Hoe kan de robotarm tafeltennissen?

Opleverdocument – AP Project – Iets Technisch 1

15-12-2015

HAN

Remco van Alen, Bas van Summeren, Michiel Buevink, Paul Verhoeven, Thomas Fransen

Versie 1.0

# Inhoudsopgave

[Inhoudsopgave 1](#_Toc440622838)

[1 Inleiding 2](#_Toc440622839)

[2 Installatie COSIMIR 3](#_Toc440622840)

[2.1 COSIMIR 3](#_Toc440622841)

[2.1.1 Crypto-key 3](#_Toc440622842)

[2.1.2 Installatie 3](#_Toc440622843)

[2.1.3 Project aan maken 3](#_Toc440622844)

[2.1.4 De robot verbinden 4](#_Toc440622845)

[3 Configuratie COM-poort 5](#_Toc440622846)

[4 Eigen commando’s 6](#_Toc440622847)

[4.1 Code bestanden toevoegen 6](#_Toc440622848)

[4.1.1 Uploaden 6](#_Toc440622849)

[4.1.2 Configuratie 6](#_Toc440622850)

[4.2 Werking Melfa Basic code 7](#_Toc440622851)

[4.2.1 Sloten 7](#_Toc440622852)

[4.2.2 1.MB4 7](#_Toc440622853)

[4.2.3 2.MB4 7](#_Toc440622854)

[4.3 Fout afhandeling 7](#_Toc440622855)

[4.4 Commando’s voor de code 8](#_Toc440622856)

[5 Standaard commando’s 9](#_Toc440622857)

[5.1 Commando’s 9](#_Toc440622858)

[5.2 Verzenden 9](#_Toc440622859)

[5.3 Configuratie 9](#_Toc440622860)

[5.4 Voorbeeld 10](#_Toc440622861)

[6 Aansturing in C++ 11](#_Toc440622862)

[7 Simulatie 12](#_Toc440622863)

[7.1 Communicatie naar simulatie 12](#_Toc440622864)

[7.2 Simulatie naar robot 12](#_Toc440622865)

[7.3 Achtergrond 12](#_Toc440622866)

[8 Onderhoud en fouten 13](#_Toc440622867)

[8.1 Onderhoudsprogramma’s 13](#_Toc440622868)

[8.2 De standaard commando’s werken niet 13](#_Toc440622869)

[8.2.1 De code voor de eigen commando’s staan op de robot en zijn geconfigureerd 13](#_Toc440622870)

[8.2.2 Er komt geen reactie 13](#_Toc440622871)

[8.2.3 Er komt een Qer terug 13](#_Toc440622872)

[8.3 Weergave FAIL op het display 13](#_Toc440622873)

[Bijlagen 14](#_Toc440622874)

[1 Daily inspection items 14](#_Toc440622875)

[2 Periodic inspection 15](#_Toc440622876)

# Inleiding

Dit document is opgesteld om gemakkelijk een begin te maken met het gebruik van de Robot: Melfa RV-2AJ. Gedurende het project in de Minor Advanced Programming is veelvuldig gebruik gemaakt van de Melfa RV-2AJ. Relevante informatie die tijdens het gebruik aan het licht is gekomen staat in dit document samengevat. Bij dit document wordt code geleverd voor het gebruik.

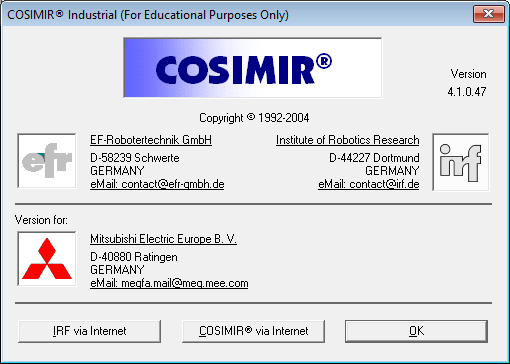
Deze handleiding gaat er vanuit dat de robot op de fabrieksinstellingen staat als er begonnen wordt met de software.

