



Středoškolská technika 2016

**Setkání a prezentace prací středoškolských studentů na
ČVUT**

Robotické vozítko

Jan Bittner

Střední průmyslová škola, Česká Lípa, Havlíčkova 426, p.o.

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.

Klíčová slova: Robot; vozítko; mikroprocesor; Atmel; Merkur; RC-controller.

Annotation

This work deals with constructing a robotic rover from the toy construction kit Merkur, creating its own control unit and programming a microprocessor. 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.

Key words: Robot; rover; microprocessor; Atmel; Merkur; RC-controller.

Obsah

Úvod.....	4
1. Hardware	5
1.1. Model robotického vozítka	5
1.2. Motory	5
1.3. Desky plošných spojů	6
1.3.1. Řídící jednotka.....	7
1.3.2. Budič motorů	9
1.4. ATmega32	10
1.5. Dálkové ovládání	12
1.6. Snímač vzdálenosti	12
2. Software.....	14
2.1. Jazyk symbolických adres	14
Nastavení	14
2.2. Obecný algoritmus	15
2.3. Řízení jízdy	16
2.4. Rozsvěcení světel.....	16
Závěr	18
Použitá literatura	19
Seznam obrázků a tabulek	20
Seznam zkratek	21
Seznam příloh	22

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 chůtíčem 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.

1. Hardware

Robotické vozítko se skládá z několika samostatných částí. První částí je samotný model robotického vozítka, sestaven ze stavebnice Merkur podle vlastního návrhu.

Pod modelem 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 pro vozítko.

Hlavní částí, 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 přijímání dálkového ovládání.

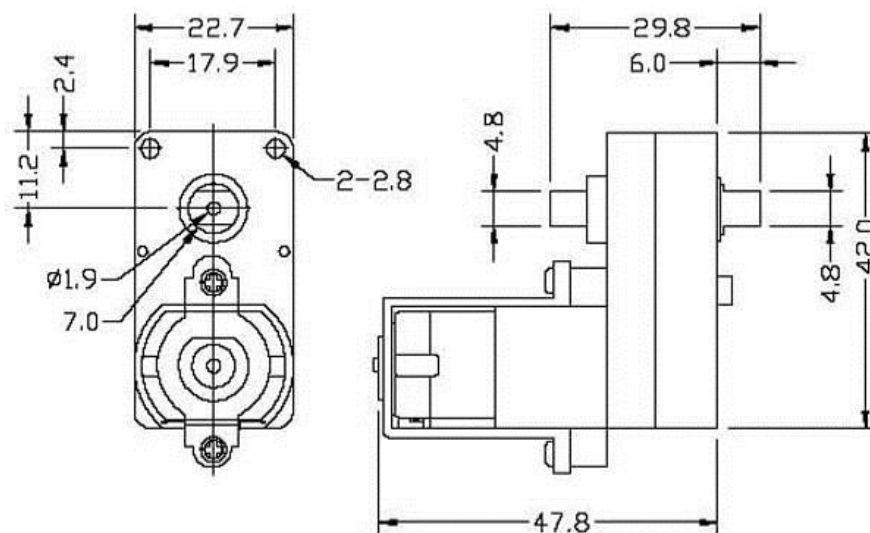
Předeek a zadek modelu obsahuje ultrazvukové snímače vzdálenosti spolu s dvojicemi LED, bílými pro přední světla a červenými pro zadní světla.

1.1. Model robotického vozítka

Model robotického vozítka je sestaven ze stavebnice Merkur tak, aby splňoval specifické požadavky, jako je např. prostor pro desky plošných spojů či nákladový prostor. Obsahuje také ochranné prvky, které slouží pro ochranu jednotlivých součástí při případném nárazu, kolizi či překlopení vozítka.

1.2. Motory

Robotické vozítko využívá motory s převodovkou BO8 ze stavebnice Merkur. Na tyto motory jsou upevněna kola, která však nelze natáčet a pro korigování směru pohybu je nutné softwarové řešení. Vozítko tedy funguje na podobném principu jako tank.



