



Středoškolská technika 2011

Setkání a prezentace prací středoškolských studentů na ČVUT

SESTAVENÍ A NAPROGRAMOVÁNÍ ROBOTŮ S VYUŽITÍM ROBOTICKÉ STAVEBNICE BIOLOID.

Jan Stránský



**Střední průmyslová škola elektrotechnická, Brno,
Brno, Kounicova 16**

PRAKTICKÁ ZKOUŠKA Z ODBORNÝCH PŘEDMĚTŮ

Studijní obor: 26-43-M/004 Slaboproudá elektrotechnika
Zaměření: Automatizační technika
Třída: 4A
Školní rok: 2010/2011
Jméno a příjmení: Jan Stránský

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

V Brně dne 11. dubna 2011.

Jan Stránský

OBSAH PRÁCE

1. Řešení maturitní práce

1.1. Bioloid

1.1.1. Bioloid CM-5

1.1.1.1. Popis servomotorků ax-12+s

1.1.1.2. Popis senzorů

1.1.2. Bioloid CM-510

1.1.2.1. Popis řídicí jednotky

1.1.2.2. Popis dalších senzorů a rozšíření

1.1.2.2.1. IR senzor

1.1.2.2.2. Distance Measurement Sensor (senzor pro měření vzdálenosti)

1.1.2.2.3. Gyro senzor (snímač polohy robota pomocí magnetického pole země)

1.1.2.3. Popis robotu humanoid

1.2. Robo plus

1.2.1. Robo plus manager

1.2.2. Robo plus motion

1.2.3. Robo plus task

1.3. Program

1.3.1. Program pro testování senzorů

1.3.2. Nastavení v robo plus manager

1.3.3. Program v robo plus motionu

1.3.4. Program v robo task

2. Závěr

3. Seznam použité literatury.

Úvod práce

Na střední školu jsem šel se snem, že se zde naučím programovat. Když jsme se ve druhém ročníku měli rozhodnout, na který obor půjdeme, byla automatizace jasná volba. Ve druhém ročníku jsem se dozvěděl, že ve škole, konkrétně na oboru automatické, se může pracovat s robotickou stavebnicí a dělal jsem všechno proto, abych se do práce s ní mohl zapojit. Ve třetím ročníku jsem se skutečně dostal na obor automatizace a hned na začátku nám paní profesorka Odstrčilíková nabídla možnost zapojit se do pracovní skupiny, která se po škole zájmově zabývala prací se stavebnicí Bioloid. Práce s touto stavebnicí mě nadchla a hned na začátku třetího ročníku jsme se začali připravovat na soutěž. Práce s Bioloidem mi jde, a proto jsem se rozhodl se touto stavebnicí a prací s ní zabývat i ve své maturitní práci. Rozhodl jsem se pro práci s humanoidem z prostého důvodu. Byla to výzva. Práce s humanoidem je totiž neobyčejně složitá z hlediska rovnováhy. Robot totiž neumí vyvažovat tak jako člověk

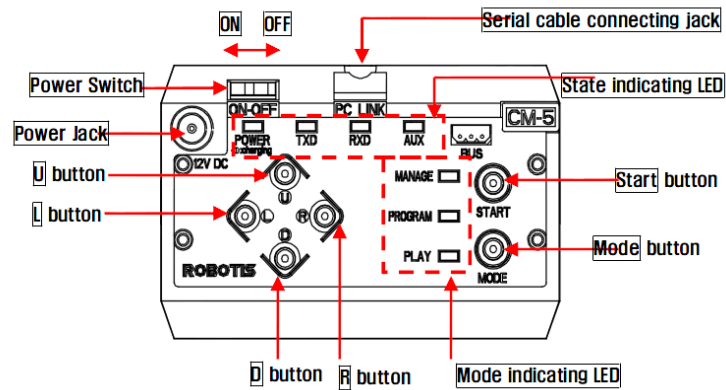
1. Řešení maturitní práce

1.1. Bioloid

Bioloid je robotická stavebnice sládající se z několika základních součástí. Mezi tyto součásti patří zejména řídicí jednotka, servomotory, senzory, baterie, spojovací díly, dálkový ovladač a podobně. Stavebnice Bioloid je vhodná pro výuku robotiky na školách. V současné době jsou přímo od výrobce k dispozici knihovny pro programování v jazyce C++ pro starší i novější verze. Tato stavebnice má dvě základní řídicí jednotky se kterými na škole můžeme pracovat, CM-5 a CM 510. Součástí stavebnice tady na škole je 60 servomotorů a z toho je asi 58 plně funkčních. Další důležitou částí jsou senzory, které robotům umožňují základní orientaci v prostoru. K dispozici máme několik různých senzorů, kterými se budu zabývat později. Další nepostrádanou součástí jsou spojovací díly, propojovací kabely a samozřejmě baterie a zdroje.

1.1.1. Bioloid CM-5

CM-5 je řídicí jednotka Bioloid, která funguje jako mozek robota. To může ovládat 28 AX-12 + s (servomotorů) a 10 AX-S1s (základní verze ze senzorů) současně. Tlačítka, které má na sobě, se používají jako vstupní zařízení a může fungovat pro základní ovládání robota. U ovládání přes tlačítka vždy záleží na programu, který je v jednotce nahraný.

