

# CONTROL OF ROBOTIC ARM MITSUBISHI MELFA 6SL

**Petr Schindler**

Bachelor Degree Programme (1), FIT BUT  
E-mail: xschin00@stud.fit.vutbr.cz

Supervised by: František Zbořil  
E-mail: zboril@fit.vutbr.cz

## ABSTRACT

The paper describes the current state of online laboratory project, which is focused on support of the teaching of artificial intelligence, primarily on solving problems from the artificial world of cubes, using robotic arm Mitsubishi Melfa 6 SL. This paper describes achievements in control of the robot.

## 1. ÚVOD

Má práce se zabývá robotickým ramenem Melfa 6SL. Toto rameno je určeno pro průmyslové užití k automatické práci. Cílem projektu je seznámit se s ovládáním ramene a především pak vypracovat knihovnu programů, která bude umožňovat jeho řízení. V konečné fázi bude tato práce pouze částí velkého projektu, který bude sloužit jako podpora pro výuku umělé inteligence.

## 2. ROBOTICKÉ RAMENO

Jedná se o robotické rameno firmy Mitsubishi se šesti klouby. K ramenu je připojena ruka Schunk PG 70. Robotické rameno Melfa 6 SL je zaměřeno na automatickou, programem řízenou práci v průmyslu. Její určení se odvíjí od připojeného nástroje. V našem případě je připojena mechanická ruka, s jejíž pomocí je možné chytat různé předměty. Díky vyměnitelným nástavcům, lze ruku přizpůsobit předmětům, které má držet.

Robotické rameno ovládá řídicí jednotka. Řídicí jednotka obsahuje několik rozhraní pro komunikaci s uživatelem.

- Teaching device je ovladač obsahující tlačítka pro ovládání pohybů ramene, z nichž pro nás nejdůležitější je červené tlačítko "stop", při jehož stisknutí se rameno okamžitě zastaví. Cílem práce je však ovládání ramene prostřednictvím počítače, tudíž teaching device není potřeba.
- COM port či Ethernet. Pomocí těchto dvou rozhraní je možné spojit řídicí jednotku s počítačem.

### 2.1. URČOVÁNÍ POZICE

Při práci s ramenem je potřeba určovat aktuální a cílové pozice ruky a často i její úplnou dráhu. Zjištěné pozice se pak využívají v programech jako parametry funkcí (např. MOV).

Pozice ruky se může určovat třemi způsoby. Prvním jsou souřadnice konce ramene (resp. ruky) s počátkem soustavy na pracovní podložce (ploše) uprostřed robota. Měřítka os je v milimetrech. Pro snadnější orientaci v tomto prostoru je na pracovní ploše několik značek udávající hodnoty na osách X a Y. Osa Z udává výšku konce ramene. Osy X a Y jsou pak naznačeny na zemi. Postavíme-li se ve směru osy X, osa Y roste doleva. U tohoto typu měření pozice je potřeba ještě určit naklonění samotného nástroje (ruky). Naklonění udávají další 3 parametry (označované jako a, b, c), které značí rotaci okolo os (X, Y a Z).

Druhým způsobem je určování pozice podle natočení jednotlivých kloubů. Natočení kloubů se určuje ve stupních a počítá se od počáteční pozice (nastavuje se pomocí parametrů).

Třetí souřadnicový systém je polární. Tři základní parametry určují vzdálenost od středu, úhel otočení (odpovídá pohybu prvního kloubu) a výšku. Vzdálenosti jsou v milimetrech, úhly ve stupních. Natočení nástroje je stejné jako u pravoúhlého souřadnicového systému.

### 3. KNIHOVNA PRO OVLÁDÁNÍ

Knihovna pro ovládání robotického ramene musí mít prostředky pro řízení pohybů ramene, odhalení případných chyb, připojení a komunikaci s robotem, zjišťování pozice ruky a pro rychlé zastavení v případě nebezpečí (tzv. softwarový emergency stop).

Protože má práce bude součástí většího projektu, bude potřeba vytvořit knihovnu, která by splňovala výše uvedené podmínky a byla by lehce importovatelná do většího projektu, kde by měla na starosti právě ovládání robotického ramene.

Jako dobré řešení se mi jeví objektový návrh. Knihovna by pak obsahovala objekt, který by byl abstrakcí ramene. Jako atributy budou sloužit informace o poloze robota. Ty umožní snadnější verifikaci dalšího kroku (ať už bezpečnost akce, či pouze zjištění, jestli danou akci lze provést). Díky informaci o aktuální poloze můžeme provádět i relativní pohyby.

