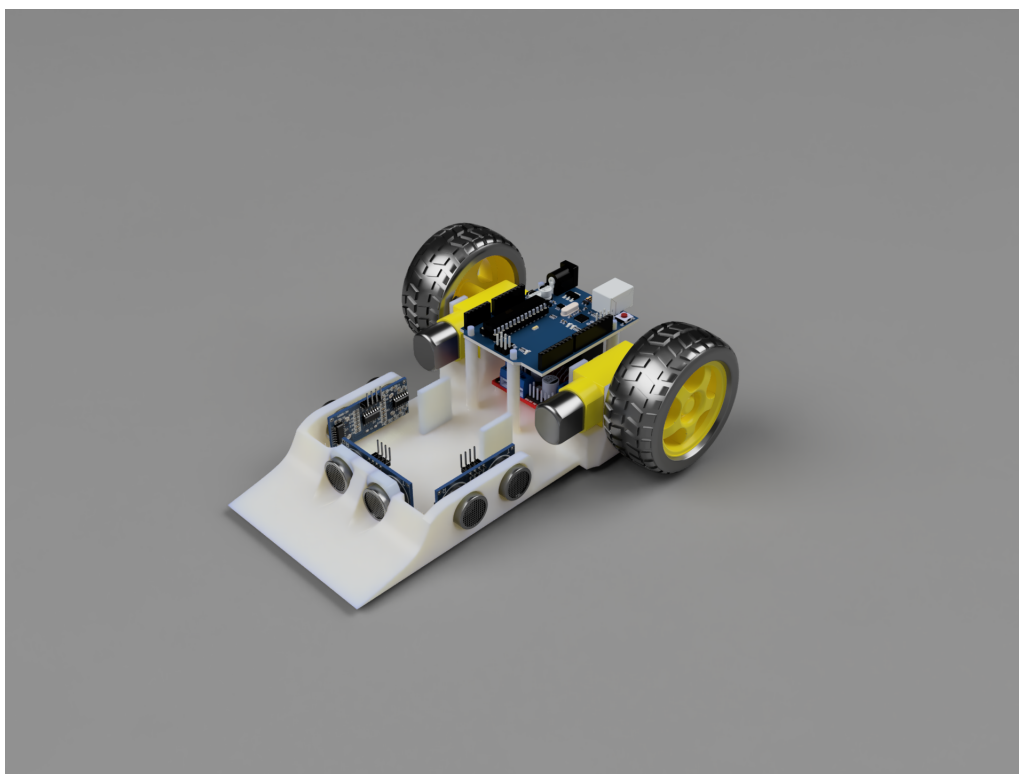


## Teorie řízení sumo robota



Západočeská Univerzita V Plzni  
Katedra Kybernetiky  
Semestrální práce - HKUI

David Žahour  
1. semestr  
3. ledna 2020

# 1 Konstrukce robota

## 1.1 Definice

Napřed je potřeba sumorobota definovat. Jedná se o robota, jenž je dán do kruhové arény, která má jedolitou barvu s kontrastním okrajejm a jeho cílem je vytlačit mimo arénu nepřátelského robota. Robotovo maximální rozměry nesmějí velikostně přesáhnout papír formátu A4 a maximální povolená výška je 15cm. Robot je konstruován tak, aby nedocházelo k destrukci arény.

## 1.2 Konstrukce

Vzhledem k tomu, že tato práce práce se zaměřuje na teorii řízení, bude popis konstrukce zjednodušen. Robot byl navržen v programu Fusion 360 a vytištěn 3D tiskárně z materiálu PLA (polymléčná kyselina). Robot ve svém základu počítá se 3 ultrazvukovými senzory, které jsou umístěny na bocích robota a vepředu. Robot je napájen liion dvojčlánkem o nominálním napětí 7.4 volt. Toto napětí je následně zvednuto na 12 volt pomocí DC/DC měniče. Na krajích robota se nachází infračervené senzory, jenž detekují okraje arény. K řízení je použit mikroprocesor atmega328pb, který je na vývojové desce Arduino. Motory jsou ovládány pomocí H bridge můstku. Mikroprocesor běží na taktu 8 mhz a jedná se architekturu avr8.

## 1.3 Infračervený senzor

Vstup z infračervených senzorů tcr5000. Senzor je složen z vysílací diody, která vysílá záření o vlnové délce 760 nm. To je odraženo od světlého povrchu (případně pohlcováno tmavým). Podle množství odraženého záření se mění napětí na vstupu. Mikroprocesor je vybaven 10 bitovým ADC převodníkem, což znamená že máme přibližnou přesnost 0.4 mV. Doba potřebná k měření na jednom ADC kanálu se pohybuje v rozmezí 60 až 250 s. Když budeme provádět dvě kontrolní měření, aby byla omezena možnost chyby měření, zjistíme, že v průběhu měření je robot maximálně schopný urazit vzdálenost 0,02 cm. To znamená, že pro rozhodovací proces je doba vzhledem ke kapacitě soustavy (hmotnost a hybnost robota) zanedbatelná.

