Inbetriebnahme MR-J4-TM Verstärkers unter Rockwell Compact Logix

Version:1.0

Inhalt

1 Haftungsausschluss **3**
1.1 Verwendete Geräte . . . 3
1.2 Verwendete Software und Dateien . . . 3

2 Aufbau einer Ethernet/IP Verbindung **4**
2.1 Installation EDS Files . . . 4
2.2 IP-Adressen . . . 5
2.2.1 Netzwerkverbindung mit RS Linx . . . 5
2.2.2 IP-Adresse Verstärker . . . 7
2.2.3 Erstellung eines Projektes und IP-Adresse der CompactLogix . . . 9
2.3 Komponenten in RS Logix 5000 hinzufügen . . . 12
2.3.1 Verbindungstest . . . 15

3 Programmierung **16**
3.1 AOI's importieren . . . 16
3.2 Main Routine . . . 17
3.3 JOG Routine . . . 21
3.4 Move Routine . . . 26

4 Troubleshooting **29**
4.1 Fehlerbehebung im Programm . . . 29
4.2 Fehlerbehebung am Verstärker . . . 30

5 Tuning Möglichkeiten **32**
5.1 One touch tuning . . . 32
5.2 Andere tuning Möglichkeiten . . . 33

1 Haftungsausschluss

Dieses Dokument soll die Inbetriebnahme eines Mitsubishi Electric Servo Verstärkers, unter einer Rockwell SPS erleichtern. Die in diesem "quick start guide" erstellten Programme haben keinerlei Sicherheitsfunktionen und sind nicht für die Automatisierung geeignet. Treffen Sie immer Sicherheitsvorkehrungen, bevor Sie mit einem Motor arbeiten.

1.1 Verwendete Geräte

Im diesem Guide wird die Rockwell SPS **CompactLogix 5370L2** in Kombination mit einem **MR-J4-TM-EIP** an einem **Windows 7 PC** in Betrieb genommen.

Zur Programmierung wird ein **Ethernet** oder **USB 2.0 Kabel** benötigt.

Für die Kommunikation zwischen dem Verstärker und der SPS , wird ein **Ethernet Kabel** benötigt. Außerdem, ein **USB 2.0 Mini-B** Kabel[1] mit beidseitigen Ferritkernen zum Auslesen des Verstärkers.

1.2 Verwendete Software und Dateien

Zur Programmierung wird die Rockwell Software **RSLogix™** verwendet.

Für die Konfiguration des EthernetIP Moduls des Verstärkers, wird die Software **IP Config** verwendet.

Zur Registrierung des MR-J4-TM Verstärkers, werden **EDS Dateien** (Electronic Data Sheet) benötigt.

Um die Funktionsbausteine zum Ansteuern des Servos verwenden zu können, müssen die **AOI's**(Add-On Instructions) importiert werden.

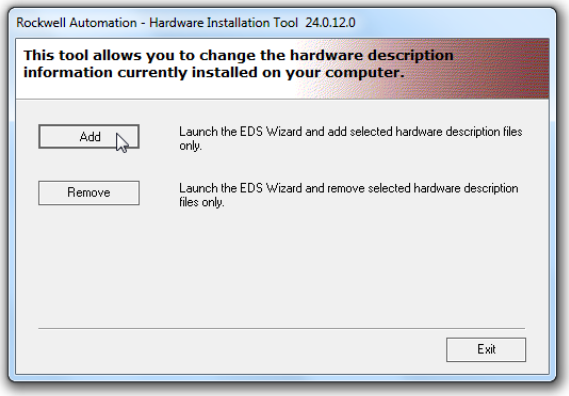
Hier kann benötigte Software und Anleitungen heruntergeladen werden. [2]

[1] USB Kabel: 160229 MR-J3USBCBL3M
[2] https://de3a.mitsubishielectric.com/opennetwork

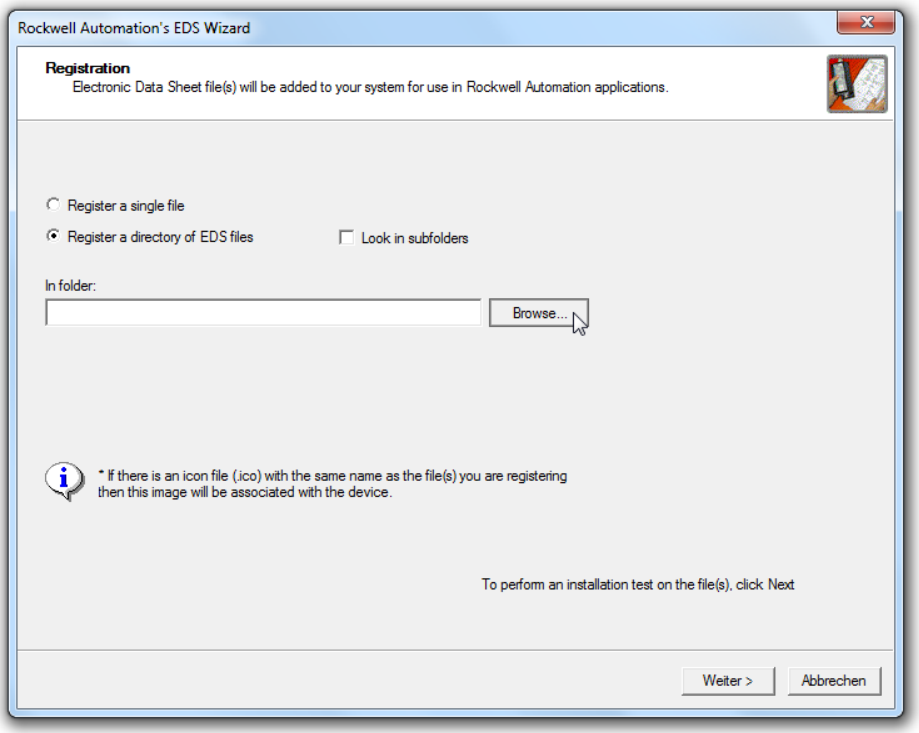
2 Aufbau einer Ethernet/IP Verbindung

2.1 Installation EDS Files

Bevor der Verstärker verwendet werden kann, müssen die EDS Files Installiert werden. Hierzu benötigt werden: **EDS Files**[3] in einem Ordner (Entpackt) und das **EDS Hardware Installation Tool**, welches im Standartpaket der Studio 5000 Software enthalten ist. Nach öffnen der Software wird über **Add** ein neues EDS File hinzugefügt.



Zum Registrieren einer neuen Datei, muss die Option **Register a directory of EDS files** ausgewählt werden. Mit einem Klick auf **Browse**, kann nun der Ordner mit den EDS Files ausgewählt werden, in dem die heruntergaeldadene Zip Datei Entpackt wurde. Mit **weiter** wird der Dialog fortgesetzt.

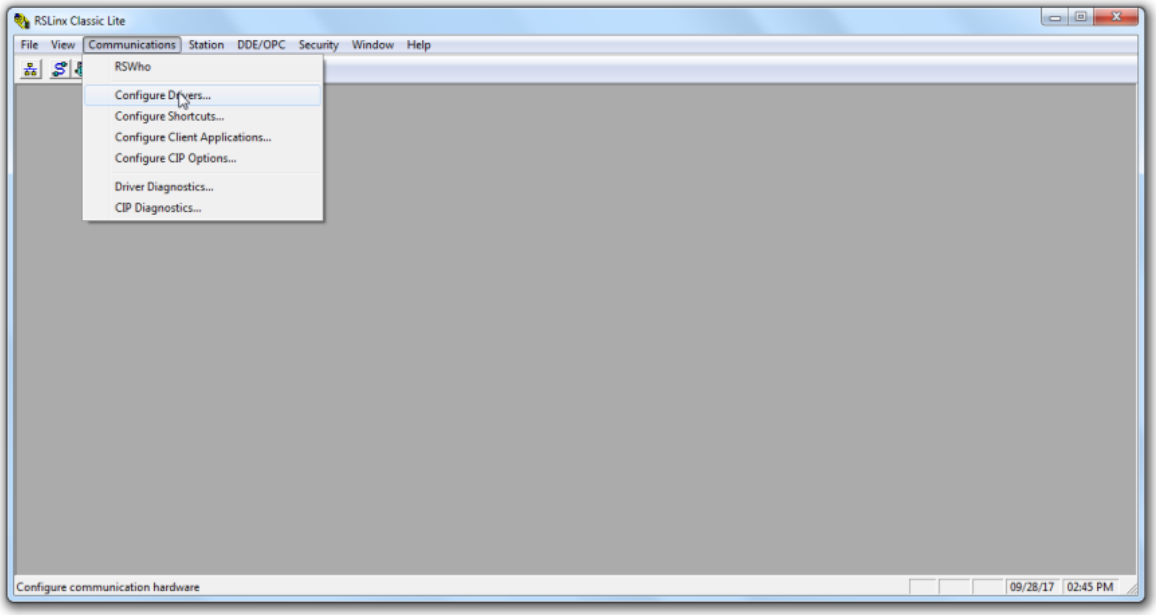


[3]Siehe Kapitel 1.2

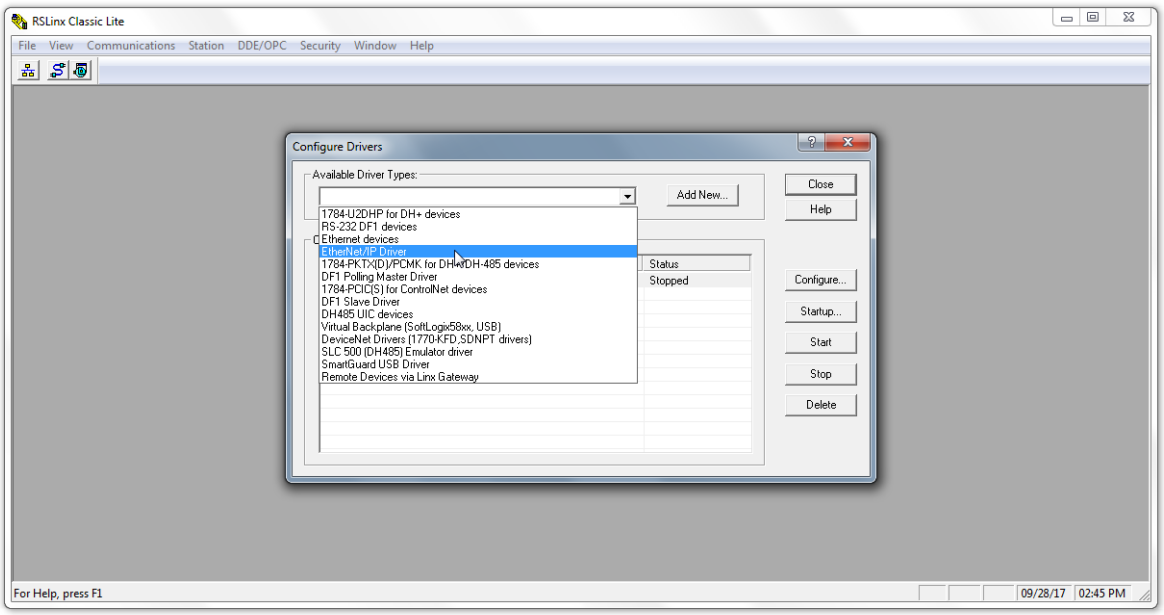
2.2 IP-Adressen

2.2.1 Netzwerkverbindung mit RS Linx

Damit die Steuerung in RSLogix 5000 angezeigt wird, muss über RSLinx eine Verbindung aufgebaut werden. Hierzu muss die Steuerung via Ethernet oder USB an einen PC angebunden werden. Nach dem Öffnen der RSLinx Software, die bei der Installation der RSLogix 5000 Software mitinstalliert wurde, wird mit einem Klick auf **Communications** → **Configure Drivers...** ein neuer Driver konfiguriert.

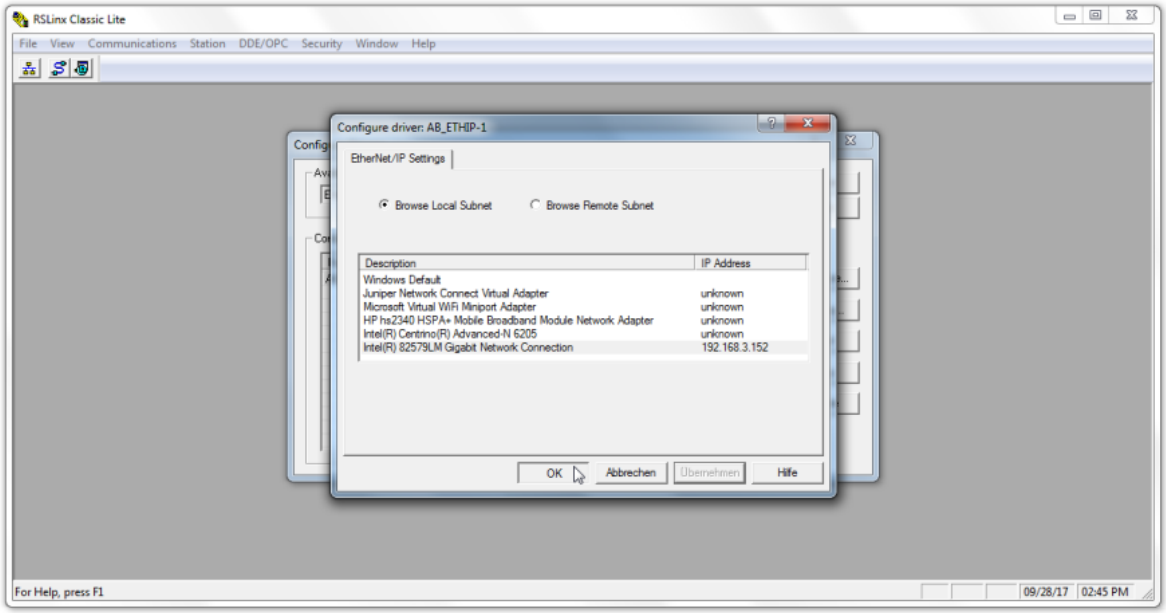


Dann muss ein "Drive Type" ausgewählt werden. Mit dem MR-J4-TM haben wir einen **Ethernet/IP Driver**. Mit **Add New** öffnet sich ein Fenster, indem ein Name, für den neuen "Driver" vergeben werden muss.

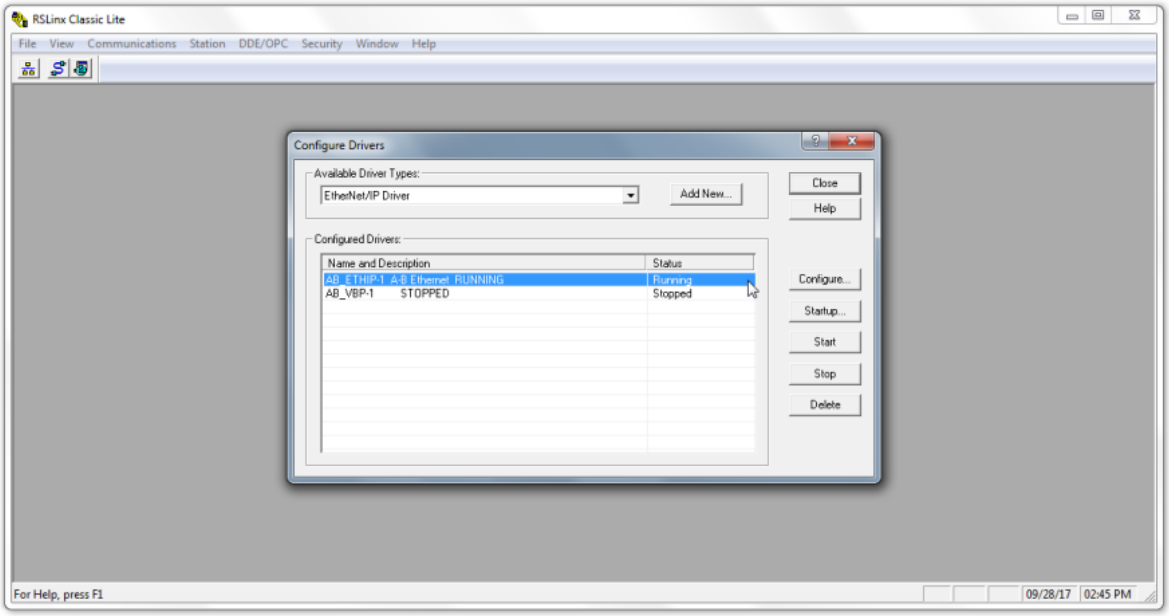


Nach dem bestätigen des Namens mit **OK**, öffnet sich ein Fenster, indem die Schnittstelle zwischen PC und Steuerung ausgewählt werden muss.

