



Střední průmyslová škola,
Česká Lípa, Havlíčkova 426, příspěvková organizace

tel.: **487 833 123**

fax: **487 833 101**

email: **sps@sps-cl.cz**

web: **www.sps-cl.cz**

MATURITNÍ PRÁCE

ROBOTICKÉ VOZÍTKO

Studijní obor: **18-20-M/01 INFORMAČNÍ TECHNOLOGIE**

Autor:

Jan Bittner

Podpis:

Vedoucí práce:

Ing. Josef Bašta

Třída: **4.D**

Školní rok: **2015/2016**



Prohlášení

„Prohlašuji, že jsem tuto práci vypracoval samostatně a použil jsem literárních a dalších pramenů a informací, které cituji a uvádím v seznamu použité literatury a zdrojů informací.“

V České Lípě, 15. 4. 2016

Jan Bittner

.....

Poděkování

Rád bych poděkoval Ing. Josefu Baštovi za obětavou pomoc, odborné konzultace a poskytnuté informace, které mi během práce poskytoval. Dále bych chtěl poděkovat Střední průmyslové škole Česká Lípa za umožnění pracovat v odborných laboratořích a poskytnutí materiálu na sestavení robotického vozítka.

Anotace

Tato práce se zabývá sestavením robotického vozítka ze stavebnice Merkur, vytvořením vlastní řídicí jednotky a naprogramováním mikroprocesoru; myšleno je i na případnou budoucí modifikace jednotlivých částí. Hlavní účel práce je zdokonalení se v oblasti robotiky a mikroprocesorové techniky.

Výsledkem práce je textová část, obsahující popis použitých součástek či částí pro sestavení robotického vozítka, funkční robotické vozítko, připravené na případné budoucí vylepšování, a software pro řídicí mikroprocesor.

Annotation

This work deals with constructing a robotic rover from the toy construction kit Merkur, creating its own control unit and programming a microprocesor. Everything is prepared with a thought of future modifications of individual parts. The main purpose of this work is to enhance my skills in the field of robotics and microprocessor technology.

The result of this work is a text part containing a description of the components to build a robotic rover, functional robotic rover prepared for any future improvements and software for control microprocessor.

Obsah

Použité značky, zkratky a symboly.....	7
1. Úvod	8
2. Hardware	9
2.1. Model robotického vozítka	10
2.2. Motory	11
2.3. Elektronika.....	12
2.3.1. Řídící jednotka.....	12
2.3.2. Budič motorů	14
2.4. ATmega32	16
2.5. Dálkové ovládání	18
2.6. Snímač vzdálenosti	19
2.6.1. Princip fungování modulu	20
3. Ovládání vozítka.....	22
3.1. Pohyb	22
3.2. Otáčení na místě.....	22
3.3. Speciální funkce.....	22
3.3.1. Přepnutí rychlosti.....	22
3.3.2. Přepnutí světel	23
4. Software.....	24
4.1. Jazyk symbolických adres	24
4.2. Nastavení	24
4.3. Hlavní cyklus	24
4.4. Řízení jízdy	26
4.5. Detekce vzdálenosti	27

4.5.1. Princip fungování přerušení snímače	28
4.6. Rychlost pohybu	31
4.6.1. Princip změny rychlosti	31
4.7. Rozsvěcení světel.....	32
5. Závěr.....	33
6. Použité zdroje	34
7. Seznam použitého software	36
8. Seznam obrázků a tabulek	37
9. Seznam příloh	39
10. Licenční ujednání	40

Použité značky, zkratky a symboly

LED – Light-Emitting Diode (dioda emitující světlo)

PWM – Pulse Width Modulation (pulzně šířková modulace)

DPS – Deska plošných spojů

JTAG – Joint Test Action Group

IR – Infrared (infračervené záření)

RF – Radio Frequency (rádiové záření)

TTL – Transistor-Transistor-Logic (tranzistorově-tranzistorová logika)

PDIP – Plastic Dual In-line Package

CPU – Central Processing Unit (centrální procesorová jednotka)

AVR – rodina 8bitových mikročipů typu RISC

RISC – Reduced Instruction Set Computing (počítač s omezenou sadou instrukcí)

SPI – Serial Peripheral Interface (sériové periferní rozhraní)

I2C – Inter-Integrated Circuit

UART – Universal Synchronous / Asynchronous Receiver and Transmitter

JSA – Jazyk Symbolických Adres

ISP – In-system programming

1. Úvod

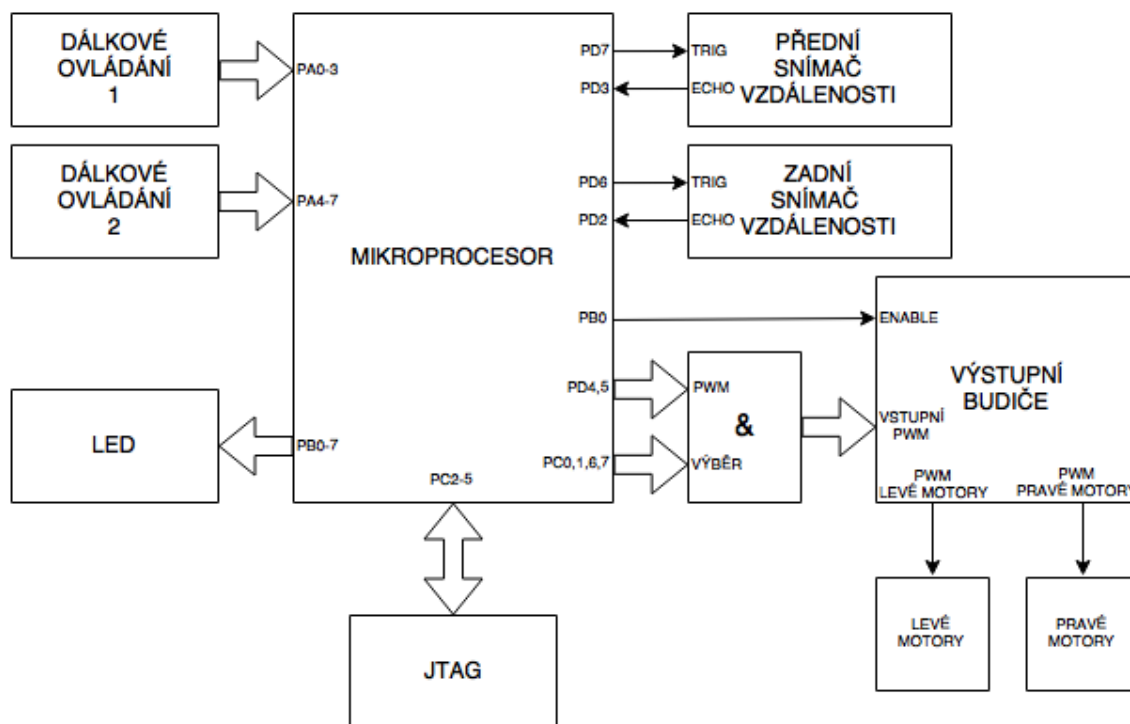
Robotika a mikroprocesorová technika jsou velmi zajímavé pojmy a ještě zajímavější jsou oblasti, které tyto pojmy reprezentují. A člověk je zajímavá bytost, která zbožňuje všelijaké „robůtky“ a nejraději by je také vytvářel, nebo alespoň řídil či ovládal. Člověk „od přírody“ zbožňuje vše, co bliká či jinak svítí a ještě více to, co se pohybuje.

Má motivace vychází právě z těchto poznatků a prožitků, navíc je umocněna chutí naučit se něco navíc o praktickém vytváření „robůtky“ a v tom se také zdokonalit.

Důležitými body celé práce je přijatelná cena všech částí výsledného robotického vozítka a vytváření hardwaru i softwaru tak, aby vše bylo pokud možno univerzální a aby se daly jednotlivé části snadno nahrazovat či vylepšovat.

2. Hardware

Robotické vozítko se skládá z několika samostatných částí. Všechny části, kromě dálkového ovládání, jsou umístěny v modelu robotického vozítka.



Obrázek 1 – Blokové schéma

Hlavní částí je mikroprocesor, který řídí veškerou logiku. Mikroprocesor je programován přes JTAG rozhraní, přes které se mikroprocesor také debuguje, tj. odchyťování chyb.

Do mikroprocesoru vstupují signály z modulů dálkových ovladačů, které slouží pro manuální ovládání vozítka. Z mikroprocesoru vystupují signály pro LED, značící stav vozítka nebo přední a zadní světla vozítka.

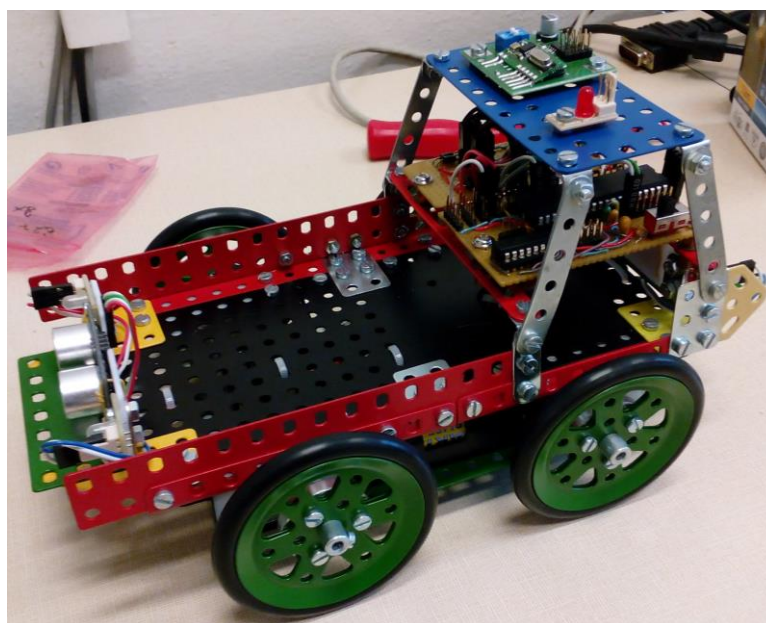
Mikroprocesor komunikuje obousměrnou komunikací s moduly snímačů vzdálenosti, kterým se pošle impuls a modul následně pošle vzdálenost od překážky.

Další důležitou částí jsou výstupní budiče, které přijímají od mikroprocesoru signál *enable*, který určuje, zdali je modul s budiči zapnut. Od mikroprocesoru se také

získává čtveřice PWM signálů, které se průchodem přes and hradlo a výstupní budiče přetvoří do výstupních PWM ovládající přímo motory vozítka.

2.1. Model robotického vozítka

Model robotického vozítka je sestaven ze stavebnice Merkur podle vlastního návrhu. Jako základ je použita stavebnice „Kolový podvozek 01“. Oproti původnímu provedení je rozšířen o 20 mm a upraven tak, aby splňoval specifické požadavky, jako je např. prostor pro desky plošných spojů, resp. elektroniku obecně, či nákladový prostor. Obsahuje také ochranné prvky sloužící pro ochranu jednotlivých součástí a modulů při případném nárazu, kolizi či překlopení vozítka.