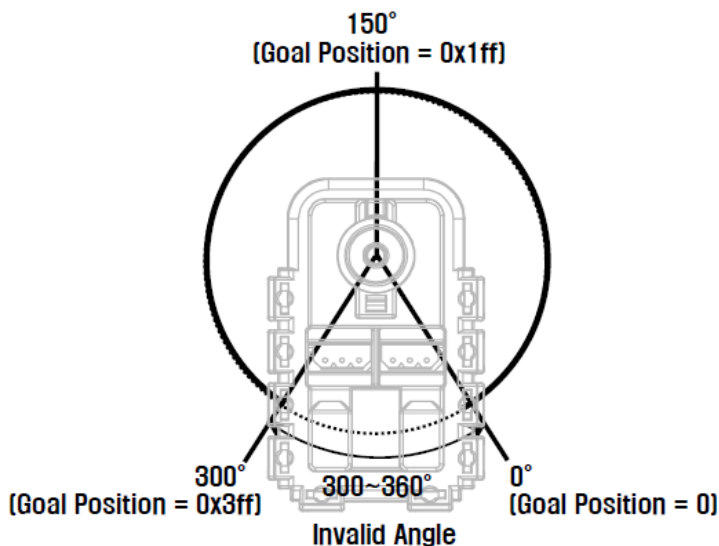


1.1.1.1. popis servomotorů AX-12+s



Servomotor ax+12- vyfocený ze dvou stran.

Ax-12+s je servomotor určený pro stavebnici Bioloid. Je to chytrý modulární pohon s převodovkou a precisním stejnosměrným motorem. Navzdory své velikosti je chopen vytvořit poměrně velký točivý moment a je vyroben z pevných materiálů, které zajišťují pevnost a odolnost motorů. Tyto motory mají 1024 pozic pro přesné nastavení polohy a jsou schopny se protáčet bez omezení počtu otočení. Jejich menší nevýhodou je, že pozice do kterých ho lze nastavit, nejsou na všech 360° kolem něj.



Nákres možnosti nastavení pozice (otočné rozpětí).

Při protáčení motor není schopen zastavit se mimo zónu 0 – 300°. Motor je možno vypnout, ručně nastavit a po následném zapnutí odečíst jeho pozici v programu, který ji přečte a zapíše jako číslo od 0 do 1023. Ve středu, který je na nákresu označen 150°, je pozice přesně 512. Tato pozice je při programování a při představování pozice zásadní.

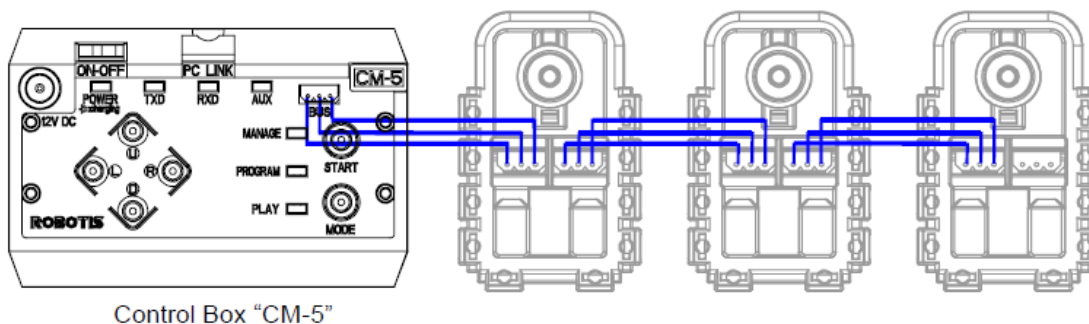


Schéma zapojení motorů a řídicí jednotky.

Motor je napájen stejnosměrným napětím o velikosti 12V. Je průchozí a lze za sebe zapojit všechny motory, které příslušná jednotka zvládne. Motor má také teplotní senzor, pokud by se motor začal přehřívat automaticky se vypne.

1.1.1.2. popis senzorů

Jednotka CM-5 zvládá pouze jeden typ senzorů a to senzory AX-S1



Foto senzoru AX-S1

Tento senzor je chopen odečítat vzdálenost ve třech směrech, snímat zvuk a také vydávat jednodušší tóny. Senzor vypadá jako servomotor AX – 12 + S jen s tím rozdílem, že má viditelné infra a fotodiody a nemá možnost otáčení jako motorek. Nemůže se otáčet sám, ale

ani s manuální pomocí. Zapojuje se stejně jako servomotor.

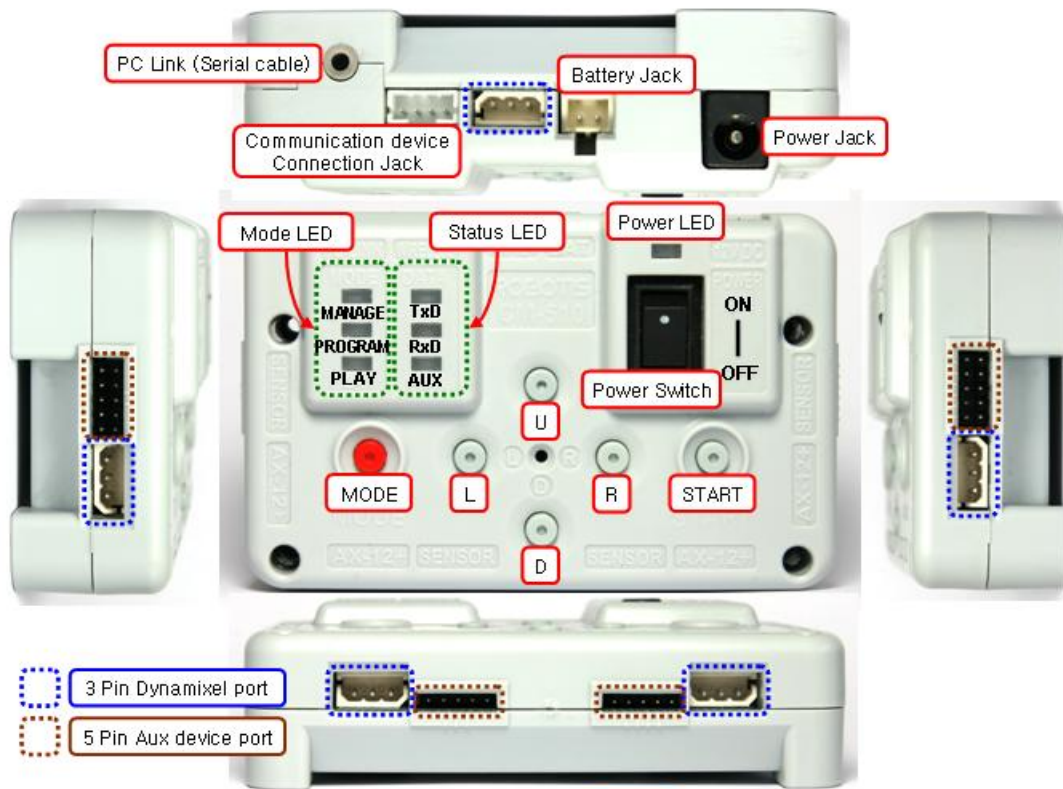
Ve své práci jsem tento senzor nakonec nepoužil a proto se o něm dál rozepisovat nebudu.

