

FEI STU

Manuál

K predmetu mobilné robotické systémy

Martin Dekan
9. 2. 2017

Obsah

1. Úlohy na cvičeniach.	2
Súťaž (0-5b)	3
Úloha 1. Lokalizácia a polohovanie robota v prostredí (10b).....	3
Úloha 2 Navigácia (10b)	4
Úloha 3 Mapovanie (5b)	5
Blok 4 Plánovanie trajektórie (10b)	5
Bonusové úlohy	6
3. Hardvér.....	7
iRobot Create.....	7
Laserový diaľkomer RPLidar.....	11

1. Úlohy na cvičeniach

Cieľom predmetu je naučiť sa základy riadenia pohybu mobilných robotov najmä z hľadiska senzorového vybavenia a k tomu určených algoritmov lokalizácie, navigácie a plánovania pohybu robota v prostredí. Úlohy budete programovať na Raspberry Pi, ktoré je umiestnené na každom robote dvoma možnými spôsobmi.

1. Priamo na Raspberry Pi s pripojeným monitorom, myšou a klávesnicou.
2. Pripojením sa na wifi AP robota a programovaním cez SSH s Xserverom, napríklad pomocou terminálu smarTTY.

Na Raspberry Pi sa budete musieť v oboch prípadoch prihlásiť.

Meno: pi

Heslo: raspberry

Raspberry sa bootuje do dotykovej obrazovky na robote, ak chcete programovať na pripojenom monitore, treba sa na tento monitor prepnúť klávesovou skratkou Ctrl+Alt+F2 a naštartovať grafické rozhranie príkazom „hdmi“.

Na ploche nájdete folder DEMO, nachádzajú sa v ňom 4 ukážkové programy, ktoré si môžete voľne spúšťať, prezerať a kopírovať, ALE NEMODIFIKOVAŤ. Skopírované programy modifikovať samozrejme môžete.

Popis demo programov:

laser_reader – ukážka práce s laserovým diaľkomerom, pripojenie laserového diaľkomera a na stlačenie gombíka vypíše aktuálne meranie lidarů.

robot_demo – ukážka práce s robotom, pripojenie robota, získanie údajov zo snímačov (callback funkcia) a bezpečnostný signál na blížiacu sa vybitie baterky (nutné grafické rozhranie, ak chcete využiť knižnicu v terminálovom programe, kontaktujte vedúceho cvičení).

quipaint – ukážka práce s prvkami vývojového prostredia QT, kreslenie, timer.

read_txt_map – ukážka načítania txt súboru s kótami mapy.

Zároveň nájdete na ploche folder OFFLINE DEMO, v tomto priečinku nájdete 2 programy (offline_robot_processor, offline_laser_processor), ktoré simulujú reálny robot a môžete si tak niektoré časti algoritmov navrhnuť doma a na reálnom robote ich len overiť/dokončiť. Zároveň sa tam nachádza program, pomocou ktorého boli pre tieto simulačné programy vytvorené dáta.

PRI PRÁCI SI ZÁLOHUJTE PRÁCU, PRI NÁHODNOM VYPNUTÍ RASPBERRY PI JE VEĽKÁ ŠANCA STRATY SÚBOROV!!!!!!

Na cvičeniach budete mať za úlohu vypracovať projekt pozostávajúci z čiastkových úloh rozdelených do štyroch blokov a súťaže. Z každého bloku si môžete vybrať buď úlohu, ktorá je

vysvetlená v tomto manuáli (a bude podrobnejšie prejdená aj na cvičení) alebo niektorú z metód vysvetlených na prednáškach (s bodovým ohodnotením podľa náročnosti).