Obrázek 2 – Sestavené robotické vozítko

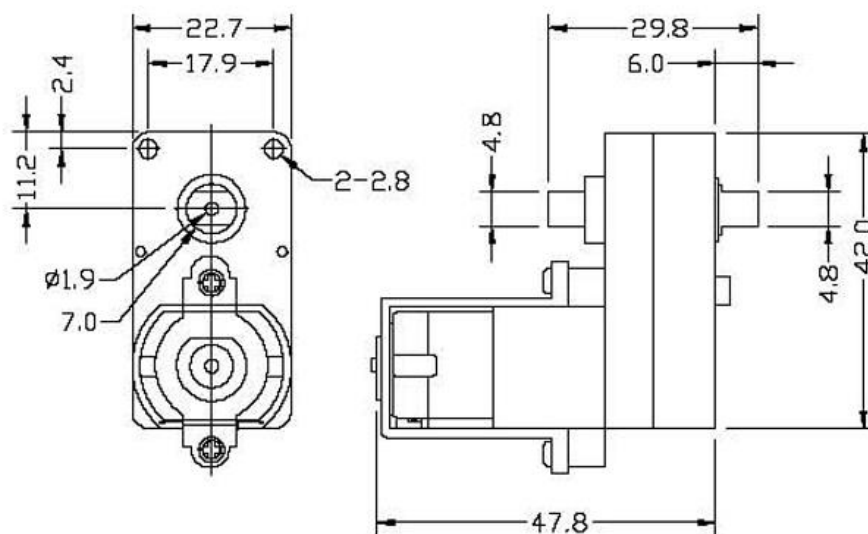
Zespodu podvozku modelu jsou umístěny baterie, pro napájení řídicí jednotky a motorů, které jsou skryty pod ochrannými dvířky. Pod modelem jsou dále umístěné 4 samostatné motory, na které se přímo připojují kola vozítka.

V kabině modelu robotického vozítka jsou 2 desky plošných spojů obsahující řídicí jednotku, zde se nachází mikroprocesor, a budiče motorů. Na střeše vozítka se nachází několik signalizačních LED a modul dálkového ovládání.

Na přední a zadní části modelu jsou umístěny ultrazvukové snímače vzdálenosti spolu s dvojicemi LED, bílými představující přední světla a červenými představující zadní světla.

2.2. Motory

Robotické vozítko využívá čtyři motory s převodovkou BO8 ze stavebnice Merkur. Na tyto motory jsou upevněna kola. Kola nelze natáčet a řízení je tak nutno provádět změnou rychlostí jednotlivých kol.



Obrázek 3 – Technický výkres motorů

Motory přijímají informaci o rychlosti otáčení z PWM, které je přiváděno z výstupu budičů motorů. Pro řízení daných motorů platí, že střída PWM signálu 100:0 je rovno plné rychlosti a 0:100 je rovno nulové rychlosti. Pro dosažení poloviční rychlosti je nutné dodat střidu signálu zhruba 65:35.

Tabulka 1 – Parametry motorů

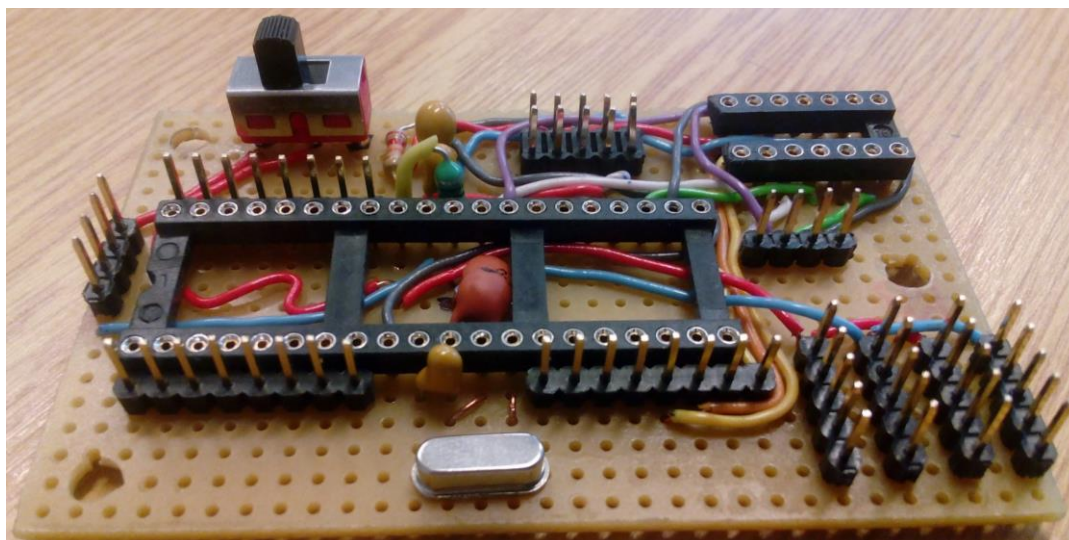
vlastnost	hodnota
cena	156 Kč
napájení	3-12 V
přechodový poměr	1:228
rychlost otáčení	36-58 ot/min

2.3. Elektronika

Robotické vozítko je vybaveno 2 vlastními DPS. Jednotlivé obvody jsou napájeny na jednostranné univerzální desky plošných spojů, které jsou cenově nejdostupnější a umožňují poměrně jednoduchou modifikaci, oproti leptaným DPS. Jelikož je využíváno univerzálních DPS, jednotlivé kontakty jsou spojeny vodiči, na povrchu obalenými bužírkou tvořící izolaci.

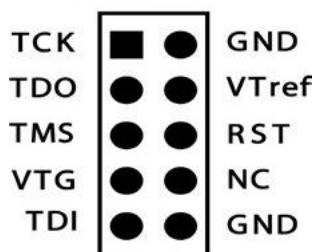
2.3.1. Řídící jednotka

Řídící jednotka zpracovává vstupně-výstupní komunikaci se všemi součástmi robotického vozítka. Jak lze vidět na obrázku 4 či v příloze B, DPS řídící jednotky obsahuje 40pinovou patici pro připojení mikroprocesoru ATmega32 v PDIP pouzdře, který vykonává a zpracovává všechny akce.



Obrázek 4 – DPS řídicí jednotky

DPS také obsahuje vypínač pro řízení přívodu napájení a piny pro připojení JTAG rozhraní, které umožňuje, např. oproti ISP rozhraní, program nahrávat i ladit. Díky tomu je možné program krokovat, kontrolovat aktuální stavy registrů nebo je měnit atp.



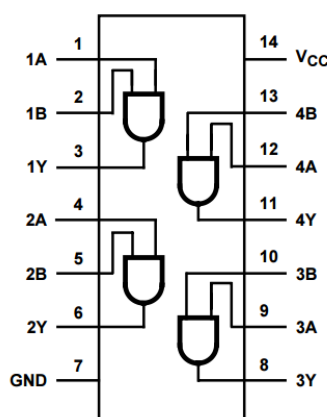
Obrázek 5 – JTAG rozhraní

DPS dále obsahuje piny pro 8bitové porty A, B a D, připojené na výstupy z patice mikroprocesoru, což umožňuje připojit „cokoli kamkoli“, pokud se změny dodrží i v programu, a obvod je tím pádem více univerzální.

Piny 2, 3, 4, 5 portu C jsou napevno propojeny s JTAG rozhraním. Piny 0, 1, 6, 7 portu C jsou napevno propojeny do AND hradel, které jsou obsaženy v součástce CD74HCT08E umístěné v 14pinové patici.

Zbylé 4 piny portu C jsou po dvojici rozděleny, zvlášť pro pravou a levou stranu motorů vozítka, na zapínání chodu motorů vpřed a vzad. Mikroprocesor také generuje 2 PWM signály, taktéž zvlášť pro pravou a levou stranu vozítka.

Do dvojice AND hradel přichází jeden PWM signál a zvlášť dvojice z pinů určujících směr motorů. Tímto způsobem lze lehce docílit ovládání směru jízdy robotického vozítka.

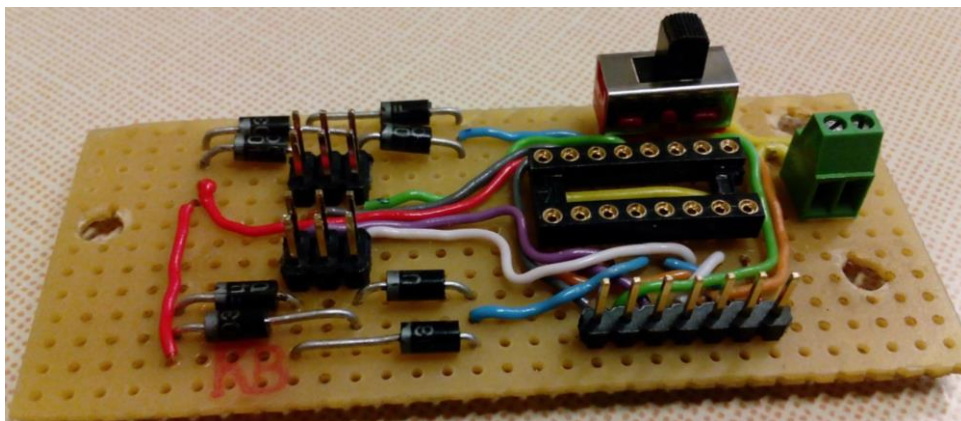


Obrázek 6 – Vnitřní zapojení AND hradel

V neposlední řadě DPS obsahuje 2 pětky pinů pro připojení k zemi a k napájení, vhodné zejména pro připojené externí součástky, např. přední a zadní světla robotického vozítka, a 4 piny pro výstup výsledných PWM signálů z AND hradel.

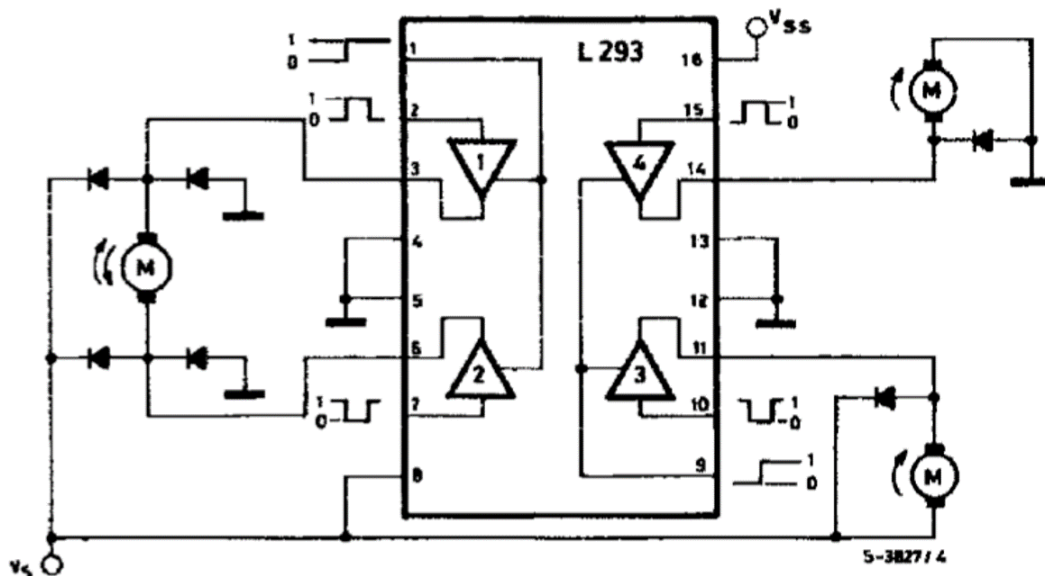
2.3.2. Budič motorů

Druhá DPS obsahuje budič motorů L293B. Obsahuje, podobně jako hlavní DPS s řídicí jednotkou, vypínač pro ovládání přívodu napájení a to hlavně z toho důvodu, jelikož napájení motorů a řídicí jednotky je oddělené. Budič motorů má provozní napětí od 4,5 V až do 36 V.



Obrázek 7 – DPS budiče motorů

DPS obsahuje 16pinovou patici pro budič motorů. Do zapojení je z řídicí jednotky přiváděno napětí a zem pro vnitřní elektroniku budiče, signál *enabled* pro zapnutí obvodu a 4 signály PWM – vždy dvojice pro jednu stranu, jeden pro směr vpřed, druhý pro směr vzad. Schéma, viz níže, je rozdvojené, používá se levá polovina na obou stranách.



