



## **Středoškolská technika 2014**

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

# **Autonomní vozítko**

Martin Škarytka

Odborný konzultant: Ing. Petr Kynčl

ISŠ Nová Paka

## **Prohlášení**

Prohlašuji, že jsem svou práci SOČ vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v seznamu vloženém v práci SOČ. Prohlašuji, že tištěná verze a elektronická verze soutěžní práce SOČ jsou shodné. Nemám závažný důvod proti zpřístupnění této práce v souladu se zákonem č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) v platném znění.

V Nové Pace dne 19. 3. 2014

podpis: .....

## **Poděkování:**

Děkuji panu Ing. Luboši Malému za obětavou pomoc a podnětné připomínky, které mi během práce poskytoval.

## **Anotace**

Práce se zabývá konstrukcí malého autonomního robota se dvěma sadami různých čidel. Robot je konstrukce dvoukolového vozítka se třetím opěrným bodem. Jedna sada čidel je optického charakteru a druhá mechanického. Informace z těchto čidel sbírá mikrokontrolér AVR, kontrolu motoru zabezpečuje dvojitý H-můstek. Celé zařízení je napájeno tříčlánkovou baterií Li-Fe.

## **Klíčová slova:**

ATmega328, Mikrokontrolér, PWM, Arduino,

## **ANOTATION**

The work deals with the assembly of a small autonomous robot with two different sets of sensors. The robot is the construction of the two-wheeled cart with the third supporting point. One set of the sensors has an optical character and the other one is mechanical. AVR microcontroller collects the information from these sensors and the double H-bridge controls the motor. The robot is powered by three cell Li-FePO<sub>4</sub> batteries.

## **Keywords:**

ATmega328, Microcontroller, PWM, Arduino

## **OBSAH:**

1. Úvod .....	5
2. Popis funkcí.....	6
3. Výroba plošného spoje.....	7
4. Stavba podvozku.....	8
5. Schémata.....	9
6. Seznam součástí.....	10
7. Čidla.....	11
8. Mikrokontrolér.....	12
9. Ovládání motorů.....	14
10. Programování.....	15
11. Program robota.....	16
12. Závěr.....	17
13. Zdroje.....	18
14. Příloha fotografie.....	19
15. Příloha výpis programu .....	21

Přílohou dokumentace je video autonomního vozítka ve formátu MP4.

## **1. Úvod**

V tomto projektu jsem se snažil vytvořit robota s autonomními schopnostmi, protože mě tato problematika velmi zajímala. Hlavním požadavkem bylo sledování čáry, později byly přidány mikrospínače pro detekci nárazu. Pro stavbu robota mi byla ze školy zapůjčena stavebnice Merkur, z níž jsem postavil podvozek robota.

Z důvodů jednoduchosti programování a značného množství různých zveřejněných příkladů, návodů a podobně jsem si pro řízení zvolil platformu Arduino.

