehratie zvukového súboru a **Control** nastavíme na **Play** - prehrávanie. Hlasitosť - **Volume** dáme na 100, aby bolo robota dobre rozumieť. Keďže chceme aby robot oznámil farbu loptičky iba raz, tak v ponuke **Function** ponecháme **Repeat** nezaškrtnuté. Vo **File** zvolíme zvukový súbor **Red**. V ponuke **Wait** zaškrtneme **Wait for Completion**.

5. b) blok



Piaty b) blok sa vykoná pokiaľ nie je splnená podmienka vo štvrtom bloku. Hodnoty nastavíme presne rovnako ako v bloku 5 a), len vo **File** zvolíme prehratie iného zvukového súboru – **Blue**.

6. blok



Posledný blok zdvihne rameno do pôvodnej polohy. **Port** nastavíme na B, **Direction** na dopredu, **Power** postačí 75. Aby sa rameno dostalo do pôvodnej polohy, musí vykonať 23 otáčok späť, preto **Duration** nastavíme na 23 **Rotations**. **Next Action** ponecháme na **Brake**. Po vykonaní tohto bloku, program skončí.

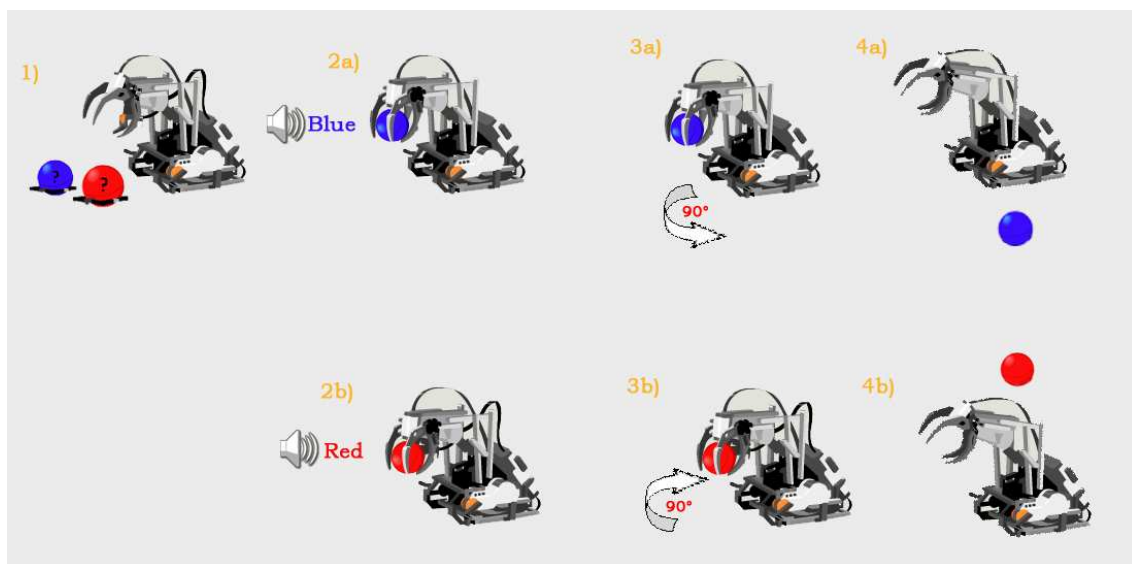
3.2.3 Úloha 3 pre robotické rameno

Vytvoriť robotické rameno, ktoré vie rozoznávať farbu loptičiek pomocou svetelného senzora. Keď je loptička modrá, otočí sa o 90° doprava a pustí ju. Ak je loptička červená, otočí sa o 90° doľava a pustí ju.

Potrebné akčné členy: tri servomotory, reproduktor

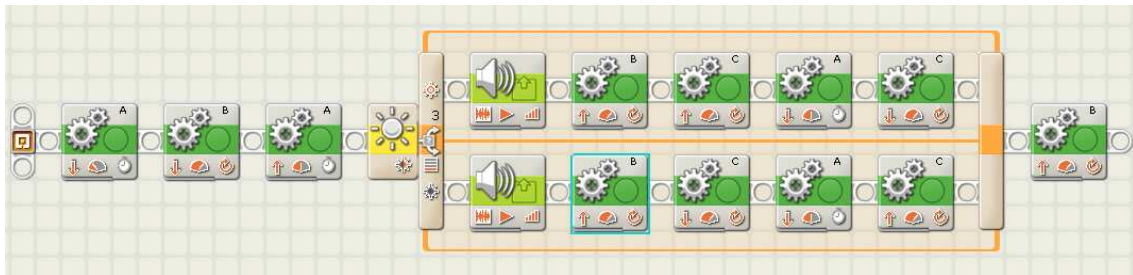
Potrebné senzory: svetelný senzor

Schéma č. 3



Robotické rameno sa zohne po loptičku a uchopí ju do klieští a pomocou svetelného senzora určí či je loptička červená alebo modrá. Ak je loptička modrá povie „Blue“, otočí sa o 90° doprava a pustí ju, pokiaľ je loptička červená povie „Red“, otočí sa o 90° doľava a pustí ju.

Program 3



Na zostavenie programu som použil tieto typy blokov:

- a) blok na ovládanie servomotorov
- b) blok s využitím svetelného senzora
- c) zvukový blok

1.blok



Blok 1 nastavíme rovnako ako blok 1 v úlohách č. 1,2 (kap. 3.2.1 a 3.2.2).

2.blok



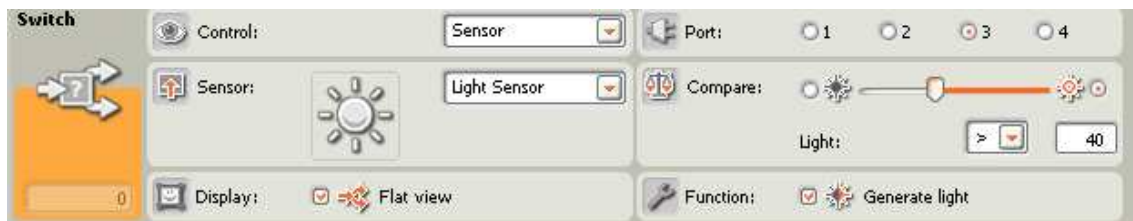
Blok 2 nastavíme rovnako ako blok 2 v úlohách č. 1,2 (kap. 3.2.1 a 3.2.2).

3.blok



Blok 3 nastavíme rovnako ako blok 3 v úlohách č. 1,2 (kap. 3.2.1 a 3.2.2).

4. blok



Blok 4 nastavíme rovnako ako blok 4 v úlohe č. 2 (kap. 3.2.2).

5. a) blok



Blok 5 a) nastavíme rovnako ako blok 5 a) v úlohe č. 2 (kap. 3.2.2).

6. a) blok



Blok 6 a) nastavíme rovnako ako blok 4 v úlohe č. 1 (kap. 3.2.1).

7. a) blok



Blok 7 a) nám zaistí pootočenie ramena o 90° v horizontálnom smere. Motor ktorý umožňuje otáčanie ramena v horizontálnom smere je pripojený ku kocke NXT prostredníctvom portu C, preto **Port** nastavíme na C. **Direction** nastavíme dopredu, pretože keď sa motor točí dopredu – rameno sa otáča doľava. **Power** nastavíme na 75 a otočenie o 90° nám zaistí 45 otáčok motora, preto **Duration** nastavíme na 45 **Rotations**. **Next Action** ponecháme na **Brake**.

8. a) blok



Blok 8 a) nastavíme rovnako ako blok 1 v úlohách č. 1,2 (kap. 3.2.1 a 3.2.2), respektíve blok č. 6 v úlohe č. 1 (kap. 3.2.1).

9. a) blok



Pomocou bloku 9. a) sa rameno pootočí späť o 90°. **Port** zvolíme C, **Direction** na dozadu, **Power** nastavíme na 75 a **Duration** na 45 Rotations. **Next Action** ponecháme na **Brake**.

5. b) blok



Piaty b) blok sa vykoná pokiaľ nie je splnená podmienka vo štvrtom bloku tohto programu. Blok 5 b) nastavíme rovnako ako blok 5 b) v úlohe č. 3 (kap. 3.2.2).

6. b) blok



Blok 6 b) nastavíme rovnako ako blok 4 v úlohe č. 1 (kap. 3.2.1), respektíve blok 6 a) v úlohe č. 3 (kap. 3.2.3).

7. b) blok



Blok 7 b) nastavíme rovnako ako blok 5 v úlohe č. 1 (kap. 3.2.1). Nastavenie bloku je rovnaké ako bloku 7 a), až na **Direction**, ktoré nastavíme na dozadu, aby sa rameno otočilo doprava.

8. b) blok



Blok 8 b) nastavíme rovnako ako blok 1 v úlohách č. 1,2 (kap. 3.2.1 a 3.2.2), respektíve blok č. 6 v úlohe č. 1 (kap. 3.2.1).

9. b) blok



Blok 9 b) nastavíme rovnako ako blok 7 v úlohe č. 1 (kap. 3.2.1).

10. blok



Blok 10 nastavíme rovnako ako blok 8 v úlohe č. 1 (kap. 3.2.1).