1.1.2. Bioloid CM-510



CM-510 je řídicí jednotka obsahující procesor Atm mega 256. Na rozdíl od předchozí veze má velkou spoustu rozšíření. Je menší než CM-5, má větší paměť, má navíc 6 portů pro své senzory viz 4.1.2.2. Při připojení přijmače je schopen přijmat informace z dálkového ovladače. Ovládání je podobné jako u CM-5 s tím rozdílem, že CM-510 je schopen zjistit aktuální výdrž baterie, ke které je připojen, respektive při přidržení tlačítka U se roznou světla nad tlačítkem mode. Pokud svítí 3, je baterie nabitá, pokud svítí 1 baterie je téměř vybitá.

1.1.2.1. popis řídicí jednotky



PC link – slouží k připojení sériového kabelu

Communication Device Connection Jack – používá se pro bezdrátovou komunikaci ZIG 110 (v tomto projektu ho nepoužívám)

BatteryJack - používá se pro připojení baterie

Power Jack – slouží k připojení zdroje

Dynamixel port – slouží k připojení servomotorů AX – 12 + a AX - S1

Aux device port - slouží k připojení IR senzoru

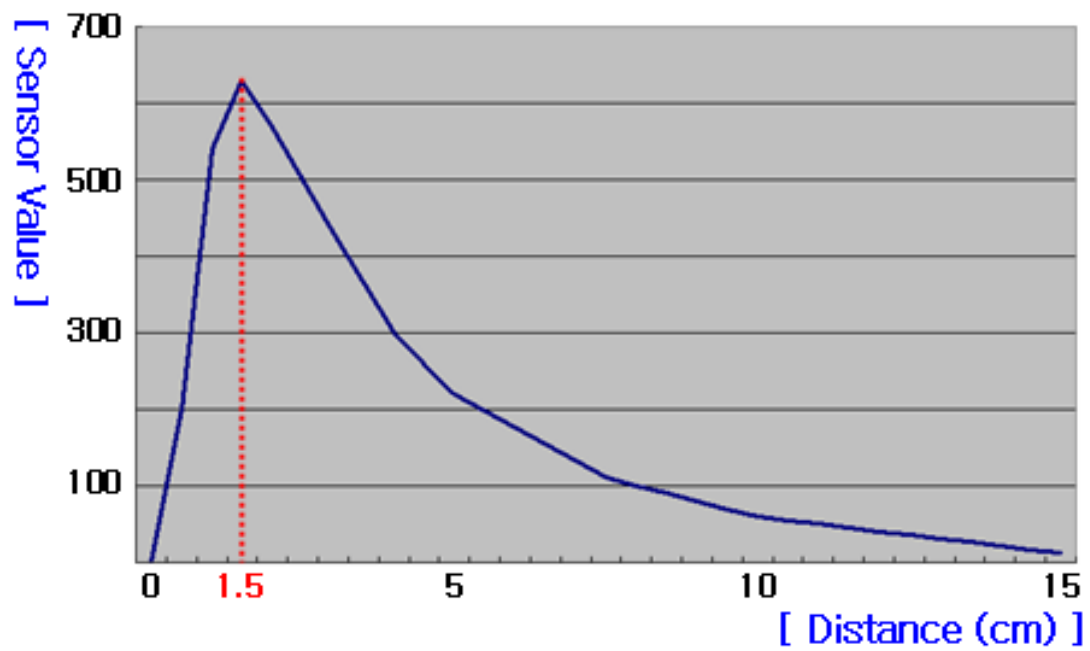
1.1.2.2. popis dalších senzorů a rozšíření

1.1.2.2.1. IR senzor



[IR Sensor]

Rozměry senzoru: 24mm*18mm*12mm.