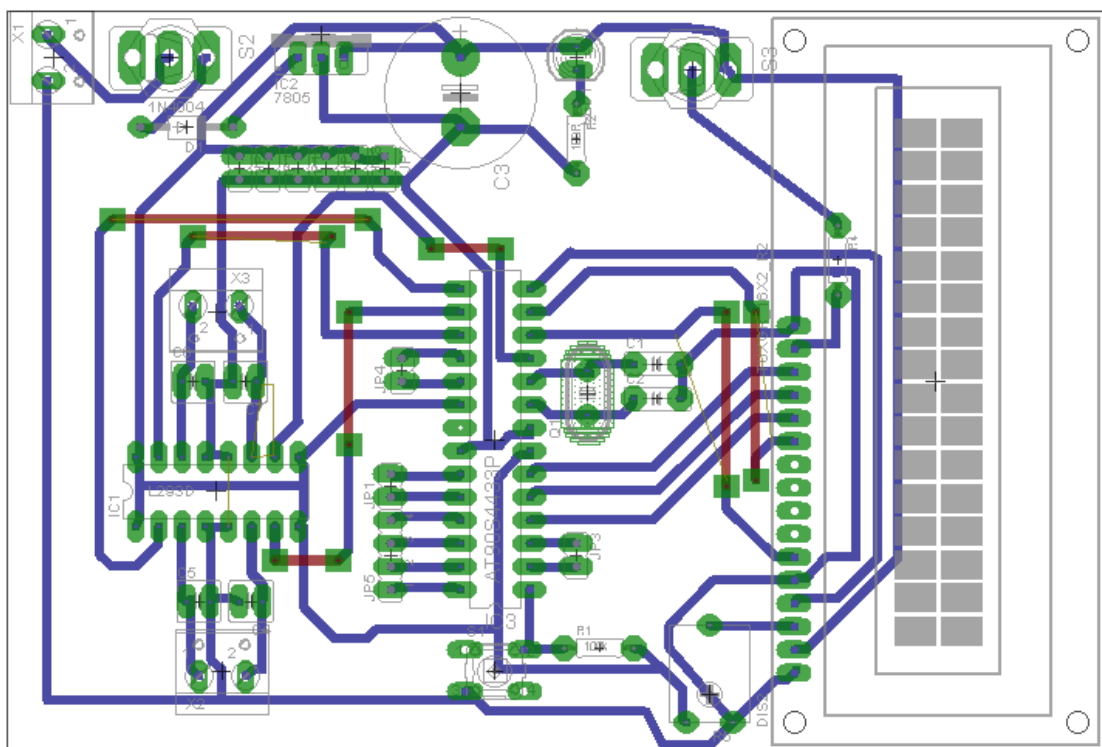
## **2. Popis funkcí**

Robot má dva hlavní algoritmy. Jeden pro sledování čáry a druhý vyhodnocuje stavy čelních nárazníku. První algoritmus vyhodnocuje polohu čáry pod čidly. Pokud je prostřední čidlo nad čarou, pojede rovně. Jakmile se čára dostane pod jedno z krajních čidel, zastaví se motor na opačné straně, a tím se robot vrátí zpět do přímého směru jízdy. Druhý algoritmus čte stavy dvou mikropsínačů. Při načtení jednoho ze dvou senzorů nárazníku začne robot couvat rovně dozadu, potom zastaví jedno určené kolo a pokračuje opět v přímé jízdě. Couvání a otáčení trvá dle nastavených hodnot v programu.

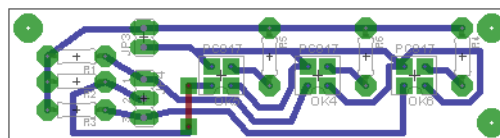
### 3. Výroba plošného spoje

Plošné spoje jsem navrhoval v programu Easel. Výsledný plošný spoj je na obrázku č. 1. Samostatná výroba probíhala na školní fréze na plošné spoje, kde jsem si mohl desku s plošnými spoji vyfrézovat.

Spoje jsou osazeny klasickými vývodovými součástkami. Robot je osazen celkem dvěma plošnými spoji, z toho jeden pro sledování čáry (obr. č. 2) je osazen třemi čidly CNY 70 a druhá deska, která je hlavní, je osazena mikroprocesorem Atmega328, H-můstkem L263D a stabilizátorem napětí.



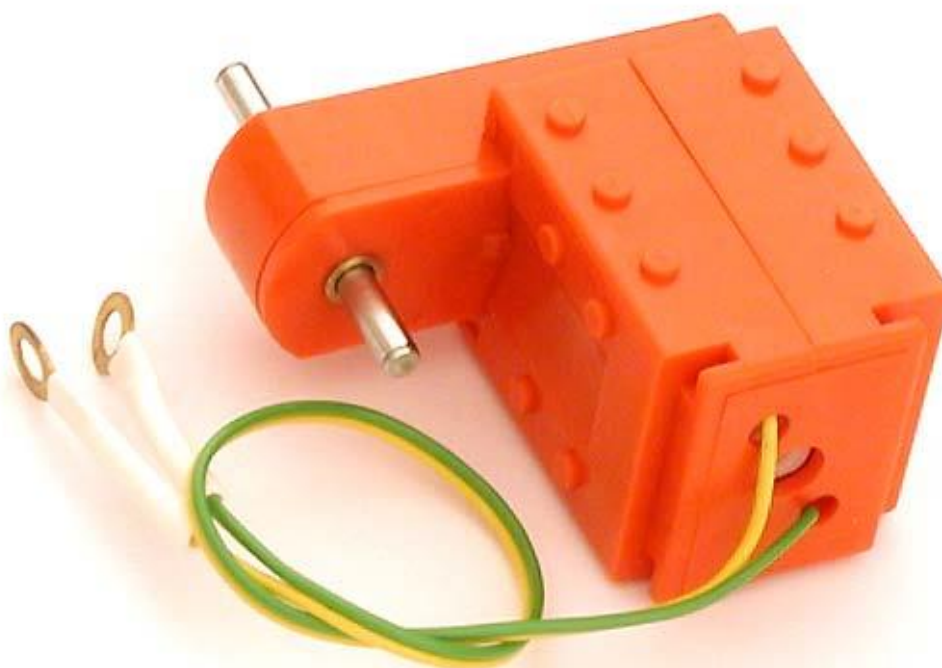
Obr. 1 plošný spoj



Obr. 2 plošný spoj čidel

#### **4. Stavba podvozku**

Podvozek je sestaven z komponentů stavebnice Merkur, je osazen dvěma motory Merkur na obrázku č. 3 s dvourychlostní převodovkou. Stavebnici tvoří mnoho univerzálních dílů z kovů a plastu, ze kterých jsem si vybral díly na stavbu svého podvozku. Jako třetí opěrný bod jsem zvolil pingpongový míček, který jsem z půlky vylil pryskyřicí. Po vytvrdnutí jsem zde vyvrtal otvor pro šroubek. Všechny díly jsou k sobě spojeny šroubky a matickami.