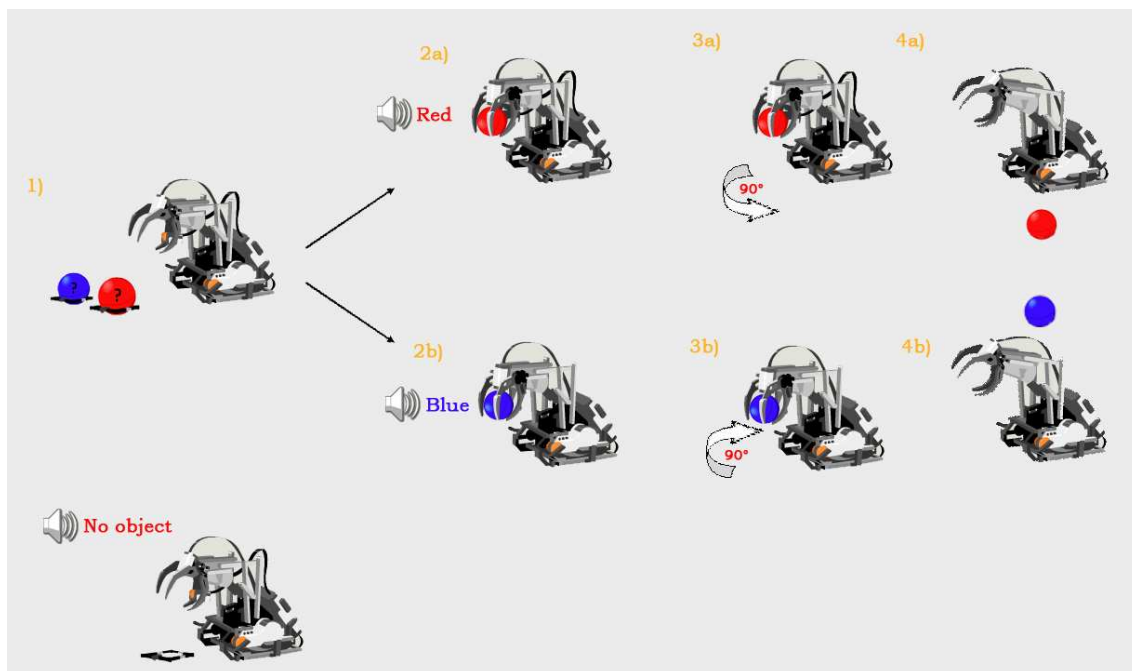
3.2.4 Úloha 4 pre robotické rameno

Vytvoriť robotické rameno, ktoré vie rozoznávať farbu loptičiek pomocou svetelného senzora. Keď je loptička modrá, otočí sa o 90° doprava a pustí ju. Ak je loptička červená, otočí sa o 90° doľava a pustí ju. Pokiaľ ale nemá loptičku oznámi – no object.

Potrebné akčné členy: tri servomotory, reproduktor

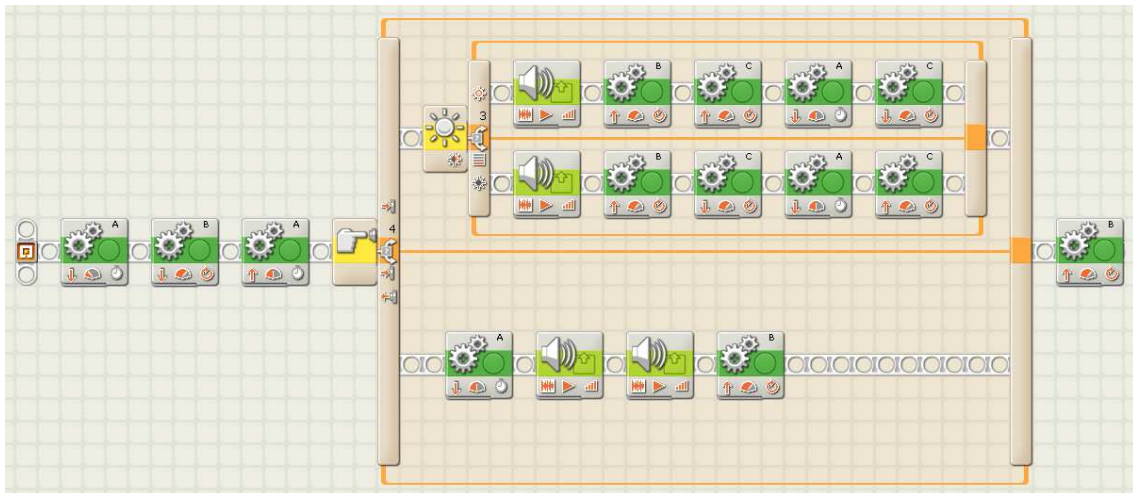
Potrebné senzory: svetelný senzor, tlakový senzor

Schéma č. 4



Robotické rameno sa zohne po loptičku a uchopí ju do klieští a pomocou svetelného senzora určí či je loptička červená alebo modrá. Ak je loptička modrá povie „Blue“, otočí sa o 90° doprava a pustí ju, pokiaľ je loptička červená povie „Red“, otočí sa o 90° doľava a pustí ju. Ak na základe tlakového senzora zistí, že loptičku nedostal, povie – no object.

Program 4



Na zostavenie programu som použil tieto typy blokov:

- a) blok na ovládanie servomotorov
- b) blok s využitím svetelného senzora
- c) blok s využitím tlakového senzora
- d) zvukový blok

1.blok



Blok 1 nastavíme rovnako ako blok 1 v úlohách č. 1,2,3 (kap. 3.2.1, 3.2.2 a 3.2.3).

2.blok



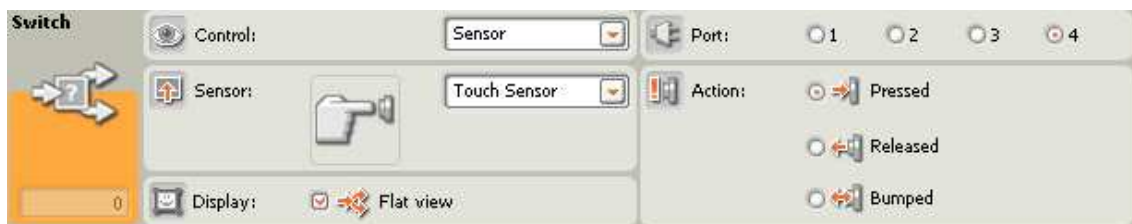
Blok 2 nastavíme rovnako ako blok 2 v úlohách č. 1,2,3 (kap. 3.2.1, 3.2.2 a 3.2.3).

3.blok



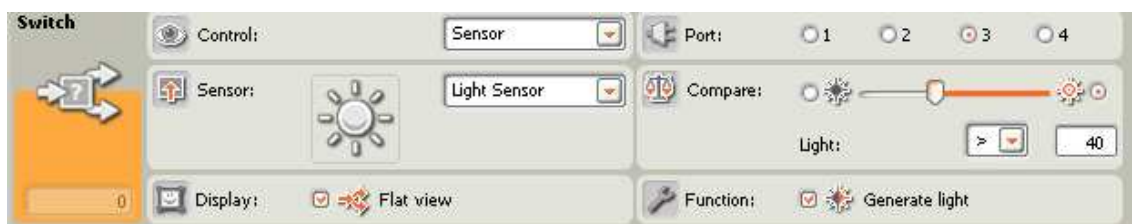
Blok 3 nastavíme rovnako ako blok 3 v úlohách č. 1,2,3 (kap. 3.2.1, 3.2.2 a 3.2.3).

4.blok



Štvrtý blok Switch využijeme na to aby robot vedel zistiť, či loptičku uchopil, alebo nie. V tomto bloku si nastavíme podmienku. Ak sa táto podmienka splní, vykoná sa určitá časť programu a ak sa nesplní vykoná sa iná časť programu. Chceme aby sa robot rozhodoval na základe senzora a to konkrétne tlakového senzora, preto v ponukách **Control** a **Sensor** nastavíme voľby: **Sensor** a **Touch Sensor**. Tlakový senzor sme si pripojili k NXT kocke cez port č. 4, preto zvolíme **Port 4**. Keď sa robotické kliešte zatvoria a zovrú loptičku stlačí sa tlakový senzor, pokiaľ sa tlakový senzor nestlačí kliešte loptičku neuchopili. Na základe toho nastavíme **Action** na **Pressed**.

5. a) blok



Blok 5 a) nastavíme rovnako ako blok 4 v úlohách č. 2,3 (kap. 3.2.2 a 3.2.3).

6. a1) blok



Blok 6 a1) nastavíme rovnako ako blok 5 a) v úlohách č. 2,3 (kap. 3.2.2 a 3.2.3).

7. a1) blok



Blok 7 a1) nastavíme rovnako ako blok 6 a) v úlohe 3 (kap. 3.2.3), respektíve ako blok 4 v úlohe č. 1 (kap. 3.2.1).

8. a1) blok



Blok 8 a1) nastavíme rovnako ako blok 7 a) v úlohe č. 3 (kap. 3.2.3).

9. a1) blok



Blok 9 a1) nastavíme rovnako ako blok 8 a) v úlohe č. 3 (kap. 3.2.3), alebo ako blok 1 v úlohách č. 1,2 (kap. 3.2.1, 3.2.2), respektíve blok č. 6 v úlohe č. 1 (kap. 3.2.1).

10. a1) blok



Blok 10 a1) nastavíme rovnako ako blok 9 a) v úlohe č. 3 (kap. 3.2.3).

6. a2) blok



Blok 6 a2) nastavíme rovnako ako blok 5 b) v úlohách č. 2,3 (kap. 3.2.2, 3.2.3).

7. a2) blok



Blok 7 a2) nastavíme rovnako ako blok 4 v úlohe č. 1 (kap. 3.2.1).

8. a2) blok



Blok 8 a2) nastavíme rovnako ako blok 7 b) v úlohe č. 3 (kap. 3.2.3).

9. a2) blok



Blok 10 a2) nastavíme rovnako ako blok 8 b) v úlohe č. 3 (kap. 3.2.3) alebo ako blok 1 v úlohách č. 1,2 (kap. 3.2.1, 3.2.2), respektíve blok č. 6 v úlohe č. 1 (kap. 3.2.1).

10. a2) blok



Blok 10 a2) nastavíme rovnako ako blok 9 a) v programe č. 3 (kap. 3.2.3), respektíve ako blok 7 v úlohe č. 1 (kap. 3.2.1).

5. b) blok



Blok 5 b) nastavíme rovnako ako blok 1 v úlohách č. 1, 2, 3 (kap. 3.2.1 3.2.2 a 3.2.3).

