

ROBOTI

VE ŠKOLE PRO PRAKTICKOU VÝUKU, MOTIVACI I ZÁBAVU

CZ.1.07/1.1.24/01.0066

MANIPULÁTORY A VOZÍKY

Ing. Michal ŘEPKA, Ph.D.

Střední škola elektrotechnická, Ostrava, Na Jízdárně 30, příspěvková organizace

2014



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

POKYNY KE STUDIU:



ČAS KE STUDIU

Čas potřebný k prostudování látky. Čas je pouze orientační a slouží jako hrubé vodítko pro rozvržení studia kapitoly.



CÍL

Cíle, kterých lze dosáhnout prostudováním kapitoly – konkrétní dovednosti, znalosti.



POJMY K ZAPAMATOVÁNÍ

Pojmy, které si je potřeba zapamatovat.



VÝKLAD

Teoretický výklad studované látky, zavedení nových pojmů a jejich vysvětlení.



ŘEŠENÉ PŘÍKLADY

Podrobný postup při řešení příkladů.



SHRNUTÍ POJMŮ

Zopakování hlavních pojmů.



OTÁZKY

Několik teoretických otázek pro ověření zvládnutí kapitoly.



PRAKTICKÉ ÚLOHY

Několik praktických příkladů pro ověření zvládnutí kapitoly.

OBSAH:

1. Základní seznámení s robotickým hardware.....	4
1.1. Pohonné jednotky Dynamixel.....	6
1.1.1. Konektory pohonu AX.....	8
1.1.2. Řídicí tabulka pohonu AX.....	9
1.1.3. Nastavení módů pohonu AX.....	14
1.1.4. Technické parametry pohonu AX.....	17
1.1.5. Mechanické propojení pohonů AX.....	18
1.2. Senzorický modul AX-S1.....	20
1.2.1. Řídicí tabulka AX-S1.....	22
1.2.2. Příklady použití AX-S1.....	28
1.3. Řídicí jednotka CM-5 / CM-510 / CM-530.....	30
1.3.1. CM-5.....	33
1.3.2. CM-510.....	35
1.4. Propojení AX-12/AX-S1 s PC.....	39
1.4.1. Pomocí řídicí jednotky CM-5/510/530.....	39
2. Základní úlohy – RoboPlus Manager a RoboPlus Task.....	42
2.1. Vložení instrukce.....	48
2.2. Natočení pohonu AX do určené pozice.....	50
2.3. Natáčení pohonu podle stisknutého tlačítka.....	53
3. Manipulátory / Vozíky.....	57
3.1. Čtyř-kolový vozík.....	58
3.2. Čtyř-kolový vozík s manipulačním ramenem.....	68
3.3. Manipulátor se třemi stupni volnosti.....	81
Použitá literatura.....	99

1. Základní seznámení s robotickým hardware



ČAS KE STUDIU: 10 minut



CÍL

Prostudováním této kapitoly získá čtenář přehled o základních komponentách robotického systému BIOLOID Beginner a Comprehensive.



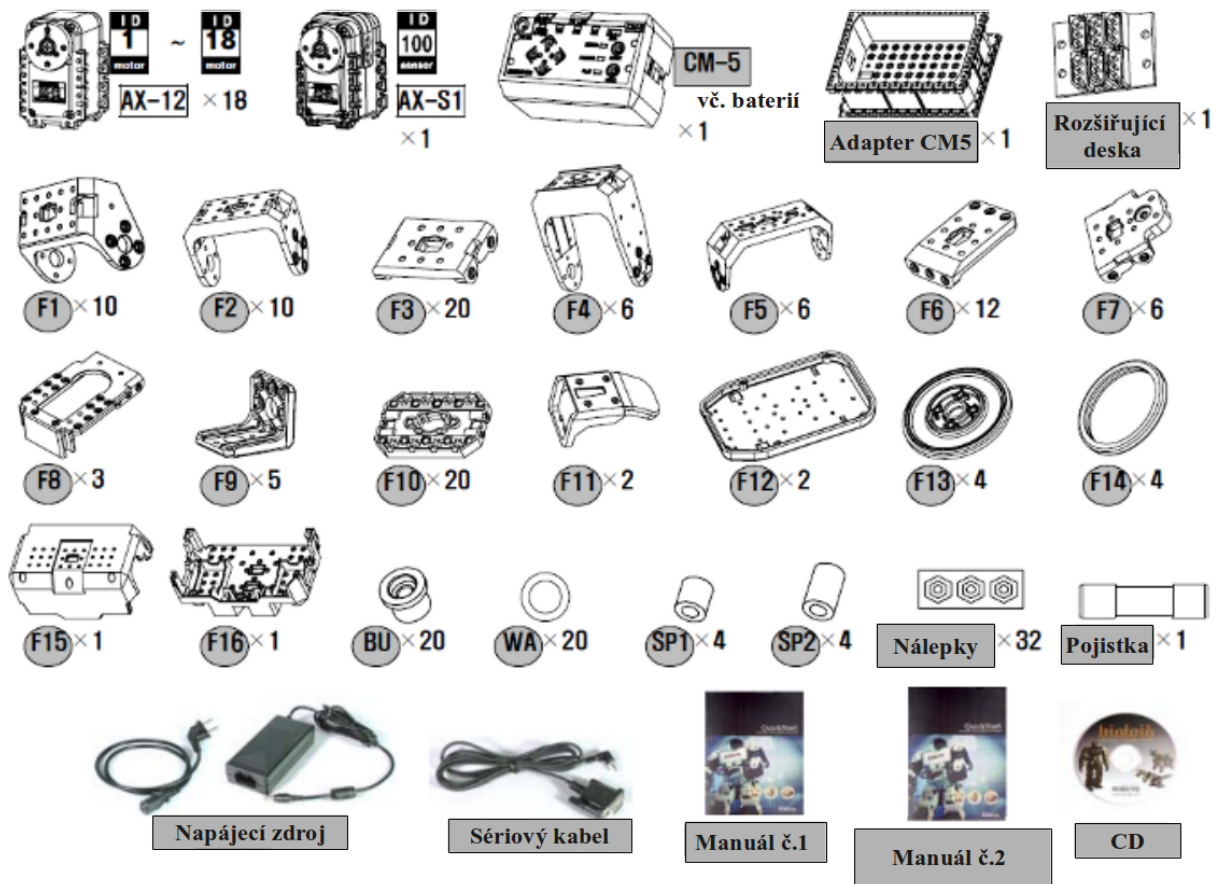
VÝKLAD

Robotický systém BIOLOID od korejské firmy Robotis je složen z několika základních částí. Jedná se o pohonné jednotky, senzorické moduly, řídicí jednotky a mechanické komponenty.

V tomto projektu budeme pracovat s robotickým systémem BIOLOID, který se dodává pod označením BIOLOID Beginner a BIOLOID Comprehensive. Tyto dva systémy jsou téměř identické, liší se pouze typem a počtem některých komponent. Na obrázku č. 1.1 je zobrazen seznam komponent, které jsou součástí systému BIOLOID Comprehensive. BIOLOID Beginner obsahuje podobnou sadu mechanických komponent, avšak v menším počtu. Oba systémy BIOLOID navíc obsahují sadu spojovacího materiálu (šroubky a matice) a sadu propojovacích tří-žilových kabelů.

Tato kapitola je rozdělena do několika podkapitol. Část věnována pohonným jednotkám je uvedena v kapitole 1.1., senzorické moduly jsou popsány v kapitole 1.2. Řídicí jednotky jsou popsány v kapitole 1.3. Kapitola 1.4. popisuje možnosti propojení pohonů s osobním počítačem.

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ



Obr. 1.1 – Seznam dílů BIOLOID Comprehensive

1.1. Pohonné jednotky Dynamixel



ČAS KE STUDIU: 60 minut



CÍL

Prostudováním této kapitoly získá čtenář přehled o pohonech firmy Robotis. Dále se v této kapitole seznámí se základními vlastnostmi pohonu AX-12.



VÝKLAD

Firma Robotis používá pohonné jednotky s označením Dynamixel, jež vyrábí v několika typových řadách, které jsou označeny dvěma písmeny. Vyráběné řady pohonných jednotek Dynamixel:

- DX (zastaralý typ, již není v prodeji)
- AX (komunikace half-duplex RS 232 TTL, plastová převodovka)
- RX (komunikace RS 485, kovová převodovka)
- EX (komunikace RS 485, kovová převodovka)
- MX (komunikace RS 485 nebo half-duplex RS 232 TTL, kovová převodovka)

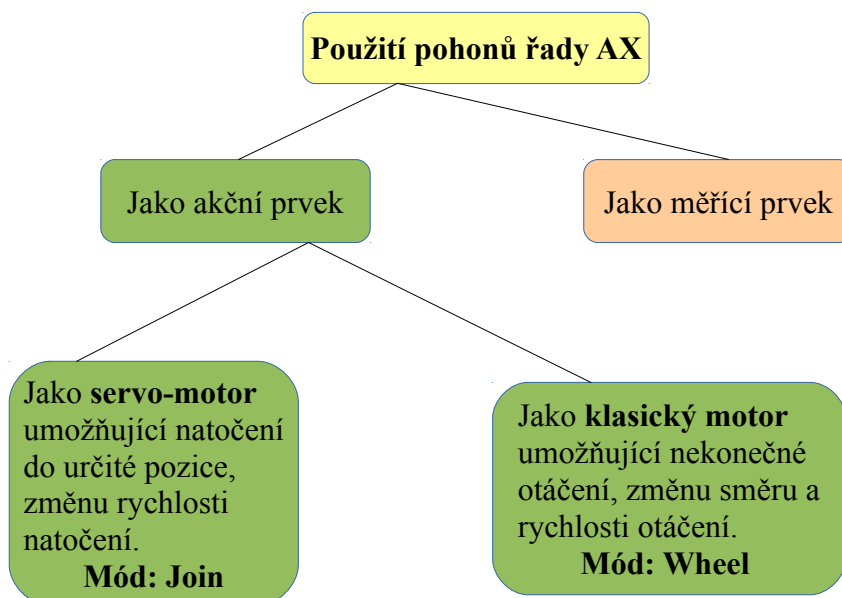
Robotický systém BIOLOID využívá pohonné jednotky řady **AX** a to zejména AX-12, AX-12+ , AX-12A. Tyto tři modely pohonů jsou navzájem kompatibilní mechanicky, elektronicky a softwarově. Každý takovýto pohon je složen z několika částí:

- *převodovka*
- *stejnoseměrný motor* (s komutátorem a permanentním magnetem ve statoru)
- *řídící jednotka + driver* (elektronika umožňující změnu napětí pomocí procesoru)
- *senzory*
- *komunikační rozhraní*

Tyto pohony lze používat několika způsoby, jak demonstruje následující obrázek č.1.1.1.

Pohony řady AX, lze nastavit do **módu Join**, ve kterém pracuje jako servo motor. Tj. V tomto módu se rotor natočí zvolenou rychlostí do zvolené polohy. Při nastavení pohonu do **módu Wheel**, se pohon chová jako stejnosměrný motor. V obou módech lze samozřejmě nastavit rychlost otáčení a další parametry (detailní seznam viz. níže kapitola 1.1.2).

Využití pohonu řady AX, jako měřicího prvku, je možné díky vestavěným senzorům. Jeden z těchto senzorů zjišťuje informaci o skutečném natočení rotoru. Abychom pohon řady AX mohli použít jako měřicí prvek je důležité nastavit tento pohon do takového režimu, kdy dojde k odpojení napájecího napětí od motoru. Tímto motor přestane klást odpor a rotorem bude možné pohybovat vnějším působením (ne však úplně volně). Aby to bylo snadnější, měl by být k rotoru mechanicky připevněna např. páka, díky čemuž bude možné překonat třecí síly motoru a také mechanický převod v převodovce.



Obr. 1.1.1 – Použití pohonů řady AX-12 viz. Kapitola 1.1.3

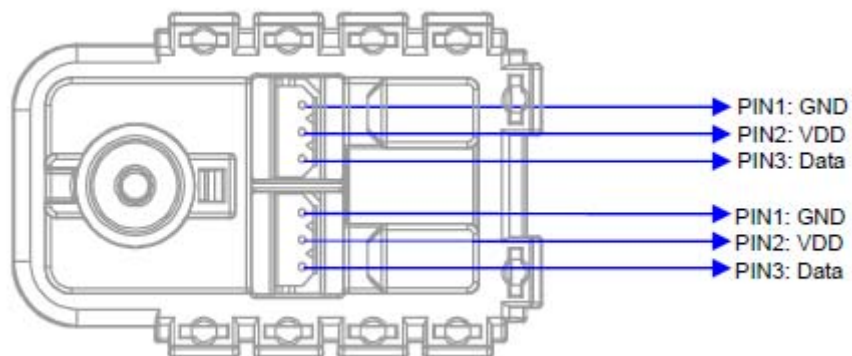
Pozn. autora: pokud se Vám bude zdát, že při vnějším otáčením rotorem, se rotor neotáčí plynule, může to být např. způsobeno indukovaným opačného napětí, které vyvolává pohyb motoru, avšak opačným směrem než je působení vnější síly. Což u může vyvolat dojem, poškozené převodovky.

1.1.1. Konektory pohonu AX

Na pohonu jsou umístěny dva 3pinové konektory. Tyto dva konektory jsou paralelně

propojeny. A to z toho důvodu, aby se pohony daly pohodlně navzájem propojovat, viz. Obrázek 1.1.2. Typ konektoru vyrábí firma Molex s označením 22-03-5035 (female).

- PIN 1: GND – referenční zem
- PIN 2: VDD – napájecí napětí
- PIN 3: Data – přenos sériových dat



Obr. 1.1.2 – Označení pinů AX-12

1.1.2. Řídicí tabulka pohonu AX

	Adresa		Název	Přístup	Počáteční hodnota	
	Dec	Hex			Dec	Hex
Umístěno v paměti EEPROM	0	0	Číslo modelu (L Byte)	RD	12	C
	1	1	Číslo modelu (H Byte)	RD	0	0
	2	2	Verze firmware	RD	?	
	3	3	ID	RD, WR	1	1
	4	4	Baud Rate – komunikační rychlost	RD, WR	1	1
	5	5	Return Delay Time	RD, WR	250	FA
	6	6	CW Angle Limit(L Byte)	RD, WR	0	0
	7	7	CW Angle Limit(H Byte)	RD, WR	0	0
	8	8	CCW Angle Limit(L)	RD, WR	255	FF
	9	9	CCW Angle Limit(H)	RD, WR	3	3
	10	A	(Rezervováno)	-	-	
	11	B	Nejvyšší dovolená teplota	RD, WR	85	55
	12	C	Nejnižší dovolené napětí	RD, WR	60	3C
	13	D	Nejvyšší dovolené napětí	RD, WR	190	BE
	14	E	Max Torque(L)	RD, WR	255	FF
	15	F	Max Torque(H)	RD, WR	3	3
	16	10	Status Return Level	RD, WR	2	2
	17	11	Alarm LED	RD, WR	4	4
	18	12	Alarm Shutdown	RD, WR	4	4
	19	13	(Rezervováno)	-	-	
	20	14	Down Calibration(L)	RD	?	
	21	15	Down Calibration(H)	RD	?	
	22	16	Up Calibration(L)	RD	?	
23	17	Up Calibration(H)	RD	?		
paměť RAM	24	18	Torque Enable	RD, WR	0	0
	25	19	LED	RD, WR	0	0
	26	1A	CW Compliance Margin	RD, WR	0	0
	27	1B	CCW Compliance Margin	RD, WR	0	0

Pozn.

RD, WR

- znamená možnost čtení nebo zápisu. Položku lze přepsat.

RD

- znamená pouze pro čtení. Položku nelze přepsat.

	Adresa		Název	Přístup	Počáteční hodnota	
	Dec	Hex			Dec	Hex
Umístěno v paměti RAM	28	1C	CW Compliance Slope	RD, WR	32	20
	29	1D	CCW Compliance Slope	RD, WR	32	20
	30	1E	Goal Position(L)	RD, WR	Hodnota z adresy 36	
	31	1F	Goal Position(H)	RD, WR	Hodnota z adresy 37	
	32	20	Moving Speed(L)	RD, WR	0	0
	33	21	Moving Speed(H)	RD, WR	0	0
	34	22	Torque Limit(L)	RD, WR	Hodnota z adresy 14	
	35	23	Torque Limit(H)	RD, WR	Hodnota z adresy 15	
	36	24	Present Position(L)	RD	?	
	37	25	Present Position(H)	RD	?	
	38	26	Present Speed(L)	RD	?	
	39	27	Present Speed(H)	RD	?	
	40	28	Present Load(L)	RD	?	
	41	29	Present Load(H)	RD	?	
	42	2A	Present Voltage	RD	?	
	43	2B	Present Temperature	RD	?	
	44	2C	Registered Instruction	RD, WR	0	0
	45	2D	(Rezervováno)		-	
	46	2E	Moving	RD	0	0
47	2F	Lock	RD, WR	0	0	
48	30	Punch(L)	RD, WR	32	20	
49	31	Punch(H)	RD, WR	0	0	

Pozn.

RD, WR

- znamená možnost čtení nebo zápisu. Položku lze přepsat.

RD

- znamená pouze pro čtení. Položku nelze přepsat.

Adresa 0x00,0x01 Model Number.
Pro AX-12, je tato hodnota 0X000C hex (12 dec).

Adresa 0x02 Firmware Version.

Adresa 0x03 ID.

The unique ID number assigned to each Dynamixel actuators for identifying them. Different IDs are required for each Dynamixel actuators that are on the same network.

Adresa 0x04 Baud Rate.

Hodnota na této adrese určuje komunikační rychlost. Komunikační rychlost se vypočte dle následujícího vzorce:

$$\text{Komunikační rychlost (BPS – bitů za sekundu)} = 2000000 / (\text{Adresa4} + 1)$$

Tabulka hodnot pro některé hlavní komunikační rychlosti

Adresa4 (dec)	Adresa4 (hex)	Nastavená rychlost	Cílová komunikační rychlost	Chyba v rychlosti v závislosti k normě
1	1	1000000,0	1000000,0	0 %
3	3	500000,0	500000,0	0 %
4	4	400000,0	400000,0	0 %
7	7	250000,0	250000,0	0 %
9	9	200000,0	200000,0	0 %
16	10	117647,1	115200	-2,124 %
34	22	57142,9	57600	0,794 %
103	67	19230,8	19200	-0,16 %
207	CF	9615,4	9600	-0,16 %

Pozn. Komunikační norma pro UART stanovuje možnou chybu v komunikačních rychlostech max. 3%, proto všechny tyto rychlosti vyhovují dané normě. Veškeré pohony AX, jsou od výroby nastavené na hodnotu 1, tj. Na komunikační rychlost 1 000 000 bps.

Adresa 0x05 Return Delay Time.

Potřebný čas pro návrat Status Packet-u po vyslání Instrukčního packet-u. Tento čas je určen hodnotou na této adrese x 2 μs (mikro-sekundy).

Adresa 0x06,0x07,0x08,0x09 Operating Angle Limit. (Úhlové omezení)

Hodnoty na těchto adresách určují úhlové omezení pohonu. Cílová pozice (Goal Position) by měla být v rozsahu: CW Angle Limit <= Goal Position <= CCW Angle Limit. Pokud cílová pozice bude mimo nastavený rozsah, bude nastaven stavový bit Angle Limit Error.

Adresa 0x0B the Highest Limit Temperature. (Limit nejvyšší teploty)

Nejvyšší dovolená pracovní teplota. Jestliže interní teplota bude větší než nejvyšší možná, tak bude nastaven (do log.1) stavový bit Over Heating Error Bit (Bit 2 of the Status Packet)a dále bude nastaven alarm na adrese 17 a 18. Hodnoty teplot jsou uváděny v Celsiových stupních.

Adresa 0x0C,0x0D the Lowest (Highest) Limit Voltage.

The Nejnižší a nejvyšší povolený napětový limit napájecího napětí. If the present voltage (Address 42) is out of the specified range, a Voltage Range Error Bit (Bit 0 of the Status Packet) will return the value 1, and an alarm will be set by Address 17, 18. Hodnoty na těchto adresách jsou násobeny desíti. Např., jestliže adresa 0x0C obsahuje hodnotu 80, pak nejnižší napětový limit je nastaven na 8V.

Adresa 0x0E,0x0F, 0x22,0x23 Max Torque.

Maximální výstupní moment pro pohon. Pokud je tato hodnota nastavena na 0, pohon Dynamixel vstoupí do Free Run režimu. Existují dvě paměťová místa, kde je definován tento maximální limit momentu, v EEPROM (adresa 0x0E, 0x0F) a v RAM (Address 0x22, 0x23). Když je zapnuto napájení, je hodnota maximálního limitu momentu definovaná v EEPROM zkopírována do umístění v paměti RAM. Moment pohonu Dynamixel je omezen hodnotami umístěnými v paměti RAM (Adresa 0x22, 0x23).

Adresa 0X10 Status Return Level.

Určuje, zda pohon Dynamixel vrátí stavový Packet po obdržení Instrukčního Packet-u.

Adresa 0X11 Alarm LED.

Pokud odpovídající bit je nastaven na 1, LED bliká, když dojde k chybě.

Adresa 0X12 Alarm Shutdown.

Pokud odpovídající bit je nastaven na 1, točivý moment pohonu bude vypnut, pokud dojde k chybě.

Adresa 0x14~0x17 Calibration.

Údaje používané na kompenzaci rozdílů mezi potenciometry používaných v Dynamixel jednotkách. Uživatel nemůže změnit tato data.

Adresa 0x18 Torque Enable.

Při prvním zapnutí, pohon Dynamixel vstoupí do stavu Free Run Torque (nulový moment). Nastavení hodnoty v adrese 0x18 na 1 umožní točivý moment.

Adresa 0x19 LED.

LED se rozsvítí při nastavení na 1 a vypne pokud je nastaveno na 0.

Adresa 0x1A~0x1D Compliance Margin and Slope.

Parametry pohonu Dynamixel definující okraje a sklon pohybové charakteristiky pohonu. Tato funkce může být využita pro pohlcení nárazů na výstupní hřídeli.

Adresa 0X1E,0x1F Goal Position (Cílová pozice)

Požadovaná úhlová poloha pro pohon Dynamixel. Nastavení této hodnoty na 0x3ff rozpohybuje výstupní hřídel do polohy na 300 °.

Adresa 0x20,0x21 Moving Speed.

Nastaví úhlovou rychlost pohybující se hřídele do cílové pozice. Nastavení této hodnoty na maximální hodnotu 0x3ff rozpožhybuje hřídel s úhlovou rychlostí 114 rpm, za předpokladu, že je dostatek energie (nejnižší rychlost je, když je tato hodnota nastavena na 1. Když je nastaveno na hodnotu 0, je rychlost největší. Maximální rychlost závisí na napájecím napětí.)

Adresa 0x24,0x25 Present Position.

Aktuální úhlová poloha hřídele Dynamixel pohonu.

Adresa 0x26,0x27 Present Speed.

Aktuální úhlová rychlost hřídele Dynamixel pohonu.

Adresa 0x28,0x29 Present Load.

Velikost provozního zatížení na pohon Dynamixel. Bit 10 určuje směr zatížení.

Adresa 0x2A Present Voltage.

Napětí, kterým je napájen pohon Dynamixel. Hodnota je 10 krát větší než skutečné napětí. Například, 10V je reprezentováno hodnotou 100 dec (0x64 hex).

Adresa 0x2B Present Temperature.

Vnitřní teplota Dynamixel pohonu ve stupních Celsia.

Adresa 0x2C Registered Instruction.

Nastaveno na 1, když je instrukce přiřazena REG_WRITE příkazem. Nastaveno na 0 po dokončení přiřazeného pokynu akčním příkazem.

Adresa 0x2E Moving.

Nastaveno na 1, když je pohon Dynamixel v pohybu.

Adresa 0x2F Lock.

Pokud je na tuto adresu zapsána hodnota 1, tak bude možné zapisovat pouze na adresy 0x18 až 0x23. Na ostatní adresy zapisovat nebude možné dokud se neodpojí napájení pohonu.

Adresa 0x30,0x31 Punch.

Hodnota na této adrese reprezentuje minimální proud do motoru během provozu. Počáteční hodnota po spuštění je nastavena na 0x20 a její maximální hodnota je 0x3ff.

1.1.3. Nastavení módů pohonu AX

Jak již bylo výše uvedeno, s pohonem AX, lze pracovat různým způsobem. Přehledně tyto možnosti ilustruje obrázek č.1.1.1. Z tohoto důvodu, jsou v této kapitole uvedeny jednotlivé příklady, které ukazují nastavení důležitých registrů pro danou činnost pohonu AX.



ŘEŠENÉ PŘÍKLADY – Pohon AX-12 jako akční prvek - mód Wheel

Nastavení režimu wheel je dosaženo zápisem čísla 0 na adresy: 6,7,8 a 9 (dec). Pohon se roztočí po zápisu čísla na adresu: 32. Principiálně jde o to, že na adresách 6 až 9 je uložena informace o minimální/maximální výchylce rotoru pohonu v Join módu. Tj. pokud zapíšeme číslo 0 na tyto adresy, pak v podstatě vypneme toto omezení a tím tak přecházíme do Wheel módu.

Ukázka programu č.1.1 v software RoboPlus Task

```

1  START PROGRAM
2  {
3      ID[2]: ADDR[6(w)] = 0
4      ID[2]: ADDR[8(w)] = 0
5  ENDLESS LOOP
6  {
7      IF ( Button == L )
8      {
9          ID[2]: Moving speed = CCW:0
10     }
11
12     IF ( Button == D )
13     {
14         ID[2]: Moving speed = CCW:512
15     }
16 }
17 }

```

Zápis čísla nula, na adresu 6 až 9. Tímto dojde k přepnutí pohonu do Wheel módu.

Po stisku tlačítka L se pohon roztočí zastaví.

Po stisku tlačítka D se pohon roztočí na 50% svých maximálních otáček.



ŘEŠENÉ PŘÍKLADY – Pohon AX-12 jako akční prvek - mód Join

Příklad nastavení pohonu AX-12 (s ID 2) do režimu Join reprezentují řádky č.4 a č.5. Tento režim je nastaven tím, že na adresu 6 bude zapsáno číslo z rozsahu 0 až 1023 a na adresu 8 bude zapsáno číslo z rozsahu 1 až 1023.

Pozor! Bude-li uložená hodnota na adrese 6 větší než uložená hodnota na adrese 8, tak pohon nebude reagovat. Resp. rotor se nebude otáčet a na první pohled se pohon bude jevit jako by byl poškozen.

Ukázka programu č.1.2 v software RoboPlus Task

```

1  START PROGRAM
2  {
3  // Nastaveni pohonu AX-12 do režimu Join
4  ID[2]: ADDR[6(w)] = 0
5  ID[2]: ADDR[8(w)] = 1023
6  // Nastaveni rychlosti
7  ID[2]: Moving speed = 500
8
9  ENDLESS LOOP
10 {
11   IF ( Button == L )
12   {
13     ID[2]: Goal position = 70
14   }
15
16   IF ( Button == R )
17   {
18     ID[2]: Goal position = 900
19   }
20 }
21 }

```

Stisknutím tlačítka L se rotor pohonu přesune do pozice 70.

Stisknutím tlačítka R se rotor pohonu přesune do pozice 900.



ŘEŠENÉ PŘÍKLADY – AX jako měřící prvek

Aby bylo možné použít pohon „pouze“ jako měřící prvek, stačí zapsat číslo 0 na adresu 24 (dec). Následující ukázka kódu z RoboPlus Task, ukazuje pohon s ID 3 nastaven jako měřící prvek. Pohon s ID 2 je nastaven jako akční prvek.

Ukázka programu č.1.3 v software RoboPlus Task

```
ID[3]: Torque Enable = FALSE  
ID[2]: Torque Enable = TRUE
```

Nastavením pohonu ID 3 jako měřícího prvku je myšleno, že pohon nedodává žádnou energii do motoru. V tomto režimu je primárně využíván snímač aktuální pozice rotoru motoru. Informace z tohoto snímače je dostupná na adrese 36 a 37 (dec).

1.1.4. Technické parametry pohonu AX

AX-12+ je vylepšená verze modelu **AX-12**. Vylepšením je provedeno v oblasti lepšího designu elektronických obvodů, použitého materiálu. Vylepšena byla také převodovka.

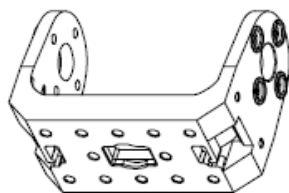
AX-12A je novější verze **AX-12+**. AX-12A má stejné vlastnosti jako AX-12+. U modelu AX-12A byl vylepšen pouze externí design plastového krytu.

- Váha: 53.5g (AX-12/AX-12+), 54.6g (AX-12A)
- Rozměr: 32mm x 50mm x 40mm
- Rozlišení: 0.29°
- Převodový poměr: 254 : 1
- Moment síly: 1.5N.m (při 12.0V, 1.5A)
- Volnoběžné otáčky: 59rpm (při 12V)
- Možnost natočení: (v módu Join 0° ~ 300°)
- Provozní teplota: -5°C ~ +70°C
- Napětí: 9 ~ 12V (Doporučené napětí 11.1V)
- Řídicí signál: Digitální Paket
- Typ komunikačního protokolu: Half duplex Asynchronous Serial Communication (8bit,1stop,No Parity)
- Propojení (fyzická vrstva ISO/OSI modelu): TTL Level Multi Drop (daisy chain type Connector)
- Možný rozsah nastavení ID (0~253)
- Komunikační rychlost: 7343bps ~ 1 Mbps
- Zpětná vazba: Pozice natočení rotoru, Teplota, Zatížení rotoru, Vstupní napětí
- Materiál: Plast

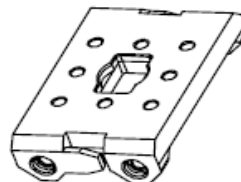
1.1.5. Mechanické propojení pohonů AX

K pohonům řady AX lze připevnit mechanické díly z robotického systému BIOLOID,

které jsou označeny F2 a F3, viz. obrázek č.1.1.3. Montáž mechanického prvku F2, je zobrazen na obrázku č.1.1.4. Montáž mechanického prvku F3 má více možných uchycení, proto je montáž zobrazena na obrázku č.1.1.5 obrázku č.1.1.6.



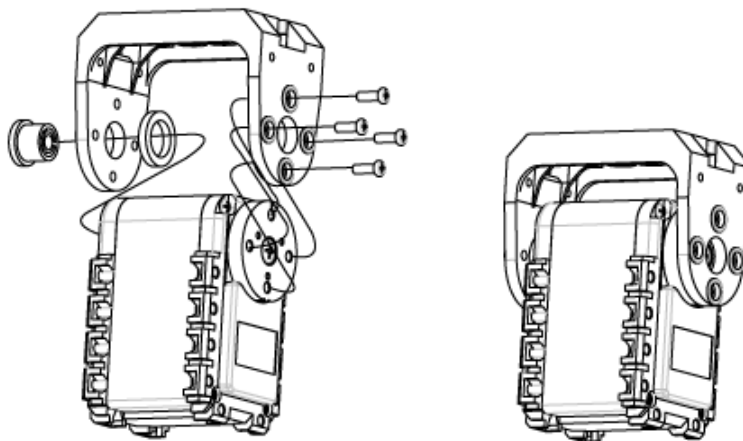
FP04-F2



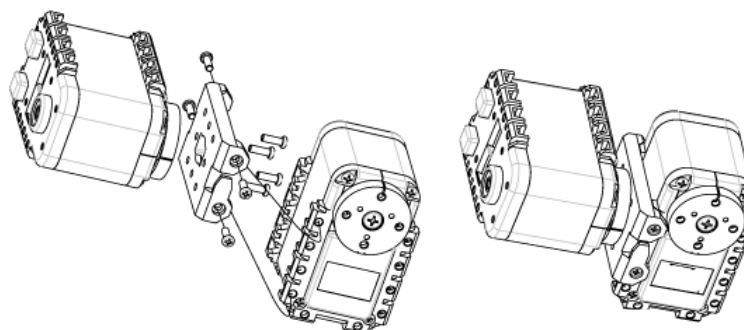
FP04-F3

Obr. 1.1.3 – Označení mechanických dílů, které slouží k mechanickému spojení pohonů

Pozn.: Kromě těchto dvou dílů, lze k pohonu připojit také mechanické díly s označením F1, F4, F5, jež jsou ekvivalentní k mechanickému dílu F2. Dále pak lze použít mechanický díl F6, jež je ekvivalentní k dílu F3.