IP-Adresse und Module könne sich unterscheiden.

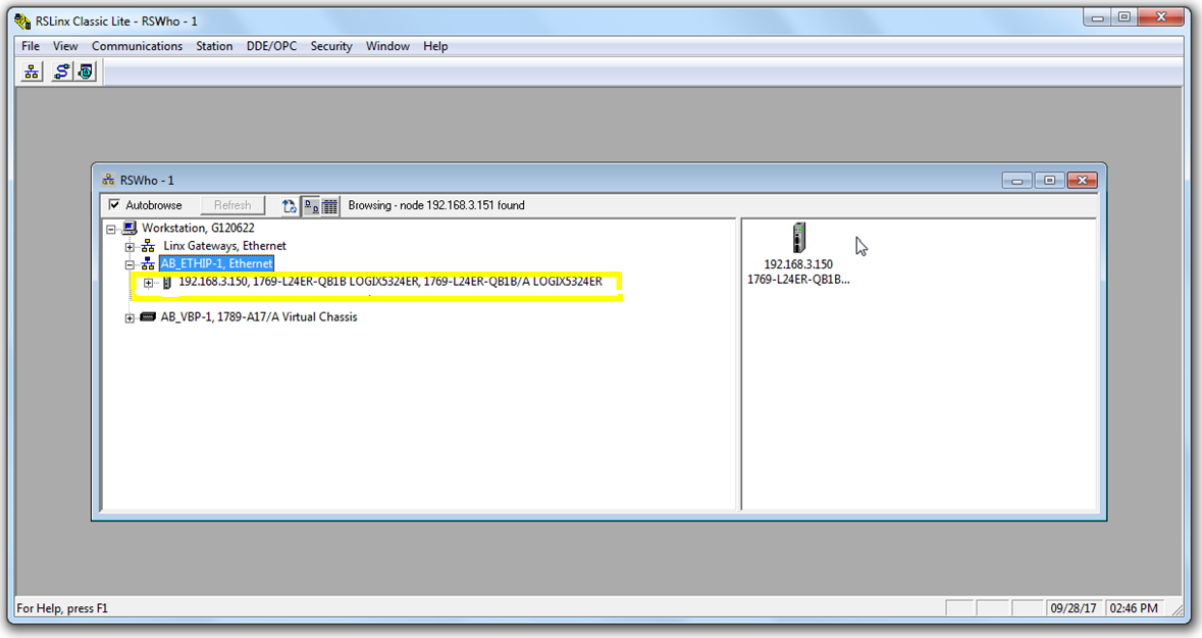


Nach der Bestätigung, sollte der Status des eben konfigurierten "Drives" auf „Running" stehen.



Ob der Verstärker von der Steuerung erkannt wurde, kann nun über einen Klick auf das Ikon mit den 3 Gelben Kästchen überprüft werden

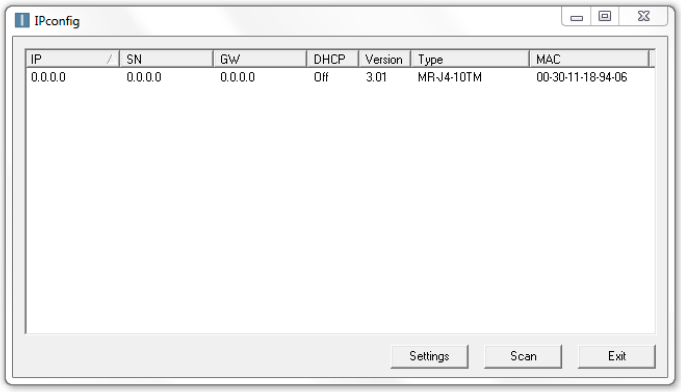
Die IP-Adressen können sich unterscheiden. Wenn der Verstärker nicht erkannt wurde, wird der Eintrag rot gefärbt. **Hier kann die Aktuelle IP Adresse der Steuerung, im rechten Teil des Fensters, abgelesen werden.** Der "Servo Amplifire" wird erst später angezeigt. Er muss zuvor dem Netzwerk hinzugefügt werden.



Tipp:
Sollte die Verbindung nicht aufgebaut werden können, kann dies an der IP4-Adresse des PC's liegen. Passen Sie die IP4-Adresse wenn nötig an.

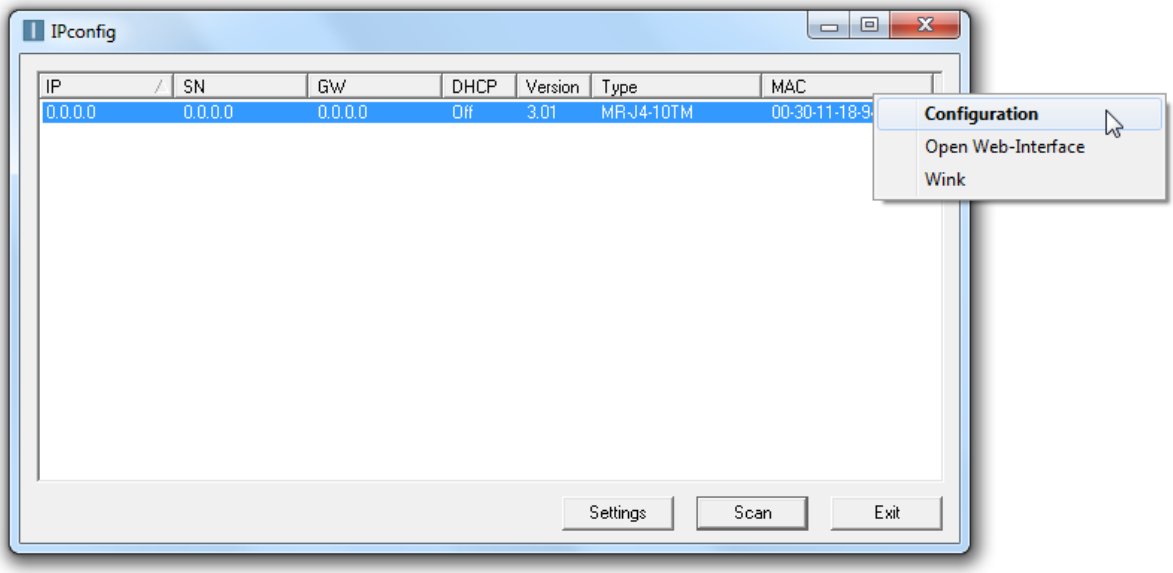
2.2.2 IP-Adresse Verstärker

Wenn der Verstärker via Ethernet mit einem PC verbunden ist[4], wird die IP-Adresse des Verstärkers mit dem **IPconfig tool** eingestellt. Wenn kein Gerät angezeigt wird, kann es helfen die Software neu zu starten oder mit **Scan** die Anzeige zu aktualisieren.

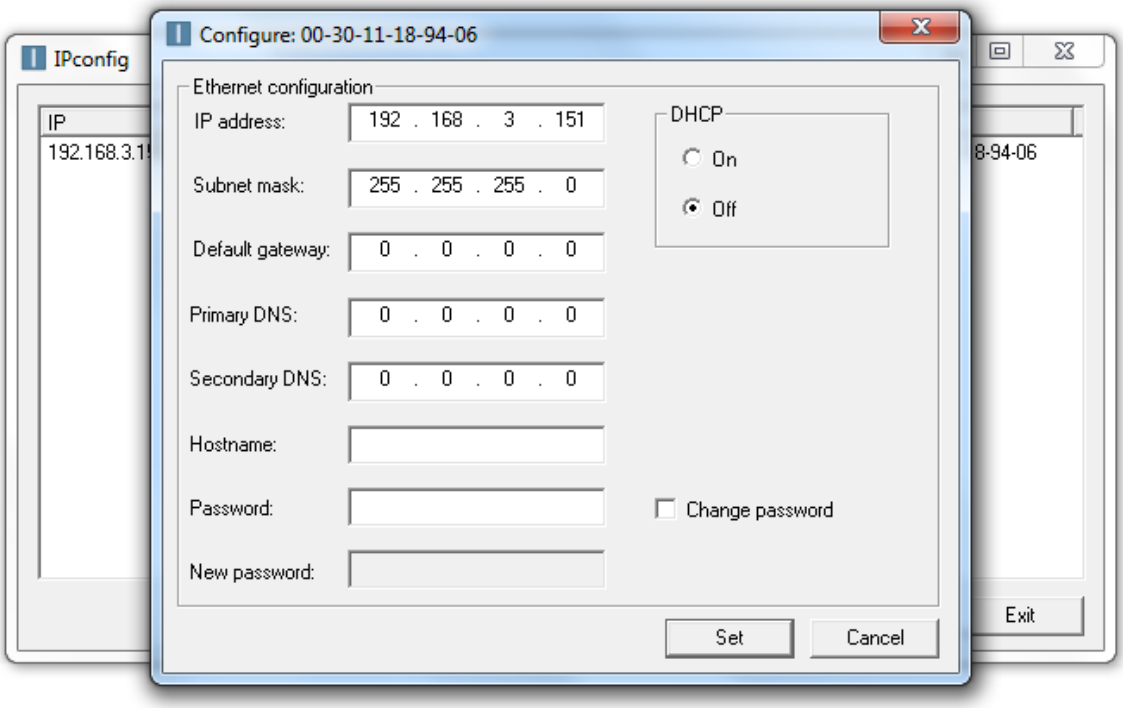


[4]Es ist auch möglich über die SPS auf den Verstärker zuzugreifen. Eine Verbindung per Ethernet Kabel vom Verstärker zur SPS und von der SPS zu einem PC genügt also.

Mit einem Rechtsklick auf den **Verstärker** → **Configuration** kann die IP Konfiguration angepasst werden.



Hier können nun die ersten 3 Gruppen der zuvor ausgelesenen Steuerungsadresse übernommen werden. Die 4. Gruppe kann von 0 bis 255 gewählt werden. **Ausgenommen der schon vergebenden Stellen.** Die "Subnet mask" muss die selbe Bandbreite wie die anderen Teilnehmer des Netzwerks besitzen.
Die IP-Adresse kann auch anders gewählt werden. Es ist zu empfehlen, wenn nicht vorhanden, sich die Netzwerk Grundalgen zu erarbeiten, wenn Sie eigene IP-Adressen vergeben wollen. Ansonsten können Sie sich an den Beispieladressen orientieren. Mit **Set** wird die Eingabe gespeichert.



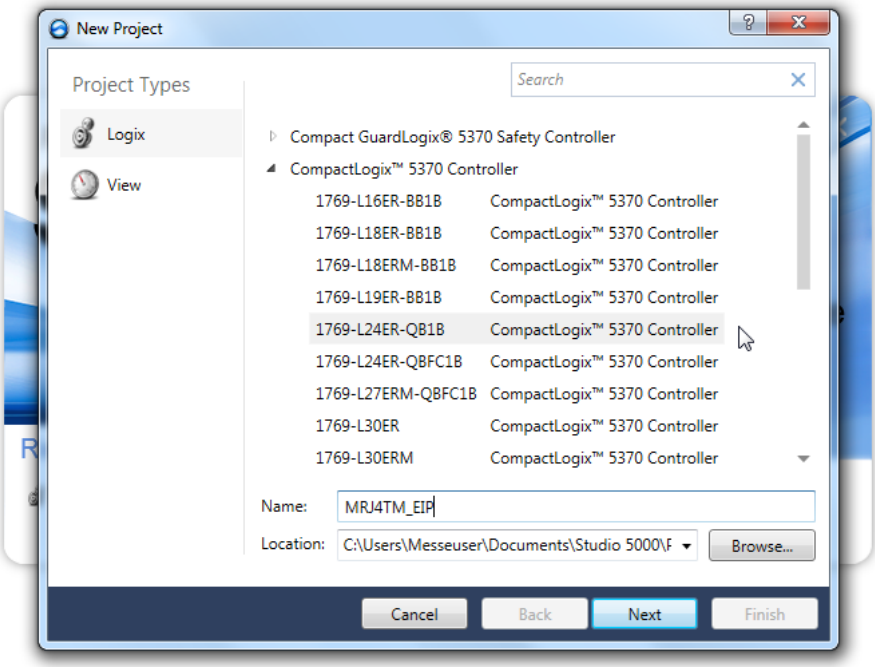
Jetzt sollte der Verstärker mit den eingegebenen Daten angezeigt werden. Wenn nicht, kann noch einmal mit **Scan** die Anzeige aktualisiert werden. Das Fenster wird mit **Exit** geschlossen.

2.2.3 Erstellung eines Projektes und IP-Adresse der CompactLogix

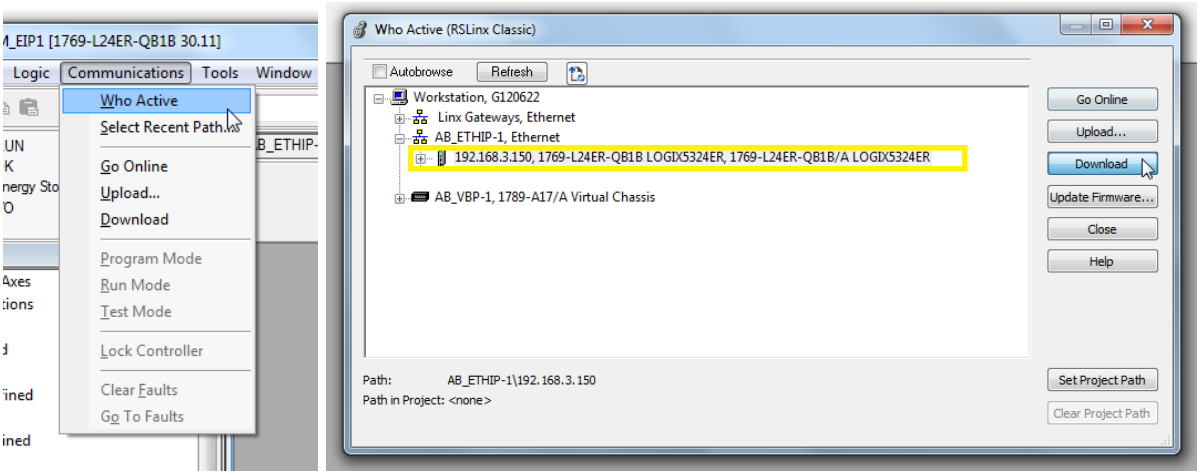
Nach dem Start der Studio 5000 Software muss als erstes ein neues Projekt erstellt werden. Das Display des Verstärkers sollte vor der hergestellten verbindung "Ab" anzeigen.

Mit **New Projekt** lässt sich ein neues Projekt erstellen.

Mit der Auswahl der Verwendeten SPS und der Benennung des Projektes, wird über **Next** → **Finish**, das Projekt erstellt.



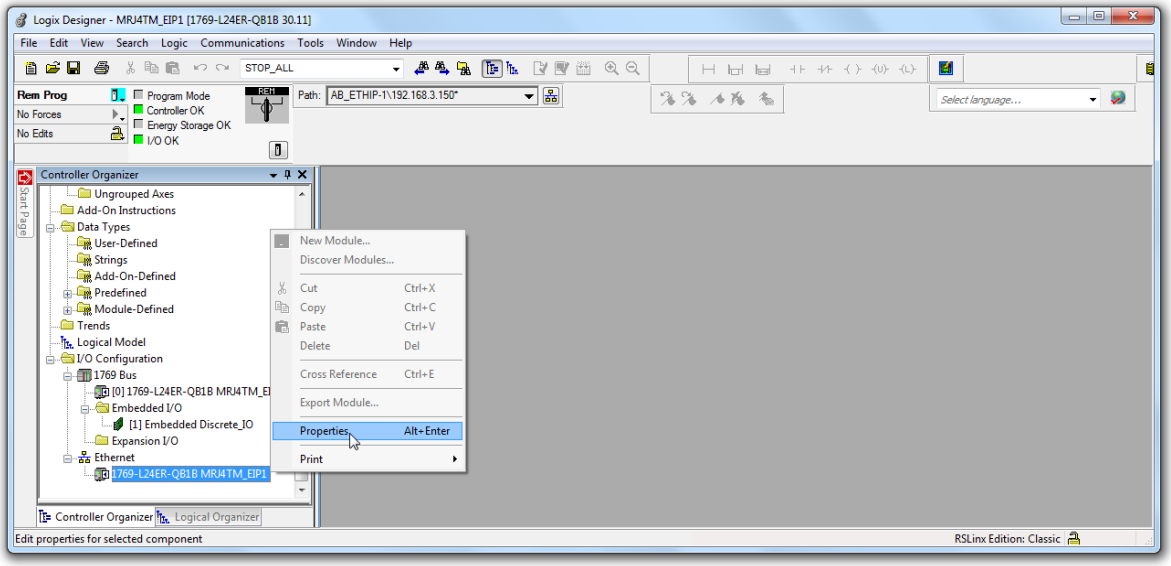
Nach der Erstellung eines Projektes kann, wenn die Steuerung an den PC (über USB oder Ethernet) angeschlossen ist, über **Communications** → **Who active** die Steuerung als Ziel ausgewählt und mit **Download** ein leeres Programm aufgespielt werden.