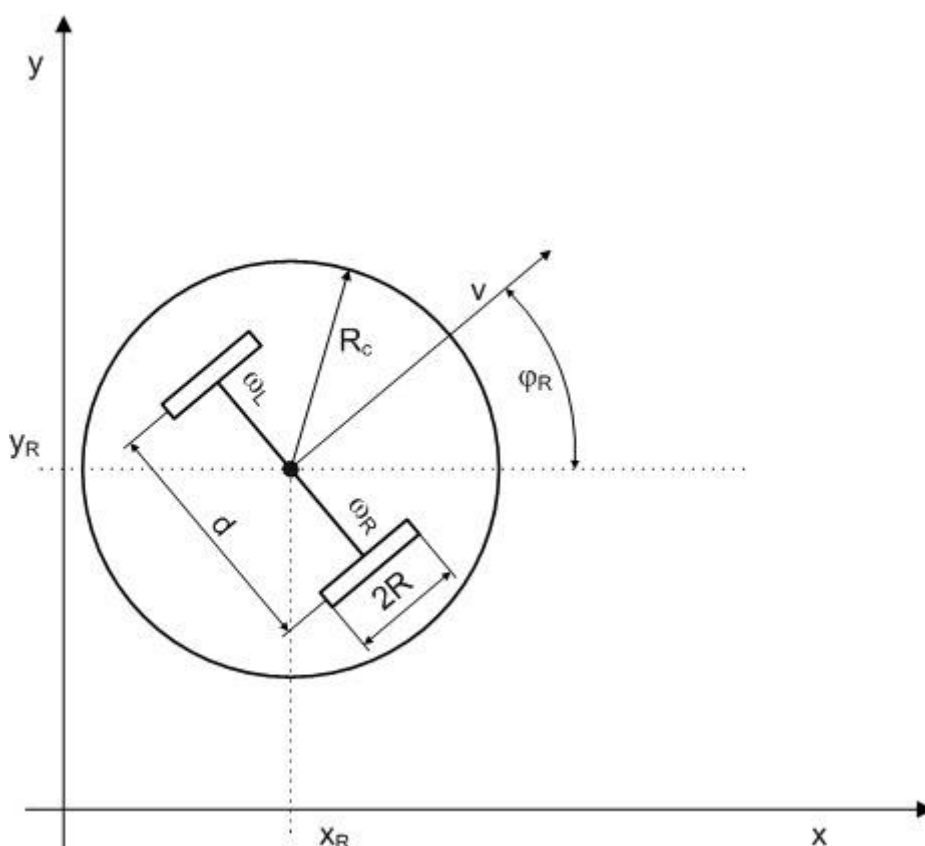
Celkovo je možné na cvičeniach získať 35 bodov za úlohy a maximálne 5 bodov za súťaž.

Súťaž (0-5b)

Súťaž prebehne na konci semestra. Podmienkou účasti je schopnosť presunúť robota z bodu A do bodu B (bez prekážok). Cieľom súťaže je dostať robota z bodu A do bodu B v priestore s prekážkami. Roboty sa budú pohybovať v čiastočne známej mape, t.j. dostanete neúplnú mapu prostredia. Pri súťaži sa vyhodnocuje čas, za ktorý sa dostanete do cieľa a presnosť s akou sa dostanete do bodu B. Najlepší získajú 5b najhorší 3b. Za účasť sú 2 body.

Úloha 1. Lokalizácia a polohovanie robota v prostredí (10b)

Úlohou lokalizácie je povedať, kde v priestore sa mobilný robot nachádza. Najjednoduchší spôsob lokalizácie mobilného kolesového robota s diferenciálnym podvozkom (t.j. taký ako sa používa na cvičeniach) je odometria. Odometria je typ relatívnej lokalizácie (určuje polohu voči predchádzajúcej polohe), ktorá prírastok polohy určuje na základe otočenia kolies.



Obrázok 1 Kinematická schéma diferenciálneho podvozku mobilného robota

Na výpočet odometrie pre robot zobrazený na obrázku 1, je nutné aplikovať tieto vzorce:

$$v_K = (\omega_{L_K} + \omega_{R_K}) \frac{R}{2}$$

$$\varphi_{R_{K+1}} = \varphi_{R_K} + \Delta t (\omega_{R_K} - \omega_{L_K}) \frac{R}{d}$$

$$x_{R_{K+1}} = x_{R_K} + \Delta t v_K \cos \varphi_{R_K}$$

$$y_{R_{K+1}} = y_{R_K} + \Delta t v_K \sin \varphi_{R_K}$$

Úlohou polohovania je dostať mobilný robot na želané súradnice. To je pre robot iRobot Create možné dosiahnuť riadením rýchlosti jednotlivých kolies. Na to, aby ste dosiahli želanú polohu presne, je nutné navrhnuť regulátor (P,PI,PID). Najjednoduchšie riešenie polohovania, je rozdeliť pohyb robota na transláciu a rotáciu (t.j. robot sa buď hýbe dopredu alebo sa točí na mieste) a pre každý z týchto pohybov navrhnuť regulátor samostatne. Pseudo-kód diskrétného regulátora vyzerá takto:

```
previous_error = 0
integral = 0
start:
    error = setpoint - measured_value
    integral = integral + error*dt
    derivative = (error - previous_error)/dt
    output = Kp*error + Ki*integral + Kd*derivative
    previous_error = error
    wait(dt)
    goto start
```