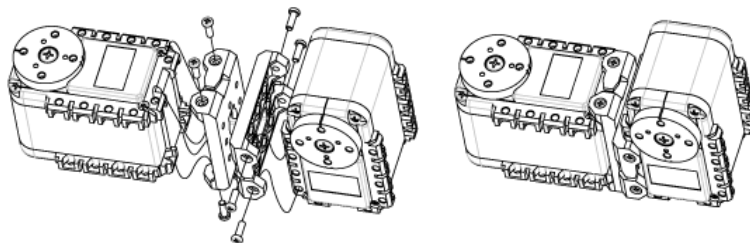


Obr. 1.1.4 – Montáž mechanického prvku F2 k pohonu AX



Obr. 1.1.5 – Montáž mechanického prvku F3 k rotoru pohonu AX

INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ



Obr. 1.1.6 – Montáž mechanického prvku F3 ke spojení dvou pohonů AX

1.2. Senzorický modul AX-S1



ČAS KE STUDIU: 60 minut



CÍL

Prostudováním této kapitoly získá čtenel přehled o vlastnostech senzorického modulu AX-S1.



VÝKLAD

Dynamixel AX-S1 je senzorický modul, který je umístěn ve stejném pouzdře jako pohony řady AX, avšak motor a převodovka již nejsou součástí. Senzorický modul AX-S1 obsahuje snímače pro měření, teploty uvnitř pouzdra, vzdálenosti od překážky, intenzitu osvětlení a přivedené napájecí napětí. Dále tento snímač obsahuje mikrofón a piezo-měnič, pomocí kterého je možné generovat jednoduché melodie.



Obr č.1.2.1 – Senzorický modul AX-S1

Senzorický modul AX-S1 je zcela kompatibilní s pohony řady AX. Proto v této kapitole jsou uvedeny jen informace, které jsou specifické jen pro AX-S1. Například konektory AX-S1 jsou zcela identické s pohony AX, proto detailní informace může čtenář najít v kapitole 1.1.1. Totéž lze říci o mechanickém propojení AX-S1. Pro detailní informace může čtenář použít kapitolu 1.1.5.

Technické parametry modulu AX-S1:

- Váha: 37g
- Rozměr: 32mm x 50mm x 40mm
- Rozlišení: 10 bitů
- Provozní teplota: $-5^{\circ}\text{C} \sim +85^{\circ}\text{C}$
- Napětí: 7 ~ 12V (Doporučené napětí 11.1V)
- Odběr el. proudu: 40 mA
- Řídicí signál: Digitální Paket
- Typ komunikačního protokolu: Half duplex Asynchronous Serial Communication (8bit,1stop,No Parity)
- Propojení (fyzická vrstva ISO/OSI modelu): TTL Level Multi Drop (daisy chain type Connector)
- Možný rozsah nastavení ID (0~253)
- Komunikační rychlost: 7343bps ~ 1 Mbps
- Zpětná vazba: Infračervený senzor, Vnitřní mikrofón, Teplota uvnitř pouzdra, Možnost bezdrátové IR komunikace (Tx/Rx), Vstupní napětí
- Materiál: Plast

Pozn.

U systému Bioloid Premium, již tento senzorický modul AX-S1 není součástí dodávky. U tohoto systému jsou některé tyto senzory dodávány zvlášť a některé jsou součástí řídicí jednotky CM-510/530. Navzdory tomu, lze bez problému využívat senzorický modul AX-S1 s řídicí jednotkou CM-510/530.

1.2.1. Řídící tabulka AX-S1

	Adresa		Název	Přístup	Počáteční hodnota	
	Dec	Hex			Dec	Hex
Umístěno v paměti EEPROM	0	0	Číslo modelu (L Byte)	RD	13	D
	1	1	Číslo modelu (H Byte)	RD	0	0
	2	2	Verze firmware	RD	?	
	3	3	ID	RD, WR	100	64
	4	4	Baud Rate – komunikační rychlost	RD, WR	1	1
	5	5	Return Delay Time	RD, WR	250	FA
	6	6	(Rezervováno)	-	-	-
	7	7	(Rezervováno)	-	-	-
	8	8	(Rezervováno)	-	-	-
	9	9	(Rezervováno)	-	-	-
	10	A	(Rezervováno)	-	-	-
	11	B	Nejvyšší dovolená teplota	RD, WR	100	64
	12	C	Nejnižší dovolené napětí	RD, WR	60	3C
	13	D	Nejvyšší dovolené napětí	RD, WR	190	BE
	14	E	(Rezervováno)	-		
	15	F	(Rezervováno)	-		
	16	10	Status Return Level	RD, WR	2	2
	17	11	(Rezervováno)	-		
	18	12	(Rezervováno)	-		
	19	13	(Rezervováno)	-		
	20	14	Obstacle Detected Compare Value	RD, WR	32	20
	21	15	Light Detected Compare Value	RD, WR	32	20
	22	16	(Rezervováno)	-		?
23	17	(Rezervováno)	-		?	
paměť RAM	24	18	(Rezervováno)	-		
	25	19	(Rezervováno)	-		
	26	1A	Left IR Sensor Data	RD	?	?
	27	1B	Center IR Sensor Data	RD	?	?

Pozn.

RD, WR

- znamená možnost čtení nebo zápisu. Položku lze přepsat.

RD

- znamená pouze pro čtení. Položku nelze přepsat.

	Adresa		Název	Přístup	Počáteční hodnota	
	Dec	Hex			Dec	Hex
Umístěno v paměti RAM	28	1C	Right IR Sensor Data	RD	?	?
	29	1D	Left Luminosity	RD	?	?
	30	1E	Center Luminosity	RD	?	
	31	1F	Right Luminosity	RD	?	
	32	20	Obstacle Detection Flag	RD	?	?
	33	21	Luminosity Detection Flag	RD	?	?
	34	22	(Rezervováno)	-		
	35	23	Sound Data	RD, WR		
	36	24	Sound Data Max Hold	RD, WR	?	
	37	25	Sound Detected Count	RD, WR	?	
	38	26	Sound Detected Time (L)	RD, WR	?	
	39	27	Sound Detected Time (H)	RD, WR	?	
	40	28	Buzzer Index	RD, WR	?	
	41	29	Buzzer Time	RD, WR	?	
	42	2A	Present Voltage	RD	?	
	43	2B	Present Temperature	RD	?	
	44	2C	Registered Instruction	RD, WR	0	0
	45	2D	(Rezervováno)		-	
	46	2E	IR Remocon Arrived	RD	0	0
	47	2F	Lock	RD, WR	0	0
48	30	IR Remocon RX Data 0	RD	?	?	
49	31	IR Remocon RX Data 1	RD	?	?	
50	32	IR Remocon TX Data 0	RD, WR	?	?	
51	33	IR Remocon TX Data 1	RD, WR	?	?	
52	34	Obstacle Detected Compare	RD, WR	?	?	
53	35	Light Detected Compare	RD, WR	?	?	

Pozn.

RD, WR

- znamená možnost čtení nebo zápisu. Položku lze přepsat.

RD

- znamená pouze pro čtení. Položku nelze přepsat.

Adresa 0x00,0x01 Model Number. (Číslo modelu)

Pro AX-12, je tato hodnota 0X000C hex (12 dec).

Adresa 0x02 Firmware Version. (Verze firmware)

Adresa 0x03 ID.

Jedinečný číselný identifikátor ID. Tento identifikátor musí být přiřazen každému pohonu.

Adresa 0x04 Baud Rate. (Komunikační rychlost)

Hodnota na této adrese určuje komunikační rychlost. Komunikační rychlost se vypočte dle následujícího vzorce:

$$\text{Komunikační rychlost (BPS – bitů za sekundu)} = 2000000 / (\text{Adresa4} + 1)$$

Tabulka hodnot pro některé hlavní komunikační rychlosti

Adresa4 (dec)	Adresa4 (hex)	Nastavená rychlost	Cílová komunikační rychlost	Chyba v rychlosti v závislosti k normě
1	1	1000000,0	1000000,0	0 %
3	3	500000,0	500000,0	0 %
4	4	400000,0	400000,0	0 %
7	7	250000,0	250000,0	0 %
9	9	200000,0	200000,0	0 %
16	10	117647,1	115200	-2,124 %
34	22	57142,9	57600	0,794 %
103	67	19230,8	19200	-0,16 %
207	CF	9615,4	9600	-0,16 %

Pozn. Komunikační norma pro UART stanovuje možnou chybu v komunikačních rychlostech max. 3%, proto všechny tyto rychlosti vyhovují dané normě. Veškeré pohony AX, jsou od výroby nastavené na hodnotu 1, tj. Na komunikační rychlost 1 000 000 bps.

Adresa 0x05 Return Delay Time.

Potřebný čas pro návrat Status Packet-u po vyslání Instrukčního packet-u. Tento čas je určen hodnotou na této adrese x 2 μs (mikro-sekundy).

Adresa 0x0B the Highest Limit Temperature. (Limit nejvyšší teploty)

Nejvyšší dovolená pracovní teplota. Jestliže interní teplota bude větší než nejvyšší možná, tak bude nastaven (do log.1) stavový bit Over Heating Error Bit (Bit 2 - Status Packetu) a dále bude nastaven alarm na adrese 17 a 18. Hodnoty teplot jsou uváděny v Celsiových stupních.

Adresa 0x0C,0x0D the Lowest (Highest) Limit Voltage.

The Nejnižší a nejvyšší povolený napětový limit napájecího napětí. If the present voltage (Address 42) is out of the specified range, a Voltage Range Error Bit (Bit 0 of the Status Packet) will return the value 1, and an alarm will be set by Address 17, 18. Hodnoty na těchto adresách jsou násobeny desíti. Např., jestliže adresa 0x0C obsahuje hodnotu 80, pak nejnižší napětový limit je nastaven na 8V.

Adresa 0X10 Status Return Level.

Určuje, zda pohon Dynamixel vrátí stavový Packet po obdržení Instrukčního Packet-u.

Adresa 0x14 Obstacle Detected Compare Value

Tato hodnota slouží k porovnávání, zda-li je detekována překážka či nikoliv. Jedná se v podstatě o prahovou hodnotu. Tj. Pokud je hodnota z infračerveného snímače větší než hodnota na této adrese 0x14, protože označuje překážku ve stanovené vzdálenosti, je bit (IR Detected - adresa 0x20) nastaven na hodnotu "1". A naopak, když hodnota snímače je nižší než hodnota na této adrese 0x14, je nastaven bit na "0". Tato hodnota je alokována v paměti ROM (Address 0x14) a RAM (adresa 0x34) a když je zapnuto napájení, je zkopírována hodnota z EEPROM do paměti RAM.

Adresa 0x15 Light Detected Compare Value

Tato hodnota je podobná jako hodnota na adrese 0x14 avšak s tím rozdílem, že detekuje intenzitu světla.

Adresa 0x1A~0x1C Infrared Sensor Data (Left/Center/Right)

Je to hodnota infračerveného čidla Dynamixel senzoru pro měření vzdálenosti. Infračervený senzor AX-S1 vyzařuje vysokofrekvenční infračervený paprsek, který se odrazí od objektu nebo od stěny zpět k senzoru. Infračervený přijímač AX-S1 dokáže množství infračerveného záření změřit. Vysoká hodnota bude získána jestliže se předmět nebo stěna nachází v blízkosti senzoru. Naměřená hodnota se pohybuje od 0 ~ 255. Vzhledem k vlastnostem měřicí metody infračerveným paprskem, hodnoty odraženého infračerveného paprsku se mohou lišit také v závislosti na barvě objektu nebo povrchu textury.

Adresa 0X1D~0x1F Luminosity (Left/Center/Right)

Je to hodnota čidla reprezentující intenzitu dopadajícího světla. Technologický koncept je podobný s infračerveným senzorem. Nicméně, toto čidlo měří pouze množství infračerveného paprsku vyzařovaného ze zdroje osvětlení. Naměřená hodnota se pohybuje od 0 ~ 255.

Adresa 0x20 Obstacle Detection Flag

Pokud je hodnota infračerveného dálkového senzoru větší než Obstacle Detected Compare Value, tak AX-S1 rozpoznává existenci objektu a nastaví bit (Object Detection) na 1. Níže je popsána bitová reprezentace každého čidla.

Bitová reprezentace:

Bit 2 – Objekt je detekována na pravé straně senzoru (Right Sensor /Light Detected)

Bit 1 – Objekt je detekována v prostřední senzoru (Center Sensor /Light Detected)

Bit 0 – Objekt je detekována na levé straně senzoru (Left Sensor /Light Detected)

Adresa 0x21 Luminosity Detection Flag

Pokud je hodnota světelného snímače větší než hodnoty na adrese 0x15 - Light Detected Compare Value, AX-S1 rozpoznává existenci zdroje osvětlení a nastaví bit na 1. Bit zastoupení každého snímače je stejný s bitovou reprezentací podobně jako u adresy 0x20.

Adresa 0x23 Sound Data

Představuje intenzitu zvukových vln zjištěných přes mikrofon AX-S1. Změřená hodnota v průběhu nehlukného stavu je reprezentována číslem 127 ~ 128 (0x7F ~ 0x80). Rozsah měření je v rozmezí od 0 do 255 (0xFF) což reprezentuje hlučný stav. Zvuková vlna měří frekvenci 3,8 kHz.

Adresa 0x24 Sound Data Max Hold

Tato hodnota reprezentuje informaci o nejhlasitějším zvuku, který byl naměřen. To znamená, že když stávající zvuková data překročí hodnotu zvukových dat Max HOLD, budou současná zvuková data zkopírována jako nová data zvuku Max HOLD.

Adresa 0x25 Sound Detected Count

Tato hodnota reprezentuje počítadlo, které počítá výskyt hlasitého zvuku než je standardní úroveň. Jako příklad lze uvést, počítání počtu tlesknutí. Nicméně, počítadlo nezapočítává další zvuky po dobu 80ms, aby nedošlo k započítání jediného tlesknutí vícekrát. 800msec po poslední detekci zvuku, bude hodnota čítače zvuku uložena. Při více-násobné detekci zvuku se počítadlo inkrementuje interně. Hodnota detekující počet zvuků bude uložena po 800 milisekundách.

Adresa 0x2A Present Voltage.

Napětí, kterým je napájen pohon Dynamixel. Hodnota je 10 krát větší než skutečné napětí. Například, 10V je reprezentováno hodnotou 100 dec (0x64 hex).

Adresa 0x2B Present Temperature.

Vnitřní teplota Dynamixel pohonu ve stupních Celsia.

Adresa 0x2C Registered Instruction.

Nastaveno na 1, když je instrukce přiřazena REG_WRITE příkazem. Nastaveno na 0 po dokončení přiřazeného pokynu akčním příkazem.

Adresa 0x2E IR Remocon Arrived

AX-S1 má infračervený komunikační modul zabudovaný v centru, a tak umožňuje komunikaci mezi AX-S1. Komunikace je realizována dvěma byty.

Jestliže jsou data získané senzorem, IR Remocon Arrived hodnota se změní na hodnotu 2, signalizující 2 bytový přenos. Pokud budou přečtena data z IR Remocon RX Data, tak IR Remocon Arrived se automaticky inicializuje zpět na 0.

Adresa 0x2F Lock.

Pokud je na tuto adresu zapsána hodnota 1, tak bude možné zapisovat pouze na adresy 0x18 až 0x23. Na ostatní adresy zapisovat nebude možné dokud se neodpojí napájení pohonu.

Adresa 0x30,0x31 IR Remocon RX Data

Adresy, kde jsou uložena data přijatá infračerveným senzorem.

Adresa 0x32,0x33 IR Remocon TX Data

Adresy, kde jsou uložena data, které budou přenášeny prostřednictvím infračerveného vysílače. Po zapsání 2 bytové hodnoty, budou data okamžitě vysílány.

Adresa 0x34 Obstacle Detected Compare Value

Toto je místo v řídicí tabulce, v RAM části, kde je ukládána hodnota z adresy 0x14.

Pozor! IR senzory AX-S1 emitují silné infračervené paprsky k detekci předmětů ve větší vzdálenosti. Proto je nemožné, detekovat objekt v krátké vzdálenosti, tj. kolem 5 cm.

Chceme-li tomu zabránit, AX-S1 podporuje režim pro detekci předmětů na krátkou vzdálenost. Pokud Obstacle Detected Compare Value je 0, je AX-S1 v režimu detekce předmětů na krátkou vzdálenost. V tomto režimu jsou snímací schopnosti na velkou vzdálenost velmi slabé, ale díky tomu je možno detekovat předměty velmi přesně a citlivě na krátké vzdálenosti.

Adresa 0x35 Light Detected Compare Value

Toto je místo v řídicí tabulce, v RAM části, kde je ukládána hodnota z adresy 0x15.

1.2.2. Příklady použití AX-S1

V této kapitole jsou uvedeny jednotlivé příklady, které ukazují nastavení důležitých registrů pro danou činnost se sensorickým modulem AX-S1.



ŘEŠENÉ PŘÍKLADY – Měření vzdálenosti od překážky

1	START PROGRAM
2	{
3	ENDLESS LOOP
4	{
5	vzdalenost = ID[100]: IR Center
6	
7	a = a + 1
8	Print = a
9	Print with Line = vzdalenost
10	
11	Timer = 1.024sec
12	WAIT WHILE (Timer > 0.000sec)
13	}
14	}
15	



ŘEŠENÉ PŘÍKLADY – Detekce zvukových impulsů (tlesknutí, lusknutí, apod.)

```

1  START PROGRAM
2  {
3      ENDLESS LOOP
4      {
5          IF ( ID[100]: Sound count == 1 )
6          {
7              ID[2]: Goal position = 0
8          }
9
10         IF ( ID[100]: Sound count == 2 )
11         {
12             ID[2]: Goal position = 1023
13         }
14     }
15 }
16

```

1.3. Řídicí jednotka CM-5 / CM-510 / CM-530



ČAS KE STUDIU: 30 minut



CÍL

Prostudováním této kapitoly získá čtenář přehled o základních vlastnostech řídicích jednotek CM-5, CM-510 a CM-530 firmy Robotis.



VÝKLAD

Firma Robotis vyrábí řídicí jednotky s označením CM. V této kapitole budou popsány pouze řídicí jednotky CM-5, CM-510 a CM-530. Srovnání základních vlastností ukazuje následující tabulka č. 3.

Řídicí jednotky slouží primárně k řízení sestaveného robota skrze připojené pohony řady AX, senzorické moduly AX-S1 a případně samostatné snímače. Připojení samostatných snímačů však podporují pouze řídicí jednotky CM-510 a CM-530 (viz. Tabulka č. 3).

Ke každé řídicí jednotce je možné připojit baterie, čímž je možné provozovat robota zcela nezávisle na externím napájení. Po vybití baterií je možné provozovat robota s externím napájením i bez připojených baterií. Protože řídicí jednotka CM-5 používá jiný typ baterií než řídicí jednotky CM-510 a CM-530, bude postup nabíjení popsán v samostatných kapitolách 1.3.1 pro CM-5 a v kapitole 1.3.2 pro CM-510 a CM-530.

Řídicí jednotky dokážou pracovat ve třech různých režimech. Jedná se o režimy MANAGE, PROGRAM a PLAY. Po zapnutí hlavního vypínače se rozbliká jedná ze tří LED, reprezentující vybraný mód. Stiskem tlačítka START je pak možné tento mód aktivovat. Aktivovaný mód lze poznat podle neustálého svitu LED. Deaktivovat vybraný mód lze stiskem červeného tlačítka MODE. Opakovaným stiskem tohoto tlačítka MODE lze přepínat mezi jednotlivými módy.

Tabulka č.3 – Srovnání základních vlastností řídicích jednotek CM-5 / 510 / 530

Funkce	CM-5	CM-510	CM-530
Využití RoboPlus software	*	*	*
Bioloid sériový kabel	*		
Mini USB kabel		*	*
Podpora pohonů Dynamixel řady AX	*	*	*
Podpora pohonů Dynamixel řady MX	*	*	*
Samostatný IR senzor		*	*
Samostatný dotykový senzor		*	*
Samostatný senzor pro měření vzdálenosti překážky		*	*
Možnost připojení modulu AX-S1	*	*	*
Možnost připojení modulu IR Sensor Array	*	*	*
Možnost připojení gyroskopického senzoru		*	*
Možnost řízení servo motoru z OLLO systému			*
Možnost řízení motoru s převodovkou z OLLO systému			
Možnost připojení LED modulu			*
Využití IR přijímače pro bezdrátovou komunikaci		*	*
Možnost připojení modulu ZIG-100	*		
Možnost připojení modulu ZIG-110A		*	*
Možnost připojení modulu BT-110		*	*
Napájení z LIPO baterií (11,1V)		*	*
Napájení z NIMH baterií (9,6V)	*		
Obsahuje pojistku 125V / 10A , LFU-10		*	*
Obsahuje pojistku 250V / 5A , FU-5	*		
Počet analogových vstupů	0	1	1
Počet binárních vstupů	0	0	0
Počet binárních výstupů	0	1	2
Počet analogových výstupů	0	0	0

Režim MANAGE:

Tento režim slouží zejména k nastavování parametrů připojených pohonů řady AX, parametrů senzorických modulů AX-S1 a také k nastavení řídicí jednotky. Dále tento režim může sloužit k testování a přípravnou fází při tvorbě řídicích programů.

Režim PROGRAM:

Tento režim slouží k realizaci nebo editaci pohybových souborů (motion files).

Režim PLAY:

Po vytvoření a nahrání řídicího programu (task files) do řídicí jednotky, lze tento program spustit aktivací tohoto módu.

Význam LED indikátorů:

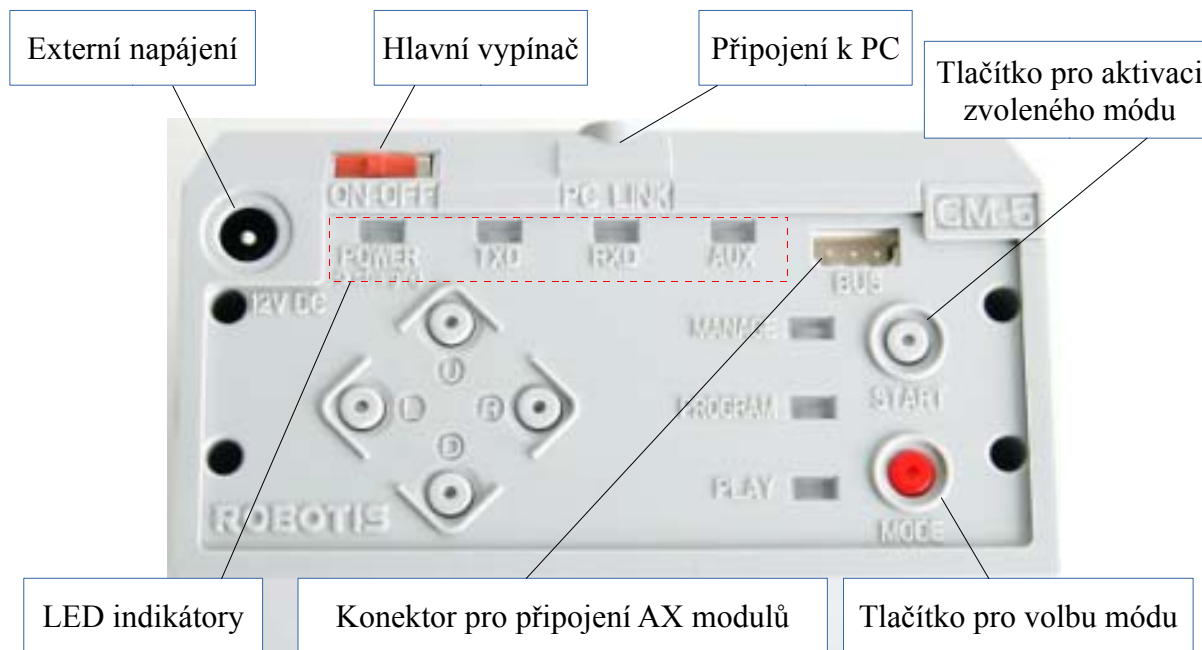
POWER: *Svítí* pokud je připojené napájení a hlavní vypínač je v pozici ON. *Bliká* pokud je nabíjena baterie nebo pokud je baterie téměř vybitá (*platí pouze pro řídicí jednotku CM-5*).

TXD: Svítí pokud jsou data vysílána.

RXD: Svítí pokud jsou data přijímána.

AUX: Tato LED má význam pouze pro uživatelské použití.

1.3.1. CM-5



Obr. č.1.3.1 – Vrchní kryt řídicí jednotky CM-5

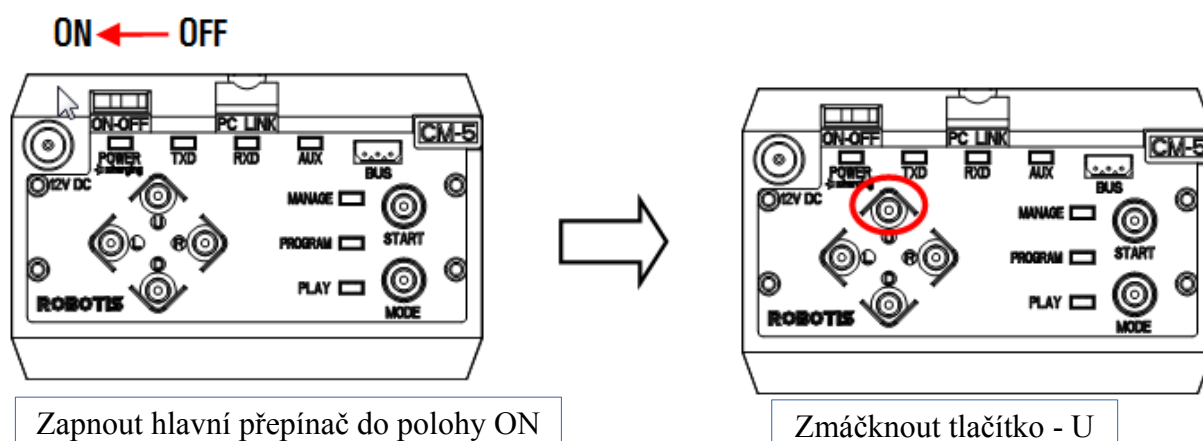
Technické parametry:

- Váha: 126g (bez baterií)
- CPU: ATmega 128
- Provozní teplota: -5°C ~ +70°C
- Napětí: 7 ~ 12V (Doporučené napětí 11.1V)
- Odběr el. proudu v IDLE režimu: 50 mA
- Maximální odběr el. proudu: 5A (limitováno pojistkou)

Nabíjení baterií pomocí řídicí jednotky CM-5:

Řídicí jednotka CM-5 obsahuje vestavěnou elektroniku určenou pro nabíjení bateriového modulu, který je tvořen osmi NIMH bateriemi. Aby bylo možné nabít tento bateriový modul, musí být připojeno externí napájení. Hlavní vypínač musí být zapnutý v poloze ON. Pro vlastní aktivaci nabíjecího režimu je nutné zmáčknout tlačítko U (poté začne blikat POWER LED). Pomalé blikání znamená, že baterie nedosáhla své plné kapacity. Rychlé blikání znamená, že baterie je plně nabitá. POZOR! Toto však platí, pokud je připojené externí napájecí napětí.

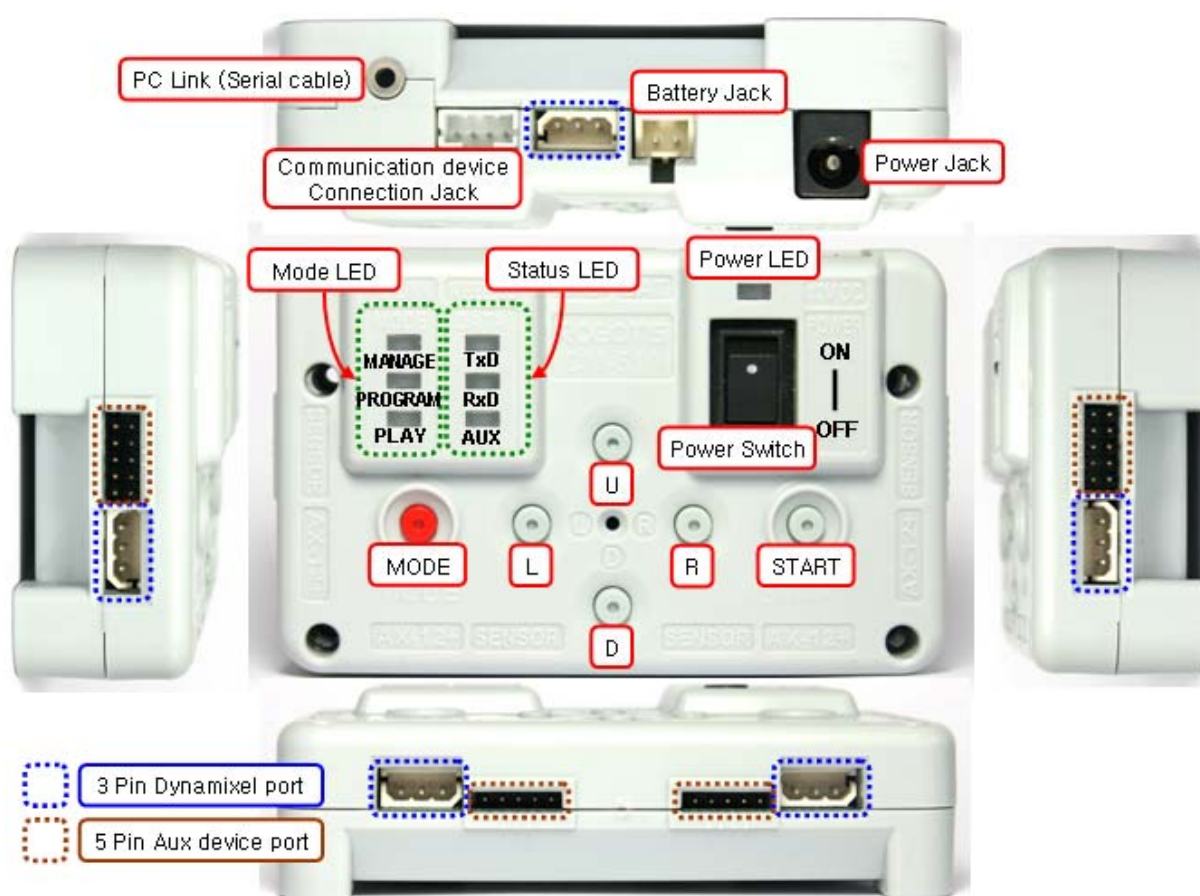
Pokud však začne blikat (pomalu) POWER LED, když externí napájení připojené není, znamená to, že baterie bude zanedlouho zcela vybita. Proto výrobce doporučuje baterii co nejdříve dobít.



Obr. č.1.3.2 – Aktivace nabíjecího režimu u CM-5

1.3.2. CM-510

Řídicí jednotka CM-510 je v podstatě vylepšený model řídicí jednotky CM-5. CM-510 umožňuje připojit stejná zařízení jako CM-5, ale navíc umožňuje připojit samostatné snímače (IR snímače vzdálenosti překážek, dotykový senzory, gyroskopické senzory a nebo také vlastní vyrobené snímače s analogovými výstupy). Navíc součástí této řídicí jednotky je mikrofón pro rozpoznávání zvukových impulsů, snímač napájecího napětí a piezo (zvukový) měnič.

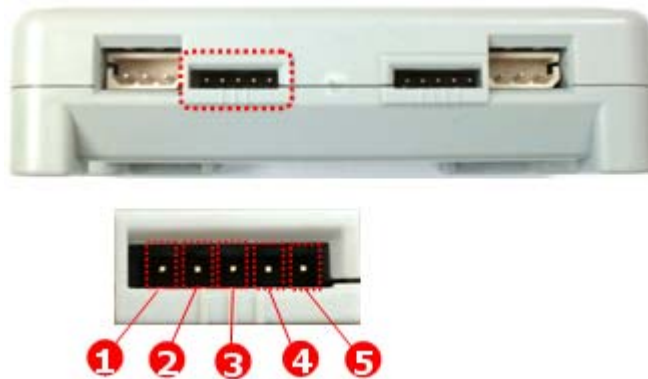


Obr. č.1.3.3 – Přehled portů, tlačítek a LED u řídicí jednotky CM-510

Technické parametry:

- Váha: 51,3g
- CPU: ATmega 2561
- Provozní teplota: -5°C ~ +70°C
- Napětí: 6,5 ~ 15V (Doporučené napětí 11.1V)
- Odběr el. proudu v IDLE režimu: 50 mA
- Maximální odběr externích vstupů a výstupů: 0,9A
- Maximální odběr el. proudu: 10A (limitováno pojistkou)

Popis pinů řídicí jednotky CM-510:



Obr č.1.3.4 –5-ti pinový port řídicí jednotky CM-510

Pin č.1 – digitální výstup 5V. Lze jej zatížit max. na 0,9A.