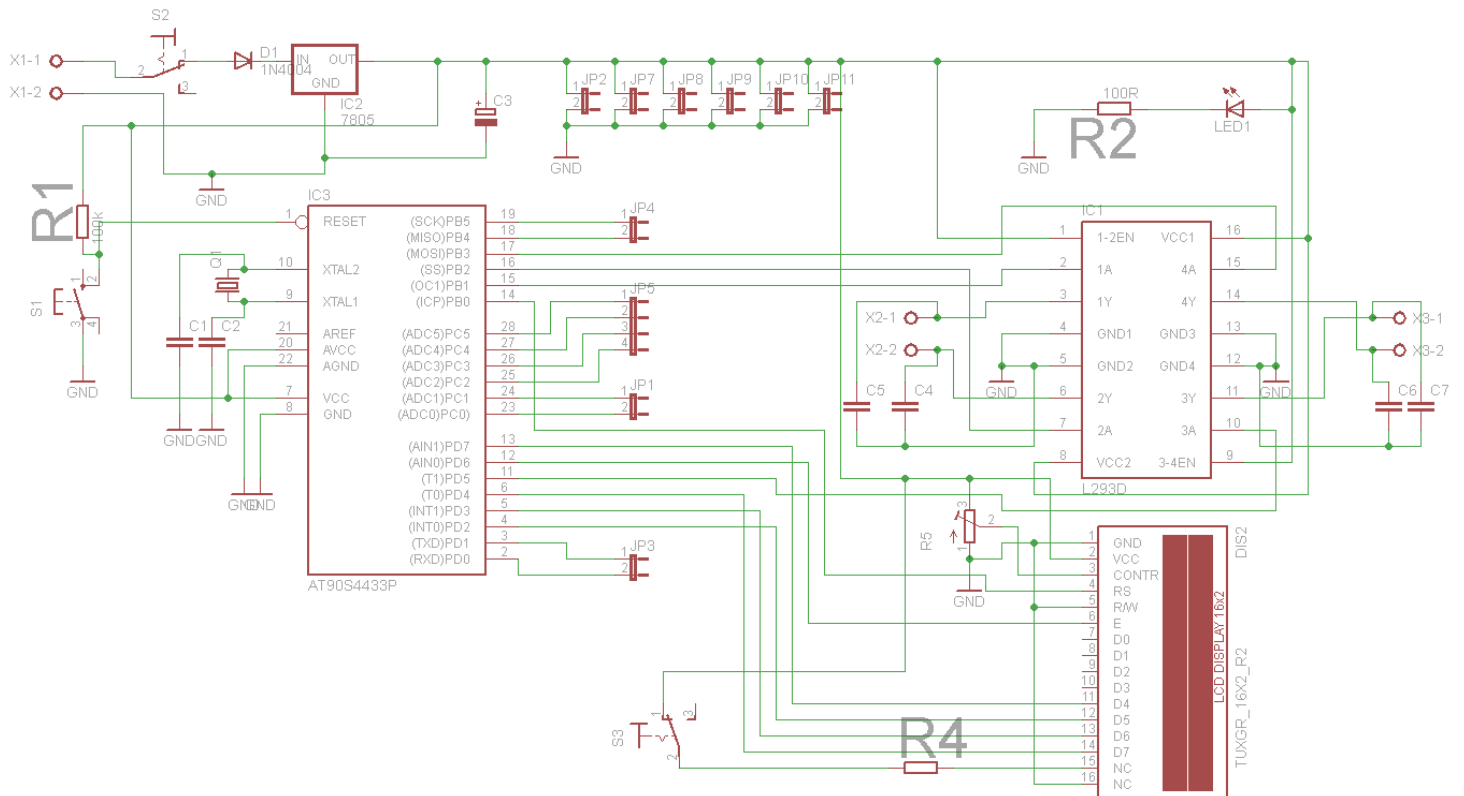


Obr. 3 motor stavebnice Merkur

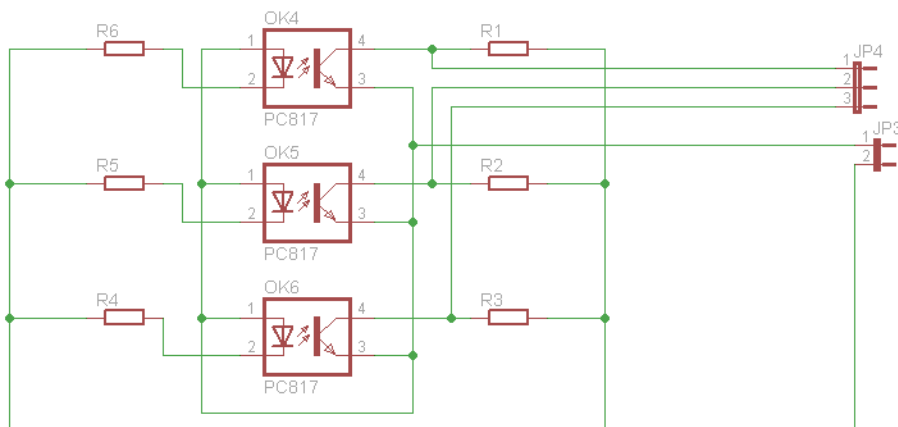


## 5. Schémata

První schéma představuje hlavní desku s H-můstkem a stabilizací pro mikrokontrolér (obr. č. 4), druhá deska obsahuje pouze trojici optických čidel pro sledování čáry (obr. č. 5).



