

Tento vzdělávací materiál vznikl v rámci projektu  
CZ.02.3.68/0.0/0.0/16\_036/0005322 **Podpora rozvíjení inforatického myšlení.**



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



Podléhá licenci Creative commons Uveďte původ-Zachovejte licenci 4.0



# Práce s robotickými stavebnicemi

Jiří Hrbáček  
Martin Kučera  
Zdeněk Hodis

# 1. Úvod do robotických stavebnic

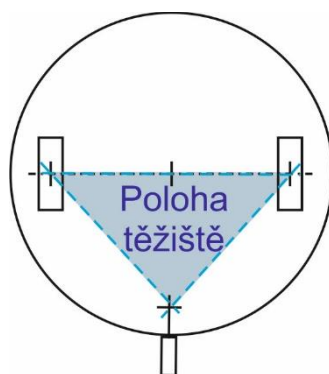


S roboty a robotickými stavebnicemi se dnes setkávají žáci a učitelé ZŠ ve výuce nebo v zájmových kroužcích. Existuje nepřehledné množství robotických systémů od různých dodavatelů a výrobců. Robotické hračky se liší konstrukcí, komplexností a svou cenou. V tomto studijním materiálu budou v prvních kapitolách popsány obecné způsoby jak vytvořit a zprovoznit robotický systém. Budou vysvětleny části, z kterých se robot skládá a způsob jakým pracuje.

V druhé části tohoto studijního materiálu bude popsán způsob práce s robotickou stavebnicí LEGO® MINDSTORMS™, Education EV3. Jedná se o poměrně známé a rozšířené robotické řešení od celosvětově známé fy LEGO. Toto komerční řešení se objevuje ve výuce na 2. stupni ZŠ a je vhodné pro seznámení starších žáků s problematikou robotiky a programování.

## 2. Pohyb robota v prostoru

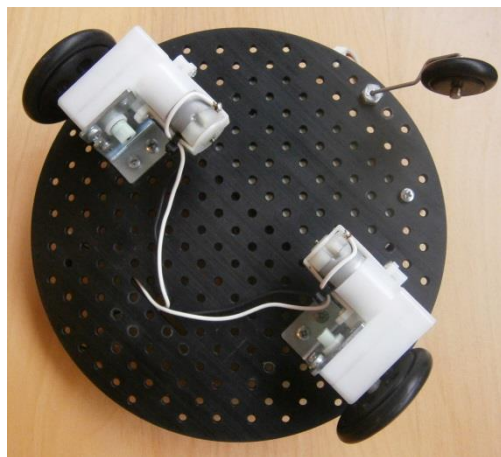
Mobilní roboti patří asi k nejzajímavějším robotům, které děti lákají. Dělíme je na indoorové a outdoorové. Indoorové jsou roboti pro jízdu „doma“. Nejsou určeni pro náročné terény. Vyžadují rovný podklad bez větších nerovností. Mezi takové patří také robot, který je osazen na kruhové desce dvěma motory s opěrným kolečkem. Takovému podvozku se říká podvozek diferenciální (Friedmann, 2018).



Obrázek 2.1 - Diferenciální podvozek

Pro výuku jsou nejvhodnější. Nejen ovšem pro výuku. Tělo robota tvoří podvozek, který se snadno řídí, dobře se vypočítává jízda po zadané dráze (odometrie) a je hojně užíván i v životě kolem nás. Například invalidní vozík, tank (nepotřebuje opěrné kolečko, má pásy), segway, stavební stroj UNC apod.

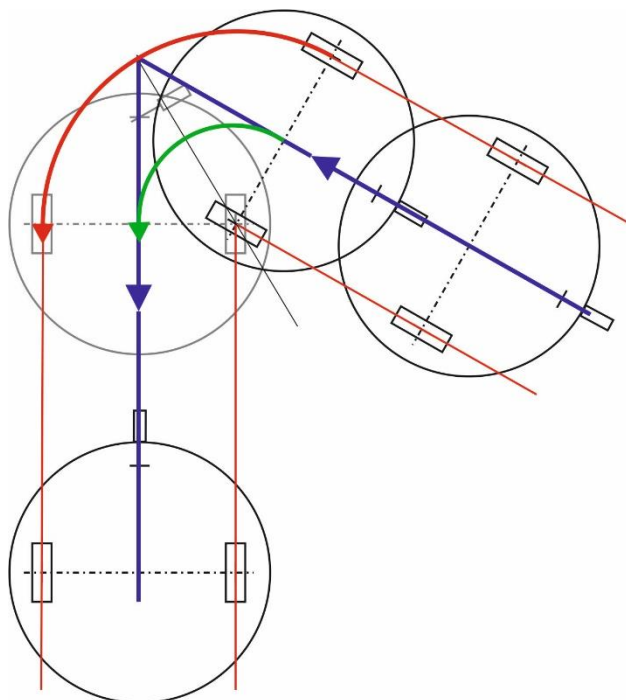
Jak může takový podvozek vypadat v praxi, vidíme na – obr. 2.2 (Hrbáček - <http://hses.cz/roboticka-sada-pro-vyvoj-a-vyuku/>).



*Obrázek 2.2 – Podvozek H&S electronic systém*

## 2.1. Použití desky bipolárních spínačů pro jízdu robota

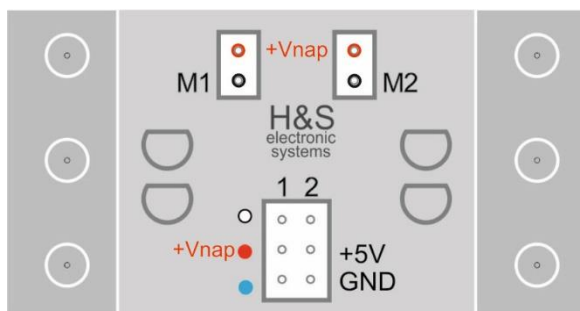
Budeme-li motory řídit tak, že motor pojede vpřed, nebo vůbec nepojede, pak může jet podvozek pouze vpřed – pokud levý i pravý motor jede vpřed, nebo zatáčet vpravo, zastavíme-li pravý motor, nebo vlevo, zastavíme-li levý motor. Dráha kol i podvozku při tomto způsobu řízení podvozku je následující.



*Obrázek 2.3 – Jízda diferenciálního podvozku pod řízením desky spínačů*

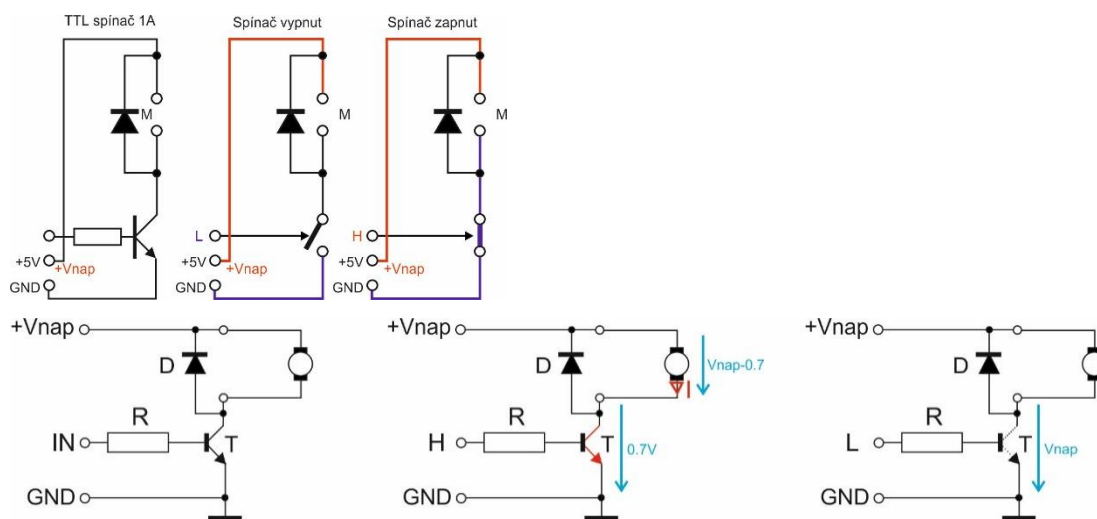
Na obrázku je rozkreslena jízda podvozku při tomto způsobu řízení. Podvozek jede jen vpřed. Červeně je zakreslena dráha kol. Modře pak požadovaná dráha

jízdy. Vidíme, že v rozích nejede podvozek přesně po požadované dráze (požadovaná dráha je dráha středu podvozku). Z obrázku tedy vyplývá, že při tomto způsobu řízení podvozku musí být poloměr zatáček dráhy minimálně polovina vzdálenosti kol od sebe (zelený rádius). Dráha vnějšího kola v zatáčce má pak poloměr minimálně rovný vzdálenosti kol od sebe (červený rádius).



**Obrázek 2.4 - Deska bipolárních spínačů 1A**

Deska spínačů obsahuje dva stejné spínače, jejichž činnost je na následujícím obrázku. Spínáme-li zařízení, u kterého je potřeba aby bylo +12V na konkrétním vývodu (motory podvozku se otáčejí na jednu, nebo druhou stranu podle toho, kde je +12V), pak musíme na spínaný výstup spínače správně spínané zařízení připojit (+12V ke kraji desky a GND ke středu desky jak je naznačeno na obrázku).



**Obrázek 2.5 a 2.6- Činnost bipolárního spínače**

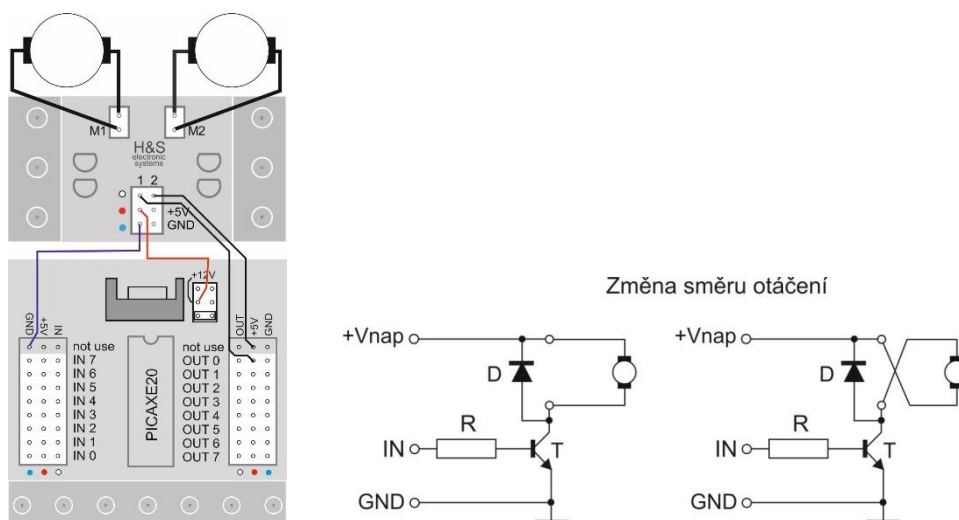
Je-li na signálovém vstupu úroveň L, je spínač (bipolární tranzistor) vypnutý a kvýstupu M je připojeno pouze kladné napájecí napětí. Motor nejede.

Přivedeme-li na signálový vstup spínače úroveň H, spínač se zapne a k výstupu M je přivedeno i +5V - GND a motor se rozběhne. Dioda zde má pouze ochrannou funkci.

**Desku spínačů jako jedinou, lze připojit svorkou +5V (+Vnap) k vyššímu napětí (až 36 V).** Motory podvozku jsou určeny primárně pro napájení 12 V. Připojíme-li je k +5V, budou se otáčet pomaleji. Obvykle je tedy připojujeme k +12V.

### Připojení desky procesoru a desky spínačů k diferenciálnímu podvozku a jeho seřízení:

Po montáži podvozku je třeba správně nastavit kola podvozku. Toho se docílí povolením motorů, jejich pootočením a opětným utažením desce podvozku. Dají se malinko pootáčet. Potřebujeme nastavit kola s motory tak, aby podvozek jel co nejrovněji.



**Obrázek 2.7 – Připojení motorů k desce spínačů pro správný směr jízdy**

Nastavování provádíme s namontovanou deskou spínačů, aby byly motory napájeny stejně, jako při řízení deskou procesoru.

Propojíme motory na výkonové výstupy M1 a M2 desky spínačů. Oba vstupy signálů 1 a 2 na úroveň H (+5V).

Motory se musí otáčet tak, že podvozek jede vpřed (ostruhové kolečko je vzadu). Otáčeli-li se některý z motorů opačně, je třeba otočit konektor připojení motoru.

Následně položíme podvozek jedním kolem na čáru (nakreslenou na zemi, nebo nalepenou izolační páskou). Osa kol by měla být přesně kolmá k této čáře. Proto je dobré si tuto kolmici na zemi také naznačit.

Zapneme vypínač napájení na baterý-packu a robot by měl jet rovně. Nejedeli, snažíme se nastavit motory tak, aby jel co nejrovněji (jak bylo popsáno výše). Důležité je mít v robotovi dobré (kvalitní a nabitě) baterie.

Že oba motory nejedou stejně rychle, je dáno vlastností motoru a jeho kvalitou. Čím je k motoru připojené větší napětí, tím se motor otáčí rychleji. V praxi je velmi obtížné vyrobit cíleně dva motory s naprosto stejnými vlastnostmi.

### **Měření základních parametrů jízdy podvozku:**

Pro další práci s podvozkem je třeba změřit základní parametry jízdy podvozku. Rychlost jízdy vpřed a úhlovou rychlost při zatáčení vlevo a vpravo.

Rychlost jízdy vpřed: Žák si stopne, jak dlouho bude robotovi trvat ujet dráhu dlouhou 3m. Čas v sec podělíme 3 a dostaneme rychlost ( $v[\text{mm/ms}]$ ).

Úhlová rychlost: Aby podvozek zatáčel, připojíme jeden vstupní signál desky spínačů na H (+5V) a druhý do L (GND). Pro druhý směr otáčení to prohodíme.

Změříme nejdříve dobu, za kterou se podvozek desetkrát otočí dokola o  $360^\circ$ . Za deset otáček se podvozek otočí celkem o  $3600^\circ$  proto tuto dobu vydělíme 3,600 a dostaneme dobu (1 stupeň/ms), za kterou se podvozek otočí o  $1^\circ$ . Tuto hodnotu změříme pro otáčení vlevo i vpravo. Mohou se trochu lišit (nejede-li podvozek přesně rovně).