6. b) blok



7. b) blok



Pomocou bloku 6 b) a 7 b) robot oznámi, že nemá loptičku – **No Object**. V **Action** vyberieme prehratie zvukového súboru (**Sound File**) a **Control** nastavíme na prehrávanie (**Play**). Hlasitosť (**Volume**) dáme na 100, aby bolo robota dobre rozumieť. V ponuke **Function** ponecháme **Repeat** nezaškrtnuté. Keďže chceme aby robot oznámil že nemá loptičku, pomocou výrazu **No Object**, ktorý sa skladá z dvoch slov, tak v bloku 6 b) vo **File** zvolíme zvukový súbor **No** a v bloku 7 b) zvolíme zvukový súbor **Object**. V ponuke **Wait** zaškrtneme **Wait for Completion**.

8. b) blok



Blok 8 b) nastavíme rovnako ako bloky 6a) 6b) v úlohe č. 3 (kap. 3.2.3), respektíve ako bloky 7 a1) a 7 a2) v programe 4 (kap. 3.2.4), alebo ako blok 4 v úlohe č. 1 (kap. 3.2.1).

11. blok



Blok 10 nastavíme rovnako ako blok 10 v úlohe č. 3 (kap. 3.2.3), respektíve blok v úlohe č. 4 (kap. 3.2.4).

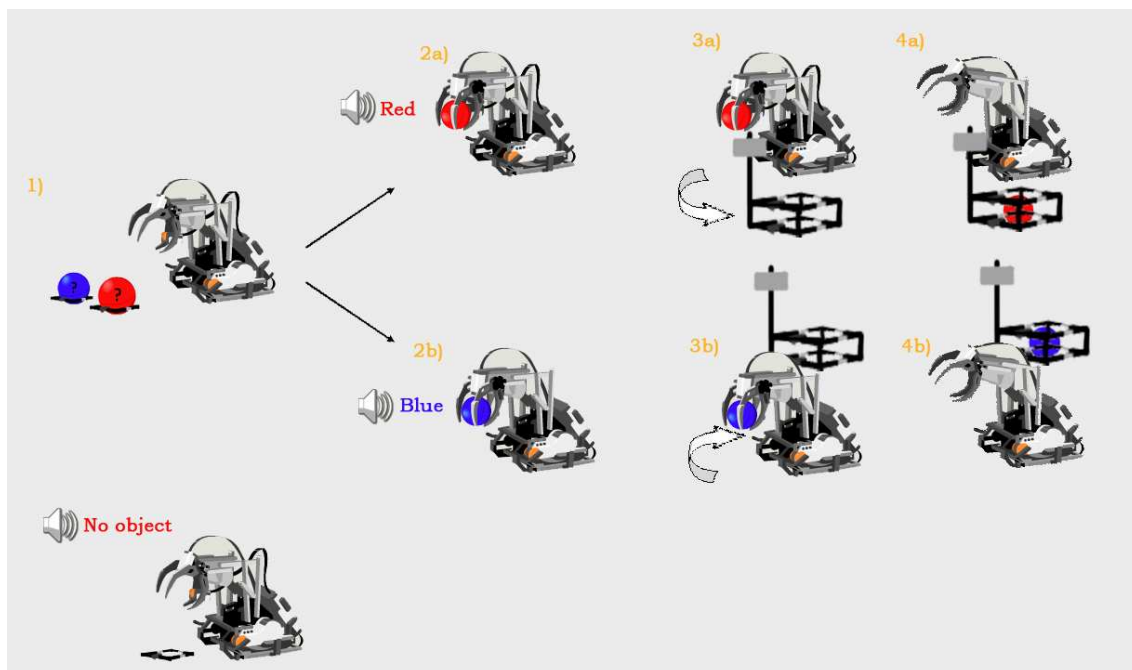
3.2.5 Úloha 5 pre robotické rameno

Vytvoriť robotické rameno, ktoré vie rozoznávať farbu loptičiek pomocou svetelného senzora. Keď je loptička modrá umiestni ju napravo do zásobníka. Ak je loptička červená umiestni ju naľavo do zásobníka. Zásobníky nemusia mať fixnú pozíciu, robot musí vedieť rozpoznať pomocou ultrazvukového senzora kde sa zásobník na loptičky nachádza a podľa toho sa pootočiť. Ak na základe tlakového senzora zistí, že loptičku nedostal, oznámi - No object.

Potrebné akčné členy: tri servomotory, reproduktor

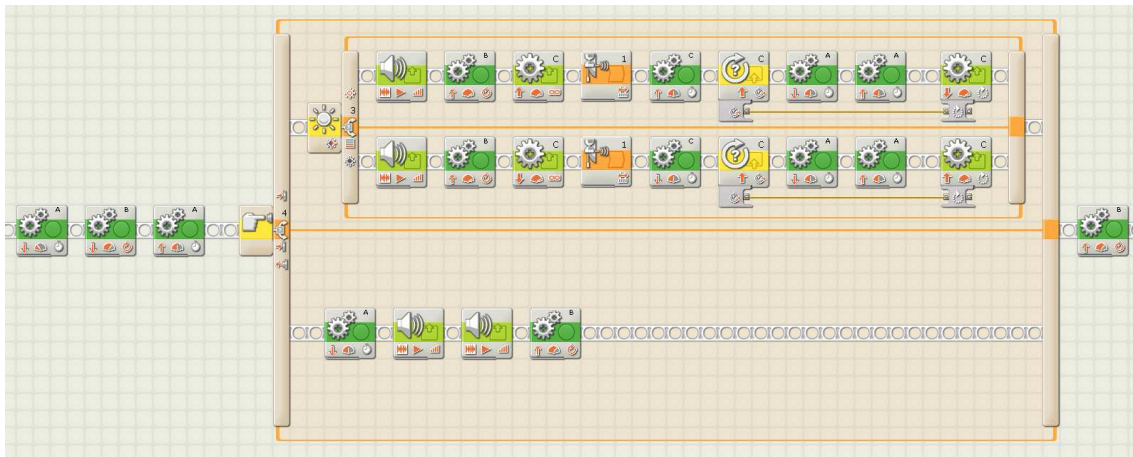
Potrebné senzory: svetelný senzor, tlakový senzor, ultrazvukový senzor, rotačný senzor

Schéma č. 5



Robotické rameno sa zohne po loptičku, uchopí ju do klieští a pomocou svetelného senzora určí či je loptička červená alebo modrá. Ak je loptička modrá povie „Blue“, otočí sa doprava a umiestni ju do zásobníka pre loptičky, pokiaľ je loptička červená povie „Red“, otočí sa doľava a umiestni ju do zásobníka pre loptičky. Robot využíva ultrazvukový senzor na zistenie o koľko sa má pootočiť robotické rameno aby mohol robot umiestniť loptičku do zásobníka. Čo znamená, že rameno sa otáča dovtedy, pokiaľ ultrazvukovým senzorom nezaznamená prekážku, ktorou je zásobník pre loptičky. Potom spätne využíva záznam z rotačného senzora aby vedel o koľko sa musí pootočiť späť aby sa dostal na začiatočnú pozíciu. Ak na základe tlakového senzora zistí, že loptičku nedostal, povie – No object.

Program 5



Na zostavenie programu som použil jeden typ blokov:

- a) blok na ovládanie servomotorov
- b) blok s využitím svetelného senzora
- c) blok s využitím tlakového senzora
- d) blok s využitím rotačného senzora
- e) zvukový blok

1.blok



Blok 1 nastavíme rovnako ako blok 1 v úlohách č. 1, 2, 3, 4 (kap. 3.2.1, 3.2.2, 3.2.3 a 3.2.4).

2.blok



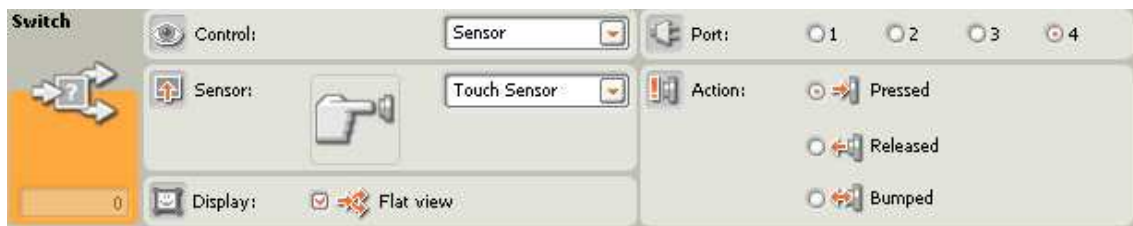
Blok 2 nastavíme rovnako ako blok 2 v úlohách č. 1, 2, 3, 4 (kap. 3.2.1, 3.2.2, 3.2.3 a 3.2.4).

3.blok



Blok 3 nastavíme rovnako ako blok 3 v úlohách č. 1, 2, 3, 4 (kap. 3.2.1, 3.2.2, 3.2.3 a 3.2.4).

4.blok



Blok 4 nastavíme rovnako ako blok 4 v úlohách č. 4 (kap. 3.2.4).

5. a) blok



Blok 5 a) nastavíme rovnako ako blok 5 a) v úlohe 4 (kap. 3.2.4), respektíve ako blok 4 v úlohách č. 2, 3 (kap. 3.2.2 a 3.2.3).

6. a1) blok



Blok 6 a1) nastavíme rovnako ako blok 6 a1) v úlohe 4.2.4, respektíve ako blok 5 a) v úlohách č. 2, 3 (kap. 3.2.2 a 3.2.3).

7. a1) blok



Blok 7 a1) nastavíme rovnako ako blok 7 a1) v úlohe 4 (kap. 3.2.4), respektíve ako blok 6 a) v úlohe 3 (kap. 3.2.3), alebo ako blok 4 v úlohe č. 1 (kap. 3.2.1).