Jednotlivé metody odpovídají příkazům, případně sledu příkazů, podle činnosti, kterou chceme provádět. Při vytvoření instance třídy je třeba, aby se program k ramenu připojil a připravil robota na práci. Toho dosáhneme zavoláním metody connectRobot, která se připojí na požadovaný (případně implicitně zadaný) COM port, kam následně pošle inicializační příkazy. Mezi tyto příkazy patří vymezení pracovní paměti robota, určení rychlosti, kterou se bude robot pohybovat a zapnutí servomotorů. V inicializaci bude též potřeba zjistit aktuální pozici ramene a tím inicializovat třídní proměnné.

Mezi nejdůležitější funkce, které jsou potřeba, patří okamžité zastavení. K tomuto účelu dobře poslouží vypnutí serv. Poněvadž je komunikace asynchronní, bude zastavení rychlé (určitě ovšem pomalejší než fyzický emergency button). Dále by měla existovat metoda sloužící ke zjištění, zda-li je robot v pořádku a zda nenastala žádná chyba. Řídící jednotka, ve chvíli kdy přijme příkaz ERROR odpoví buď err s číslem chyby, či qok (tzn. všechno je v pořádku). Stačí pouze zachytit odpověď a pokud není vše v pořádku, s chybou dále pracovat (bude potřeba vypnout alarm a informovat uživatele).

Nejdůležitější je pohyb - ten bude buď relativní (o předepsanou vzdálenost a směr z aktuální pozice) nebo absolutní (na přesně zadanou pozici). U pohybů bude dále možnost vybírat mezi příkazy mov (Joint interpolation movement) a mvs (Linear interpolation movement), v aplikaci se pravděpodobně bude využívat pouze jeden, ale pro obecnost a možnost výběru budou v knihovně obě možnosti.

Nejtěžší na komunikaci s ramenem bude asynchronnost komunikace. Řešením je určit čas dokončení operace (rameno neoznamuje ukončení jednotlivých operací).

### 3.1. JAZYK MELFA-BASIC IV

Tento jazyk se používá pro základní ovládání ramene. Je to interpretovaný jazyk, jehož příkazy zpracovává řídicí jednotka. Ta pak posílá příkazy jednotlivým servům ramene. Můžeme buď zasílat jednotlivé příkazy nebo můžeme vytvořit celý program, který poté uložíme do paměti řídicí jednotky, odkud ho můžeme pak kdykoliv spustit.

Jazyk obsahuje celou řadu příkazů. Jsou zde příkazy pro různé pohyby, nastavení různých parametrů jako např. rychlosti, dále příkazy pro ukládání pozic (momentální prostorové nastavení robota), matic pozic, příkazy pro spuštění programu apod.

### 3.2. KOMUNIKACE S ŘÍDÍCÍ JEDNOTKOU

Cílem mého projektu je ovládat robotické rameno vzdáleným přístupem prostřednictvím počítače a počítačové sítě. Počítač je k řídicí jednotce připojen přes COM port. Nejprve bylo potřeba zjistit, jak počítač komunikuje s řídicí jednotkou. Využil jsem toho, že k robotickému rameni je dodáván program Cosirop, který slouží k jeho ovládání. Tento program má intuitivní ovládání, obsahuje editor programů v jazyku Melfa-Basic, a také funkce pro řízení pohybů ramene.

Pro zjištění komunikačního protokolu mezi PC a řídicí jednotkou jsem odposlouchával komunikaci přes COM port. Po analýze průchozích zpráv jsem zjistil, že protokol je textový a že příkazy zasílané ramenu jsou téměř totožné s jazykem Melfa-Basic.

Komunikace s robotem začíná ustavením spojení přes COM port. Dále proběhne inicializace pracovního prostoru (v paměti kontroléru). Je vypnut alarm a nastavena rychlost. Nakonec se sepnou serva a čeká se na příkazy.

Formát příkazů zasílaných přes COM port vypadá takto:

```
1;1;EXEC<příkaz v jazyce Melfa-Basic>
```

Příklad příkazu mov s parametrem P1:

```
1;1;EXECMOV P1
```

## 4. ZÁVĚR

V nynější době je již knihovna pro ovládání ramene téměř hotová. Díky objektovému návrhu nebude těžké přidávat další vlastnosti a funkce. Knihovnu lze využít k přímé interpretaci jazyka Melfa-Basic. Obsahuje ovšem i metody, které generují celé části kódu (např. sekvence příkazů pro ustavení spojení).

Další fáze projektu bude zaměřena především na bezpečnost práce ramene a v součinnosti s kamerovým systémem pak na rozpoznávání kostek a na jejich přemísťování.

## LITERATURA

[1] Mitsubishi Industrial Robot: Instruction manual, Melfa BFP-A5993-K, 2005