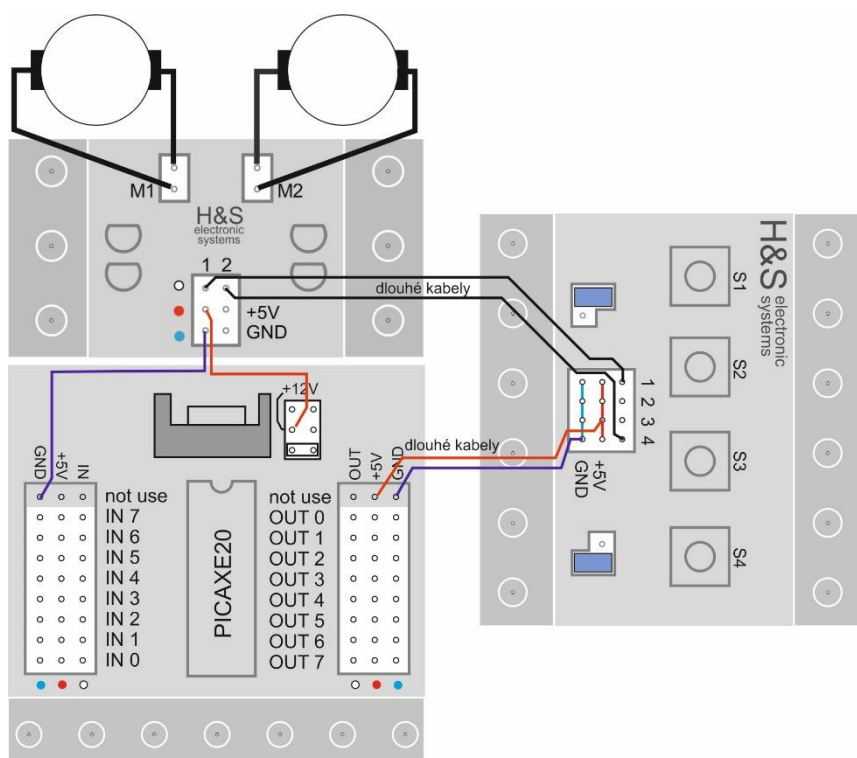
Hodnoty si žáci poznačí a budou s jejich pomocí programovat jízdu po zadané dráze.

### **Nácvik způsobu řízení podvozku:**

Ke vstupům desky spínačů připojíme výstupy 1 a 4 desky tlačítek. Díky tomu můžeme ovládat jízdu podvozku tlačítky.

Aby bylo ovládání podvozku intuitivní a názorné, drží-li žák tlačítka v ruce, levá ruka ovládá zapínání levého motoru podvozku a pravá ruka pravého motoru podvozku.

Žáci vyzkouší, jak se podvozek řídí. Připravíme jim slalom, vytyčíme dráhu kužely, láhvemi apod. a necháme je takto řídit podvozek jako autíčko na ovládání. Necháme je jezdit kolem po čáře (klikatě se zatázkami, jakých je schopen tento podvozek). Nakonec je necháme jezdit po drahách, pro které budou v budoucnu psát programy.



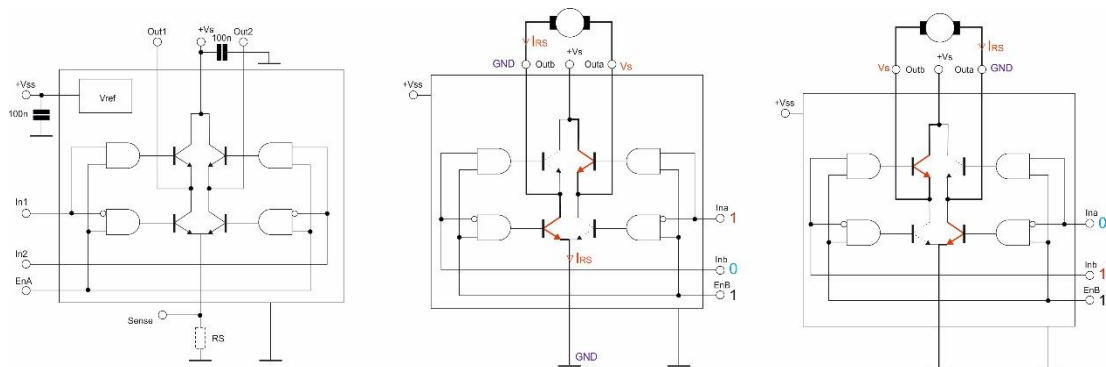
**Obrázek 2.8 – Připojení motorů k desce spínačů pro správný směr jízdy**

Spolehlivé nacvičení jízdy s diferenciálním podvozkem je velmi důležitý pochopení, jak se podvozek řídí a jak jej pak programovat.

## 2.2. Použití desky 2 x H-bridge pro jízdu robota

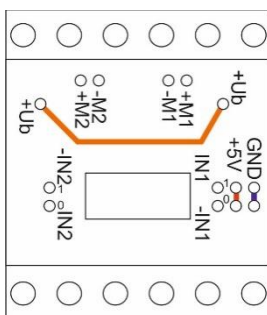
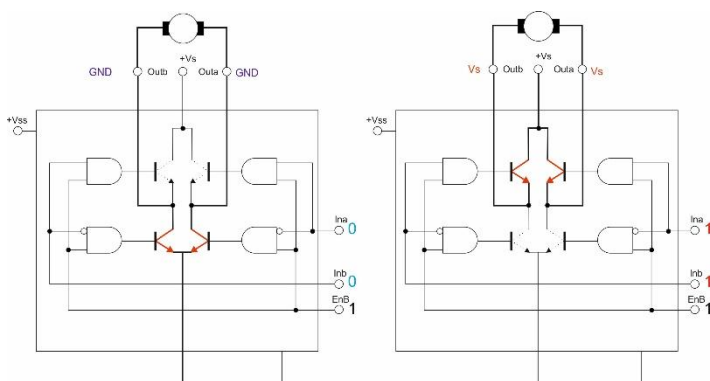
H-bridge je super deska, která umožňuje napájet motory opět až 36 V, a navíc umí měnit směr otáčení motoru. Dá se k ní připojit více motorů a umí zastavovat motor brzděním, nebo dojezd setrvačností. Jejím nasazením není jen řízení komutátorových motorů, ale tato tematika přesahuje rozsah této publikace.

Omezíme se tedy pouze na její použití s komutátorovým motorem a trvale nastavenou hodnotou vstupu  $En = 1$ .



**Obrázek 2.9 – základní činnost H-bridge, reverzace otáček**

Na obrázku je zjednodušené vnitřní zapojení H-bridge, na kterém si snadno vysvětlíme jeho činnost. Silnější čarou je nakresleno, kudy teče proud v H-bridge. Při běhu motoru, je to cesta od  $+Vs$  po GND. Při brždění jsou vývody motoru připojeny ke GND, nebo k  $+Vs$ .



**Obrázek 2.10 – činnost H-bridge – brždění, deska H-bridge H&S electronic systems**



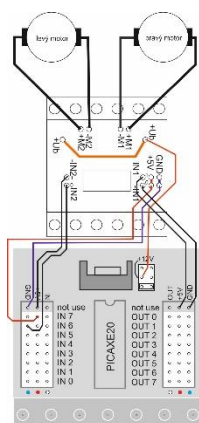
+Ub odpovídá +Vs je připojení napájecích napětí k motorům (max. 36 V) na H-bridgi. M1+, M1-, M2+ a M2- jsou výstupy dvou H-bridgů Outb a Outa. IN1, IN1-, IN2 a IN2- jsou řídicí signály Ina a Inb těchto H-bridgů. Na desce je vstup Ena a Enb trvale připojen k úrovni 1.

Činnost desky H-bridge je tedy následující:

IN	IN-	činnost připojeného motoru
0	0	motor stop
0	1	motor vpřed
1	0	motor vzad
1	1	motor stop

Aby bylo snadné se orientovat na ve vývodech desek H-bridge (to platí obecně). Můžeme si tuto situaci velmi zjednodušit. Levá část desky ovládá levý motor a pravá část pak motor pravý. Přední řídicí signál bude-li v H, spustí motor vpřed, zadní řídicí signál v H zapne motor pro jízdu vzad (za předpokladu, že je vždy druhý řídicí signál v L). Pokud bude přední i zadní řídicí signál v H, nebo v L, pak se motory zastaví.

Aby tomu tak bylo, zapojíme podvozek s deskou H-bridge i procesoru podle následujícího obrázku.

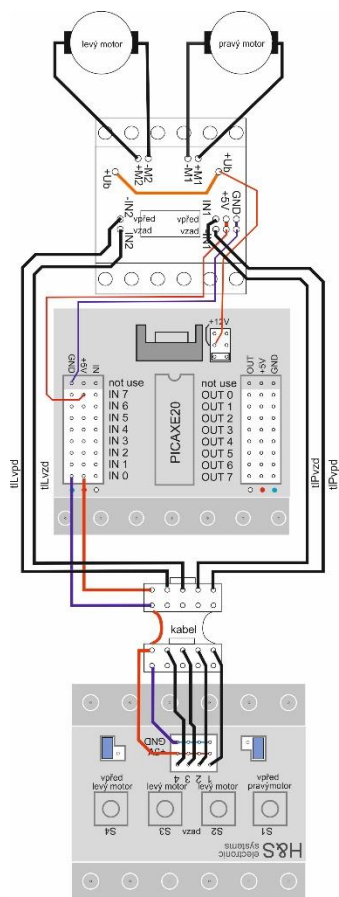


**Obrázek 2.11 – deska H-bridge H&S electronic systems**

Motory připojíme k výstupům M1 a M2. Pokud se motory točí tak, že jede příslušné kolo vpřed, je motor k H-bridge připojen správně. Pokud se některý

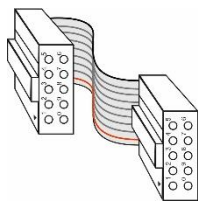
z motorů bude točit opačně, otočíme pouze jeho konektor na příslušných výstupech M. Tomuto říkáme - kalibrace motorů k H-bridge.

Nyní je třeba, podobně jako při řízení motorů pomocí desky spínačů, naučit žáky spolehlivě ovládat motory nejprve pomocí tlačítek, než začneme jízdu programovat. Propojení desky tlačítek je na následujícím obrázku.



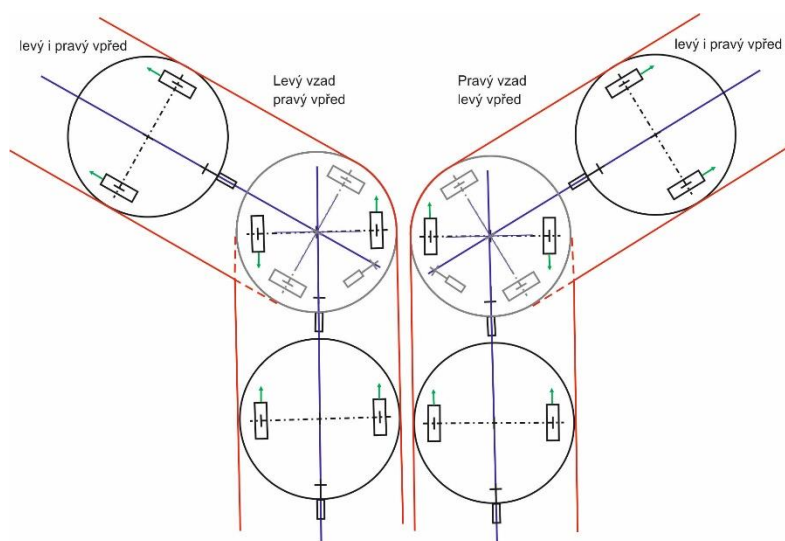
**Obrázek 2.12 – propojení desky H-bridge k desce s tlačítky pro nácvik jízdy diferenciálního podvozku s deskou H-bridge**

Prodlužovací kabel je skvělou jednoduchou pomůckou, která umožní pohodlně řídit robota na větší vzdálenost pomocí tlačítek, nebo propojit dva systémy od sebe vzdálené. Lze si jej pohodlně vyrobit ze dvou samořezných konektorů pro ploché kabely PFL10, nebo samořezných patič LPV-10 a plochého vícežilového kabelu AWG28-10H

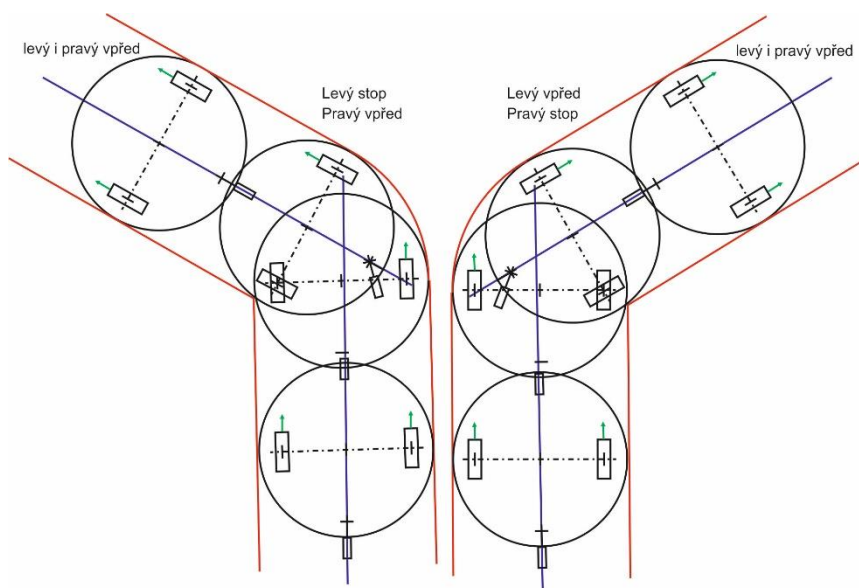


**Obrázek 2.13 – prodlužovací kabel**

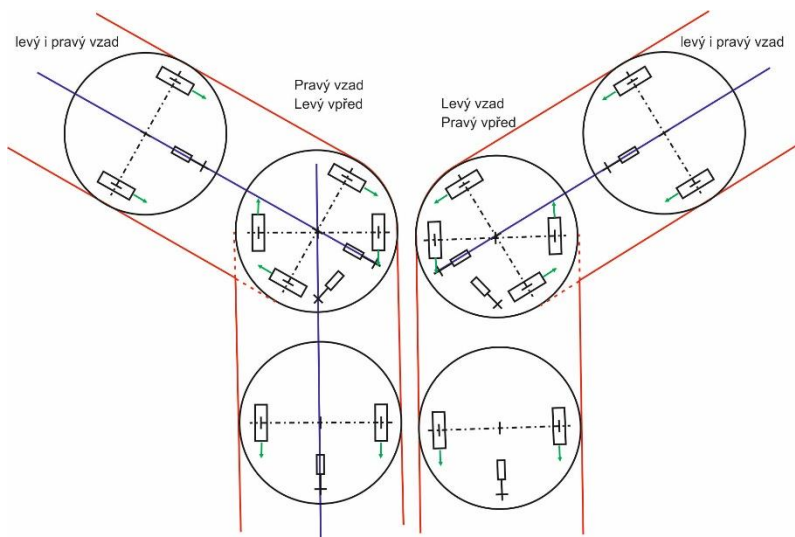
Pokud nepoužíváme řízení rychlosti otáčení kol, H-bridge umožňuje následující způsoby jízdy podvozku.