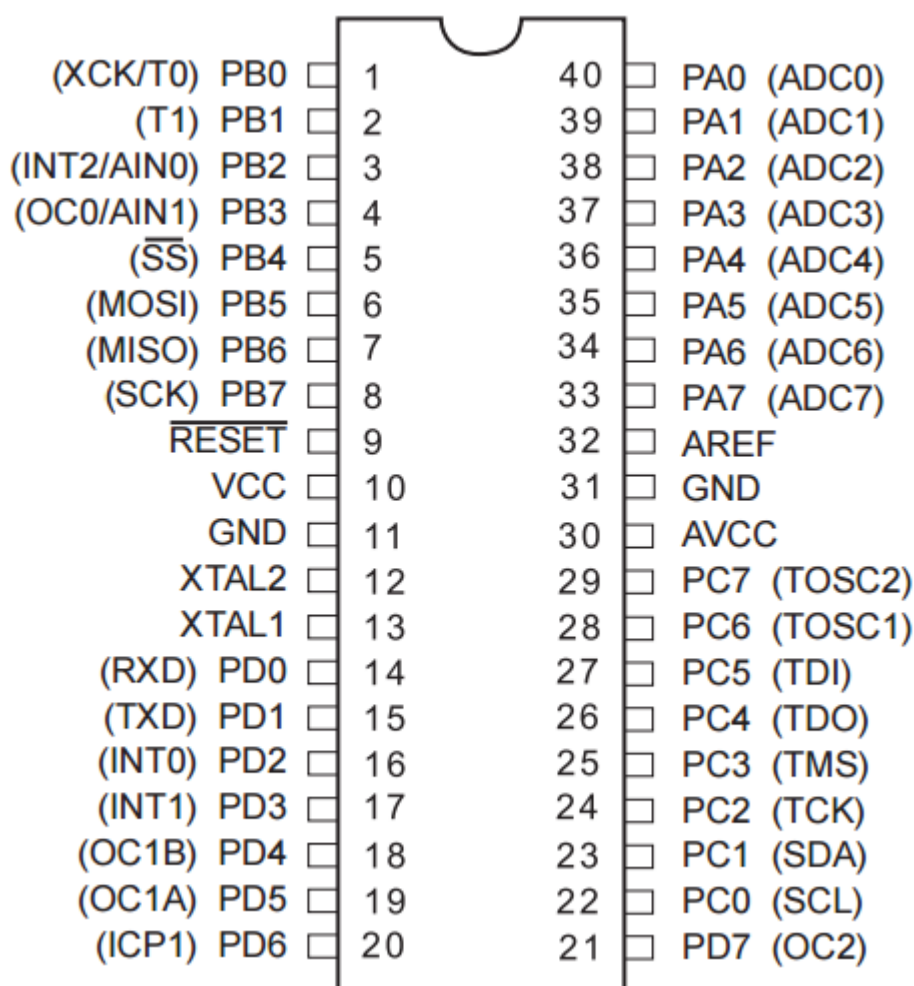
Obrázek 8 – Schéma budiče motorů

Jako výstup této DPS jsou dva třípinové konektory, pro motory pravé a levé části, sloužící pro připojení motorů.

2.4. ATmega32

Robotické vozítko je řízeno řídicí jednotkou, která obsahuje mikroprocesor ATmega32 vyráběný firmou Atmel. Tento mikroprocesor je využíván pro řízení všech částí autíčka.

Mikroprocesor poskytuje čtyři 8bitové I/O porty využívané pro získávání a poskytování dat, např. pro ovládání světel, signalizačních LED, získávání dat z dálkového ovládání nebo snímačů vzdálenosti. Disponuje také generátory PWM, které jsou využívány pro ovládání motorů.



Obrázek 9 – ATmega32 PDIP pouzdro

Paměť mikroprocesor lze opakovaně přepisovat, což se hodí zejména pro testování a vývoj; je možno uskutečnit přibližně 10000 zápisů. Program se do mikroprocesoru nahrává pomocí JTAG rozhraní, jak již bylo popsáno výše.

Pro vývoj programu je k dispozici, volně ke stažení na webových stránkách společnosti Atmel, AVR studio, které podporuje ladění v debug módu, což lze využít pro nastavování stavů registrů i I/O portů, nebo pro získávání jejich stavu, za běhu. Program lze jednoduše krokovat, přičemž se projevují změny v registrech a I/O portech, které mohou pomoci k odhalení chyby.

Tabulka 2 – Parametry ATmega32

vlastnost	hodnota
Flash paměť	32 kB
SRAM paměť	2 kB
EEPROM paměť	1 kB
počet pinů	40
vstupně-výstupní piny	32
registry	32 8bitových
CPU	8-bit AVR
rozhraní	SPI, I2C, UART...
ladící rozhraní	JTAG
pracovní napětí	2,7 V – 5,5 V
čítače / časovače	2x 8bitový, 1x 16bitový

Přibližná cena mikroprocesoru ATmega32 při koupi jednoho kusu je 220 Kč. Při koupi např. 100 ks vychází cena na 150 Kč na kus.

Pro testování modulů, např. modulu snímače vzdálenosti či testování funkčnosti motorů, bylo využíváno polského vývojového kitu EVB 5.1 vyvíjeného společností

And-Tech. Při zapojování mikroprocesoru ATmega 32 bylo vycházeno z manuálu a schémat zapojení tohoto vývojového kitu.

2.5. Dálkové ovládání

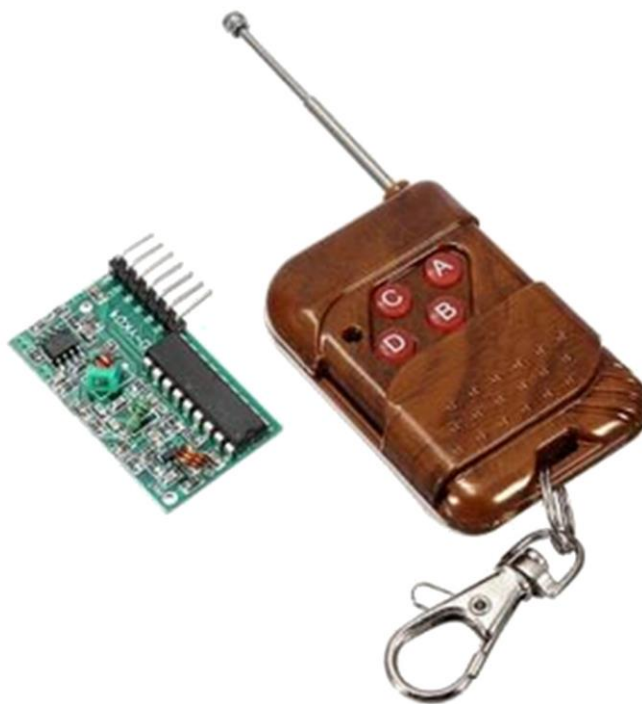
V průběhu práce se bylo nutno utkat s několika problémy. Největším problémem byla nefunkčnost dálkového ovladače dodávaného společností Merkur k jejich stavebnici, u kterého bylo nutno pro správnou funkčnost zaměnit původní velkou anténu za kus drátu a také odstínit přijímač, což ale ve výsledku stálou správnou funkčnost nenavodilo, jelikož se přijímač po zhruba 5 až 10 minutách zahltl a přestal na čas fungovat.

Dálkovému ovládání je vyčleněn samostatný port, tedy 8 pinů, na mikroprocesoru, kde jeden pin určuje stav jednoho tlačítka. Zmáčknuté tlačítko je detekováno na logickou „0“, a tedy pokud se nedrží žádná tlačítka, všechny piny daného portu budou obsahovat logickou „1“.

Do té doby, dokud bude vysílač a přijímač fungovat tímto režimem, a většina takto funguje, je možno libovolně vyměňovat vysílače a jejich přijímače. Při nasazení vysílače jeho přijímače, které mají kódování tlačítek v jiném režimu, je nutné změnit chování programu změnou masek pro daná tlačítka.

Pro dálkové ovládání robotického vozítka je doporučeno a realizováno využití RF vysílače a přijímače, což má oproti IR ovládání několik výhod, např. průchodnost překážkami.

Ze zahraničních portálů, zejména z Číny, lze přijímač a vysílač koupit pod 100 Kč, v závislosti na počtu tlačítek, kvality a dosahu. Nevýhodou oproti IR je, že se nedá levně koupit vysílač a přijímač s velkým počtem tlačítek a tedy např. při ceně 100 Kč lze pořídit jednoduchý RF vysílač se 4 tlačítky, jak lze vidět na obrázku 10 představující zakoupený modul dálkového ovládání, zatímco za stejnou cenu lze pořídit IR vysílač a přijímač s 20 tlačítky.



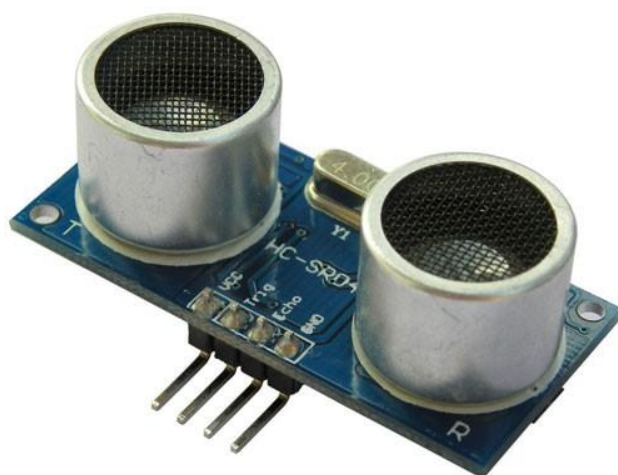
Obrázek 10 – Přijímač a vysílač modulu dálkové ovládání

Vozítko obsahuje 2 moduly dálkového ovládání, spolu s jejich ovladači po 4 tlačítkách. Aby se vysílače nerušily, je využíváno 433 MHz a 315 MHz varianty.

2.6. Snímač vzdálenosti

Robotické vozítko využívá dva ultrazvukové snímače vzdálenosti. Ty jsou umístěny vpředu a vzadu na modelu a lze pomocí nich zjistit vzdálenost od překážek a zabránit tak případným kolizím.

Je využíváno modulu HC-SR04. Ten vyžaduje napětí 5 V a dokáže detekovat vzdálenost od 2 cm do 400 cm. Tento modul lze v Česku pořídit za 120 Kč a výše, ale na zahraničních portálech nabízejících zboží např. z Číny lze součástku zakoupit za cenu kolem 20 Kč.



Obrázek 11 – Ultrazvukový snímač vzdálenosti HC-SR04

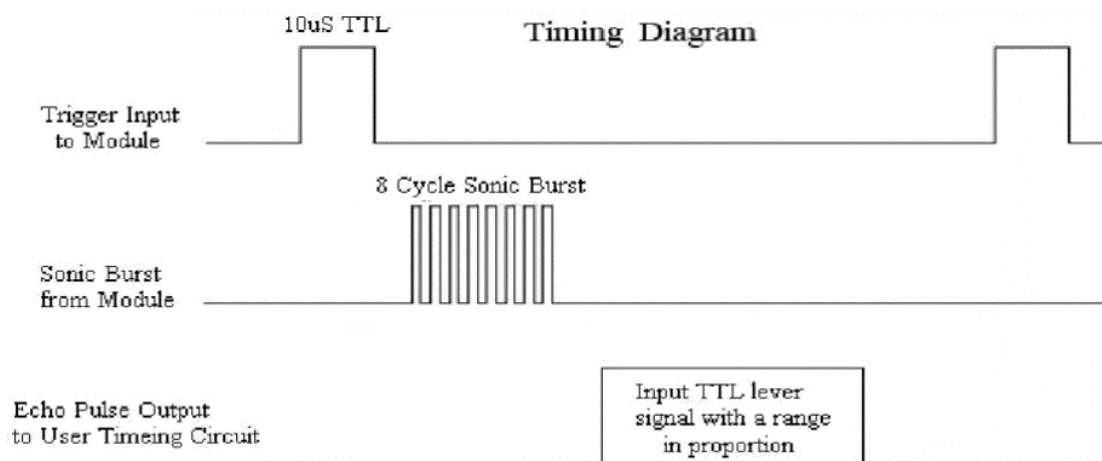
Snímače vzdálenosti jsou používány pro detekci vzdálenosti od překážky, kdy se u hodnot nižších než je hodnota pro vhodné přiblížení k překážce vozítko zpomalí a u hodnot nižších než je bezpečná hranice přiblížení se vozítko, v daném směru, zastaví.