Obrázek 1 - Technický výkres motorů

Motory přijímají informaci o rychlosti otáčení z PWM, které je přiváděno z výstupu budiče 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 střídu 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

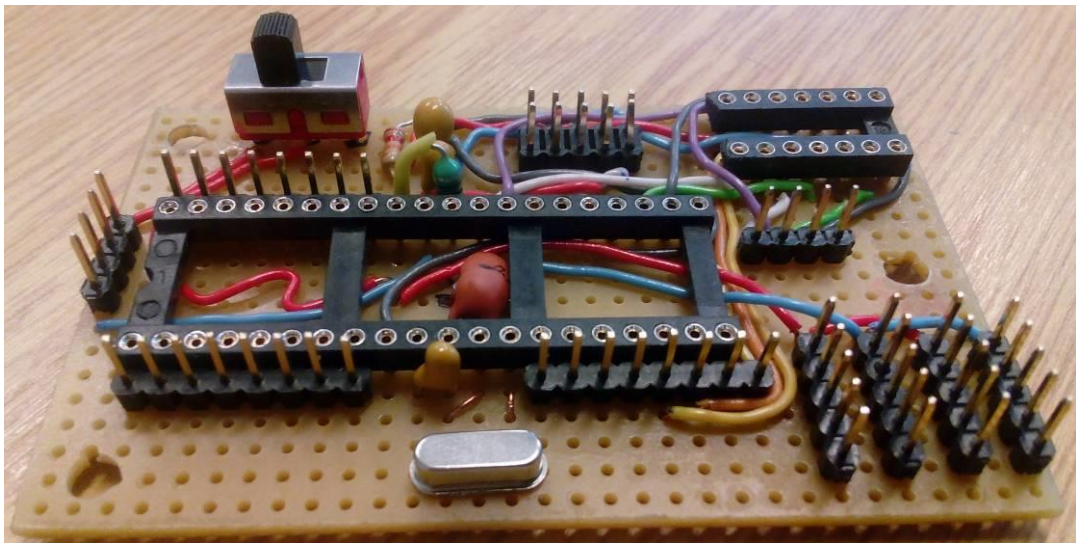
1.3. Desky plošných spojů

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 využíváme

univerzálních DPS, jednotlivé kontakty jsou spojeny vodiči, na povrchu obalenými bužírkou.

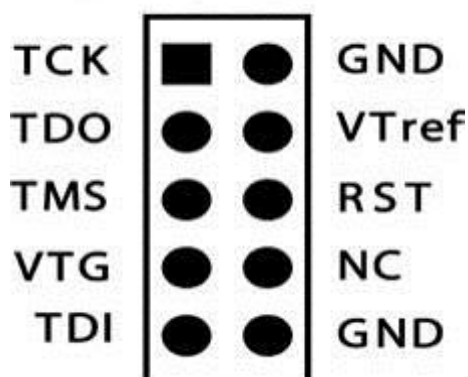
1.3.1. Řídící jednotka

Řídící jednotka zpracovává vstupně-výstupní komunikaci se všemi součástmi robotického vozítka. Obsahuje 40pinovou patici pro připojení mikroprocesoru ATmega32 v PDIP pouzdře, který vykonává a zpracovává všechny akce.



Obrázek 2 - DPS řídicí jednotky

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



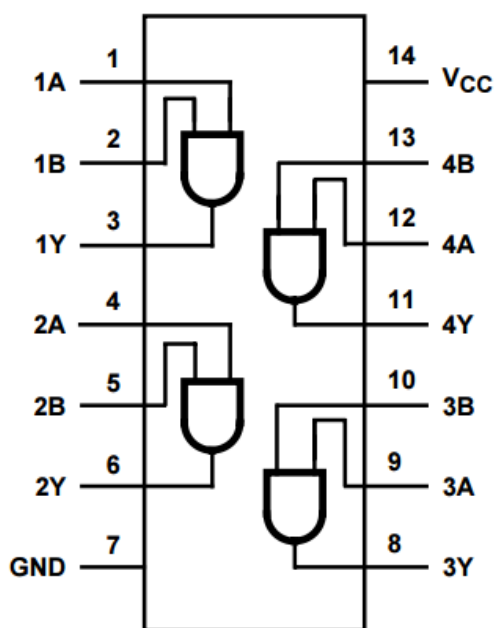
Obrázek 3 - JTAG rozhraní

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.