Pin č.2 – VDD (5V)

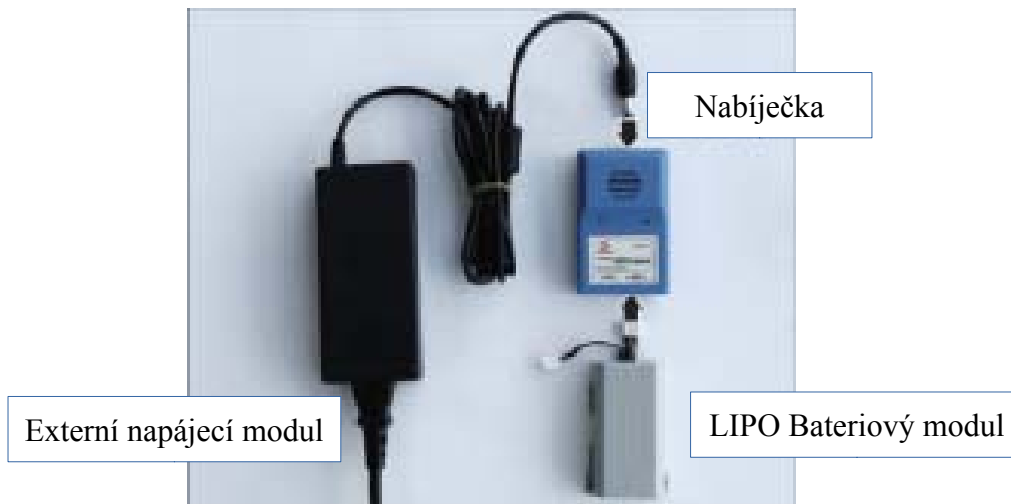
Pin č.3 – Analogový vstup

Pin č.4 – GND

Pin č.5 – NC: není využitý

Nabíjení baterií pomocí řídicí jednotky CM-510:

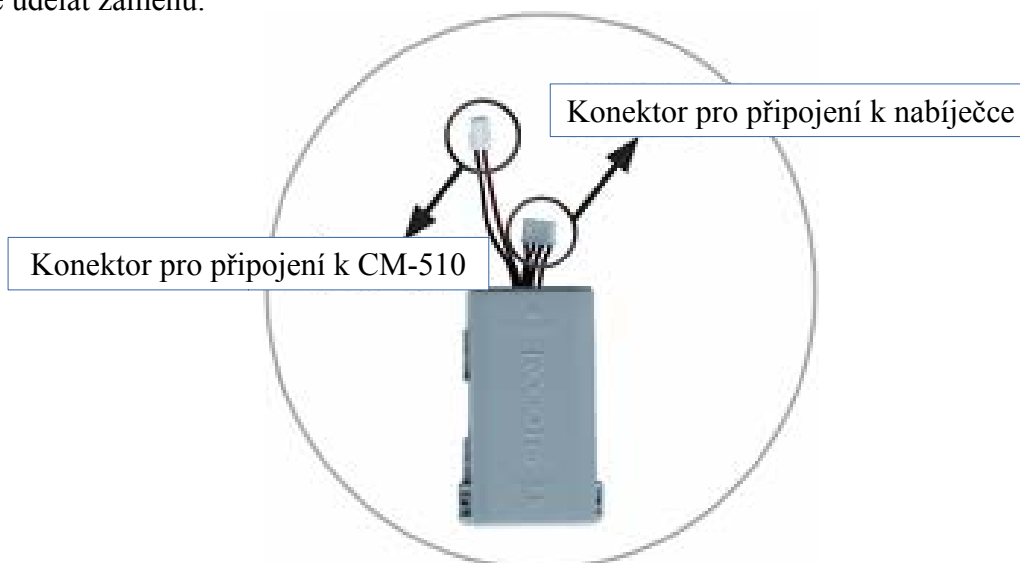
Řídicí jednotka CM-510 je napájena LIPO bateriovým modulem. Pro nabíjení tohoto bateriového modulu se již nepoužívá řídicí jednotka CM-510, ale firma Robotis dodává nabíječku pro tento bateriový modul.



Obr č.1.3.4 – Přehled HW pro nabíjení bateriového modulu řídicí jednotky CM-510

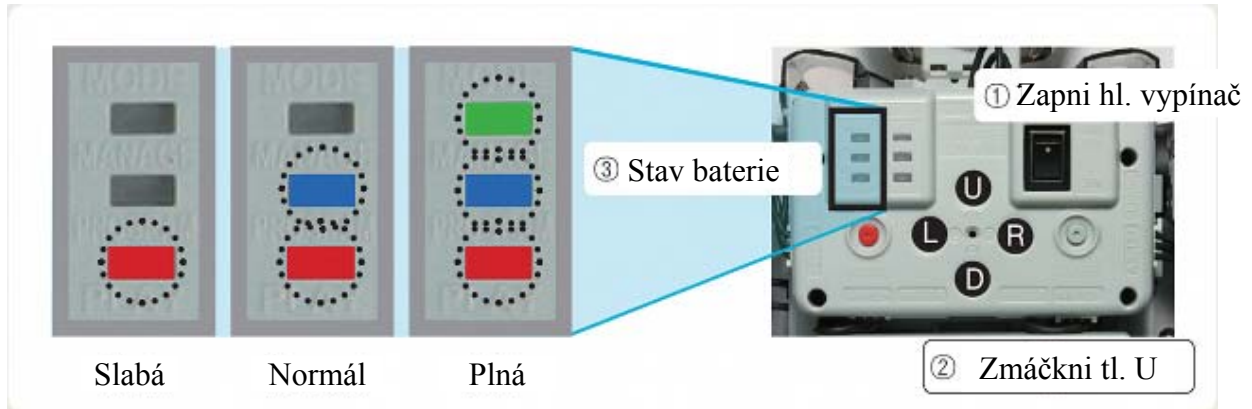
Proces nabíjení spočívá v propojení jednotlivých součástí, jak ukazuje obrázek č. 1.3.4. Pokud se baterie začne nabíjet rozsvítí se červená LED na nabíječce. Po nabití červená LED zhasne a rozsvítí se zelená LED.

Bateriový modul obsahuje dva konektory. Jeden slouží pouze k dobíjení bateriového modulu a druhý slouží pro připojení k řídicí jednotce CM-510. Konektory jsou rozdílné, takže nelze udělat záměnu.



Obr č.1.3.5 – Detail bateriového modulu řídicí jednotky CM-510

U řídicí jednotky CM-510 lze zjistit stav nabití připojené LIPO baterie tím, že se stiskne tlačítko U. Stav baterie pak ukazují LED, jenž primárně slouží pro informaci o vybraném módu, což lze vidět na obrázku č.1.3.6.



Obr č.1.3.6 – Zjištění stavu nabití bateriového modulu řídicí jednotky CM-510

Pokud je baterie slabá, začne tento stav oznamovat řídicí jednotka zvukovým signálem, který je tvořen třemi krátkými zvukovými impulsy, které se cca co 0,5 sekund opakují.

POZOR!

Baterie typu LIPO jsou velice choulostivé a snadno je lze zničit! Výrobce proto např. varuje:

1. Uchovávat vybité baterie po dlouhou dobu, drasticky snižuje jejich životnost.
2. Nepoužívat baterie v příliš vlhkém nebo teplém prostředí.

1.4. Propojení AX-12/AX-S1 s PC



ČAS KE STUDIU: 30 minut



CÍL

Prostudováním této kapitoly získá čtenář přehled o základních možnostech propojení modulů AX-S1 a pohonů AX-12 s PC.

1.4.1. Pomocí řídicí jednotky CM-5/510/530



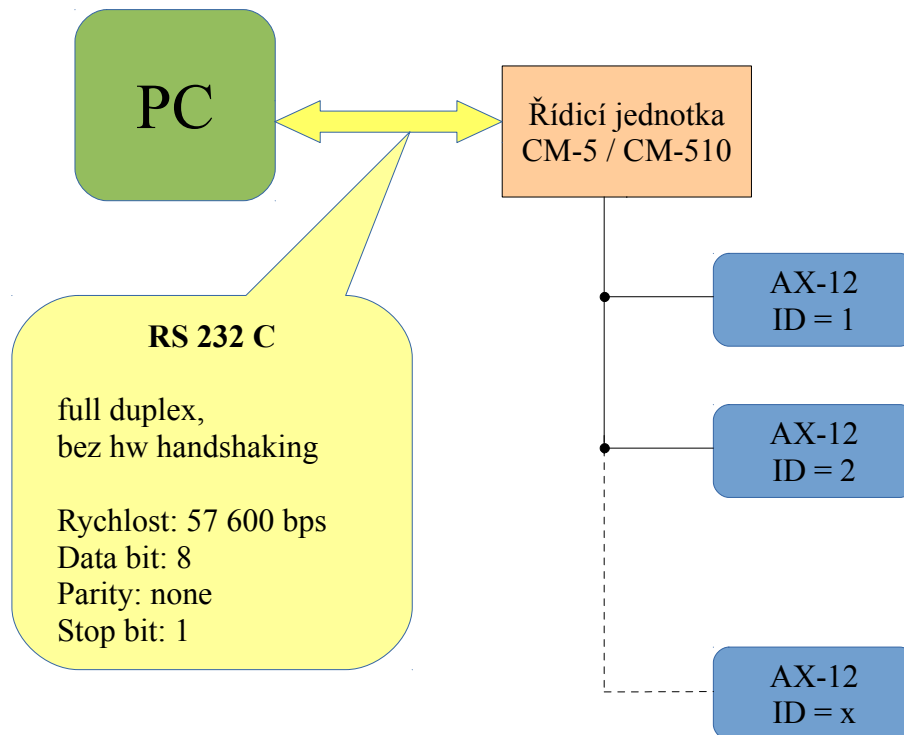
VÝKLAD

Propojení pohonu AX s řídicí jednotkou CM-5 (resp. s CM-510/ CM530) demonstruje následující obrázek č.1.4.1. Zmíněné řídicí jednotky obsahují tři pinové konektory Molex, které jsou určeny pro připojení pohonů AX. Všechny tyto konektory jsou na zmíněných řídicích jednotkách vzájemně propojeny, proto místo (pozici) připojení pohonu AX volíme nejčastěji na základě mechanických možností konstrukce robota.

POZOR - komunikace mezi pohony AX (a také mezi senzorickým modulem AX-S1) a řídicí jednotkou CM-5 / CM-510 /CM-530 je realizována pomocí RS 232 TTL (s úrovněmi napětí 0 / 5V). Pokud je použito více pohonů, pak jsou propojeny paralelně. Každý pohon AX má defaultně nastaveno jedinečné identifikační číslo tzv. ID, pomocí kterého řídicí jednotka CM-x určuje, s kterým pohonem právě komunikuje.

Pokud je z nějakého důvodu (nový motor s jiným ID nebo byl použitý příkaz RESET) potřeba změnit identifikační číslo pohonu, musí být k řídicí jednotce připojen pouze jediný pohon AX-12 (to jen v případě, že minimálně dva pohony mají stejný ID). Pokud pohony mají různá ID a je nutné jejich „přečíslování“, pak lze tato ID měnit aniž by bylo nutné mít připojený pouze jeden pohon, avšak v jednom čase nesmí být připojeny dva pohony se stejným ID!

Komunikace mezi PC a řídicí jednotkou CM-5 probíhá po standardní sériové lince RS 232. Parametry komunikace jsou uvedeny na obrázku č.1.4.1.



Obr. č.1.4.1 - Koncepce propojení pohonů AX-12 a řídicí jednotky s PC

Princip komunikace:

Řídicí jednotka může přímo ovládat pohony AX-12 na základě přijatých instrukcí resp. příkazů z PC v módu „**Manage Mode**“. Nastavení a spuštění tohoto módu je popsáno v kapitole 1.3 nebo v dokumentaci na <http://support.robotis.com/en/>

Po aktivaci tohoto módu řídicí jednotka CM-x očekává sekvenci ASCII znaků ukončenou znakem Carriage Return (CR), resp. ASCII kod v desítkové soustavě dec.=13 a nebo v šestnáctkové soustavě hex.= 0d.

Řídicí jednotka automaticky odesílá zpět do PC, každý přijatý znak. (Může sloužit jako potvrzení o správném přijetí, nebo pro realizaci terminálu.) Řídicí jednotka posílá do PC také data, která mohou být odpovědí na přijatou instrukci. Veškerá komunikace (příjem / vysílání) probíhá ve formě ASCII znaků.

Např. Pohon AX-12 má zdokumentovaný, viz. kapitola 1.1.2, řídicí paměťový registr (Control Table – Řídicí tabulka), jehož hodnoty určují chování konkrétního pohonu AX-12. Pokud pošleme řídicí jednotce instrukci **dump**, tak nám řídicí jednotka, kromě slova dump a znaku CR, pošle také kompletní výpis tohoto paměťového registru.

Po aktivaci módu „**Manage Mode**“ je nutné poslat řídicí jednotce CM-5 instrukci „**cid**“, která říká, s kterým pohonem daného ID se bude nadále komunikovat. Ostatní příkazy jsou určeny pouze tomu pohonu, který byl původně nastaven. Toto platí, dokud nedojde k vypnutí řídicí jednotky CM-5 a nebo k deaktivaci módu „**Manage Mode**“.



ŘEŠENÉ PŘÍKLADY

Ukázka realizace události v jazyce C#, která je vykonána po stisku tlačítka button2

```
private void button2_Click(object sender, EventArgs e)
{
    // nastaví řídicí jednotku pro komunikaci s pohonem ID = 14
    // znak "\r" reprezentuje ASCII znak CR, resp. Stisk klávesy Enter
    serialPort1.Write("cid 14" + "\r");

    // pohon AX-12 se natočí do pozice 1000 s rychlostí 500
    serialPort1.Write("go 1000 500" + "\r");
}
```

2. Základní úlohy – RoboPlus Manager a RoboPlus Task



ČAS KE STUDIU: 30 minut



CÍL

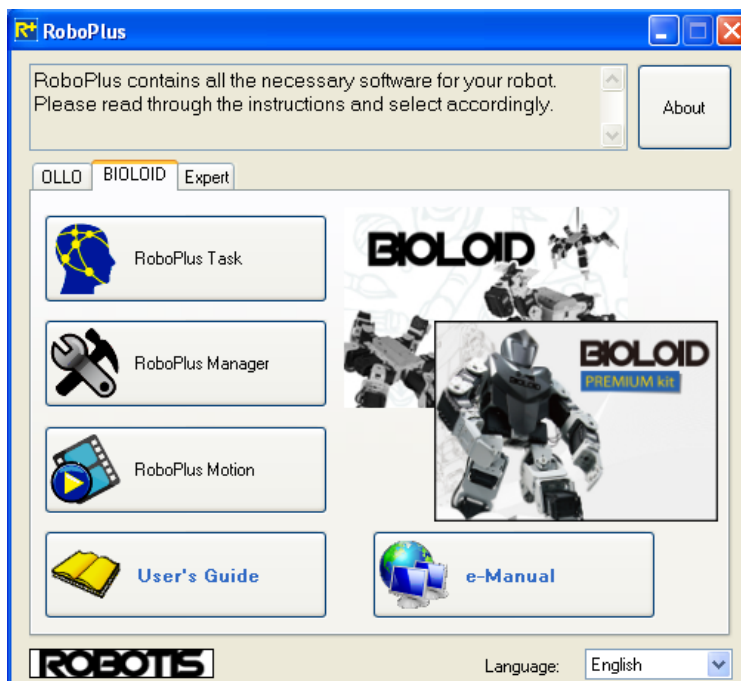
Prostudováním části této kapitoly získá čtenář přehled o software RoboPlus Manager.



VÝKLAD

Tato kapitola je zaměřena na popis základů tvorby algoritmů pomocí software **RoboPlus Task**. Tento software je zjednodušený vývojový nástroj, který je určený k tvorbě programů, které mohou být nahrány do řídicí jednotky CM-5, CM-510 nebo CM-530. Tato řídicí jednotka pak může pomocí tohoto programu řídit robotický systém.

Po instalaci a spuštění software RoboPlus Studia, je k dispozici dialogové okno se třemi kartami. Karta OLLO, BIOLOID a EXPERT reprezentují různé skupiny aplikací, které je možné využívat pro práci s robotickým systémem firmy ROBOTIS.

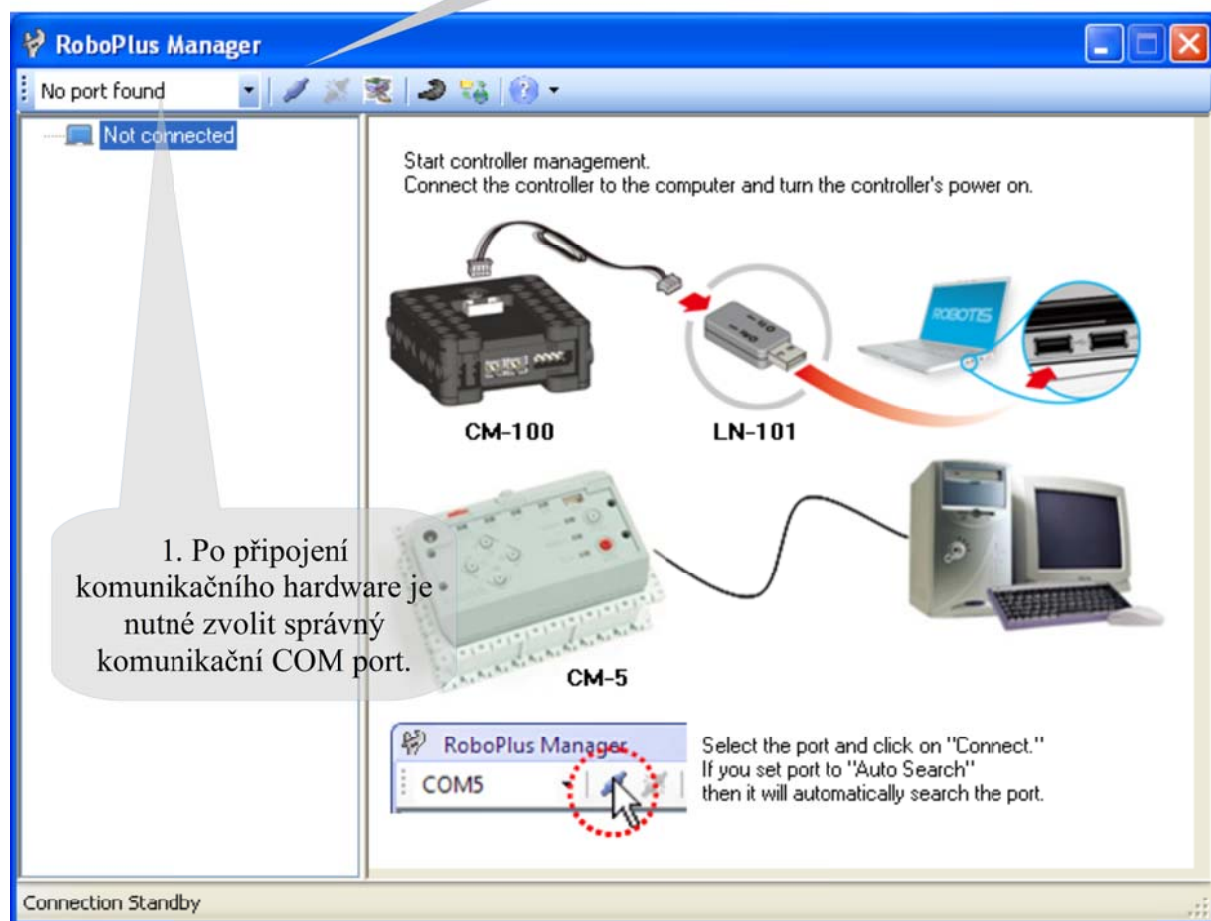


Obr. č. 2.1 – Hlavní obrazovka software RoboPlus Studia

V našem případě budeme využívat kartu BIOLOID. Pouze v případě kdy je potřeba upravit firmware jednotlivých pohonů, lze na kartě EXPERT najít navíc dvě aplikace, které umožňují pokročilou práci se systémem Bioloid.

Kromě aplikace RoboPlus Task, která slouží pro vytváření konkrétních algoritmů, je na této kartě k dispozici také aplikace **RoboPlus Manager**. Tato aplikace slouží pro základní konfiguraci řídicí jednotky CM-5, CM-510 nebo CM-530 a také připojených pohonů AX-12, případně sensorických modulů AX-S1.

2. Stisknutím této ikony se provede propojení mezi řídicí jednotkou a PC.



1. Po připojení komunikačního hardware je nutné zvolit správný komunikační COM port.

Obr. č. 2.2 – Hlavní okno software RoboPlus Manager

Pro začátečníky se doporučuje před započatím tvorby jakéhokoliv algoritmu v aplikaci RoboPlus Task, provedení zkušebního připojení robotického hardware pomocí aplikace RoboPlus Manager. Tímto je možno ověřit správnou funkci komunikačního hardware a konfiguraci jednotlivých pohonů.

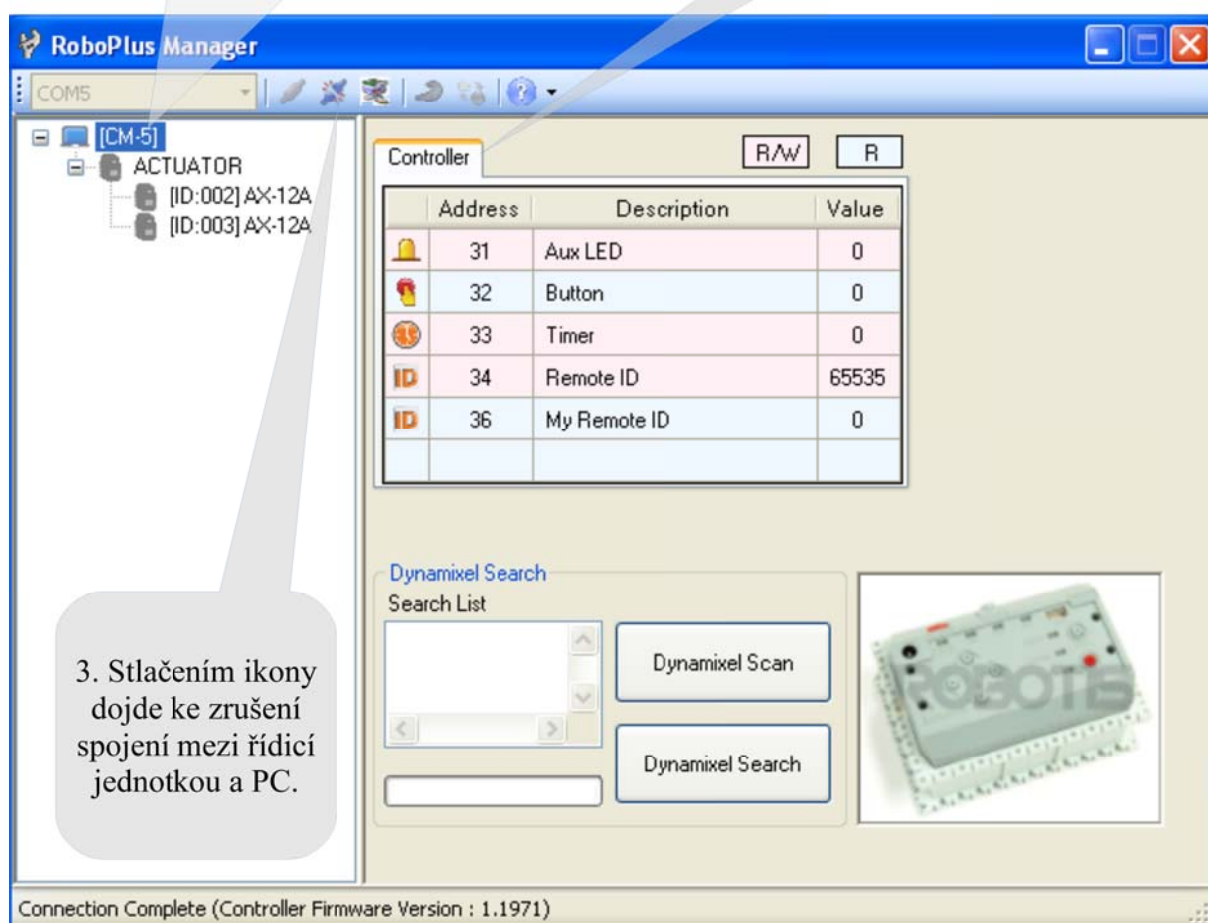
Po úspěšném připojení řídicí jednotky k PC vypadá obrazovka programu RoboPlus Manager-u přibližně tak, jak ukazuje následující obrázek.

1. Připojená řídicí jednotka. V tomto poli je možné vybrat, kterýkoliv z připojených pohonů AX-12 nebo sensorických modulů AX-S1.

Pokud v tomto seznamu nemůžeme najít pohon nebo sensorický modul s příslušným ID, je nutné prověřit zda-li je daná jednotka připojena.

Pokud připojena je, tak je velmi pravděpodobné, že ID jednotky může být stejné s jinou připojenou jednotkou. Proto se doporučuje odpojit všechny jednotky a připojit problematickou jednotku samostatně.

2. V tomto poli se zobrazují paměťové registry vybraného zařízení.



The screenshot shows the RoboPlus Manager software interface. On the left, a tree view shows a connected controller [CM-5] with two actuators: [ID:002] AX-12A and [ID:003] AX-12A. The main window displays the 'Controller' settings, including a table of memory registers (Address, Description, Value) and a 'Dynamixel Search' section with a 'Search List' and buttons for 'Dynamixel Scan' and 'Dynamixel Search'. A small image of the controller hardware is shown in the bottom right corner. The status bar at the bottom indicates 'Connection Complete (Controller Firmware Version : 1.1971)'.

Address	Description	Value
31	Aux LED	0
32	Button	0
33	Timer	0
34	Remote ID	65535
36	My Remote ID	0

3. Stlačením ikony dojde ke zrušení spojení mezi řídicí jednotkou a PC.

Obr. č. 2.3 – Zobrazené informace při úspěšném spojení řídicí jednotky s PC



ČAS KE STUDIU: 30 minut



CÍL

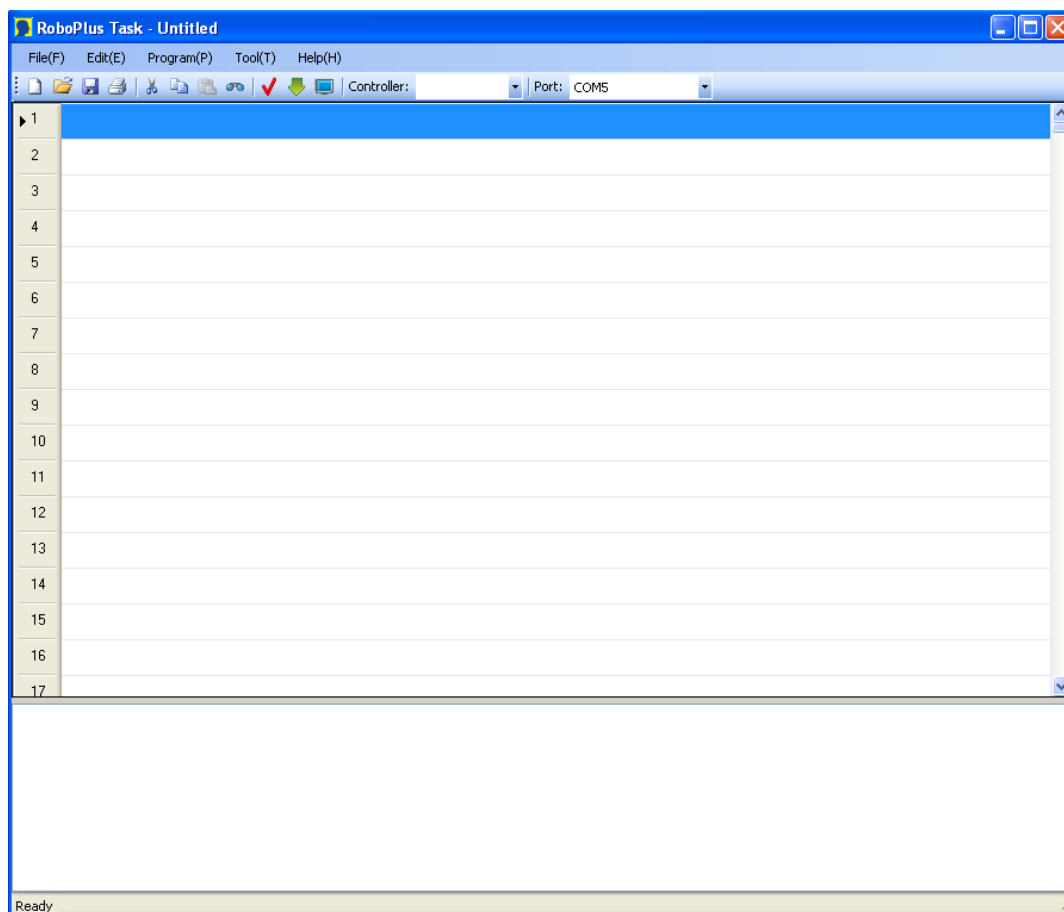
Prostudováním této kapitoly získá čtenář přehled o software RoboPlus Task.



VÝKLAD

Po úspěšné konfiguraci jednotlivých ID a ověření propojení je možné začít tvořit algoritmus, který bude sloužit k ovládní robotického systému. K tomuto účelu již využijeme program RoboPlus Task.

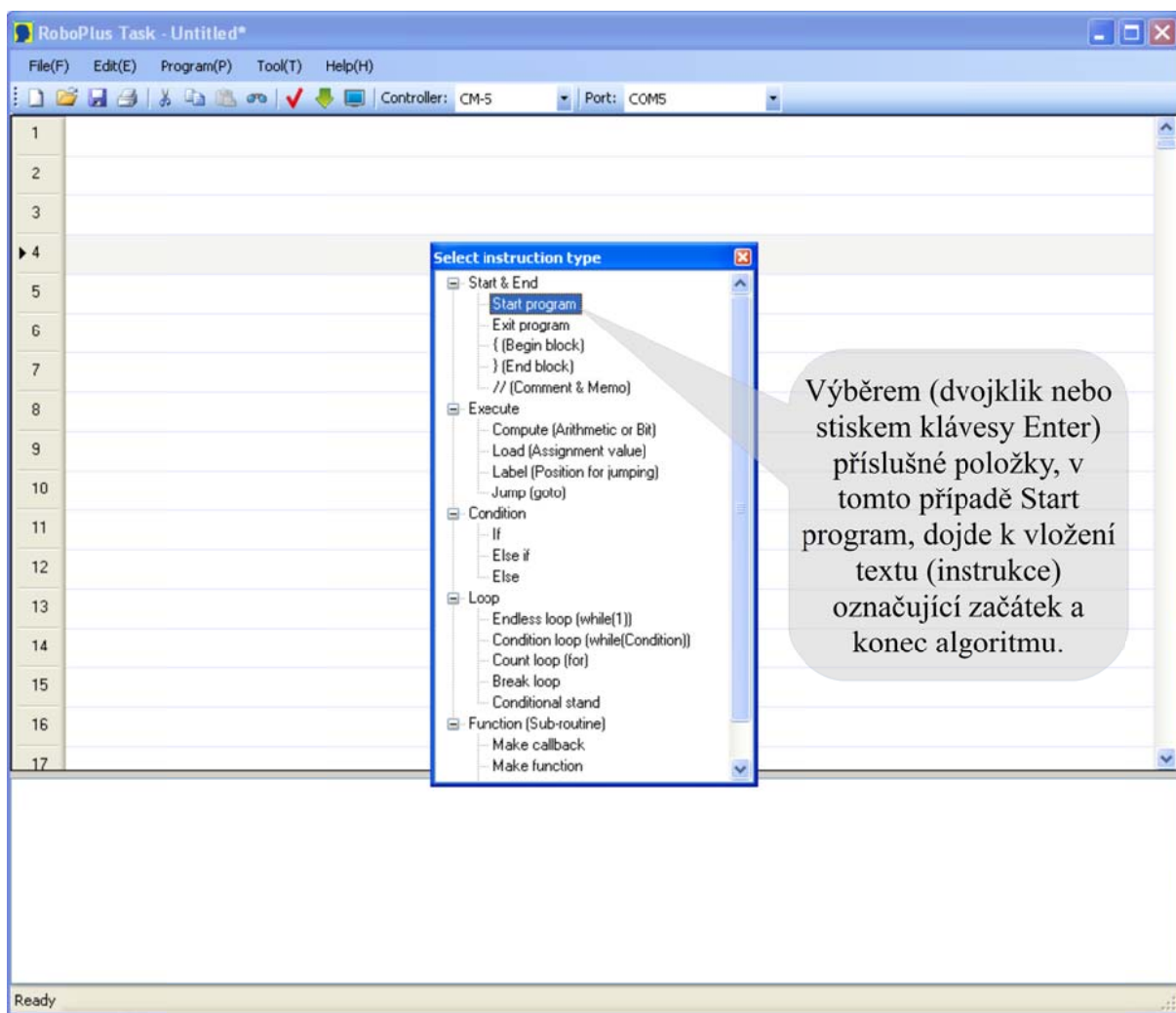
Jedná se o zjednodušené vývojové prostředí tzv. IDE (*Integrated Development Environment*). Toto prostředí je vytvořeno speciálně pro začátečníky, proto neumožňuje běžnou editaci textů, jak je tomu zvykem u jiných vývojových prostředí.