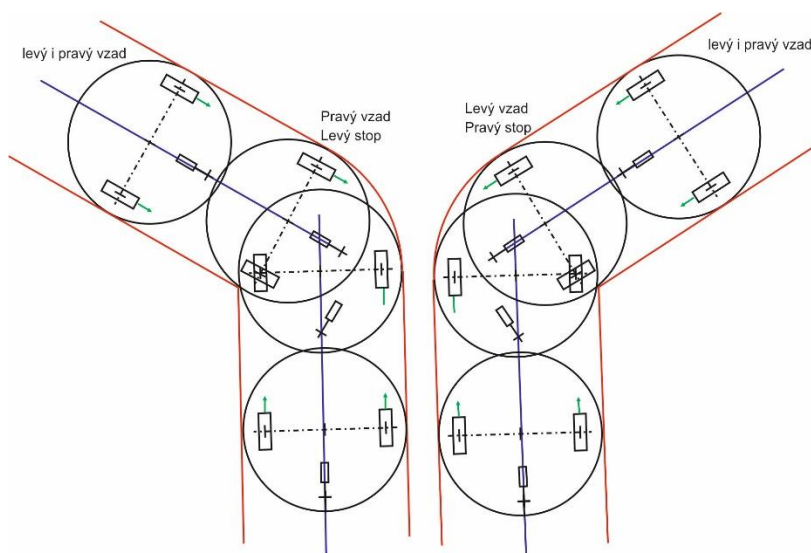
**Obrázek 2.14 – zatáčení vlevo/vpravo vpřed protichodem**



**Obrázek 2.15 – zatáčení vlevo/vpravo vpřed po kole**



**Obrázek 2.16 – zatáčení vlevo/vpravo vzad protichodem**



**Obrázek 2.17 – zatáčení vlevo/vpravo vzad po kole**

Nyní je opět na místě připravit pro žáky několik různě složitých drah a naučit je ovládat diferenciální podvozek nejprve pomocí tlačítek tak, aby si uvědomovali, jaké signály a na jak dlouho je vystavují.

Žáci pracují ve dvojicích. Do tabulky si zapisují časy a simulují reálnou jízdu po zadané trati. Později vypočítávají doby jízdy a doby zatáčení z parametrů svého vlastního podvozku, kteří zjistí podobně jako v případě řízení podvozku pomocí desky tranzistorových spínačů.

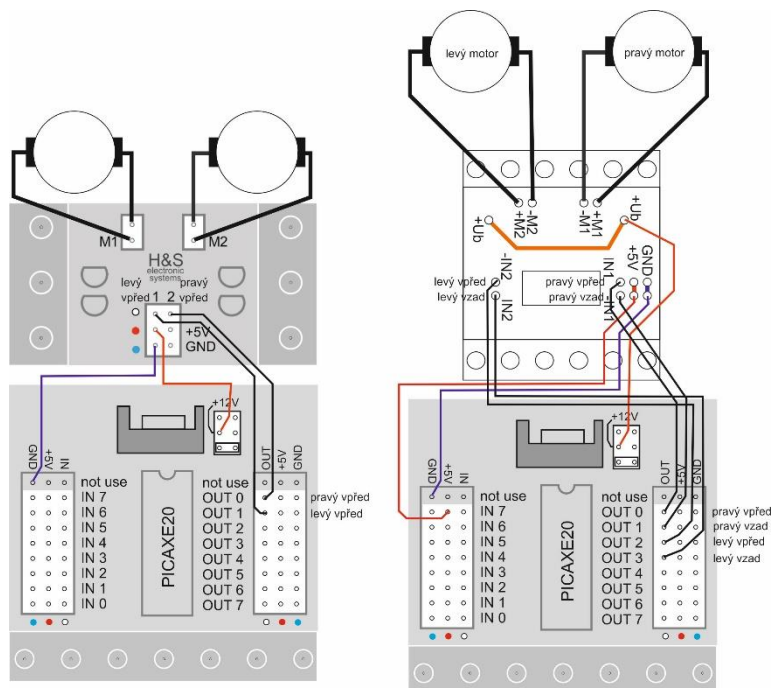
Každý žák tedy bude mít svou tabulku pro svůj podvozek:

řízení podvozku přes desku spínačů			
<b>1 mm vpřed ujede za xx.yyyy ms</b>		<b>o 1 stupeň vpravo zatočí za xx.yyyy ms</b>	
		<b>o 1 stupeň vlevo zatočí za xx.yyyy ms</b>	
řízení podvozku pomocí H-bridgů			
<b>1 mm vpřed ujede za xx.yyyy ms</b>		<b>1 mm vzad ujede za xx.yyyy ms</b>	
<b>o 1 stupeň vpravo vpřed protichodem zatočí za xx.yyyy ms</b>		<b>o 1 stupeň vpravo vpřed po kole zatočí za xx.yyyy ms</b>	
<b>o 1 stupeň vlevo vpřed protichodem zatočí za xx.yyyy ms</b>		<b>o 1 stupeň vlevo vpřed po kole zatočí za xx.yyyy ms</b>	
<b>o 1 stupeň vpravo vzad protichodem zatočí za xx.yyyy ms</b>		<b>o 1 stupeň vpravo vzad po kole zatočí za xx.yyyy ms</b>	
<b>o 1 stupeň vlevo vzad protichodem zatočí za xx.yyyy ms</b>		<b>o 1 stupeň vlevo vzad po kole zatočí za xx.yyyy ms</b>	

Tabulka jízdy po zadané dráze (simulace na živo) příprava pro programování jízdy:

Popis jízdy	úhel (počet stupňů) /vzdálenost (mm)	čas jízdy zokrouhlu na celé ms (ms)	řídící signály H-bridge			
			levý motor vpřed	levý motor vzad	pravý motor vpřed	pravý motor vzad
2.1 m vpřed	2100	$2.485 * 2100 = 5218.5$	1	0	1	0
90 stupňů vpravo protichodem	90	$90 * 1.1125 = 100,125$	0	1	1	0

## 2.3. Způsob programování jízdy diferenciálního podvozku



**Obrázek 2.18 – Příklad připojení desky procesoru k H-bridge a desce spínačů pro řízení diferenciálního podvozku**

Programovat lze podvozek až po propojení H-bridge s výstupními signály desky procesoru. Vždy ale musíme napřed připravit celou požadovanou dráhu do tabulky a také její řízení reálně simulovat, abychom předešli pozdějším problémům s jízdou.

Tento způsob programování jízdy není moc přesný. Otáčky motorů jsou závislé na napětí. Čím vyšší napětí tím vyšší otáčky. Při tomto způsobu (čas = ujetá vzdálenost) se mění ujetá vzdálenost s nabitím baterii (aktuálním napětím) robota.

Po naprogramování provedeme korekce časů v programu, kde se projeví nepřesnosti, kterých jsme se dopustili při výpočtech a analýze požadované dráhy.

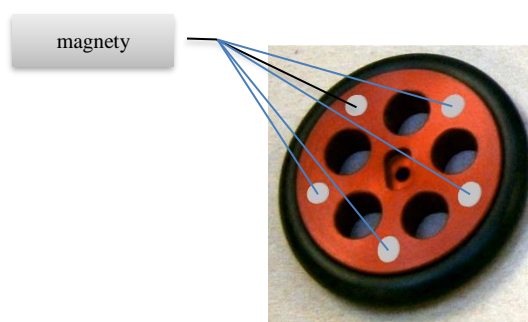
## 2.4. Programování jízdy diferenciálního podvozku kdy se ujetá vzdálenost kola odvíjí od jeho otáček.

Pro přesnější programování dráhy je nutné snímat otáčky kol a v závislosti na jejich otáčkách určovat ujetou dráhu kola. Tento způsob v nejjednodušší verzi využívají bicykly (jízdní kola). Na kole je připevněn magnet a na vidlici kola snímač, který sepne vždy, když kolem něj projíždí magnet kola. Tento způsob snímání otáček kola umožňuje určit ujetou dráhu s přesností na jednu otáčku kola. Počet otáček kola za čas určuje rychlost jízdy kola.



**Obrázek 2.19 – Příklad instalace snímače otáček kola Apache**  
(<https://ekolo.cz/Felektrosada-se-stredovym-motorem-ts250>)

Na kolo robota můžeme nalepit i několik malých neodymových magnetů a tím zvýšit přesnost měření otáček robota. Použijeme například magnety o průměru 5 mm a tloušťce 1 mm.



**Obrázek 2.20 – Neodymové magnety na vnitřní straně kola diferenciálního podvozku**

Jako snímač se používá spínací kontakt jazýčkového relé.



**Obrázek 2.21 – Spínací kontakt jazýčkového relé**

Spínací kontakt jazýčkového relé zapojíme podle následujícího schématu a celý jej umístíme do smršťovací bužírky.



**Obrázek 2.22 – Schéma zapojení snímače otáček**

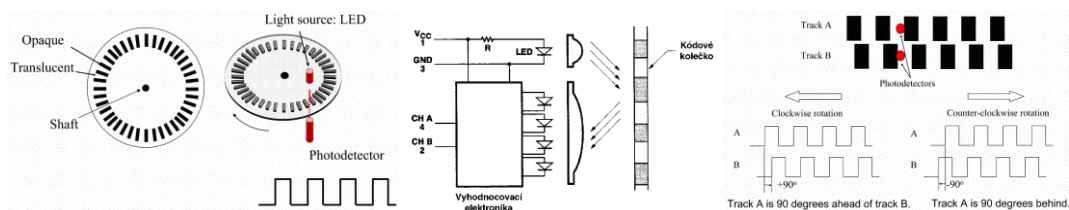
Kontakt pak umístíme nad magnety na kole tak, aby každý magnet jazýček sepnul při průchodu kolem magnetu. Tím při každé otáčce kola sepne kontakt pětkrát, tedy přesnost určení ujeté vzdálenosti je  $1/5$  obvodu kola, tedy  $(\pi * \text{průměr kola} / 5)$ .

Pro vzdálenost menší, než je  $1/5$  obvodu kola je třeba ujetou dráhu počítat jako v minulém případě, tedy z času jízdy.

## 2.5. Další možnosti snímání dráhy ujeté robotem

V zásadě se již moc nových způsobů nevyskytuje. Musíme určit co nejpřesněji ujetou dráhu každého kola diferenciálního podvozku, nebo celého robota. Nejjednodušší je zpřesnit snímání otáček kol robota. Toho lze dosáhnout například přesnějším snímáním směru a počtu otáček vlastního kola. Směs snímat nemusíme, kolo je poháněno motorem, tak směr určuje směr otáčení motoru. Pokud jde o zvětšení počtu pulzů na otočku kola, pak lze kolo opatřit například přesnějším optickým snímačem otáčky kola. Takové řešení můžeme vidět například zde: <https://automatizace.hw.cz/rotacni-enkodery-pro-rizeni-pohonu-a-pohybu.html>.





**Obrázek 2.23 – Optické snímače otáček**

Další možností je, jste-li venku a je-li silný signál GPS, využít tohoto signálu, popřípadě například využít gyroskopů a elektronických kompasů apod. Všechny uvedené možnosti dalece přesahují rozsah této učebnice, a proto se jimi podrobně zabývat nebudeme.

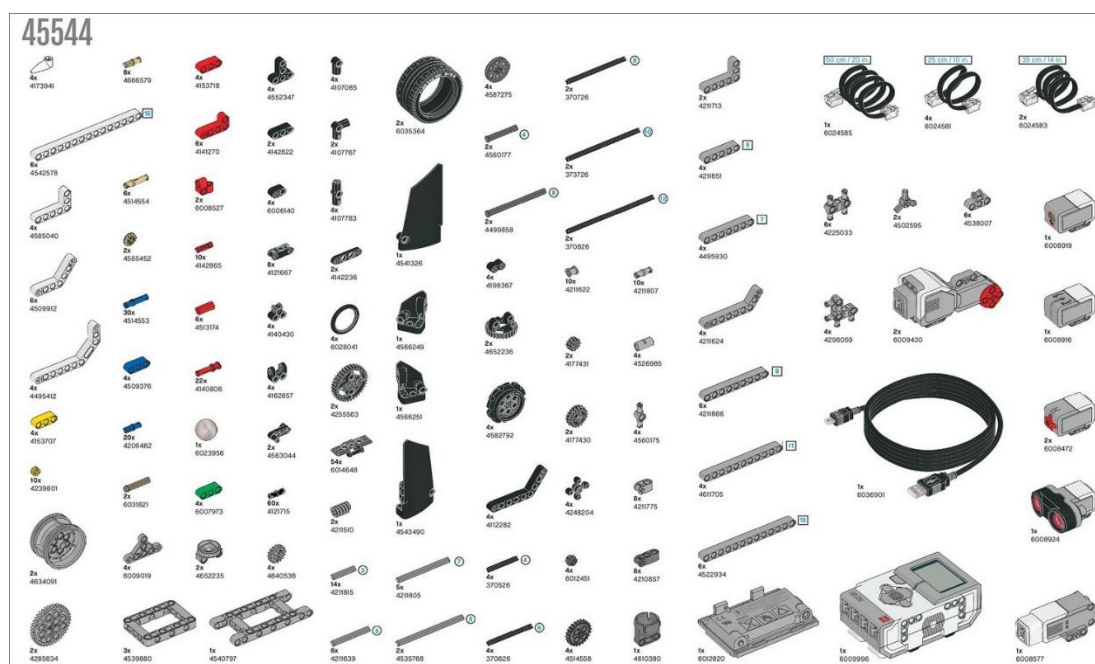
## 2.6. Závěr

Robotické systémy je možné na 2. stupni ZŠ zapojovat do výchovně-vzdělávacího procesu mnoha způsoby. Žáci a učitelé mohou využívat robotických systémů ve výuce nebo v zájmových kroužcích. Mohou se účastnit nejrůznějších soutěží robotiky, které rozvíjejí prostorovou a konstrukční představivost žáků.



### 3. Práce s robotickou stavebnicí LEGO® MINDSTORMS™, Education EV3

Stavebnice LEGO® MINDSTORMS™ bývá často označovaná za oblíbenou programovatelnou stavební sadu, umožňující žákům a studentům rozvíjet informatické myšlení cestou řešení problémů z reálného světa v oblasti vědy, techniky a matematiky. V současné době se jedná již o třetí generaci LEGO® MINDSTORMS™, přičemž zkratka "EV" znamená evoluci, odtud tedy EV3. Na trh byla uvedena v srpnu 2013. Sada obsahuje 1x řídicí jednotku EV3, 2x velký a 1x malý servomotor, 2x dotykový senzor, 1x barevný/optický senzor, 1x ultrazvukový senzor, 1x gyroskopický senzor a celkem 541 dílů LEGO TECHNIC (<https://www.robotworld.cz/lego-mindstorms-education-ev3>).