# Installatie COSIMIR

Voor het aansturen van de robot via een RS232 verbinding is een Melfa Basic programma nodig. Met dit programma kunnen hoeken van de gewrichten ingesteld worden. Hierbij kan er gekozen worden voor exacte hoeken of relatieve hoeken. De software is ook verantwoordelijk voor het melden van fouten en het resetten van de software.

## COSIMIR

Voor het uploaden en bewerken van de software is de COSIMIR ontwikkelomgeving te gebruiken. Deze software werkt enkel onder Windows en wordt al geruime tijd niet meer geüpdatet. In deze handleiding wordt gebruik gemaakt van de onderstaande versie.



Figuur Versie window COSIMIR

### Crypto-key

Voor het gebruik van COSIMIR is te allen tijde een Crypto-key nodig. Deze bijgeleverde key dient aangesloten te zijn op een USB-poort en voorzien te zijn van de correcte drivers. Als Windows de drivers niet correct installeert kunnen de driver van de volgende pagina worden gedownload: <https://www.marx.com/en/support/downloads>.

### Installatie

Om COSIMIR te installeren dient de Crypto-key aangesloten te zijn en Windows Defender of een eventuele andere antivirus software uitgeschakeld te zijn. Zonder deze maatregelen kan de installatie niet afgerond worden. Vervolgens kan de installatie gestart worden via de Setup.exe op de cd. Het duurt enige tijd voor de installatie zal starten. Gedurende de installatie zullen bij moderne besturingssystemen foutmeldingen komen. Ondanks de meldingen zal de installatie wel slagen.

### Project aan maken

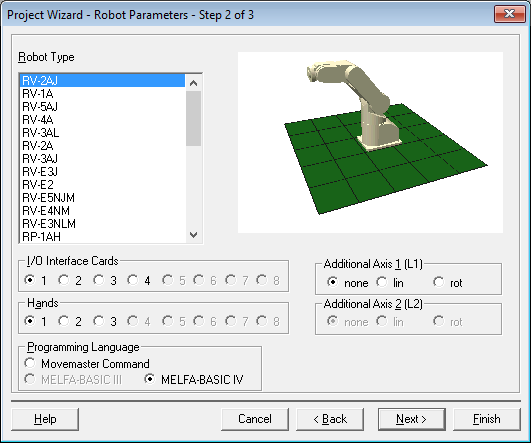
Via File-> Project Wizard kan de wizard gestart worden.

#### Stap 1

Hier zijn geen verplichte keuzes, echter dient de locatie van het project aangepast te worden naar een locatie waar de Windows gebruiker schrijf rechten heeft.

#### Stap 2

Pas de instellingen aan zoals te zien in Figuur 2.



Figuur Instellingen voor stap 2

#### Stap 3

Deze stap kan overgeslagen worden. Voltooi de wizard met de knop “Finish”.

### De robot verbinden

Om de robot te verbinden dient bij Extra’s->Settings-> Communication Port de poort ingesteld te worden met de daarbij behorende baudrate. De baudrate is te vinden in het volgende hoofdstuk. Vervolgens kan via Execute-> init connection de verbindingen worden opgezet.

# Configuratie COM-poort

Voor de communicatie via de RS-232C poort dient de poort te worden geconfigureerd. Dit kan door middel van de Teaching Pendant of via de COSIMIR software. Om het met de COSIMIR software te configureren dient in de RCI Explorer gebruik onder RV-2AJ parameter aangeklikt te worden en kunnen de benodigde parameters aangepast worden.

Welke parameters dit zijn, waarop ze ingesteld moeten zijn en wat het betekend is te vinden in de Tabel 1 Overzicht van de in te stellen parameters (Mitsubishi Electric, 2005). Om met deze communicatie methode de robot te laten bewegen moet de sleutelschakelaar op de controller op auto(ext) staan.