Port C je napevno propojen s JTAG rozhraním a zbylé 4 piny jsou propojeny do AND hradel, konkrétně součástka CD74HCT08E, v 14pinové patici, další součást obvodu, která obsahuje 4 AND hradla.

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

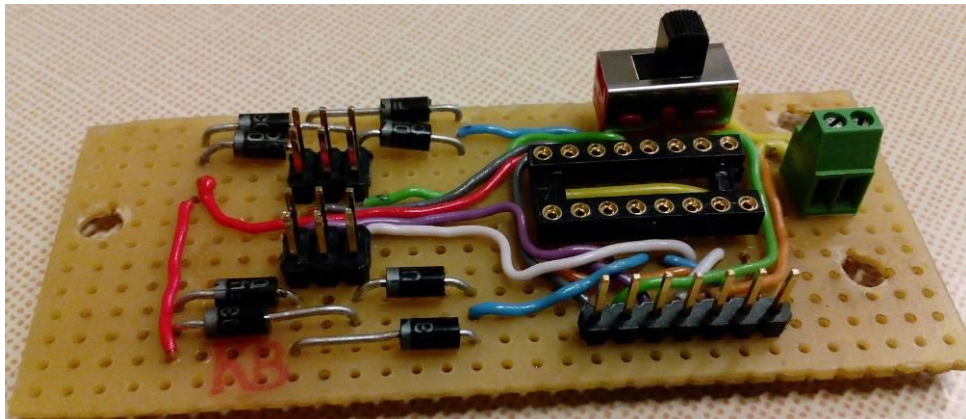


Obrázek 4 – Vnitřní zapojení AND hradel

V neposlední řadě DPS obsahuje 2 pětice 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.

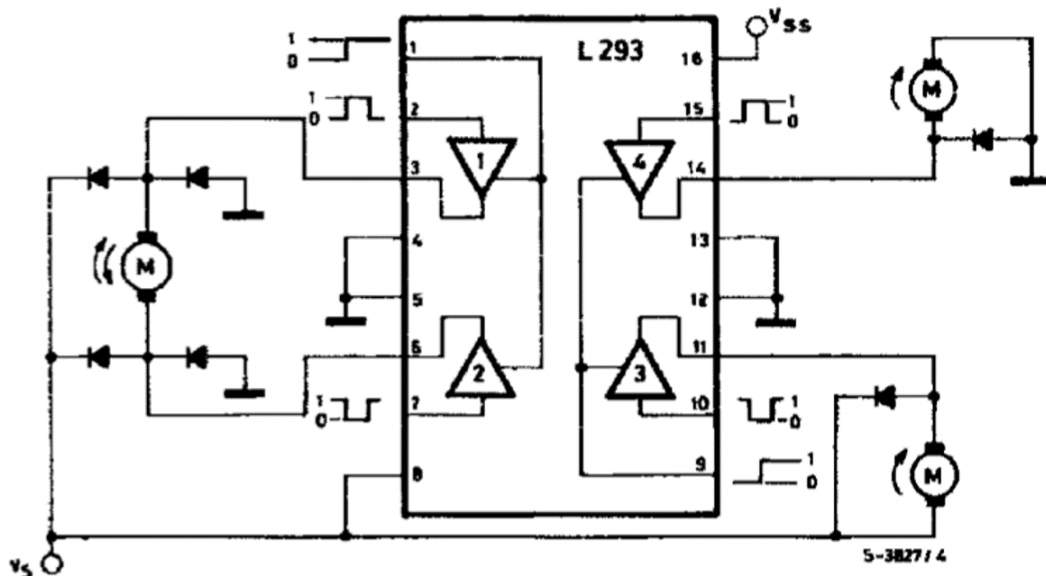
1.3.2. Budič motorů

Druhá DPS obsahuje budič motorů L293B. Obsahuje, podobně jako hlavní DPS, vypínač pro ovládání přívodu napájení a to 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 5 - DPS budiče motoru

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.