8. a1) blok



Blok 8. a1) nám zaistí pootočenie ramena doľava v horizontálnom smere. Motor ktorý umožňuje otáčanie ramena v horizontálnom smere je pripojený ku kocke NXT prostredníctvom portu C, preto **Port** nastavíme na C. **Direction** nastavíme na dopredu, pretože keď sa motor točí dopredu – rameno sa otáča doľava. **Power** nastavíme na 100. V ponuke **Control – Motor Power** ponecháme neaktivované. **Duration** nastavíme **Unlimited**, čo znamená neobmedzené otáčanie motora. Otáčanie motora obmedzíme v ďalšom bloku.

9. a1) blok



Blok **Wait** nám preruší otáčanie motora. V tomto bloku využijeme ultrazvukový senzor, preto v ponuke **Control** zvolíme **Sensor** a zo senzorov vyberieme Ultrasonic Sensor (ultrazvukový senzor). Ultrazvukový senzor máme pripojený cez port č. 1, preto **Port** nastavíme na 1. **Until** nám určuje po zaznamenaní akej vzdialenosti ultrazvukovým senzorom sa splní podmienka a tým sa program posunie na ďalší blok. **Distance** nastavíme na vzdialenosť menšiu ako 25 cm. Čiže, keď ultrazvukový senzor zaznamená objekt vo vzdialenosti menšej ako 25 cm (objektom bude zásobník loptičiek) posunie sa program na ďalší blok.

10. a1) blok



Potom ako ultrazvukový senzor zaznamená zástavku na zásobníku pre loptičky, musí sa robotické rameno ešte o niečo pootočiť. Dôvodom je excentrické umiestnenie ultrazvukového senzora na robotickom ramene.

Tento blok nám zaistí pootočenie. **Port** zvolíme C. **Direction** nastavíme na dopredu. **Power** nám postačí 70. **Duration** nastavíme tak aby sa rameno točilo ešte 5 sekúnd. **Next Action** nastavíme na **Brake**.

11. a1) blok



Pomocou tohto bloku robot vie koľko spravil zvolený motor otáčok. Z ponuky **Port** vyberieme motor C, pretože chceme zistiť o koľko sa nám pootočilo rameno a koľko otáčok musí vykonať motor opačným smerom aby sa dostal na základnú pozíciu. **Action** nastavíme na **Read**, pretože chceme zistiť o koľko sa nám motor pootočil. **Compare** nemusíme nastavovať, pretože nechceme získanú hodnotu s ničím porovnávať.

12. a1) blok



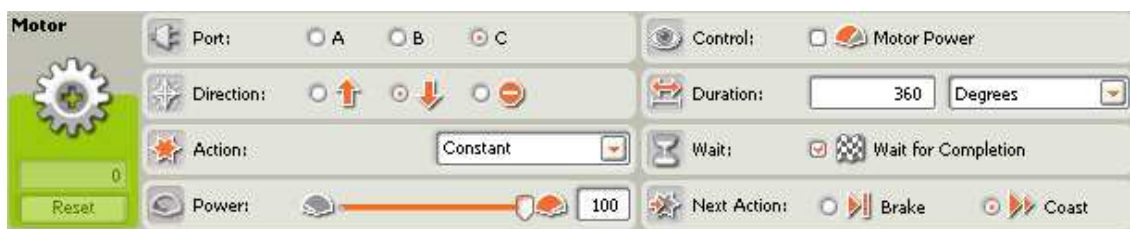
Blok 12 a1) nastavíme rovnako ako blok 9 a1) v úlohe č. 4 (kap. 3.2.4), ako blok 8 a) v úlohe č. 3 (kap. 3.2.3), alebo ako blok 1 v úlohách 1, 2 (kap. č. 3.2.1 a 3.2.2), respektíve blok č. 6 v úlohe č. 1 (kap. 3.2.1).

13. a1) blok



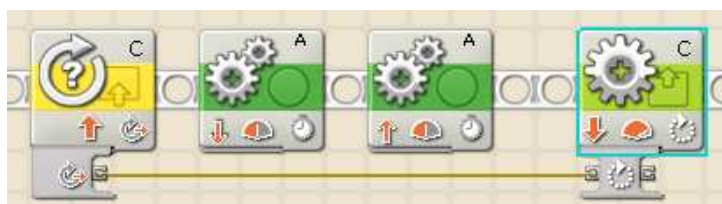
Pomocou tohto bloku kliešte zavrieme. Všetky hodnoty nastavíme ako v bloku 12. a1), až na **Direction**, ktoré nastavíme na dopredu.

14. a1) blok



Tento blok nám umožní vrátenie ramena do pôvodnej polohy. **Port** zvolíme C. **Direction** nastavíme opačne ako v blokoch: 8 a1) a 10 a1), čiže dozadu. **Action** nastavíme **Constant**, čo znamená že motor sa nám bude otáčať konštantnou rýchlosťou, bez akcelerácie. **Power** nastavíme na 100. V ponuke **Control** ponecháme **Motor Power** neaktivované. **Duration** nastavíme na ľubovoľnú hodnotu, pretože motor sa bude otáčať na základe informácii získaných z rotačného senzora – blok. 11 a1).

Informácie z rotačného senzora 11 a1) dostaneme do bloku na ovládanie motora pomocou prepojenia (vid'. Obrázok).



Ďalej **Wait** nastavíme na **Wait for Completion**, čo znamená že ďalší blok sa začne vykonávať až po ukončení aktuálne vykonávaného. **Next Action** nastavíme na **Coast**.

6. a2) blok



Blok 6 a2) nastavíme rovnako ako blok 6 a2) v úlohe 4 (kap. 3.2.4), respektíve ako blok 5 b) v úlohách č. 2, 3 (kap. 3.2.2 a 3.2.3).

7. a2) blok



Blok 7 a2) nastavíme rovnako ako blok 7 a2) v úlohe 4 (kap. 3.2.4), respektíve ako blok 6 b) v úlohe 3 (kap. 3.2.3), alebo ako blok 4 v úlohe č. 1 (kap. 3.2.1).

8. a2) blok



Blok 8 a2) nastavíme rovnako ako blok 8 a1) v tomto programe, až na **Direction**, ktoré nastavíme na dozadu, pretože chceme aby sa rameno otáčalo do pravej strany.

9. a2) blok



Blok 9 a2) nastavíme rovnako ako blok 9 a1) v tomto programe.

10. a2) blok



Potom ako ultrazvukový senzor zaznamená zástavku na zásobníku pre loptičky, musí sa robotické rameno ešte o niečo pootočiť. Dôvodom je excentrické umiestnenie ultrazvukového senzora na robotickom ramene. Keďže ultrazvukový senzor som umiestnil na ľavú stranu robotického ramena, rameno sa musí pootočiť o menšiu vzdialenosť ako v prípade bloku 10 a1). **Port** zvolíme C. **Direction** nastavíme dopredu. **Power** nám postačí 70. **Duration** nastavíme tak aby sa rameno točilo ešte 3 sekundy. Next action nastavíme na **Brake**.

11. a2) blok



Blok 11 a2) nastavíme rovnako ako blok 11. a2) v tomto programe.

12. a2) blok



Blok 12 a2) nastavíme rovnako ako blok 9 a2) v úlohe č. 4 (kap. 3.2.4), ako blok 8 b) v úlohe č. 3 (kap. 3.2.3), alebo ako blok 1 v úlohách č. 1,2 (kap. 3.2.1, 3.2.2), respektíve blok č. 6 v úlohe č. 1 (kap. 3.2.1)

13. a2) blok



Pomocou tohto bloku kliešte zavrieme. Všetky hodnoty nastavíme ako v bloku 12. a2), až na **Direction**, ktoré nastavíme na dopredu.

14. a2) blok



Blok 14 a2) nastavíme rovnako ako blok 14 a1). Nutnosťou je prepojenie blokov 11 a2) s blokom 14 a2).

5. b) blok



Blok 5 b) nastavíme rovnako ako bloky 1, 12 a1), 12 a2) v tomto programe.

6. b) blok



7. b) blok



Bloky 6b) a 7b) nastavíme rovnako ako bloky 6b) a 7b) v programe č. 4 (kap. 3.2.4).

8. b) blok



Blok 8 b) nastavíme rovnako ako blok 8 b) v úlohe č. 4 (kap. 3.2.4).

15. blok



Blok 15 nastavíme rovnako ako blok 10 v úlohách č. 3, 4 (kap. 3.2.3, 3.2.4), respektíve blok v úlohe č. 1 (kap. 3.2.1).

Rôzne modifikácie úlohy č. 5 pre robotické rameno:

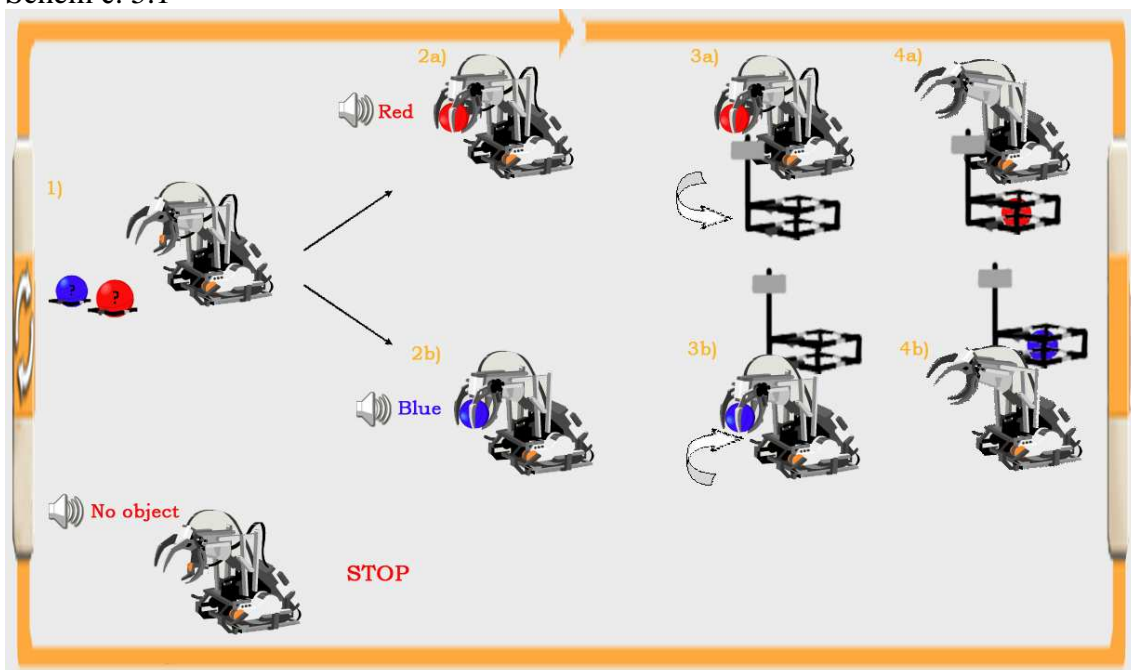
3.2.5.1 Úloha 5.1 pre robotické rameno

Vytvoriť robotické rameno, ktoré bude triediť loptičky rovnako ako v úlohe č. 5, s tým rozdielom, že rameno bude triediť loptičky dovtedy pokiaľ dostane loptičku. Ak už loptičku nedostane povie - No object a program skončí.

