



Středoškolská technika 2014

Setkání a prezentace prací středoškolských studentů na ČVUT

RobPi autonomní robot

Miroslav Konečný

SPŠ sdělovací techniky
Panská 3, Praha 1

Rozhodl jsem se navrhnout a zkonstruovat průzkumné vozítko, které bude schopné se autonomně pohybovat v domácím prostředí, v době mé nepřítomnosti. Vozítko jsem nazval RobPi. Pro realizaci jsem použil procesor PIC16F887, na který jsou napojeny detektory překážek.

RobPi je schopen po sériovém portu komunikovat s RaspberryPi, na kterém běží webový server. Webový server umožňuje uživateli komunikovat s robotem, nastavovat jeho periferie a ovládat jeho kameru tzn.: human interface device [HID]. Toto však není součástí zadání dlouhodobé maturitní práce, a proto se tím nebudu podrobně zabývat v dokumentaci.

Pokud chceme řídit robota, potřebujeme k tomu vhodný mikroprocesor. Měl by podporovat rozhraní RS232, abychom s ním mohli komunikovat. Dále by měl být vyroben v pouzdru s co nejvíce vývody, pro možnost připojení vícero čidel, a určitě by měl být vybaven i jinými sběrnicemi I2C, SSP pro možnosti dalšího rozšíření HW. Je nutný vhodný výběr čidel a to s ohledem na prostředí, ve kterém bude robot pracovat. S tím souvisí i výběr pohonu a podvozku robota. Podle nároků pohonného subsystému je zapotřebí zvolit i správný zdroj elektrické energie. Po důkladném prostudování a zvážení všech kladů a protikladů různých konstrukcí, jsem se rozhodl, vyrobit konstrukci vlastní. Nejprve jsem si konstrukci navrhl v grafickém editoru Google Sketchup.

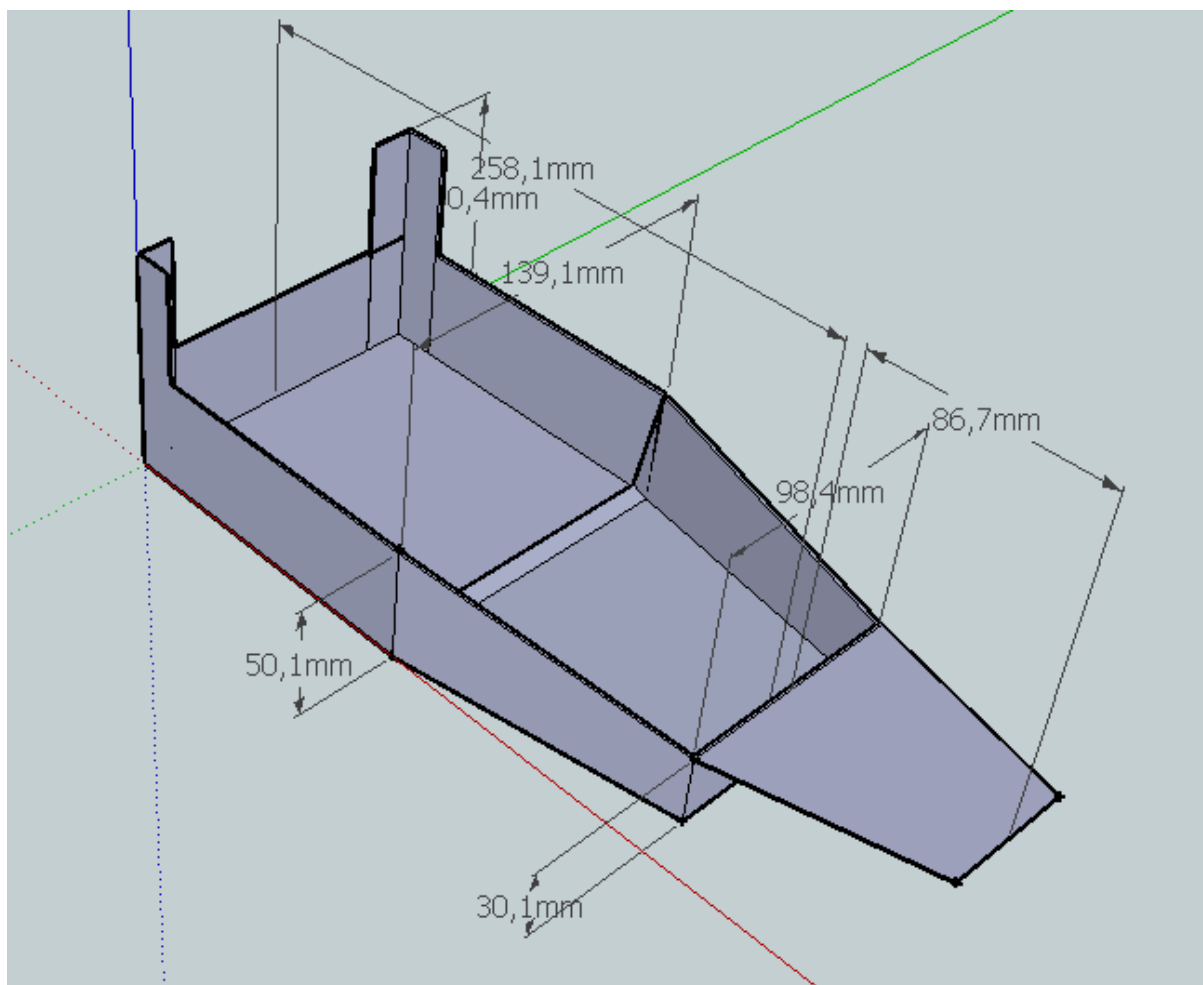
Navržená konstrukce má jednoduchý tvar a je dostatečně velká, aby se do ní vešly veškeré komponenty.