Obr. č. 2.4 – Hlavní okno software RoboPlus Task

Důležité je, že každá instrukce algoritmu, v tomto případě, musí být umístěna na samostatném řádku. A protože algoritmus může obsahovat komentáře a uživatelsky definované funkce je nutné na počátku práce definovat kde začíná a končí algoritmus. Toto se definuje tzv. Hlavní funkcí.

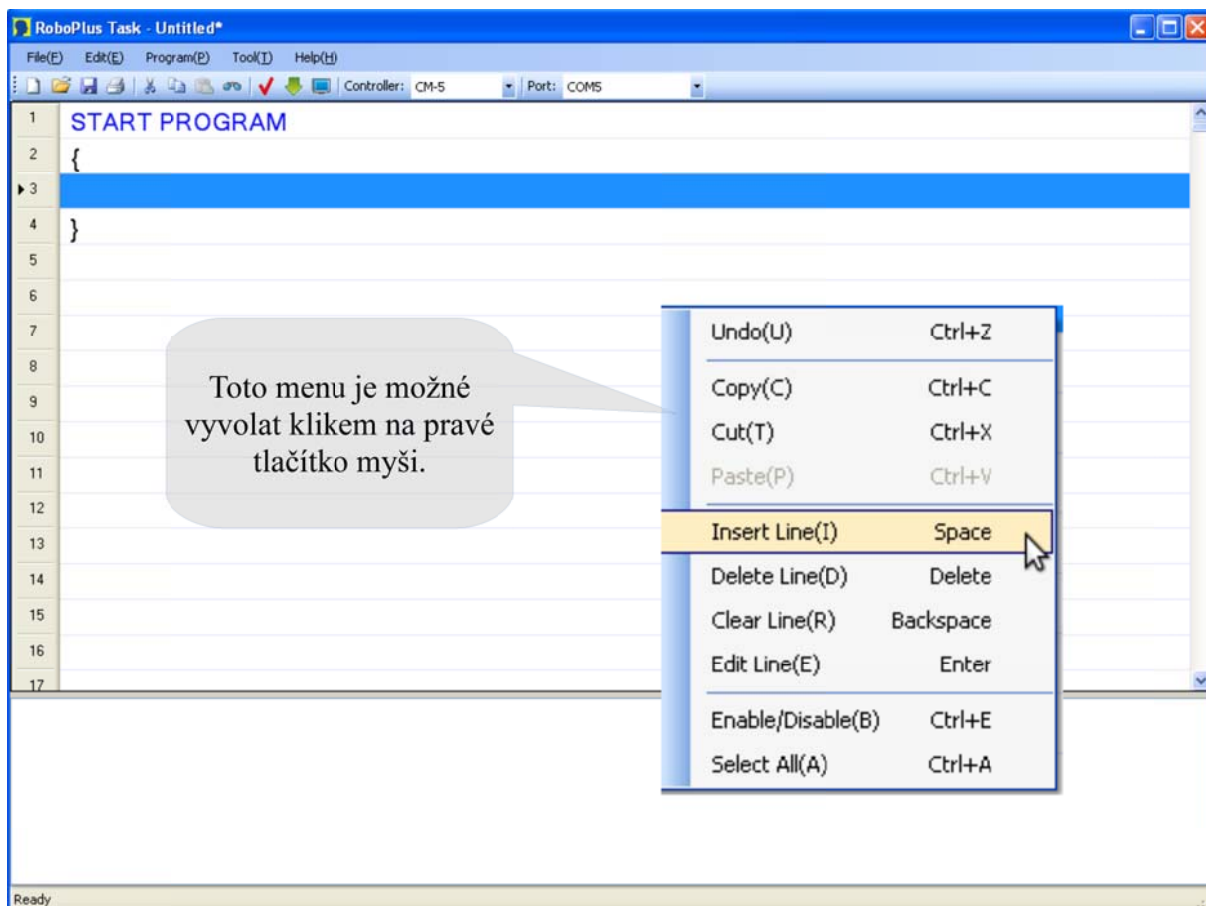
Vložení hlavní funkce případně jiných instrukcí se provádí přes speciální menu, které je možné zobrazit stiskem klávesy Enter, dvojklikem myši na nějakém řádku nebo výběrem z menu.



Obr. č. 2.5 – Seznam instrukcí v software RoboPlus Task

Takto vypadá vložená „instrukce“ START PROGRAM, která má jediný účel. A to, jasně definovat kde začíná a končí tvořený algoritmus. Tvořený algoritmu se postupně vkládá mezi složené závorky.

V tomto případě je mezi těmito složenými závorkami pouze jeden řádek – tj. prostor pouze na jednu instrukci. Doplnovat řádky lze stiskem klávesy mezerník, případně výběrem příslušné položky z menu. Tímto je možné vytvářet více řádkové algoritmy.

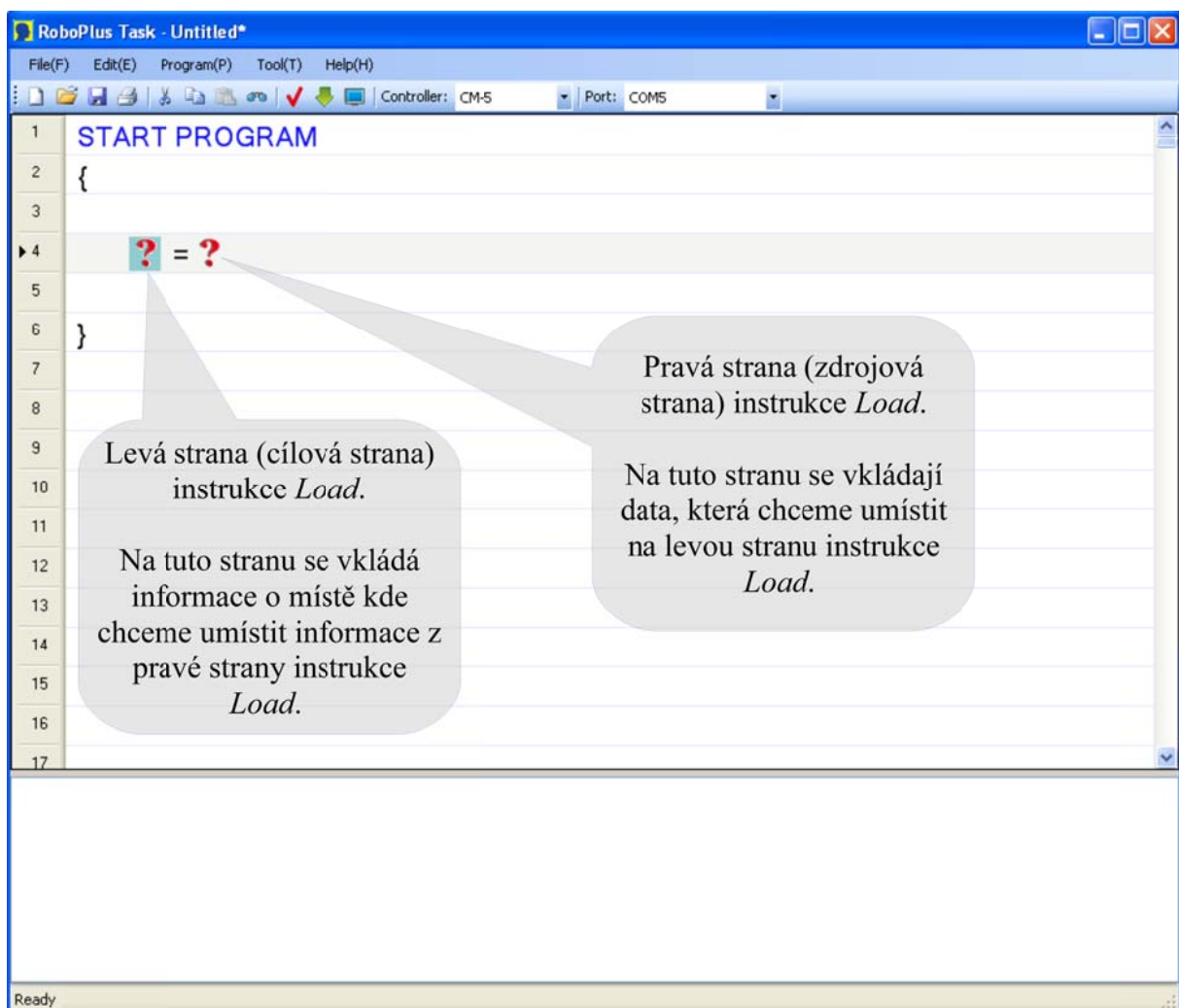


Obr. č. 2.6 – Pomocné menu v software RoboPlus Task

2.1. Vložení instrukce

Jak bylo ukázáno v předchozí kapitole (obrázek č. 2.5), instrukce se dělí do několika typů. Instrukce, pomocí kterých je možné zapisovat nebo číst do/z paměťových tabulek pohonů AX-12, senzorických modulů AX-S1 nebo řídicích jednotek, patří do skupiny vykonávajících (Execute).

V této skupině jsou umístěny celkem čtyři instrukce, ale pouze dvě jsou prakticky využívány. Tyto instrukce se nazývají *Compute* (vypočítá a přiřadí) a *Load* (přiřadí). Další dvě instrukce se nazývají *Label* (návěští) a *Jump* (skok). Tyto dvě instrukce se v profesionálních programovacích jazycích nedoporučují používat. Hlavním důvodem je, že díky těmto instrukcím vzniká menší přehlednost v programu. Avšak v odůvodněných případech použití těchto instrukcí může být opodstatněné.



Obr. č. 2.1.1 – Vložená instrukce *Load* v software RoboPlus Task

Použití instrukce *Load* je podobné, jako například v matematice zápis $x=5$. Tj. Z toho zápisu asi každý pochopí, že do proměnné x se vloží číslo 5. A v případě zápisu např. $y = x + 2$, by mělo být jasné, že $y = 7$.

V případě, že na počátku napíšeme $5 = x$, a pak pokračujeme zápisem $y = x + 2$, tak pro člověka je to stále stejný výsledek. Protože člověk „přirozeně“ (alespoň takto se to každý naučil ve škole) vkládá číslo do písmene (proměnné).

Avšak počítač je stroj, který nechápe věci tak jako člověk a proto při psaní algoritmů je potřeba dodržovat určitá pravidla aby bylo jasné co požadujeme. Podobně jak se používá přiřazení v matematice, tak se toto pravidlo začalo používat i v programování.

Proto levou stranu, je nutné vnímat vždy jako stranu kam se vkládají informace a pravou stranu je nutné vnímat, jako stranu odkud se berou informace.

Jako ukázka použití této instrukce může být následující program, který po spuštění rozsvítí informační LED diodu na pohonu s ID = 10.

```

1  START PROGRAM
2  {
3  ID[10]: LED = TRUE
4  }
5
6

```

Levá strana (cílová strana) instrukce *Load*.

Jak lze vidět, je zde informace o modulu s ID = 10 a s paměťovým místem LED. Do tohoto místa je možné vložit logickou hodnotu TRUE (rozsvítí LED-ku) nebo FALSE (zhasne LED-ku).

Pravá strana (zdrojová strana) instrukce *Load*.

Obsahuje konstantu. V tomto případě logického typu.

Pozn. autora: Zkratka LED (Luminiscence Emitting Diode) označuje konkrétní typ diody. Proto slovní spojení LED dioda, v podstatě zbytečně obsahuje slovo dioda, protože je již obsaženo v uvedené zkratce. Vzhledem k tomu, že se toto nepřesné spojení běžně používá, uvedl jsem je zde také.

2.2. Natočení pohonu AX do určené pozice

Jednou ze základních operací při práci s robotickým systémem je, že potřebujeme natočit pohon AX-12 do určité pozice. Předpokládejme, že takový pohon nejprve připojíme k počítači pomocí software RoboPlus Manager, pomocí kterého nastavíme tento pohon do režimu Join. (Režim, při kterém se pohon chová jako servo pohon.)

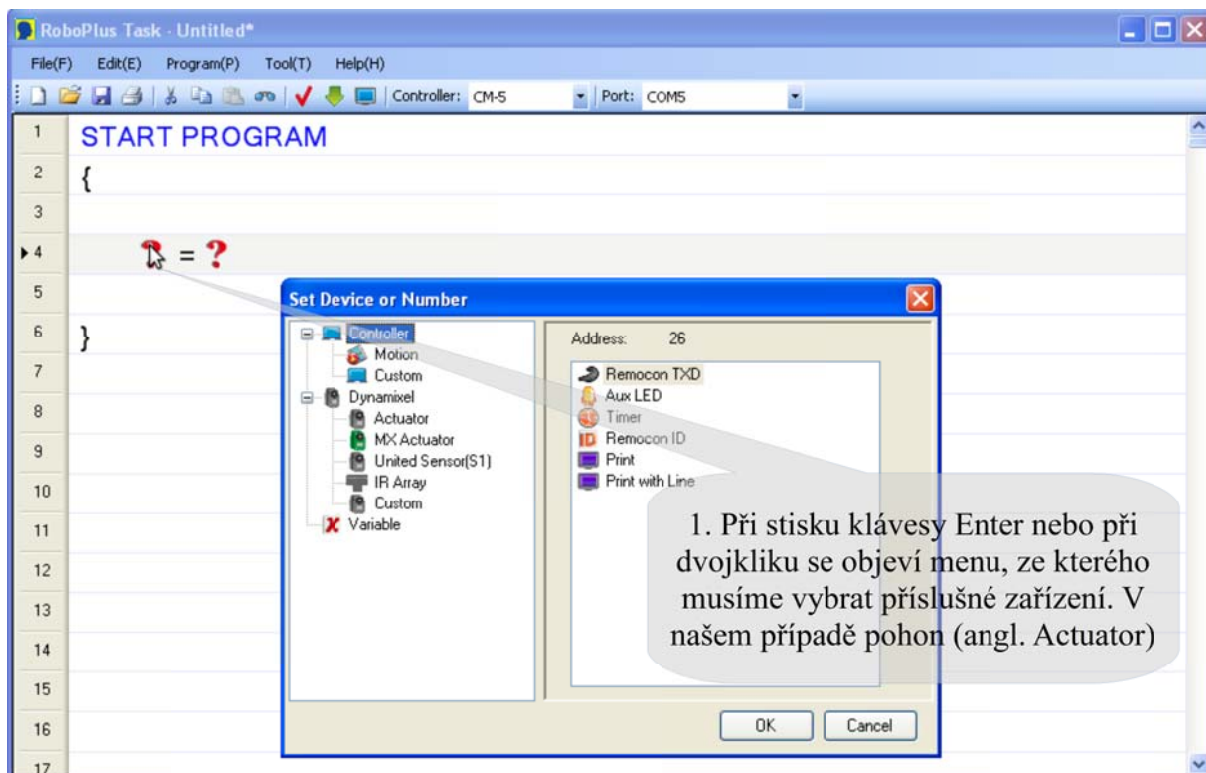
Proto abychom natočili pohon do určité pozice stačí napsat následující program.

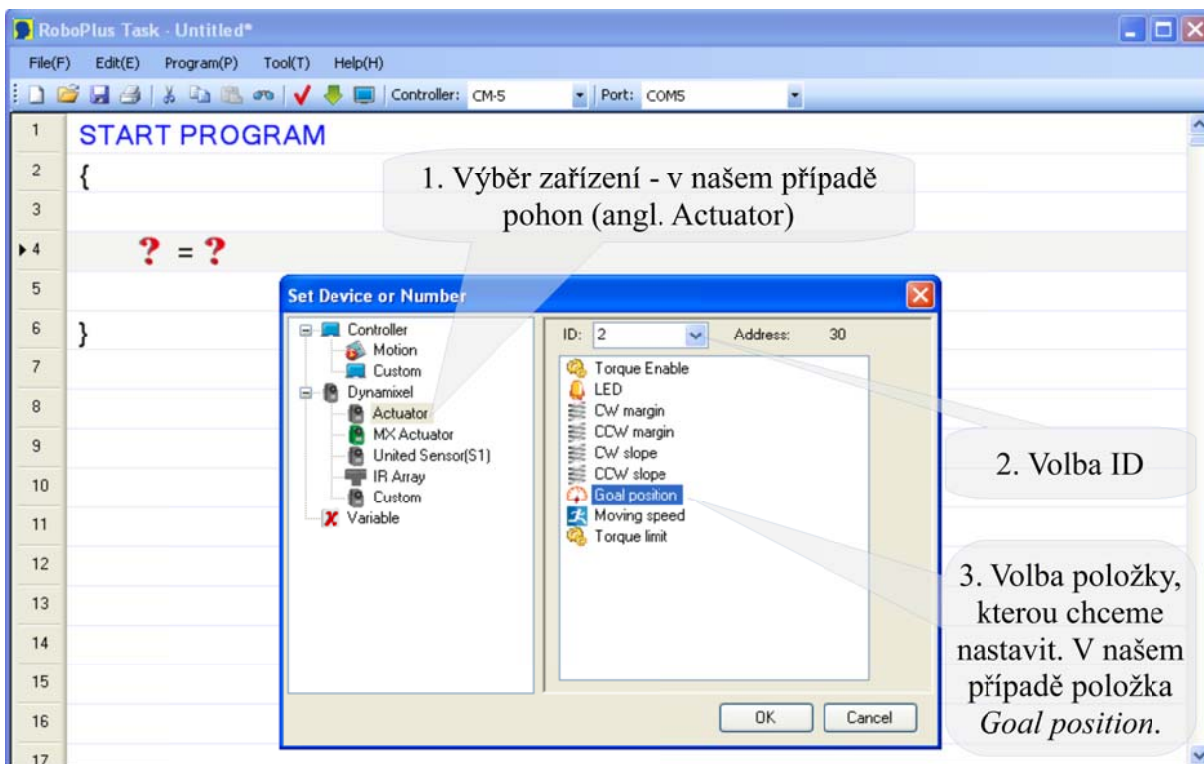
```

1  START PROGRAM
2  {
3
4  ID[2]: Goal position = 717
5
6  }
```

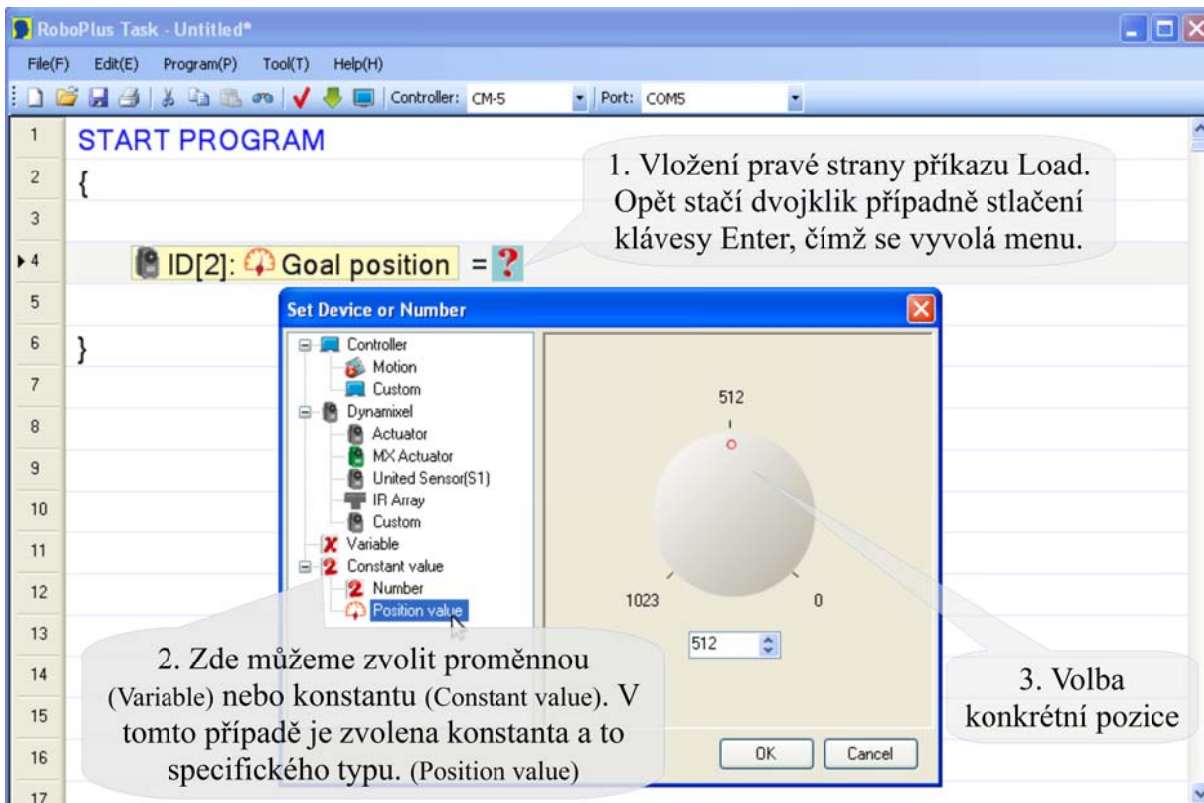
Tento program je vytvořen následujícím postupem:

1. Otevřeme si aplikaci RoboPlus Task a zvolíme si příslušnou řídicí jednotku a komunikační port.
2. Vložíme instrukci *Start* - označující začátek a konec programu. (Viz. Obr. č. 2.5)
3. Vložíme si několik volných řádků mezi složené závorky. (Viz. Obr. č. 2.6)
4. Vložíme instrukci *Load*, podobně jako bod č.2
5. Nejprve vložíme informace o levé straně.





6. Vložíme hodnotu, která reprezentuje pozici, kde se má otočit pohon.



Po spuštění tohoto programu dojde k natočení pohonu s ID = 2 do pozice 717. (Pozice je definována rozsahem celých čísel od 0 do 1023.) Pokud tento pohon se v dané pozici již nachází, tak po spuštění programu se nic nestane.

2.3. Natáčení pohonu podle stisknutého tlačítka

Předchozí program je velmi jednoduchý a neumožňuje natáčení pohonu do jiných směrů. Proto je v této kapitole ukázka programu, který na základě stisknutého tlačítka na řídicí jednotce, natočí pohon AX-12 příslušným směrem.

Zmíněný program může vypadat například takto:

1	START PROGRAM
2	{
3	IF (Button == R)
4	{
5	ID[2]: Goal position = 956
6	}
7	
8	IF (Button == L)
9	{
10	ID[2]: Goal position = 81
11	}
12	}
13	

Poznámka:

Aby se pohon natočil příslušným směrem po spuštění programu, je nutné stisknout tlačítko „R“ nebo „L“ před vlastním spuštěním programu. Tj. před stiskem tlačítka „START“.

V programu je použita instrukce *if*. Tato instrukce patří do skupiny Condition (podmínkové instrukce). Instrukce *if* slouží k větvení programu, tj. umožňuje v algoritmech reagovat na různé situace. Proto se tato instrukce skládá minimálně ze dvou částí.

První část instrukce if, reprezentuje podmínka, která může být splněna nebo nesplněna. Tato podmínka může být jednoduchá (např. $a > 1$) nebo složená (použití několika jednoduchých podmínek).

Druhá část instrukce if, reprezentuje tělo instrukce *if*, jenž je ohraničeno složenými závorkami „{, a „}““. Instrukce umístěné mezi těmito složenými závorkami se vykonají pouze když je podmínka pravdivá.

Třetí část instrukce if, reprezentuje tělo instrukce *if*, jenž je také ohraničeno složenými závorkami „{, a „}““. Instrukce umístěné mezi těmito dalšími složenými závorkami se vykonají pouze když je podmínka není pravdivá. Tato část je však nepovinná, proto se nemusí používat. (V naší ukázce tato část instrukce *if* použita není.)

V této ukázce je základním blokem algoritmu instrukce if, jejíž podmínka „kontroluje“ zda tlačítko stisknuto bylo či nikoliv.

```

1  START PROGRAM
2  {
3  IF ( Button == R )
4  {
5  ID[2]: Goal position = 956
6  }
7  }
8

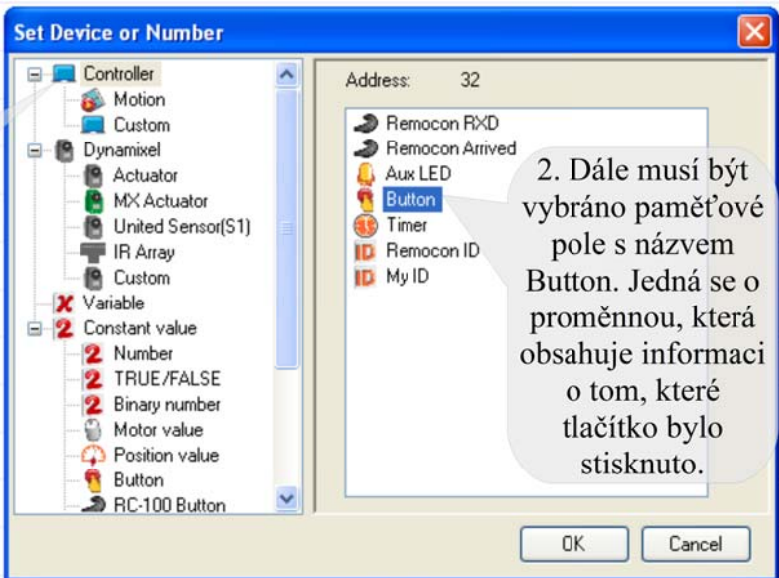
```

Instrukce if se vkládá podobně jako ostatní instrukce. Jen při tvorbě podmínky, která rozhoduje zda-li bylo stisknuto příslušné tlačítko musíme vybrat z menu zařízení položku Controller (řídící jednotka).

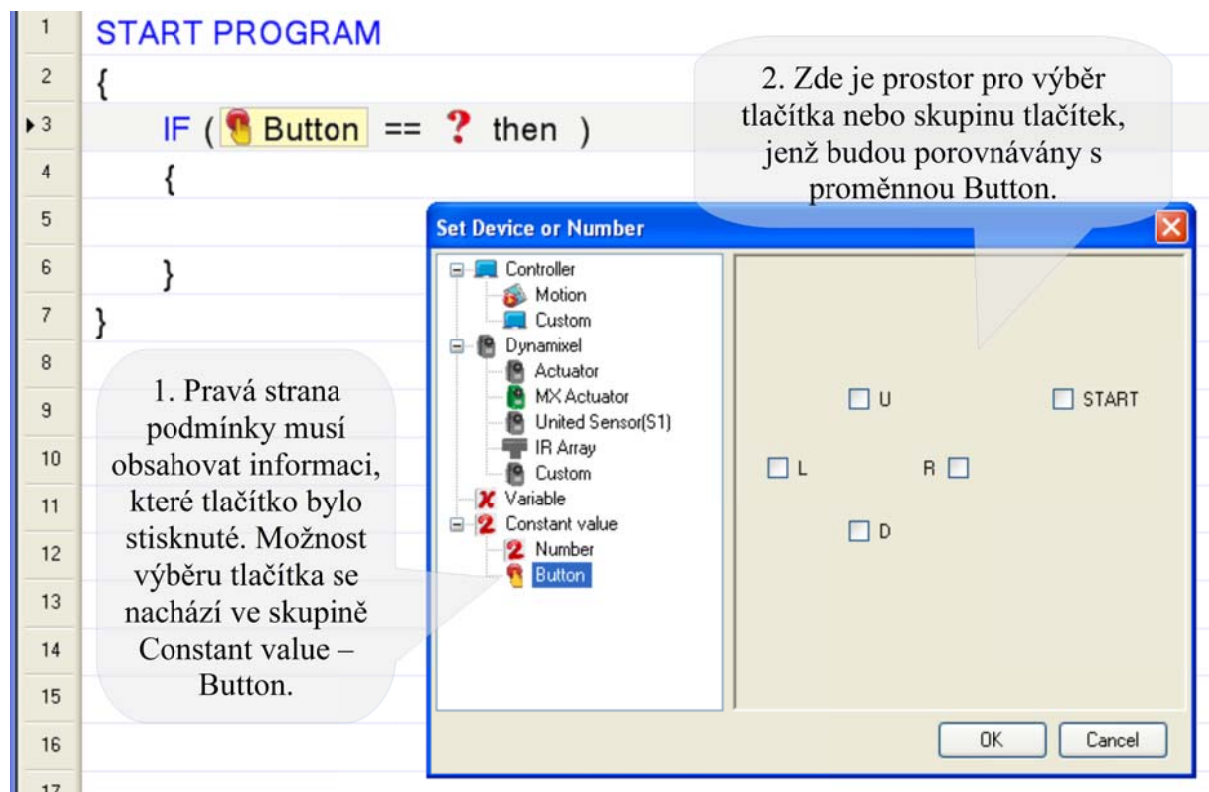
```

1  START PROGRAM
2  {
3  IF ( ? == ? then )
4  {
5
6  }
7  }
8
9
10
11
12
13
14
15
16
17

```



Při definování pravé strany podmínky je možné zvolit příslušné tlačítko, případně kombinaci stisknutých tlačítek. Výběr zvoleného tlačítka případně kombinace tlačítek lze provést označením zaškrťovacího políčka typu Checkbox.



1. Pravá strana podmínky musí obsahovat informaci, které tlačítko bylo stisknuté. Možnost výběru tlačítka se nachází ve skupině Constant value – Button.

2. Zde je prostor pro výběr tlačítka nebo skupinu tlačítek, jež budou porovnávány s proměnnou Button.

Tímto způsobem lze definovat další podmínky a tím je možné program rozšiřovat o další možnosti na které bude reagovat.

Uvedený program sice funguje tak, že na základě stisku tlačítka dojde k natočení pohonu příslušným směrem, avšak způsob jakým to lze provést je nestandardní a málo uživatelsky přívětivý. Tj. abychom nemuseli držet stisknuté tlačítko před spuštěním programu a zároveň aby program po spuštění kdykoliv zareagoval na naše stisknutí, je nutné program doplnit o další instrukci.

Instrukce, která toto umožní se nazývá ENDLESS LOOP. Jak jde vidět z názvu, jedná se o instrukci reprezentující nekonečnou smyčku. To znamená, že když algoritmus vstoupí do této smyčky, již se samostatně neukončí a program lze ukončit pouze stisknutím tlačítka MODE na řídicí jednotce.

```
1 START PROGRAM
2 {
3     ENDLESS LOOP
4     {
5         IF ( Button == R )
6         {
7             ID[2]: Goal position = 956
8         }
9
10        IF ( Button == L )
11        {
12            ID[2]: Goal position = 81
13        }
14    }
15 }
16
```


3. Manipulátory / Vozíky

3.1. Čtyř-kolový vozík



PRAKTICKÁ ÚLOHA

Sestavení čtyř-kolového vozíku se sensorickým modulem AX-S1. Tento modul je možné využít pro interakci s okolím. Např. pro detekci překážky před vozíkem, případně detekci hrany stolu.