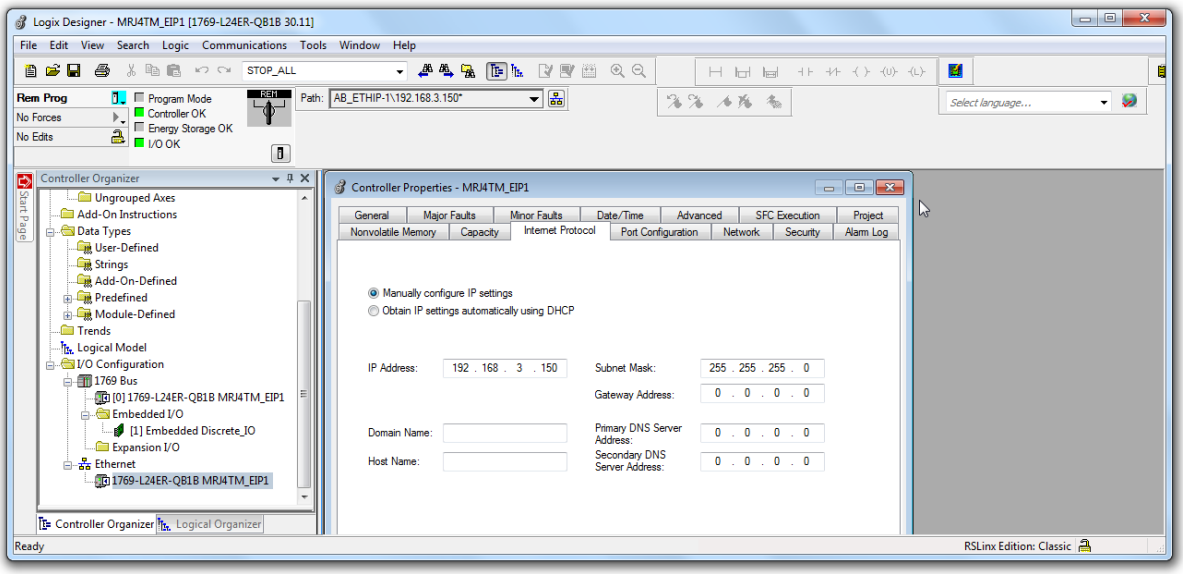


Die Steuerung hat vom Werk aus eine IP-Adresse. Diese kann in der Studio 5000 Software verändert werden. Es kann auch mit der Standard Adresse weiter gearbeitet werden.

Wenn die IP-Adresse der Steuerung angepasst werden soll, kann mit Rechtsklick auf die **Steuerung** → **Properties** → **Internet Protocol** die IP-Adresse geändert werden. Wenn die IP Adresse der Steuerung verändert wird, müssen alle vorherigen Schritte wiederholt werden.

Es ist zu empfehlen, eine Liste über die Änderungen der IP-Adresse zu führen.

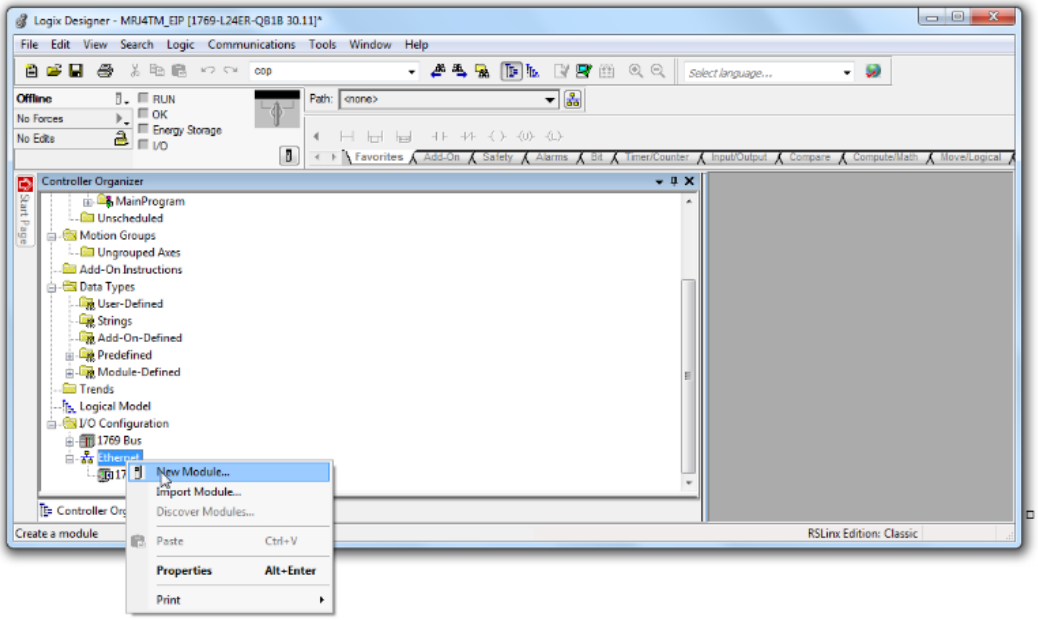




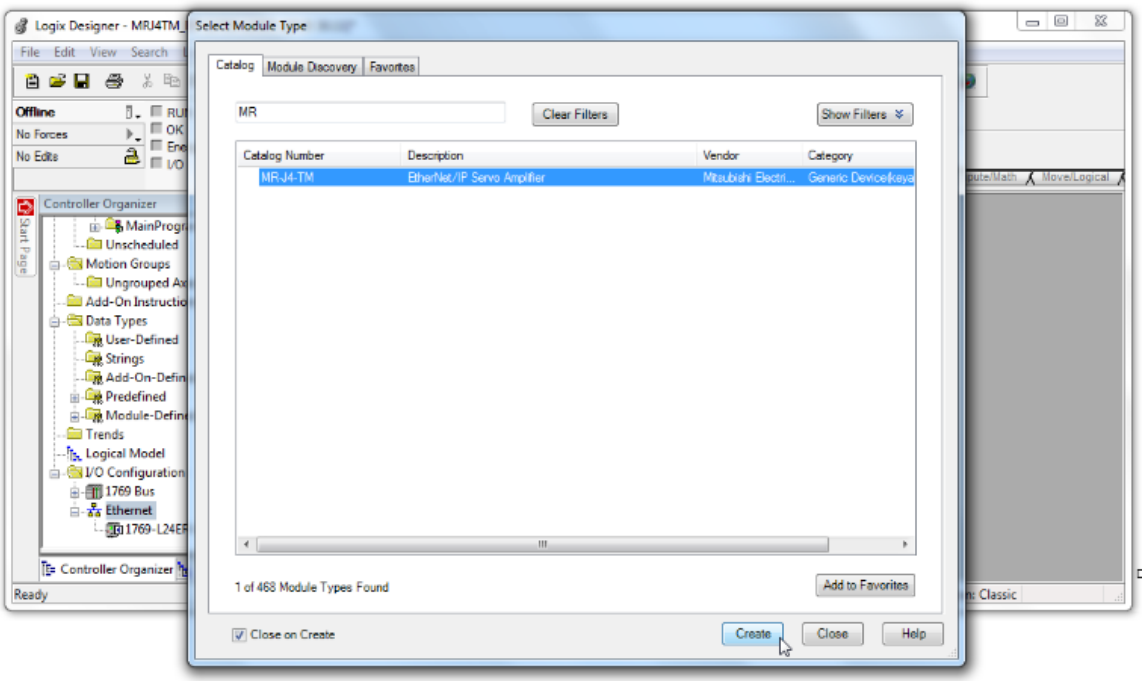
Die Änderung wird mit **Übernehmen** → **OK** gespeichert.

2.3 Komponenten in RS Logix 5000 hinzufügen

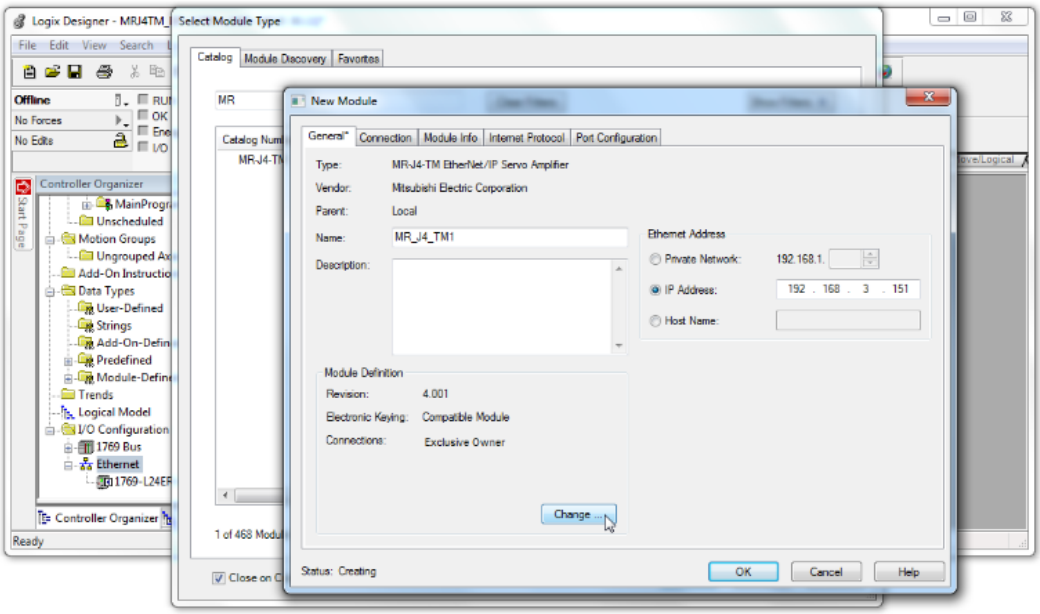
Unter dem Reiter **I/O Configuration** wird die Steuerung angezeigt. Hier wurde die Steuerung über Ethernet angeschlossen. Die Verbindung über ein USB Kabel ist auch möglich. Um eine Verbindung mit dem Verstärker herzustellen, muss der Verstärker mit Rechtsklick auf **Ethernet→New Module** hinzugefügt werden. Der Verstärker muss hierfür per Ethernet Kabel mit der Steuerung verbunden sein.



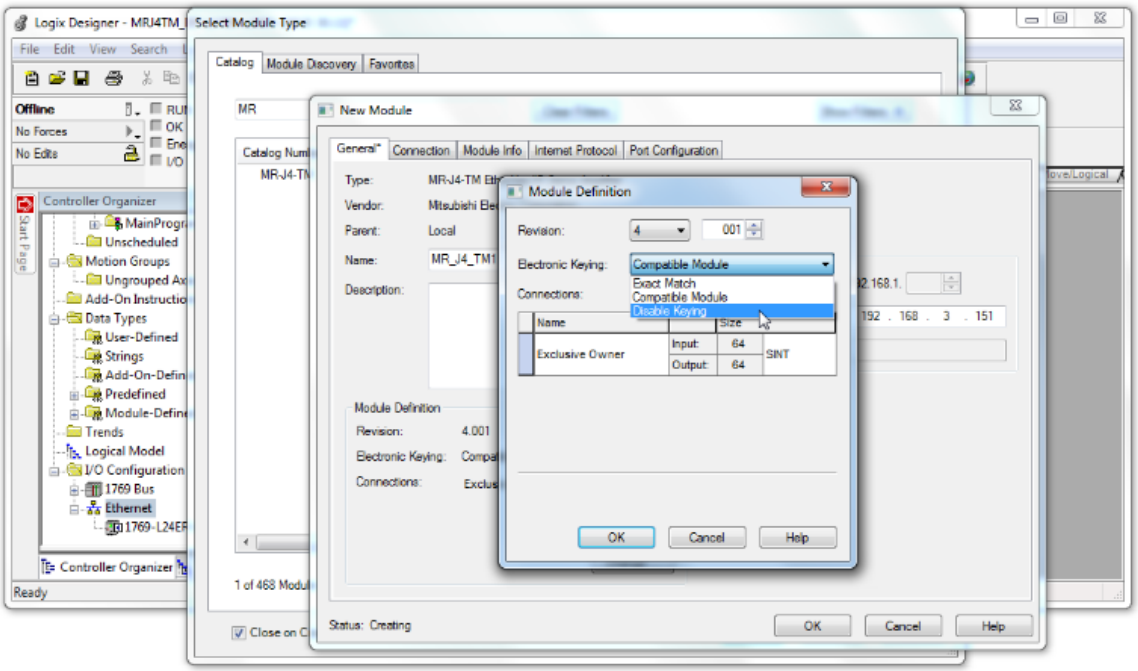
Das gewünschte Modul kann mit **Create** erstellt werden.



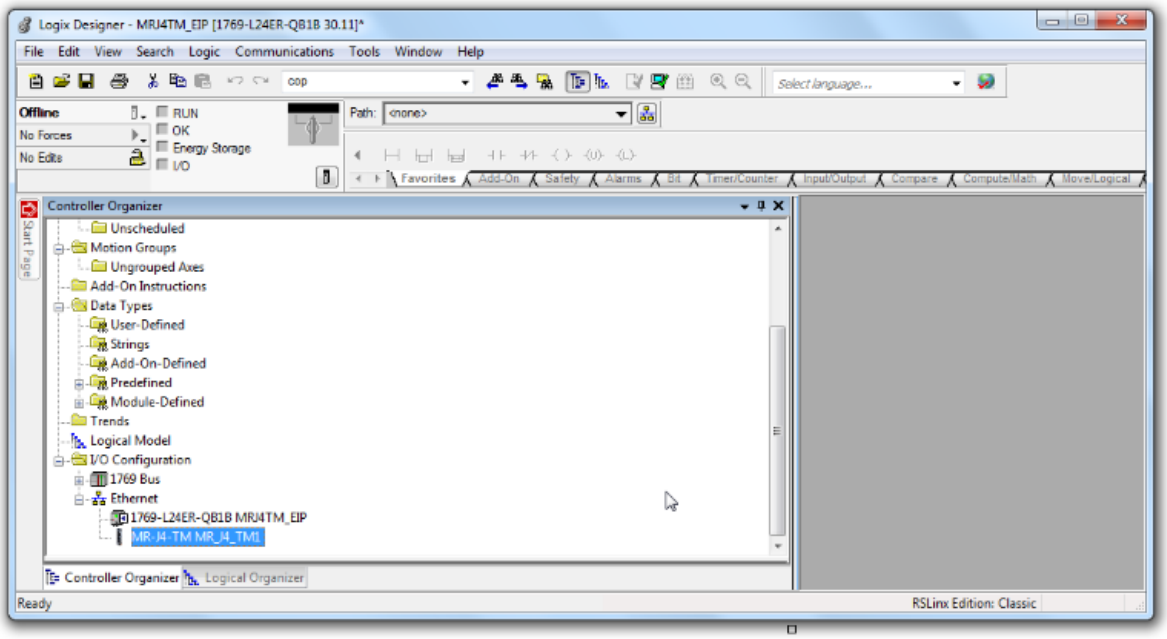
Nun muss ein Name und eine IP Adresse angeben werden. Hier müssen wieder die ersten 3 Gruppen der IPAdresse übernommen werden. Die 4. Gruppe kann zwischen 0 und 255 variieren. Ausgenommen der Stellen die schon vergeben wurden.



An dieser Stelle sollte das Keying deaktiviert werden. Die Oberfläche lässt sich über **Change** öffnen. Die Änderung muss mit **OK** → **OK** bestätigt werden.



Jetzt wird der Verstärker angezeigt. Mit einem anschließenden Verbindungstest kann nun überprüft werden, ob mit dem hinzugefügten Verstärker kommuniziert werden kann.