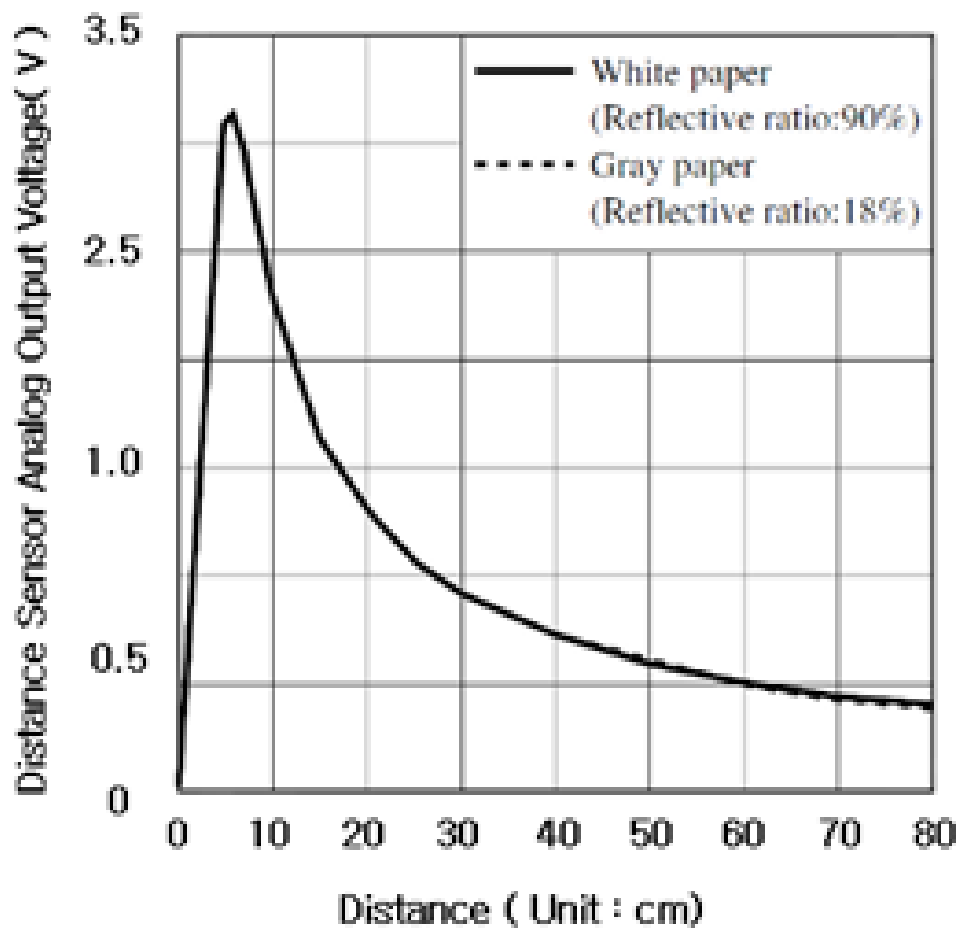
Hodnota vysílaná senzorem v závislosti na vzdálenosti.

1.1.2.2. Distance Measurement Sensor

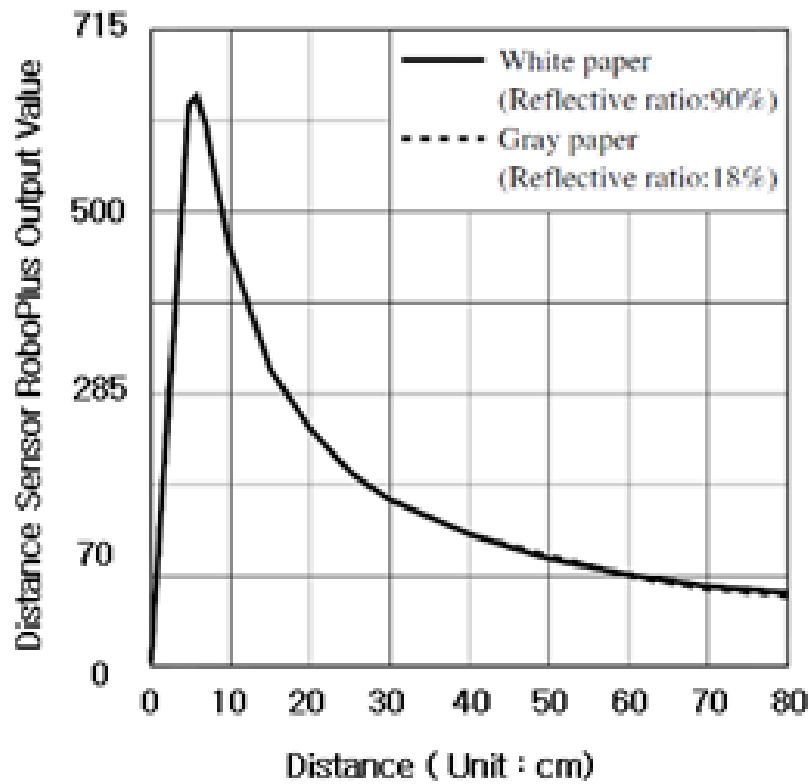


[Distance Measurement Sensor]

Váha: 4,4g
Maximální dosah: 80 cm
Napájecí napětí 4,5 – 5,5 V



Graf výstupu senzoru ve voltech c závislosti na vzdálenosti.



Graf hodnoty kterou senzor odesílá v závislosti na vzdálenosti.

Senzor se připojuje na řídicí jednotku CM-510 na Aux device port.

1.1.2.2.3. gyro senzor



Gyroskop AX-S20 je schopen měřit svou polohu vůči zemským magnetickým pólům ve všech třech osách. Díky němu je robot schopen určit o kolik se otočil, nebo jak rychle se

otáčí. Připojuje se stejně jako senzor AX-S1.

1.1.2.3. popis robotu Humanoid



Obrázek robotu tak, jak je vytištěn v návodu.

Robot humanoid, se kterým pracuji v této maturitní práci, je sestaven z velké části podle návodu, konkrétně typ A. Jedinou úpravou je vynechání modulu pro přímání signálu z dálkového ovládní RC 100 (při práci jsem ho nepoužíval, proto se o něm nezmiňuji).

Místo něj jsem do hlavy umístil 3 vysoce svítivé led diody.