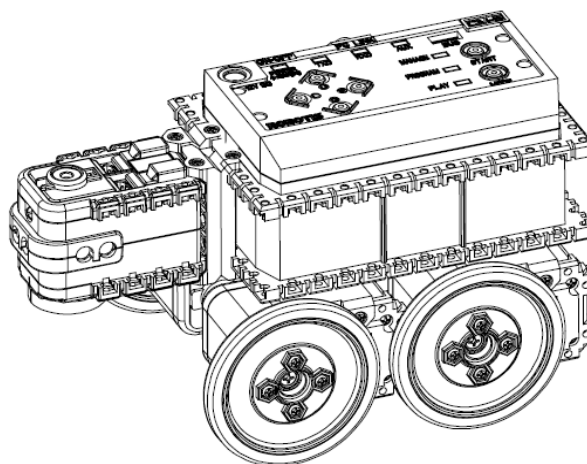


ČAS K SESTAVENÍ: 90 minut

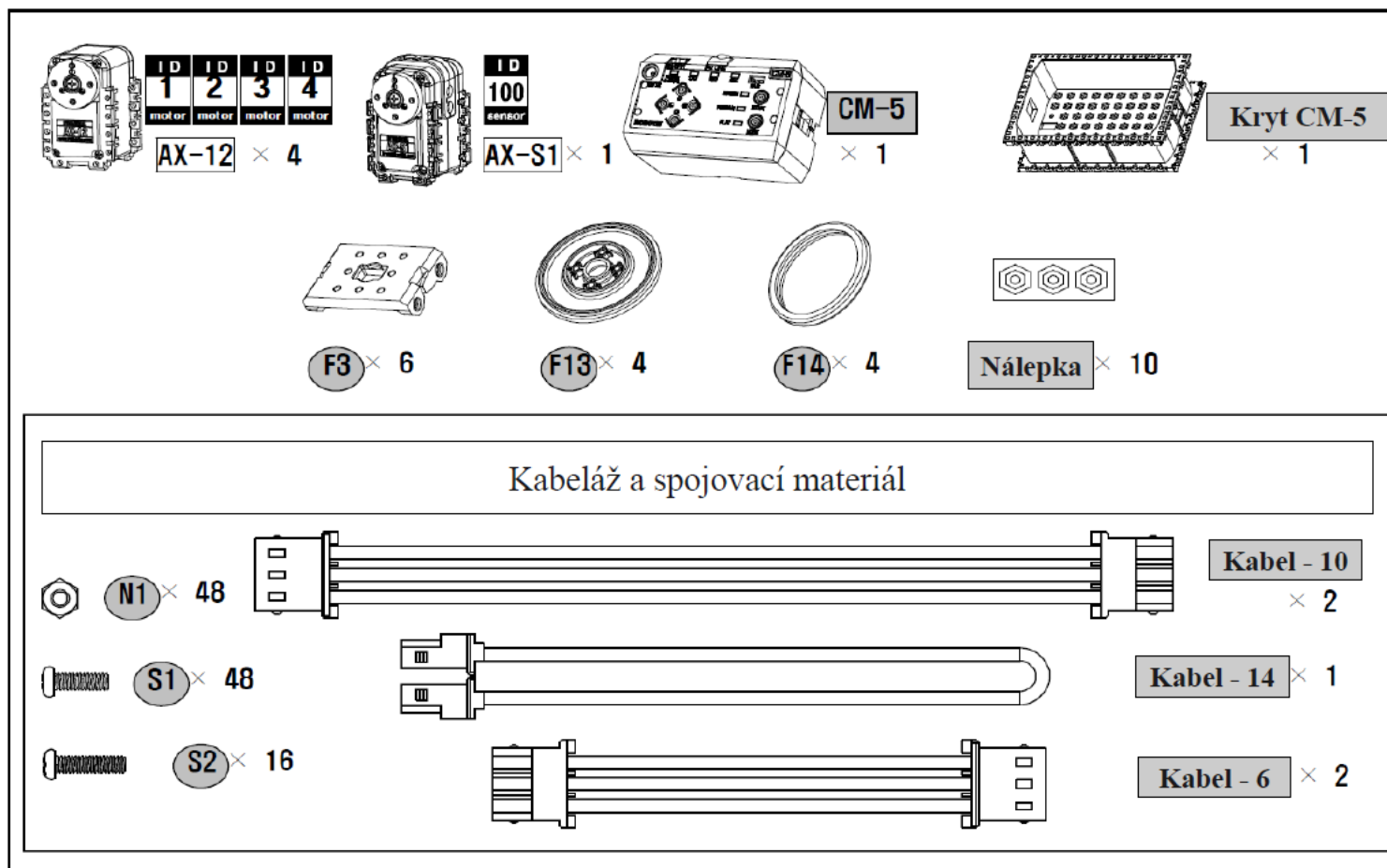


CÍL

Pomocí této kapitoly čitatel může sestavit konkrétní model robotického vozíku se sensorickým modulem AX-S1.



Seznam potřebných dílů:

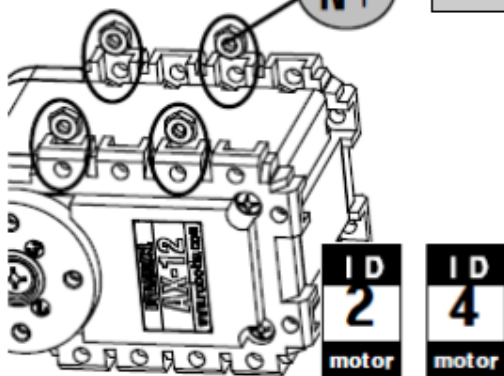


Postup sestavení robota:

Krok 1 – Sestavení dvou levých kol

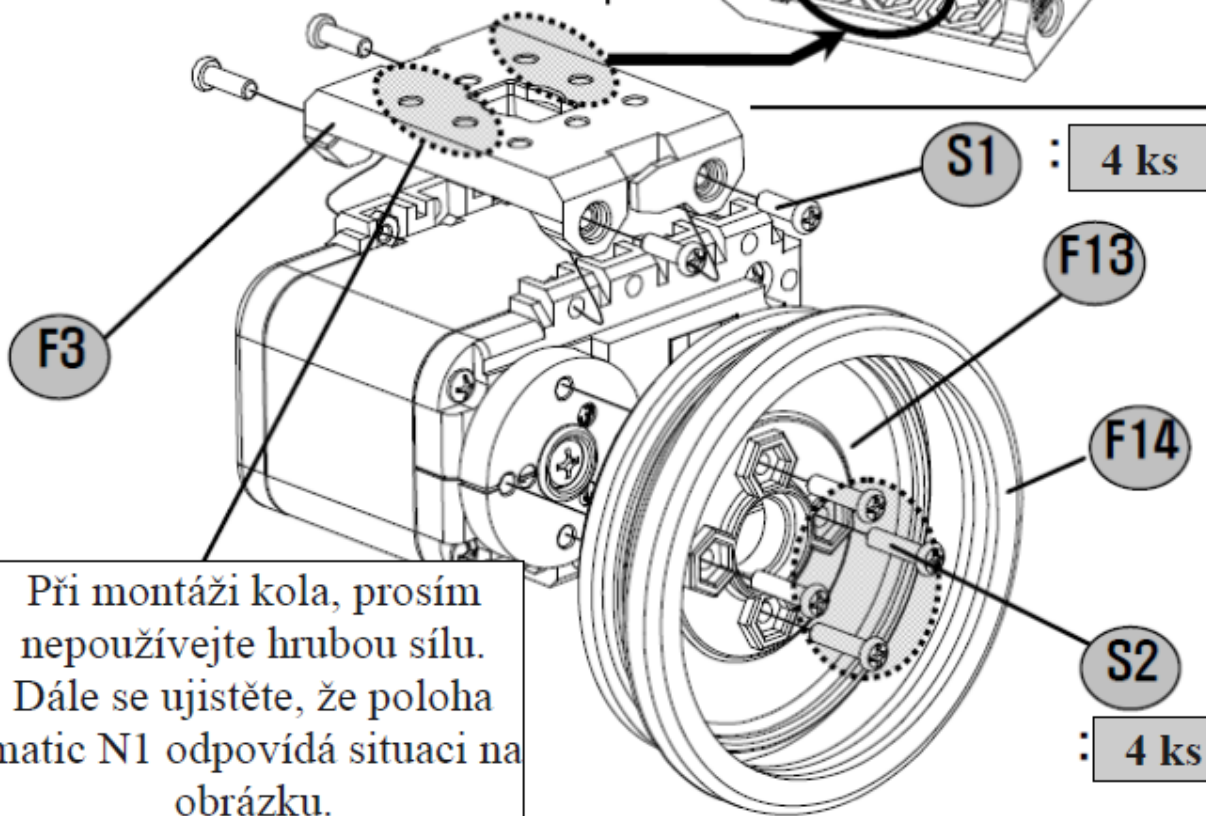
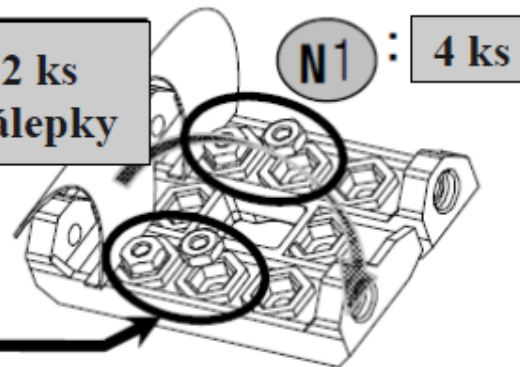
Sestavte kolo s použitím pohonů ID 2 a ID 4

Vložte matice N1 : 4 ks



Vložte matice N1 do vyznačených otvorů a pak je přelepte lepicí páskou.

2 ks nálepky

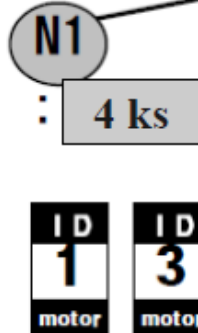


Při montáži kola, prosím nepoužívejte hrubou sílu. Dále se ujistěte, že poloha matic N1 odpovídá situaci na obrázku.

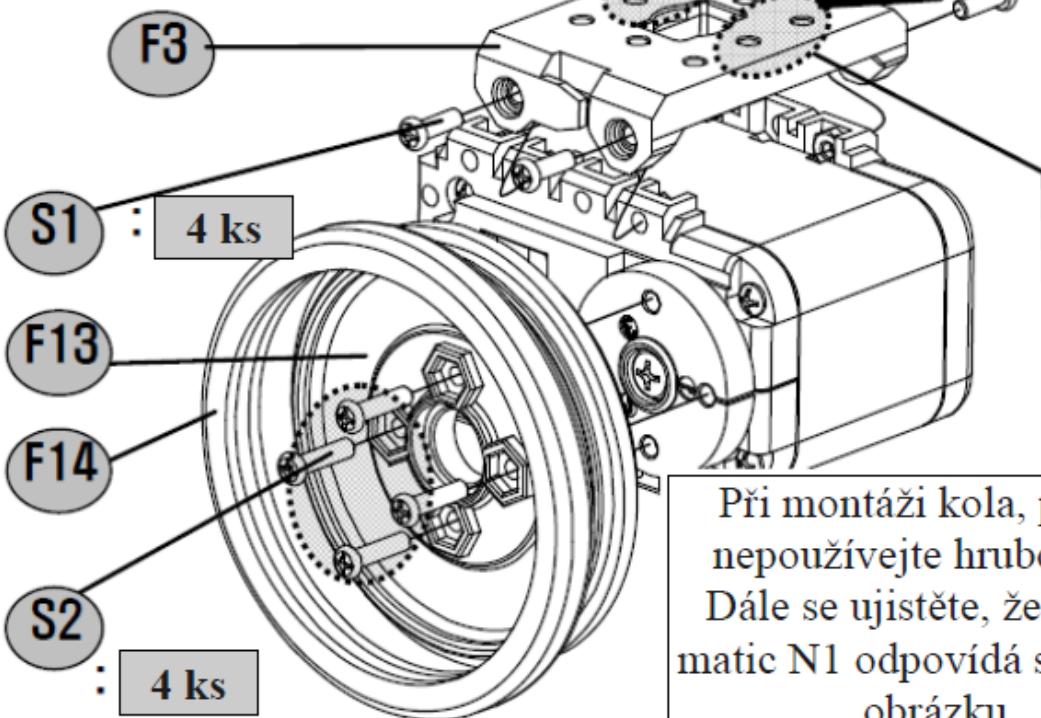
Krok 2 – Sestavení dvou pravých kol

Sestavte kolo s použitím pohonů ID 1 a ID 3

Vložte matice

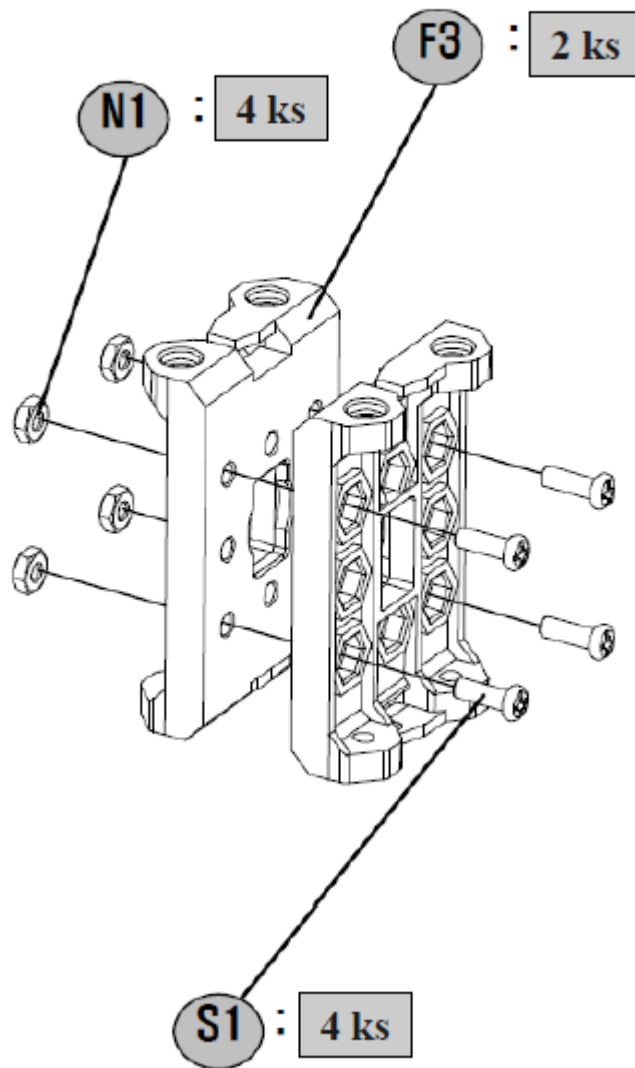


Vložte matice N1 do vyznačených otvorů a pak je přelepte lepicí páskou.



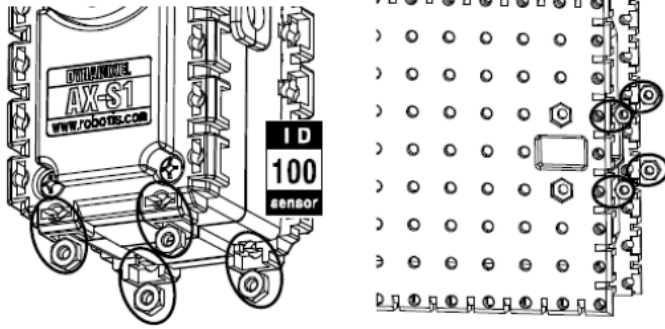
Při montáži kola, prosím nepoužívejte hrubou sílu. Dále se ujistěte, že poloha matic N1 odpovídá situaci na obrázku.

Krok 3 – Sestavení spojky

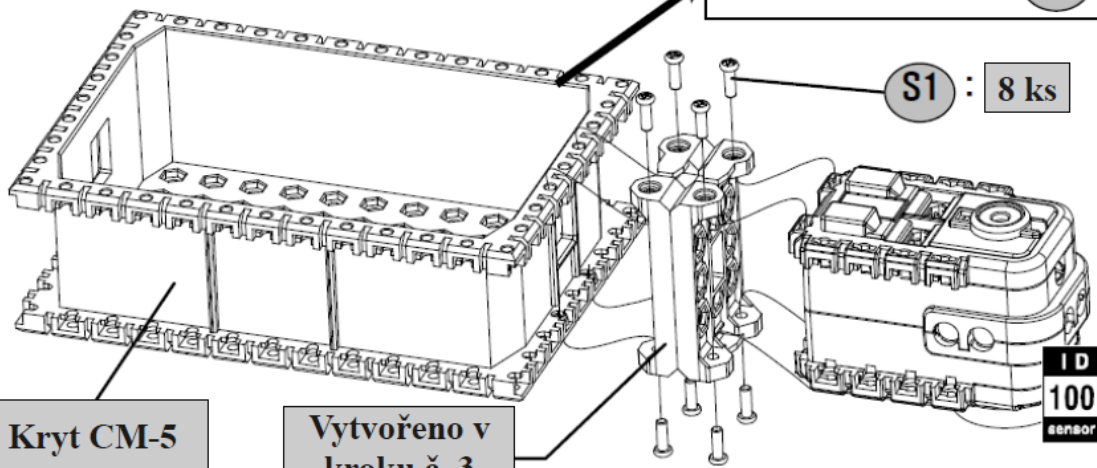
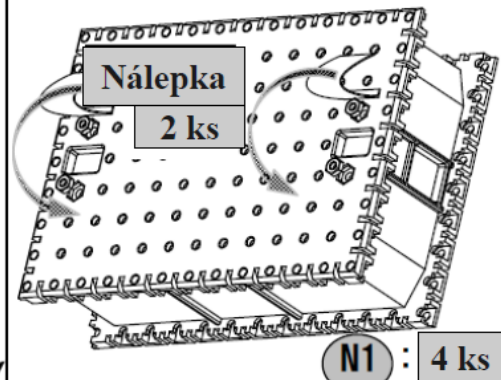


Krok 4 – Připojení snímací hlavy

Vložte matice N1 : 8 ks



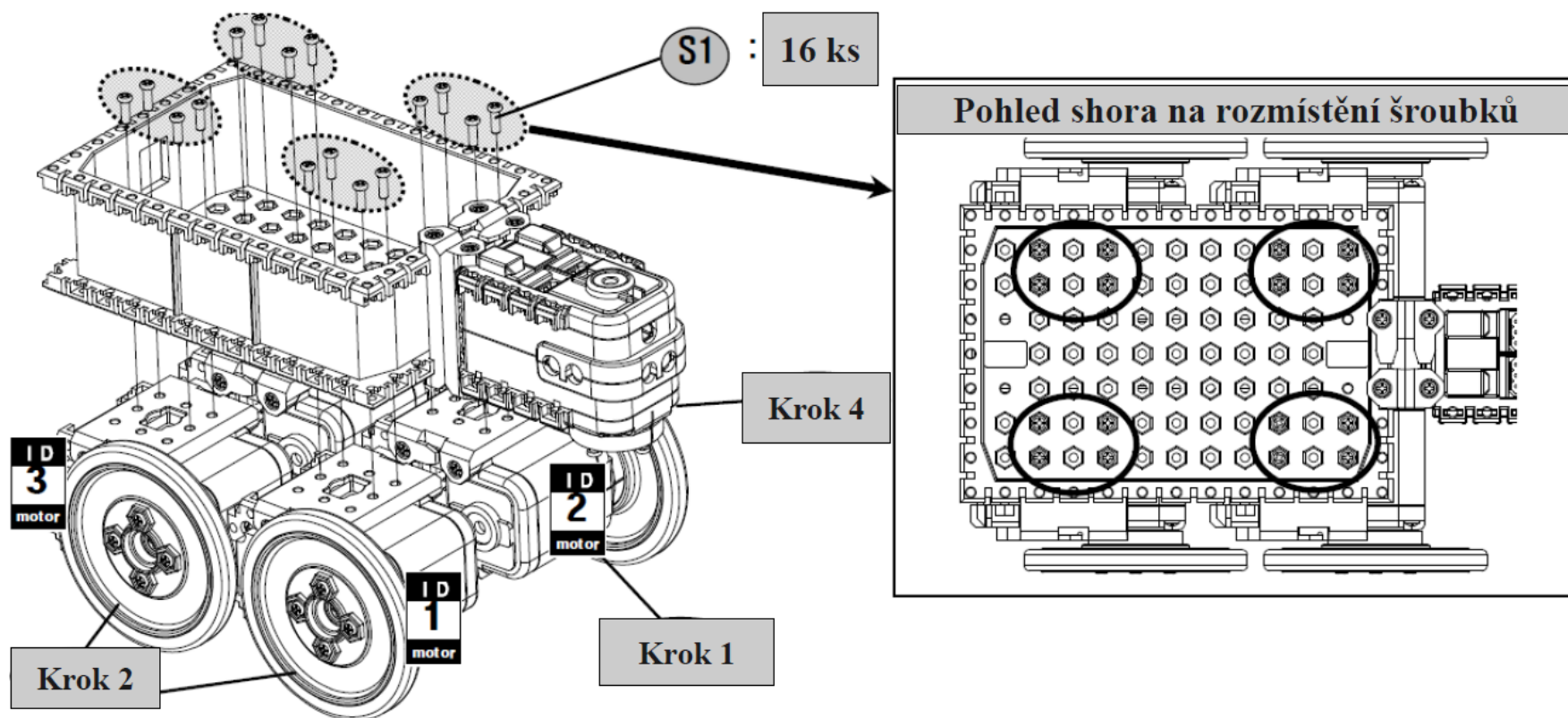
Vložte matice N1 do vyznačených otvorů a pak je přelepte lepicí páskou.



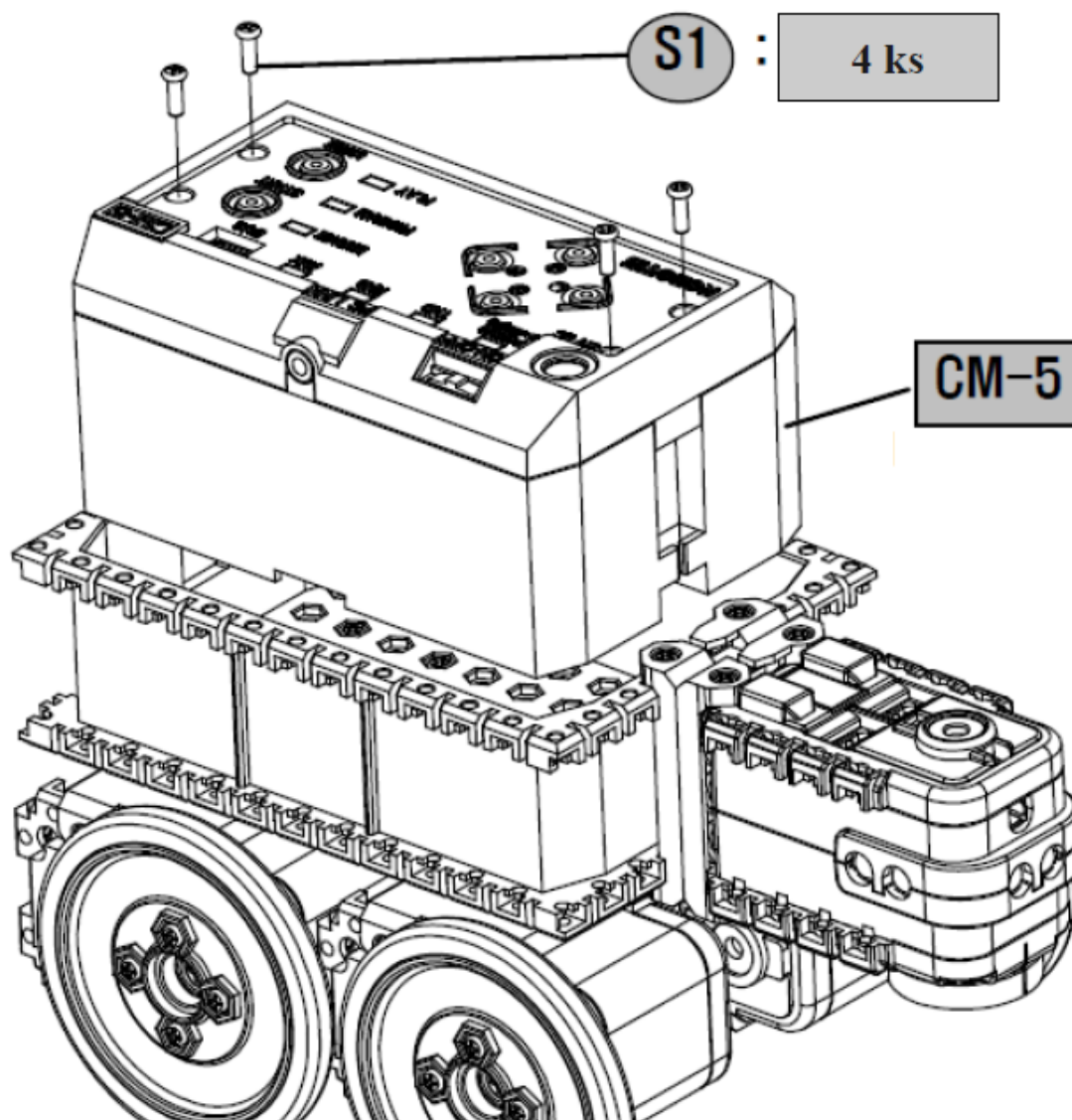
Kryt CM-5

Vytvořeno v kroku č. 3

Krok 6 – Kompletace vozidla



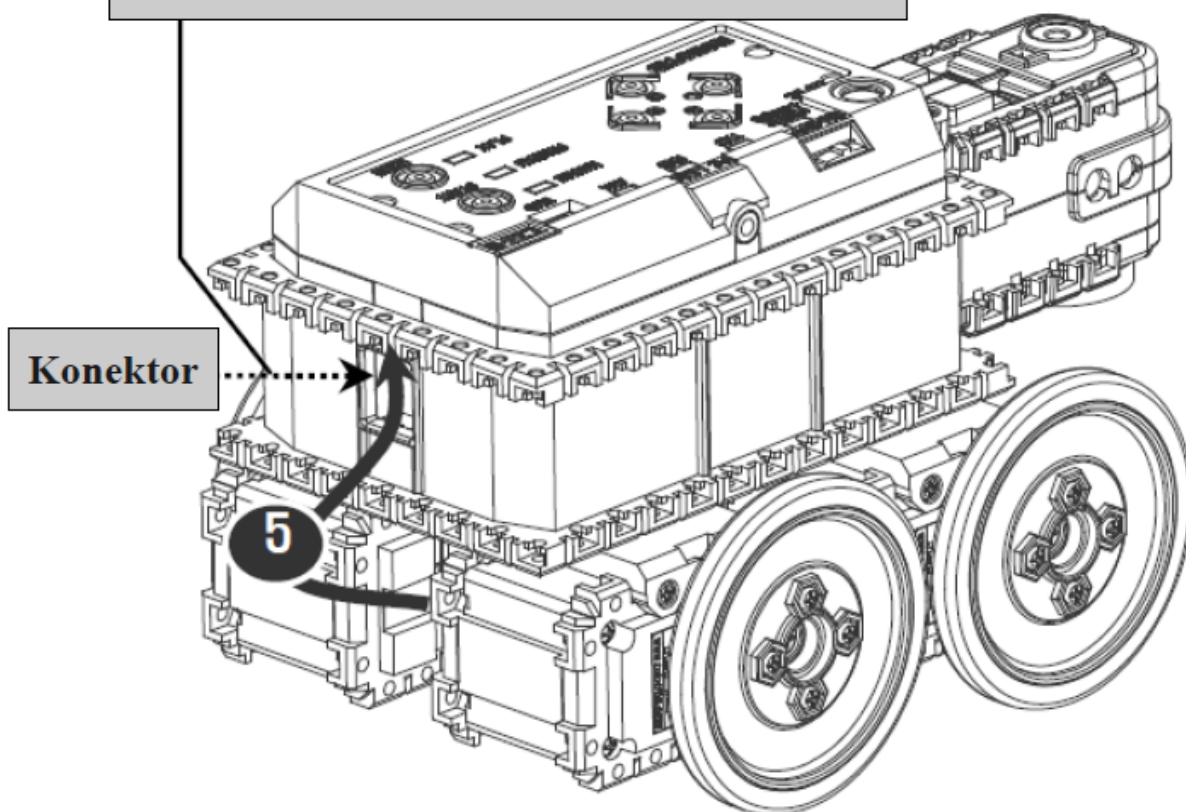
Krok 7 – Přichycení řídicí jednotky



Toto přichycení, pro funkčnost modelu, není nutná.

Krok 8 – Propojení řídicí jednotky s vozidlem

Tento konektor může být propojen s řídicí jednotkou i shora.
Tj. nemusí být využitý konektor, který je
naznačen na obrázku.



3.2. Čtyř-kolový vozík s manipulačním ramenem



PRAKTICKÁ ÚLOHA

Na první pohled se může zdát, že tato úloha je stejná jako úloha v kapitole 3.1, avšak není tomu tak. Základním rozdílem je, že v této úloze bude robotický vozík doplněn o manipulační rameno. Aby toto bylo možné musí být změněna také konstrukce celého vozíku. Proto sestavovaný robotický vozík, v této úloze, je konstrukčně odlišný v několika bodech. Tudíž nelze použít sestavený vozík z předchozí kapitoly a pouze k němu doplnit manipulační rameno.

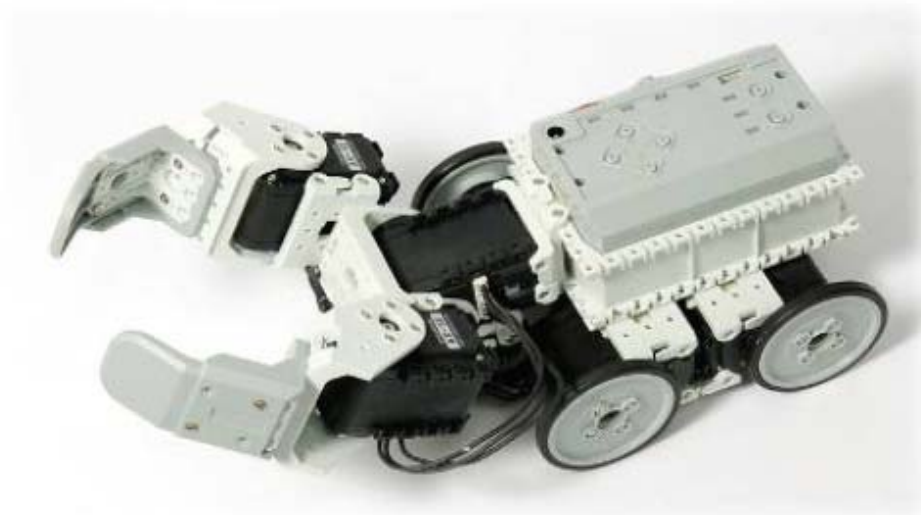


ČAS K SESTAVENÍ: 120 minut

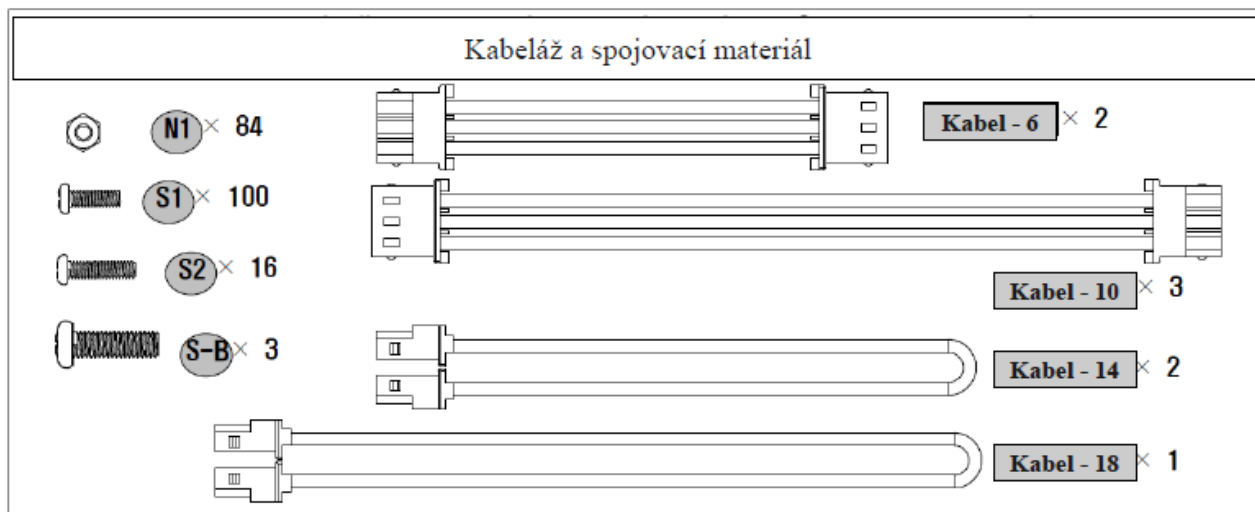
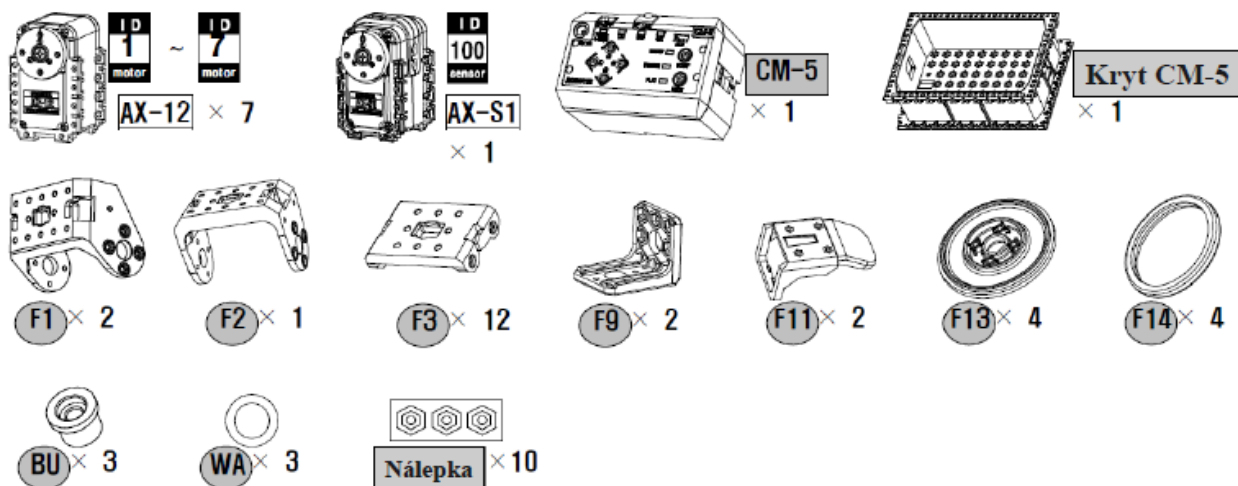


CÍL

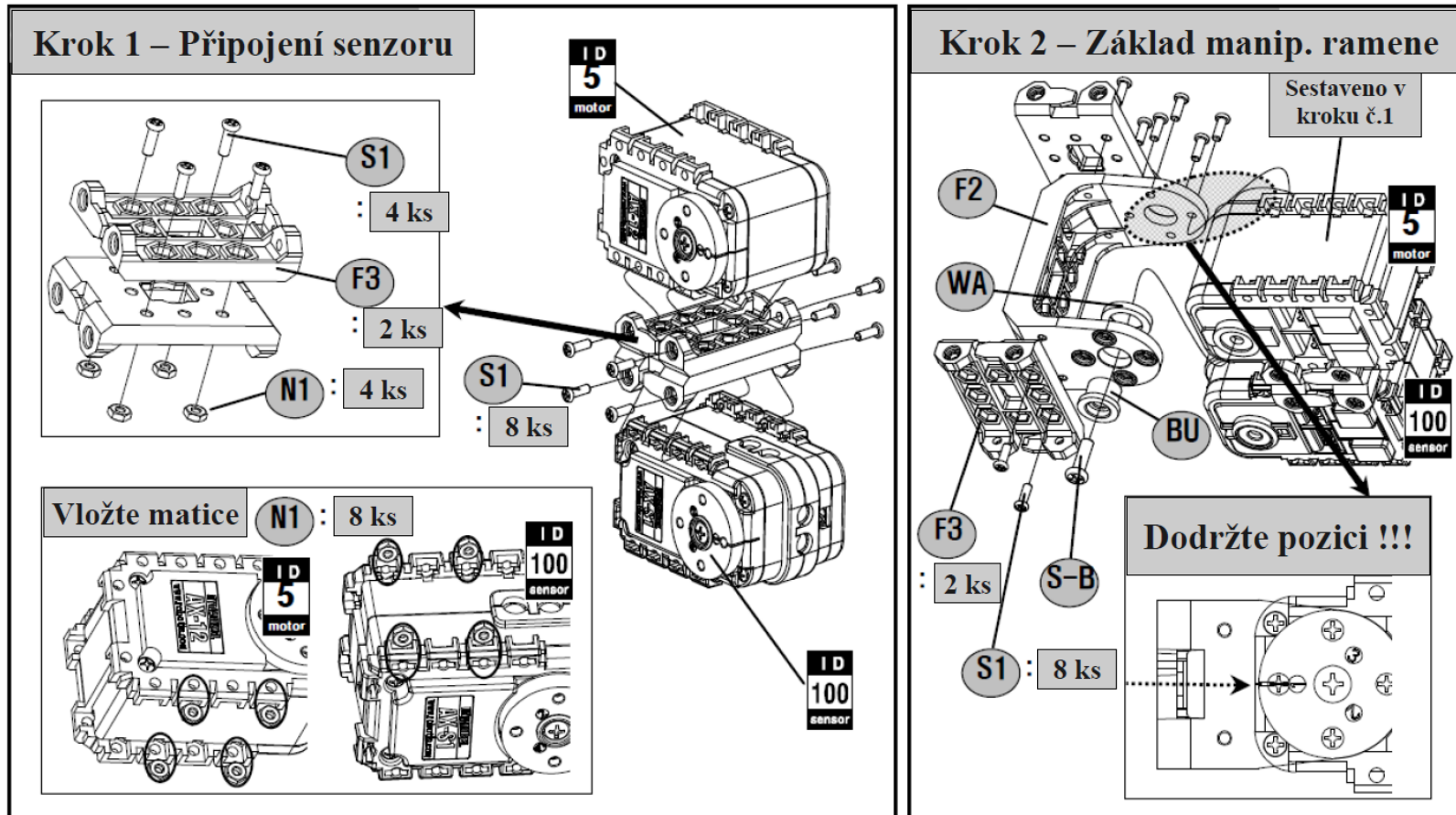
Cílem této úlohy je ukázat použití pohonů AX-12 jak v režimu *wheel*, tak i v režimu *join*. Dále je snaha ukázat potřebu jiné konstrukce robotického vozíku, má-li mít k dispozici manipulační rameno.