Tabel Overzicht van de in te stellen parameters (Mitsubishi Electric, 2005)

|  |  |  |  |
| --- | --- | --- | --- |
| Parameter | Standaard waarde | Nieuwe waarde | Betekenis |
| COMDEV | RS232, , , , , , , | RS232, OPT11, , , , , , | Zet het communicatie kanaal open voor invoer van buiten. De lege ruimte is om extra invoer van de controller te activeren. OPT11 is om op slot 1 de COM poort uit te lezen. |
| CBAU232 | 9600 | 19200 | Baudrate van de COM poort |
| CPRTY232 | 2 | 2 | Pariteit van de verbinding (even) |
| CSTOP232 | 2 | 2 | Aantal stop bits |
| CTERM232 | 0(CR) | 0(CR) | Terminator voor commando’s |
| CPRC232 | 0 | 2 | Communicatie op datalink zetten zodat er in de software gebruik gemaakt kan worden van de COM poort |

# Eigen commando’s

Voor het aansturen van de robot zijn twee mogelijke methodes aanwezig. De eerste optie is om dit te doen via de commando’s die er standaard voor aanwezig zijn (te vinden in hoofdstuk 5). De tweede optie is om dit te doen via eigen Melfa Basic code.   
De eigen Melfa Basic code biedt diverse voordelen:

* Gestuurde commando’s kunnen onderbroken worden
* Bij een fout wordt de controller automatisch gereset
* De software start automatisch bij het starten van de controller
* Simpelere commando’s

Echter vereist het gebruik van deze eigen commando’s extra configuratie. In paragraaf 4.1 wordt dit toegelicht.

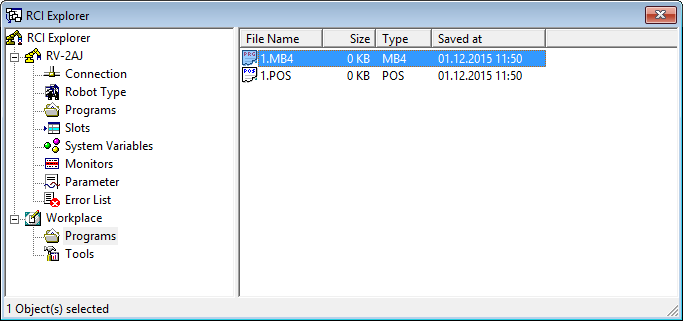
## Code bestanden toevoegen

Standaard maakt het project het bestand 1.MB4 aan. Omdat het met moderne besturingssystemen niet mogelijk is om nieuwe bestanden in COSIMIR op te staan dient op een andere manier een tweede bestand aangemaakt te worden.

Hiervoor moet gebruik gemaakt van de RCI Explorer. Klap in RCI Explorer het veld Workplace uit. Klik met de rechter muisknop op 1.MB4 en kies Copy. Er zal dan een 11.MB4 aangemaakt. Hernoem dit bestand naar 2.MB4.

Plak in deze bestanden de inhoud van de gelijknamige bestanden in de bijlagen en sla deze op.

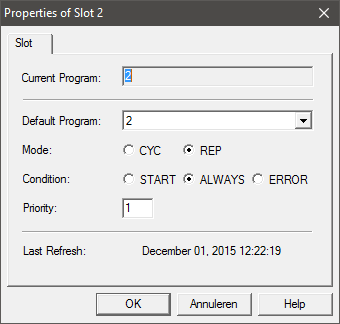
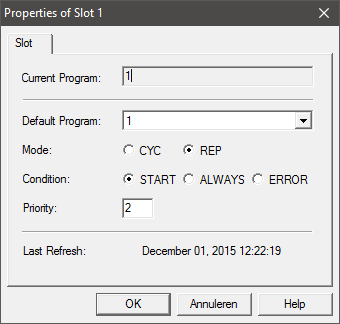
### Uploaden

Om de bestanden naar de robot te uploaden kies in de RCI Explorer voor “Workspace” en vervolgens “Programs”. Om 1.MB4 en 2.MB4 te uploaden kan er met de rechter muisknop op geklikt worden en vervolgens gekozen worden voor Download.