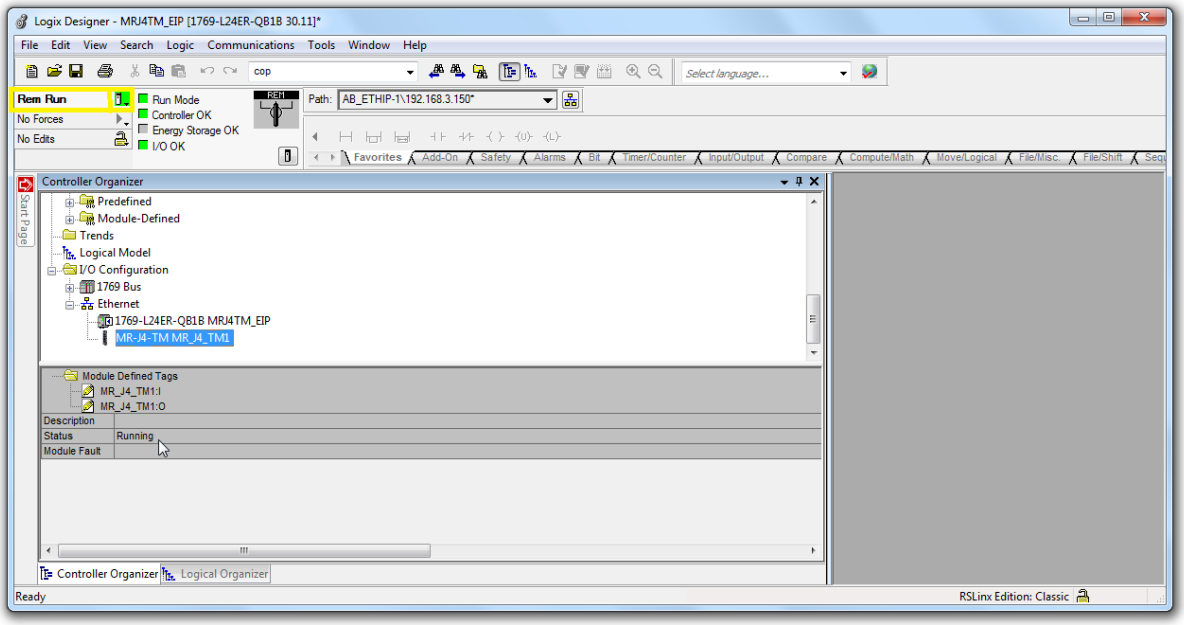


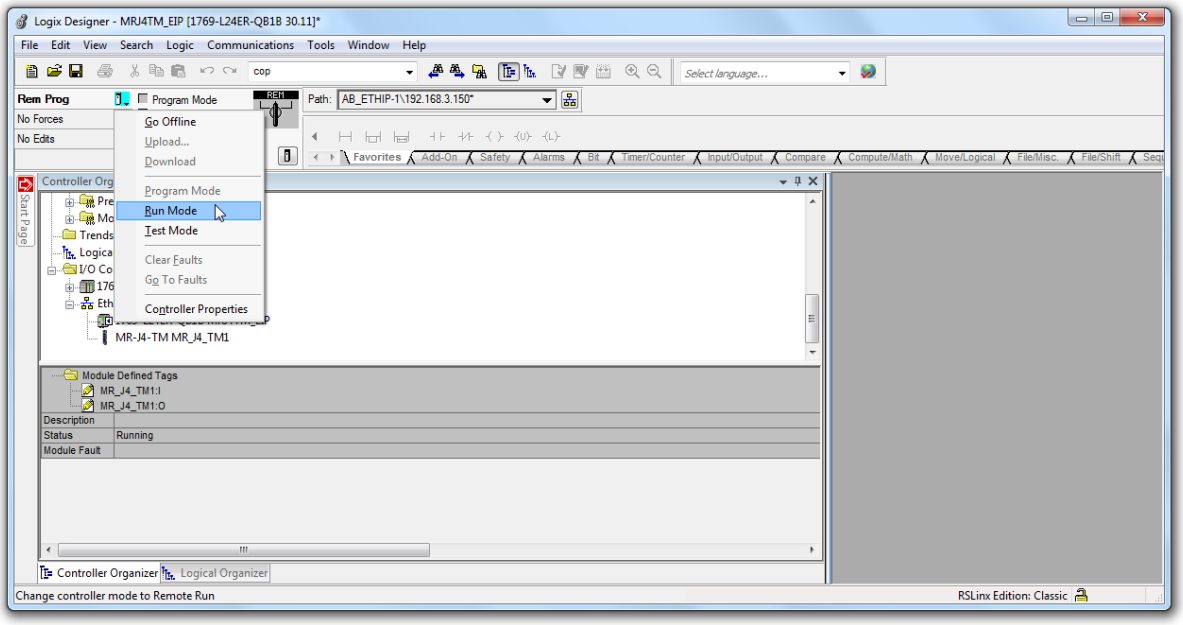
2.3.1 Verbindungstest

Um zu überprüfen, ob die Steuerung mit dem Verstärker kommuniziert, muss mit der Software, wie in 2.2.3 beschrieben, „online“ gegangen werden.

Nachdem online gegangen wurde, sollte I/O OK grün leuchten (wenn blinkt, ist die Steuerung nicht mit dem Verstärker verbunden). Über das Steuerungsikon bei **RemProg** kann die Steuerung in den **Run Mode** versetzt werden. Der Hardware Schalter an der SPS muss hierfür auf REM umgeschaltet werden. REM bedeutet "remote" und erlaubt die Bedinung über die Steuerung.

Im Run Mode wird auf dem Verstärker Display b(Achsnummer) angezeigt.



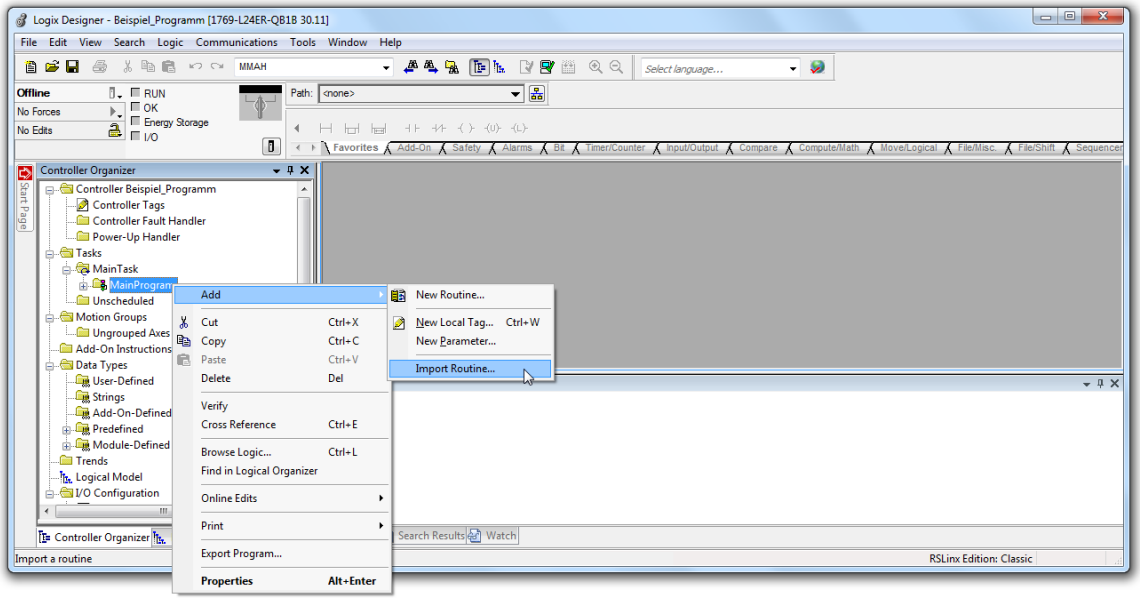


3 Programmierung

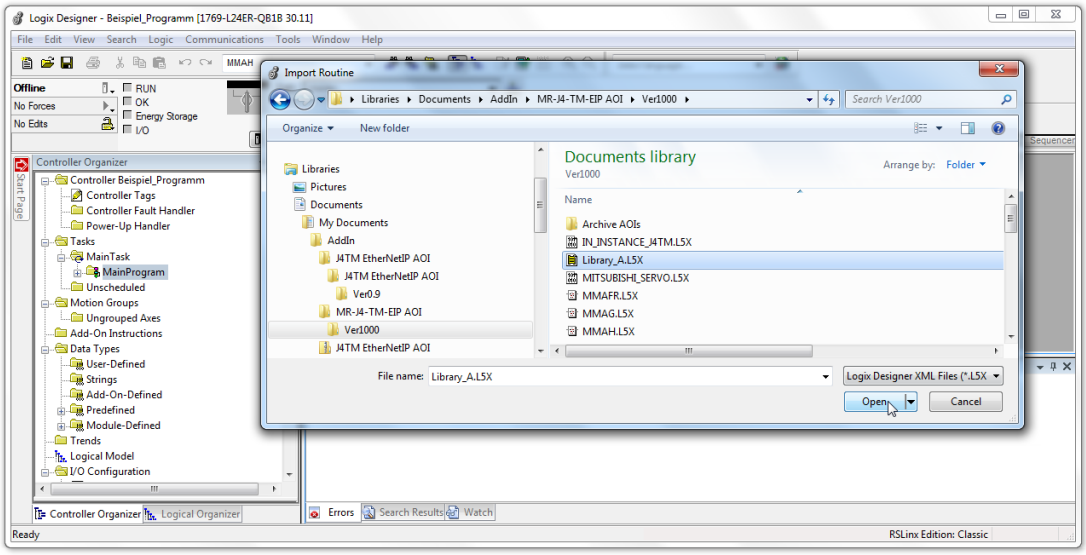
Gehen Sie immer offline, bevor Sie das Programm verändern. Für die Programmierung des Servo-Verstärkers können AOI's verwendet werden. Bei allem, was nicht direkt den Verstärker anspricht, wie Kopieranweisungen oder Jump Anweisungen, können die vorhanden Bausteine Verwendet werden.

3.1 AOI's importieren

Bevor es an die Programmierung geht, müssen spezielle AOI's importiert werden. Hierzu genügt es eine Routine zu importieren, die alle AOI's enthält.
Eine Routine kann mit Rechtsklick auf das **MainProgram** → **Add** → **Import Routine** importiert werden.

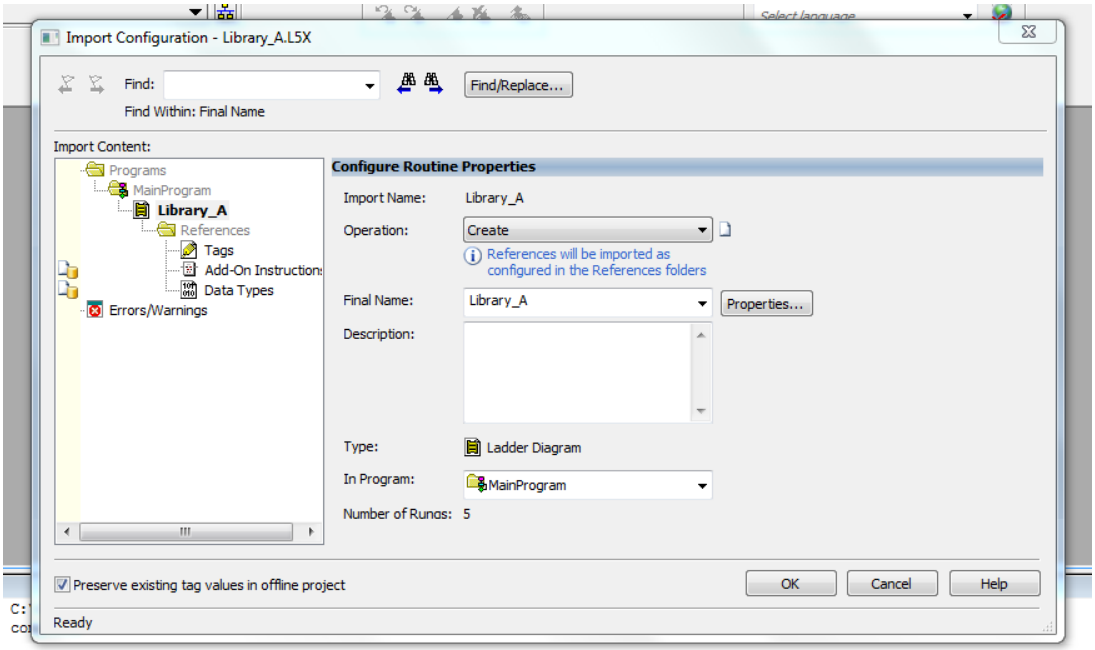


Es ist die **Library_A.L5X**[5] Datei zu importieren.



[5]Diese Datei ist im AOI Ordner zu finden. Siehe Kapitel 1.2.

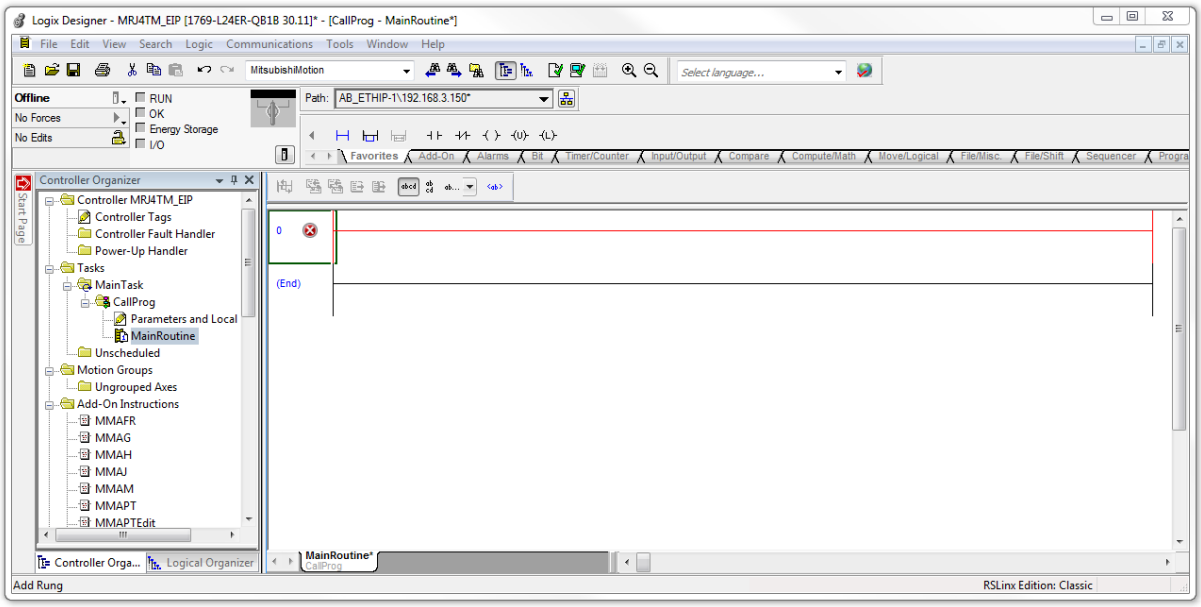
An der "Import Konfiguration" muss nicht zwingend etwas angepasst werden. Mit Ok wird der Importprozess fortgesetzt.



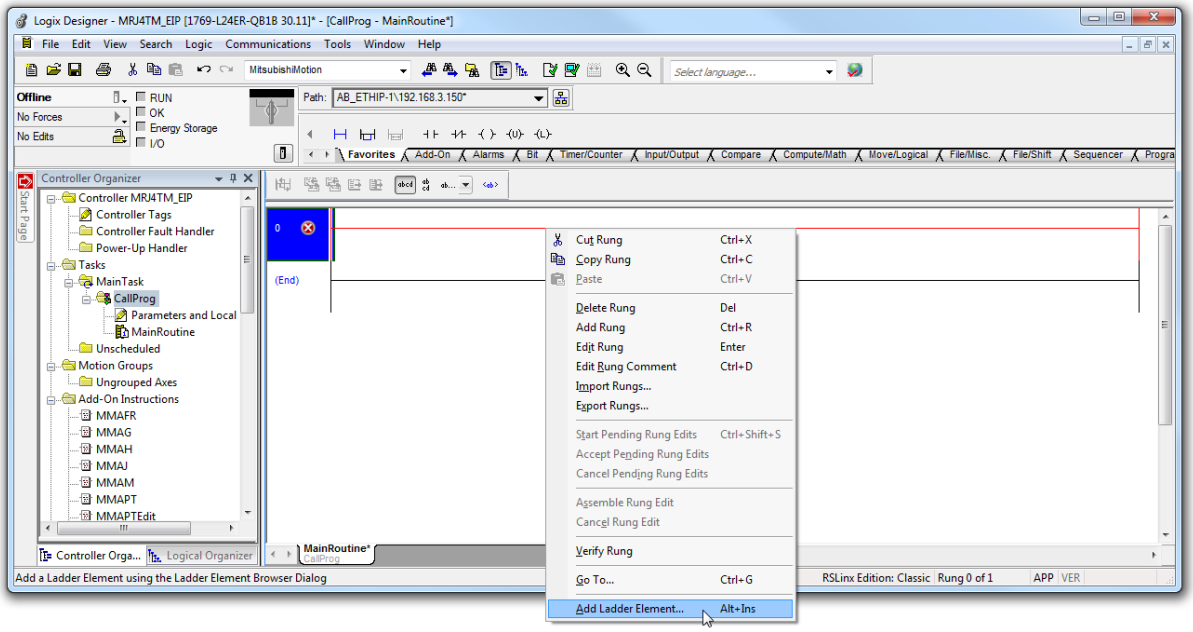
Wenn der Prozess abgeschlossen ist, werden eventuelle Fehler im Error Monitor angezeigt. Die durch das importieren der Routine erstellten Routinen könne nach dem Importieren wieder gelöscht werden.