Obr.4 hlavní řídicí deska



Obr.5 deska s čidly

## 6. Seznam součástek

### Hlavní deska

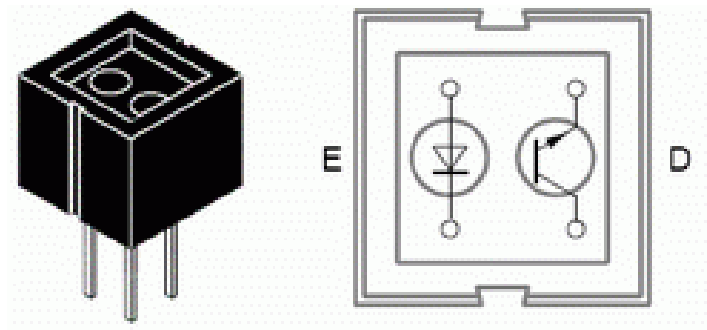
R1.....	10k $\Omega$
R2.....	100 $\Omega$
R3.....	100 $\Omega$
R4.....	100k $\Omega$
C1.....	22pF
C2.....	22pF
C3.....	4700 $\mu$ F
C4 - C7 .....	22nF
JP1 - JP10.....	hřeben s piny
IO1.....	L293D
IO2.....	7805
IO3.....	ATmega328
krystal.....	16MHz
S1.....	mikrospínač
S2 - S3.....	páčkový mikrospínač

### Deska čidel

R1 - R3.....	10k $\Omega$
R4 - R6.....	100 $\Omega$
JP1 - JP2.....	hřeben s piny
3x optická závora....	CNY70

## 7. Čidla

Pro sledování čáry jsou použity optické závory. Zvolil jsem čidla CNY70 (obr. č. 6). Snímače pracují na principu infračerveného paprsku (infračervená led dioda vysílá paprsky, které se od bílého podkladu odrazí a dopadnou na fototranzistor, ten sepne a na výstupu se objeví log. 0).



Obr.6 optické čidlo

Snímání nárazů zajišťují dva mikrospínače s dlouhými páčkami (obr. č. 7), které při nárazu sepnou, a na základě toho program vyhodnotí stav a provede příslušnou operaci.



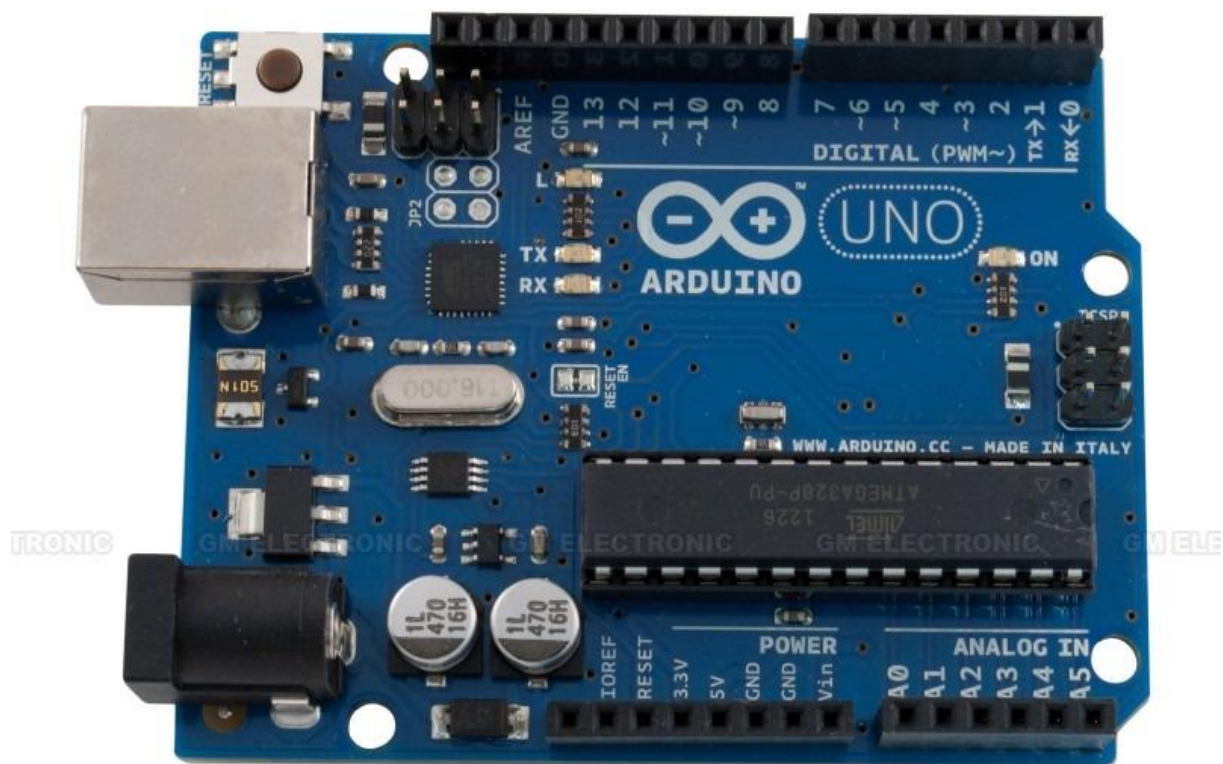
Obr.7 mikrospínač

## 8. Mikrokontrolér

Hlavním řídicím obvodem je ATmega 328 (obr. č. 8), ve kterém je nahrán bootloader Arduina a samotný řídicí program. Mikrokontrolér obsahuje 6 A/D převodníků 10bit, 13 digitálních vstupních i výstupních pinů, z nichž dva slouží pro sériovou komunikaci a 6 pinů má funkci PWM (pulzní a šířková modulace). Je zasazen ve vývojové desce, která obsahuje veškeré potřebné komponenty pro běh mikrokontroléru. Pro jeho programování slouží ATmeg16u2, který tvoří převodník USB => RS232 (obr. č. 9).