Vnitřní rozměry:

Délka – 35 cm

Šířka v nejširším místě – 15 cm

Šířka v nejužším místě – 10 cm



Obr. 1: Vlastní návrh podvozku

Použitý materiál:

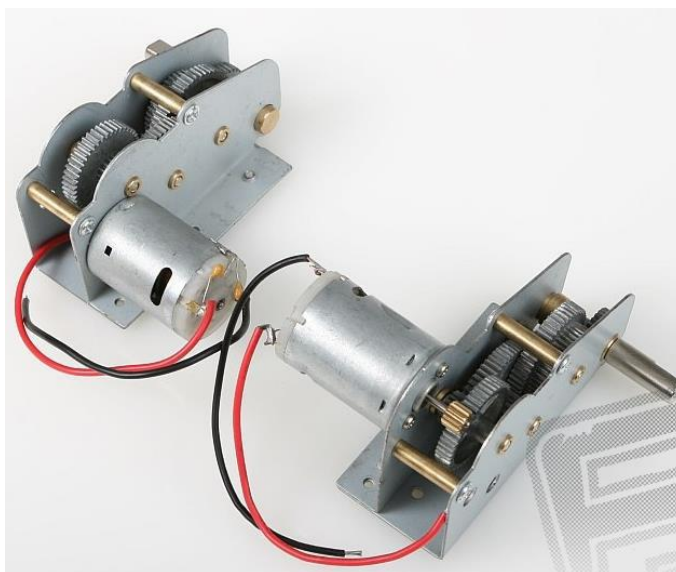
- 2 mm hliníkový plech
- „L“ profil z duralu
- Konstrukce je spojena pomocí nýtů

Veškerý použitý materiál na výrobu podvozku je vysoce vodivý. Proto je nutná řádná izolace od elektrických komponentů. Dno podvozku jsem proto izoloval černou nevodivou folií o tloušťce 4 mm.



Obr. 2: Realizace vlastního podvozku

Pohon obstarává dvojice stejnosměrných motorů a převodovek, které jsem zakoupil jako náhradní díly k RC tanku. Stejnosměrné motory jsou praktickou a levnou volbou. Ozubená kovová kolečka u převodovek zaručují nízké opotřebení.