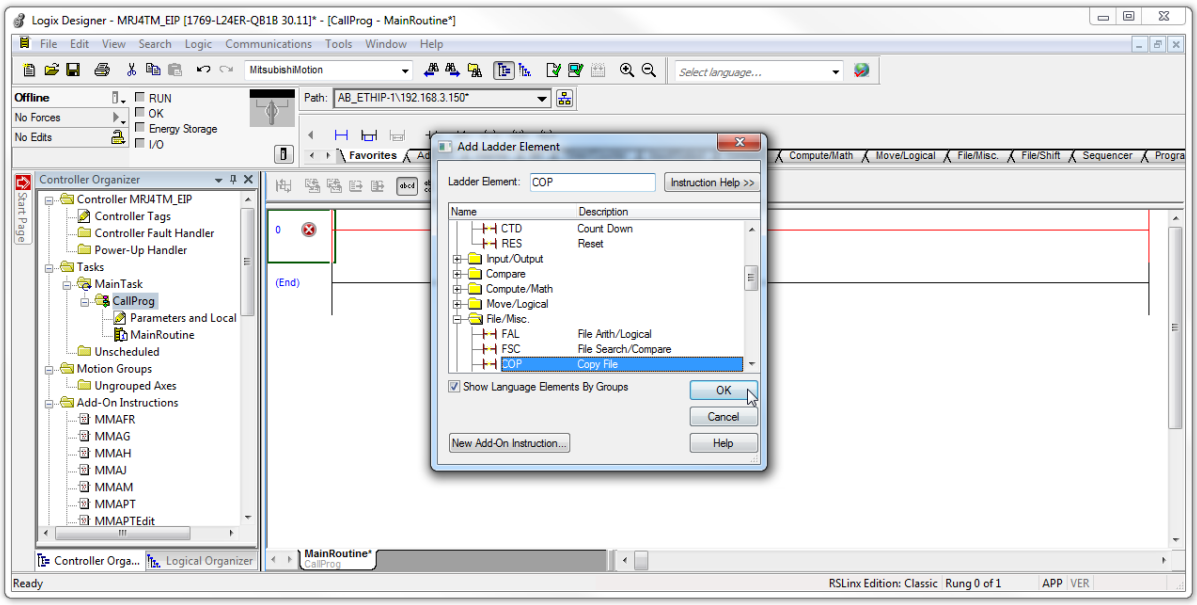
3.2 Main Routine

In der Main Routine werden alle Werte vom Verstärker importiert und am Ende wieder an den Verstärker exportiert. Zwischen diesen beiden Anweisungen werden die AOI's verwendet. Um die Übersicht zu behalten, bietet sich an mit Subroutinen zu arbeiten.

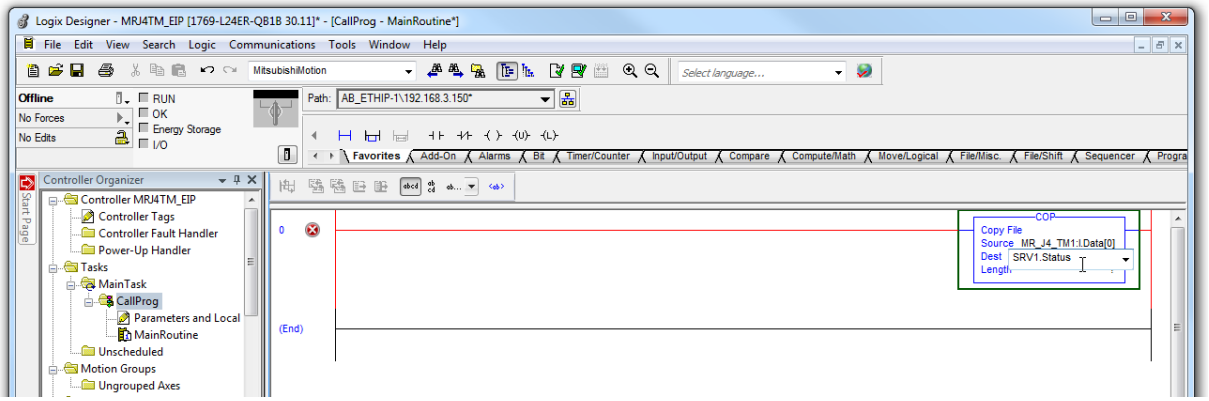


Mit einem Rechtsklick auf den Jeweiligen Strang, kann ein neues Ladder Element hinzugefügt werden. Für die Importanweisung wird ein COP(Copy) Baustein benötigt.

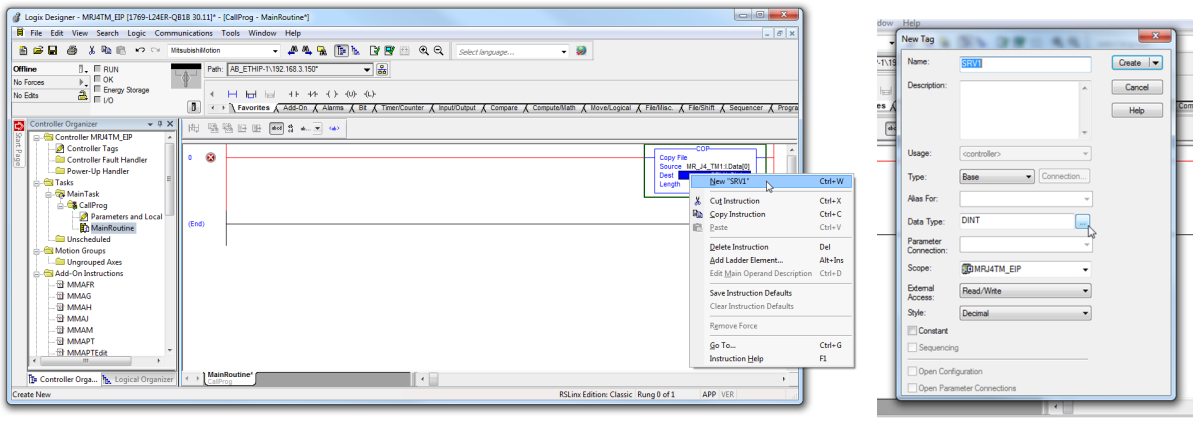


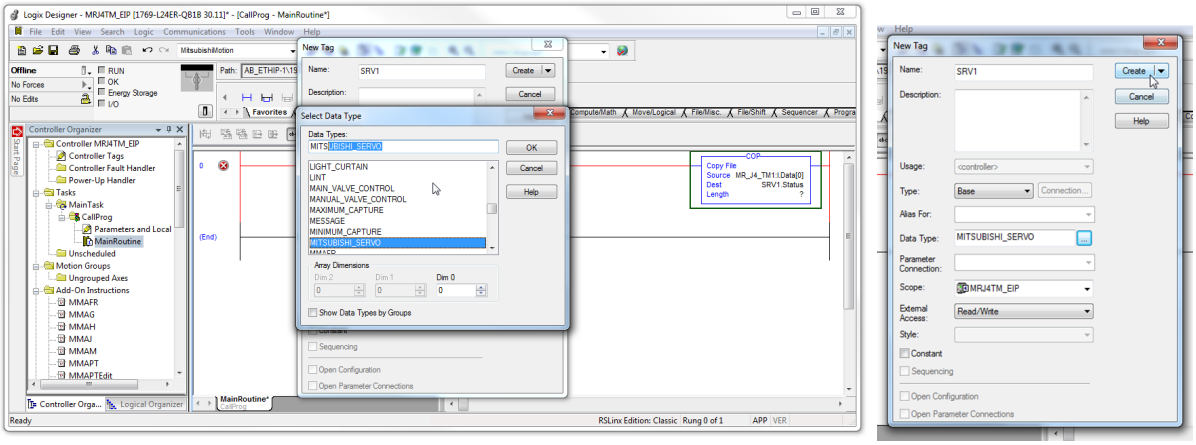


Der COP Baustein kopiert einen definierten Bereich der Source (Quell-Datei) in eine Destination (Ziel-Datei). Für eine ausführliche Beschreibung kann mit **F1 die Instruction Help geöffnet werden**. Als Source Datei muss hier **(Vergebener Verstärkername):I.Data[0]** verwendet werden. Als **Destination** muss eine **Mitsubishi Servoachse** erstellt werden. Das Format der Variable in Destination muss dem Format **beliebiger Achsname.Status** folgen. Beispielsweise SRV1.Status. Die **Länge** des zu kopierenden Teils ist **1**.

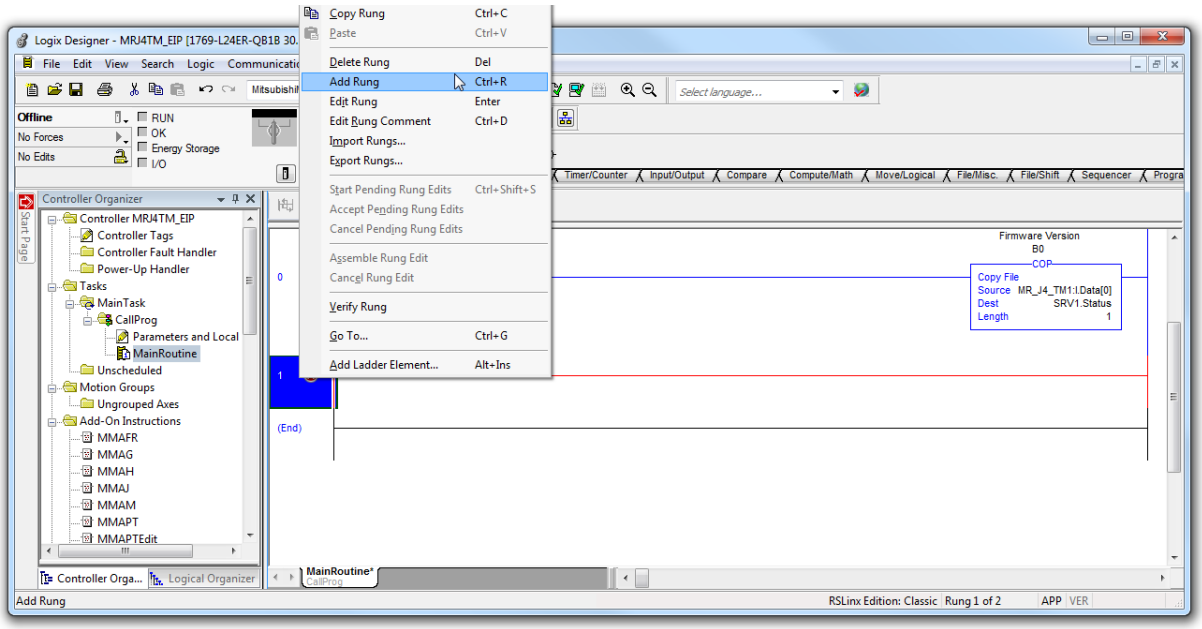


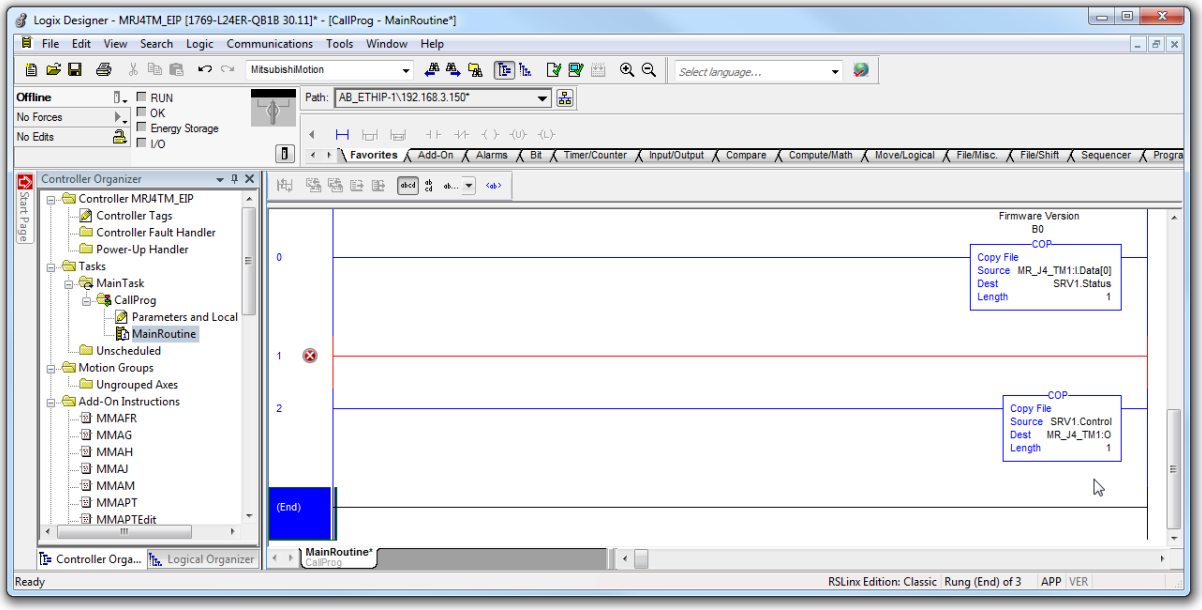
Mit Rechtsklick auf **Achsname.Status** → **New „SRV1“**, bzw. den vergebenden Achsnamen, kann die Achse angelegt werden. Als **Datentype muss MITSUBISHI_SERVO** ausgewählt werden. Wenn die Farbe des Strangs von Rot zu Blau wechselt, ist der Strang fehlerfrei.





Mit **Rechtsklick** → **Add Rung**, kann ein neuer Strang angelegt werden. Hier muss nun die Importanweisung angelegt werden. Es muss wieder ein COP Baustein verwendete werden. Als Source, wird diesmal **Achsname.Control** verwendet. Als Destination wird **(Vergebener Verstärkername):O** verwendet. Die **Länge** des zu kopierenden Teils ist **1**.



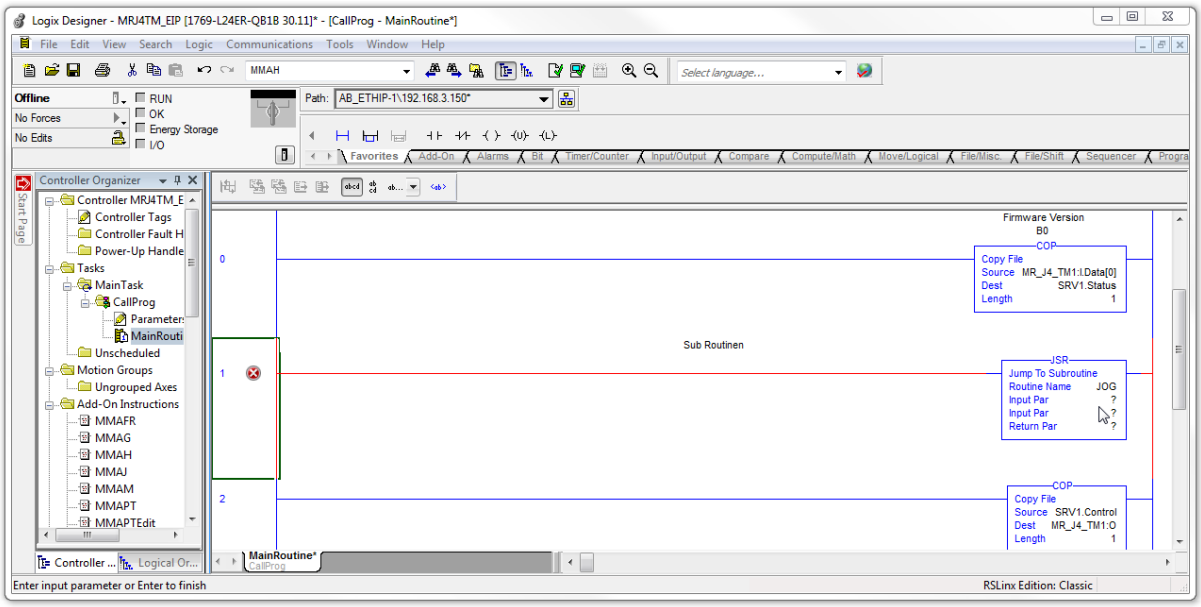


Die Subroutinen können nun auf dem Strang zwischen den beiden COP Bausteinen aufgerufen werden.