Úloha 2 Navigácia (10b)

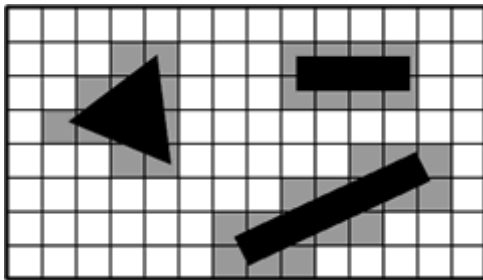
Navigácia mobilného robota zabezpečuje bezkolízny prechod robota prostredím. Na svoju činnosť využíva aktuálne údaje zo snímačov a nepotrebuje poznať prostredie vopred (mapu). K najjednoduchším algoritmom navigácie patria hmyzie algoritmy (bug algorithm), z toho len niektoré na svoje fungovanie využívajú laserový diaľkomer. Takýmto algoritmom je hmyzí algoritmus - typ dotyčnica (tangent bug algorithm).

Princíp algoritmu je:

1. Ak sa dá ísť na cieľ a nie si v móde sledovanie steny, chod' na cieľ.
2. Ak je v ceste prekážka a nie si v móde sledovania steny, tak smeruj na jej hranu tak, aby bola euklidovská vzdialenosť, ktorú má robot prejsť čo najmenšia. Zapamätaj si aktuálnu vzdialenosť do cieľa.
3. Ak sa niekedy vzdialenosť od cieľa začne zväčšovať prepni sa na mód sledovanie steny.
4. Ak sleduješ stenu, tak ju sleduj až do prípadu až kým vzdialenosť do cieľa nie je menšia ako najkratšia zapamätaná vzdialenosť do cieľa.
5. Zastav, ak si dosiahol cieľ .

Úloha 3 Mapovanie (5b)

Úlohou mapovania je vytvoriť reprezentáciu prostredia na základe údajov zo snímačov robota. Inak povedané, mapovanie je fúzia (spájanie) informácií o polohe robota s informáciami o prekážkach získaných z laserového diaľkomera. Najjednoduchšia reprezentácia prostredia je mriežka obsadenia. Mriežka obsadenia rozdelí priestor na konečný počet prvkov (buniek), kde každý prvok obsahuje informáciu o jeho obsadenosti (t.j či sa tam nachádza prekážka).



Prerátanie údajov získaných z laserového diaľkomera do súradníc robota je možné na základe homogénnej transformácie. V prípade, že zanedbáme pohyb v osi Z (t.j hýbeme sa v rovine), ako aj rotácie okolo osi X a Y (robot sa pri pohybe nenakláňa do žiadneho smeru) je možné použiť homogénnu transformáciu pre 2 rozmery. 2D transformáciu je možné zapísať ako:

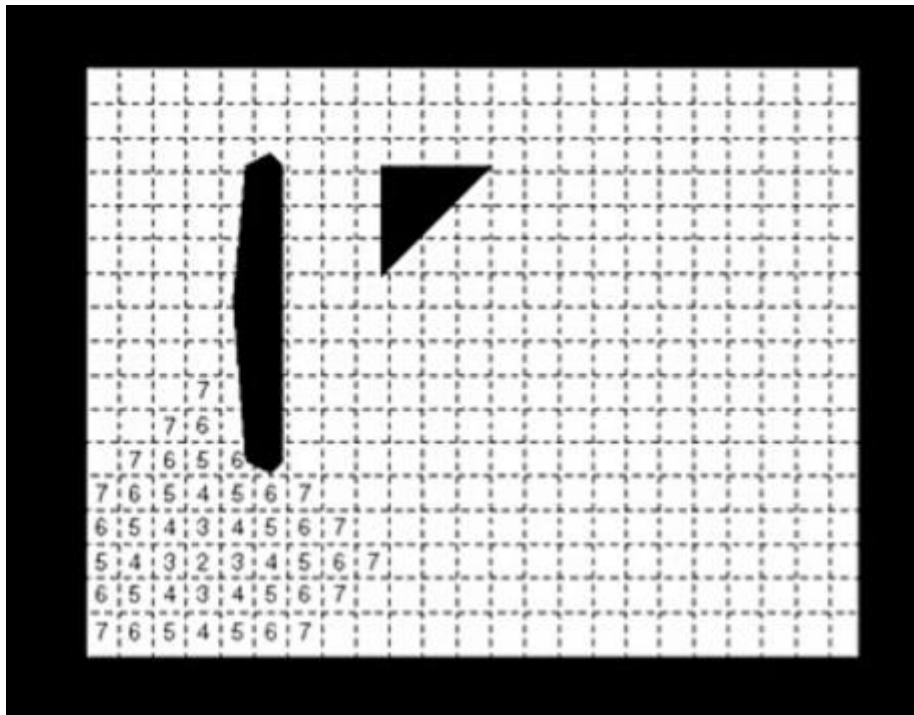
$$\begin{bmatrix} x_r \\ y_r \\ 1 \end{bmatrix} = \begin{bmatrix} \cos \varphi_k & -\sin \varphi_k & 0 \\ \sin \varphi_k & \cos \varphi_k & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_l \\ y_l \\ 1 \end{bmatrix} + \begin{bmatrix} x_k \\ y_k \\ 1 \end{bmatrix}$$