2.6.1. Princip fungování modulu

Pro získání vzdálenosti ze snímače je nutné vyslat na vstup *trigger* alespoň 10 μ s TTL impuls, který aktivuje cyklus měření.

Snímač následně vyšle osm 40 kHz impulsů, podle kterých vypočítá vzdálenost a na výstupní signál *echo* vygeneruje úroveň TTL signálu s šířkou rovnu vzdálenosti, která je rovna 58 násobku času v μ s.

Výrobce je doporučeno cyklus měření neopakovat alespoň 60 ms, čímž se zabrání špatnému měření spojenému se zpožděným příchodem signálů z předchozích měření.



Obrázek 12 – Časový diagram snímače vzdálenosti

3. Ovládání vozítka

Vozítko je vybaveno moduly dálkového ovládání, což jej umožňuje ovládat pomocí dvojice dálkových ovladačů. Je využíváno dvojice modulů dálkového ovládání, jelikož větší RF moduly dálkového ovládání s ovladači byly v cenové kategorii nedostupné.

Dohromady je dostupné k využití 8 tlačítek. 4 tlačítka se využívají na pohyb, což jsou tlačítka *dopředu*, *dozadu*, *doleva* a *doprava*. Zbýlá 4 tlačítka se využívají pro speciální funkce určených v programu.

3.1. Pohyb

Pro pohyb dopředu je nutno stisknout tlačítko *dopředu*. Pro současný pohyb do stran, tj. šikmý pohyb, je poté nutno stisknout tlačítko *doleva* či tlačítko *doprava*. Obdobný postup je i pro jízdu dozadu, kde se namísto tlačítka *dopředu* využívá tlačítko *dozadu*.

3.2. Otáčení na místě

Vozítko je vybaveno funkcí pro otáčení na místě, které je možno provádět díky konstrukci kol. Při samotném stisku tlačítka *doleva* či *doprava* se vozítko začne otáčet pomalou rychlostí daným směrem.

Otáčení na místě je možné jen tehdy, pokud vozítko stojí.

3.3. Speciální funkce

Vozítko podporuje až 4 tlačítka speciálních funkcí. Funkce každého z nich je závislá na aktuálním nahraném programu.

3.3.1. Přepnutí rychlosti

Prvním speciálním tlačítkem je tlačítko pro přepnutí rychlosti. Rychlost se přepíná držením tlačítka se zpožděním půl sekundy. Při držení tlačítka delší dobu se bude rychlost přepínat tam a zpět.

V základu jsou dostupné 2 stavy – pomalu a rychle. Pokud je však vozítko zastaveno snímači vzdálenosti, tj. vozítko je v daném směru od překážky v nebezpečné vzdálenosti, rychlost se přepne do 3. stavu, který značí vypnutí rychlosti.

Pokud se stiskne tlačítko při tomto třetím stavu, rychlost se přepne na rychlost pomalou. Při pokusu o pohyb v daném směru s nebezpečnou vzdáleností od překážky se rychlost opět přepne do třetího stavu „zastaveno“.

3.3.2. Přepnutí světel

Druhým tlačítkem se speciální funkcí je tlačítko pro přepínání světel. Obdobně jako u přepínání rychlosti, i zde se přepínání aktivuje podržením tlačítka. Při delším držení je tak možné se světly blikat.

V momentálním natavení programu nelze přepínat přední a zadní světla zvlášť, možné je pouze přepínání předních i zadních světel současně.

4. Software

4.1. Jazyk symbolických adres

Na programování ATmega32, resp. AVR mikroprocesorů obecně, lze použít jazyky JSA (mnohdy nepřesně označováno Assembler, což je program pro překlad JSA do strojového kódu), C anebo Pascal. Má práce je napsána v jazyku symbolických adres.

JSA je nízkoúrovňový programovací jazyk pro symbolickou reprezentaci strojových instrukcí mikroprocesoru. Každý mikroprocesor má rozdílnou sadu instrukcí, kterou tvoří zejména výrobce daného mikroprocesoru. To znamená, že program většinou není přenositelný mezi různými typy mikroprocesorů. Název každé instrukce většinou vyjadřuje zkratku, většinou o 3 až 4 znacích, pro název celé instrukce popisující její funkci, např. *SBC* pro *Subtract with Carry*.

Vzhled programu psaného v jazyce symbolických adres lze vidět v příloze D obsahující finální program pro robotické vozítko.

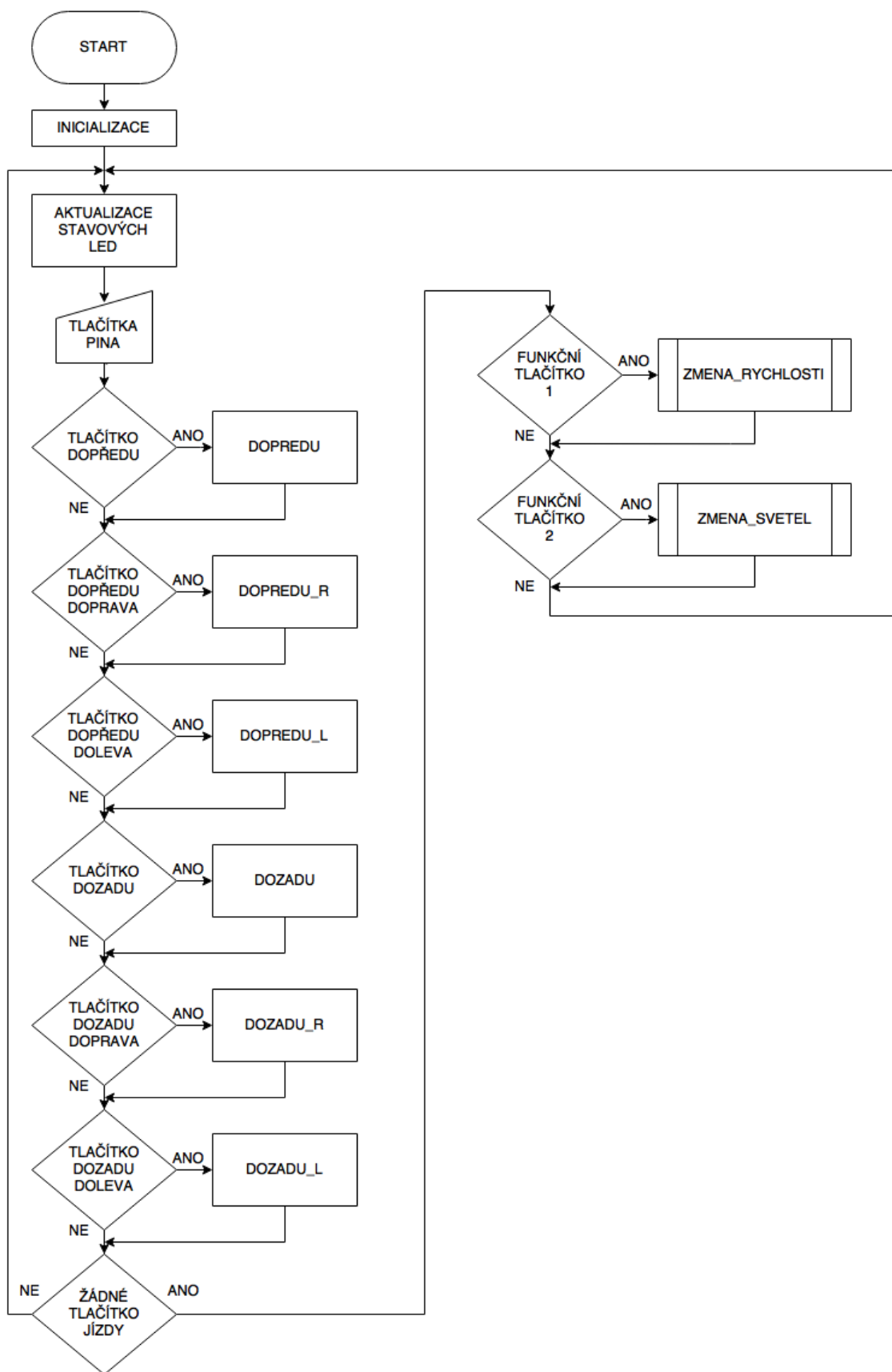
4.2. Nastavení

V úvodu programu je nejprve nutno provést nastavení proměnných, jako např. počáteční stav signalizačních LED, vytvoření proměnných pro rozmístění tlačítek a následné vytvoření maker pro definování nastavení pohybu a rychlostí. Poté jsou nastaveny, které porty, či jejich části, jsou vstupní a které výstupní. Je také nutno vynulovat a nastavit čítače a 2 komparační registry pro správný chod PWM sloužící k ovládání motorů.

4.3. Hlavní cyklus

Hlavní cyklus obsahuje podmínky a funkcionalitu, která se opakuje v nekonečném cyklu. Obsahuje nejpodstatnější logiku programu. Celý hlavní cyklus je vyobrazen na diagramu na obrázku 13.

Hlavní cyklus začíná po prvotní inicializaci programu, tj. nastavení proměnných, registrů, čítačů a komparačních registrů atp.



Obrázek 13 – Hlavní cyklus

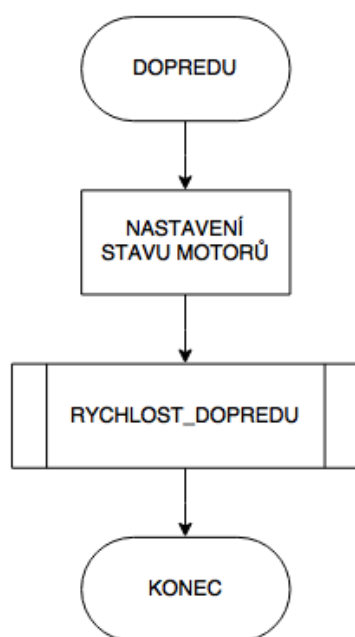
První věcí, která se provede je aktualizace stavových LED a LED pro světla. Jednotlivé změny stavů se v programu mohou provádět na jakémkoli místě a proto je pro zjednodušení a zefektivnění nastavování přesunuto na začátek cyklu.

Dále se načtou data z modulu dálkového ovládání umístěného na portu A, které slouží pro manuální ovládání vozítka. Získaná data je nutné porovnat se všemi tlačítky. Porovnává se kód získaný z portu A s maskou pro dané tlačítko. Při shodě se vykoná daný blok kódu a při neshodě se program přesune na testování dalšího tlačítka. K testování tlačítek se speciální funkcí se program dostane pouze tehdy, pokud není stisknuto žádné z tlačítek pro pohyb, tzn. tlačítka *dopředu*, *dozadu*, *doleva* a *doprava*.

Testování tlačítek pro speciální funkce probíhá stejně jako testování tlačítek pro pohyb. Po dokončení porovnávání se program vrátí na začátek cyklu a vše se opakuje.

4.4. Řízení jízdy

Řízení jízdy v programu probíhá voláním maker, které dále definují nastavení registrů určených pro uchovávání stavu motorů tak, aby na výstupu z řídicí jednotky vystupovaly správné signály PWM a aby se kola správně točila. Na konci každého makra je volání podprogramu pro nastavení správné rychlosti, kde se rozlišuje mezi směry dopředu a dozadu.