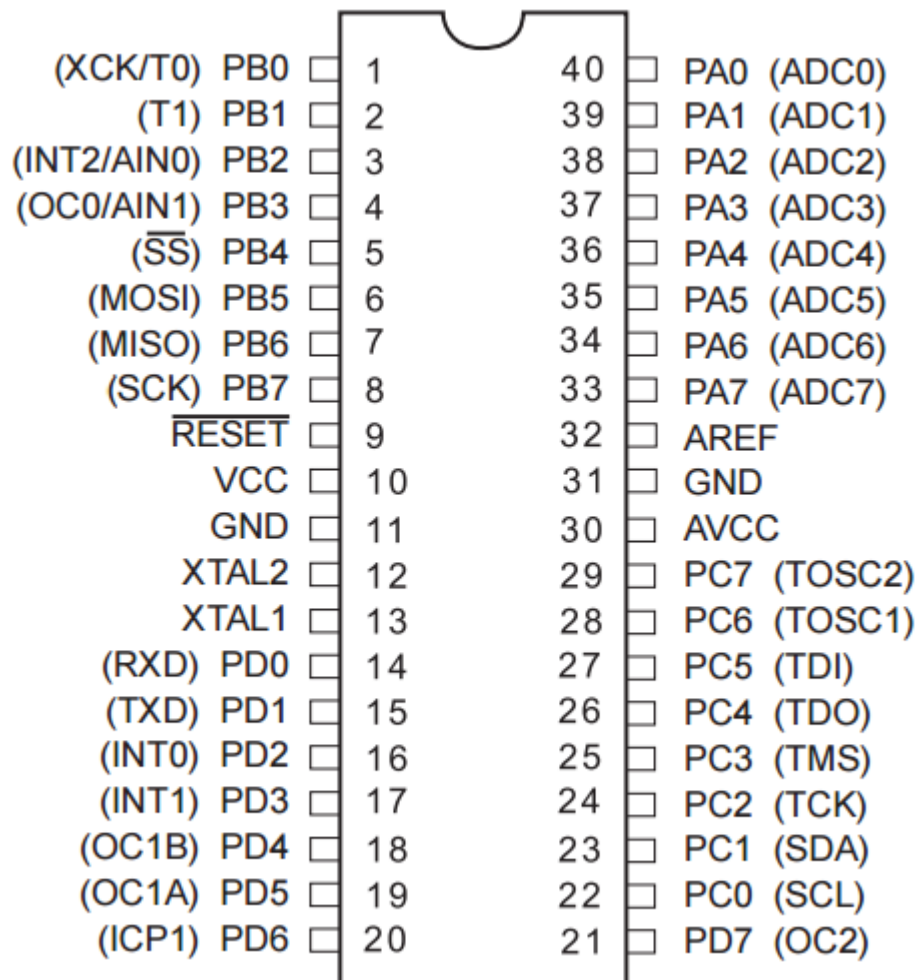


Obrázek 6 - Schéma budiče motorů

Jako výstup této DPS jsou jenom dvě trojice, pro každou stranu, které dále slouží pro připojení motorů, které se připojí vždy na jednu trojici.

1.4. ATmega32

Robotické vozítko je řízeno řídicí jednotkou, která obsahuje mikroprocesor ATmega32, který vyrábí firma Atmel. Tento mikroprocesor se využívá pro řízení všech hlavních částí autíčka. Poskytuje čtyři 8bitové I/O porty, které jsou využívány pro získávání a poskytování dat, např. pro ovládání světel či signalizačních LED nebo získávání dat z dálkového ovládání nebo snímačů vzdálenosti. Disponuje také generátory PWM, které se využívají pro ovládání motorů.