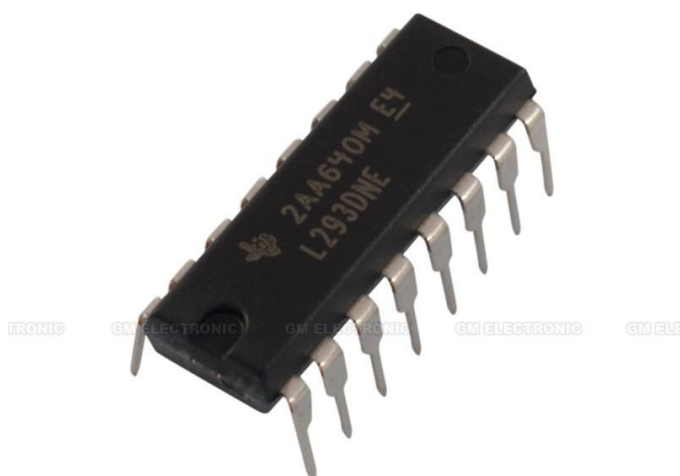
Obr.8 mikroprocesor



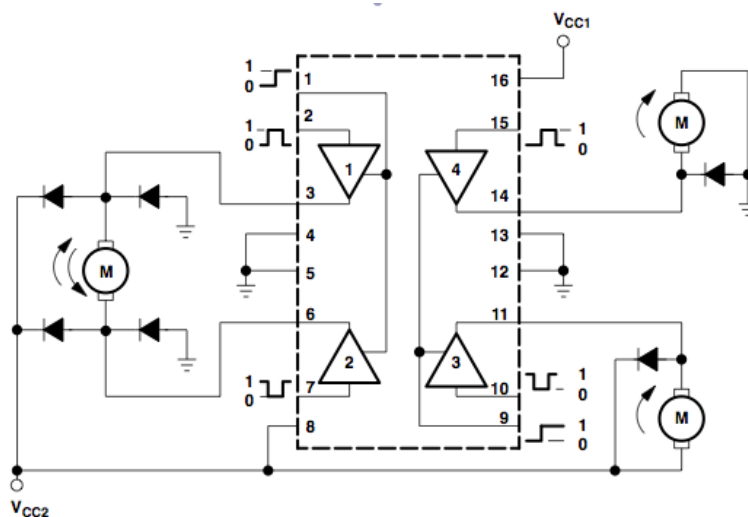
Obr.9 vývojová deska Arduino

## 9. Ovládání motorů

K řízení motorů jsem použil dvojitý H-můstek L293D (obr. č. 10). Výkon motorů je regulován pomocí PWM pulzu. Můstek je schopen dodat proud o velikosti 0,5 A. Na pouzdro je připevněn chladič z důvodu nadměrného zahřívání. V pouzdře jsou integrované ochranné diody. Na obrázku č. 11 můžete vidět vnitřní zapojení, kde se nacházejí dva bloky, z nichž každý obsahuje dva vstupy a dva výstupy.