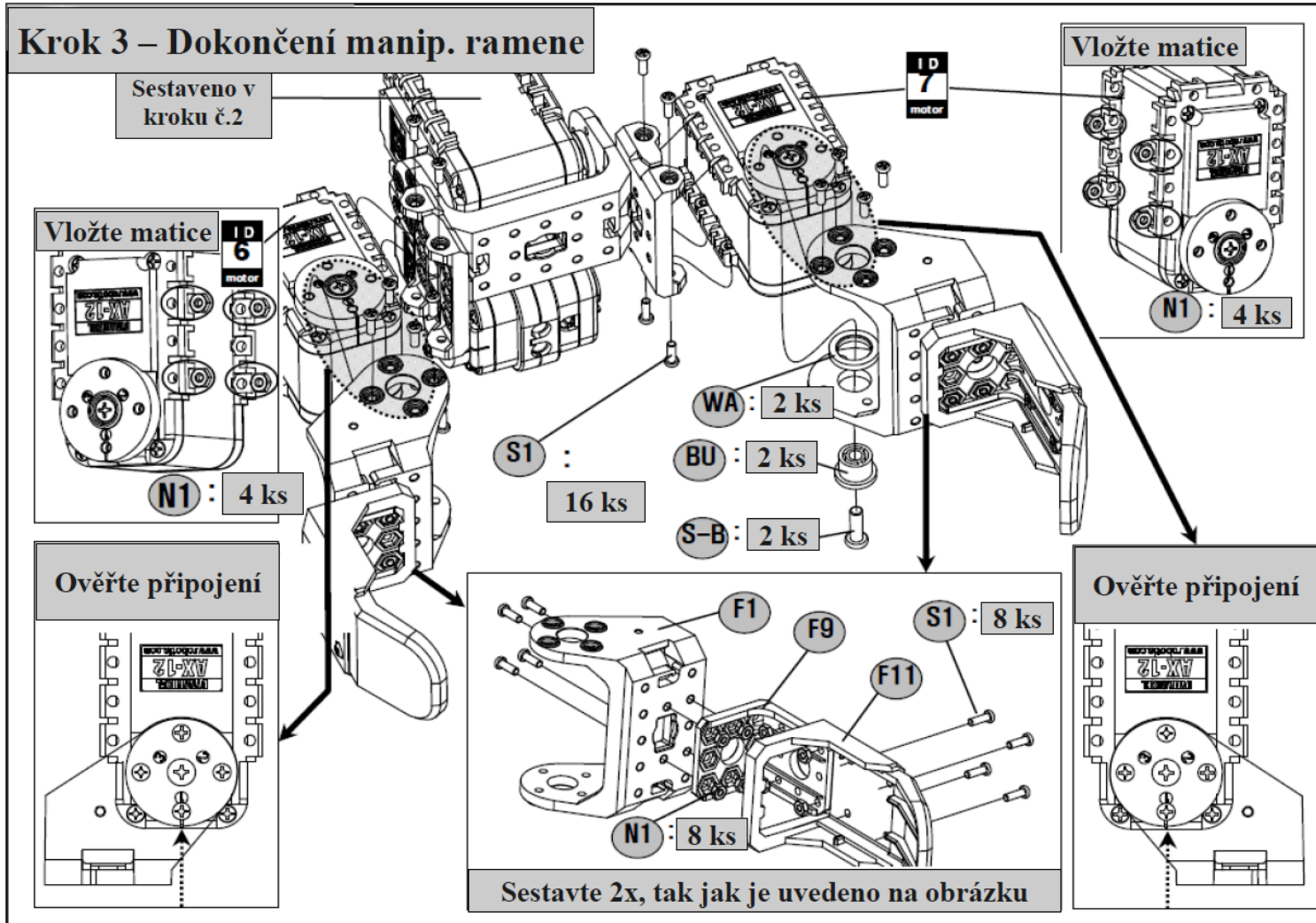
Seznam potřebných dílů:



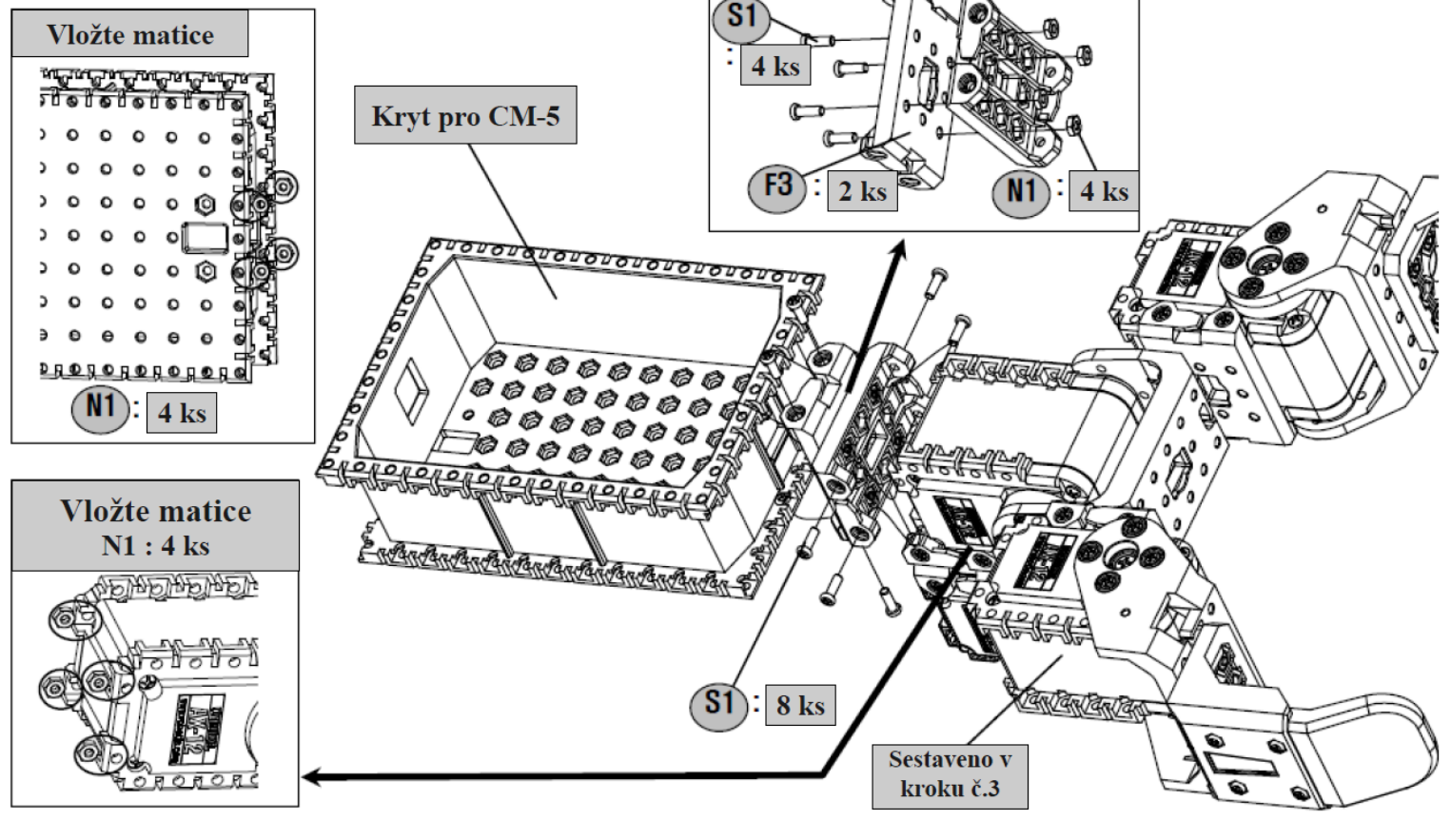
Postup sestavení robota:



Krok 3 – Dokončení manip. ramene



Krok 4 – Nosný skelet konstrukce robota



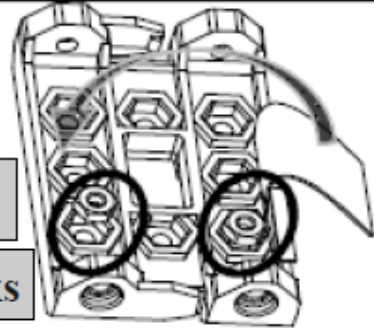
Krok 5 – Pohony na pravé straně

Vložte matice N1 do vyznačených otvorů a pak je přelepte lepicí páskou.

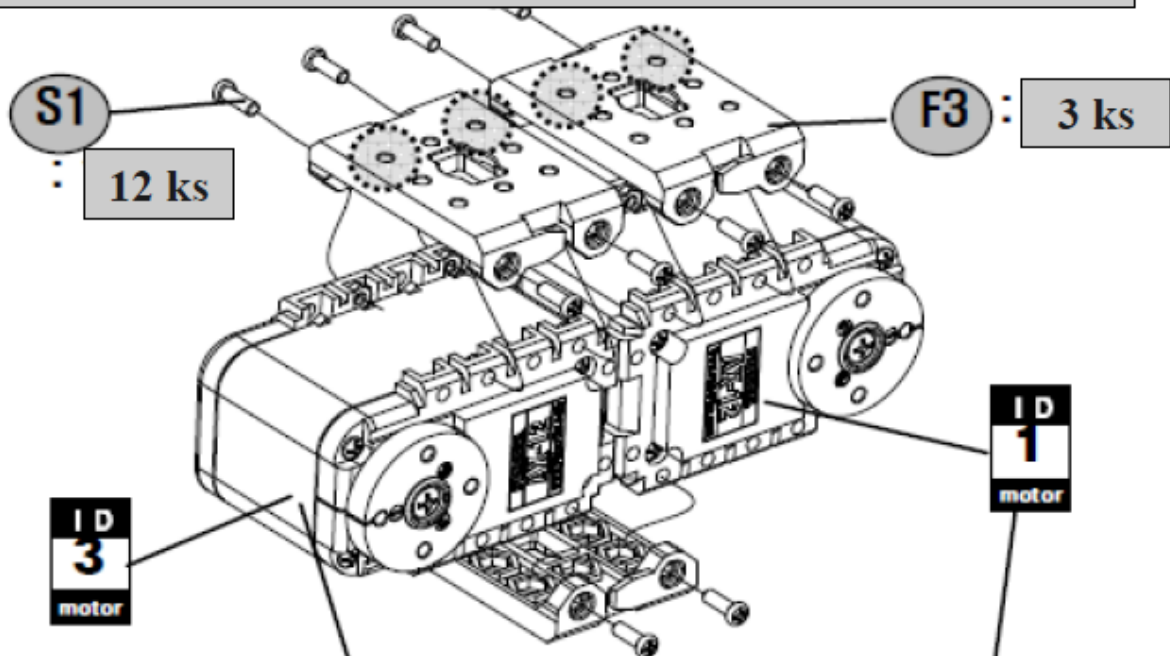
Nálepka

Tuto operaci proveďte 2x

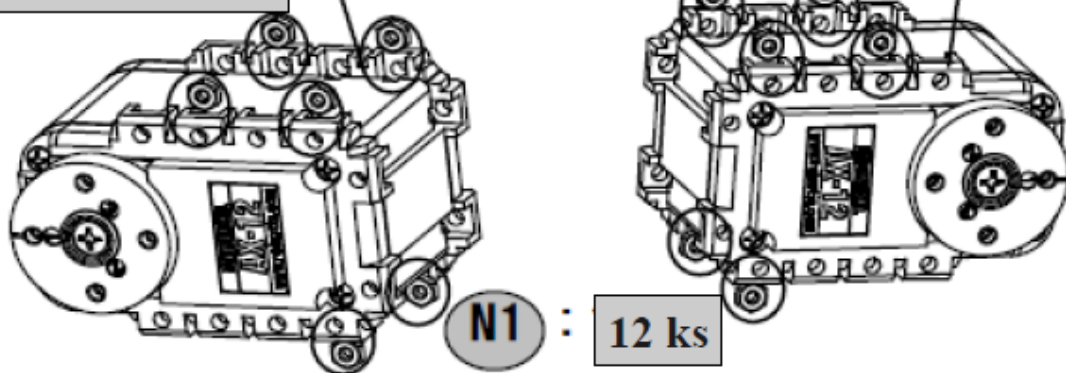
N1 : 2 ks



Dbejte na dodržení jednotlivých pozic, jak je níže zobrazeno.



Vložte matice



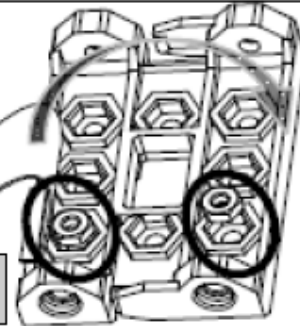
Krok 6 – Pohony na levé straně

Vložte matice N1 do vyznačených otvorů a pak je přelepte lepicí páskou.

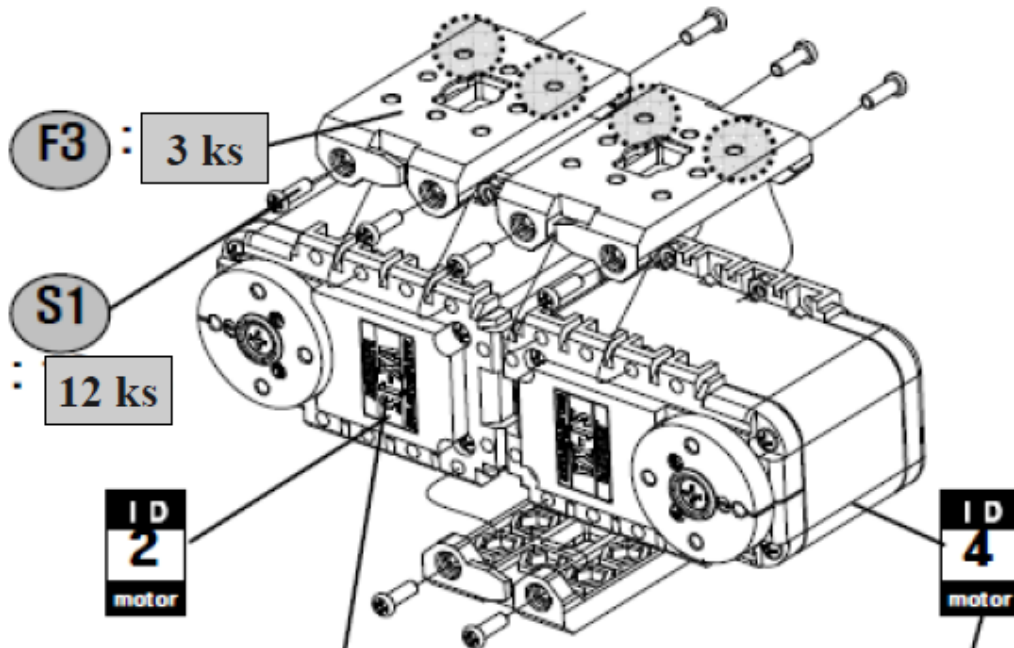
Nálepka

Tuto operaci proveďte 2x

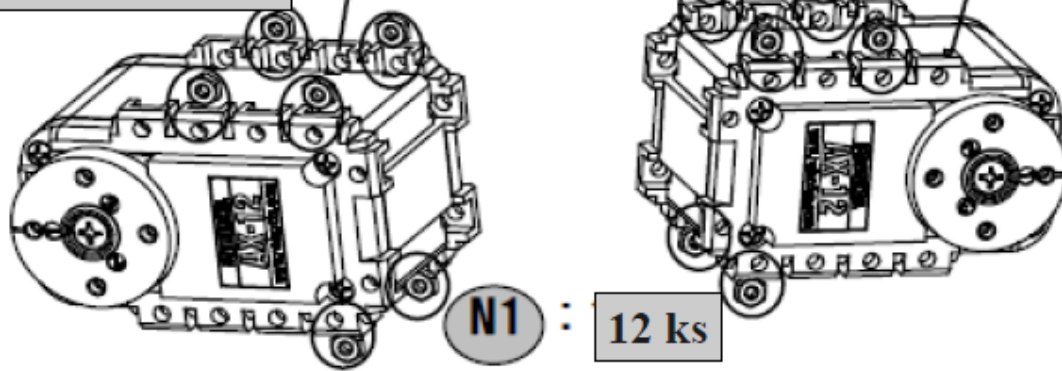
N1 : 2 ks



Dbejte na dodržení jednotlivých pozic, jak je níže zobrazeno.

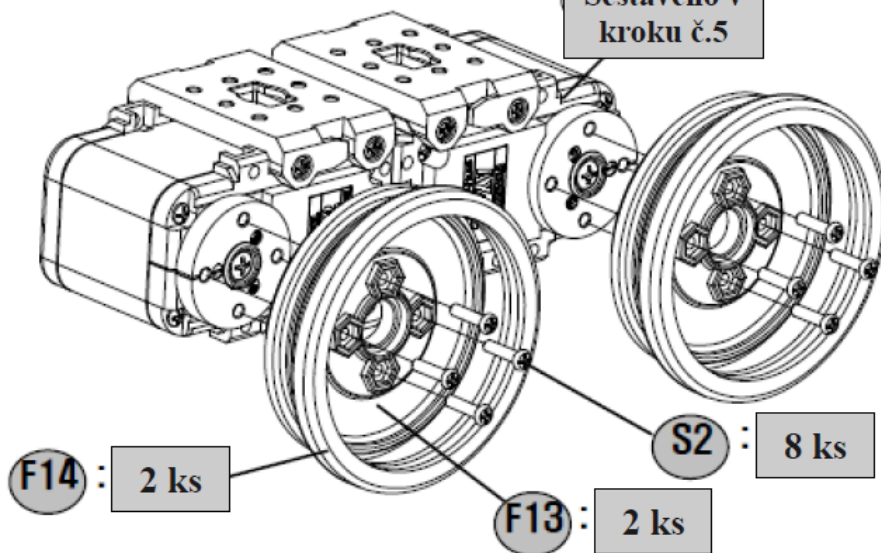


Vložte matice



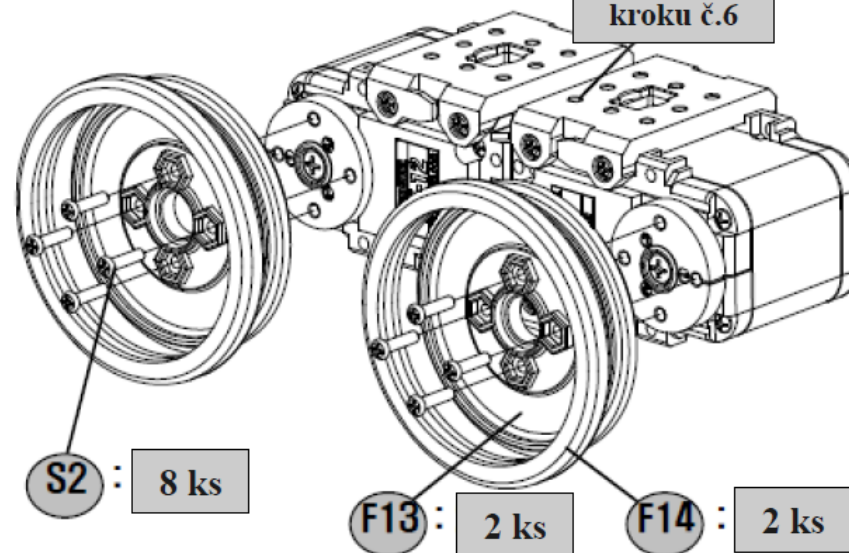
Krok 7 – Kola na pravé straně

Sestaveno v kroku č.5



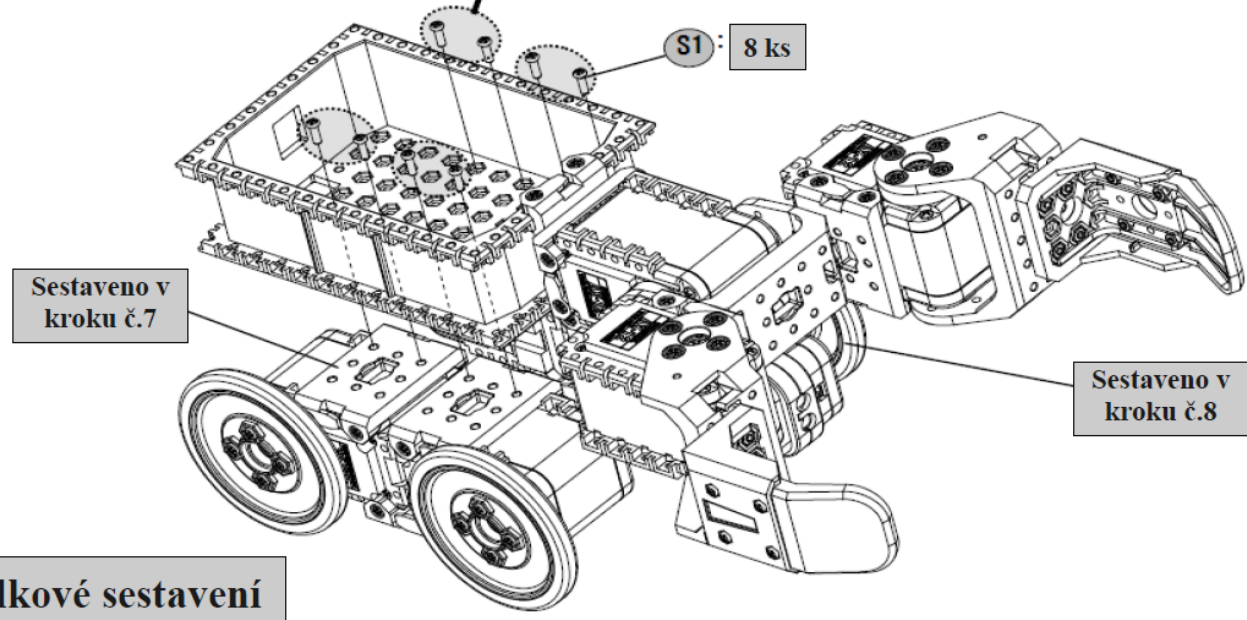
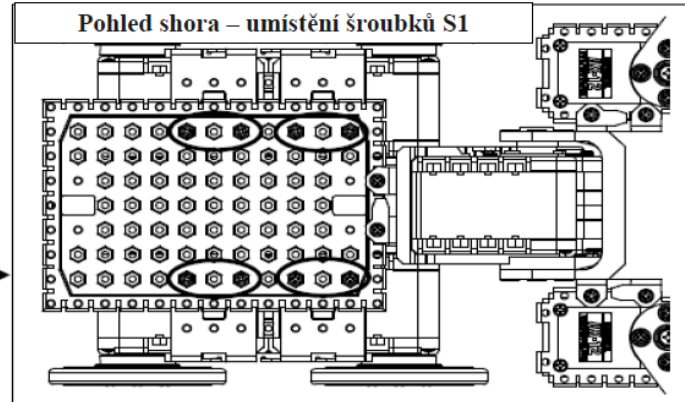
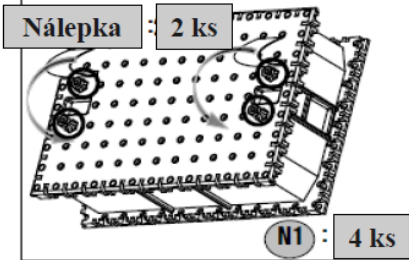
Krok 8 – Kola na levé straně

Sestaveno v kroku č.6



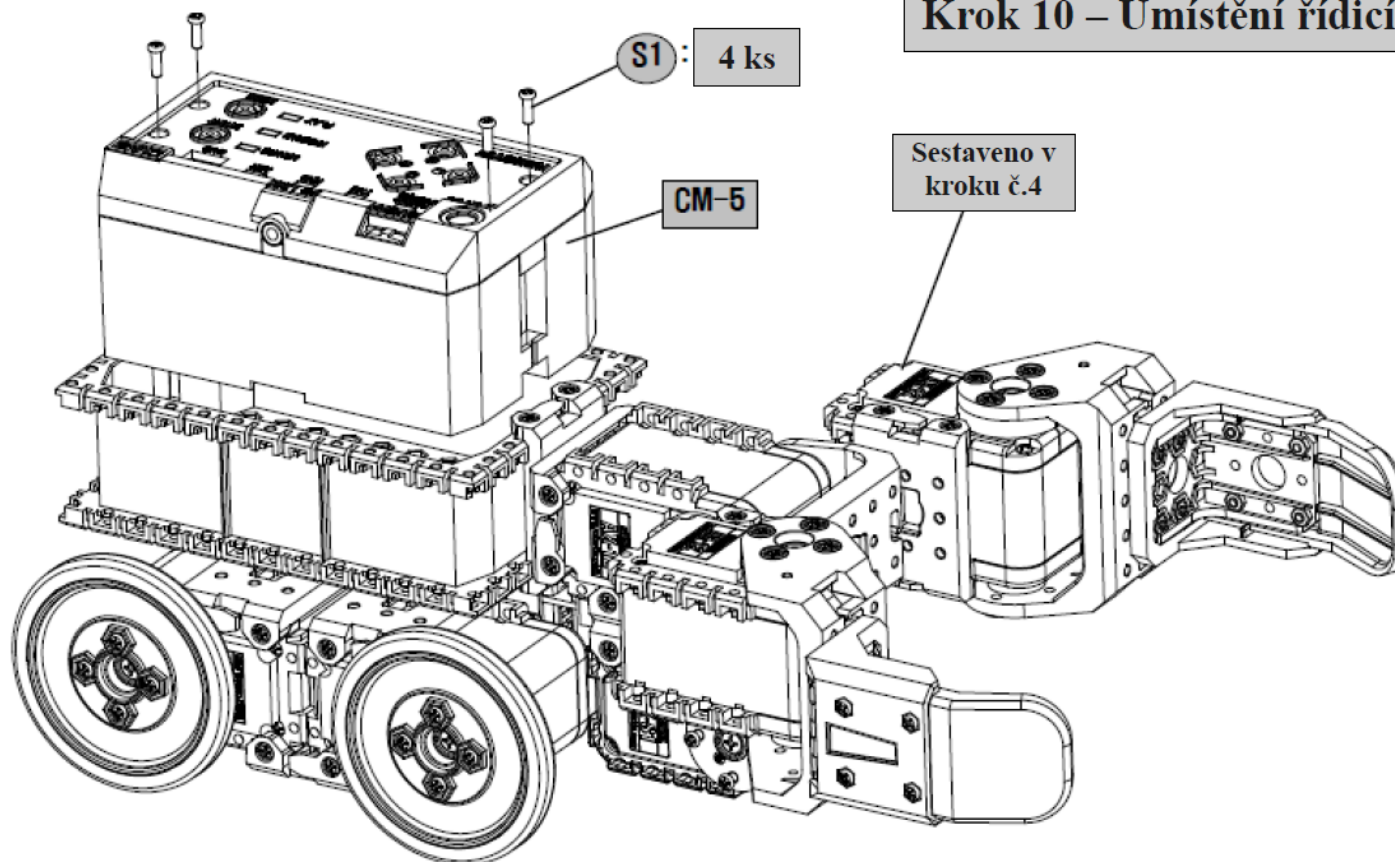
INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

Vložte matice N1 do vyznačených otvorů a pak je přelepte lepicí páskou.