Obrázek 7 - ATmega32 PDIP pouzdro

Mikroprocesor se dá opakovaně přepisovat, což se hodí zejména pro testovací a vyvíjecí potřeby; 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. 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.

1.5. 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 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.

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, zatímco za stejnou cenu lze pořídit IR vysílač a přijímač s 20 tlačítky.

1.6. Snímač vzdálenosti

Robotické vozítko obsahuje ultrazvukový snímač vzdálenosti vpředu a vzadu na modelu a lze tak zjistit vzdálenost a zabránit případným kolizím a také lze zpomalit při přiblížení k překážce.

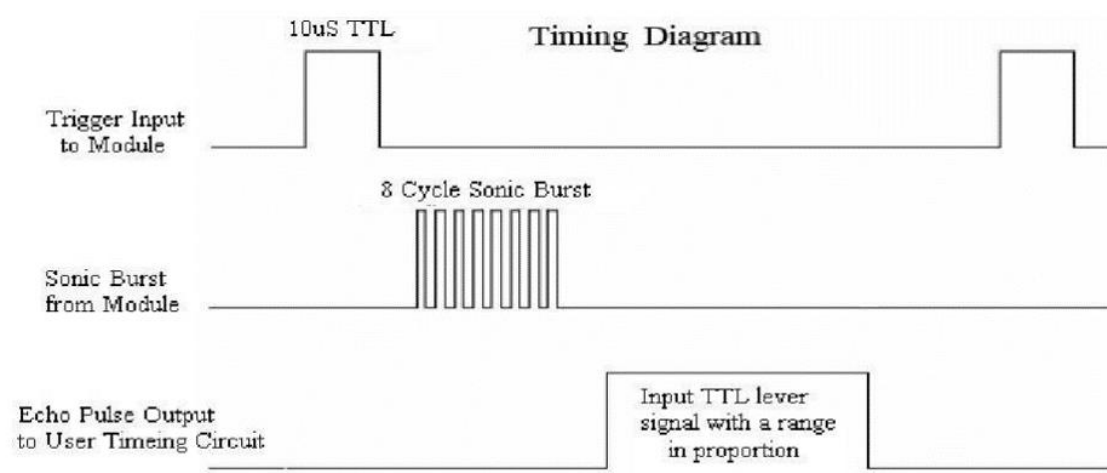
Je využíváno součástky HC-SR04, která dokáže detekovat vzdálenost od 2 cm do 400 cm. Tato součástka se dá v Česku pořídit za 120 Kč a výše, ale v zahraničních portálech, zejména z Číny, lze součástku zakoupit za cenu pod 20 Kč.

Vypočítání vzdálenosti zjednodušenou rovnicí lze vypočítat jako podíl μs (čas odezvy echo signálu) ku 58. Snímač vyžaduje napájení 5 V a pro získání vzdálenosti je nutné vyslat alespoň 10 μs signál na vstup *trigger*.



Obrázek 8 - Ultrazvukový snímač vzdálenosti HC-SR04

Snímač následně vyšle osm 40 kHz signálů, podle kterých vypočítá vzdálenost a vrátí ji výstupním signálem *echo* jako úroveň TTL signálu s šířkou podle vzdálenosti. Je doporučeno neopakovat cyklus alespoň 60 ms, aby se zabránilo špatnému měření.



Obrázek 9 - Časový diagram snímače

2. Software

V následujících kapitolách popíši základní algoritmus programu mikroprocesoru pro ovládání robotického vozítka. Ukázkový program je přiložen jako část přílohy A.

2.1. 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 nízkourovňový programovací jazyk pro symbolickou reprezentaci strojových instrukcí mikroprocesoru. Každý mikroprocesor má rozdílnou sadu instrukcí, kterou tvoří zejména výrobce. Tzn. 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 – Subtract with Carry*.