Figuur 3 Voorbeeld RCI Explorer

### Configuratie

Als laatste stap dient die software op de robot geconfigureerd worden. In de RCI Explorer van COSIMIR kan in de sectie “Slots” de verschillende sloten geconfigureerd worden. Door met de rechtermuisknop op een slot te klikken en vervolgens “properties” te kiezen kunnen de instellingen van een slot worden geopend. Zet voor slot 1 en 2 de instellingen zoals getoond in Figuur 4. Herstart de robot om de instellingen toe te passen. De setting “Always” zorgt er ook direct voor dat dit programma niet kan worden overschreven, omdat deze altijd aan staat. Daarom is het belangrijk dat deze setting op “Start” wordt gezet voordat dit programma overschreven kan worden.



Figuur Instellingen voor de sloten

## Werking Melfa Basic code

Beide bestanden die op de controller worden gezet hebben een eigen verantwoordelijkheid.

### Sloten

Op de robot zijn verschillende sloten aanwezig. In elk slot kan een stuk code draaien die parallel zullen worden uitgevoerd.

### 1.MB4

Dit bestand is verantwoordelijk voor de beweging van de robot. Bij de start worden alle gewrichten op 0 gezet. Vervolgens worden de snelheid en acceleratie op maximaal gezet. De code wacht dan tot er vanuit het andere bestand een opdracht gepusht wordt. Zodra een opdracht binnen is wordt deze uitgevoerd en wanneer een nieuwe opdracht binnen komt wordt de uitvoer gestopt. Daarna springt de software terug naar het wachten op een nieuwe opdracht.

### 2.MB4

Dit bestand is verantwoordelijk voor de flow van de Melfa Basic software. Hierin wordt 1.MB4 gestart. Ook wordt een interrupt aangemaakt voor wanneer er fouten optreden en wordt de COM-poort geopend. Vervolgens wordt gewacht tot er een opdracht binnenkomt via de seriële verbinding. Als er een opdracht binnenkomt wordt gekeken welk type opdracht dat het is en aan de hand daarvan de doelpositie bepaald. Deze wordt naar 1.MB4 gepusht en 2.MB4 stuurt dan een bevestiging terug over de seriële verbinding.

## Fout afhandeling

Wanneer er een fout optreedt, zal de code in 1.MB4 stopen met uitvoeren, 2.MB4 zal echter doorlopen en een interrupt ontvangen. Bij die interrupt komt 2.MB4 in de interrupt routine van de error. Hierin wordt via de seriële verbinding een bericht verstuurd over de error dat op treed in de vorm van: “Qer [errornummer]”. Dan zal geprobeerd worden de error te resetten en een seconde gewacht worden wanneer daarna de error nog niet verholpen is wordt dit nog eenmaal geprobeerd. Dit gaar door tot de robot herstart wordt of de error daadwerkelijk is verholpen. Daarna wordt 1.MB4 opnieuw gestart waarna er weer opdracht gestuurd kunnen worden.

## Commando’s voor de code

Commando’s dienen verstuurd te worden over de RS-232 met de syntax: “PRN commando”.

|  |  |  |
| --- | --- | --- |
| Commando | Return | Effect |
| 1,(0,0,0,0,0,0,0) | QoK (0,0,0,0,0,0) | Beweegt de robot naar de opgegeven positie. De doelpositie wordt meegegeven bij de return. |
| 2,(0,0,0,0,0,0,0) | QoK (0,0,0,0,0,0) | Beweegt de robot relatief naar de opgegeven positie. De doel positie wordt meegegeven bij de return |

# Standaard commando’s

Melfa heeft standaard een mogelijkheid ingebouwd om de robot vanuit COSIMIR aan te sturen. Deze commando’s zijn ook te gebruiken om vanuit andere software de commando’s aan te roepen. Deze commando’s zijn door Melfa niet gedocumenteerd en zijn verkregen door te luisteren op de communicatie tussen COSIMIR en de controller. Dus bij een mogelijke software update kunnen deze commando’s stoppen met functioneren.

## Commando’s

De onderstaande commando’s kunnen worden uitgevoerd om de robot te bewegen. Hiervoor zullen de eerste vier commando’s eerst moeten worden uitgevoerd voor dat de andere commando’s uitgevoerd kunnen worden.