Blok 4 Plánovanie trajektórie (10b)

Úlohou plánovania trajektórie je vygenerovať optimálnu trasu pre mobilný robot na základe existujúcej mapy prostredia. Na tento účel sa využije typ mapy z predchádzajúcej úlohy a použitý algoritmus na plánovanie trajektórie je záplavový algoritmus (flood fill algorithm). Jeho princíp je nasledovný:

1. Cieľová bunka je ohodnotená na 2. Všetky bunky susediace s touto bunkou sú ohodnotené na 3.

2. Nulové bunky susediace s bunkami s hodnotami 3 sú ohodnotené na 4.
3. Procedúra pokračuje dovtedy, kým nie je spojený štart a cieľ.



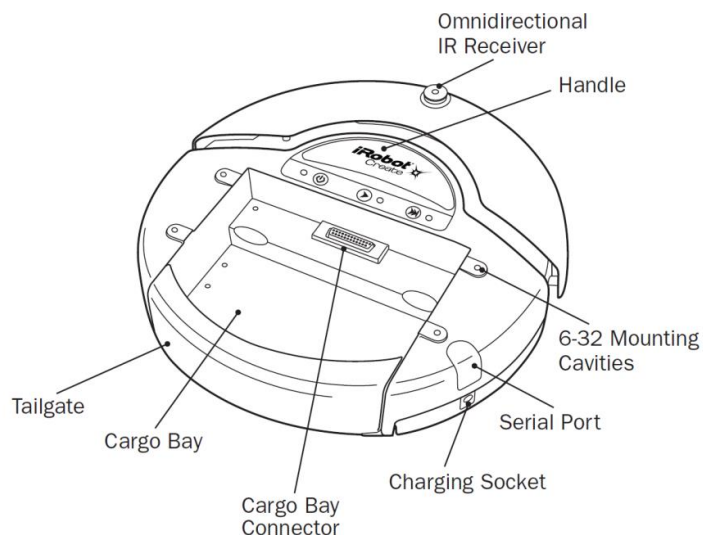
Bonusové úlohy

Bonusové úlohy sú úlohy, ktoré nie je možné zaradiť do predchádzajúcich blokov, ale napriek tomu patria k úlohám, ktoré sa riešia v mobilnej robotike.