Obr.10 integrovaný obvod L293D



Obr.11 vnitřní zapojení H-můstku

## **10. Osazení plošného spoje**

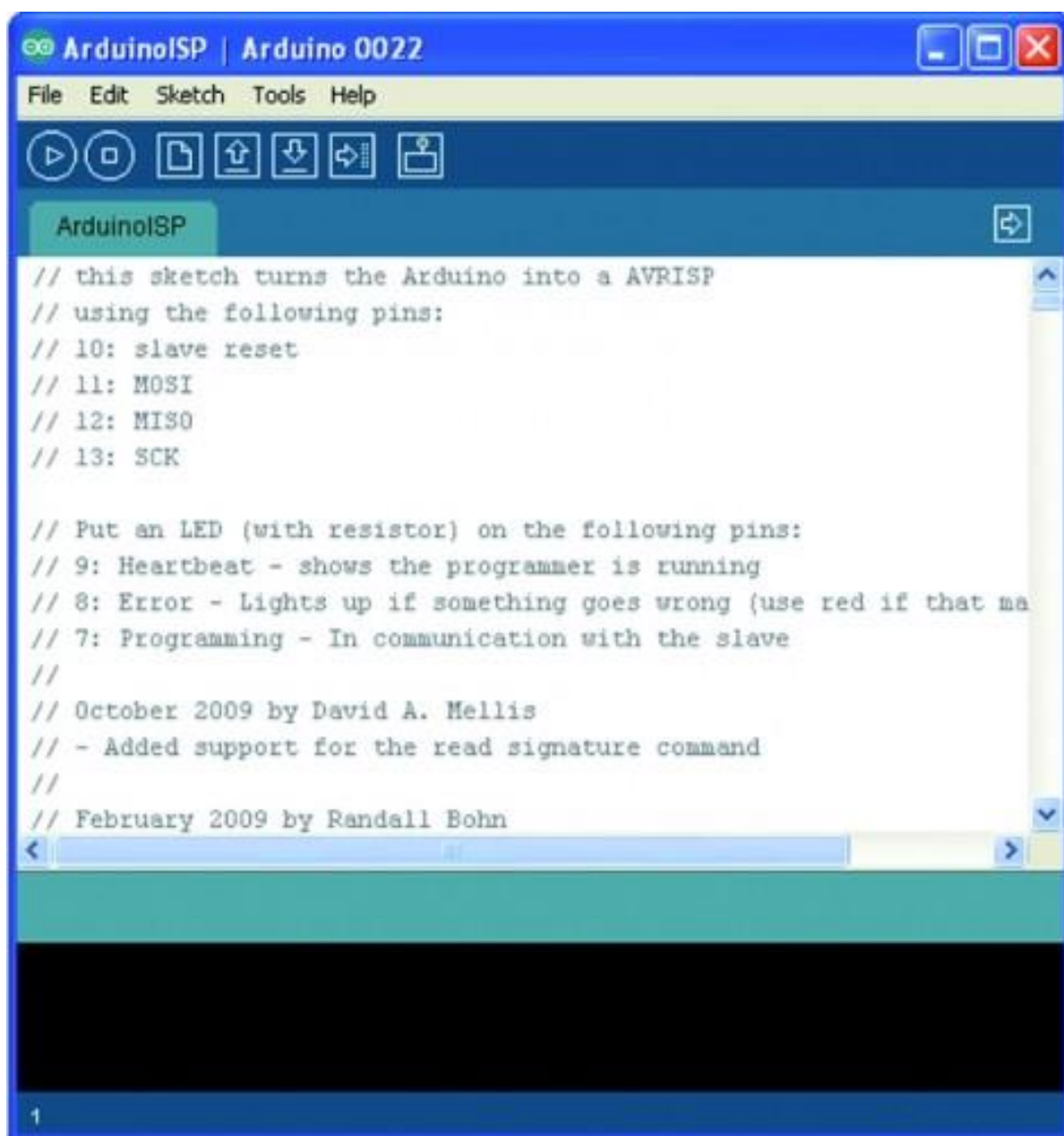
Po vyvrtání plošného spoje jsem desku ze strany spojů očistil od koroze obyčejnou gumou na gumování a potřel ji rozpuštěnou kalafunou v lihu. Jako první jsem osadil propojky s rezistory, patice integrovaných obvodů, kondenzátory, diodu a led diodu, hřeben s piny pro připojení čidel, dále vypínače, krystal a nakonec velký elektrolytický kondenzátor.

## **11. Oživení**

Při oživování nastal problém s rušením mikrokontroléru od motorů, který jsem odstranil přímým napájením jejich regulátoru. Jinak nedošlo k žádným vážnějším problémům.

## **12. Programování**

Program je napsán v jazyce Wiring, což je zjednodušený jazyk C. Na tento programovací jazyk je na internetu mnoho návodů a příkladů. Programování je v něm poměrně snadné a přehledné. Samotné programování se provádí přímo na platformě od Arduina, které můžete vidět na obrázku č. 12. Programování probíhá zjednodušeně, napíšete program, platforma Arduino ho zkompiluje, zda půjde nahrát do mikrokontroléru. Dalším krokem bude připojení vývojové desky k PC pomocí USB kabelu. Následně je třeba vybrat správný port a vývojovou desku a nahrát program.



Obr.12 programovací prostředí

### 13. Program robota

Výpis programu robota je uveden na konci dokumentu z důvodu jeho délky.



## 14. Závěr

Při stavbě tohoto robota jsem se setkal s několika problémy, které jsem musel odstranit. První překážkou byly časové výpadky mikroprocesoru nejspíše z důvodu rušení motorů. Tento problém jsem vyřešil tím způsobem, že jsem H-můstek připojil přímo na baterii a přerušil napájení ze stabilizátoru, který napájí i samotný mikrokontrolér. Tím vznikl druhý problém, na baterii bylo vyšší napětí, než jaké potřebují motory, toto jsem vyřešil nastavením PWM regulace na motorech. Při navrhování plošného spoje došlo k několika chybám jako například jiná knihovna LCD, než jsem měl já, prohození drah, místo displeje jsem na jeho výstup napojil dvojbarevnou led diodu, která signalizuje aktuální program. Po delším provozu robota v kuse jsem zkontroloval orientačně teplotu H-můstku. Protože jsem zjistil, že teplota je příliš vysoká, použil jsem pro jeho chlazení nouzové řešení, tavnou pistolí jsem přilepil kus plechovky od nápoje. Jako třetí opěrný bod pro robota jsem použil pingpongový míček, což se ukázalo jako nepříliš vhodné řešení, jelikož po čase ježdění se míček probrousil o podklad, po kterém jezdí. Při prvním spuštění robota na zemi jsem zjistil, že jsou kola z poměrně tvrdého plastu a protácejí se, proto jsem na kola nalepil oboustrannou lepicí pásku.

V budoucnu bych chtěl tohoto robota rozšířit o některé další senzory jako například senzor vzdálenosti, kompas, akcelerometr atd.

V příloze s dokumentací přikládám video s autonomním vozítkem.