Obrázek 14 – Pohyb dopředu

Jelikož se Signály PWM generují neustále, v programu stačí nastavit 2 dvojice pinů, které, jak již bylo zmíněno, určují propustnost signálů PWM a také směr točení motorů. Kola po stranách se ovládají jako celek, a tedy není možné roztočit jen pravý přední motor a pravý zadní neroztočit, protože motory na každé straně jsou připojeny do stejných vývodů z budiče motoru.

Kromě řízení směru točení motorů na každé straně robotického vozítka je nutné korigovat rychlost jízdy. Robotické vozítko se může otáčet na místě, pro což je nadefinovaná pomalá, zhruba poloviční, rychlost.

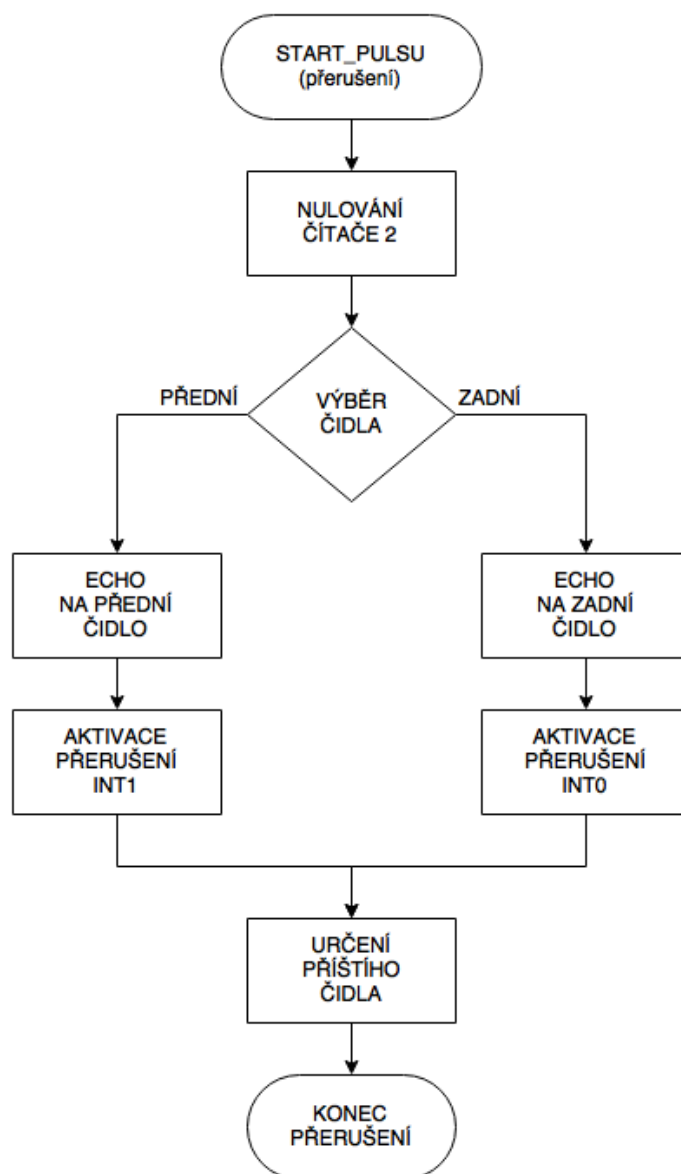
4.5. Detekce vzdálenosti

Vzdálenost je získávána automaticky, tj. mimo chod hlavního cyklu programu. Je využíváno přerušení čítače 2. Čítač 2 čítá, resp. inkrementuje hodnotu, a při shodě hodnoty s hodnotou uloženou v komparačním registru je vyvoláno přerušení.

Čítač a hodnota v komparačním registru je nastavena tak, aby se vzdálenost detekovala 16x za sekundu, resp. 8x za sekundu pro každé z čidel. Detekce vzdálenosti probíhá na přeskáčku mezi detekováním hodnoty předního a zadního snímače vzdálenosti. Přepínání snímačů funguje tak, že se inkrementuje hodnota registru *R2* a pokud je hodnota v registru sudá, bude se detekovat hodnota na předním snímači, pokud je hodnota lichá, bude se detekovat vzdálenost na zadním snímači.

V tomto přerušení se nejprve vynuluje čítač 2. Následně se určí, pro který snímač se bude podprogram vykonávat. Pokud je vybrán přední snímač, vyšle se krátký impuls na signál *echo* předního snímače a následně se aktivuje přerušení *INT1*, které odposlouchává *trig* signál ze snímače a při změně náběžné či sestupné hrany se vyvolá přerušení. Pro zadní snímač je proces obdobný, jen se vyšle krátký impuls na *echo* signál zadního snímače a následně se aktivuje přerušení *INT0*.

Na závěr se provede již zmíněná inkrementace registru *R2*.



Obrázek 15 – Požadavek pro měření vzdálenosti

4.5.1. Princip fungování přerušení snímače

Po aktivaci přerušení *INT0* či *INT1* a vyslání krátkého impulsu na echo signál zvoleného snímače se vyčkává, až na signálu *trig* nastane změna hrany. Jak již bylo vysvětleno, signál *trig* generuje TTL signál, kde se časem vysílání logické „1“ určuje vzdálenost od překážky.

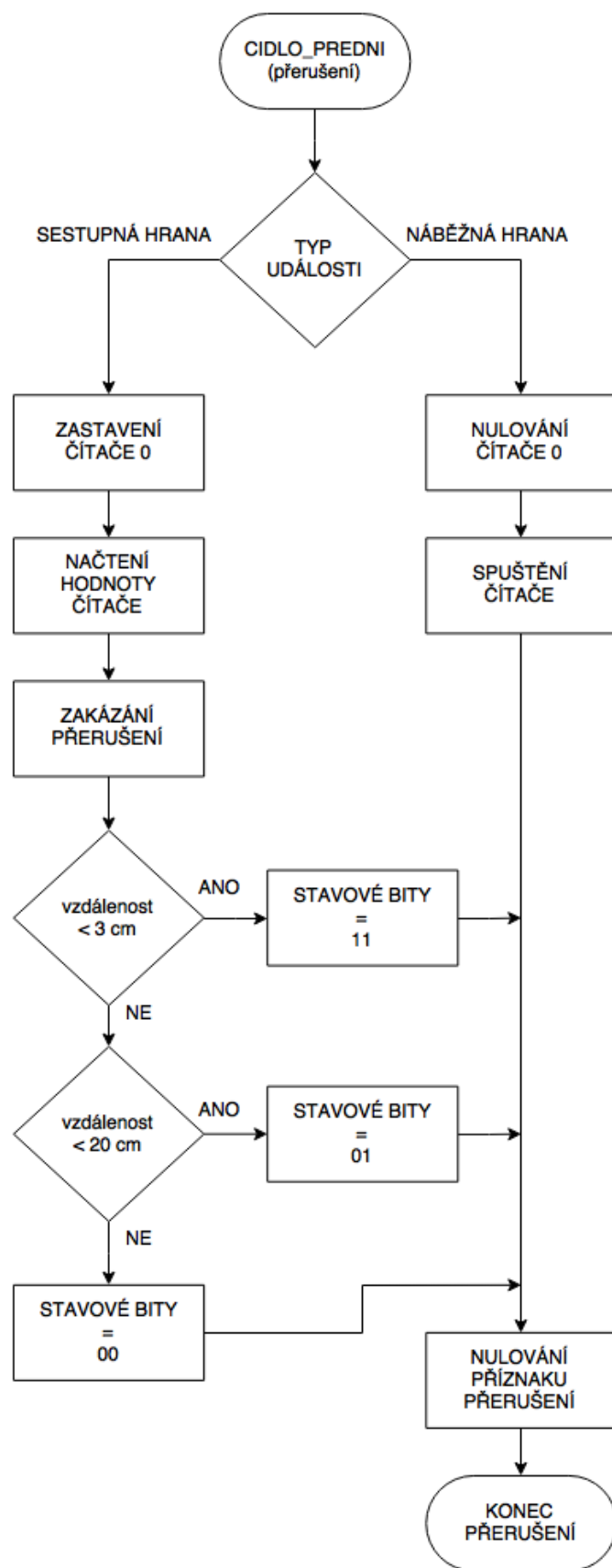
Změna hrany určuje typ události. Při náběžné hraně se vynuluje čítač 0 a čítač se spustí. Tím se začne počítat vzdálenost jako počet vzorků s logickou hodnotou „1“ na

signálu snímače vzdálenosti *trig* s periodou podle „vzorkovací frekvence“ určené nastavením čítače.

Při sestupné hraně je čítač 0 zastaven a hodnota čítače se uloží do registru a zakážou se přerušení. Hodnota v registru reprezentuje vzdálenost od překážky.

Nyní je nutno zjistit, zda je vzdálenost menší než 3 cm, v tomto případě se stavové bity nastaví na hodnotu 11, menší než 20 cm, v tomto případě se stavové bity nastaví na hodnotu 01, nebo je větší než 20 cm, v tomto případě se stavové bity nastaví na hodnotu 00.

Nakonec se vynulují příznaky přerušení a podprogram přerušení se ukončí.



Obrázek 16 – Měření vzdálenosti snímačem vzdálenosti

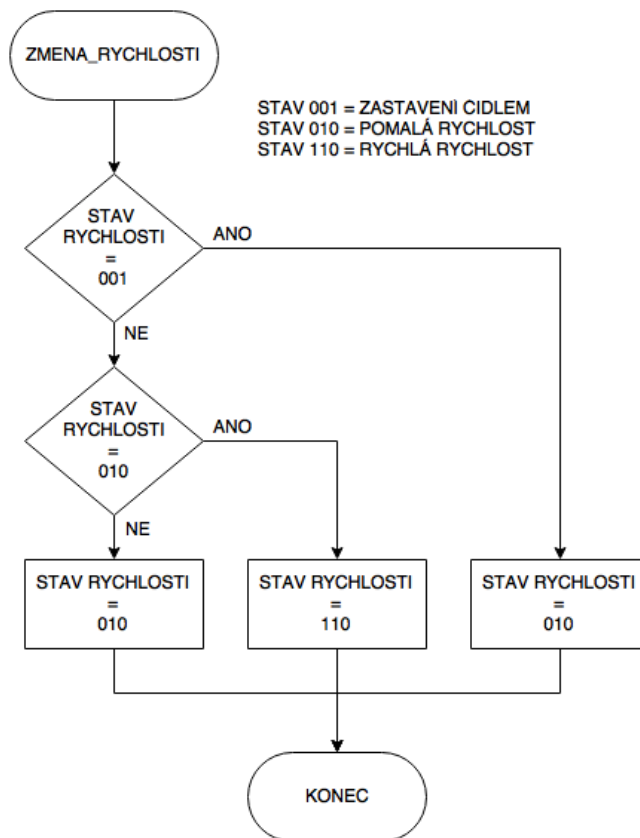
4.6. Rychlost pohybu

Rychlost točení motorů, resp. pohybu vozítka, lze nastavit úpravou hodnoty 8bitových komparačních registrů, kde hodnota 0 znamená stav zastaveno a hodnota 255 stav rychlé jízdy. Kvůli vlastnostem PWM a spínáním motorů není polovina, tedy hodnota 128, přesná polovina výkonu motorů, ale lze rychlost označit za pomalou jízdu.

Rychlost pohybu je určována zvlášť pro jízdu vpřed a jízdu vzad, aby se mohlo vozítko pohybovat pomalu směrem k překážce a naopak rychle směrem od překážky.