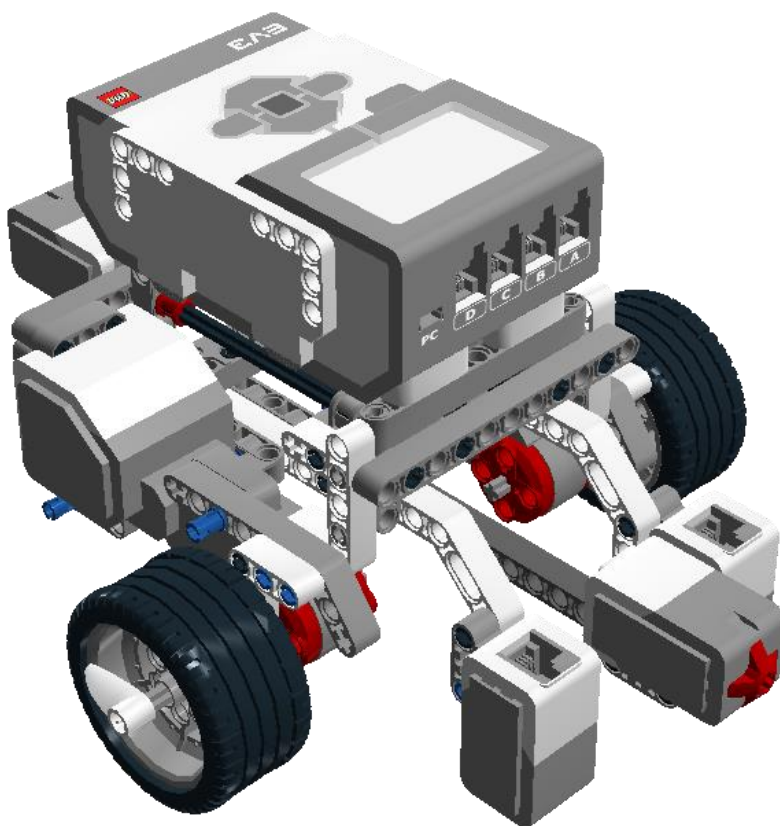


Obrázek 3.1- Přehled sady dílů stavebnice LEGO® MINDSTORMS™

Začínajícím uživatelům nabízí výrobce na svých stránkách online průvodce. Ten jim pomůže se získáním a instalací bezplatného softwaru EV3 Programmer, sestavením vybraného robota dle konstrukčního návodu, jeho spárováním s PC či tabletem. Nechybí ani video-tutoriály seznamující s orientací v programovacím prostředí a základy programování (<https://www.lego.com/en-my/themes/mindstorms/videos>).

### 3.1. Stavba robota vlastní konstrukce

Jak již bylo uvedeno výše - součástí online průvodce stavebnice jsou také návody na stavbu robotů, kteří mohou chodit, mluvit, manipulovat s předměty a dělat mnoho dalších věcí. Tento studijní text nabízí vlastní konstrukční řešení z důvodu možného srovnání a ověření si postupů popsaných pro robotickou stavebnici HSES. LEGO robotické jezdítko využívá stejně jako stavebnice HSES diferenciální typ podvozku s obdobným rozvorem kol. Obsahuje barevné senzory, jejichž vzdálenost od osy kol lze měnit, dva pevně umístěné dotekové senzory, dva velké motory a řídicí kostku EV3.

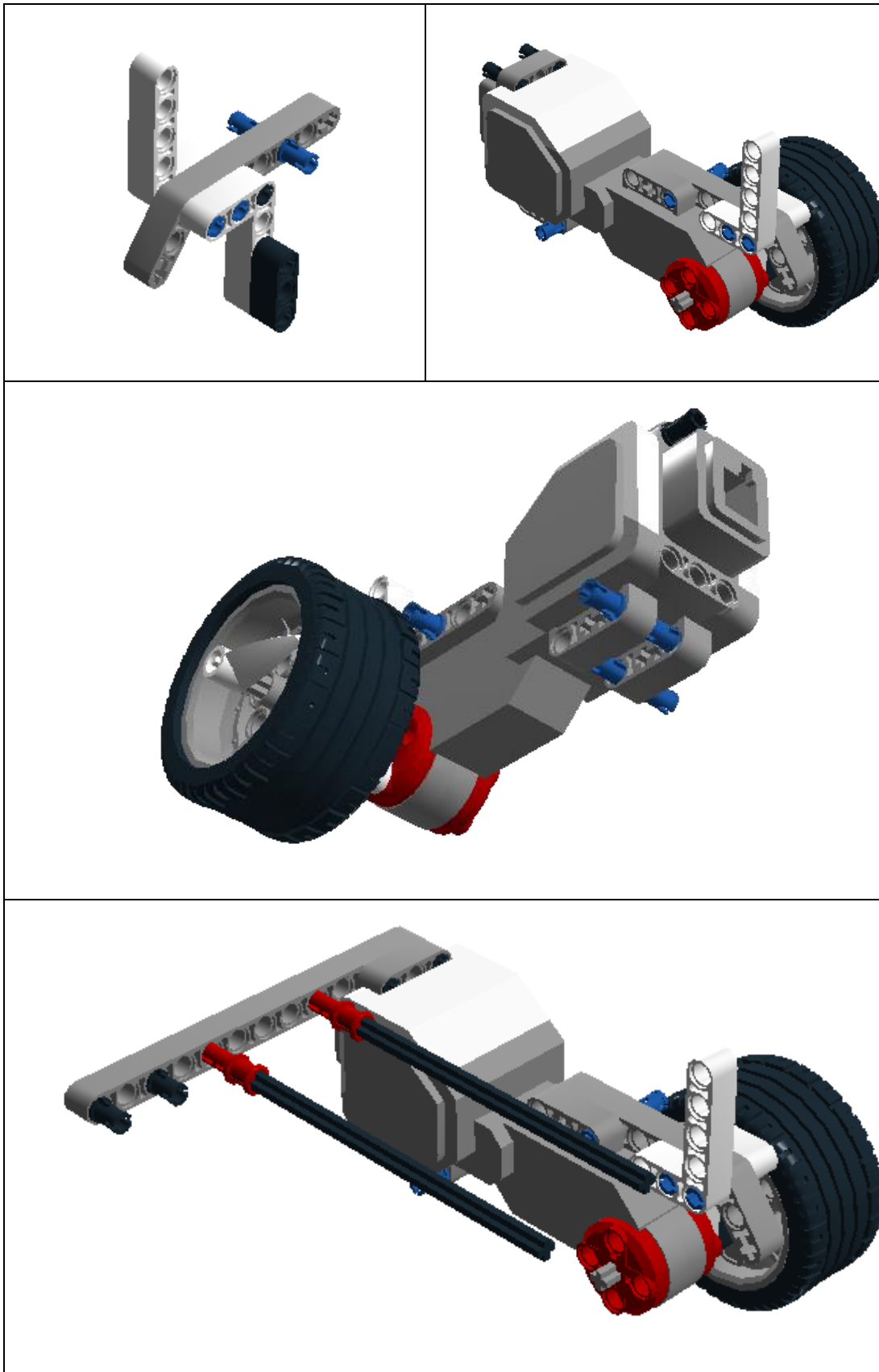


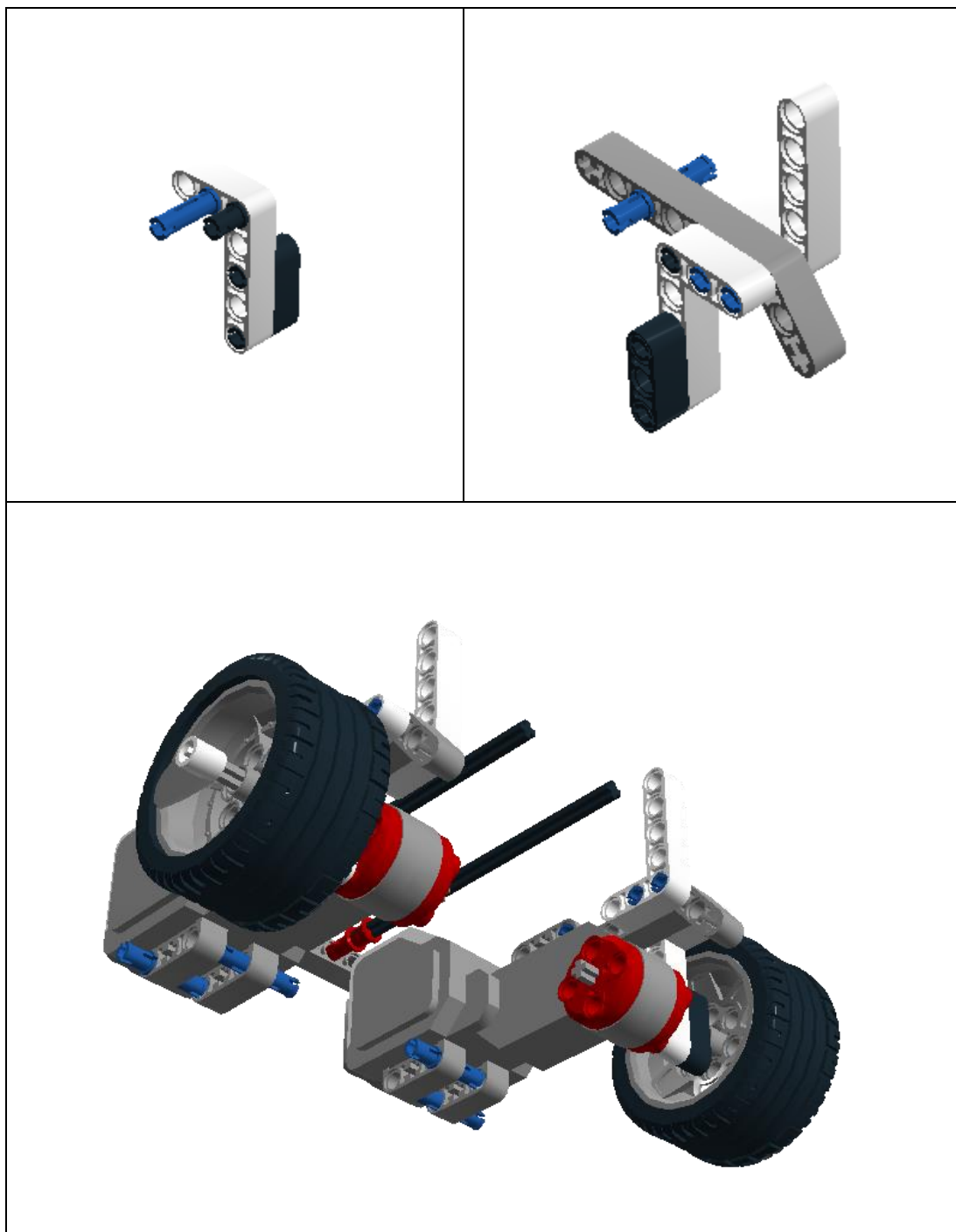
*Obrázek 3.2 - Ukázka sestaveného robotického podvozku*

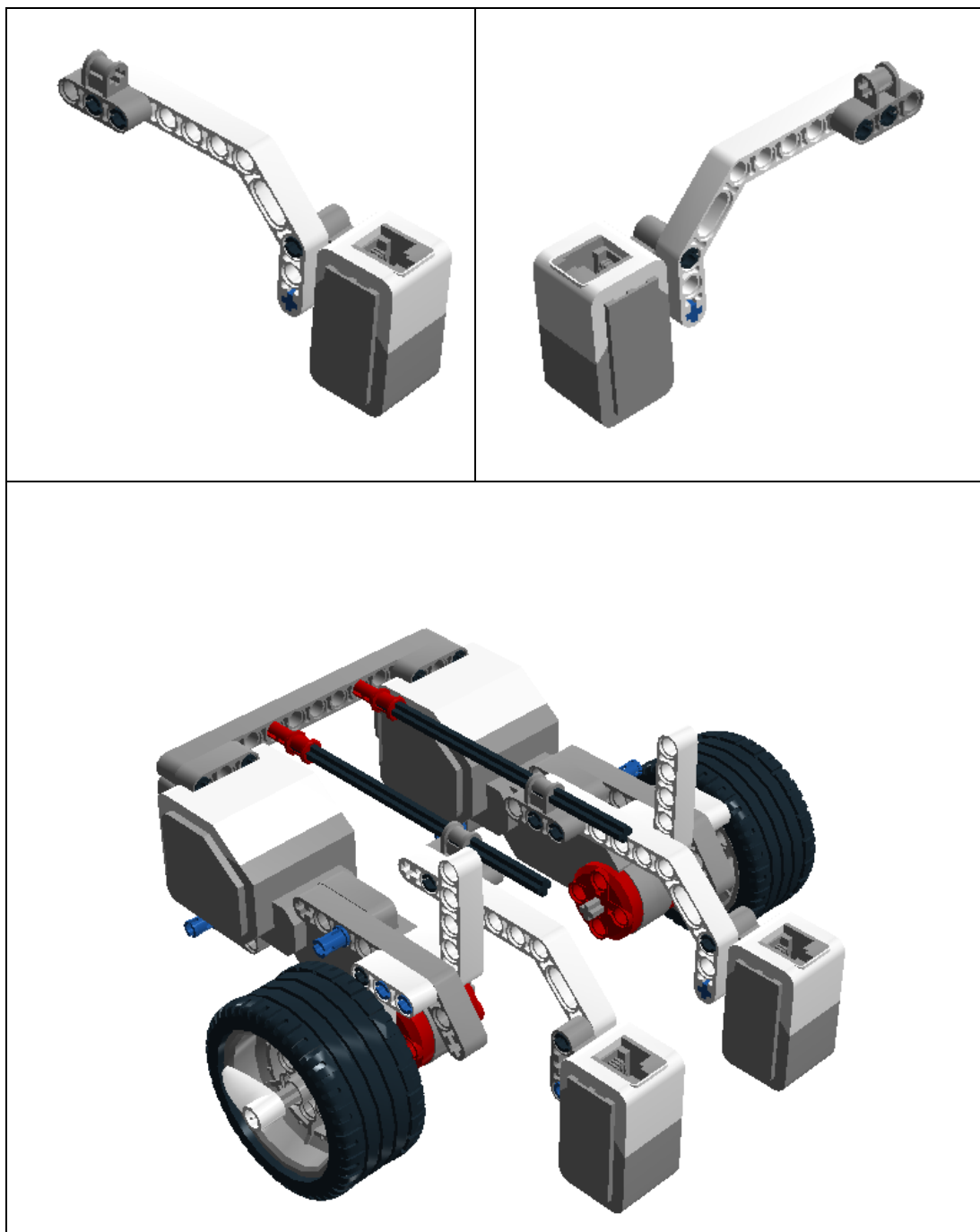
Uvedené robotické jezdítko by měli zvládnout sestavit i začátečníci. K vytvoření konstrukčního návodu byl využit program LEGO Digital Designer – ocení jej zejména učitelé, kteří chtějí ve výuce využívat robota vlastní konstrukce a potřebují pro žáky vytvořit potřebné návody na stavbu (<https://www.lego.com/en-us/ldd>).