Tabel Beschrijving van niet gedocumenteerde commando's

|  |  |
| --- | --- |
| Commando | Effect |
| OPEN=usertool | Verbinding openen |
| CNTLON | Control starten |
| SRVON | Servo aan |
| EXECJOVRD 100.0 | Snelheid instellen |
| EXECJCOSIROP=(-90.00,-60.00,-30.00,-30.00,-30.00,-30.00) | Joint Coördinaten instellen |
| EXECMOV JCOSIROP | Arm bewegen coördinaat |
| RUNnaam;slotnr | Start een programma in een slot |
| JPOSF | Joint posities |
| STATE | Huidige toestand van de arm |
| PPOSF | Huidige coördinaten |
| EXECPCOSIROP=(70.00,0.00,782.00,0.00,0.00,0.00)(6,0) | XYZ Coördinaten instellen |
| EXECMOV PCOSIROP | Arm bewegen naar coördinaat |
| CNTLOFF | Control uit |
| SRVOFF | Servo uit |
| RSTALRM | Alarm resetten |
| EXECSPD 200.0 | Uitvoersnelheid instellen |
| EXECMVS PCOSIROP | Jog arm beweging |
| PARRLNG | Onbekend |

## Verzenden

Om een commando te verzenden zal het commando verstuurd moeten worden naar een specifieke robot en een specifiek slot. Dit is nodig omdat er in theorie meerdere robots op één com-poort verbinding aangesloten kunnen zijn. Daarnaast kan de software draaien in verschillende sloten (zie alinea 4.2.1). Dus de syntax voor het verzenden van de bovenstaande commando’s zal zijn: “1;1;Commando”. Waarbij het eerste getal staat voor het robotnummer en het tweede getal is het gekozen slot.

## Configuratie

Omdat de genoemde commando’s normaal gesproken gebruikt worden voor de communicatie tussen de controller en de COSIMIR software, is de COSIMIR software nodig om het te configureren. De commando’s worden opgevangen door een standaard code bestand genaamd COSIROP. Om dit bestand te herstellen op de robot moet in COSIMIR in de RCI-Explorer, onder Workplace > Tools, gekozen worden voor de “Jog-operation”. Dan zal de software de bestanden op de robot plaatsen. Dan kan COSIMIR gesloten worden.

## Voorbeeld

Hieronder een voorbeeld om de robot, bijvoorbeeld vanuit c++, naar een bepaalde positie te bewegen met de standaard commando’s:

1;1;OPEN=usertool<\r>

1;1;CNTLON<\r>

1;1;SRVON<\r>

1;1;EXECJCOSIROP=(-90.00,-60.00,-30.00,-30.00,-30.00,-30.00)<\r>

1;1;EXECMOV JCOSIROP<\r>

# Aansturing in C++

In de c++ code zal een seriële poort worden geopend. Deze moet dezelfde instellingen hebben als de robot. Tijdens dit project is er gebruik gemaakt van de Boost library. Hierna wordt een string verstuurd naar de robot. Deze string zal van het volgende formaat zijn: “PRN 1,(0,0,0,0,0,0)\r” of “PRN 2,(0,0,0,0,0,0)\r”. Het eerste commando zal worden gestuurd wanneer alle assen moeten worden gezet. Bij het tweede commando worden alleen de assen gezet die niet 0 zijn.

# Simulatie

Voor de robot is een simulatie geschreven. Dit is handig, aangezien maar een robot beschikbaar was. Hierdoor kunnen toch meerdere mensen tegelijkertijd werken aan een aansturingsapplicatie voor de robotarm.

De 3D simulatie kan gedraaid worden door met een willekeurige muisknop ingedrukt te slepen. Zoomen kan door te scrollen.

## Communicatie naar simulatie

De communicatie naar de simulatie toe verloopt via een socket. Het is immers niet logisch om hiervoor een COM-poort te gebruiken, aangezien het vaak op dezelfde PC draait. Daarnaast kan bij de socket-connectie eenvoudig localhost, oftewel IP: 127.0.0.1 gebruikt worden met poort: 63258. Indien het netwerk/firewall het toelaat (poort openen/forwarden), zou de simulatie dus ook via het netwerk aangesproken kunnen worden.