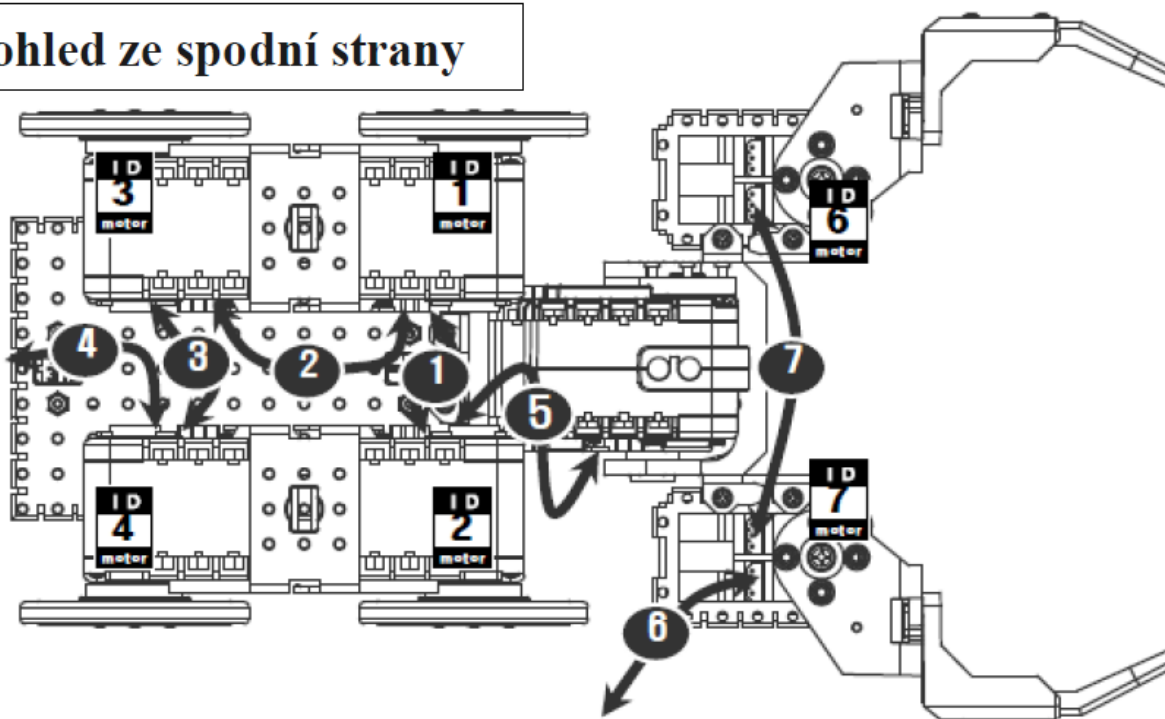
Krok 9 – Celkové sestavení

Krok 10 – Umístění řídicí jednotky



Krok 11 – Propojení pohonů I. část

Pohled ze spodní strany

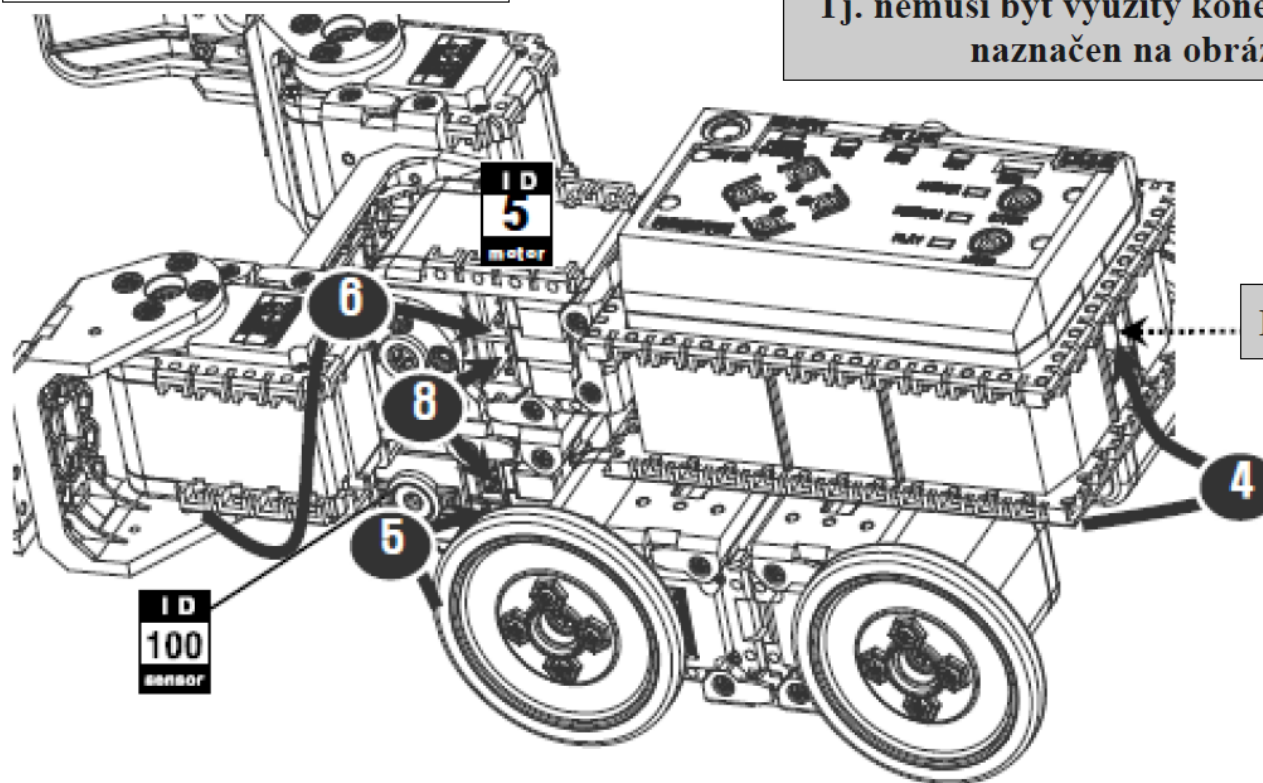


Tabulka jednotlivých propojení

ID 1 motor	← 1 →	ID 2 motor	Kabel - 6
ID 1 motor	← 2 →	ID 3 motor	Kabel - 10
ID 3 motor	← 3 →	ID 4 motor	Kabel - 10
ID 4 motor	← 4 →	CM-5	Kabel - 18
ID 2 motor	← 5 →	ID 100 sensor	Kabel - 14
ID 5 motor	← 6 →	ID 7 motor	Kabel - 14
ID 6 motor	← 7 →	ID 7 motor	Kabel - 10
ID 5 motor	← 8 →	ID 100 sensor	Kabel - 6

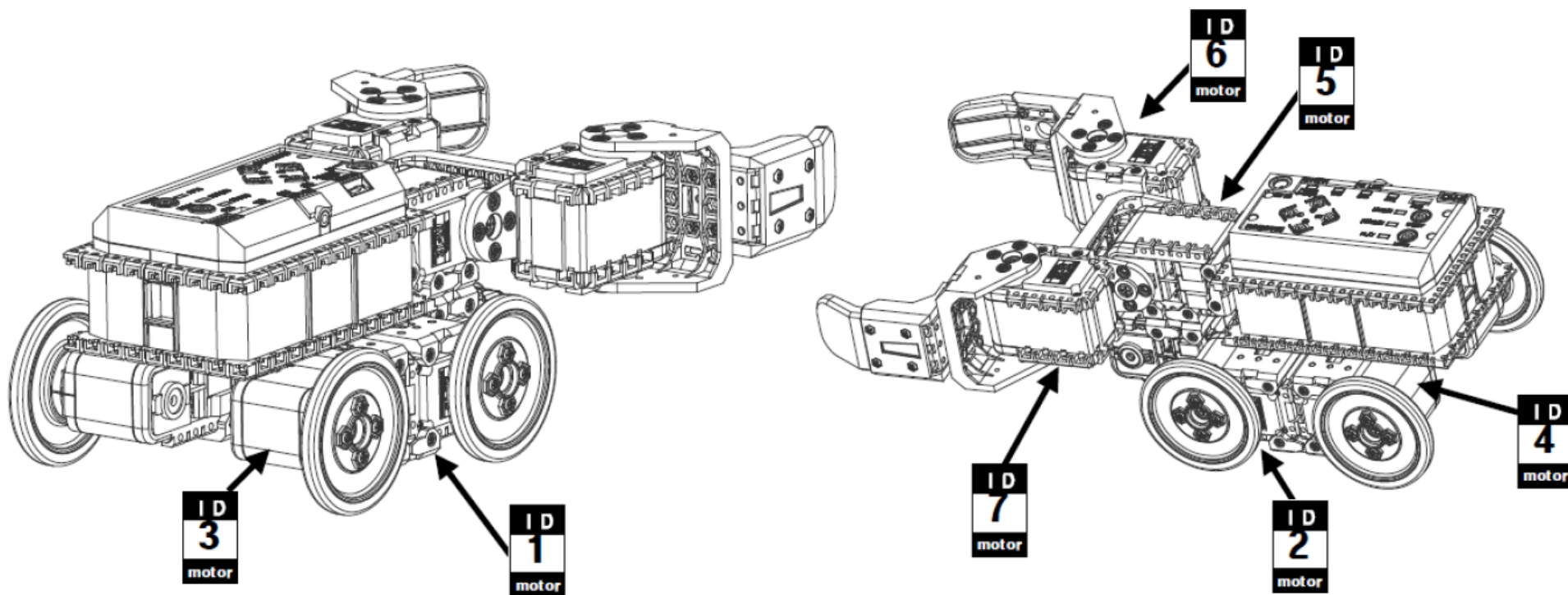
Krok 12 – Propojení pohonů II. část

Pohled z levé strany



Tento konektor může být propojen s řídicí jednotkou i shora.
Tj. nemusí být využitý konektor, který je naznačen na obrázku.

Krok 13 – Závěrečná kontrola rozmístění ID jednotlivých pohonů



3.3. Manipulátor se třemi stupni volnosti



PRAKTICKÁ ÚLOHA

Pomocí této kapitoly lze sestavit manipulátor se třemi stupni volnosti. Manipulátor lze ručně ovládat skrze vytvořené ovládací rozhraní, jež je tvořeno pohony AX-12. Použité pohony AX-12 pro ovládací rozhraní jsou využívány pouze jako měřicí prvek.

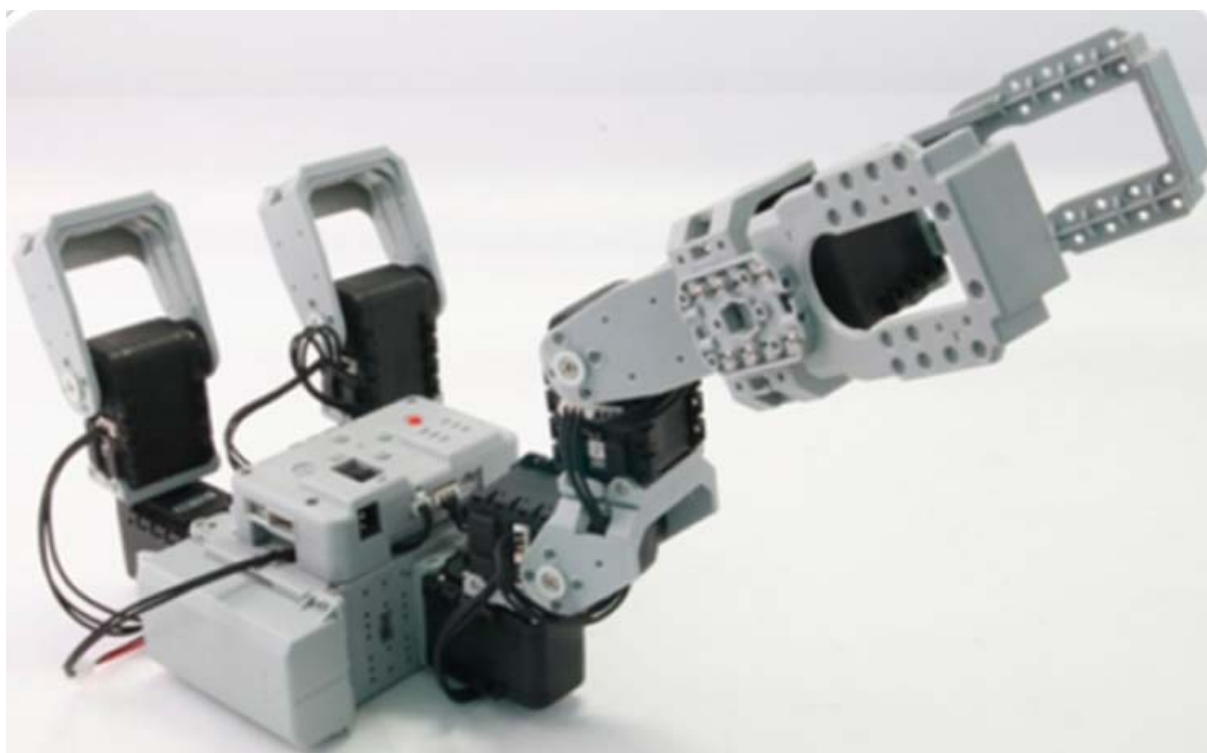


ČAS K SESTAVENÍ: 120 minut



CÍL

Cílem této úlohy je ukázat použití pohonů AX-12, jako měřicího prvku, což může být například využito při tvorbě ovládacího rozhraní robotického manipulátoru.

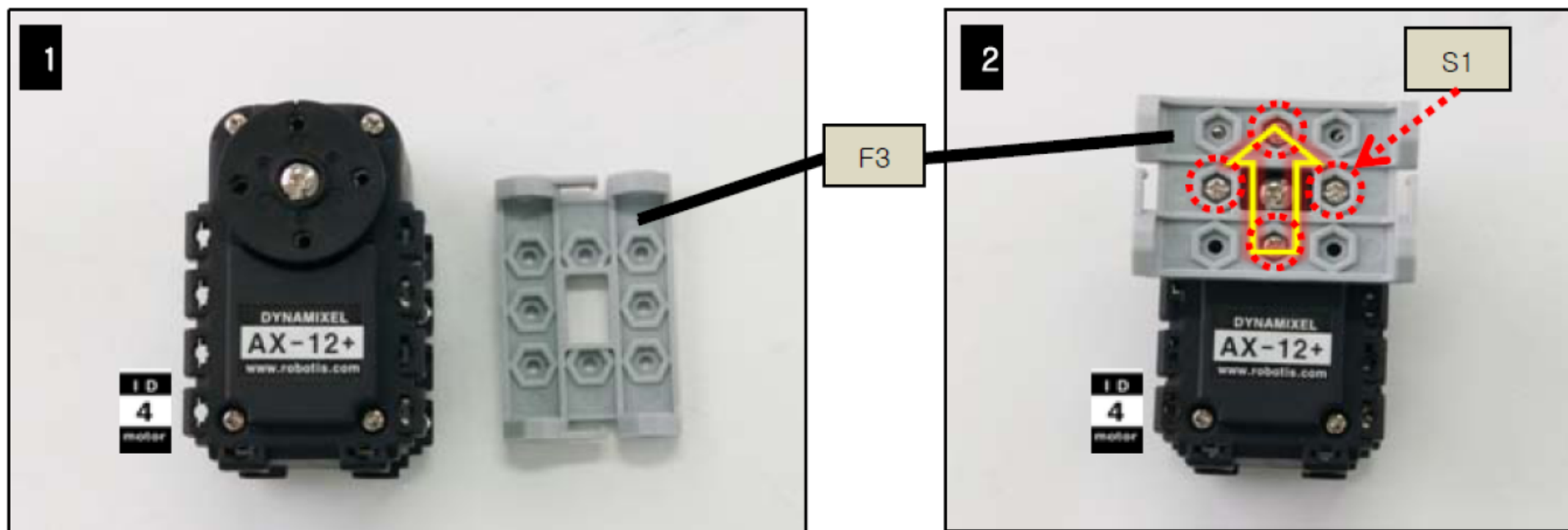


Postup sestavení robota:

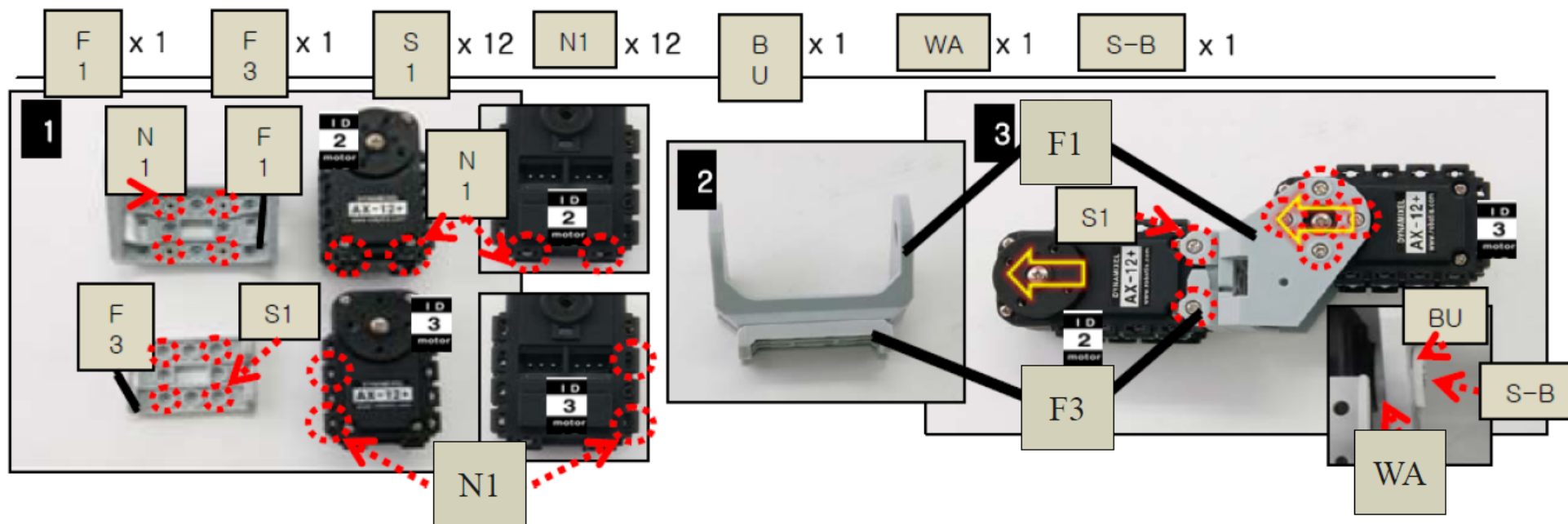
Krok 1

F3 x 1

S1 x 4

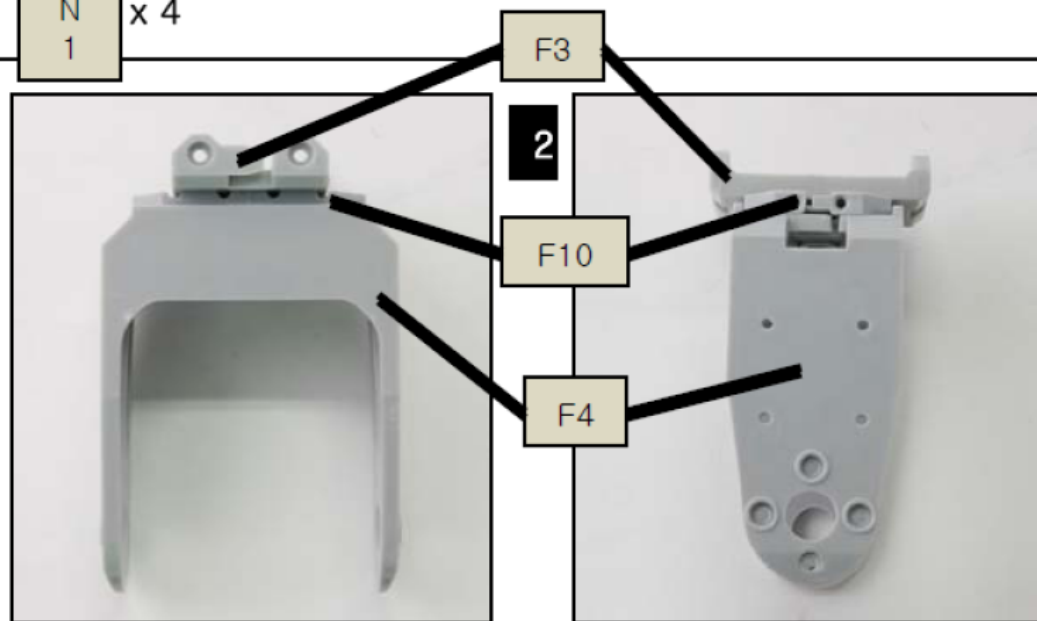
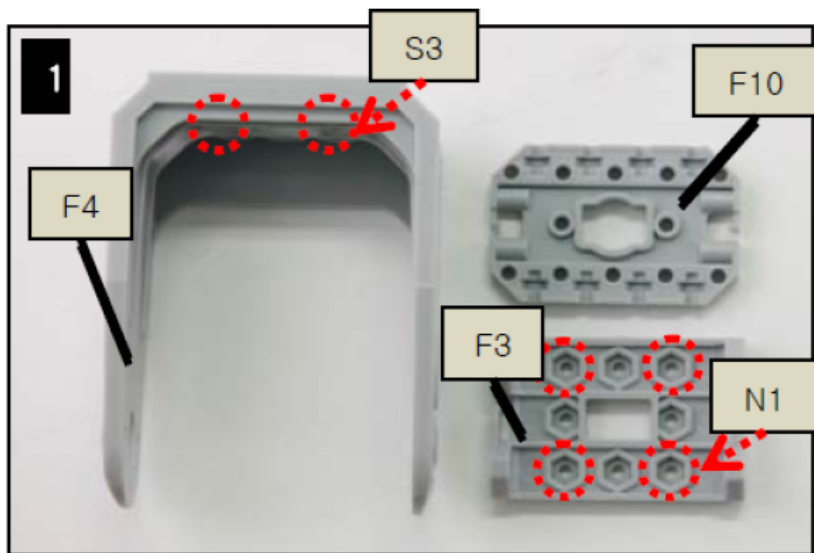