4.6.1. Princip změny rychlosti

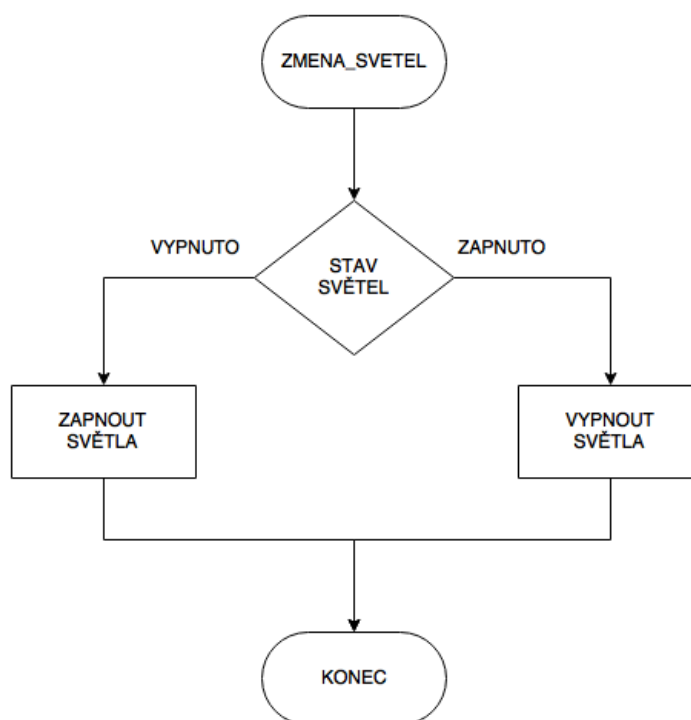
Podprogram pro změnu rychlosti funguje tak, že se porovnává stav rychlosti uložené ve stavovém registru. Při hodnotě 001, která se nastaví pouze čidlem a zastaví autíčko, se hodnota změní na 010, což reprezentuje pomalou rychlost. Při hodnotě 010 se hodnota změní na 110, což reprezentuje rychlou rychlost. Při hodnotě 110 se hodnota změní na hodnotu 010 reprezentující pomalou rychlost.



Obrázek 17 – Změna rychlosti

4.7. Rozsvěcení světel

Kromě cyklu program obsahuje např. algoritmus pro přepínání stavů rychlosti a rozsvěcení světel. To probíhá tak, že se znegují bity z původní „1“ na „0“ či z původní „0“ na „1“. Samotná změna v registru rozsvěcení, či zhasnutí, světel neprovede, to provede až počátek hlavního cyklu, kde se vypíše aktuální hodnoty z registru na port.



Obrázek 18 – Změna světel

5. Závěr

Podářilo se mi vytvořit funkční robotické vozítko, připravené na případnou budoucí modifikaci, schopné jízdy a ovládání vnitřních stavů (např. změna rychlosti, rozsvěcení světel atp.) pomocí dálkového ovladače. Robotické vozítko disponuje ultrazvukovými senzory, umístěnými vpředu a vzadu, využitelnými např. pro určení vzdálenosti od překážky, což může zajistit např. zastavení vozidla před překážkou a tedy zamezení kolize, resp. nabourání.

Kompletní realizace robotického vozítka kladně rozšířila mé znalosti v oblasti programování mikroprocesorů. Také jsem se naučil a zdokonalil v oblasti návrhu, sestavování a pájení univerzálních desek plošných spojů.

Vozítko má mnoho možností pro budoucí vylepšení, např. přidání módu autonomního řízení, přidání web kamery a následné umožnění ovládání vozítka pomocí poskytovaného obrazu, přidání prvků či funkcí známých z reálných dopravních prostředků, jako je např. klakson či signalizace změny směru atp.

6. Použité zdroje

- [1] ATmega32A [online]. [cit. 2016-03-10]. Dostupné z:
<http://www.atmel.com/devices/atmega32a.aspx>
- [2] Atmel-8155-8-bit-Microcontroller-AVR-ATmega32A_Datasheet [online]. [cit. 2016-03-11]. Dostupné z: http://www.atmel.com/Images/Atmel-8155-8-bit-Microcontroller-AVR-ATmega32A_Datasheet.pdf
- [3] Atmel [online]. [cit. 2016-03-10]. Dostupné z: <https://cs.wikipedia.org/wiki/Atmel>
- [4] Atmel [online]. [cit. 2016-03-12]. Dostupné z: <https://en.wikipedia.org/wiki/Atmel>
- [5] 69545_DS [online]. [cit. 2016-03-15]. Dostupné z:
http://pdf.datasheetcatalog.com/datasheets/208/69545_DS.pdf
- [6] HC-SR04 [online]. [cit. 2016-03-15]. Dostupné z:
<http://www.micropik.com/PDF/HCSR04.pdf>
- [7] Kolový podvozek 01 - PIC AXE + RC | MERKURTOYS s. r. o. [online]. [cit. 2016-03-16]. Dostupné z: <http://www.merkurtoys.cz/vyroby/kolovy-podvozek-01>
- [8] Pulzně šířková modulace [online]. [cit. 2016-03-17]. Dostupné z:
https://cs.wikipedia.org/wiki/Pulzn%C4%9B_%C5%A1%C3%AD%C5%99kov%C3%A1_modulace
- [9] Plošný spoj [online]. [cit. 2016-03-19]. Dostupné z:
https://cs.wikipedia.org/wiki/Plo%C5%A1n%C3%BD_spoj
- [10] Vývojový kit EvB 4.3 a možnosti programování AVR ATMEGA [online]. [cit. 2016-03-20]. Dostupné z: <http://www.dps-az.cz/vyvoj/id:6185/vyvojovy-kit-evb-4-3-a-moznosti-programovani-avr-atmega>
- [11] CD54HC08, CD74HC08, CD54HCT08, CD74HCT08 (Rev. C) [online]. [cit. 2016-03-20]. Dostupné z: <http://www.ti.com/lit/ds/symlink/cd74hc08.pdf>
- [12] Jazyk symbolických adres [online]. [cit. 2016-03-20]. Dostupné z:
https://cs.wikipedia.org/wiki/Jazyk_symbolick%C3%BDch_adres
- [13] Uživatelská příručka [online]. [cit. 2016-04-09]. Dostupné z: <http://and-tech.pl/wp-content/uploads/downloads/2013/10/Instrukcja-EvB5.1-v1-cze.pdf>

- [14] Set EVB 5.1 [online]. [cit. 2016-04-09]. Dostupné z: <http://and-tech.pl/zestaw-evb-5-1/>
- [15] Výukové materiály používané při výuce předmětu „Mikroprocesorová technika“ na SPŠ Česká Lípa [cit. 2016-04-10]
- [16] BAŠTA, Josef. ATMEGA32abc.pdf [cit. 2016-04-10]

7. Seznam použitého software

[17] Microsoft Corporation: Microsoft Office Word 2016

[18] Atmel Corporation: AVR Studio 4.19

[19] JGraph Ltd: draw.io v 5.4.1.7

[20] CadSoft: EAGLE 7.5.0

[21] The GIMP Team: Gimp 2.8.14

8. Seznam obrázků a tabulek

Obrázky

Obrázek 1 – Blokové schéma	9
Obrázek 2 – Sestavené robotické vozítko.....	10
Obrázek 3 – Technický výkres motorů.....	11
Obrázek 4 – DPS řídicí jednotky	13
Obrázek 5 – JTAG rozhraní.....	13
Obrázek 6 – Vnitřní zapojení AND hradel	14
Obrázek 7 – DPS budiče motorů	15
Obrázek 8 – Schéma budiče motorů	15
Obrázek 9 – ATmega32 PDIP pouzdro	16
Obrázek 10 – Přijímač a vysílač modulu dálkové ovládání	19
Obrázek 11 – Ultrazvukový snímač vzdálenosti HC-SR04	20
Obrázek 12 – Časový diagram snímače vzdálenosti	21
Obrázek 13 – Hlavní cyklus	25
Obrázek 14 – Pohyb dopředu	26
Obrázek 15 – Požadavek pro měření vzdálenosti.....	28
Obrázek 16 – Měření vzdálenosti snímačem vzdálenosti	30
Obrázek 17 – Změna rychlosti.....	31
Obrázek 18 – Změna světla	32
Obrázek 19 – Testování kolového podvozku	43
Obrázek 20 – Testování snímače vzdálenosti.....	43
Obrázek 21 – Počátek realizace řídicí jednotky	44
Obrázek 22 – Dokončení řídicí jednotky.....	44
Obrázek 23 – Počátek realizace budičů motorů	44

Obrázek 24 – Dokončení realizace budičů motorů	45
Obrázek 25 – Testování ovládání motorů budiči motorů	45
Obrázek 26 – Sestavení modelu robotického vozítka	46
Obrázek 27 – Pohled zezadu na model robotického vozítka.....	46
Obrázek 28 – Dokončení propojení elektroniky na modelu robotického vozítka.	46

Tabulky

Tabulka 1 – Parametry motorů	12
Tabulka 2 – Parametry ATmega32.....	17

9. Seznam příloh

A. Schéma zapojení řídicí jednotky

B. Schéma zapojení budičů motorů

C. Fotografie z průběhu vývoje

D. Program pro řídicí mikroprocesor

E. Datový disk

- Práce ve formátu Microsoft Word
- Práce ve formátu PDF
- Zdrojový kód ve formátu PDF
- Složka program – program pro robotické vozítko
- Složka dokumentace – dokumentace k použitým součástkám
- Složka fotodokumentace – fotodokumentace vývoje
- Složka schémata – schémata zapojení a blokové schéma ve formátu XML a PNG
- Složka diagramy – diagramy programu ve formátu XML a PNG

10. Licenční ujednání

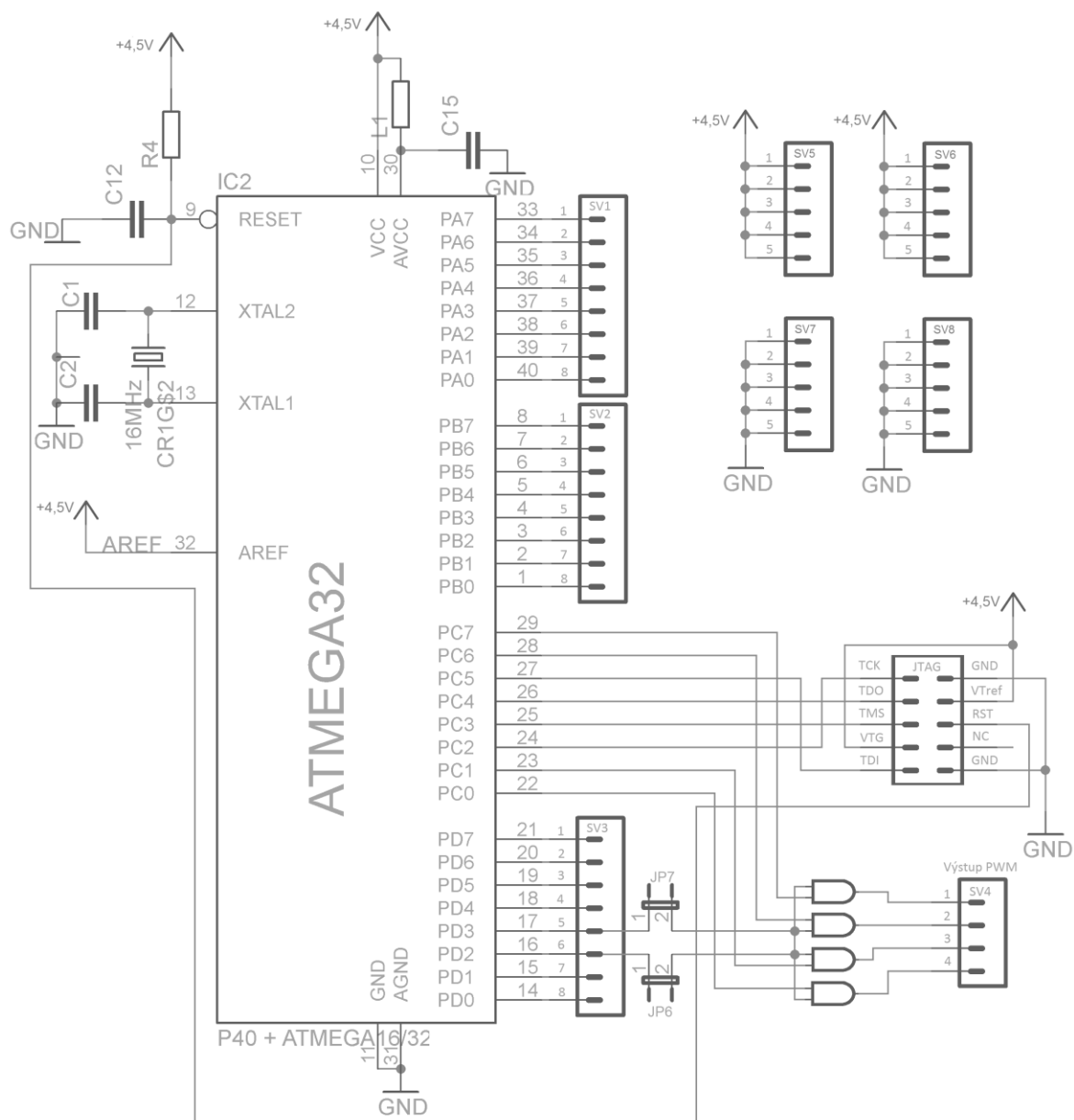
Ve smyslu zákona č. 121/2000 Sb., O právu autorském, o právech souvisejících s právem autorským, ve znění pozdějších předpisů (dále jen autorský zákon) jsou práva k maturitním nebo ročníkovým pracím následující:

Zadavatel má výhradní práva k využití práce, a to včetně komerčních účelů.