Als via de (lokale)-socket een bericht: ”*PRN 1,(1,2,3,4,5,6)\r “* wordt gestuurd (waarin” \r” een carriage return is), zal de robot de motoren op de hoeken (in graden) 1,2,3,5,6 zetten. Ook hier geldt dat J4 niet gezet wordt (deze is bij de Melfa RV2-AJ niet aanwezig).

Het PRN 2 bericht is op de simulatie niet beschikbaar. De simulatie geeft geen “QoK” terug bij het ontvangen van een commando. De simulatie is een server en moet dus eerder worden opgestart als de client (indien deze verbinding maakt bij opstarten).

Vanuit de simulatie kunnen X,Y,Z coördinaten (in millimeters) worden gestuurd naar de robot. Deze coördinaten (bijvoorbeeld x 100, y 200, z 300) worden verpakt in een bericht als: *100;200;300;\r* (waarin \r een carriage return is). Dit bericht wordt gestuurd naar alle verbonden clients. Ook zal dit punt in de simulatie worden weergeven als een rood balletje.

## Simulatie naar robot

De simulatie kan ook de robot aansturen. Bij het opstarten zoekt de simulatie naar COM-poorten en zal het de eerst gevonden COM-poort binden met baudrate: 19200, parity: even, databits: 8 en stop bits: 2.

De simulatie en een ander programma kunnen niet tegelijkertijd verbinding maken met de COM-poort. (De simulatie zal standaard bij opstarten proberen de eerst gevonden COM-poort te binden)

## Achtergrond

De simulatie is gemaakt in Processing 3.0.1 (Java). Voor de simulatie moet de juiste versie van Java geïnstalleerd zijn, 32-bits/64-bits, Java 8. Belangrijk is dat de .OBJ (3D bestanden) zich in dezelfde map als het programma bevinden.

# Onderhoud en fouten

De controller is uitgebreid gedocumenteerd wat betreft fouten. De controller geeft deze fouten weer op het display. De fouten zijn te resetten met behulp van de “Error Reset” knop op de remote. De fout code is op te zoeken in de Instruction Manual – Troubleshooting (boekje 9 van de 9). Niet alle fouten staan uitgelegd in de Troubleshooting manual. Hieronder staat één van deze fouten toegelicht.

## Onderhoudsprogramma’s

De robot moet regelmatig gecontroleerd worden op defecten en losse onderdelen. Het vereiste onderhoud staat in de Instruction Manual – Robot arm setup en maintenance (boekje 3 van de 9). In bijlage 1 en 2 staat een korte samenvatting van het periodieke onderhoud.

## De standaard commando’s werken niet

Wanneer de standaard commando’s niet werken kunnen er diverse problemen zijn opgetreden.

### De code voor de eigen commando’s staan op de robot en zijn geconfigureerd

Wanneer de code voor de eigen commando’s draait, is het niet mogelijk om de standaard commando’s uit te voeren. De code voor eigen commando’s wordt gestart wanneer de robot wordt opgestart, om dit te voorkomen kan de robot opgestart worden met de noodknop ingedrukt. Maar zodra er dan een fout op treed zal de code voor eigen commando’s een interrupt krijgen en gestart worden. Daarna is het niet meer mogelijk om de standaard commando’s te versturen.

### Er komt geen reactie

Wanneer er geen reactie komt bij het versturen is het waarschijnlijk dat er een fout is in de configuratie van de seriële verbinding. Controleer de seriële verbinding en welke endlines worden meegezonden.

### Er komt een Qer terug

Een Qer wordt altijd gevolgd door een fout code. Deze foutcode is op te zoeken in de Instruction Manual – Troubleshooting (boekje 9 van de 9) voor een exacte oorzaak van het probleem.

## Weergave FAIL op het display

Wanneer de batterijenspanning de instellingen van de uitgeschakelde robotcontroller niet meer voldoen, geeft het display na het inschakelen de foutmelding FAIL aan. Om dit probleem op te lossen moet er een RAM data reset plaatsvinden. Na deze reset zijn alle programma’s, positiedata in de robotcontroller verloren en de parameters naar de fabrieksinstellingen teruggezet.