3. Hardvér

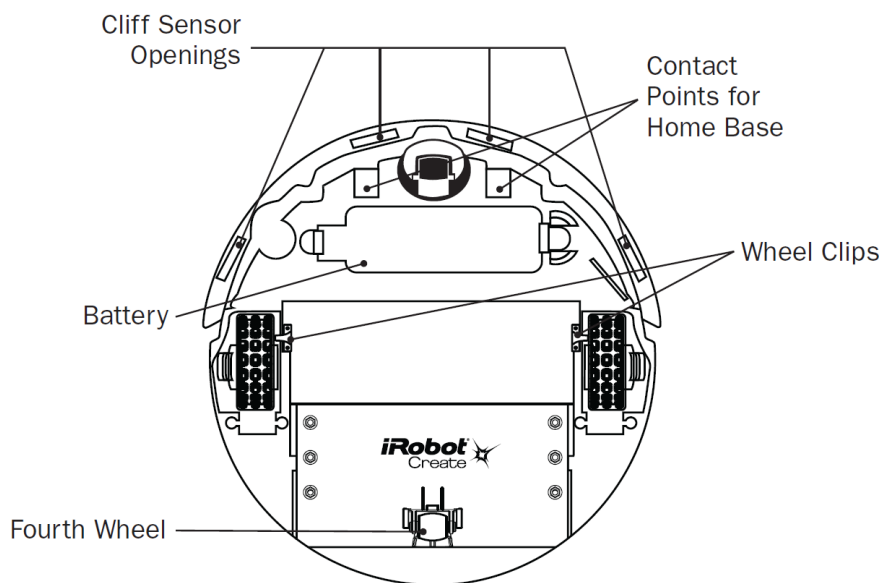
iRobot Create

Platforma iRobot Create vychádza zo známeho produktu firmy iRobot Roomba. Kinematika iRobot Create je postavená na diferenciálnom podvozku s dvoma pohonnými kolesami a dvoma opornými kolieskami.



Obr 1. iRobot Create (z vrchu)

Pre navigáciu poskytuje platforma údaje o posunutí ťažiska robota, jeho natočení a informácie z jeho početných snímačov. Jedná sa hlavne o bezpečnostné snímače o narazení, výskytu schodov ako aj o stave baterky.



Obr 2. iRobot Create (zo spodu)

Protokol Open interface

iRobot Create vyžíva na riadenie protokol Open interface(OI). Komunikácia prebieha na sériovej linke s parametrami

Baud: 57600

Dátové byty: 8

Parita: None

Stop bit: 1

Po zapnutí robota je nutné ho inicializovať príkazom Start(0x80), a následne nastaviť režim v akom chceme s robotom komunikovať a riadiť ho. Na tento účel poskytuje OI tri režimy, passive, safe a full. V režime passive je možné získavať informácie zo snímačov, ale nie je možné meniť hodnoty aktuátorov. Režim safe nám umožňuje robotom hýbať, ale v prípade ak nastane nebezpečná situácia(klesnutie kola, senzor schodov) robot zastane a prejde do režimu passive. V režime full sú všetky nebezpečné stavy na programátorovi ako ich ošetrí.

Príkaz	kód	dátový byte: 1	dátový byte: 2	dátový byte: 3	dátový byte: 4	nasledujúce byty
Start	128/0x80	-	-	-	-	-
Baud	129/0x81	Baud Code: (0 – 11)	-	-	-	-
Control	130/0x82	-	-	-	-	-
Safe	131/0x83	-	-	-	-	-
Full	132/0x84	-	-	-	-	-
Spot	134/0x86	-	-	-	-	-
Cover	135/0x87	-	-	-	-	-
Demo	136/0x88	Demá (-1 - 9)	-	-	-	-
Drive	137/0x89	Rýchlosť (-500 – 500 mm/s)		Polomer (-2000 – 2000 mm)		-
Low Side Drivers	138/0x8A	výstupné Bity (0 – 7)	-	-	-	-
LEDs	139/0x8B	LED Bity (0 – 10)	Power LED farba (0 – 255)	Power LED intenzita (0 – 255)	-	-
Song	140/0x8C	Číslo skladby(0 - 15)	Dĺžka skladby(1 - 16)	Číslo noty 1(31 – 27)	Trvanie noty 1(0 - 255)	Číslo noty 2, atď.
Play	141/0x8D	Číslo skladby (0 – 15)	-	-	-	-
Sensors	142/0x8E	Packet ID: (0 – 42)	-	-	-	-
Cover and Dock	143/0x8F	-	-	-	-	-
PWM Low Side Drivers	144/0x90	Low Side Driver 2 pracovný cyklus (0 - 128)	Low Side Driver 1 pracovný cyklus (0 - 128)	Low Side Driver 0 pracovný cyklus (0 - 128)	-	-
Drive Direct	145/0x91	Rýchlosť pravého kola (-500 – 500 mm/s)		Rýchlosť ľavého kola (-500 – 500 mm/s)		-
Digital Outputs	147/0x93	Output Bits (0 – 7)	-	-	-	-
Stream	148/0x94	Počet packetov	Packet ID 1 (0 – 42)	Packet ID 2, etc.	-	-