## 15. Seznam zdrojů

[http://www.solarskit.wz.cz/source/robotika/century/CENTURY\\_amaro.pdf](http://www.solarskit.wz.cz/source/robotika/century/CENTURY_amaro.pdf)

<http://www.solarskit.wz.cz/source/robotika/bob/cny70obr.gif>

<http://www.gme.cz/vyvojovy-kit-arduino-arduino-uno-p772-008>

<http://www.gme.cz/arduino-uno-nahradni-mcu-mega328-p772-010>

<http://www.gme.cz/l293d-p399-017>

<http://www.open-electronics.org/arduino-isp-in-system-programming-and-stand-alone-circuits/>

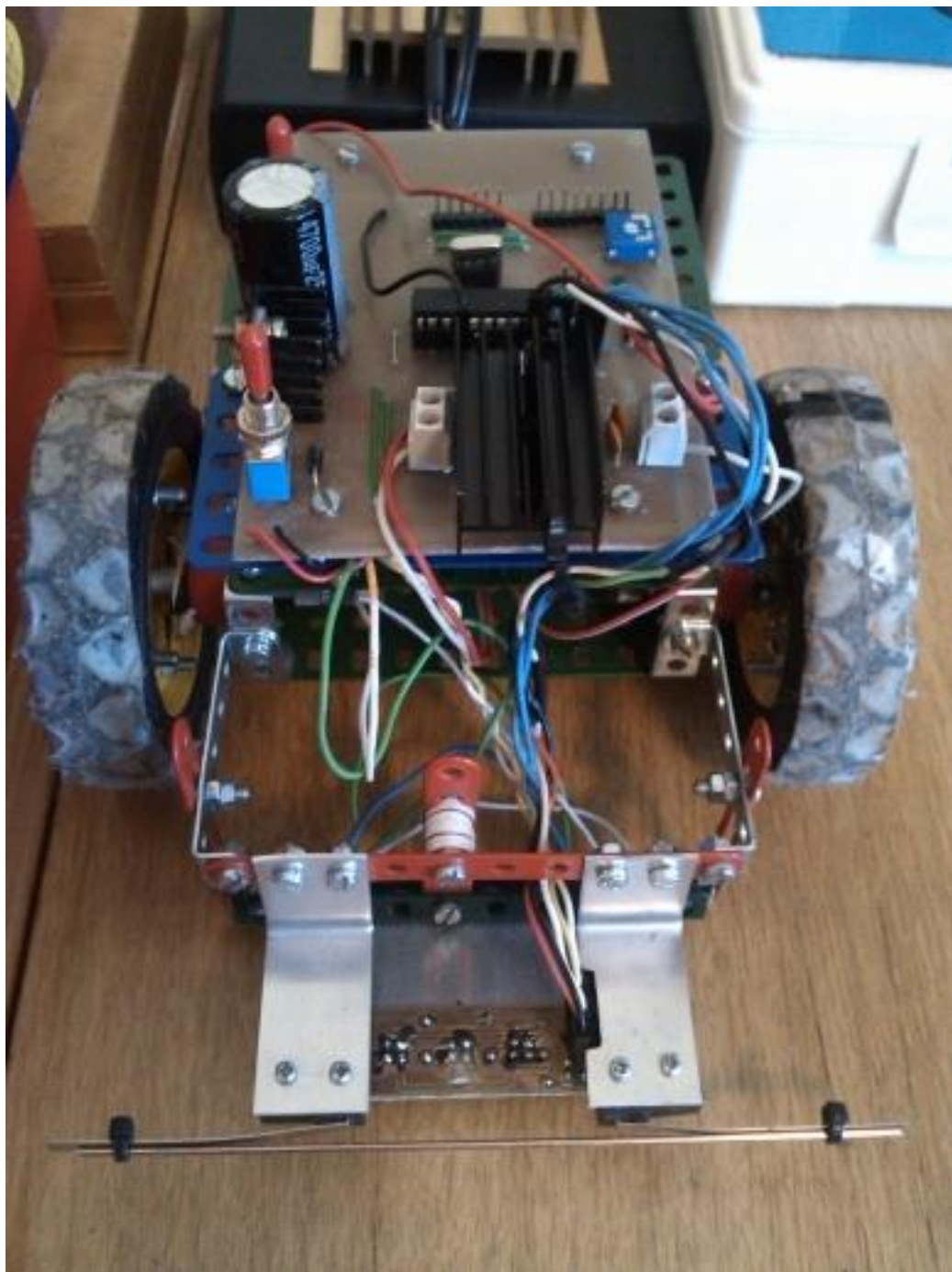
[http://www.merkursets.ca/popup\\_image.php?plD=74&osCsid=ccbvlbytgg](http://www.merkursets.ca/popup_image.php?plD=74&osCsid=ccbvlbytgg)