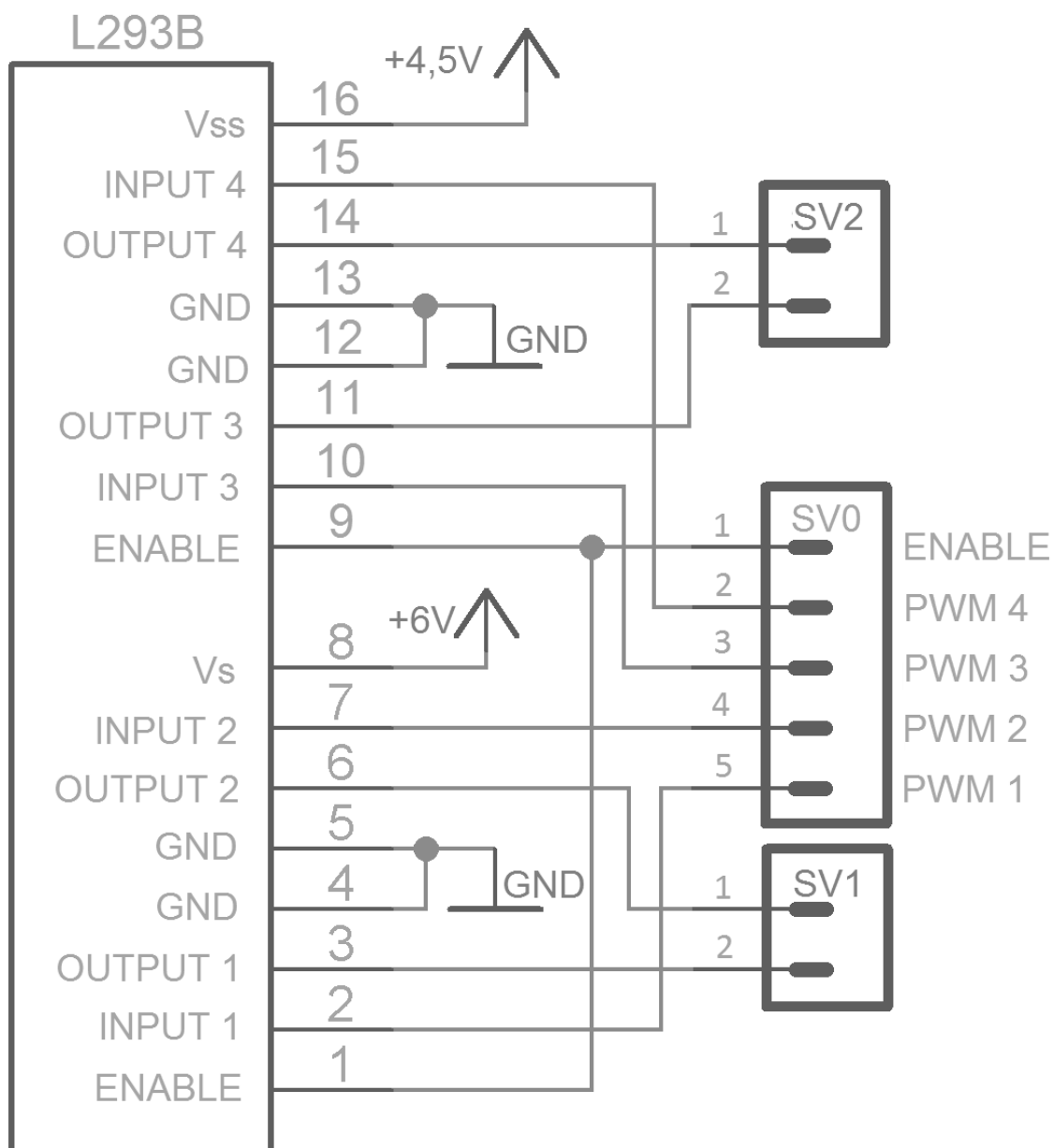
Autor práce bez svolení zadavatele nesmí využít práci ke komerčním účelům.

Škola má právo využít práci k nekomerčním a výukovým účelům i bez svolení zadavatele a autora práce.

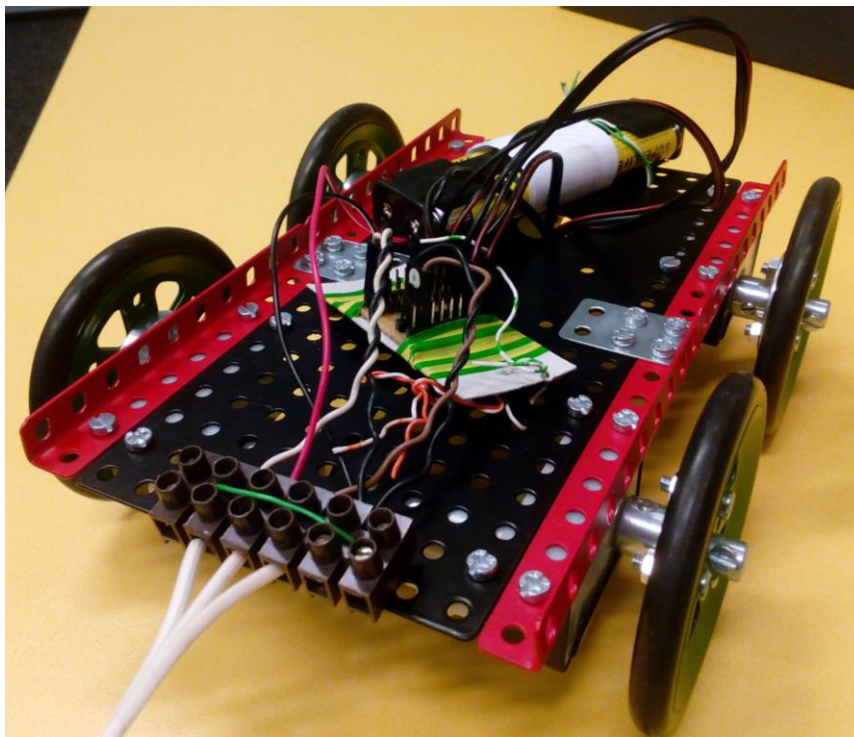
Příloha A – Schéma zapojení řídicí jednotky



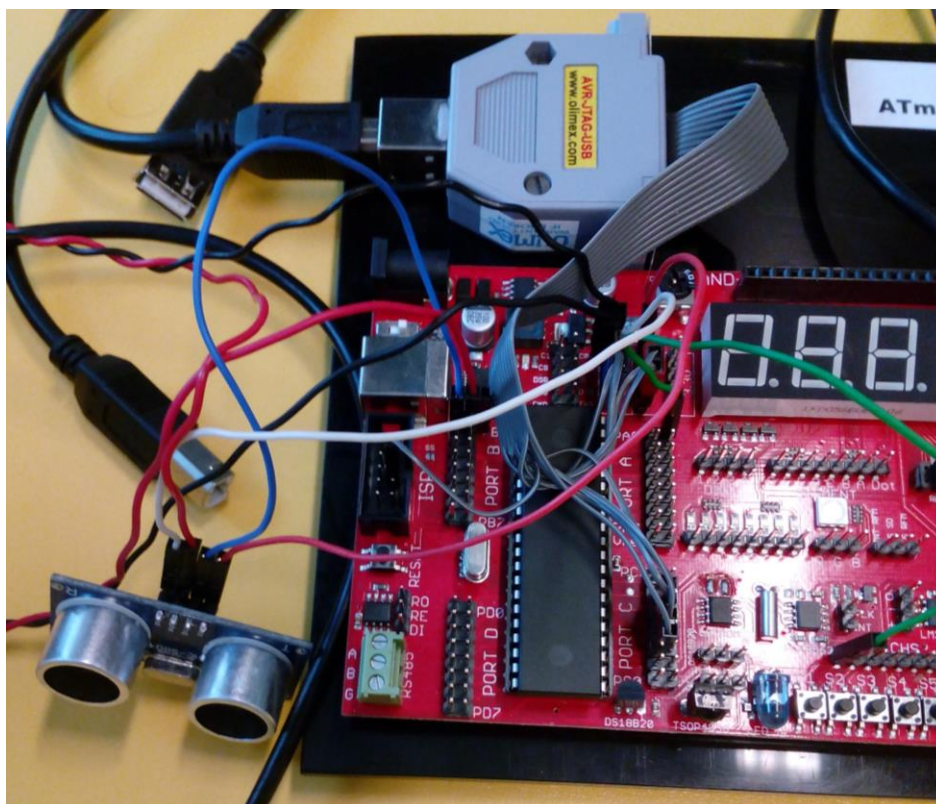
Příloha B – Schéma zapojení budičů motorů