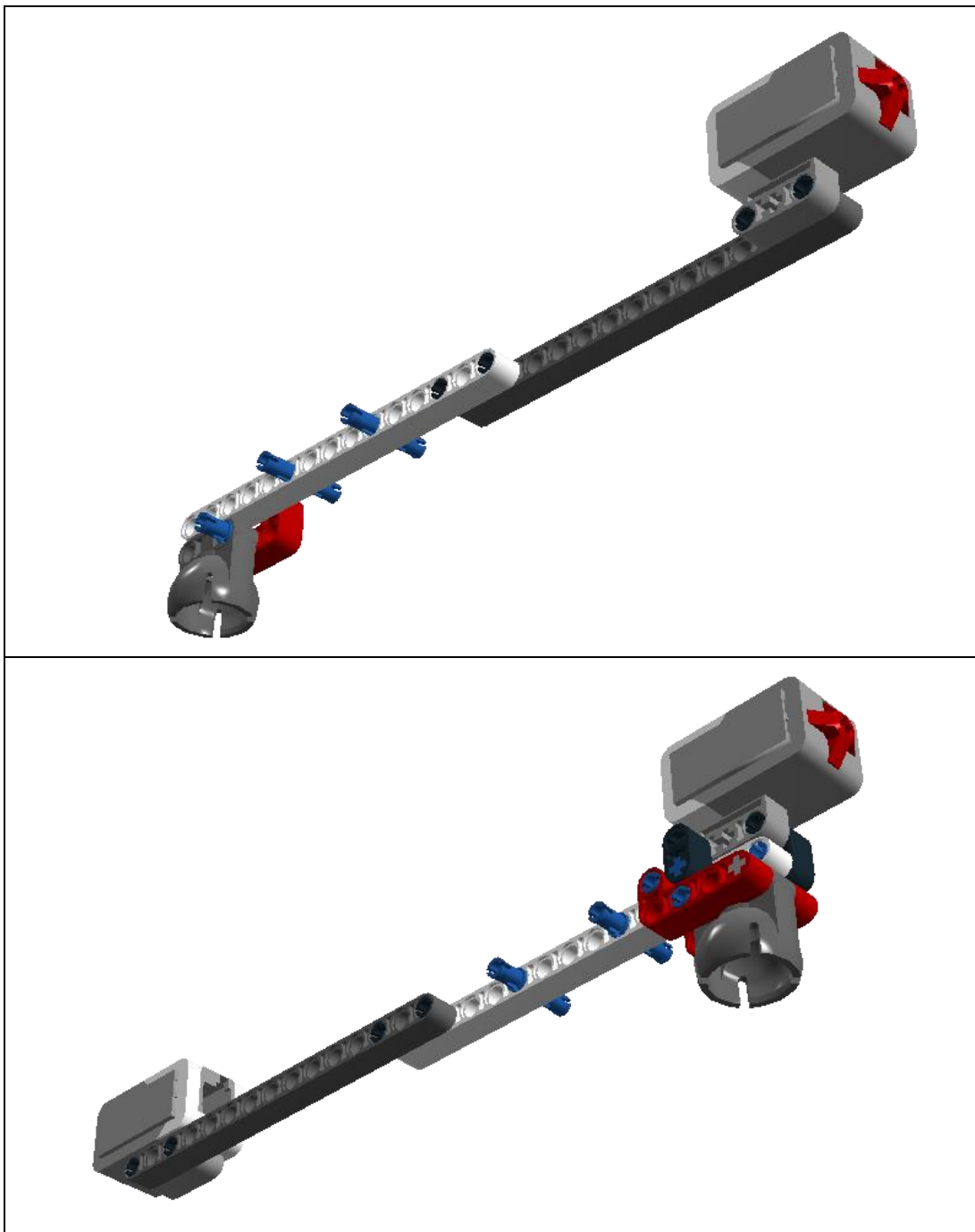
#### **Postup stavby robotického jezdítka v několika krocích:**

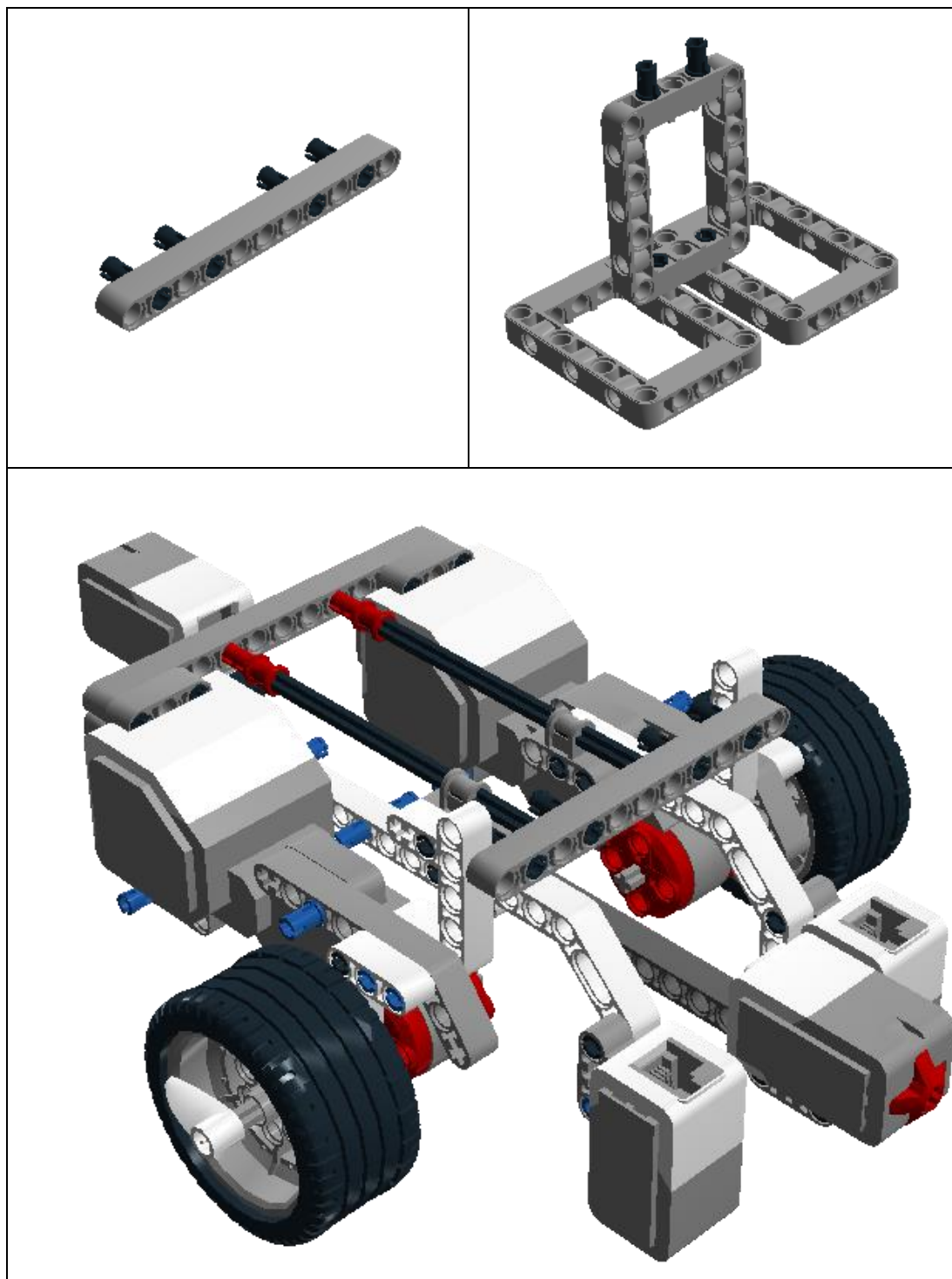
Viz obrázek 3.3 a kroky 1 až 11 (hotové jezdítko).



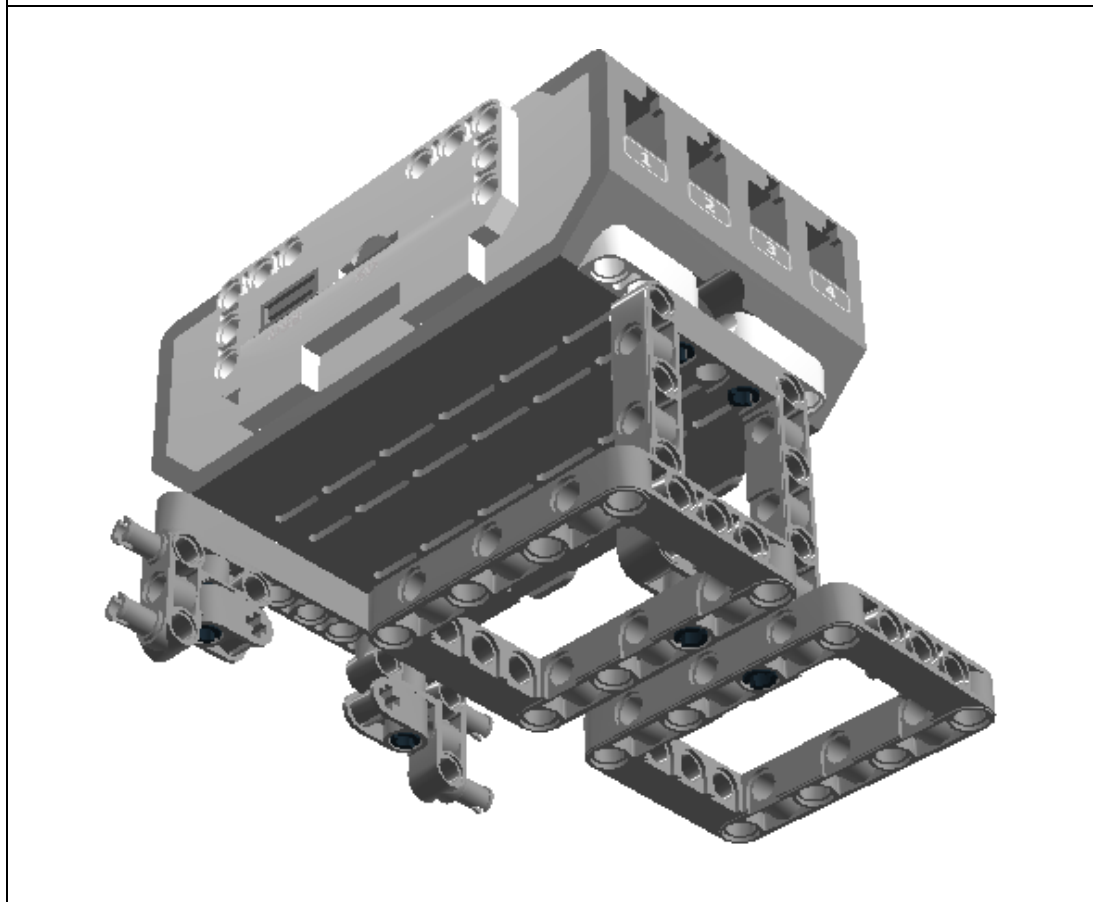
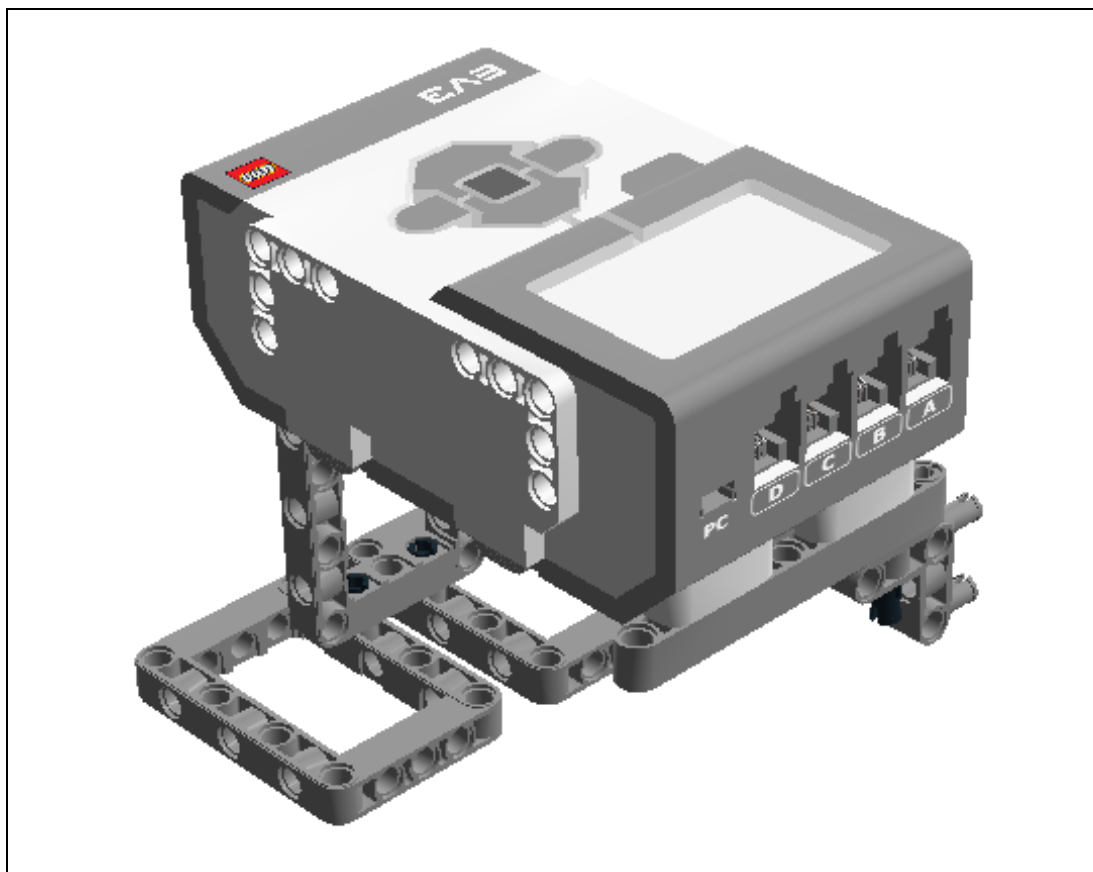