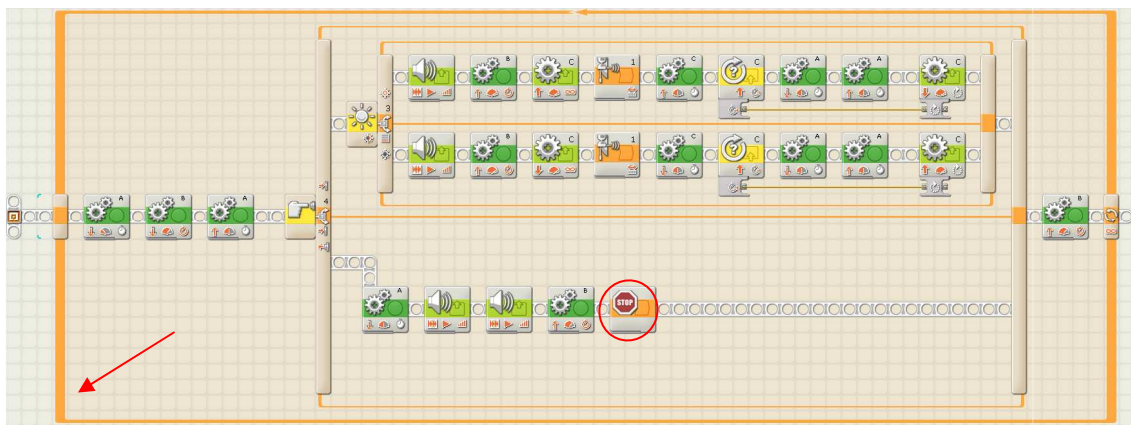
Potrebné akčné členy: tri servomotory, reproduktor

Potrebné senzory: svetelný senzor, tlakový senzor, ultrazvukový senzor, senzor snímajúci počet otáčok

Schéma č. 5.1



Program 5.1



Program 5.1 je presne taký istý ako program 5. Pridané boli iba dva bloky:



Prvým blokom, ktorý sme pridali je blok **Loop**, ktorý nám vytvorí slučku a program v ňom pracuje do nekonečna, respektíve skončí pokiaľ sa splní niektorá z podmienok. To aby program pracoval donekonečna zaistíme pomocou nastavenie v ponuke **Control**, kde nastavíme **Forever**.



Blok ktorý nám zaistí ukončenie programu umiestnime za blok 8 b) v programe 4.2.5. Tento blok nám na tomto mieste zaistí, že potom ako robot oznámi – No object program skončí.

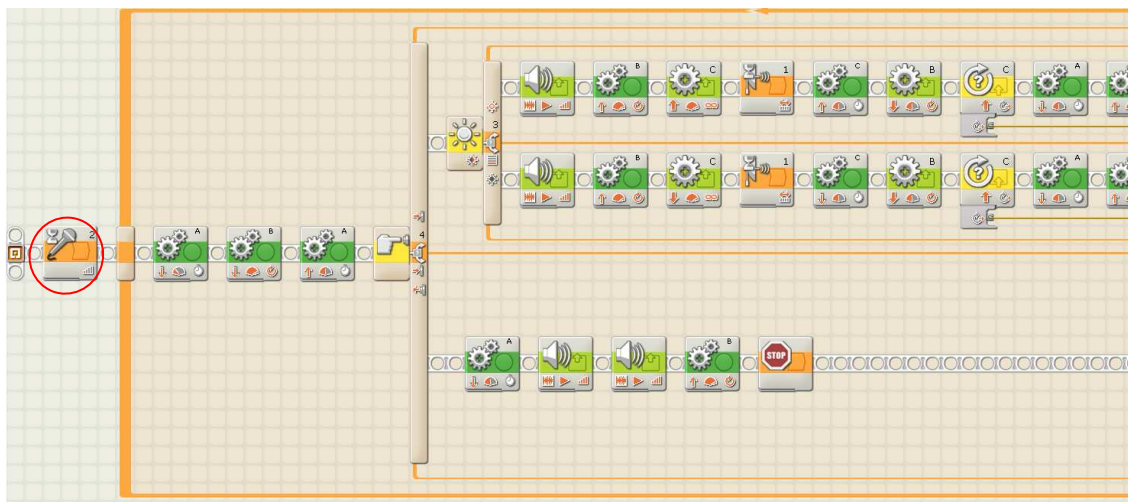
3.2.5.2 Úloha 5.2 pre robotické rameno

Vytvoriť robotické rameno, ktoré bude triediť loptičky rovnako ako v úlohe č. 5.1, s tým rozdielom, že robot začne pracovať až na náš povel, ktorým bude tlesknutie.

Potrebné akčné členy: tri servomotory, reproduktor

Potrebné senzory: svetelný senzor, tlakový senzor, ultrazvukový senzor, senzor snímajúci počet otáčok, zvukový senzor

Program 5.2



Program 5.2 je ten istý ako program 5.1 pridaný bol len jeden blok:

-



Pridaným blokom je blok **Wait**, ktorý nám zaistí, že robot nezačne vykonávať činnosti, pokiaľ mu na to nedáme povel. Povelom bude tlesknutie, ktoré zaznamená pomocou zvukového senzora. Preto v ponuke **Control** vyberieme **Sensor** a to konkrétne **Sound sensor**. Keďže zvukový senzor je pripojený k NXT konce cez port č. 2, tak v ponuke **Port** zvolíme 2. Pomocou **Until** určíme ako hlasno musíme tlesknúť aby robot tento povel zaregistroval. Okolité zvuky môže robot vnímať ako povel, preto nastavíme **Until** na viac ako 30. Čo znamená, že robot bude reagovať iba na tlesknutie a nie na okolité ruchy.

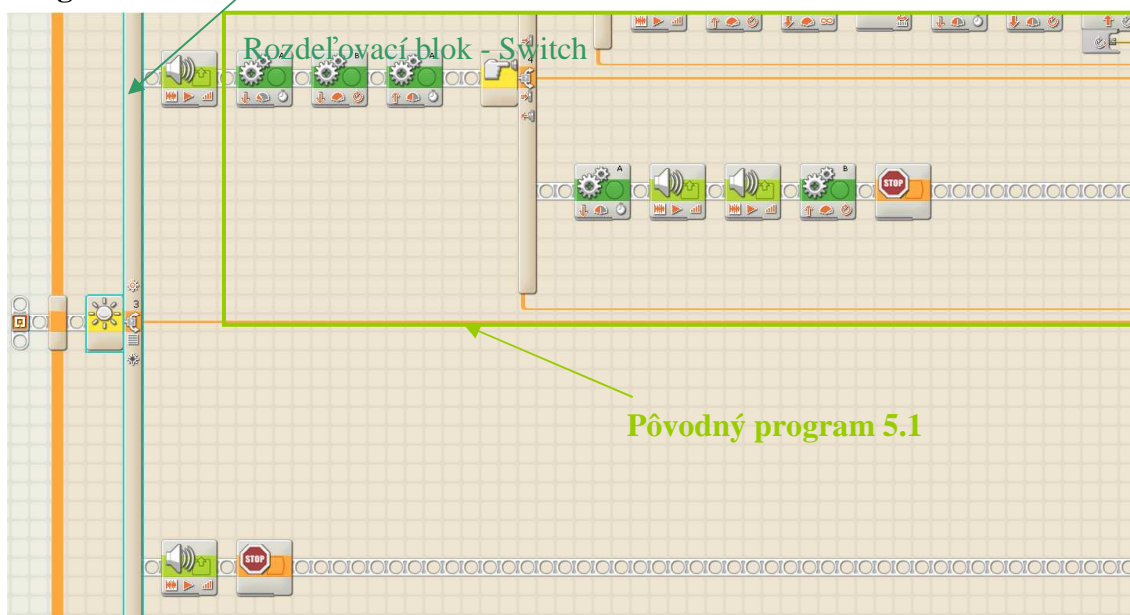
3.2.5.3 Úloha 5.3 pre robotické rameno

Vytvoriť robotické rameno, ktoré bude triediť loptičky rovnako ako v úlohe č. 5.1, s tým rozdielom, že robot bude pracovať iba cez deň, v noci nepracuje.