Krok 2

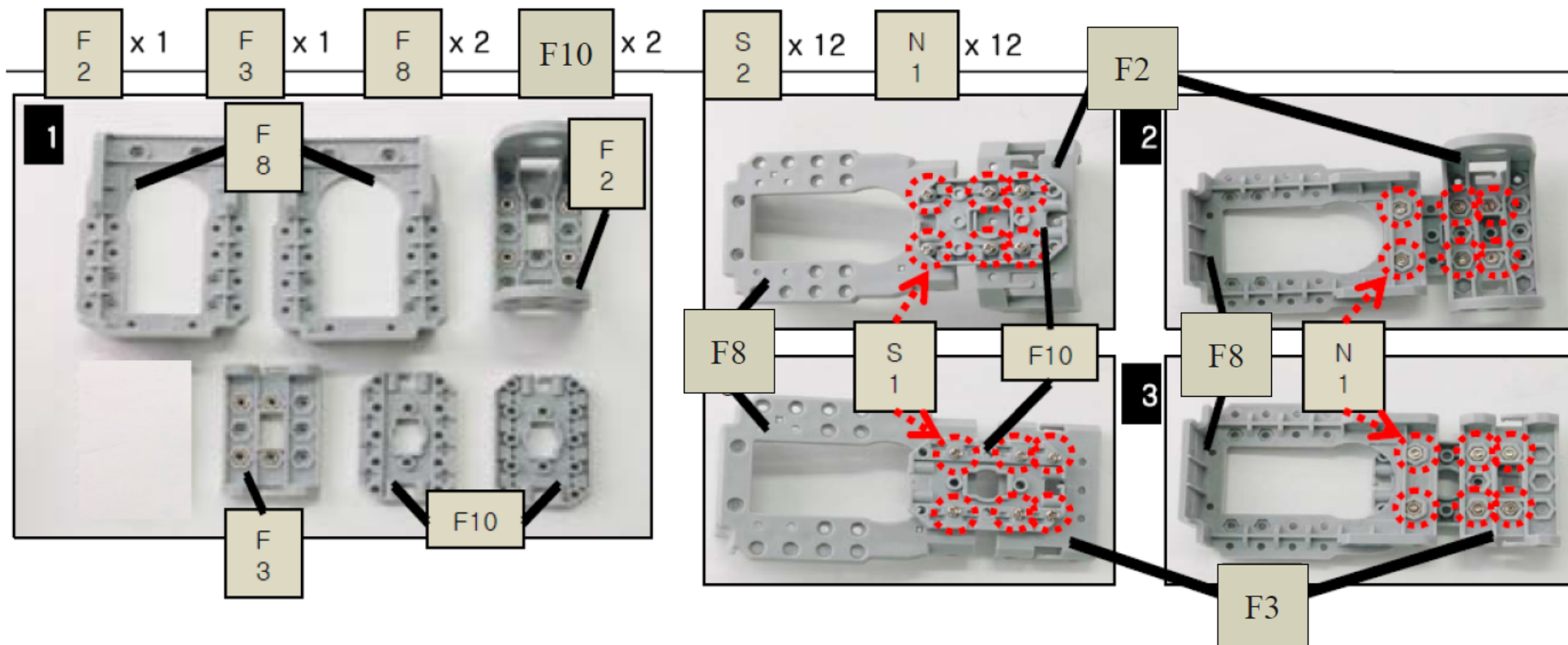


Krok 3

- F3 x 1
- F4 x 1
- F10 x 1
- S3 x 4
- N 1 x 4

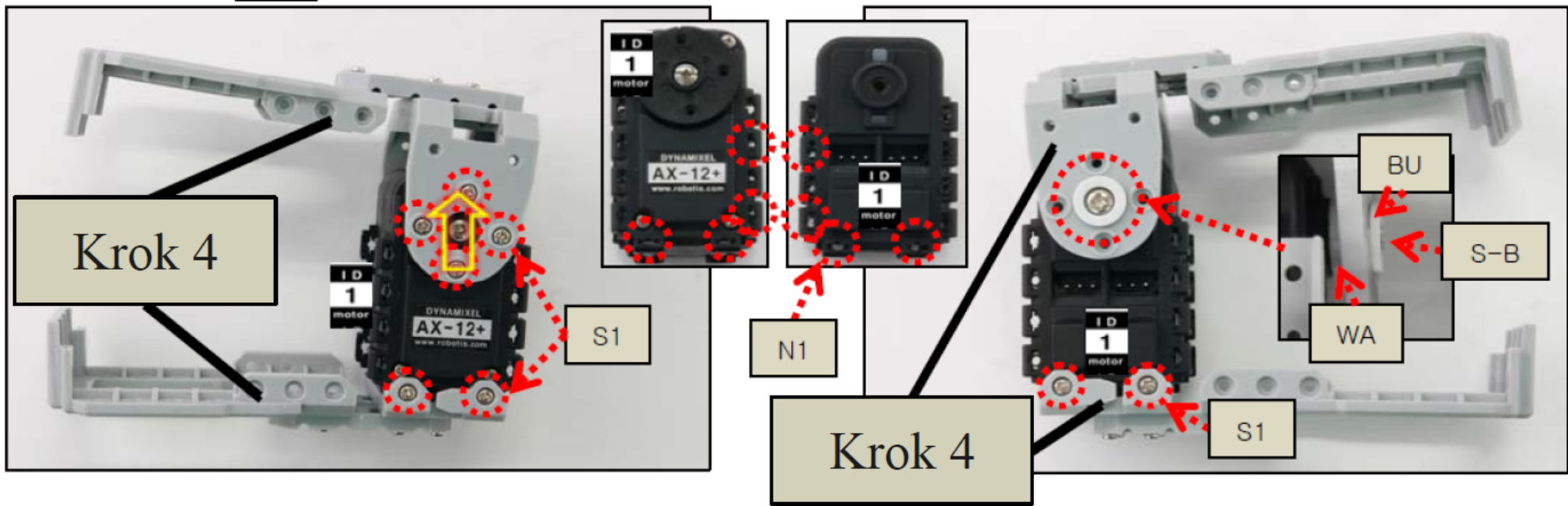


Krok 4



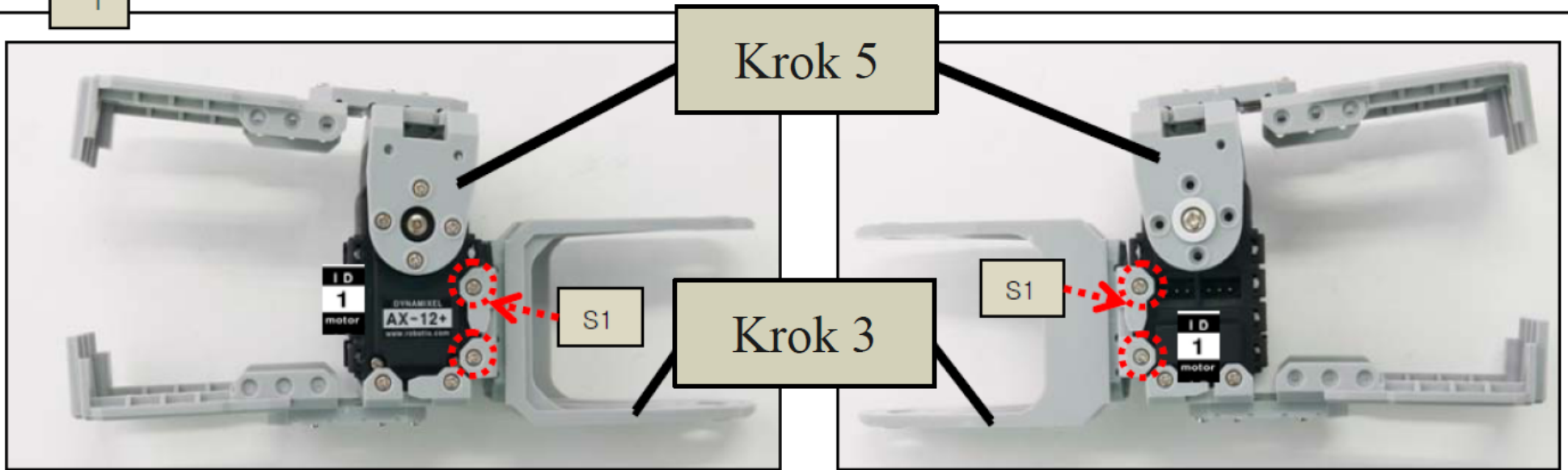
Krok 5

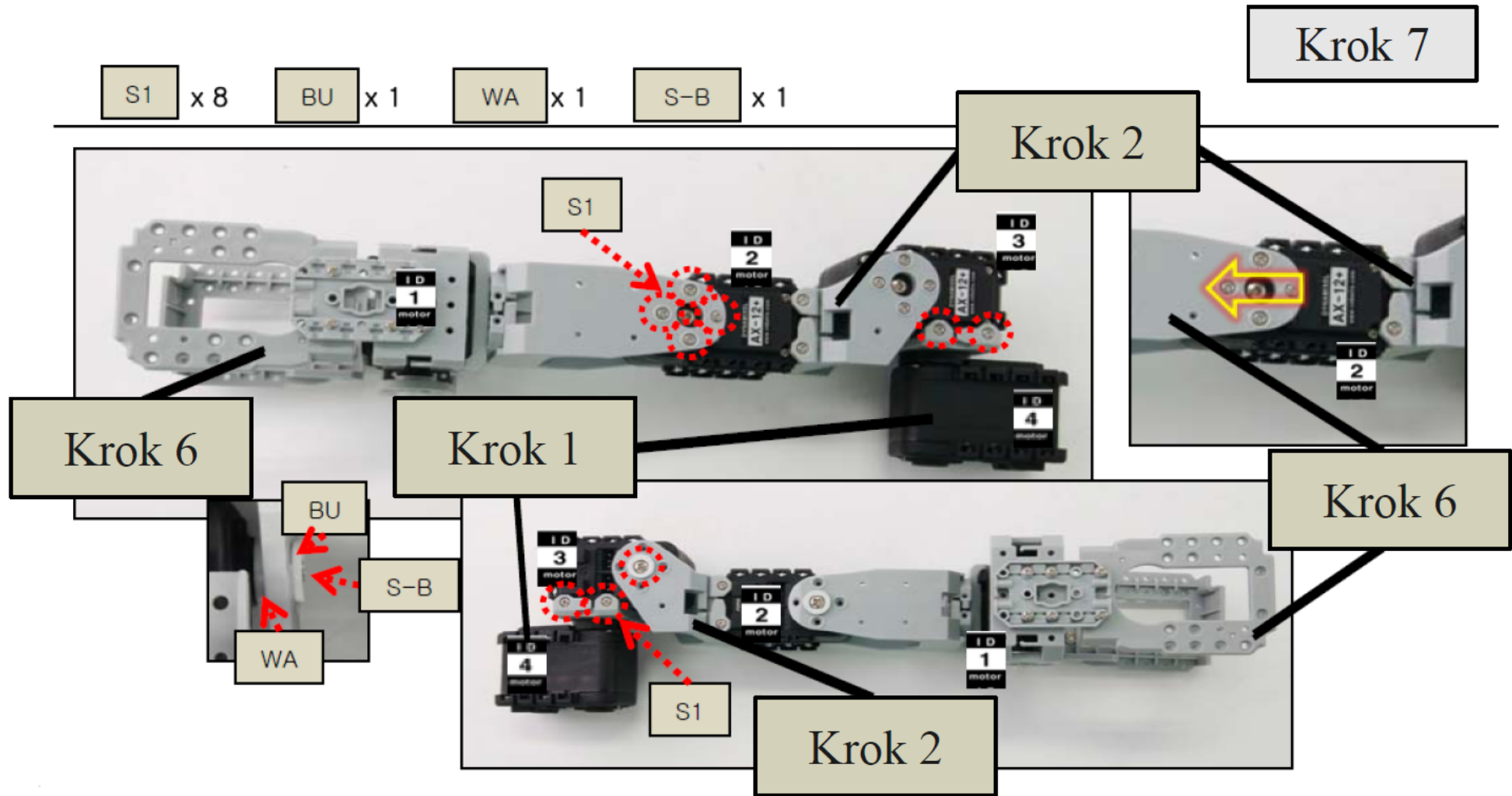
- S1 x 8 N 1 x 8 BU x 1 WA x 1 S-B x 1



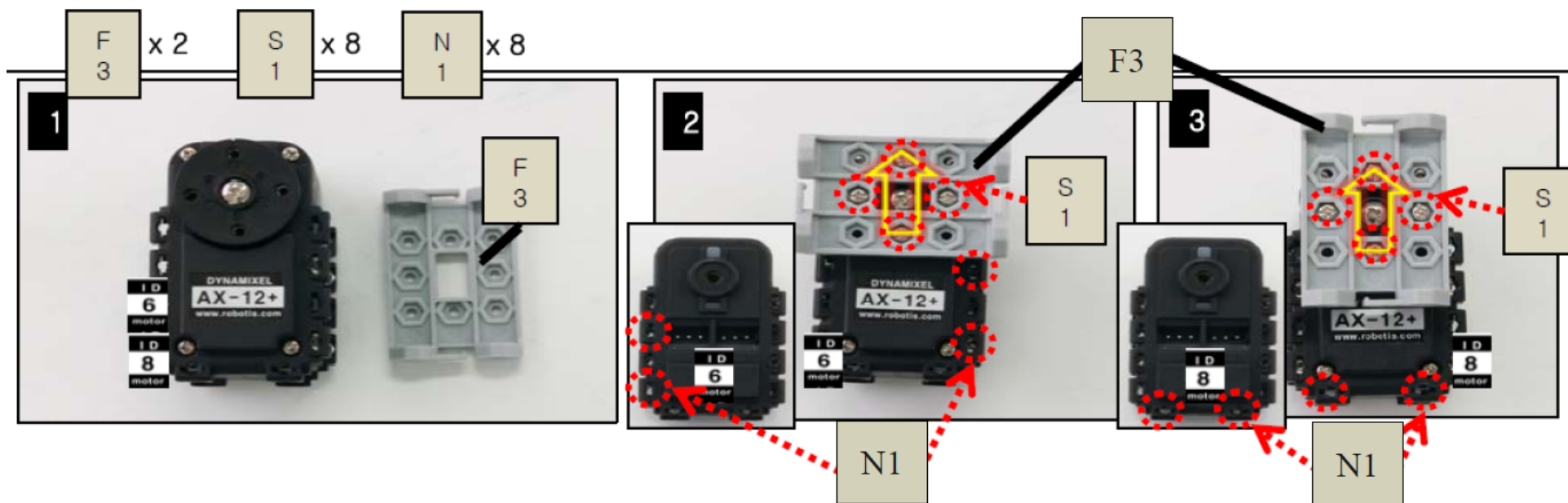
Krok 6

S1 x 4





Krok 8



Krok 9

F4 x 2

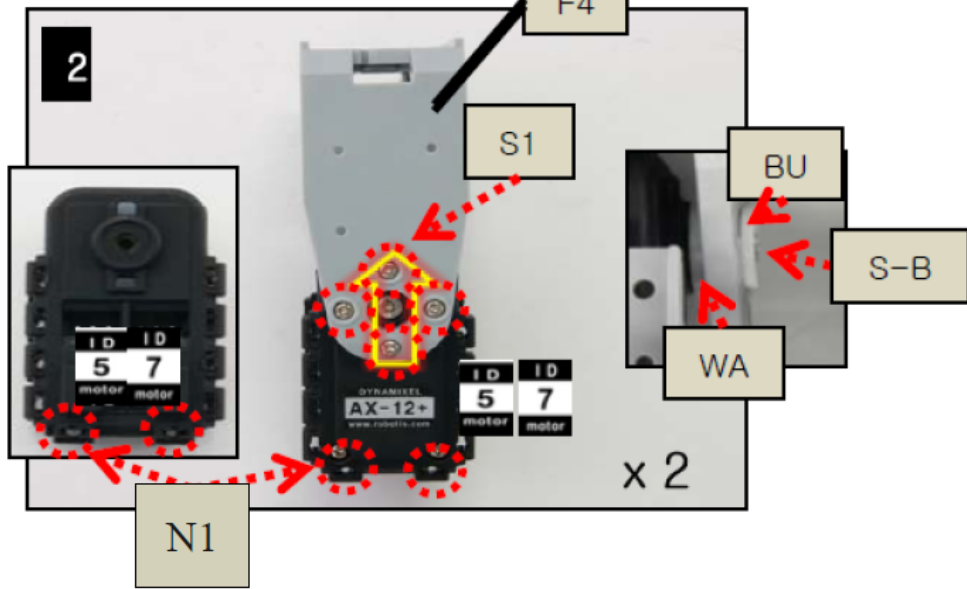
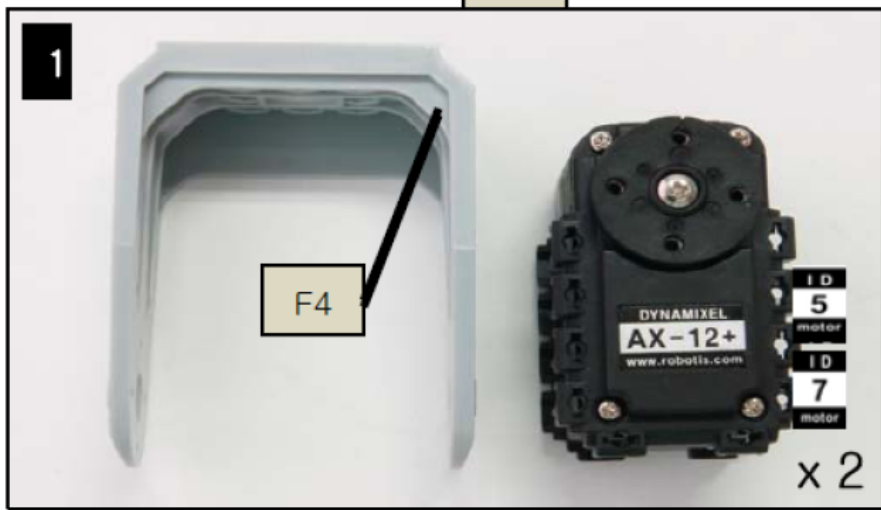
S1 x 8

N1 x 8

BU x 2

WA x 2

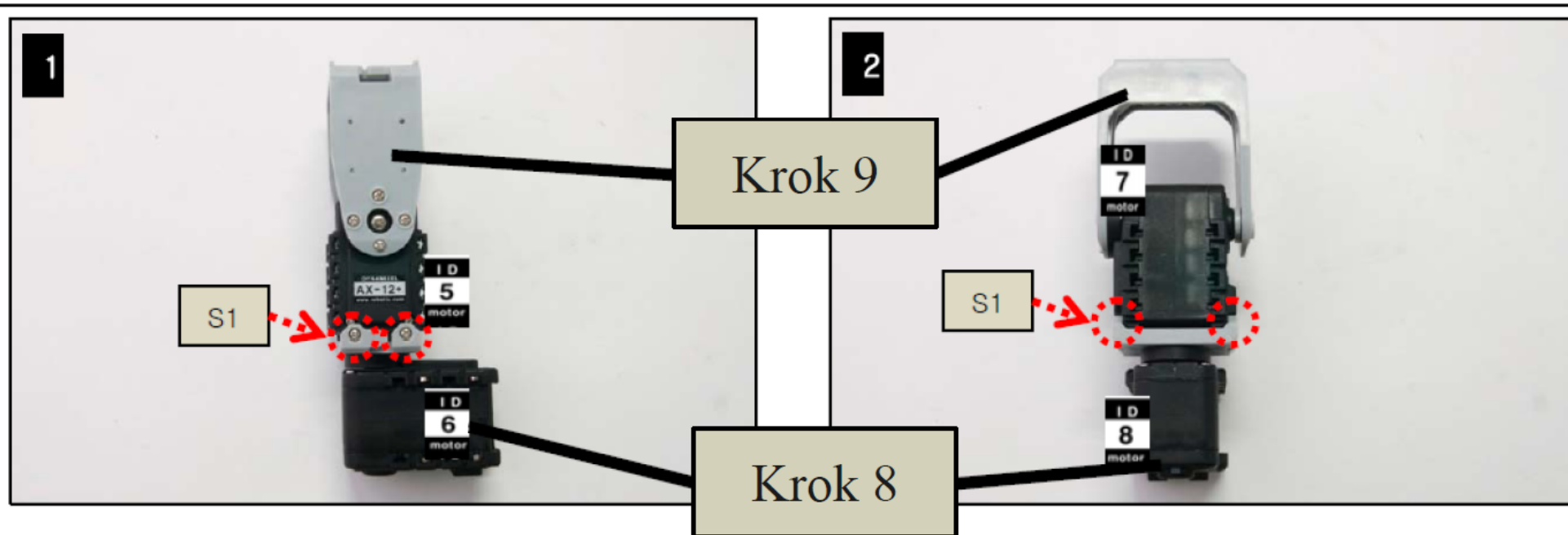
S-B x 2



Krok 10

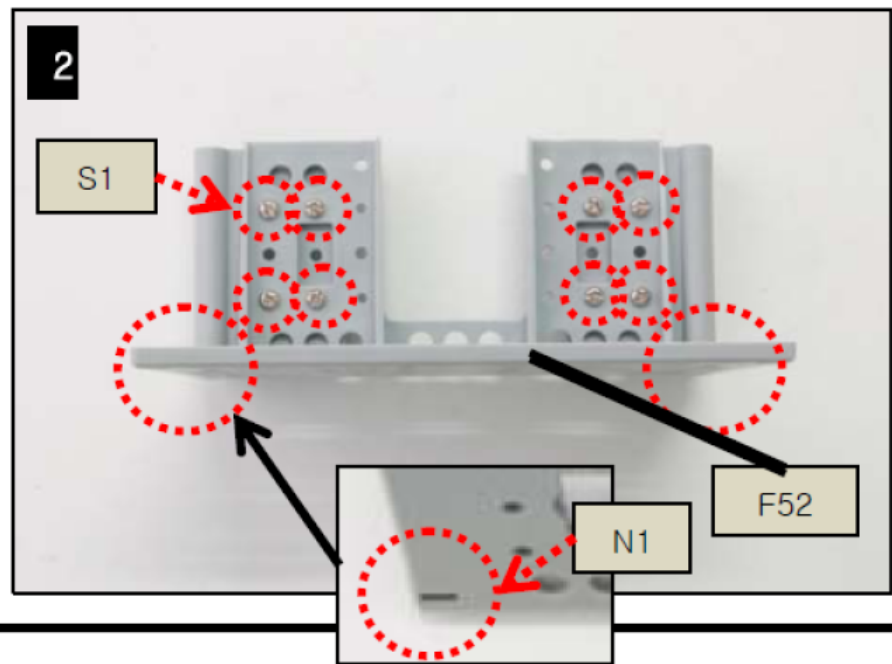
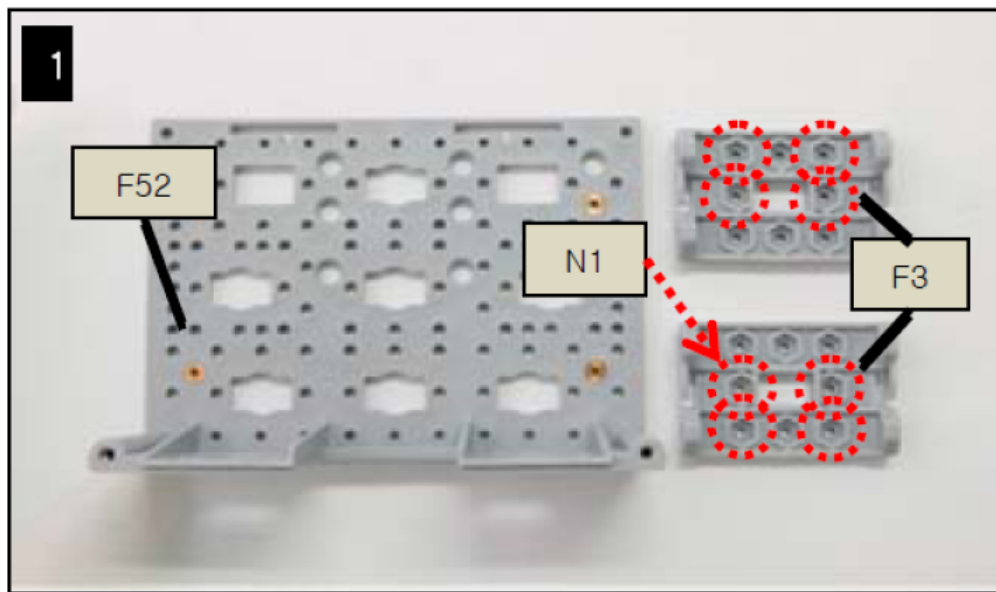
S1

x 8



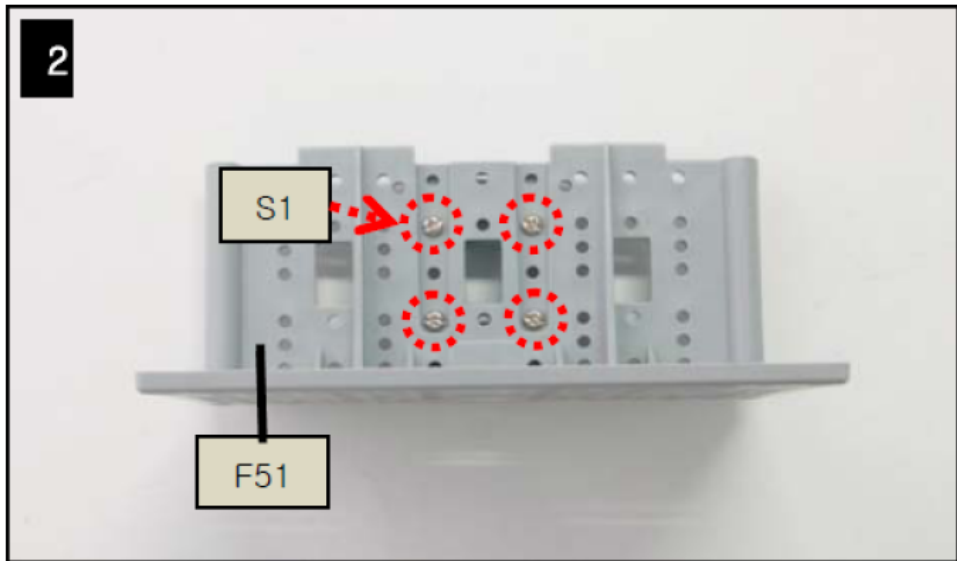
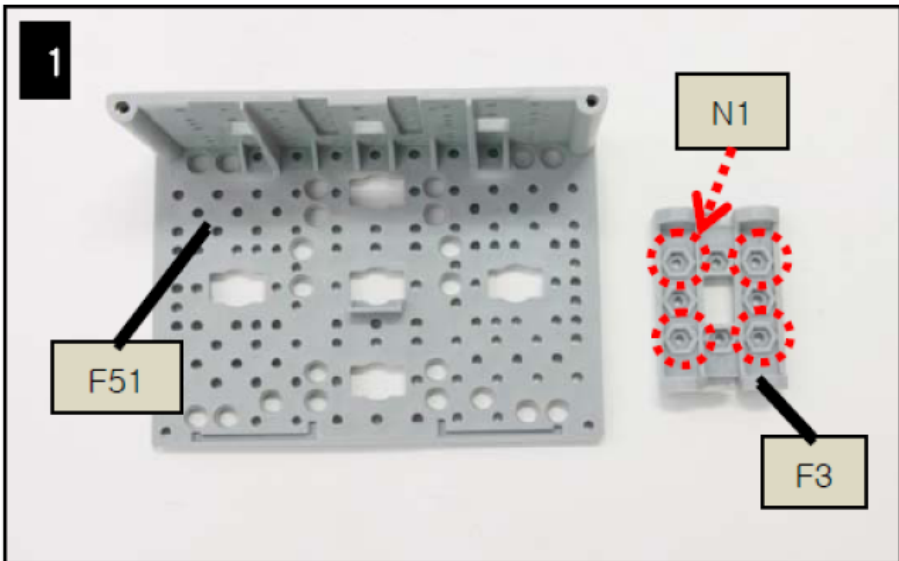
Krok 11

F3 x 2 F52 x 1 S1 x 8 N1 x 8



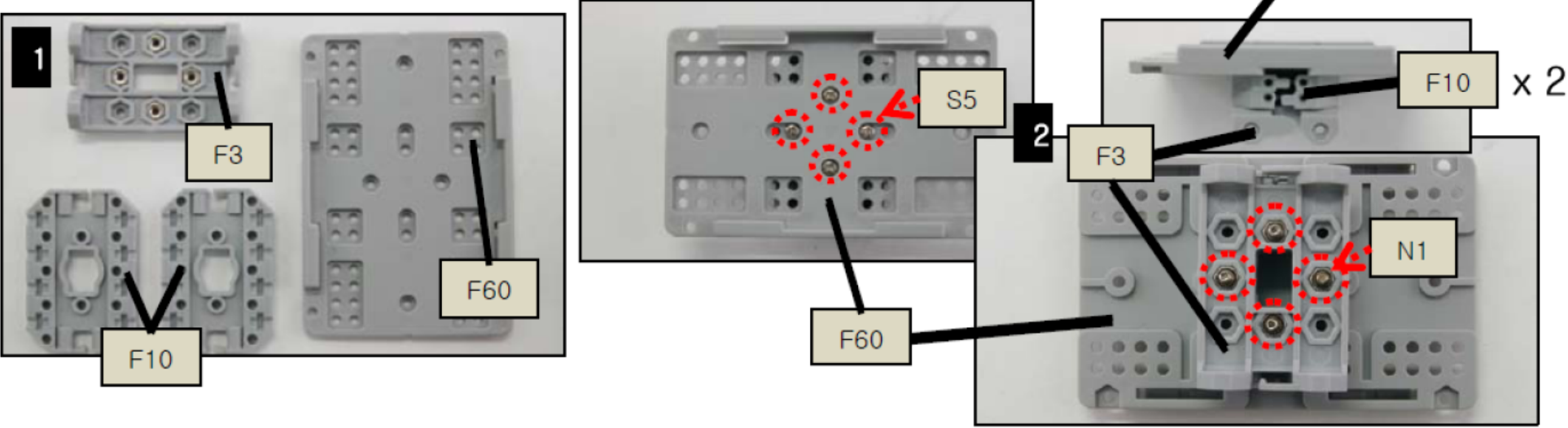
Krok 12

F3 x 1 F51 x 1 S1 x 4 N1 x 4



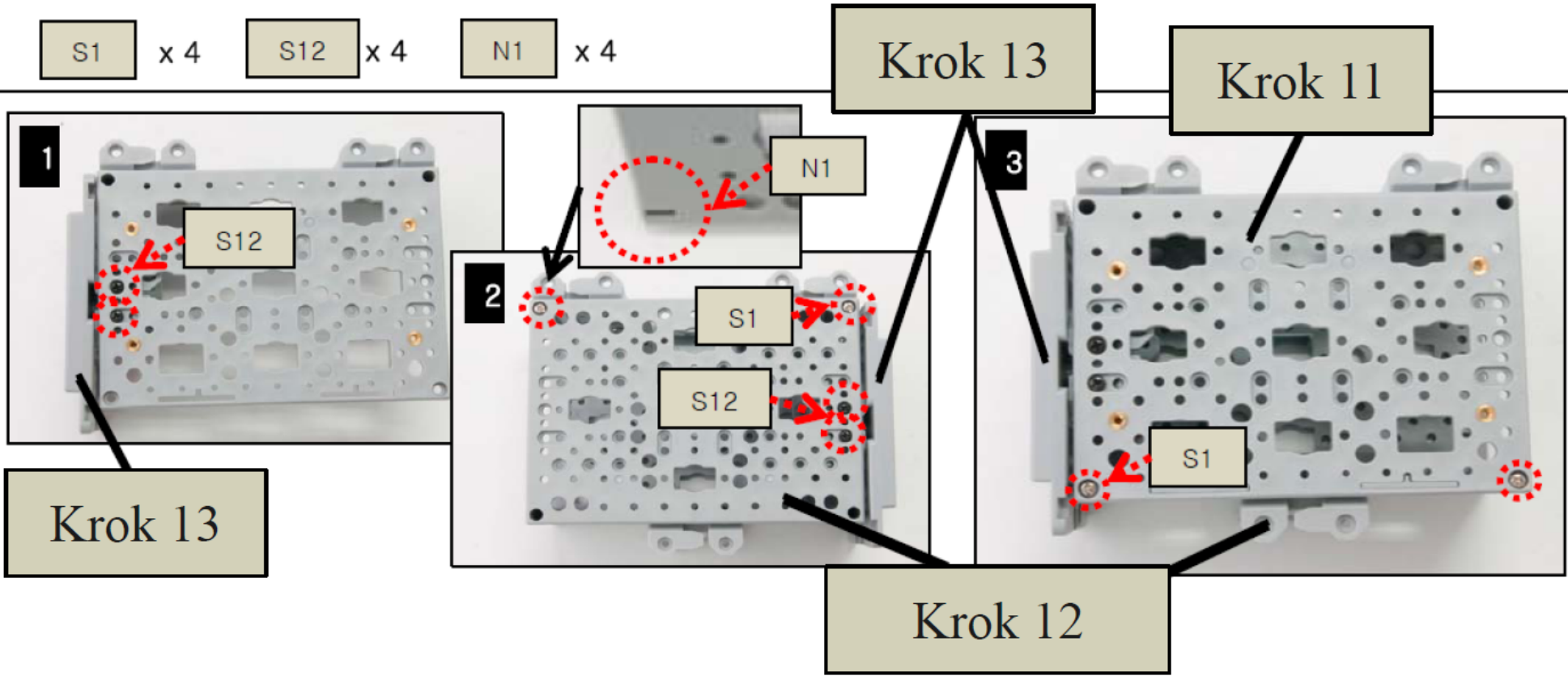
Krok 13

F3 x 1 F10 x 2 F60 x 1 S5 x 4 N1 x 4



Krok 14

S1 x 4 S12 x 4 N1 x 4



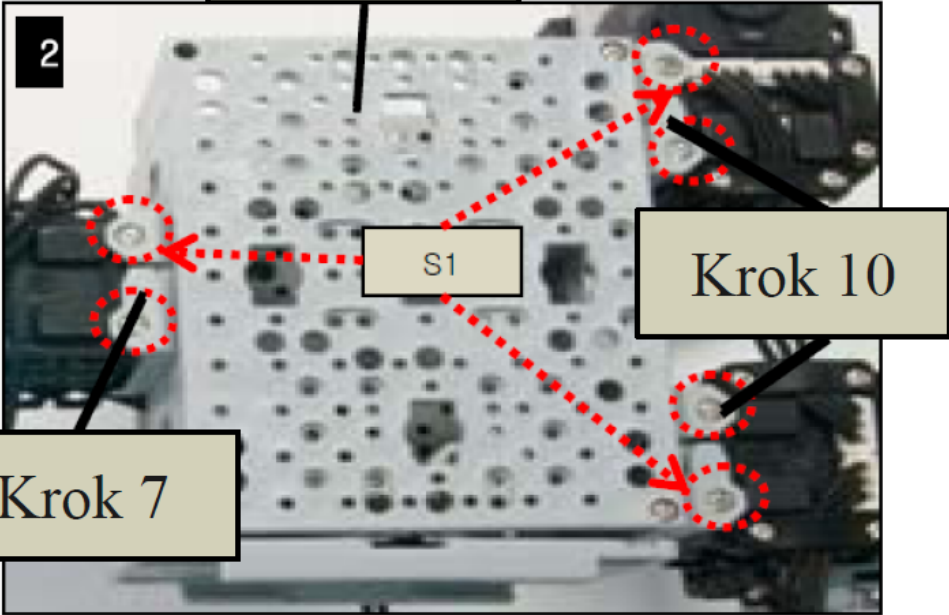
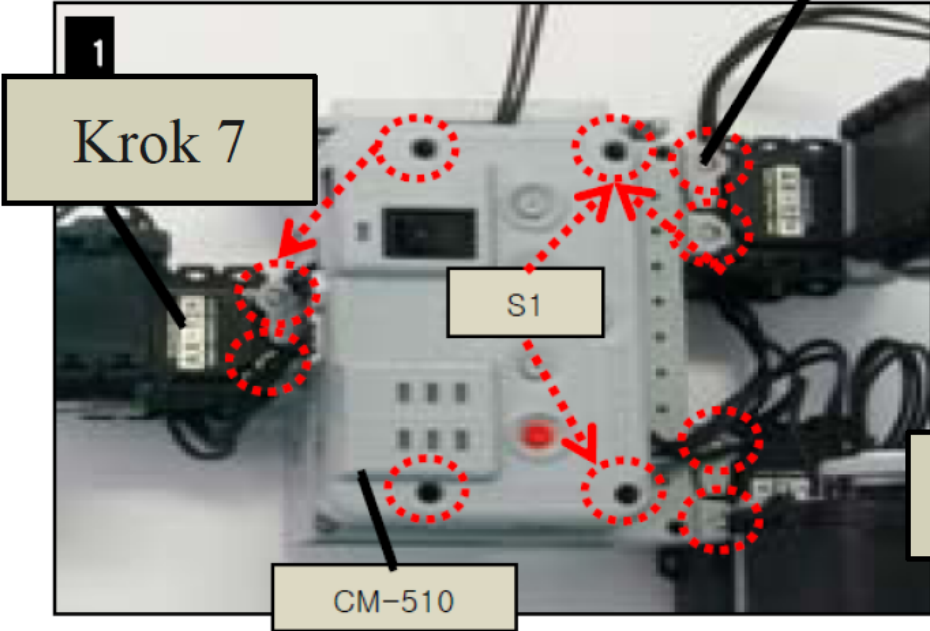
Krok 15

S1 x 16

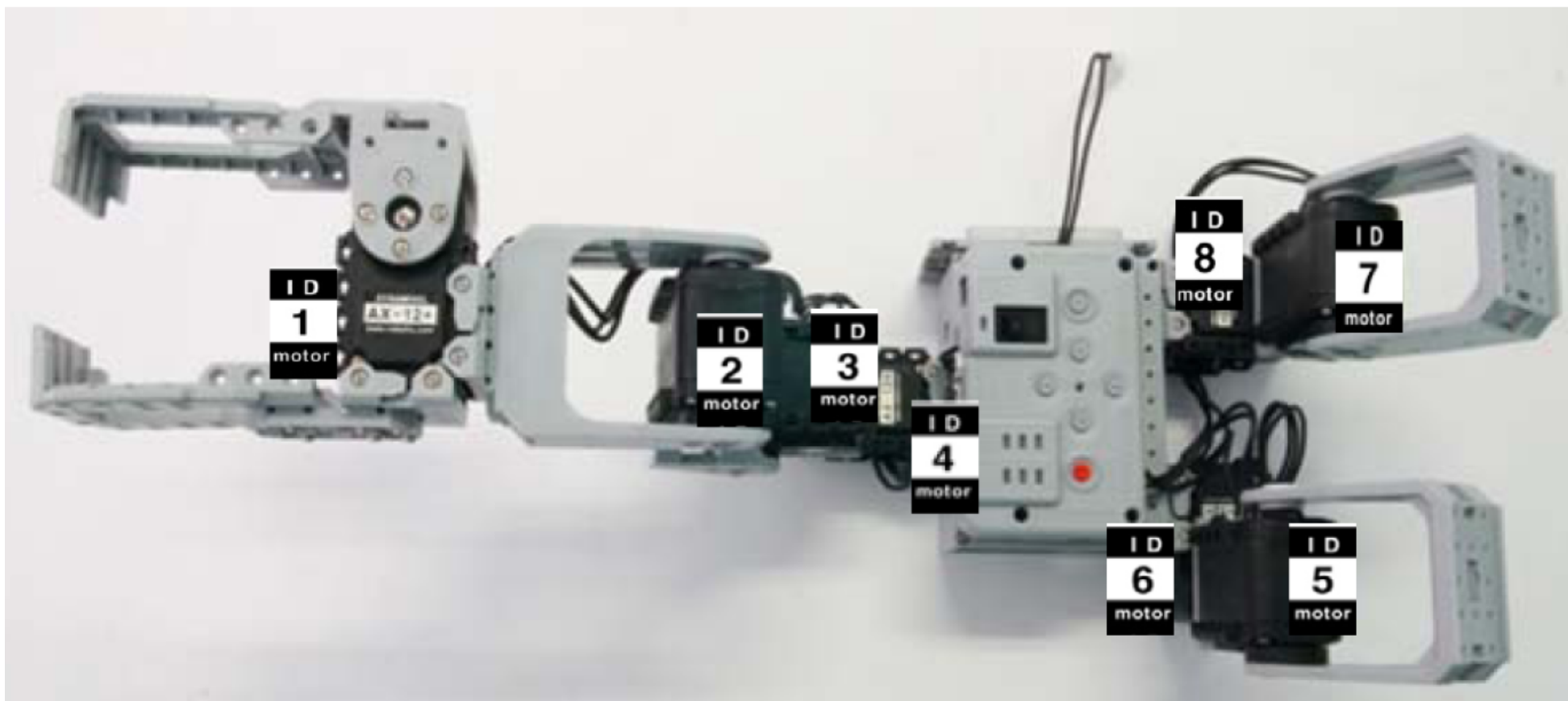
CM-510 x 1

Krok 10

Krok 14

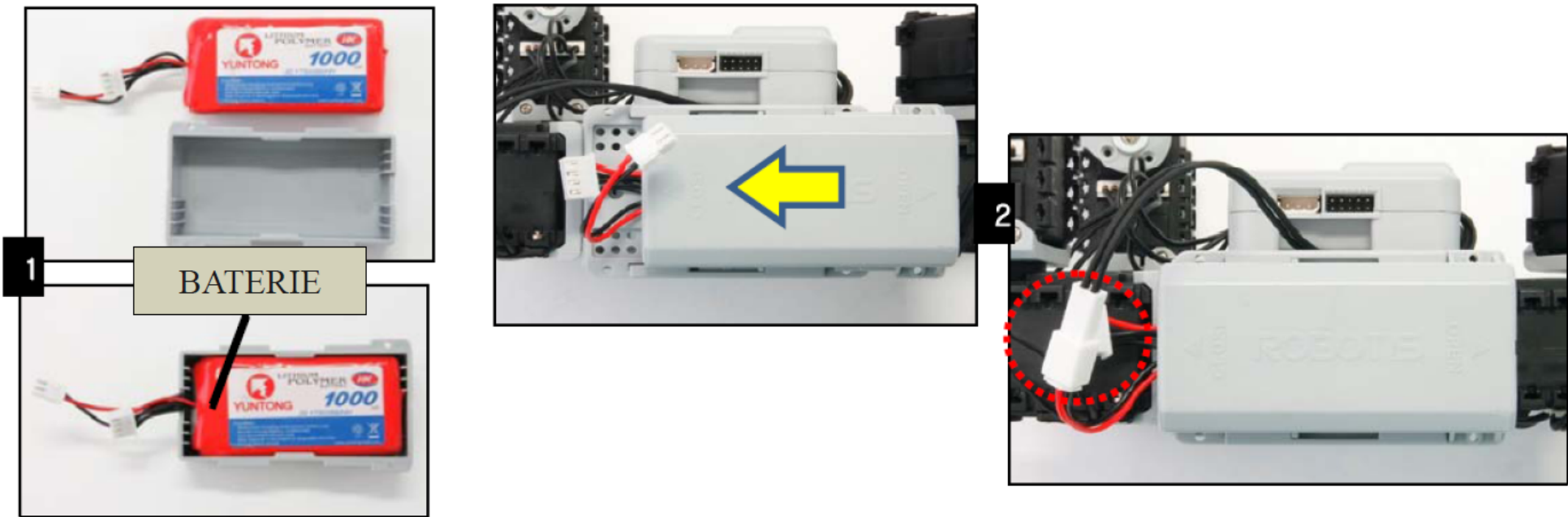


Krok 16



Krok 17

BATERIE x 1



Použitá literatura

- [1] ROBOTIS: *Dynamixel AX-12*. User's manual, Korea 2006
- [2] ROBOTIS: *Dynamixel Sensor Module AX-S1*. User's manual, Korea 2006
- [3] ROBOTIS: *USB2Dynamixel*. User's manual, Korea 2006, Version 1.2

ROBOTI

VE ŠKOLE PRO PRAKTICKOU VÝUKU, MOTIVACI I ZÁBAVU

CZ.1.07/1.1.24/01.0066



INVESTICE DO ROZVOJE VZDĚLÁVÁNÍ

MANIPULÁTORY A VOZÍKY

Ing. Michal ŘEPKA, Ph.D.