## 1.4 Ultrazvukové senzory

Měření a detekce dalších robotů. K detekci dalších robotů se používá senzor hc sr04. Senzor pracuje na principu vysílání a přímání ultrazvukových vln. Senzor se aktivuje vysláním 15 mikrosekundového obdelníku na pinu s označením Trig. Po pulsu je vyslán signál modulován na 40kHz. Zároveň s vysláním signálu dojde k nastavení logické jedničky na pin Echo. Echo pin je nastaven v logické jedničce dokud nedojde k odrazu zpět do senzoru nebo vypršení timeoutu. Zde záleží na rychlosti zvuku, abychom detekovali vzdálenost od objektu. Budeme proto uvažovat dané podmínky (Suchý vzduch při teplotě 20C). Při těchto podmínkách je rychlost šíření vln ve vzduchu 343m/s. Maximální přesnost měření času v našem mikroprocesoru vychází ze zdroje hodinového signálu, který je roven 8Mhz. Když je vnitřní časovač nastaven do režimu přičítání, tak doba jednoho tiků odpovídá 125 nano sekundám. To znamená, že za jeden tik vlna urazí vzdálenost 0.042875 milimetru. Nesmíme ovšem zapomenout, že vlna poputuje danou vzdáleností dvakrát, tudíž jeden tik odpovídá vzdálenosti 0.08575. Můžeme vidět, že lze použít daleko menší přesnost na čítačích, protože hlavní neznámou je vlhkost a teplota okolí.

## 1.5 Maximalní rychlost

Maximalní rychlost robota vychází z kola robota. Průměr kola je 65 mm ,to znamená, že při jedné otáčce byla uražená vzdálenost 204 mm. Motor udává maximálně 110 otáček za minutu, to znamená, že maximální rychlost za optimálních podmínek je 0.4 metru za vteřinu.

## 1.6 Prioritizace vstupů

Nyní, když můžeme začít uvažovat vstupy, je potřeba je seřadit dle priorit, čímž zároveň začneme sestavovat základní pyramidu potřeb robota. Potřeba číslo jedna je potřeba přežití, to znamená nevyjždět na bílé pole a optimálně se od něj co nejvíce vzdálit. Priorita číslo dvě je detekovat objekt na bocích robota, to znamená riziko potenciálního vytlačení z kruhu. Poslední potřeba robota je potřeba útočit.

## 1.7 Prosec přežití

Teď je otázka, jak dané chování naprogramovat. Při použití programovacího jazyka C bohužel nemáme výhodu dostupnosti pokročilých knihoven pro umělou inteligenci. Vytvoříme proto základní cyklus operací seřazených dle priorit potřeb robota. Jelikož architektura AVR8 nemá k dispozici subroutine proces, není možné nechat zjišťování, zda se nenachází na bílém pozadí. Proto je potřeba toto měření provádět periodicky za každý časový úsek. Tento úsek spočteme z maximální rychlosti a velikosti bílého pole. Rozdíl mezi okrajem arény a černým polem je 5 cm bílého pruhu. Při maximální rychlosti je robot schopný za 1 vteřinu překonat tento prostor až 6 krát. Tudíž je potřeba měřit minimálně každých 166 milisekund. Z výpočtů výše můžeme vidět, že doba měření je zanedbatelná. Abychom měli dostatečnou rezervu, tak měření provedeme každých 100 milisekund. O měření času se nám bude starat Timer0 s nastavenou největší prioritou přerušování.

## 1.8 Proces hledání a útoku

Další proces bude detekce objektů na straně robota. Zde se budeme snažit o co největší frekvenci. Po detekci objektu je potřeba rozhodnout o vlastnostech objektu - o rychlosti a o tom, zda se přibližuje či vzdaluje. V případě, že dojde ke změně vzdálenosti mimo odchylku, spočteme rychlost nepřátelského objektu (dle vzorce).

$$v = \frac{s_{\Delta}}{t_{\Delta}} \quad (1)$$

V případě že se objekt přibližuje vysokou rychlostí je potřeba co nejrychleji změnit pozici, tudíž se navýší rychlost v současném směru pohybu a následně začne natáčení ve směru nepřátelského robota. Další možnost je, že robot se nepřiblížil ani neodálil. V tento okamžik je nejlepší možnost okamžitě zahájit operaci natočení ve směru nepřátelského robota a zahájit útok. Nevýhoda takového přístupu k obraně je, že se nebere v úvahu možnost blízkosti čáry, tudíž při útěku může dojít k nekončené smyčce, protože motory nejsou vybaveny jakoukoliv formou snímání otáček. Také je možné, že dojde k překmitu či nedokonalému natočení.

## 1.9 Další možnosti vylepšení robota

Robot z hlediska strojového učení. Poměrně zajímavá otázka je, jak by se robot dal trénovat pomocí strojového učení, respektive co by měl algoritmus za vstupy a výstupy. Jako vstup by se dali uvažovat analogové hodnoty ze senzoru čáry to znamená  $3 * 2$  na 10 vstupů, dalším vstupem jsou ultrazvukové senzory a to by znamenalo, že při rozlišení 1mm a max vzdálenosti 4 metry máme další  $4 * 1000$  vstupních hodnot. Jako hodnocení by se dala použít doba od vytlačení objektu, to by znamenalo, že v průběhu učení by robot měl snahu dosáhnout co největšího optima.