De procedure is als volgt:

* Druk de STOP en RESET knop tegelijk in, tijdens het inschakelen van de controller.
* Het display geeft “|---|” weer.
* Laat de STOP en RESET knop los.
* Druk de RESET knop zo vaak in totdat het display “EnG” weergeeft.
* Druk een keer op de START knop.
* Druk op de RESET of STOP knop totdat het display “2AJ” weergeeft.
* Druk een keer op de START knop.
* Het display geeft “rAMc9” weer.
* Na ongeveer 15-25 seconden (afhankelijk van de hoeveelheid RAM geheugen) telt het display af van 100 tot 0.
* Het display geeft “ok” weer.
* De controller zal zich nu opnieuw op gaan starten.

# Bijlagen

## 1 Daily inspection items

Tabel 3 Daily inspection items (Mitsubishi Electric Corporation, 2004)

|  |  |  |  |
| --- | --- | --- | --- |
| Procedure | Inspection item (details) | | Remedies |
| Before turning power ON (Check the following items before turning the power ON.) | | | |
| 1 | Are any of the robot installation bolts loose? | (Visual) | Securely tighten the bolts. |
| 2 | Are any of the cover tightening screws loose? | (Visual) | Securely tighten the screws. |
| 3 | Are any of the hand installation bolts loose? | (Visual) | Securely tighten the bolts |
| 4 | Is the power supply cable securely connected? | (Visual) | Securely connect. |
| 5 | Is the machine cable between the robot and controller securely connected?  (Visual) | | Securely connect. |
| 6 | Are there any cracks, foreign contamination or obstacles on the robot and controller cover? | | Replace with a new part, or take remedial measures. |
| 7 | Is any grease leaking from the robot arm? | (Visual) | After cleaning, replenish the grease. |
| 8 | Is there any abnormality in the pneumatic system? Are there any air leaks, drain clogging or hose damage? Is the air source normal?  (Visual) | | Drain the drainage, and remedy the air leaks (replace the part). |
| After turning the power ON (Turn the power ON while monitoring the robot.) | | | |
| 1 | Is there any abnormal motion or abnormal noise when the power is turned ON? | | Follow the troubleshooting section. |
| During operation (try running with an original program) | | | |
| 1 | Check whether the movement points are deviated? Check the following points if there is any deviation.  1. Are any installation bolts loose?   1. Are any hand installation section bolts loose? 2. Are the positions of the jigs other than the robot deviated? 3. If the positional deviation cannot be corrected, refer to "Troubleshooting", check and remedy. | | Follow the troubleshooting section. |
| 2 | Is there any abnormal motion or abnormal noise? | (Visual) | Follow the troubleshooting section. |

## 2 Periodic inspection

Tabel 4 Periodic inspection items (Mitsubishi Electric Corporation, 2004)

|  |  |  |
| --- | --- | --- |
| Procedure | Inspection item (details) | Remedies |
| Monthly inspection items | | |
| 1 | Are any of the bolts or screws on the robot arm loose? | Securely tighten the bolts. |
| 2 | Are any of the connector fixing screws or terminal block terminal screws loose? | Securely tighten the screws. |
| 3 | Remove the cover at each section, and check the cables for wear damage and adherence of foreign matter. | Check and eliminate the cause.  If the cables are severely damaged, contact the Mitsubishi Service Department. |
| 3-month inspection items | | |
| 1 | Is the timing belt tension abnormal? | If the timing belt is loose or too tense, adjust it. |
| 6-month inspection items | | |
| 1 | Is the friction at the timing belt teeth severe? | If the teeth are missing or severe friction is found, replace the timing belt. |
| Yearly inspection items | | |
| 1 | Replace the backup battery in the robot arm. | Exchange it referring to "5.3.5 Replacing the backup battery" on page 54. |
| 2-year inspection items | | |
| 1 | Lubricate the grease at the harmonic reduction gears for each axis. | Lublicate it referring to "5.3.4 Lubrication" on page  52. |