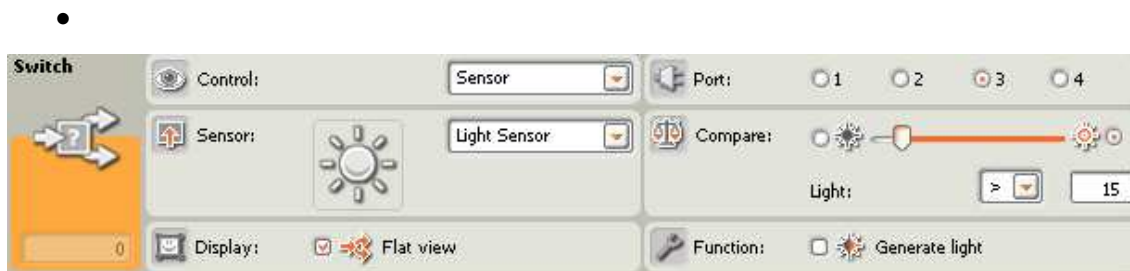
Potrebné akčné členy: tri servomotory, reproduktor

Potrebné senzory: svetelný senzor, tlakový senzor, ultrazvukový senzor, senzor snímajúci počet otáčok,

Program 5.3



Program 5.3 sa skladá z pôvodného programu 5.1 (zelená farba) a nových blokov:



Pomocou bloku **Switch** sa robot rozhodne či bude pracovať – keď je deň (svetlo) alebo nebude pracovať, keď je noc (tma). Toto rozhodnutie bude robiť na základe získaných informácií zo svetelného senzora. Z ponuky **Control** vyberieme **Sensor** a to konkrétne **Light Sensor**. Zvolíme **Port** č. 3, pretože cez ten sme pripájali svetelný senzor k NXT kocke. V ponuke **Compare**, bude robot porovnávať nameranú intenzitu osvetlenia s tou, ktorú mu nastavíme. Keď bude táto intenzita väčšia ako 15, robot ju bude vnímať ako dostatočnú a bude pracovať ďalej. Ak bude okolité osvetlenie menšie robot vykoná inú časť programu, ktorej záverom je ukončenie programu. V ponuke **Function** neaktivujeme **Generate light**, pretože chceme merať intenzitu okolitého osvetlenia.



Keď je splnená podmienka v predchádzajúcom bloku, robot pomocou tohto bloku oznámi, že je deň a pokračuje vo vykonávaní časti programu, ktorou je vlastne program 4.2.5.1. Action nastavíme na **Sound File**, pretože chceme prehrať zvukový súbor. V **Control** zvolíme **Play**. **Volume** nastavíme na požadovanú hlasitosť. V ponuke **Function** ponecháme **Repeat** neaktivované, pretože nechceme aby sa nám zvolený zvukový súbor prehrával dokola. Zo zvukových súborov v ponuke **File** vyberieme **Day**. Vo **Wait** aktivujeme **Wait for Completion**.



Pomocou tohto bloku nám robot oznámi že je noc. Pokiaľ nie je splnená podmienka v bloku **Switch**, tak sa prehrá zvukový súbor **Night**, čím nám robot oznámi, že je noc. Ostatné nastavenia sú také isté ako v predchádzajúcom zvukovom bloku.



Za blok, ktorý nám oznámi, že je noc umiestnime tento blok, ktorý nám na tomto mieste zaistí, že program skončí.

3.3 Hlavná úloha (Triediaca linka)

Podávač loptičiek a triedič loptičiek:

Vytvoriť model triediacej linky, ktorá sa skladá z dvoch robotov, ktorí sú schopní spolu spolupracovať a komunikovať.

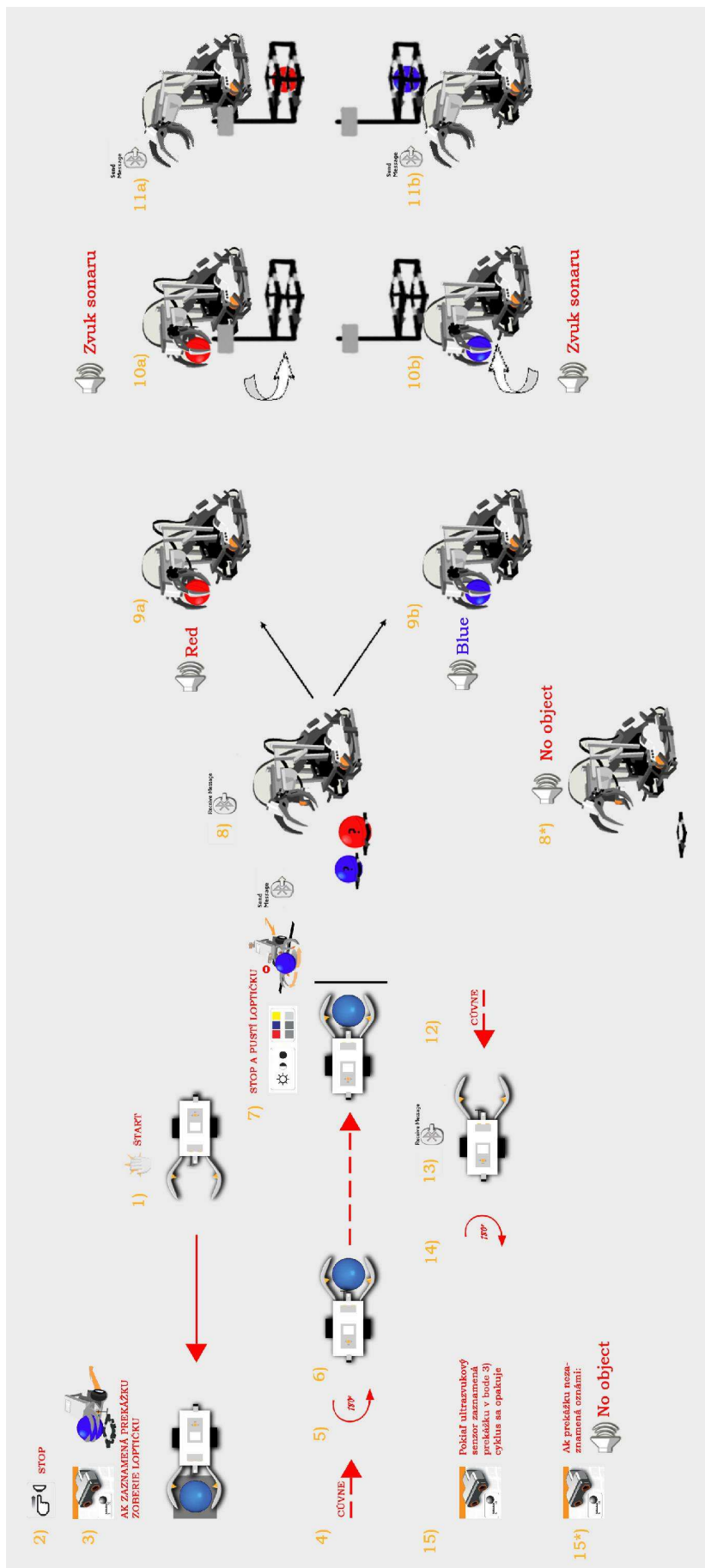
Prvým je podávač loptičiek, ktorého úlohu bude podávať loptičky triediču loptičiek. Podávač loptičiek je motoricky poháňaný robot, ktorý využíva rôzne senzory a akčné členy pre dopravu výrobkov, ktorými sú v tomto prípade loptičky. Podávač loptičiek má možnosť posielat' správy prostredníctvom technológie bluetooth.

Triedič loptičiek je robotické rameno využívajúce mnohé senzory a akčné členy za účelom triedenia výrobkov podľa rôznych kritérií kvality, v tomto prípade loptičiek na základe ich farby. Robot čaká na správu o úspešnom doručení výrobku od podávača loptičiek. Po doručení správy uchopí loptičku a pomocou svetelného senzora posúdi jeho kvalitu, čiže farbu. Na základe farby loptičku potom umiestni do príslušného zásobníka loptičiek, ktorý nemusí mať stabilnú pozíciu. Po umiestnení loptičky do zásobníka, robotické rameno požiada pomocou bluetooth správy podávač loptičiek o ďalšiu loptičku.

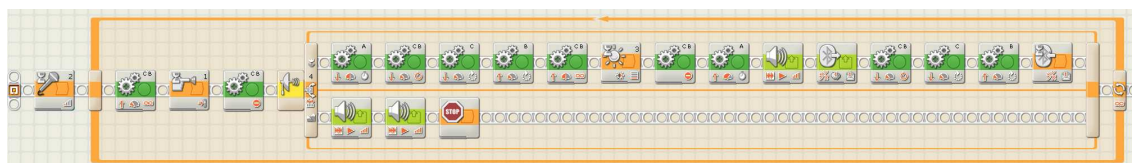
Táto triediaca linka pracuje dovtedy pokiaľ má výrobky – loptičky.

Potrebné akčné členy: tri servomotory, reproduktor, možnosť posielat' bluetooth správy

Potrebné senzory: svetelný senzor, tlakový senzor, ultrazvukový senzor, zvukový senzor, senzor snímajúci počet otáčok



3.3.1 Program pre dopravný vozík (podávač loptičiek)



- a) blok na ovládanie servomotorov
- b) blok s využitím svetelného senzora
- c) blok s využitím tlakového senzora
- d) blok s využitím ultrazvukového senzora
- e) blok s využitím zvukového senzora
- f) blok s využitím rotačného senzora
- g) bluetooth blok
- h) zvukový blok
- i) blok pre vytvorenie cyklu

1. blok



Blok 1 nastavíme rovnako ako blok 1 v úlohe č. 5 (kap. 3.1.5).

2.blok



Blok 2 nastavíme rovnako ako blok 2 v úlohe č. 5 (kap. 3.1.5).

3.blok



Blok 3 nastavíme rovnako ako blok 3 v úlohe č. 5 (kap. 3.1.5).

4.blok



Blok 4 nastavíme rovnako ako blok 4 v úlohe č. 5 (kap. 3.1.5).

5.blok



Blok 5 nastavíme rovnako ako blok 5 v úlohe č. 5 (kap. 3.1.5).

6. a) blok



Blok 6 a) nastavíme rovnako ako blok 6 a) v úlohe č. 5 (kap. 3.1.5).

7. a) blok



Blok 7 a) nastavíme rovnako ako blok 7 a) v úlohe č. 5 (kap. 3.1.5).

8. a) blok



Blok 8 a) nastavíme rovnako ako blok 8 a) v úlohe č. 5 (kap. 3.1.5).

9. a) blok



Blok 9 a) nastavíme rovnako ako blok 9 a) v úlohe č. 5 (kap. 3.1.5).