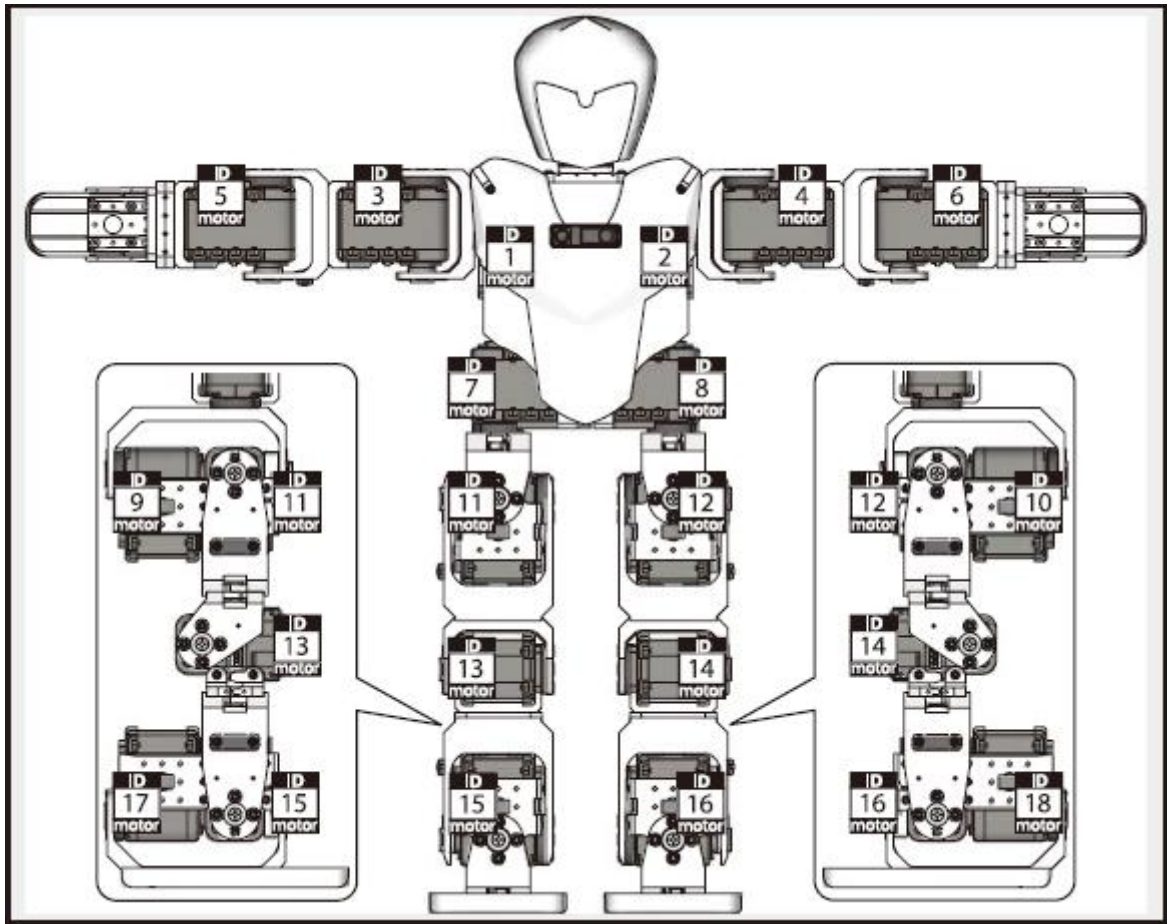


Schéma rozestavení motorů robota.

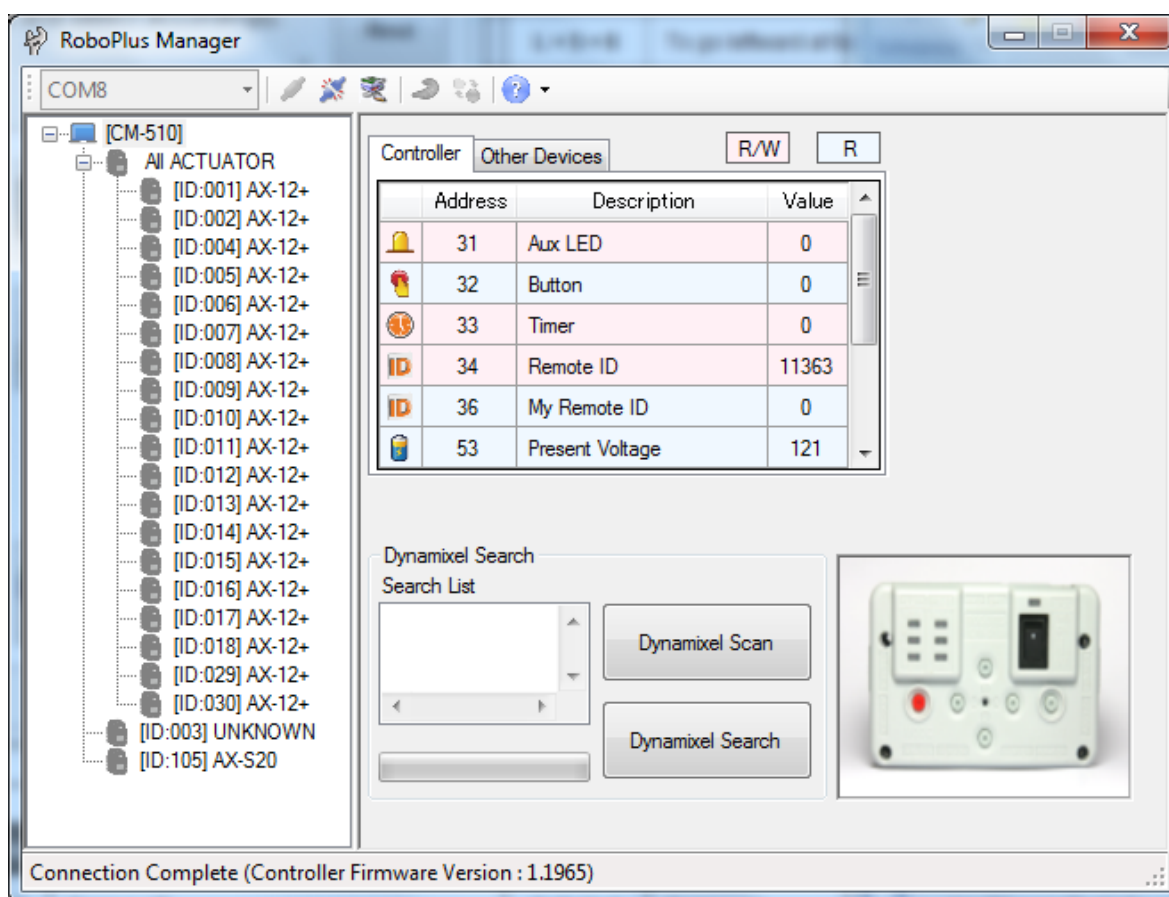
Při stavění robota jsem toto rozestavení nepoužil a čísla na servomotorech byla změněna později viz. bod 1.2.1. a 1.3.2.1.