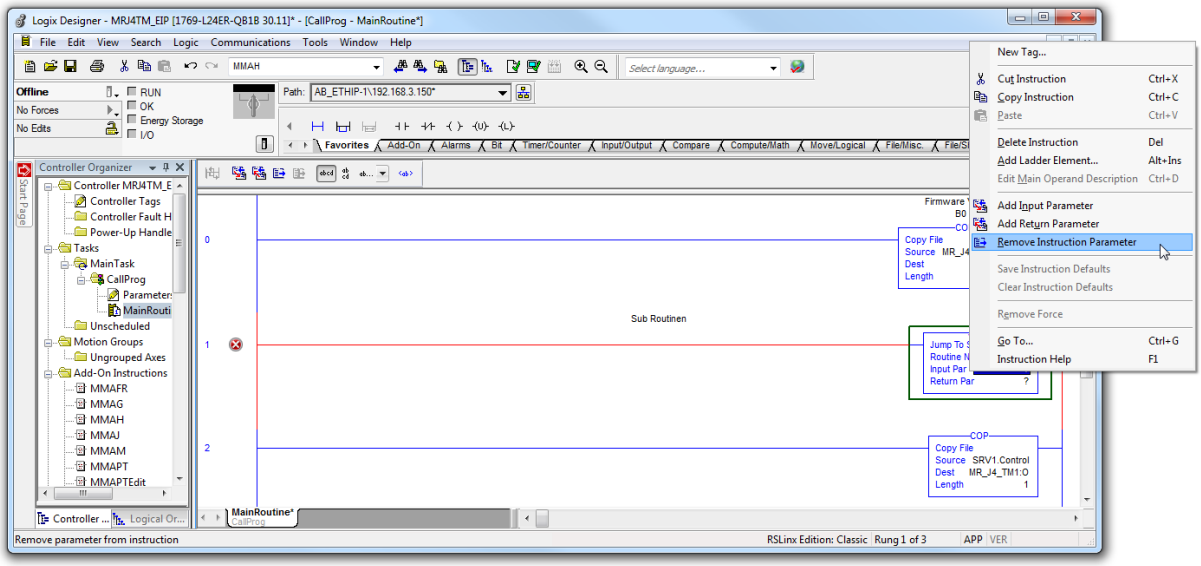
3.3 JOG Routine

Es ist zu empfehlen aus der Mainroutine in Unterroutinen zu springen. Hier werden Jog und Move in zwei verschieden Unterroutinen der Mainroutine realisiert.

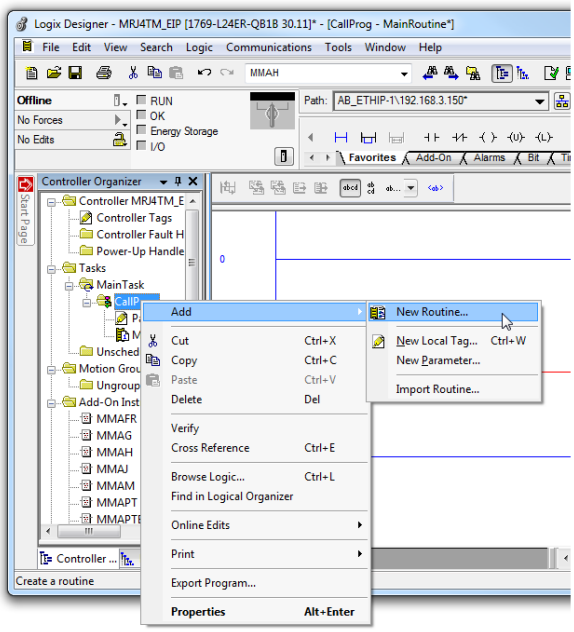
Um die Subroutine aufzurufen wird der Baustein **JSR** verwendet. Beim Springen in die Routine oder beim Zurückkehren aus der Routine, können Input und Return Parameter übergeben werden.

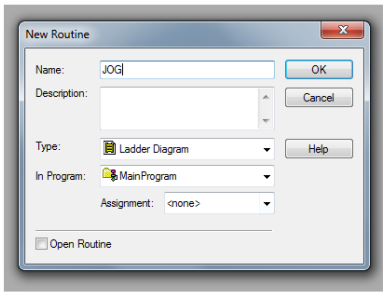


Wenn diese nicht benötigt werden, müssen sie mit **Rechtsklick auf die Parameter** → **Remove Instruction Parameter** entfernt werden.



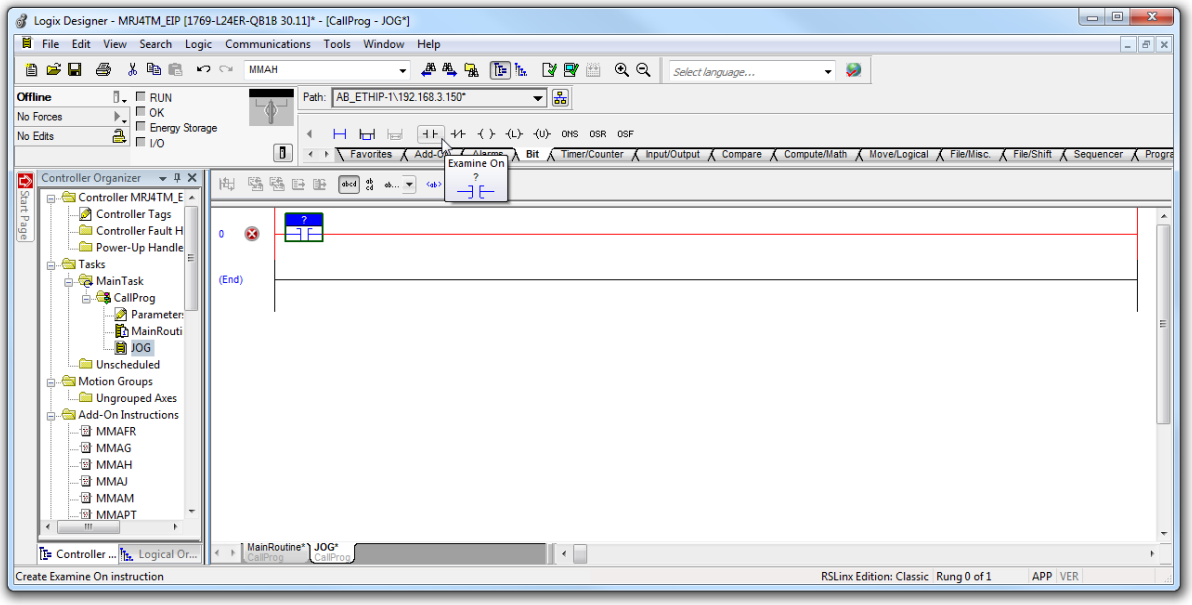
Eine neue Routine kann mit Rechtsklick auf den Überordner **CallProg** → **Add** →**New Routin...** erstellt werden.





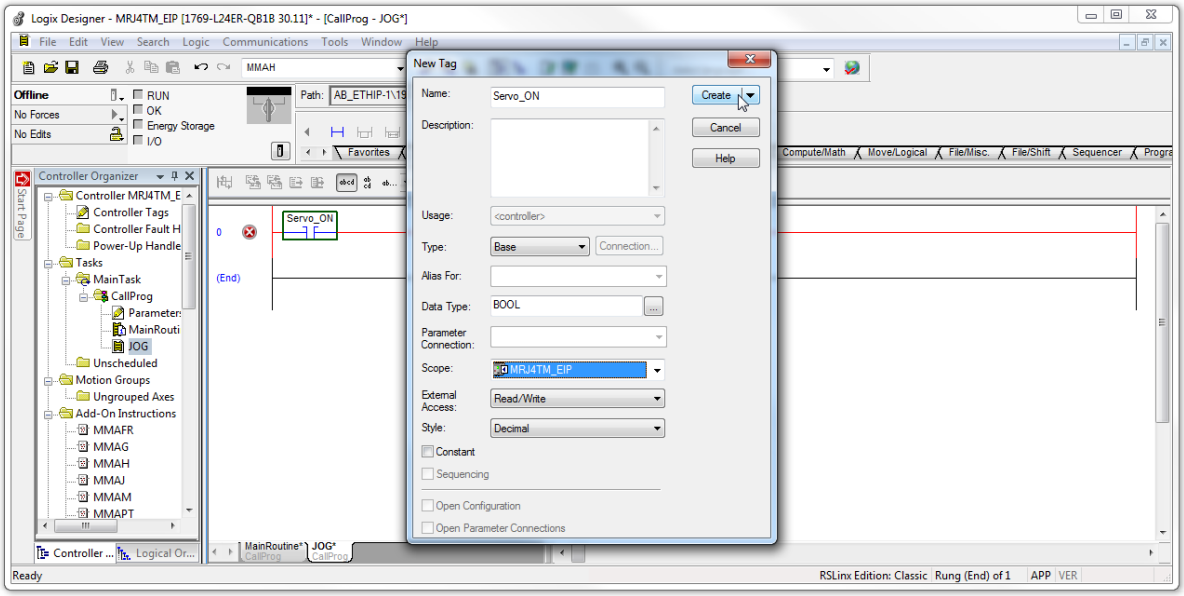
Nachdem die Unterroutine erstellt wurde, gibt es neben der Main-Routine "MainRoutine" eine Unterroutine JOG, welche aus der Main-Routine aufgerufen wird.

Bevor der Motor bewegt wird, muss er eingeschaltet, bzw. in Regelung gebracht werden. Hierzu kann über ein Bit der **AOI Baustein MMSO** aktiviert werden. (Anleitung zu allen AOI's[6]). Der Baustein zum Schalten des Bit's wird aus der Schnellauswahl an die gewünschte Position gezogen.

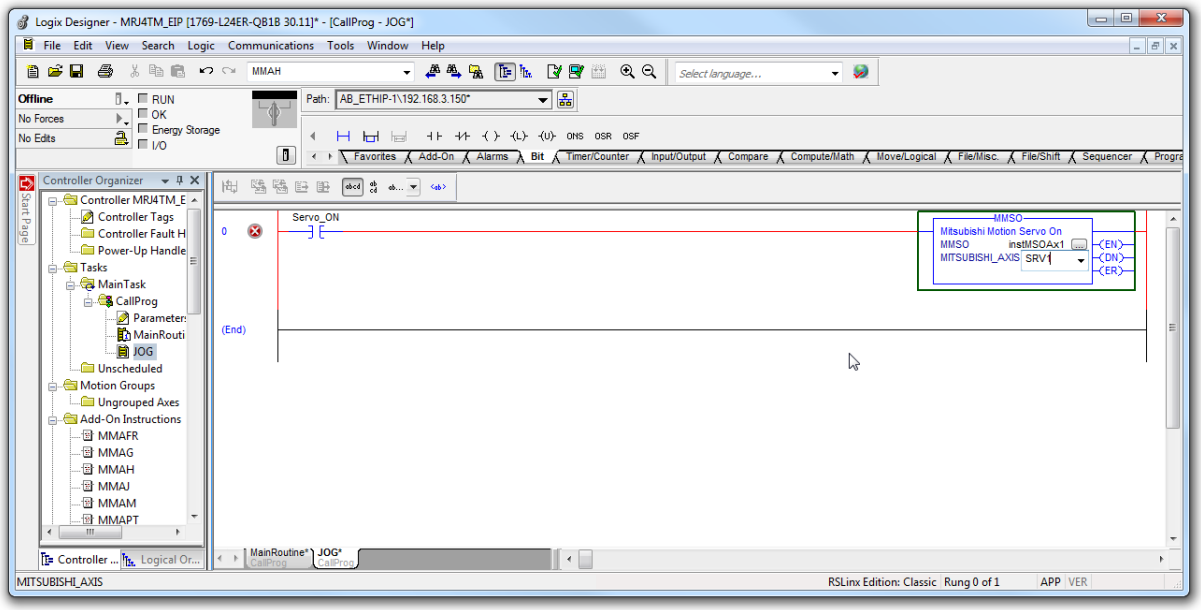


[6]Siehe Kapitel 1.2

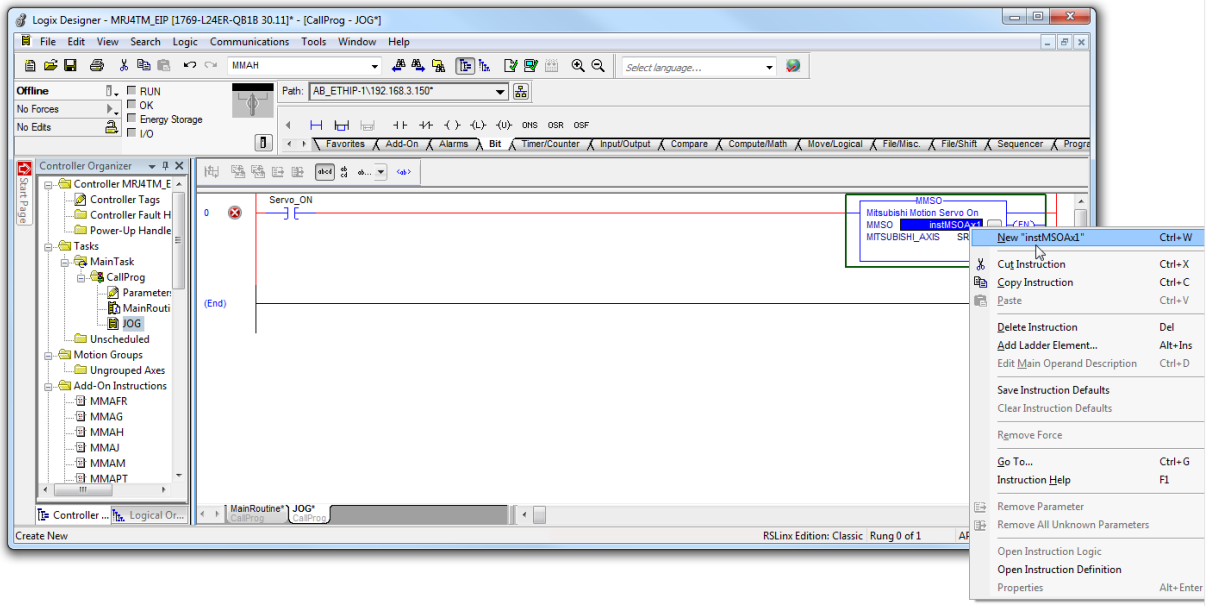
Per **Rechtsklick** → **NewTag** kann das Bit Beschriftet werden. Der Datentyp ist **BOOL**. Servo_On wird Sicherlich öfters gebraucht. Somit bietet es sich an dieses Bit als globale Variable zu definieren. Wo die Variable bekannt ist, kann unter **Scope** eingestellt werden. die Standarteinstellung ist Global.

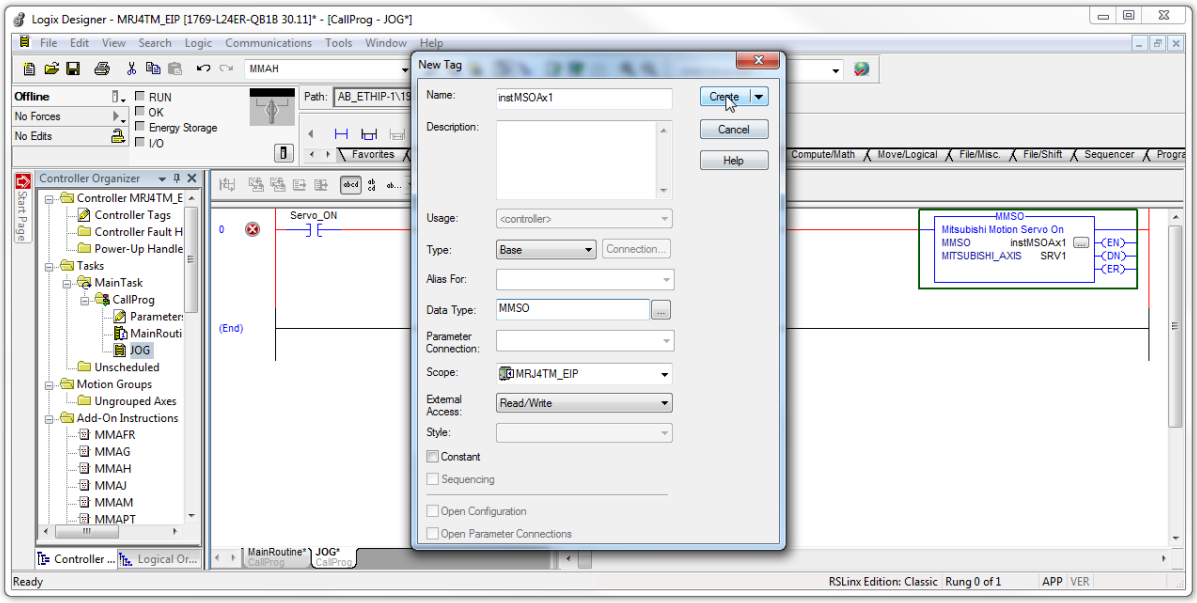


Das Bit **Servo_ON** soll den Baustein **MMSO (Mitsubishi Servo On)** aktivieren. Im Baustein müssen Variablen belegt werden. Es ist zu empfehlen, ein Format bei der Benennung der Variablen beizubehalten.