Query List	149/0x95	Packet ID 1 (0 – 42)	Packet ID 2, etc.	-	-	-
Pause/Resume Stream	150/0x96	(0-1)	-	-	-	-
Send IR	151/0x97	Byte (0 - 255)	-	-	-	-
Script	152/0x98	Dĺžka scriptu:(1 – 100)	Kód príkazu 1	Dátové byty príkazu 1, etc.	Kód príkazu 2 atď.	
Play Script	153/0x99	-	-	-	-	-
Show Script	154/0x9A	-	-	-	-	-
Wait Time	155/0x9B	čas (0 – 255seconds/10)	-	-	-	-
Wait Distance	156/0x9C	vzdialenosť (-32767 - 32768 mm)		-	-	-
Wait Angle	157/0x9D	uhol(-32767 - 32768 degrees)		-	-	-
Wait Event	158/0x9C	ID udalosti(1 to 20 and-1 to -20)	-	-	-	-

Tab. 1. Protokol open interface

Informácie o snímačoch sú v OI posielané po packetoch. Každý packet obsahuje informácie o iných snímačoch. Je teda možné vyžiadať z Create len informácie ktoré sú pre riadenie potrebné. Create obnovuje snímače s periódou 15ms. Za tento čas je pri sériovej linke s baud rate 57600 možné poslať 86 bytov. Preto nie je vhodné znižovať rýchlosť komunikácie ak chceme žiadať informácie zo všetkých snímačov.

	Packet membership						Name	Bytes	Value range	Unit	
0	1					6	7	Senzory narazenia a zdvihnutia kolies	1	0-31	
						8	Stena	1	0-1		
						9	Ľavý senzor schodov	1	0-1		
						10	Ľavý predný senzor schodov	1	0-1		
						11	Pravý predný senzor schodov	1	0-1		
						12	Pravý senzor schodov	1	0-1		
						13	Prítomnosť virtuálnej steny	1	0-1		
						14	nadprúdy	1	0-31		
						15	-	1	0		
						16	-	1	0		
		2	17	IR Byte	1	0-255					
			18	Gombíky	1	0-15					
			19	Vzdialenosť	2	-32768-32767	mm				

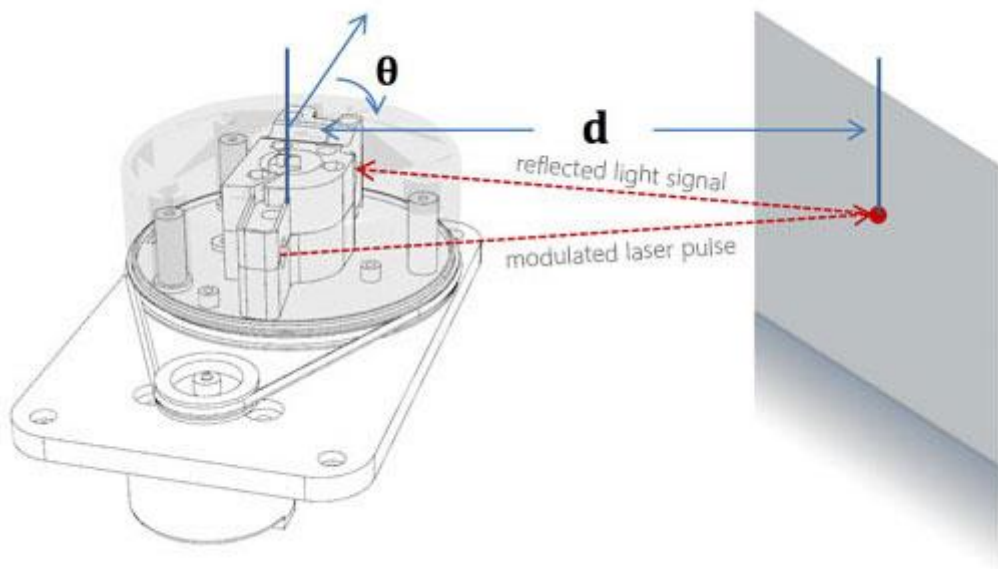
			3			20	Uhol	2	-32768-32767	degree
						21	Stav napájania	1	0-5	
						22	napätie	2	0-65535	mV
						23	prúd	2	-32768-32767	mA
						24	Teplota batérie	1	-128-127	Degree celsius
						25	Stav nabitia batérie	2	0-65535	mAh
						26	Kapacita batérie	2	0-65535	mAh
						27	Veľkosť signálu steny	2	0-4095	
						28	Veľkosť signálu ľavého senzoru schodov	2	0-4095	
						29	Veľkosť signálu ľavého predného senzoru schodov	2	0-4095	
						30	Veľkosť signálu pravého predného senzoru schodov	2	0-4095	
						31	Veľkosť signálu pravého senzoru schodov	2	0-4095	
						32	Digitálny vstup	1	0-31	
						33	Analógový vstup	2	0-1023	
						34	Prítomnosť nabíjacej stanice	1	0-3	
					5	35	OI mód	1	0-3	
						36	Číslo skladby	1	0-15	
						37	Ktorá skladba hrá	1	0-1	
						38	Počet packetov v streame	1	0-42	
						39	rýchlosť	2	-500-500	mm/s
						40	polomer	2	-32768-32767	mm
						41	Rýchlosť pravého kolesa	2	-500-500	mm/s