10. a) blok



Blok 10 a) nastavíme rovnako ako blok 10 a) v úlohe č. 5 (kap. 3.1.5).

11. a) blok



Blok 11 a) nastavíme rovnako ako blok 11 a) v úlohe č. 5 (kap. 3.1.5).

12. a) blok



Blok 12 a) nastavíme rovnako ako blok 12 a) v úlohe č. 5 (kap. 3.1.5).

13. a) blok



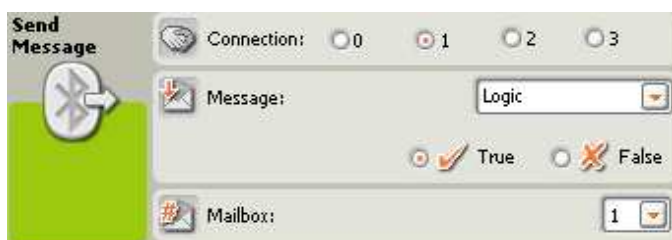
Blok 13 a) nastavíme rovnako ako blok 13 a) v úlohe č. 5 (kap. 3.1.5).

14. a) blok



Pomocou tohto bloku robot oznámi, že loptička je pripravená na triedenie. Oznámi to tým, že prehrá zvukový súbor object. **Action** nastavíme na **Sound File**, **Control** na **Play**. Pomocou **Volume** nastavíme požadovanú hlasitosť. **Repeat** v ponuke **Function** ponecháme neaktívované. Z ponuky **File** vyberieme zvukový súbor na prehratie. V tomto prípade je to **Object**. Vo **Wait** aktivujeme **Wait for Completion**.

15. a) blok



Tento blok nám slúži na odosielanie správ pomocou technológie bluetooth. Podávač loptičiek tak oznámi robotickému ramenu, že má pripravenú loptičku na triedenie. V **Connection** máme možnosť zvoliť jeden zo štyroch kanálov, cez ktoré budú roboti spolu komunikovať. Prvý, druhý a tretí kanál slúžia na komunikáciu medzi robotom, ktorý je zvolený ako Master s robotmi, ktorým bola pridelená hodnota Slave. Robot, ktorý je prepojený ako Slave má možnosť s Master-om komunikovať iba cez nultý kanál. Keďže podávač loptičiek je v tomto prípade Master, v **Connection** nastavíme prvý kanál. V ponuke **Message** zvolíme typ správy. Z ponuky vyberieme **Logic** a hodnotu **True**. **Mailbox** nastavíme na 1.

16. a) blok



Blok 16 a) využijeme na cúvnutie robota. Nastavenie tohto bloku je identické s blokom 7 a) v tomto programe.

17. a) blok



Blok 17 a) nastavíme rovnako ako blok 8 a) v tomto programe.

18. a) blok



Blok 18 a) nastavíme rovnako ako blok 9 a) v tomto programe.

19. a) blok



Na základe tohto bloku podávač loptičiek čaká na spätnú správu od robotického ramena, potom ako umiestni loptičku do zásobníka na loptičky. V bloku **Wait** zvolíme **Control Sensor** a to konkrétne **Receive Message**. **Message** nastavíme na **Logic** a **Compare to True**. Čiže ak bude správa **True**, program prejde na ďalší blok. **Mailbox** zvolíme 1.



Keďže sme použili cyklus, tak všetky bloky a) sa vykonávajú dovtedy, pokiaľ je splnená podmienka v 5 bloku!

Ak podmienka nie je splnená, program pokračuje blokmi:

6. b) blok



7. b) blok



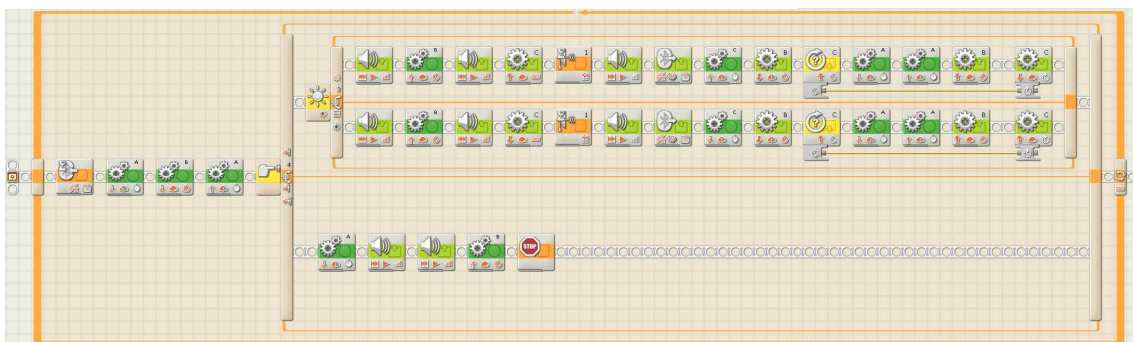
Bloky 6 b) a 7 b) nastavíme rovnako ako bloky 6 b) a 7 b) v programe č. 5 (kap. 3.15)

8. b) blok



Blok Stop zastaví program.

3.3.2 Program pre robotické rameno (triedič loptičiek)



Na zostavenie programu som použil tieto typy blokov:

- a) blok na ovládanie servomotorov
- b) blok s využitím svetelného senzora
- c) blok s využitím tlakového senzora
- d) blok s využitím ultrazvukového senzora
- e) blok s využitím rotačného senzora
- f) bluetooth blok
- g) zvukový blok
- h) blok pre vytvorenie cyklu

1.blok



Na základe nastavenia prvého bloku robotické rameno čaká kým dostane správu cez **bluetooth** od podávača loptičiek o tom, že môže začať triediť. V bloku **Wait** zvolíme **Control Sensor** a to konkrétne **Recieve Message**. **Message** nastavíme na **Logic** a **Compare to True**. Čiže ak bude správa **True**, program prejde na ďalší blok. **Mailbox** zvolíme 1.

2. blok



Blok 2 nastavíme rovnako ako blok 1 v úlohe č. 5 (kap. 3.2.5).

3.blok



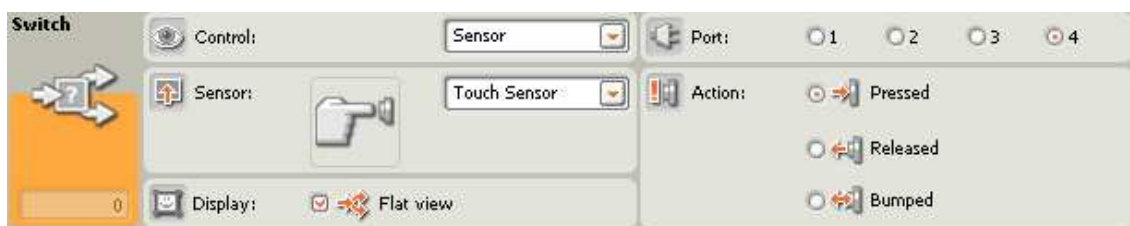
Blok 3 nastavíme rovnako ako blok 2 v úlohe č. 5 (kap. 3.2.5).

4.blok



Blok 4 nastavíme rovnako ako blok 3 v úlohe č. 5 (kap. 3.2.5).

5.blok



Blok 5 nastavíme rovnako ako blok 4 v úlohe č. 5 (kap. 3.2.5).

6. a) blok



Blok 6 a) nastavíme rovnako ako blok 5 a) v úlohe č. 5 (kap. 3.2.5).

7. a1) blok



Blok 7 a1) nastavíme rovnako ako blok 6 a1) v úlohe č. 5 (kap. 3.2.5).

8. a1) blok



Blok 8 a1) nastavíme rovnako ako blok 7 a1) v úlohe č. 5 (kap. 3.2.5).

9. a1) blok



Tento blok nám umožní aby triedič loptičiek vydával zvuk sonaru, počas toho ako hľadá zásobník loptičiek. **Action** nastavíme na **Sound File**. **Control** na **Play**, pretože chceme zvukový súbor prehrať. **Volume** zvýšime na 100, aby bolo počuť sonar aj cez hluk

motorov. Vo **Function** zapneme **Repeat**, pretože chceme aby sa zvuk sonaru prehrával dokola. Vo **File** vyberieme zvukový súbor **Sonar**.

10. a1) blok



Blok 10 a1) nastavíme rovnako ako blok 8 a1) v úlohe č. 5 (kap. 3.2.5).

11. a1) blok



Blok 11 a1) nastavíme rovnako ako blok 9 a1) v úlohe č. 5 (kap. 3.2.5).

12. a1) blok



Keď sa splní podmienka v predchádzajúcom bloku a robotické rameno nájde zásobník loptičiek, tento blok nám zaistí zastavenie prehrávania zvuku sonaru. **Action** nastavíme na **Sound File** a v **Control** aktivujeme **Stop**.

13. a1) blok



Potom ako robotické rameno nájde zásobník loptičiek, pošle správu podávaču loptičiek aby mu pripravil ďalšiu loptičku. To nám zaistí tento blok. Keďže triedič loptičiek je v tomto prípade ako Slave, môže poslať podávaču loptičiek správu len prostredníctvom

0 kanálu. Preto **Connection** nastavíme na 0. V ponuke **Message** zvolíme typ správy. Vyberieme **Logic** a hodnotu **True**. **Mailbox** nastavíme na 1.

14. a1) blok



Blok 14 a1) nastavíme rovnako ako blok 10 a1) v úlohe č. 5 (kap. 3.2.5).

15. a1) blok



Tento blok nám zaistí mierne zohnutie ramena, aby loptička nepadala z výšky. **Port** zvolíme B. **Direction** nastavíme dozadu, pretože vtedy sa rameno posúva smerom dole. **Power** nastavíme na ľubovoľnú hodnotu, postačí 60. **Duration** nastavíme na 10 otáčok. Aktivujeme **Wait for Completion** a **Next action** ponecháme na **Brake**.

16. a1) blok



Blok 16 a1) nastavíme rovnako ako blok 11 a1) v úlohe č. 5 (kap. 3.2.5).