Obr. 3: Zakoupená převodovka

Požadavky na řídicí mikroprocesor:

Paměť programu : 4K

Paměť dat : 256

Sběrnice : USART, SSP, I2C

Počet vstupů/výstupů : ≥ 25

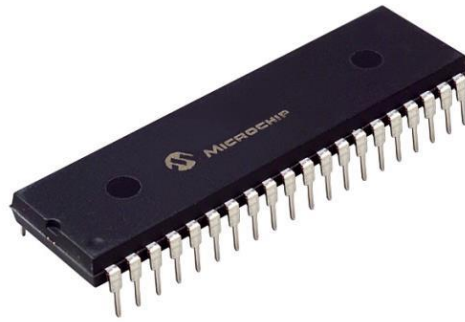
A/D převodník : ANO

PWM : ANO

Časovače : ≥ 3

Frekvence : $\geq 20\text{MHz}$

Bez dlouhého uvažování jsem zvolil PIC16F887. Je založen na stejné architektuře, jako PIC16F88, se kterým jsem se již setkal ve škole. Všechny jeho parametry jsou dostačující a je vybaven všemi periferiemi, které potřebuji.



Obr. 4: Mikroprocesor

Výběr čidel detekujících překážku

Hlavními kritérii při výběru čidel byly:

Cena

Jednoduchost (je/není překážka)

Rozměry



Obr. 5: IR dioda

Parametry:

Cena: 5,- Kč

Infračervená LED

Vlnová délka 940nm

Vyzařovací úhel 20°

Vhodný zdroj signálu pro SFH5110

Průměr 5mm Napájení 10-20mA



Obr. 6: IR přijímač

Parametry:

Cena: 25,-Kč

Infračervený přijímač

Novější náhrada za SFH506

Úhel příjmu 100° horizontálně, 60° vertikálně

Vhodný k detekci soupeře či překážek, komunikaci s PC, ovládání robota

Vyžaduje zdroj IR signálu modulovaného 36kHz (IR-LED nebo dálkové ovládání)

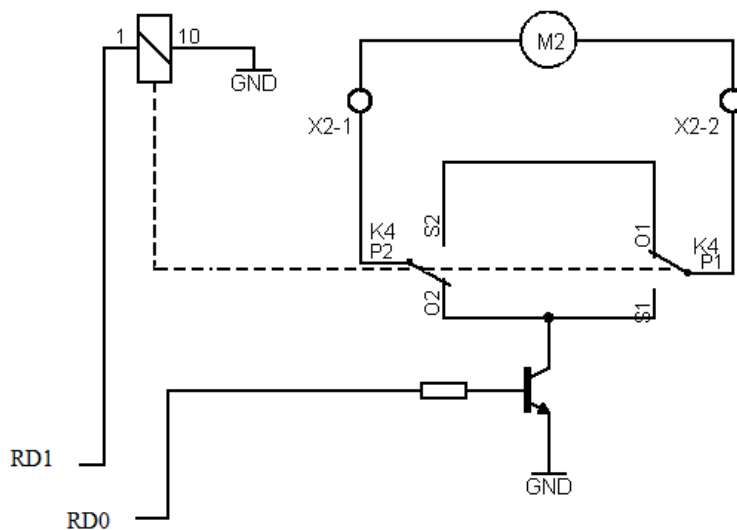
Rozměry 6x6x5mm

Napájení 5V/1.3mA

Výstupem je logický signál

Výběr vhodného ovládání pohybu

Možné řešení je kombinace tranzistoru s relé. Tranzistor by zapínal/vypínal chod motoru a relé by určovalo směr jízdy. Výhodou tohoto zapojení jsou pouze 2 součástky na motor, což velmi zlevní náklady a uvolní místo na plošném spoji. Odhadovaná cena je přibližně 50,- Kč. Toto je pro mě ideální řešení, a proto jsem si ho vybral. Nijak mě neomezuje doba sepnutí cívky relé, která je přibližně 120ms, jelikož relé bude měnit pouze polaritu na motoru.