<http://www.gme.cz/mikrospinac-zippy-sm-05h-09p0-z-p630-050>

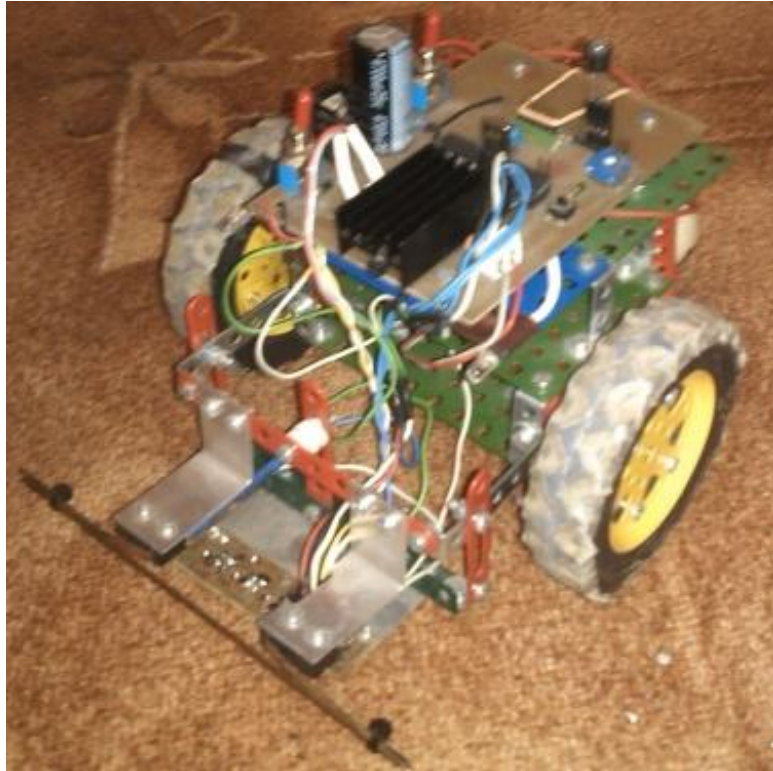
<https://www.adafruit.com/blog/2012/05/30/ask-an-educator-how-can-i-control-a-solenoid-or-motor-with-an-h-bridge/>

[http://www.josefnav.cz/Ardubot\\_clanek\\_3.html](http://www.josefnav.cz/Ardubot_clanek_3.html)

## 16. Příloha - fotografie



Obr. 13 autonomní vozítko



Obr. 14 autonomní vozítko – pohled zepředu



Obr. 15 autonomní vozítko – pohled zezadu

## 17. Příloha - výpis programu

```
int mod;
//motory
#define A 10
#define B 11
#define C 9
#define D 5
//čidla
#define C1 A0
#define C2 A1
#define C3 A2
//nárazníky
#define N1 A3
#define N2 A4
//tlačítko ,led
#define T1 12
#define LedR 8
#define LedG 6

void setup()
{
    pinMode(A, OUTPUT);
    pinMode(B, OUTPUT);
    pinMode(C, OUTPUT);
    pinMode(D, OUTPUT);
    pinMode(LedR, OUTPUT);
    pinMode(LedG, OUTPUT);
    pinMode(N1, INPUT);
    pinMode(N2, INPUT);
    pinMode(C1, INPUT);
    pinMode(C2, INPUT);
    pinMode(C3, INPUT);
    pinMode(T1, INPUT);
    delay(2000);
}

void loop()
{
    menu();
    start();
}

void menu()
{
    while(digitalRead(T1) == 1)
    {
        if(analogRead(N1) > 900)
        {
            if(mod == 1){
                mod = mod = 1;
            }
        }
    }
}
```

```

}
if(analogRead(N2) > 900 )
{
  if(mod = 2){
    mod = mod = 2;
  }
}
switch (mod)
{
case 1:
  //sledování čáry
  digitalWrite(LedR, HIGH);
  digitalWrite(LedG, LOW);
  break;

case 2:
  //autonomní s nárazníky
  digitalWrite(LedG, HIGH);
  digitalWrite(LedR, LOW);
  break;
}
delay(100);
while(digitalRead(N1) == 0 &&
digitalRead(N2) == 0
  && digitalRead(T1) == 0){
}
}

}
}

void start()
{
  switch (mod)
  {
case 1:
  sledovani_cary();
  break;

case 2:
  narazniky();
  break;
}
}

void sledovani_cary()
{
  while(digitalRead(T1) == 0)
  {
    digitalWrite(LedG, LOW);
    digitalWrite(LedR, LOW);
    if(analogRead(C1) < 700)
    {
      if(mod = 1){
        mod = mod = 1;

```

```

    }
}
if(analogRead(C3) < 700 )
{
    if(mod = 2){
        mod = mod = 2;
    }
}
if(analogRead(C2) < 700 )
{
    if(mod = 3){
        mod = mod = 3;
    }
}

switch (mod)
{
case 1:
    digitalWrite(A, LOW);
    digitalWrite(B, LOW);
    analogWrite(C, 90);
    digitalWrite(D, LOW);
    break;

case 2:
    digitalWrite(A, LOW);
    digitalWrite(B, LOW);
    analogWrite(C, 90);
    analogWrite(D, 90);
    break;

case 3:
    digitalWrite(A, LOW);
    digitalWrite(B, LOW);
    digitalWrite(C, LOW);
    analogWrite(D, 90);
    break;
}

delay(10);
}

void narazniky()
{
    while(digitalRead(T1) == 0)
    {
        digitalWrite(LedG, HIGH);
        digitalWrite(LedR, HIGH);
        if(analogRead(N1) > 900)
        {
            if(mod = 1){
                mod = mod = 1;
            }
        }
    }
}

```



```

    }
}
else if(analogRead(N2) > 900 )
{
    if(mod = 3){
        mod = mod = 3;
    }
}
else
{
    if(mod = 2){
        mod = mod = 2;
    }
}

switch (mod)
{
case 1:
    analogWrite(A, 80);
    analogWrite(B, 80);
    digitalWrite(C, LOW);
    digitalWrite(D, LOW);
    delay(700);
    analogWrite(A, 80);
    digitalWrite(B, LOW);
    digitalWrite(C, LOW);

    digitalWrite(D, LOW);
    delay(700);
    break;
}
delay(10);
} }

```