Navíc má ještě dva motory s ID 29 a 30 na nohách. Na těchto motorech má senzory.

1.2.Robo Plus

Robo plus je program dodávaný přímo od výrobce. Prostřednictvím tohoto programu jsem dělal veškeré softwarové nastavení a programování. Program obsahuje RoboPlus Task, což je program pro vytváření logiky, RoboPlus Manager, ve kterém se dají přecíslovat servomotory a získat informace o použitých portech a jiném nastavení robota. Další nezbytnou věcí v je RoboPlus Motion. V tomto programu se programují všechny složitější kroky, RoboPlus Terminál je program pro přímou komunikaci s robotem po sériové lince. Další je program manuálů.

1.2.1. RoboPlus Manager



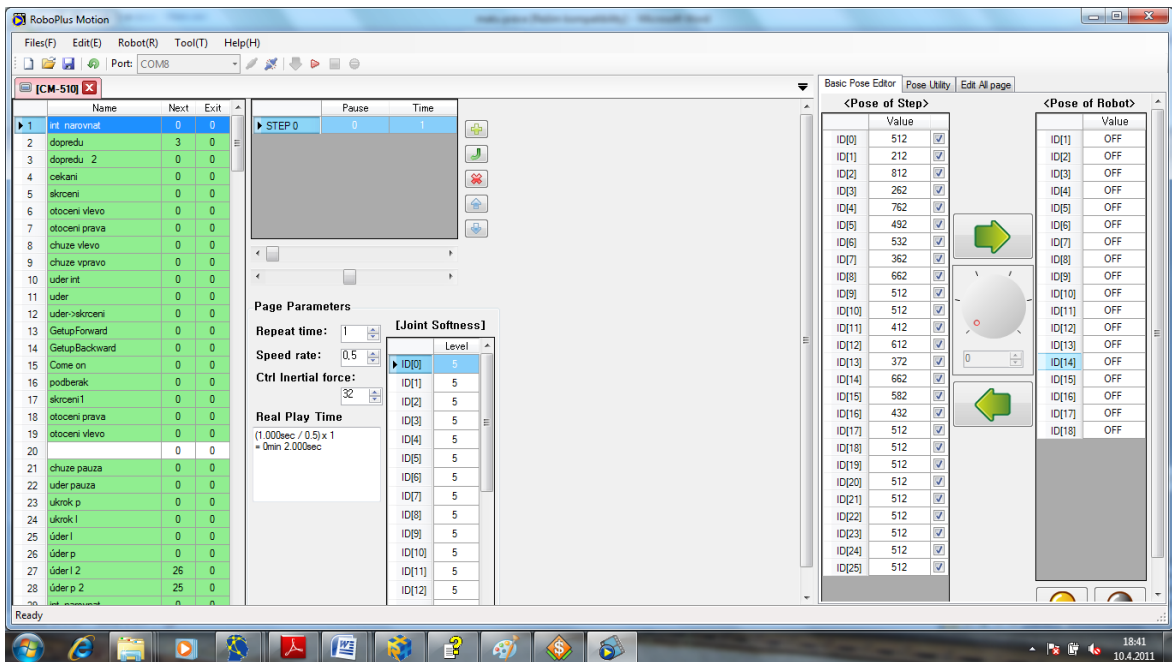
Sejmutá obrazovka RoboPlus Manager při připojeném robotovi.

V tomto programu můžu měnit čísla motorů, nepřímo měnit teplotu, při které se motory vypínají nebo zjistit osy AX-S20.

V tomto programu lze také například zapínat světelné označení jednotlivých motorků, což značně usnadňuje jejich přechíslování.

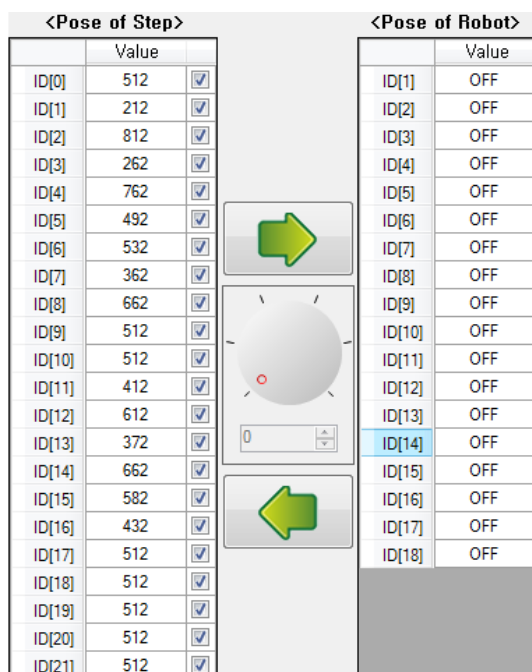
Program pracuje s připojeným robotem v reálném čase. Neustále komunikuje po sériové lince.

1.2.2. RoboPlus Motion



Sejmutá obrazovka RoboPlus Motion.

V tomto programu lze programovat jednotlivé kroky, které má robot vykonávat. Lze zde vytvořit až 256 pozic a každá pozice může mít až 7 kroků. V každém kroku se může u robota ovládat motory číslované od 0 do 25. Program také umožňuje přímou manipulaci s motorky pomocí uchopení kolečka představujícího motor.



V levém sloupci v tomto obrázku je pozice, která je z programu načtená. V pravém sloupci je aktuální pozice všech motorů. V tomto sloupci můžou vypínat motory, které ručně nastavím a po zapnutí odečtu pozici. Uprostřed je zmíněné kolečko pro přímou manipulaci s motorkem

Program pracuje s připojeným robotem v reálném čase.