Ein neuer Baustein muss **immer mit einer Variablen verknüpft werden**. Mit Rechtsklick auf den neue Baustein → **New "VERGEBENER NAME"** kann die Variable erstellt werden. Als **Datentyp muss der Name des Bausteins** verwendet werden.



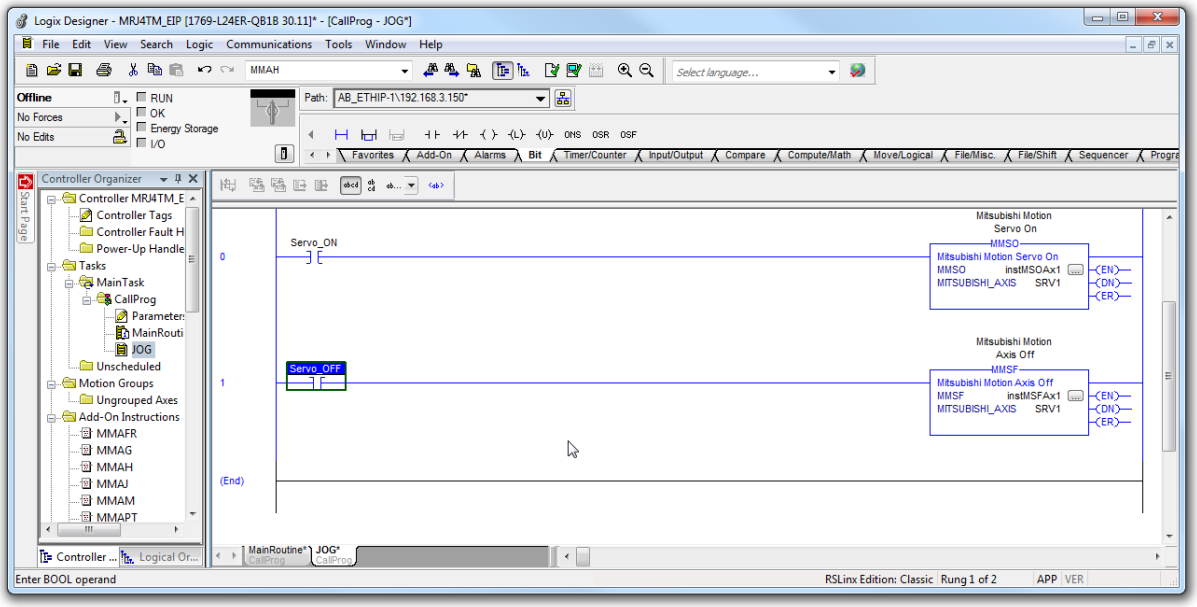


Die Variable im Parameter **MITSUBSIHI_AXIS** muss hier **SRV1** (oder wie die selbst erstellte Achse) genannt werden. Diese Variable wurde auch schon bei den beiden COP Bausteinen der Mainroutine verwendet.

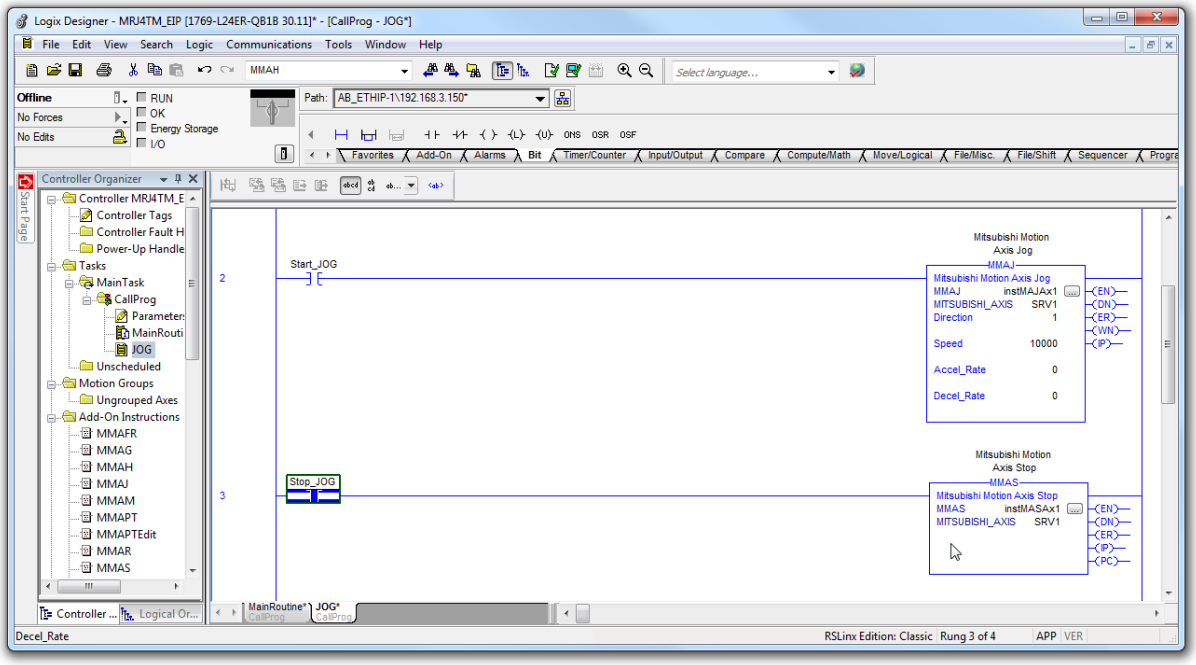
Good to know:
Was ist bei der Erstellung der Servo Achse SRV1 nun passiert?

Status und Control sind unter Register von SRV1. Bei der Erstellung der Achse SRV1, mit dem Datentype MITSUBISHI_SERVO, wurden festgelegt, welche Unterregister die Variable SRV1 enthält. Mit „Name.Unterregister" können diese angesprochen werden. Mit „Name" werden alle unter Register angesprochen. Hier sucht sich der Funktionsbaustein selbstständig die richtigen Register.

Es wird auch eine Anweisung zum Ausschalten des Servo Motors benötigt. Dafür wird der **Baustein MMSF** verwendet.



Der Baustein MMAJ regelt den Motor auf eine Konstante Geschwindigkeit und hält diese Geschwindigkeit, bis der Jogprozess unterbrochen wird. Beispielsweise durch **MMAS „Axis Stop“**.



Für eine Ausführliche Beschreibung kann mit **F1 die Hilfe zu der Bausteine geöffnet** werden.

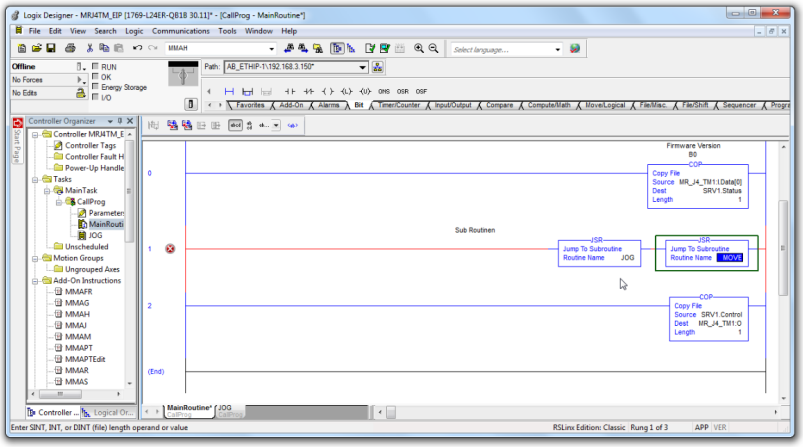
Um das erstellte Programm zu testen, muss das Projekt, wie beim Verbindungstest, in die Steuerung geladen werden. Im „Run Mode“ kann, mit **Rechtsklick → Toggle Bit oder Ctrl+T**, ein Bit geschaltet werden. Der Servo muss vor dem Joggen mit MMSO (Mitsubishi Motion Servo On) in Regelung gebracht werden.

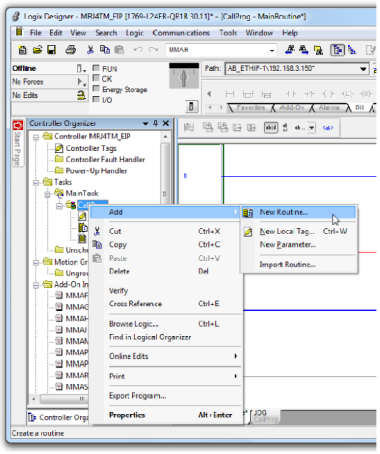
Bitte Treffen Sie die Nötigen Sicherheitsvorkehrungen

3.4 Move Routine

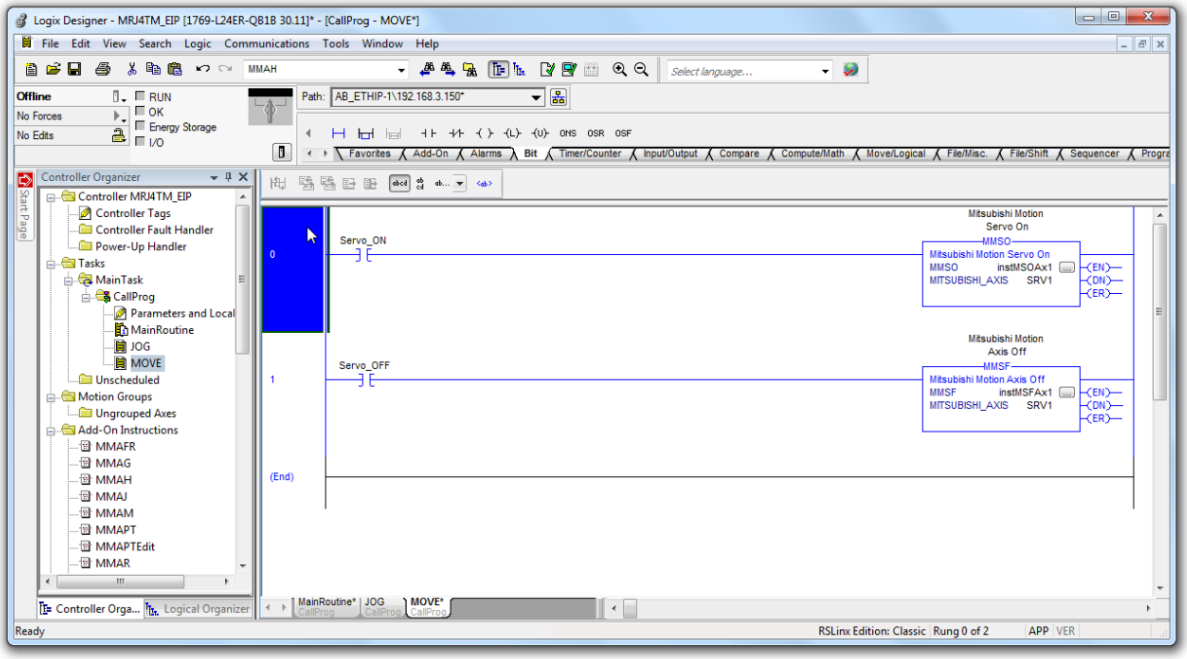
Die **MOVE Routine** baut auf der **JOG Routine** und dem **erstellen einer Unterroutine** auf. Es ist zu empfehlen sich vor dem Bearbeiten dieses Kapitels, mit dem Kapitel Jog zu beschäftigen.

Es ist zu empfehlen aus der Mainroutine in Unterroutinen zu springen. Hier werden Jog und Move in zwei verschieden Unterroutinen der Mainroutine realisiert.

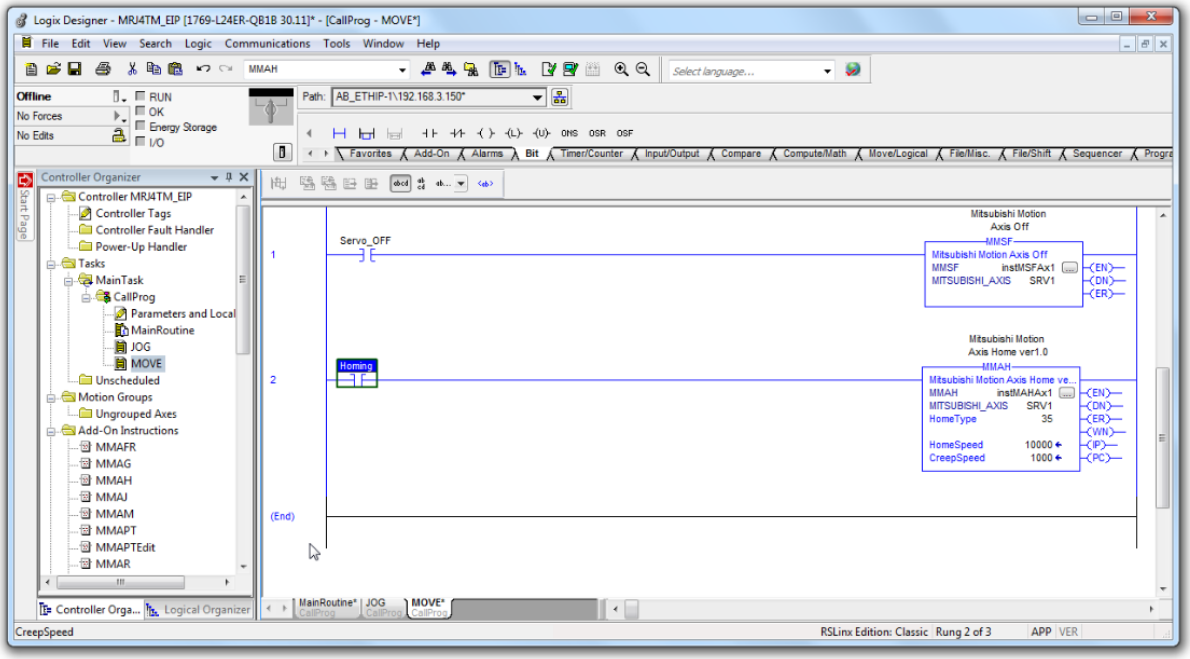




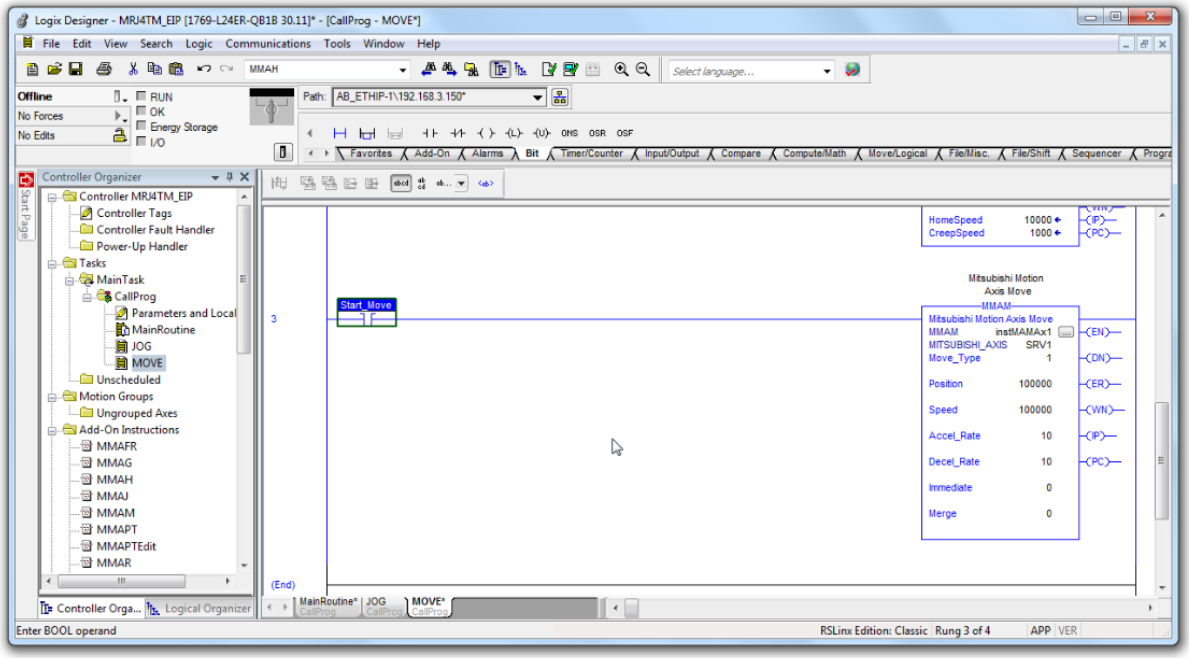
Es besteht die Möglichkeit, Stränge zu kopieren und wieder einzufügen. Da die Variablen, die bisher erstellt wurden, als globale Variablen angelegt wurden, bietet es sich hier an, einfach dieselben Variablen zu verwenden. Die Variablen erfüllen hier dieselbe Aufgabe, wie in der JOG-Unterroutine.