17. a1) blok



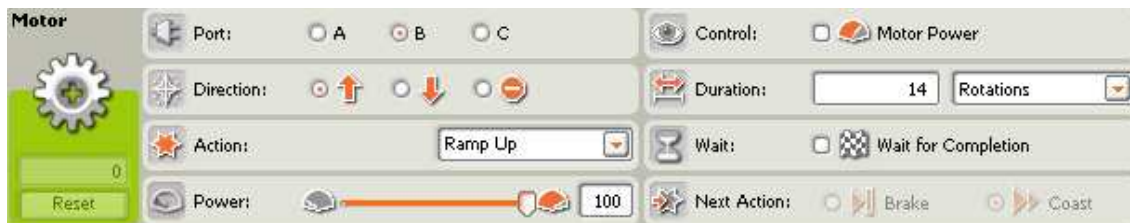
Blok 17 a1) nastavíme rovnako ako blok 12 a1) v úlohe č. 5 (kap. 3.2.5).

18. a1) blok



Blok 18 a1) nastavíme rovnako ako blok 13 a1) v úlohe č. 5 (kap. 3.2.5).

19. a1) blok



Vďaka tomuto bloku sa rameno dostane do pôvodnej vertikálnej polohy. **Port** zvolíme B, **Direction** dopredu. **Action** nastavíme na **Ramp Up**, čo znamená, že rýchlosť motoru sa bude meniť - motor bude akcelerovať. **Power** nastavíme na 100. **Duration** nastavíme na 14 otáčok, pretože o toľko sa ešte musí rameno zdvihnúť aby sa dostalo do pôvodnej výšky.

20. a1) blok



Blok 20 a1) nastavíme rovnako ako blok 13 a1) v úlohe č. 5 (kap. 3.2.5).

Nezabudnúť na prepojenie blokov 16 a1) a 20 a1).

7. a2) blok



Blok 7 a2) nastavíme rovnako ako blok 6 a2) v úlohe č. 5 (kap. 3.2.5).

8. a2) blok



Blok 8 a2) nastavíme rovnako ako blok 7 a2) v úlohe č. 5 (kap. 3.2.5).

9. a2) blok



Blok 9 a2) nastavíme rovnako ako blok 9 a1) v tomto programe.

10. a2) blok



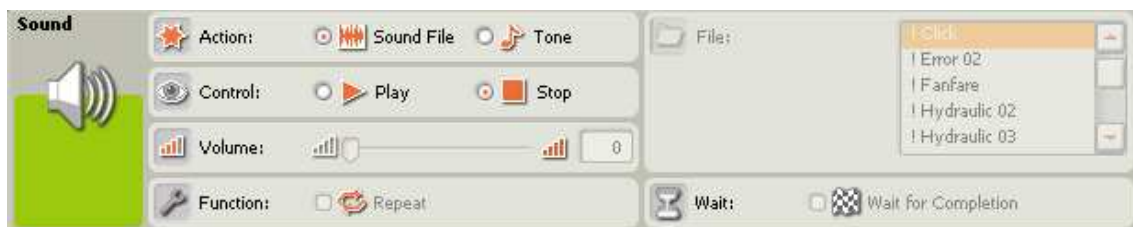
Blok 10 a2) nastavíme rovnako ako blok 8 a2) v úlohe č. 5 (kap. 3.2.5).

11. a2) blok



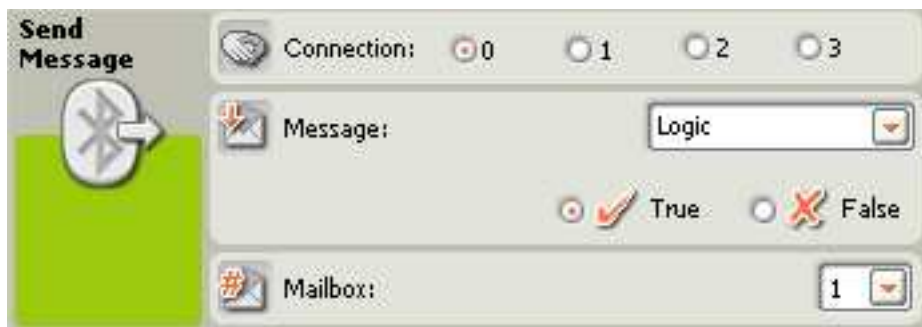
Blok 11 a2) nastavíme rovnako ako blok 9 a2) v úlohe č. 5 (kap. 3.2.5).

12. a2) blok



Blok 12 a2) nastavíme rovnako ako blok 12 a1) v tomto programe.

13. a2) blok



Blok 13 a2) nastavíme rovnako ako blok 13 a1) v tomto programe.

14. a2) blok



Blok 14 a2) nastavíme rovnako ako blok 10 a2) v úlohe č. 5 (kap. 3.2.5).

15. a2) blok



Tento blok nám zaistí mierne zohnutie ramena, aby loptička nepadala z výšky. **Port** zvolíme B. **Direction** nastavíme dozadu, pretože vtedy sa rameno posúva smerom dole. **Power** nastavíme na ľubovoľnú hodnotu, postačí 60. **Duration** nastavíme na 10 otáčok. Aktivujeme **Wait for Completion** a **Next Action** ponecháme na **Brake**.

16. a2) blok



Blok 16 a2) nastavíme rovnako ako blok 11 a2) v úlohe č. 5 (kap. 3.2.1), respektíve ako blok 16 a1) v tomto programe.

17. a2) blok



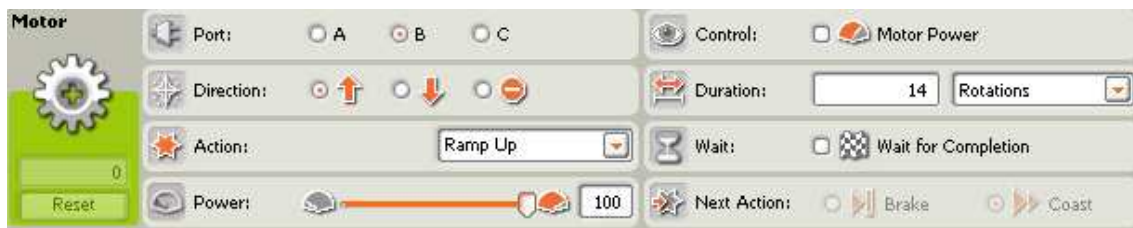
Blok 17 a2) nastavíme rovnako ako blok 12 a2) v úlohe č. 5 (kap. 3.2.5).

18. a2) blok



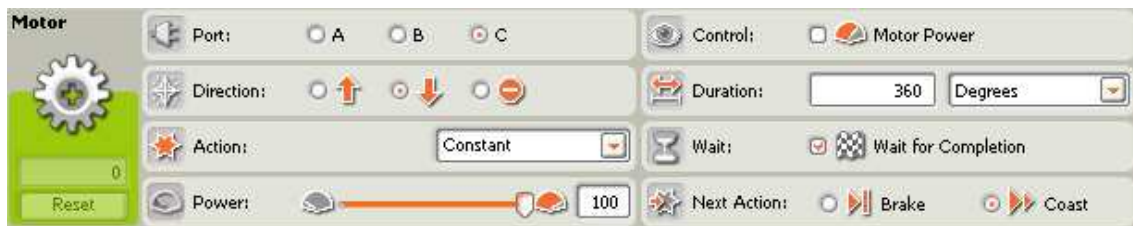
Blok 18 a2) nastavíme rovnako ako blok 13 a2) v úlohe č. 5 (kap. 3.2.5).

19. a2) blok



Blok 19 a2) nastavíme rovnako ako blok 19 a1) v tomto programe.

20. a2) blok



Blok 20 a2) nastavíme rovnako ako blok 14 a2). Nutnosťou je prepojenie blokov 16 a2) s blokom 20 a2).

6. b) blok



Blok 6 b) nastavíme rovnako ako blok 5 b) v úlohe č. 5 (kap. 3.2.5).

7. b) blok



8. b) blok



Bloky 7b) a 8b) nastavíme rovnako ako bloky 6b) a 7b) v programe č. 5 (kap. 3.2.5).

9. b) blok



Blok 9b) nastavíme rovnako ako blok 8b) v programe č. 5 (kap. 3.2.5).

10. b) blok



Blok Stop zastaví program.

ZÁVER

Hlavnými cieľmi práce bolo preskúmanie možností riadenia robotov stavebnice Lego Mindstorms a vytvorenie podrobného popisu vypracovania jednotlivých úloh, ktoré môžu slúžiť ako návod na laboratórne cvičenia k predmetom spätým s témou práce.

Výsledkom práce je podrobný popis technických informácií, ktoré zahŕňajú všetky dostupné senzory a akčné členy. V jednotlivých kapitolách sú rozobraté možnosti využitia senzorov a akčných členov, ich nastavenia a výhody. Časť práce sa venuje programovaniu v programe Lego Mindstorms NXT. Spomenuté sú všetky programovacie bloky. Najväčšia časť práce je praktická časť, v ktorej sú rozpísané jednotlivé úlohy pre robotov. Každá úloha obsahuje svoje zadanie, schému v ktorej je podrobne vysvetlený algoritmus riešenia jednotlivých úloh, zoznam použitých senzorov a akčných členov, samotný program, zoznam použitých programovacích blokov a podrobný popis nastavení jednotlivých blokov. Celkový počet vytvorených a podrobne rozpracovaných úloh je 13. Z toho je 5 úloh určených pre dopravný vozík (podávač loptičiek), 5 úloh a 2 pod úlohy pre robotické rameno (triedič loptičiek) Hlavná úloha (triediaca linka), je výsledkom prepojenia jednotlivých úloh pre dopravný vozík a robotické rameno.

Podrobne vypracované úlohy sú vhodné ako návod na laboratórne cvičenia k predmetom spätým s témou práce.

LITERATÚRA

- [1] Mindstorms education – Metodika NXT © 2006 The Lego Group