Při práci v tomto programu je velice náročné udržet neustále spojení s robotem, program se totiž často odpojuje. Při práci program neustále komunikuje s robotem po sériové lince. Je možné pracovat jak s programem, tak s jakýmkoliv programem který je uložený v počítači. Při práci s programem, který je nahraný v řídicí jednotce, může program zkoušet více pozic přímo za sebou, při práci s programem, který není v jednotce nahraný, lze zkoušet pouze kroky, ale nikoli jejich navazování.

1.2.3. RoboPlus Task

```

RoboPlus Task - Untitled*
File(F) Edit(E) Program(P) Tool(T) Help(H)
Controller: CM-510 Port: COM8
1  START PROGRAM
2  {
3      Motion Page = 1
4      WAIT WHILE ( Motion Status == TRUE )
5      ENDLESS LOOP
6      {
7      AAA :
8          IF ( Button == U )
9              // auto je pryč
10         {
11         bezkostky :
12             ID[30]: Goal position = 200
13             // chůze bez kostky
14         BBB :
15             IF ( PORT[5] >= 100 )
16         r
    
```

Sejmutá obrazovka RoboPlus Task.

V tomto programu lze tvořit základní veškerou logiku robota. Program funguje na principech programovacího jazyku C++. Umí základní příkazy typu IF, ELSE, ELSE IF, konečnou a nekonečnou smyčku, smyčku s určitým počtem opakování. Příkazy JMP.

Tento program je asi nejdůležitější z celé řady programů robo plus. Lze v něm nastavit teplotní limit motorků a otočení motorků, případně i vypínání a zapínání. Je schopen číst hodnotu všech připojených senzorů a na základě nich například spouštět jednotlivé kroky, vytvořené v motion editoru, nebo natočit některé motory, případně vyslat signál přes zařízení určené ke komunikaci s jinými roboty (nic takového jsem nepoužil).

Program nefunguje přímo. Po naprogramování se program nahraje do jednotky pro zjištění funkčnosti programu. Je většinou potřeba spustit robota a odzkoušet ho. Pokud do programu umístím správné příkazy, můžu při testování odečítat například hodnoty senzorů, teplotu motorů nebo pozici motorů přes robot terminál. Spouštěcí tlačítko robot terminálu je přímo umístěno a RoboPlus Task.

1.3. Programy

Robot by se bez programů neobešel. Důležité pro něj jsou programy dělané v RoboPlus Motion a Task. Nezbytný byl i program, kterým jsem testoval citlivost senzorů.

1.3.1. Program pro testování citlivosti senzorů.

```
START PROGRAM
{
AAA :
  x = ✓ PORT[5]
  y = ✓ PORT[6]
  Print with Line = x
  Print with Line = y
  JUMP AAA
}
```

Tento program jen vypisuje hodnotu senzorů do obrazovky robot terminálu.

1.3.2. Nastavení v RoboPlus Manager

Nastavení nebylo náročné, stačilo jen přečíslovat motory. Dal jsem vždy příkaz pro rozsvícení aktuálního motoru a podle návodu jsem jen nastavil správné ID. Zde jsem si také zjistil osy pro AX-S20

1.3.3. Program v RoboPlus Motion

Vytvořit kroky nebylo náročné z hlediska programování samotného, ale velkým problémem tu byla rovnováha. Musel jsem vyřešit, jak udržím robota při vypnutých motorech ve správné poloze.

Příklad programu:

Strana 1:

Motor	Krok 0.	Krok 1.	Krok 2.	Krok 3.	Krok 4.	Krok. 5	Krok 6.
1.	335	-	-	-	-	-	-
2.	698	-	-	-	-	-	-
3.	306	-	-	-	-	-	-
4.	720	-	-	-	-	-	-
5.	284	-	-	-	-	-	-
6.	741	-	-	-	-	-	-
7.	362	-	-	-	-	-	-
8.	662	-	-	-	-	-	-
9.	512	-	-	-	-	-	-
10.	512	-	-	-	-	-	-
11.	412	-	-	-	-	-	-
12.	612	-	-	-	-	-	-
13.	372	-	-	-	-	-	-
14.	662	-	-	-	-	-	-
15.	582	-	-	-	-	-	-
16.	432	-	-	-	-	-	-
17.	512	-	-	-	-	-	-
18.	512	-	-	-	-	-	-

Strana 2:

Motor	Krok 0.	Krok 1.	Krok 2.	Krok 3.	Krok 4.	Krok. 5	Krok 6.
1.	335	335	335	-	-	-	-
2.	698	698	698	-	-	-	-
3.	309	309	309	-	-	-	-
4.	720	720	720	-	-	-	-
5.	284	284	284	-	-	-	-
6.	741	741	741	-	-	-	-
7.	362	362	362	-	-	-	-
8.	662	662	662	-	-	-	-
9.	512	512	512	-	-	-	-
10.	512	512	512	-	-	-	-
11.	412	412	412	-	-	-	-
12.	612	642	642	-	-	-	-
13.	372	382	332	-	-	-	-
14.	662	682	682	-	-	-	-
15.	582	582	622	-	-	-	-
16.	432	442	442	-	-	-	-
17.	532	532	532	-	-	-	-
18.	532	532	532	-	-	-	-