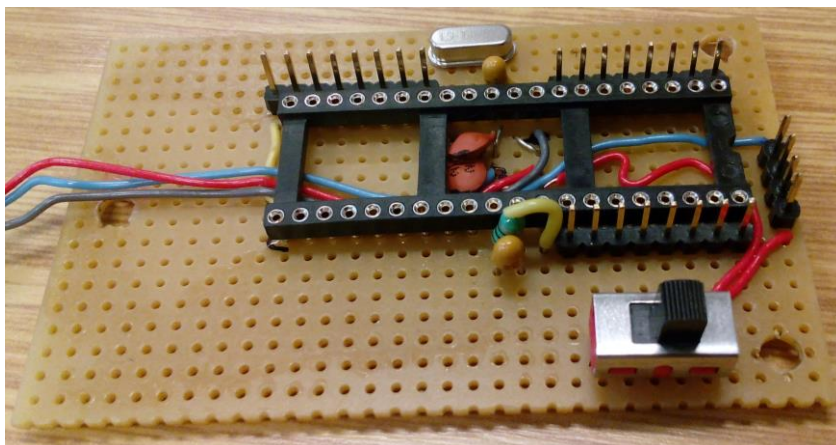
Příloha C – Fotografie z průběhu vývoje



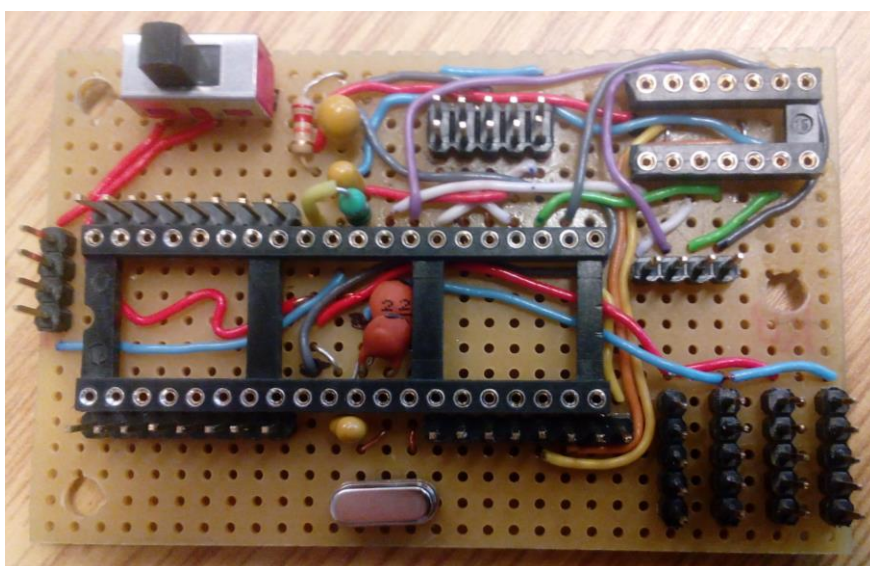
Obrázek 19 – Testování kolového podvozku



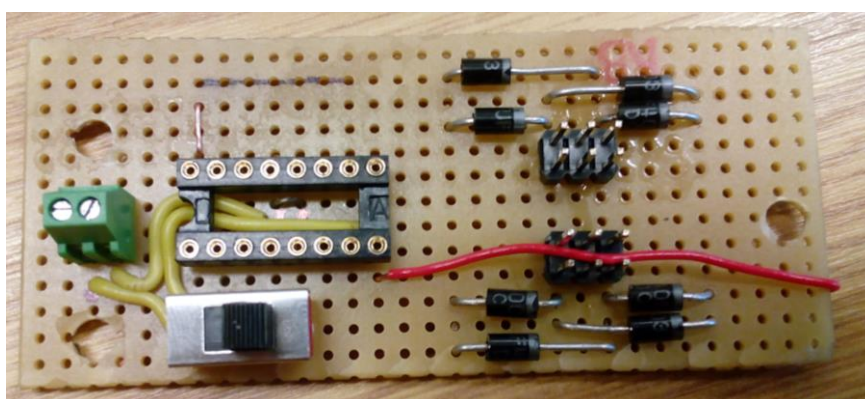
Obrázek 20 – Testování snímače vzdálenosti



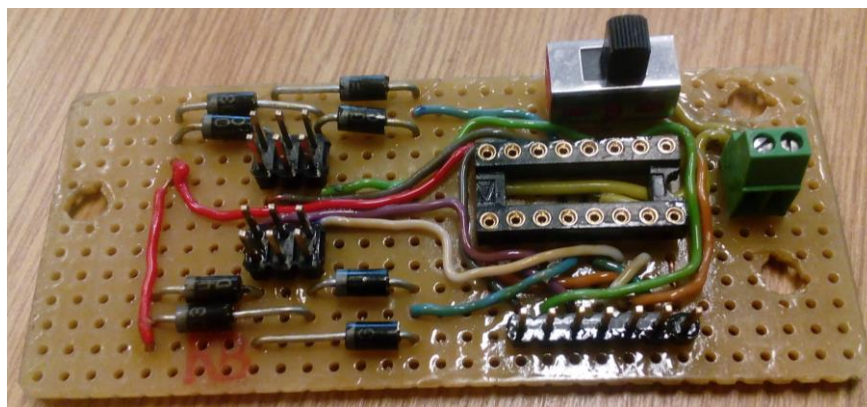
Obrázek 21 – Počátek realizace řídicí jednotky



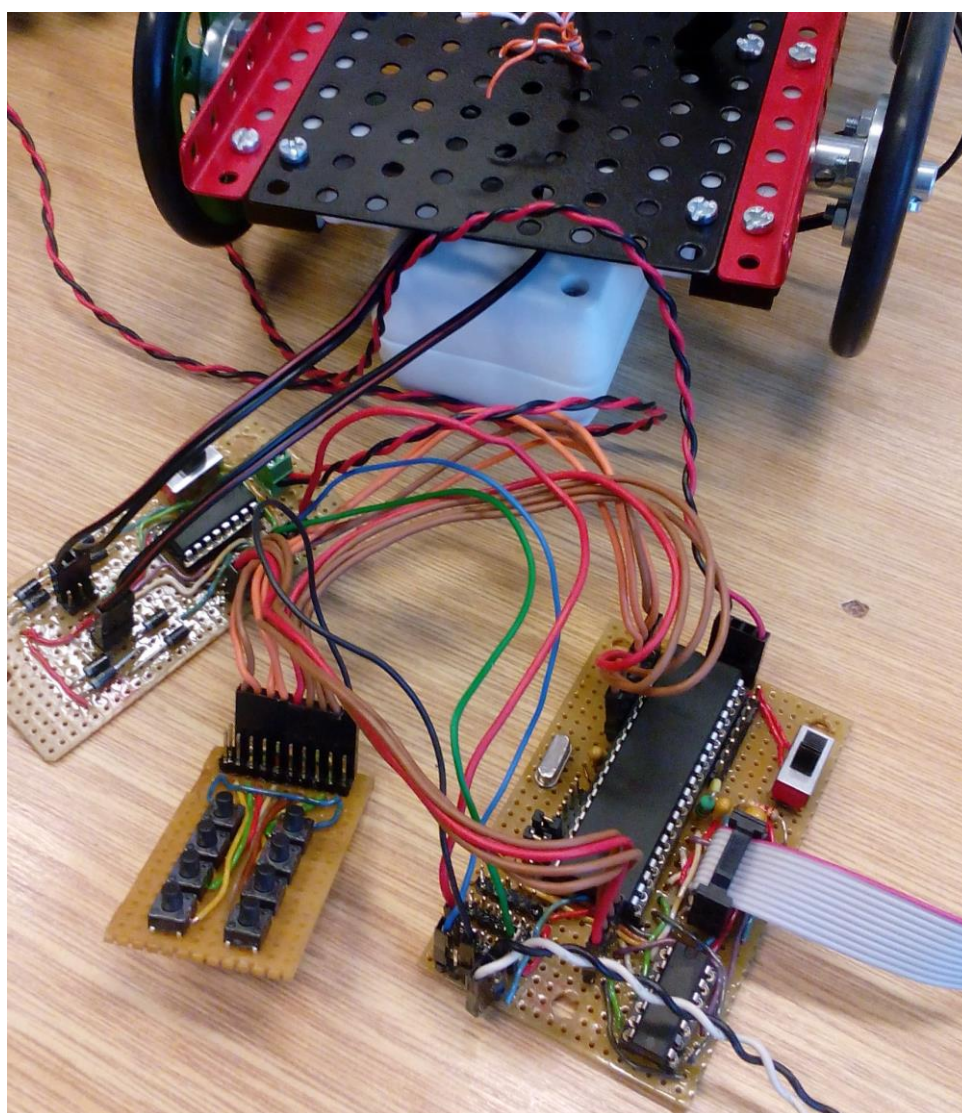
Obrázek 22 – Dokončení řídicí jednotky



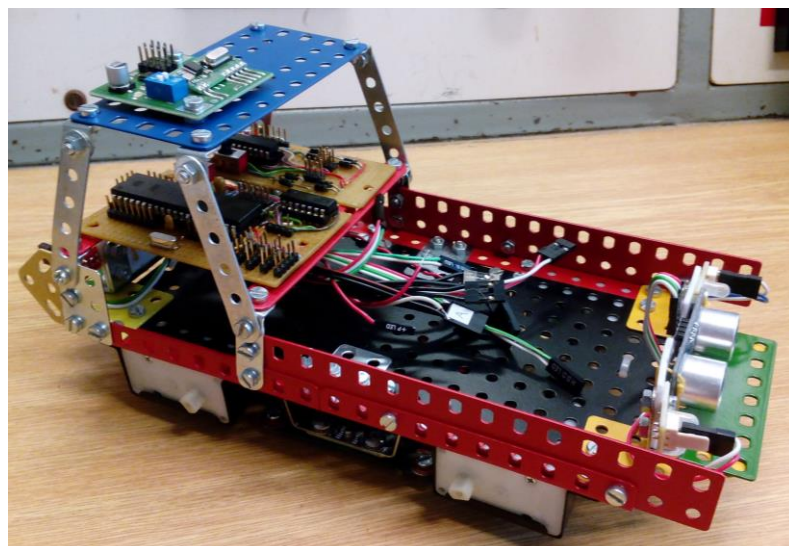
Obrázek 23 – Počátek realizace budičů motorů



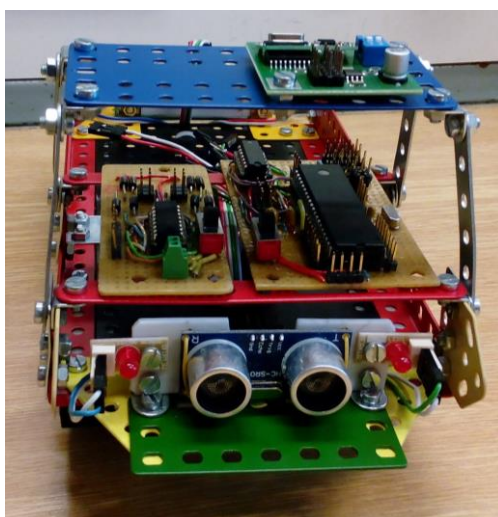
Obrázek 24 – Dokončení realizace budičů motorů



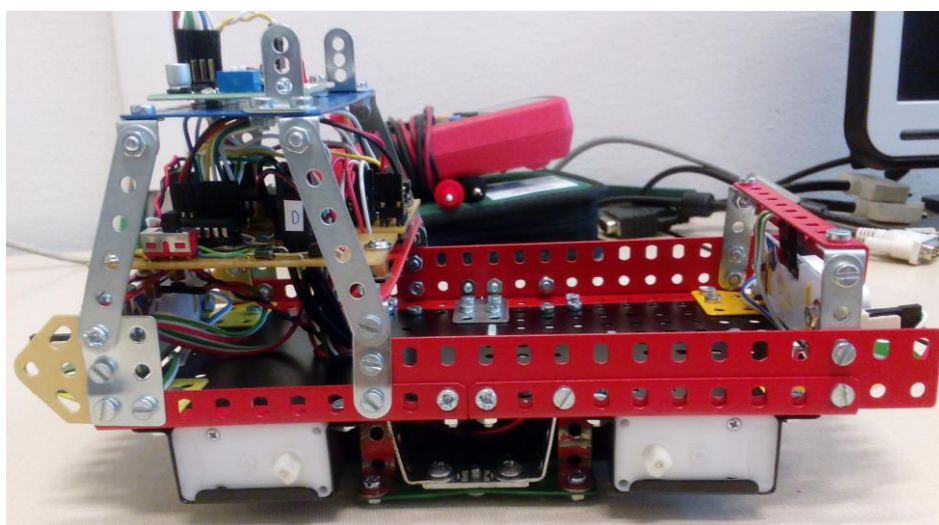
Obrázek 25 – Testování ovládání motorů budiči motorů



Obrázek 26 – Sestavení modelu robotického vozítka



Obrázek 27 – Pohled zezadu na model robotického vozítka



Obrázek 28 – Dokončení propojení elektroniky na modelu robotického vozítka

Příloha D – Program pro řídící mikroprocesor

Příloha E – Datový disk