Nastavení

V programu se nejdříve provede nastavení proměnných, jako např. počáteční stav signalizačních LED, vytvoří se proměnné pro rozmístění tlačítek a následně se vytvoří makra pro definování nastavení pohybu a rychlostí. Poté se nastaví jaké porty, či jejich části, jsou vstupní. Musí se také vynulovat a nastavit čítače a 2 komparační registry pro správný chod PWM, tedy ovládání motorů.

```
.macro DOPREDU
    ldi r16, 0b10000001
    out PORTC, r16
    call rychlost
.endmacro

.macro DOZADU
    ldi r16, 0b01000010
    out PORTC, r16
    call rychlost
.endmacro
```

Obrázek 10 - Ukázka matek pro pohyb

2.2. Obecný algoritmus

Obecný algoritmus, resp. hlavní cyklus, aktualizace LED ze speciálního vyčleněného registru a následného načtení stavu tlačítek dálkového ovládání do pomocných vyčleněných registrů.

Následně probíhá blok porovnávání s předem definovanými tlačítky, kde se přeskakuje na návěští s testováním dalšího tlačítka, pokud současné porovnávání není shodné. Takto program pokračuje až do doby, kdy se testuje, jestli není stisknuto nic ze směrových tlačítek, poté se testují speciální tlačítka pro změnu rychlosti či rozsvícení/zapnutí světel robotického vozítka. Cyklus se opakuje.

cyklus:

```
out PORTB, r19           ;Aktualizace LED
call Delay

in r17, PINA            ;Načtení stavu tlačítek
mov r18, r17
andi r17, 0xf0         ;R17 - řízení jízdy
andi r18, 0x0f         ;R18 - přidavné funkce

cpi r17, TlacDopredu   ;dopredu
brne c01
DOPREDU|

c01: cpi r17, TlacDopreduR ;dopredu a doprava
brne c02
DOPREDU_R

c02: cpi r17, TlacDopreduL ;dopredu a doleva
brne c03
DOPREDU_L

c03: cpi r17, TlacDozadu   ;dozadu
brne c04
DOZADU
```

Obrázek 11 - Ukázka počátku cyklu

2.3. Řízení jízdy

Řízení jízdy v programu probíhá voláním maker, které dále definují nastavení registrů tak, aby na výstupu z řídicí jednotky vystupovaly správné signály PWM a aby se kola správně točila.

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, podobně jako tank, otáčet na místě, pro což je nadefinovaná pomalá, zhruba poloviční, rychlost.

Rychlost točení motorů 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 128, přesná polovina výkonu motorů, ale lze rychlost označit za pomalou jízdu.

2.4. 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.

zmena_svetel:

```
    sbrs r19, 6
    rjmp c51

    ldi r16, MaskaSvetla
    com r16
    and r19, r16          ;kdyz je 1 -> zmena na 0
    rjmp c52

c51:  ori r19, MaskaSvetla ;kdyz je 0 -> zmena na 1

c52:  call Delay2
      ret
```

Obrázek 12 - Ukázka programu pro přepnutí světel

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. naborání.

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.

Použitá literatura

- [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/vyrobky/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

Seznam obrázků a tabulek

Obrázky

Obrázek 1 - Technický výkres motorů	6
Obrázek 2 - DPS řídicí jednotky.....	7
Obrázek 3 - JTAG rozhraní	7
Obrázek 4 – Vnitřní zapojení AND hradel.....	8
Obrázek 5 - DPS budiče motoru.....	9
Obrázek 6 - Schéma budiče motorů	9
Obrázek 7 - ATmega32 PDIP pouzdro.....	10
Obrázek 8 - Ultrazvukový snímač vzdálenosti HC-SR04	13
Obrázek 9 - Časový diagram snímače	13
Obrázek 10 - Ukázka matek pro pohyb	14
Obrázek 11 - Ukázka počátku cyklu.....	15
Obrázek 12 - Ukázka programu pro přepnutí světél.....	17

Tabulky

Tabulka 1 - Parametry motorů.....	6
Tabulka 2 - Parametry ATmega32	11

Seznam zkratek

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

Seznam příloh

A. Elektronická příloha:

- Ukázkový program pro řídicí mikroprocesor
- Dokumentace k použitým součástkám