							42	Rýchlosť kolesa	ľavého	2	-500-500	mm/s
--	--	--	--	--	--	--	----	--------------------	--------	---	----------	------

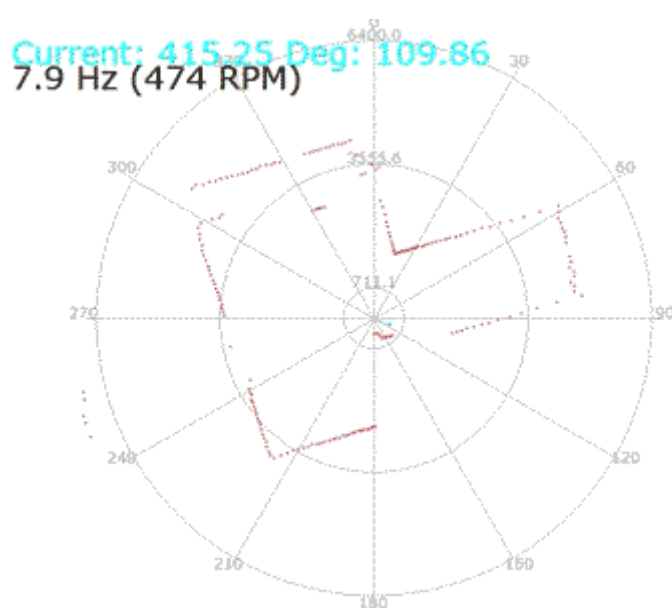
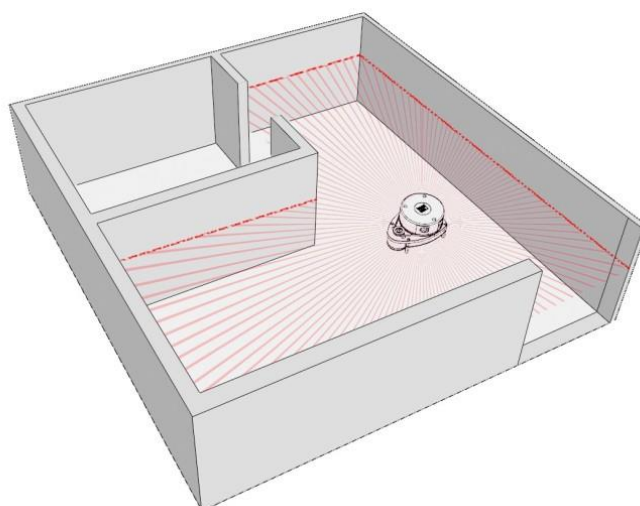
Tab. 2. Údaje na snímačoch a príslušnosť snímačov k jednotlivým packetom

Laserový diaľkomer RPLidar.

Laserový diaľkomer RPLidar je diaľkomer, ktorý meria vzdialenosť od robota k prekážkam v jednej rovine. Princíp fungovania je na základe triangulácie zosnímaného odrazu vysielaného laserového lúča.



Pokrytie celej plochy v okolí robota je dosiahnuté rotáciou hlavy laserového diaľkomera. Uhlové rozlíšenie snímača, je závislé od rýchlosti otáčania, na cvičeniach je nastavené na 0.8° až 1.3°.



Lidar poskytuje údaje o meraní vo forme stream-u meraní, kde každý blok obsahuje informácie o uhle (natočenie hlavy) , nameranej vzdialenosti a intenzite (množstvo energie, ktoré sa odrazí od povrchu).