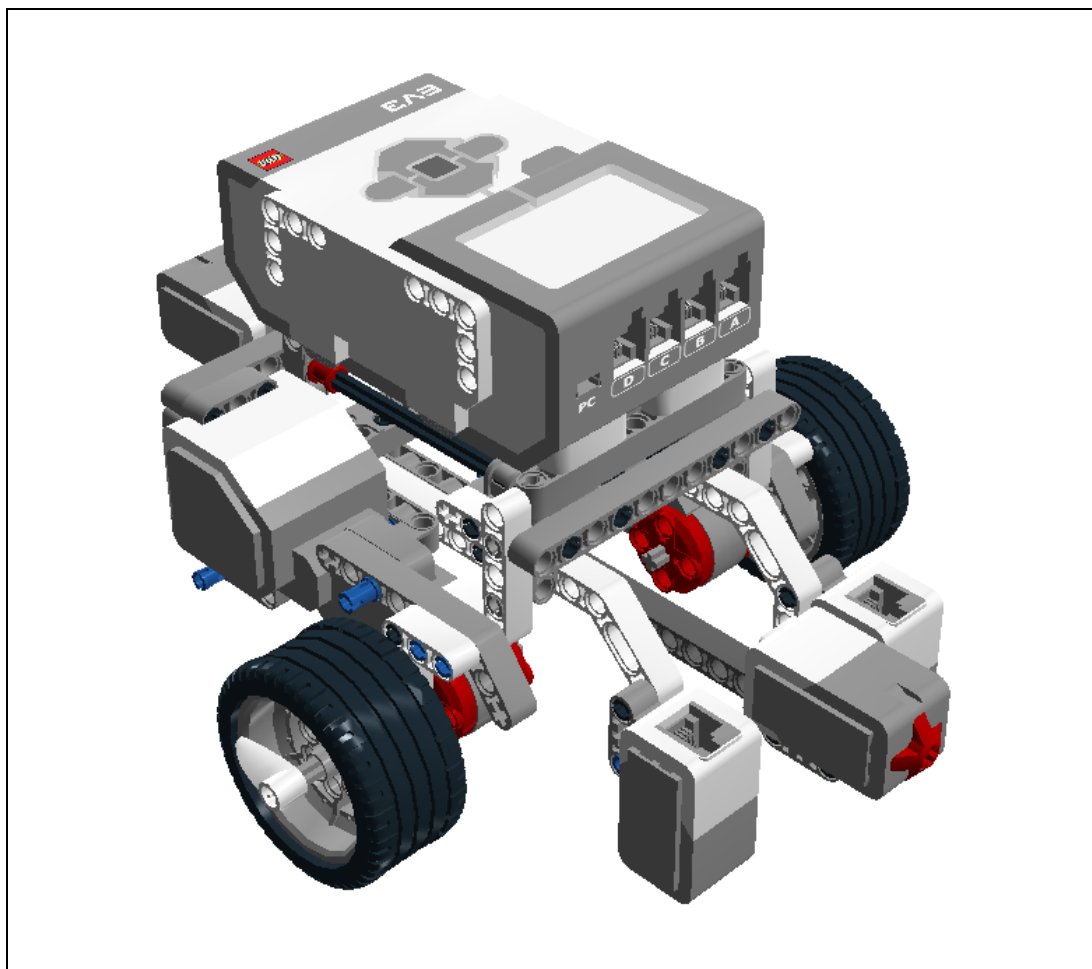












*Obrázek 3.3 - Ukázka sestaveného robotického jezdítka a postup konstrukce v krocích*

## 3.2. Propojení řídicí jednotky s periferiemi

### **Kostka EV3:**

Mozkem celého robota je tzv. kostka EV3. Disponuje čtyřmi vstupními porty pro připojení senzorů (označeno číslicemi 1 až 4), čtyřmi výstupními porty (A až D) pro připojení motorů, LED indikací aktuálního stavu kostky/programu, reproduktorem, slotem pro SD kartu (rozšíření paměti kostky) a mini-USB konektorem pro připojení k PC a jedním hostitelským USB portem k připojení externího zařízení jako např. wifi adaptéru. Podrobné technické parametry naleznete v uživatelském manuálu (dostupný na stránkách výrobce).



*Obrázek 3.4 – Ukázka řídicí jednotky – kostka EV3*

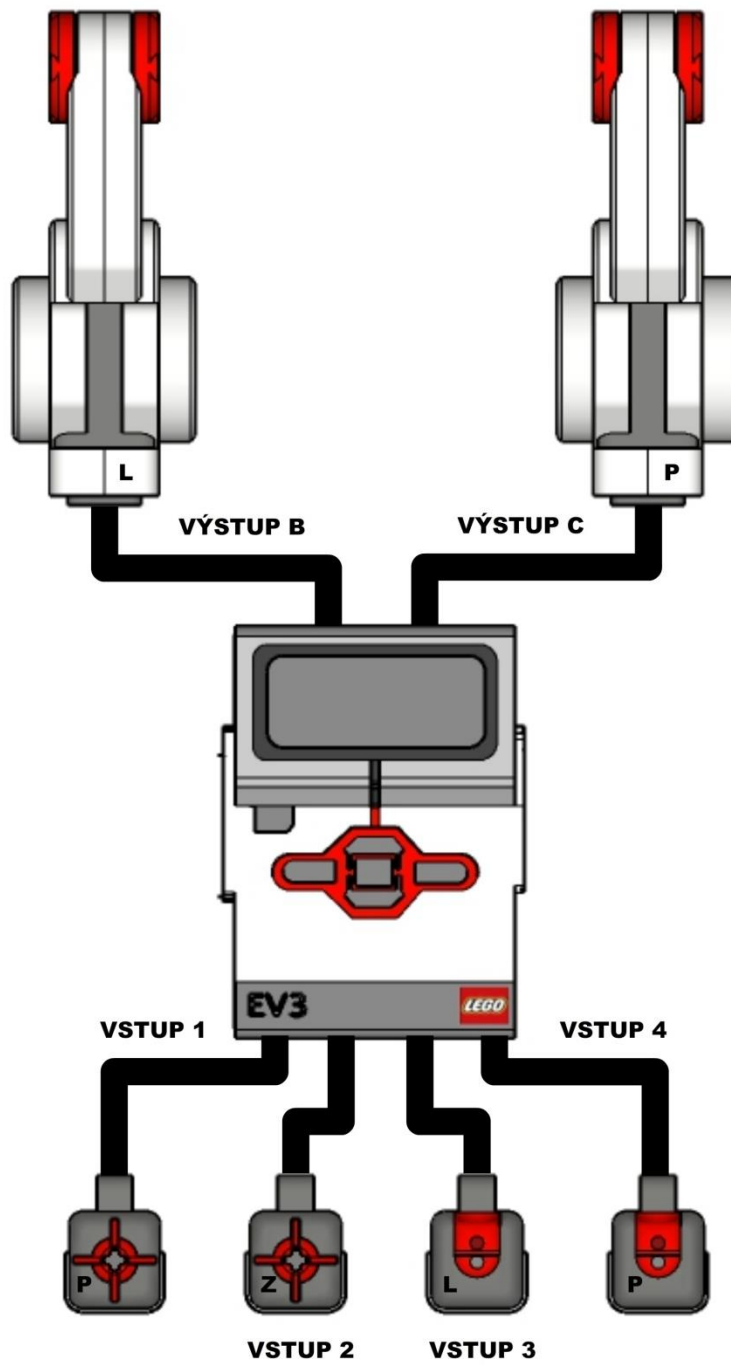
Pokud při vytváření programů není kostka EV3 připojena k počítači, software automaticky přiřadí senzory a motory k výchozím portům. Přiřazení k výchozím portům vypadá takto:

- Port 1: dotykový senzor
- Port 2: Žádný senzor
- Port 3: barevný senzor
- Port 4: infračervený senzor
- Port A: střední motor
- Port B: velký motor
- Port C: velký motor
- Port D: velký motor

Pokud je při programování kostka EV3 Brick připojena k počítači, software automaticky identifikuje, k jakému portu jsou senzory nebo motory připojeny.

Před zahájením programování robotického jezdítka je nutné propojit všechny senzory a motory s řídicí kostkou EV3. Zapojení reflektuje výchozí přiřazení portů k motorům a sensorům.

Na výstup B je připojený levý motor (kukáte-li sešora na kostku EV3 ve směru jízdy robota). Na vstupu 1 je připojený přední dotekový senzor. Na vstup 3 je připojen levý barevný senzor.

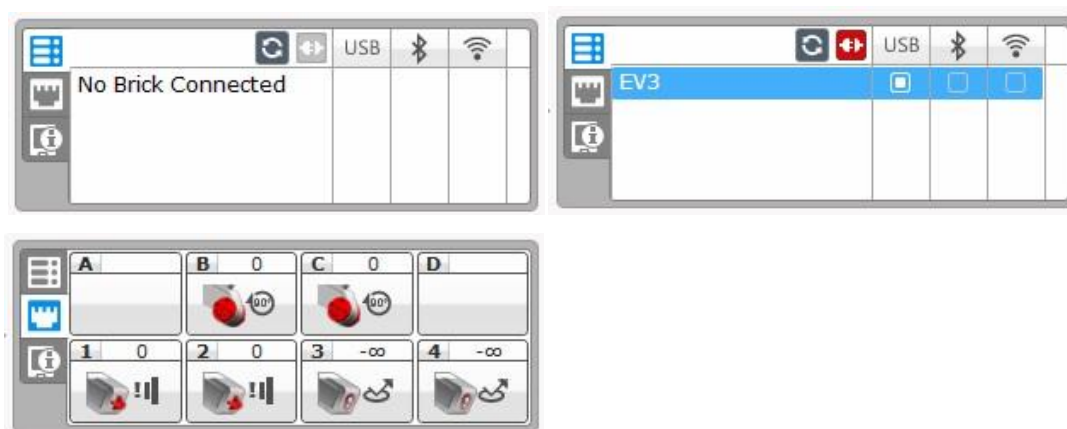


*Obrázek 3.5 - Blokové schéma znázorňující propojení kostky EV3 s periferiemi*

Kostka EV3 umožňuje více způsobů propojení počítačem či tabletem. Nejjednodušší způsob je připojení pomocí USB kabelu. Můžete však využít některou z podporovaných bezdrátových technologií (Bluetooth nebo WIFI).

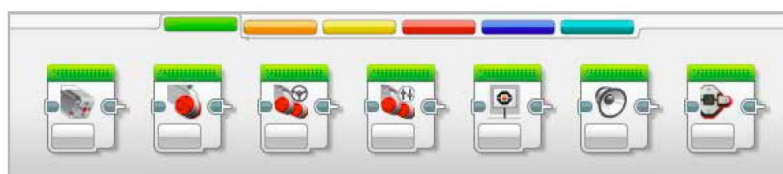
### 3.3. Začínáme programovat

Po spuštění programovacího softwaru lze ověřit spojení na stránce hardwaru. Zde se také uskutečňuje řízení kostky EV3. Pod záložkou „Port view“ je možné zkontrolovat správné přiřazení periférií k jednotlivým portům.



**Obrázek 3.6 - Panel hardwaru – EV3 nepřipojeno; jednotka EV3 připojena; port view – kontrola zapojení**

Programovací prostředí je založeno na spojování programovacích bloků. K dispozici je hned několik typů (rozlišeno barvou). První skupinu tvoří tzv. Bloky akcí, které umožňují ovládat motory, řídit pohyb, ovládat stavovou kontrolku kostky, zvuky a display.



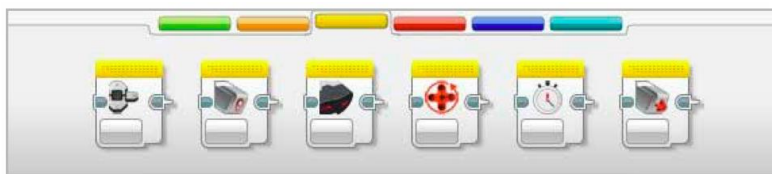
**Obrázek 3.7 - Skupina programovacích bloků Action**

Oranžovou barvou je rozlišena další skupina programovacích bloků, tzv. bloky toků. Zde se nachází bloky čekání, cyklů, podmínek a přerušení.



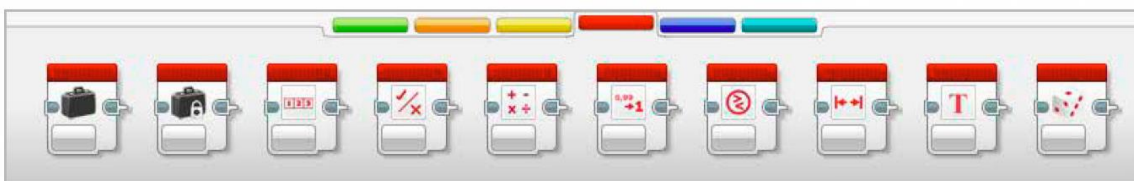
**Obrázek 3.8 - Skupina programovacích bloků Flow Control**

Bloky senzorů umožňují zjišťovat informace od vstupních periférií jako například tlačítek na kostce EV3, dotekového senzoru, barevného senzoru, infračerveného senzoru apod.



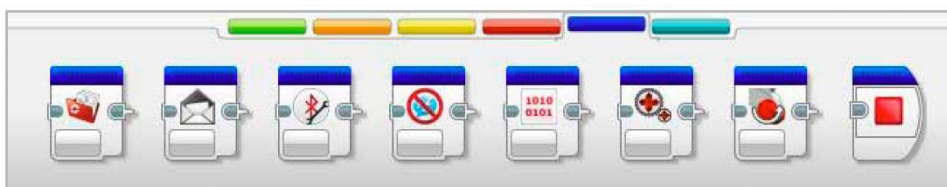
**Obrázek 3.9 - Skupina programovacích bloků Sensor**

Pomocí bloků dat můžete definovat proměnné, konstanty, pole operací, logické operace, matematické výpočty, porovnávat, generovat náhodné hodnoty atp.



**Obrázek 3.10 - Skupina programovacích bloků Data Operations**

Bloky pokročilých funkcí umožňují například přístup k souborům, zpracování zpráv, správu Bluetooth připojení, zastavování programu apod.



**Obrázek 3.11 - Skupina programovacích bloků Advanced**

Poslední blok, nazvaný *Moje bloky*, slouží k vytváření vlastních podprogramů. Toto může být užitečné, například když často používáte stejný segment programu opakovaně na více místech kódu.

### 3.4. Pohyb robota v prostoru

Pro pohyb robota v prostoru bývají nejčastěji využívány tzv. velké motory. Díky svým parametrům jsou k těmto účelům opravdu nejvhodnější. Mají integrovaný senzor otáček s rozlišením 1°, točivý moment 20Ncm a 160 až 170 otáček za minutu.



**Obrázek 3.12 - Large Motor**

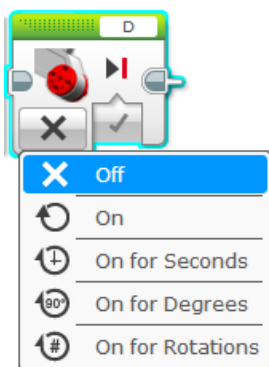
Pro ovládání velkých motorů lze využít hned několik programových bloků z palety **Action**. První z nich je blok **Large Motor**. Tímto blokem lze ovládat pouze jeden motor – volba konkrétního motoru se provádí určením výstupního portu kostky EV3, ke kterému je motor připojen (zvýrazněno červeně).



**Obrázek 3.13 - Programovací blok Large Motor – volba výstupního portu**

Blok velkého motoru nabízí pět režimů ovládání.