1.3.4. Program v RoboPlus task

```
version 2.03
platform bioloid2
o main
o begin
o load param_dest:motion:24 param_src:dec_num:1
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:var:a param_src:dec_num:1
o label laddr_src:spust
o if param_src:cm:32 lop:== param_src:button_num:8 rop:then
o begin
o label laddr_src:opak
o load param_dest:motion:24 param_src:dec_num:1
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o if param_src:s1:102:26 lop:> param_src:dec_num:50 rop:then
o begin
```

```

o // param_text:je v mezeře
o label laddr_src:mez
o if param_src:s1:103:26 lop:> param_src:dec_num:200 rop:then
o begin
o jump laddr_dest:A
o end
o else
o begin
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o jump laddr_dest:mez
o end
o end
o else
o begin
o label laddr_src:A
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o if param_src:s1:103:26 lop:< param_src:dec_num:30 rop:then
o begin
o // param_text:přešel první stěnu
o label laddr_src:B
o load param_dest:motion:24 param_src:dec_num:7
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:motion:24 param_src:dec_num:7
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o if param_src:s1:103:27 lop:< param_src:dec_num:10 rop:&&
param_src:s1:103:29 lop:< param_src:dec_num:10 rop:then
o begin
o // param_text:otočil se o 90
o label laddr_src:C
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o if param_src:s1:103:26 lop:< param_src:dec_num:30 rop:then
o begin
o // param_text:je daleko od stěny
o load param_dest:motion:24 param_src:dec_num:23
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o jump laddr_dest:C
o end
o elseif param_src:s1:103:26 lop:< param_src:dec_num:10 rop:then
o begin
o // param_text:přešel druhou stěnu
o label laddr_src:D
o load param_dest:motion:24 param_src:dec_num:7
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:motion:24 param_src:dec_num:7
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o if param_src:s1:103:27 lop:< param_src:dec_num:10 rop:&&
param_src:s1:103:29 lop:< param_src:dec_num:10 rop:then
o begin
o // param_text:otočil se o 90
o label laddr_src:E

```

```

o if param_src:s1:102:27 lop:> param_src:dec_num:1 rop:||
param_src:s1:103:27 lop:> param_src:dec_num:1 rop:then
o begin
o // param_text:vidí auto
o label laddr_src:F
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o if param_src:s1:102:27 lop:> param_src:dec_num:60 rop:||
param_src:s1:103:27 lop:> param_src:dec_num:60 rop:then
o begin
o // param_text:Je u auta
o if param_src:var:a lop:== param_src:dec_num:1 rop:then
o begin
o load param_dest:var:a param_src:dec_num:0
o // param_text:kostka je na autě
o load param_dest:motion:24 param_src:dec_num:34
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o if param_src:s1:102:27 lop:> param_src:dec_num:50 rop:||
param_src:s1:103:27 lop:> param_src:dec_num:50 rop:then
o begin
o label laddr_src:G
o if param_src:s1:102:27 lop:< param_src:dec_num:50 rop:||
param_src:s1:103:27 lop:< param_src:dec_num:50 rop:then
o begin
o // param_text:auto odjelo
o jump laddr_dest:opak
o end
o else
o begin
o jump laddr_dest:G
o end
o end
o else
o begin
o label laddr_src:H
o if param_src:s1:102:27 lop:< param_src:dec_num:50 rop:||
param_src:s1:103:27 lop:< param_src:dec_num:50 rop:then
o begin
o label laddr_src:I
o if param_src:s1:102:27 lop:< param_src:dec_num:50 rop:||
param_src:s1:103:27 lop:< param_src:dec_num:50 rop:then
o begin
o // param_text:auto odjelo
o jump laddr_dest:opak
o end
o else
o begin
o jump laddr_dest:I
o end
o end
o else
o begin
o jump laddr_dest:H
o end
o end
o end
o else

```

```

o begin
o load param_dest:var:a param_src:dec_num:1
o // param_text:kostu má humanoid
o load param_dest:motion:24 param_src:dec_num:36
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o if param_src:s1:102:27 lop:> param_src:dec_num:50 rop:||
param_src:s1:103:27 lop:> param_src:dec_num:50 rop:then
o begin
o label laddr_src:J
o if param_src:s1:102:27 lop:< param_src:dec_num:50 rop:||
param_src:s1:103:27 lop:< param_src:dec_num:50 rop:then
o begin
o // param_text:auto odjelo
o jump laddr_dest:opak
o end
o else
o begin
o jump laddr_dest:J
o end
o end
o else
o begin
o label laddr_src:K
o if param_src:s1:102:27 lop:< param_src:dec_num:50 rop:||
param_src:s1:103:27 lop:< param_src:dec_num:50 rop:then
o begin
o label laddr_src:L
o if param_src:s1:102:27 lop:< param_src:dec_num:50 rop:||
param_src:s1:103:27 lop:< param_src:dec_num:50 rop:then
o begin
o // param_text:auto odjelo
o jump laddr_dest:opak
o end
o else
o begin
o jump laddr_dest:L
o end
o end
o else
o begin
o jump laddr_dest:K
o end
o end
o end
o end
o else
o begin
o jump laddr_dest:F
o end
o end
o else
o begin
o jump laddr_dest:E
o end
-
o end
o else
o begin
o jump laddr_dest:D

```

```

o end
o end
o else
o begin
o jump laddr_dest:C
o end
o end
o else
o begin
o jump laddr_dest:B
o end
o end
o elseif param_src:s1:103:26 lop:< param_src:dec_num:30 rop:&&
param_src:s1:103:26 lop:> param_src:dec_num:150 rop:then
o begin
o // param_text:je daleko od stěny
o load param_dest:motion:24 param_src:dec_num:23
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o jump laddr_dest:A
o end
o else
o begin
o load param_dest:motion:24 param_src:dec_num:2
o wait param_src:motion:25 lop:== param_src:bool_num:1 rop:then
o end
o end
o end
o else
o begin
o jump laddr_dest:spust
o end
o end
o end
□

```

Tištěný přehledný program viz příloha č 2.

2. závěr

Všechny programy jsou plně funkční a odzkoušené. Robot má problémy s rovnováhou a přepadáváním dopředu. Při práci bylo mnoho obtíží ale podařilo se je vyřešit.

3. Seznam použité literatury

- [1] AX-12(english) – návod pro práci s mototry
- [2] AX-S1(english) – návod pro práci se senzory
- [3] BioloidUsersGuide(english) – práce s jednotkou CM-5 a CM – 510