Obr. 7: Blokové schéma zapojení relé a motorů

Vlastnosti autonomního režimu

Pokud má robot reagovat na překážky či předměty, musí být naprogramován tak, aby se řídil třemi základními pravidly robotiky:

1. Robot nesmí ublížit člověku nebo svou nečinností dopustit, aby bylo člověku ublíženo.
2. Robot musí poslechnout člověka, kromě případů, kdy je to v rozporu s prvním zákonem.
3. Robot se musí chránit před poškozením, kromě případů, kdy je to v rozporu s prvním nebo druhým zákonem.

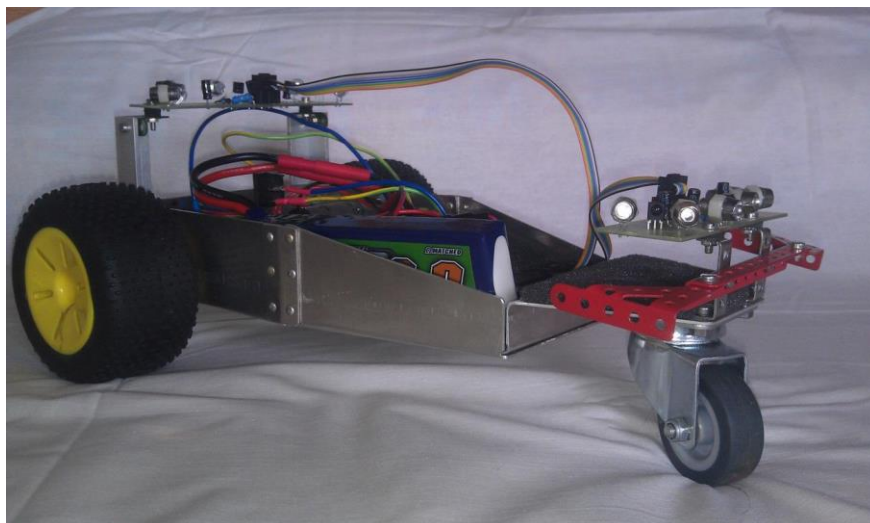
Autonomní režim pohybu neznamena správně vyhodnotit překážku a patřičně změnit směr jízdy tak, aby nedošlo ke střetu, ale jedná se o mnohem propracovanější stavbu programu. Zde je možno hovořit o jednoduché inteligenci.

Pokud RobPi zaznamená překážku, nejdříve si uloží do paměti, kterými čidly ji zaznamenal. Poté z pravdivostní tabulky vyčte řešení dané situace a následně změni směr jízdy. Dále vyhodnotí, zda nově zaznamenaná situace mu pomohla se vyhnout původní překážce. Pokud zaznamenaná z paměti, že se dostal do komplikovanější situace, kde je více překážek, znamená to, že špatně reagoval na danou předchozí situaci a veškeré údaje si uloží do paměti. RobPi sám dokáže vyhodnotit, že špatně zareagoval, a proto po „nějaký Z“ čas bude používat změněnou pravdivostní tabulku, podle které se bude řídit.

Tento druh intelligence se nazývá neuronová síť, která je řízena Hebbovým pravidlem – učitel, žák. Já jako programátor, jsem naučil robota několik základních pravidel (pravdivostních tabulek) a on si jenom vybírá, která je pro něho lepší v dané situaci. Vše se snaží zapamatovat, aby věděl, jak se má zachovat příště.

Programová část

Celý program jsem vymýšlel tak, aby byl logický, jednoduchý a snadno pochopitelný. Zvolil jsem programovací jazyk C. Tento programovací jazyk však zpomaluje rychlost konání některých instrukcí, proto jsem přemýšlel, že některé pasáže programu napíši v assembleru. Nakonec jsem od tohoto kroku ustoupil a s rozmyslem jsem vybíral pouze instrukce v jazyku C, které jdou do assembleru lehce přeložit.



Obr. 8: RobPi z levého boku