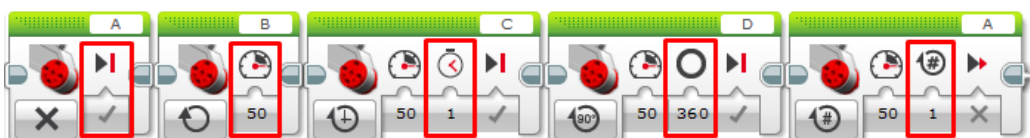
- **Off** = Vypnutí chodu motoru,
- **On** = Neomezený běh motoru po dobu běhu programu,
- **On for Seconds** = Běh motoru po určitou dobu,
- **On for Degrees** = Běh motoru na základě míry natočení udané v úhlových stupních od 0 do 360°,
- **On for Rotations** = Běh motoru podle definovaného počtu otáček.



Obrázek 3.14 - Programovací blok Large Motor – volba režimu

Od zvoleného režimu se odvíjí také množství dostupných vstupních portů:

- **Break at End** - způsob zastavení motoru. Okamžité zastavení (Break) reprezentuje logická hodnota True a pozvolné dojetí (Coast) hodnota False.
- **Power** - rychlost otáčení motoru udávaná v procentech na stupnici od 100 (vpřed) do -100 (vzad).
- **Seconds** - doba otáčení motoru udávaná ve vteřinách.
- **Degrees** - míra natočení udávaná v úhlových stupních od 0 do 360°.
- **Rotations** - počet prováděných otáček motoru.



Obrázek 3.15 - Programovací blok Large Motor – příklady vstupních portů

Jak již bylo uvedeno výše, blokem **Large motor** lze ovládat pouze jeden motor. Pro pohyb dvou motorů souběžně je určen programovací blok **Move Steering**. Počet režimů je shodný s Large motorem, jen u výběru výstupního portu se volí



kombinace portů a nově přibývá jeden vstupní port – **Steering** - řízení směru natáčení robota.



*Obrázek 3.16 - Ukázka programovacího bloku Move Steering*

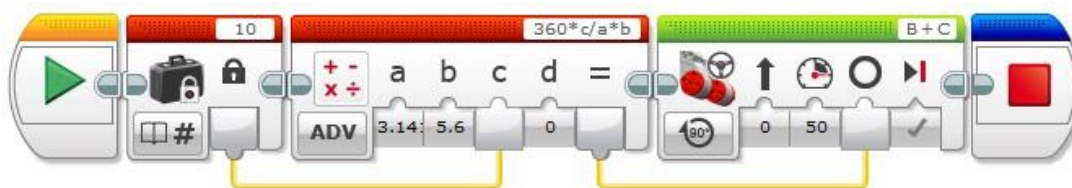
Na základě dosavadních informací by nemělo být složité sestavit program, realizující jízdu robota rovně po dobu 10 vteřin rychlostí 20%. Zajímavým pokusem by mohlo být změření ujeté vzdálenosti například s plně nabitými bateriemi a poté s téměř vybitými články. Obdobným způsobem by bylo možné zkoumat změnu ujeté vzdálenosti v závislosti na změně celkové hmotnosti robota/zátěži na motory.

Při využití programovacího bloku **Move Steering** v režimu **Seconds** není využita zpětná vazba, kterou motor u jiných režimů poskytuje kostce EV3. Díky tomu se bude ujetá vzdálenost měnit v závislosti na velikosti napětí nebo měnícím se zatížení motoru.

Je zřejmé, že pokud bychom zadání úkolu modifikovali a požadovali, aby robot při jízdě rovně vpřed ujel přesně 10cm, bude předchozí způsob ovládání nevyhovující. V tomto případě by bylo vhodné použít režim Degrees. Tento režim využívá zpětné vazby z motoru, tzn. pootočení kola, bude nezávislé na velikosti napětí či zatížení motoru. Při výpočtu, o kolik stupňů se mají motory otočit, aby robot urazil požadovanou vzdálenost, je třeba zohlednit průměr kol (napsáno na pneumatice – robotické jezdítko používá průměr 5,6).

$$S = \frac{360 * l}{\pi * p} [^\circ]$$

Kde S = úhel natočení, l = ujetá vzdálenost, p = průměr kol



*Obrázek 3.17 - Ukázka možného řešení zpracovaného v Programmeru*

Na začátku programu je pomocí programovacího bloku **Constant** definovaná požadovaná ujetá vzdálenost (na obrázku nastaveno 10 = 10cm). Následuje blok **Math**. V něm se provede výpočet dle výše uvedeného vzorce. Proměnná *c* je dosazena z bloku **Constant**. Výsledek výpočtu je předáván vstupnímu portu **Degrees**.

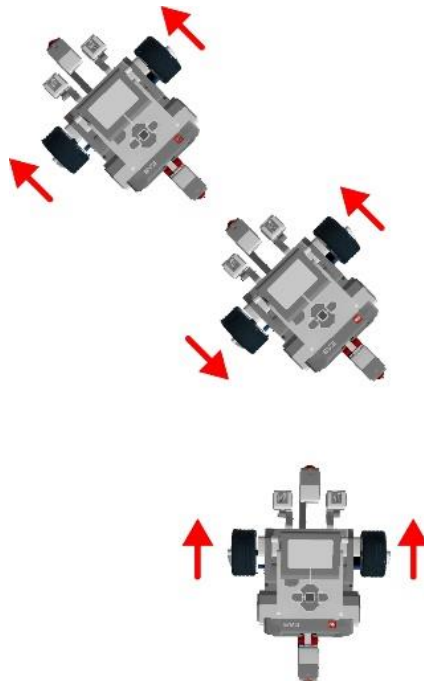
Pokud není použito řízení rychlosti otáčení kol ani řízení směru pomocí vstupního portu **Steering**, umožňuje diferenciální podvozek následující způsoby změny směru jízdy podvozku:

### Zatáčení vlevo/vpravo vpřed protichodem:



Obrázek 3.18 - Ukázka zpracování v Programmeru

### Levý motor vzad; pravý motor vpřed:



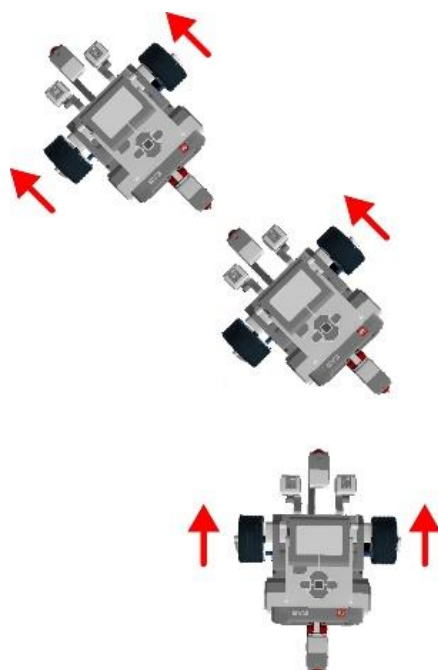
Obrázek 3.19 - Znárodnění pohybu robotického jezdítka

**Zatáčení vlevo/vpravo vpřed po kole:**



*Obrázek 3.20 - Ukázka zpracování v Programmeru*

**Levý motor stop; pravý motor vpřed:**



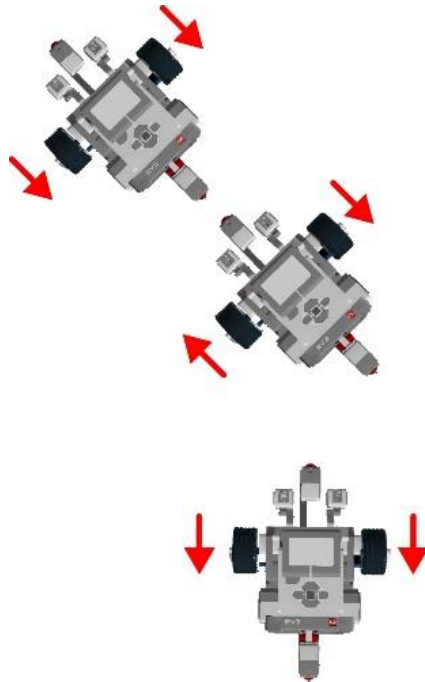
*Obrázek 3.21 - Znázornění pohybu robotického jezdítka*

**Zatáčení vlevo/vpravo vzad protichodem:**



*Obrázek 3.22 - Ukázka zpracování v Programmeru*

**Pravý motor vzad; levý motor vpřed:**



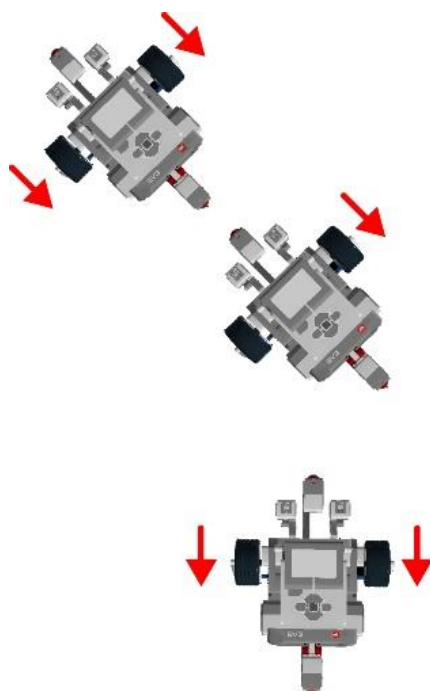
*Obrázek 3.23 - Znáznornění pohybu robotického jezdítka*

**Zatáčení vlevo/vpravo vzad po kole:**



*Obrázek 3.24 - Ukázka zpracování v Programmeru*

**Pravý motor vzad; levý motor stop:**



*Obrázek 3.25 - Znáznornění pohybu robotického jezdítka*

**Úkol:**

Naprogramujte robota tak, aby přešel od školy do pizzerie. Váš robot se umí pohybovat pouze rovně vpřed o 15cm, rovně vzad o 15cm, vlevo – otočí se na místě o 90 °, vpravo otočí se na místě o 90 °. Robot smí jezdit pouze po vyznačených silnicích.

Zkuste Váš návrh optimalizovat tak, aby robot při cestě do pizzerie vykonal co nejméně kroků (příkazů vpřed, vzad, vlevo, vpravo).





Obrázek 3.26 - Podklad k realizaci úkolu

### 3.5. Využití vstupních a výstupních modulů

Vstupní periferie robota jako například dotekové senzory, sonary, gyroskopy apod. bývají často označovány za smyslové orgány robota, s jejíž pomocí vnímá okolní svět.

#### **Dotekový senzor:**

Jedná se o analogový senzor, který umí detekovat tři události: stisk, uvolnění a stisknutí s následným uvolněním.

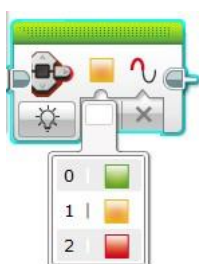


Obrázek 3.27 -Dotekový senzor

Robotické jezdítko obsahuje dva dotekové senzory – jeden vepředu a druhý v zadní části. Lze je využít například k detekci překážky/nárazu. Programovací blok dotekového senzoru podporuje dva základní režimy – zjištění stavu tlačítka nebo testování stavu tlačítka.

Pokud je v režimu zjištění stavu tlačítka sensor stisknutý, výstupní port vrací hodnotu True. V opačném případě je návratová hodnota rovna False.

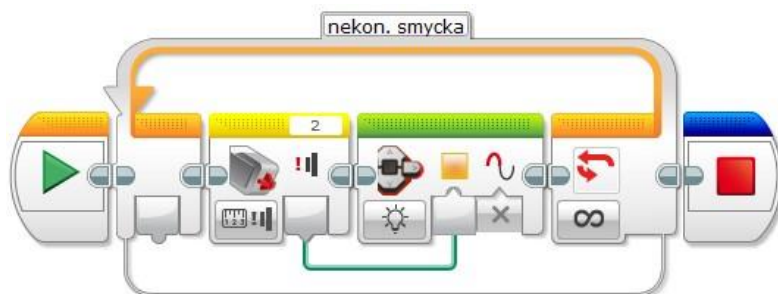
Pro názorné ověření lze využít například programovací blok **Brick Status Light**, který mění podsvětlení kostky EV3. Zelenému podbarvení odpovídá hodnota 0, oranžovému hodnota 1 a červenému hodnota 2 viz obrázek níže.



*Obrázek 3.28 - Programovací blok Brick Status Light*

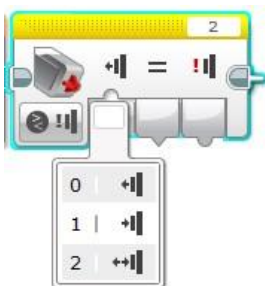
Následující příklad demonstuje změnu stavu tlačítka pomocí změny barvy stavové LED indikace na kostce EV3. V nekonečně se opakující smyčce se testuje stav tlačítka na portu 2. Pokud není tlačítko dotekového senzoru stisknuté, výstupní port vrací hodnotu False, která odpovídá číselné hodnotě 0. Tento výstupní port je přiveden na vstupní port **Colour** programovacího bloku **Brick Status Light**.

Jakou barvu podsvětlení bude mít kostka EV3? Jak se barva změní, stiskneme-li tlačítko?



*Obrázek 3.29 - Příklad ověření stavu tlačítka*

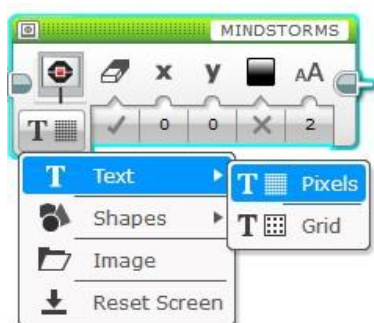
Druhý režim - testování stavu tlačítka umožňuje vyhodnotit tři stavy, které mohou na tlačítku nastat - uvolněno = návratová hodnota 0, stisknuto = návratová hodnota 1 a stisknuto a uvolněno = návratová hodnota 2. Testovaný stav lze nastavit na vstupním portu **State**. Tento je dále porovnáván s aktuálním stavem tlačítka (výstupní port **Measured Value**). Je-li nastavený stav na vstupním portu State shodný s aktuální hodnotou výstupního portu **Measured Value**, pak výstupní port **Compare Result** vrací návratovou hodnotu **True**.



Obrázek 3.30 - Programovací blok Touch Sensor – volba stavu

### Tlačítka kostky EV3

Programovací blok **Brick Buttons** (tlačítek kostky EV3) stejně jako programovací blok dotykového senzoru podporuje dva základní režimy - zjištění stavu tlačítka nebo testování stavu tlačítka. Každému tlačítku je přiřazeno jedinečné identifikační číslo. Mapování tlačítek lze zjistit v nabídce vstupního portu **Set of Brick Buttons IDs** v režimu testování stavu tlačítek. Druhou možností je praktické ověření například za pomoci programovacího bloku **Display**. Display Kostky EV3 se běžně využívá pro výpis nejrůznějších hlášek, výpis proměnných v průběhu lazení programu apod. Programovací blok disponuje čtyřmi režimy. Pro účely ověření id tlačítek je vhodný režim **Text - Pixels** umožňující vypsání libovolného textu na konkrétní pozici displeje.





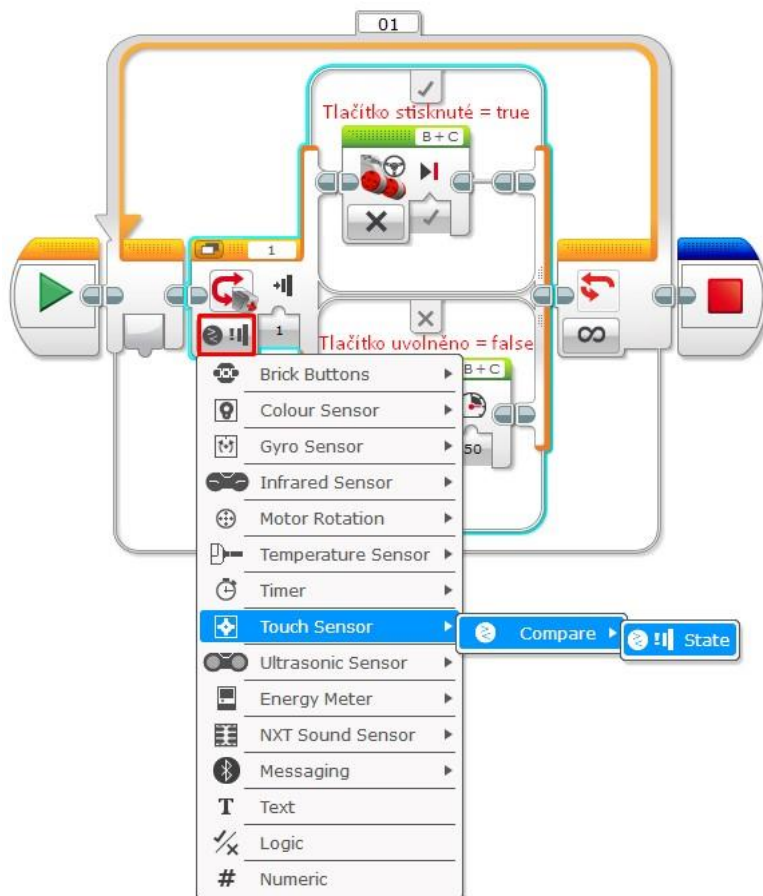
**Obrázek 3.31 - Programovací blok Display - volba režimu Text**

Program opět běží v nekonečné smyčce, v níž opakovaně testuje stav tlačítek kostky EV3. Výstupní port **Button ID** programovacího bloku **Brick Buttons** vrací id stisknutého tlačítka. Tento výstupní port je propojen se vstupním portem Text, programovacího bloku **Display**. ID stisknutého tlačítka se tedy zobrazí na displeji.

**Obrázek 3.32 - Mapování Button ID programovacího bloku Brick Buttons**

### Podmíněné příkazy:

Doposud uvedené příklady se senzory měly za úkol pouze demonstrovat činnost konkrétního senzoru. Je zřejmé, že pouhé získání informace o aktuálním stavu senzoru je nedostatečné, pokud tuto informaci robot adekvátně nevyhodnotí a nezareaguje na ni odpovídajícím způsobem. Pokud bude dotekový senzor sloužit k detekci překážky, tak po jeho stlačení se bude muset robot rozhodnout, co bude dělat dále, zdali například couvne a překážku zkusí objet, nebo bude čekat tak dlouho, dokud překážka nezmizí apod. Pro rozhodování je určen programovací blok **Swith** ze záložky **Flow Control**. Tento blok před spuštěním určité části programu nejprve otestuje, zda došlo ke splnění podmínky pro jeho spuštění.



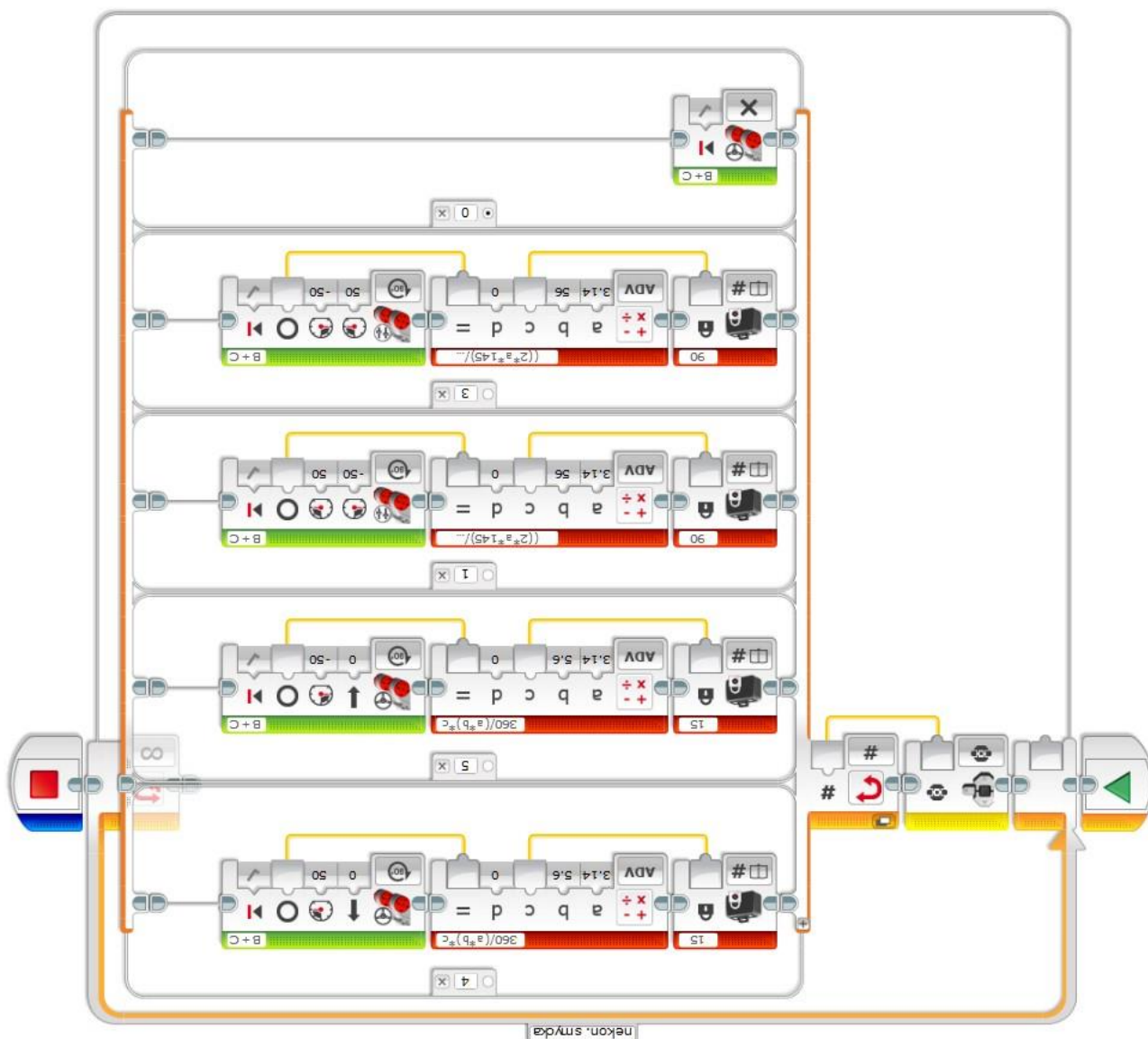
Obrázek 3.33 - Ukázka vyhodnocení překážky z dotekového senzoru

V tomto případě programovací blok testuje stav dotekového senzoru. Pokud vrátí hodnotu **true**, odpovídající stisknutému tlačítku, oba motory jsou zastaveny a robot čeká na odstranění překážky. Pokud dotekový senzor vrátí hodnotu **false**, robot jede vpřed.

#### Úkol:

**Pokuste se upravit program robota jedoucího ze školy do pizzerie tak, aby bylo možné cestu zadávat pomocí tlačítek na kostce EV3. Například dojde-li ke stisku tlačítka vpřed, robot se rozjede rovně, urazí dráhu dlouhou 15cm, poté zastaví a bude čekat na další instrukci.**





Obrázek 3.34 - Řešení úkolu

**Barevný senzor:**

Jedná se o digitální senzor se vzorkovací frekvencí 1kHz, který dokáže detekovat barvu nebo intenzitu světla prostupujícího malým okénkem na čelní straně senzoru. Tento senzor lze používat ve třech různých režimech: barevný režim, režim intenzity odraženého světla a režim intenzity okolního světla.

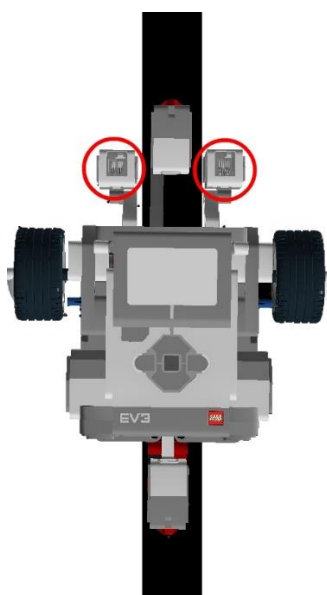
V barevném režimu senzor rozpoznává sedm barev – černou, modrou, zelenou, žlutou, červenou, bílou, hnědou. Díky schopnosti rozlišovat barvy lze robota naprogramovat podobně jako robota Ozobot, který umí jezdit po čáře a vykonávat instrukce na základě přečtení kombinace barevných značek nakreslených na papíře.

V režimu intenzity okolního světla senzor měří sílu světla prostupujícího do okénka z okolního prostředí, například ze světelného kuželu svítilny. Senzor opět používá škálu od 0 (velmi tmavá) do 100 (velmi světlá). Můžete například naprogramovat robota tak, aby zastavil akci, když zhasnou světla.



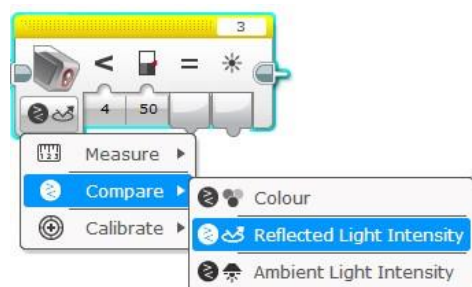
*Obrázek 3.35 - Barevný senzor*

V režimu intenzity odraženého světla senzor vysílá červené světlo. Zároveň měří jeho intenzitu, která dopadá zpět na senzor po odrazu od překážky. Senzor používá škálu od 0 (velmi tmavá) do 100 (velmi světlá). Můžete například naprogramovat robota tak, aby sledoval černou čáru. K tomuto účelu lze využít jeden barevný senzor, dva a více. Dále v textu bude uvedena pouze varianta se dvěma senzory z důvodu propojení s jiným předmětem, kde žáci sestavují řídicí logiku pro dvě reflexní čidla z diskretních elektronických součástí tzn. princip činnosti i způsob řízení kol jim je dobře známý.



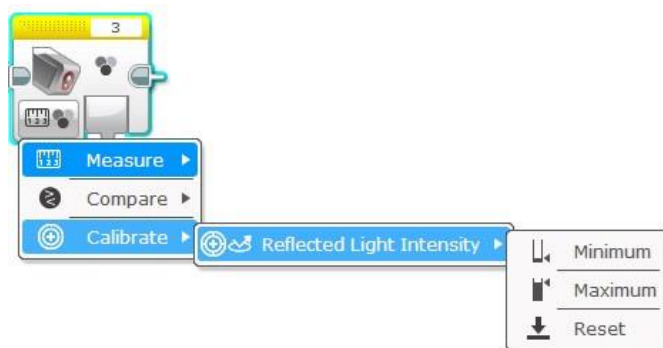
*Obrázek 3.36 - Ukázka umístění barevných senzorů vůči černé čáře*

Pro jízdu po čáře je vhodný režim **Compare - Reflected Light Intensity** programovacího bloku Barevného senzoru. Tento režim slouží k porovnání zjištěné intenzity odraženého světla se zadanou prahovou hodnotou.



**Obrázek 3.37 - Ukázka programovacího bloku Colour Sensor – volba režimu**

U reflexních čidel je důležité provést jejich kalibraci. Nejprve senzorem změříme intenzitu odraženého světla od černé čáry a tuto hodnotu nastavíme jako minimum. Obdobný postup uplatníme u kalibrace maxima, jen senzor bude umístěn nad bílou plochou.



**Obrázek 3.38 - Ukázka programovacího bloku Colour Sensor – kalibrace**

Kalibrace senzorů se většinou provádí pouze jednou, při spuštění systému nebo při změně okolního prostředí. Pro snadnou konfiguraci lze využít programovací blok **Wait** v režimu **Brick Buttons**. Systém bude čekat na stisk zvoleného tlačítka řídicí jednotky.



### 3.6. Závěr

Práce s robotickou stavebnicí LEGO® MINDSTORMS™, Education EV3 může být pro žáky na 2. stupni velmi zajímavá. Toto známé a rozšířené robotické řešení je populární především z hlediska dostupnosti LEGO stavebnic. S některou z četných LEGO stavebnic se žáci mohli potkat v nižších ročnících ZŠ, případně mohli mít nějakou stavebnici i doma.

Práci s robotickými stavebnicemi je vhodné směřovat již do ZŠ. Jejich využití pomáhá rozvíjet oblast designu a konstruování, digitální kompetence a poutavou formou seznámit žáky s principy konstruování a algoritmického myšlení. Tyto kompetence mohou podpořit následnou orientaci žáků na volbu povolání a studium technických SŠ a učebních oborů.





## 4. Zdroje

Elektrosada se středovým motorem [Online]. [cit. 2019-04-17]. Dostupné z: <https://ekolo.cz>.

Friedmann, Z., a kol. *Trendy a aspekty ve výuce techniky a informatiky pro potřeby mateřských a základních škol*. 1. vyd. Brno: Masarykova univerzita, 2018. ISBN 978-80-210-9014-9.

Hrbáček, J. *Robotická sada pro vývoj a výuku* [Online]. [cit. 2019-03-14]. Dostupné z: <http://hses.cz/roboticka-sada-pro-vyvoj-a-vyuku/>.

Inspiration videos LEGO [Online]. [cit. 2019-05-10]. Dostupné z: <https://www.lego.com/en-my/themes/mindstorms/videos>.

Digital designer LEGO [Online]. [cit. 2019-05-10]. Dostupné z: <https://www.lego.com/en-us/ldd>.

Představení LEGO Mindstorms Education [Online]. [cit. 2019-05-10]. Dostupné z: <https://www.robotworld.cz/lego-mindstorms-education-ev3>.

Vojáček, A. *Rotační enkodéry pro řízení pohonů a pohybu* [Online]. [cit. 2019-05-27]. Dostupné z: <https://automatizace.hw.cz/rotacni-ekodery-pro-řízení-pohonu-a-pohybu.html>.