Für eine Bewegung zu einem gezielten Punkt, benötigt das System eine Referenzposition. Für die Referenzierung gibt es den **MMAH (Mitsubishi Axis Home)** Baustein. Dieser Baustein kann auf verschieden Referenzeiermethoden parametriert werden. Die verschiedene Methoden sind im **MR-J4-_TM_Handbuch (EtherNet/IP)**[7] beschrieben. Weite Erklärungen zu diesem Baustein sind in der Hilfe und im **AOI Handbuch** zu finden. Hier wird der **Home-Typ 35** verwendet, welcher die aktuelle Position als Referenzpunkt speichert.

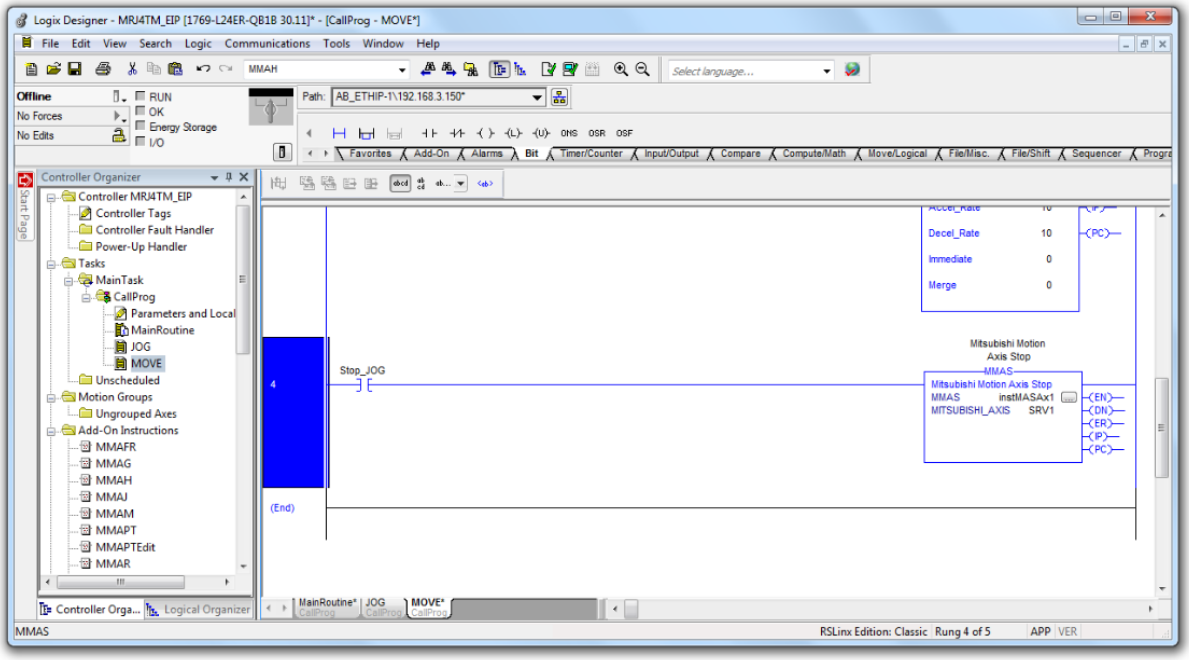


Mit dem **MMAM (Mitsubishi Motion Axis Move)** Baustein kann mit einer 0 in Move_Type absolut und mit einer 1 in Move_Type inkrementell verfahren werden. Weitere Informationen sind in der Hilfe(F1), oder im **AOI Handbuch** zu finden.



[7] Siehe Kapitel 1.2

Zum Stoppen der Achse kann die Stoppanweisung aus der JOG Unterroutine kopiert werden.

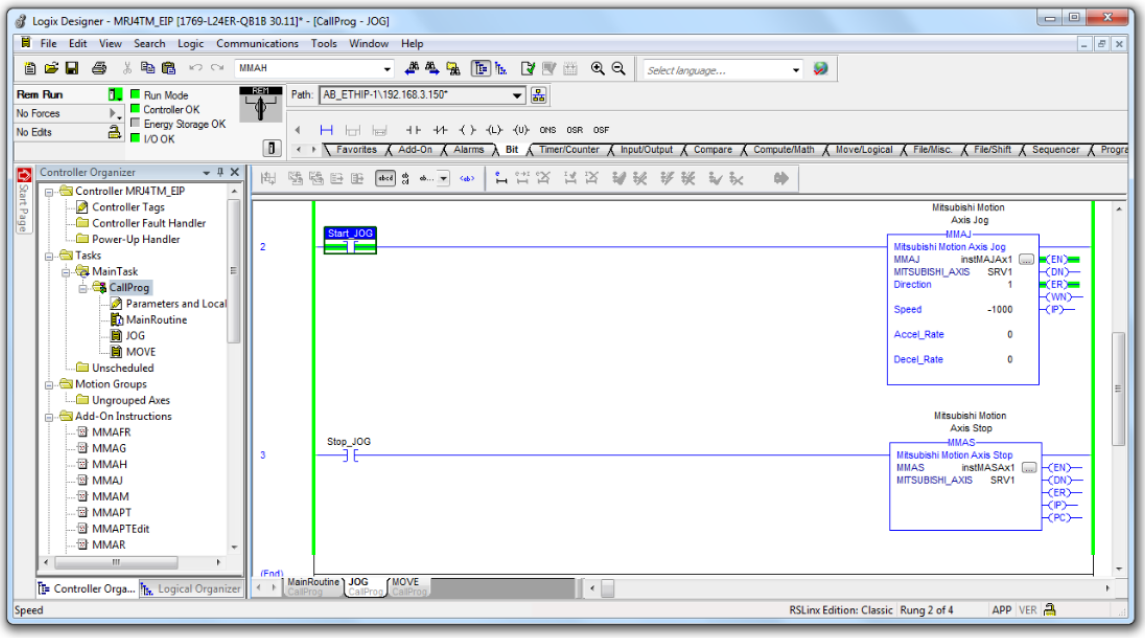


4 Troubleshooting

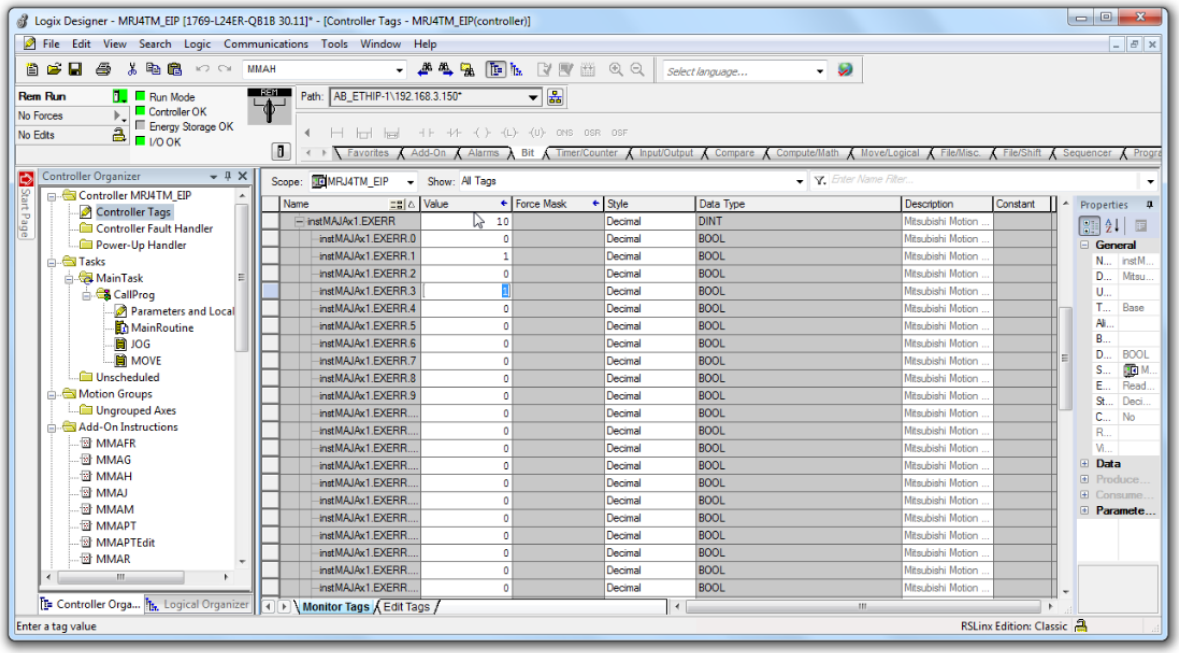
Um Fehler im Programm zu beheben, ist es notwendig die jeweilige Fehlermeldung zu interpretieren.

4.1 Fehlerbehebung im Programm

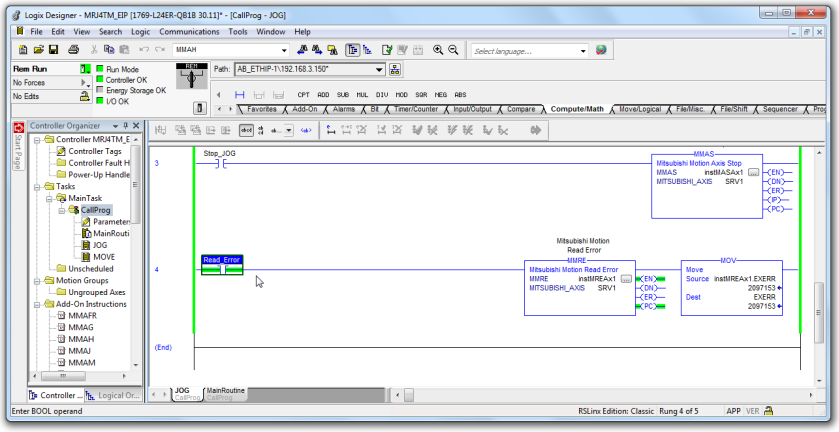
Wie im Bild zu sehen, ist beim Ausführen des MMAJ Bausteins ein Fehler Aufgetreten. Wenn der Ausgang **ER** markiert ist, liegt eine Fehlermeldung an.

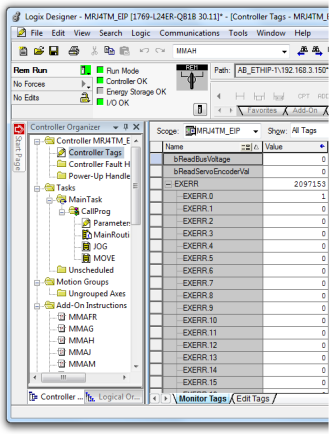


Bausteine mit einem **ER** Output besitzen ein eigenes **EXERR Register**, indem die Fehlermeldung ablesbar ist. Das EXERR Register ist bei Globalen Variablen unter **Controller Tags** zu finden.



Ein Error des Verstärkers kann über den Baustein **MMRE (Mitsubishi Motion Read Error)** gelesen werden. Allerdings lässt sich die Fehlernummer kaum interpretieren.



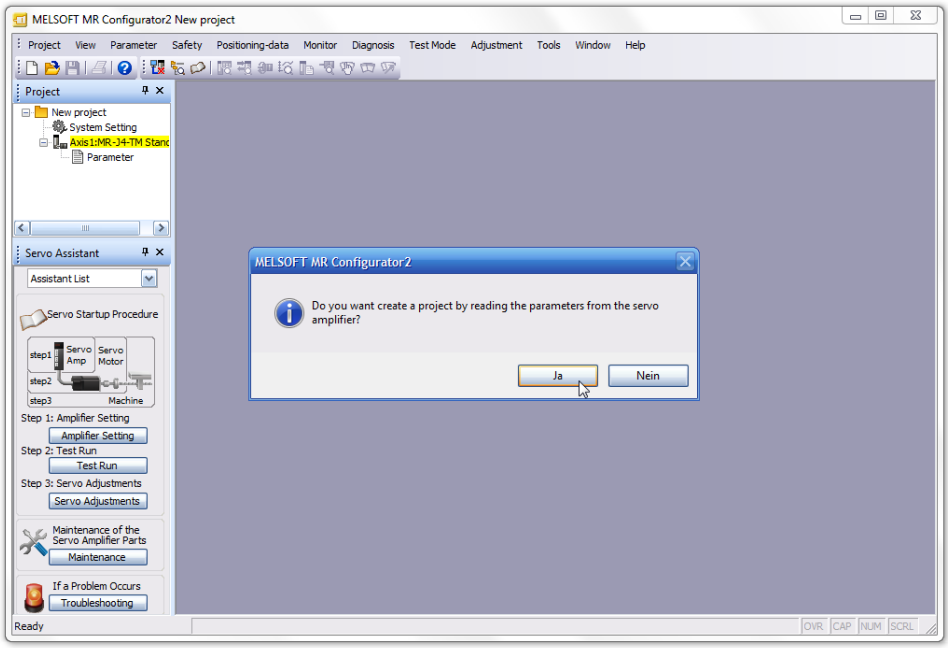


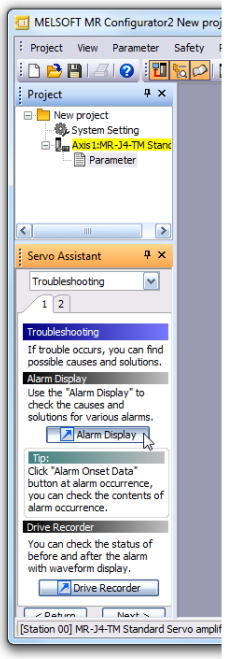
Bei einem anliegenden Verstärker Errors, sollte der **MR Configurator 2** verwendet werden. Über den **Axis Monitor**, kann die Fehlernummer abgelesen und eine Interpretation des Fehlers angezeigt werden. Hierzu muss sich über USB mit dem Verstärker verbunden werden.

4.2 Fehlerbehebung am Verstärker

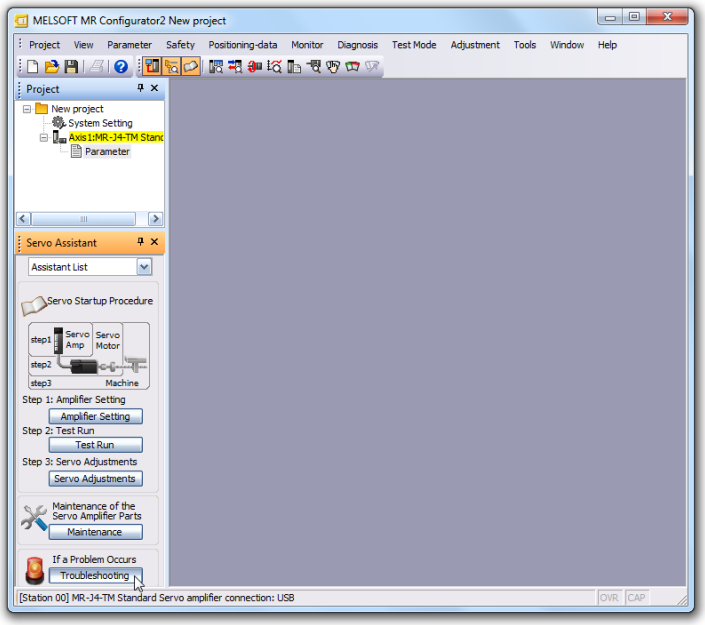
Um den Angezeigten Fehlercode bei einem Verstärker Error zu interpretieren, sollte der **MR Configurator 2** verwendet werden.

Der Verstärker muss mit einem USB 2.0 Mini-B Kabel (beidseitigen Ferritkernen) an einen PC angeschlossen werden. Nach dem Start des MR Configurators, können die Einstellungen des angeschlossenen Verstärkers importiert wenden. Mit **Ja** wird ein neues Projekt erstellt, indem der angeschlossene Verstärker implementiert ist.

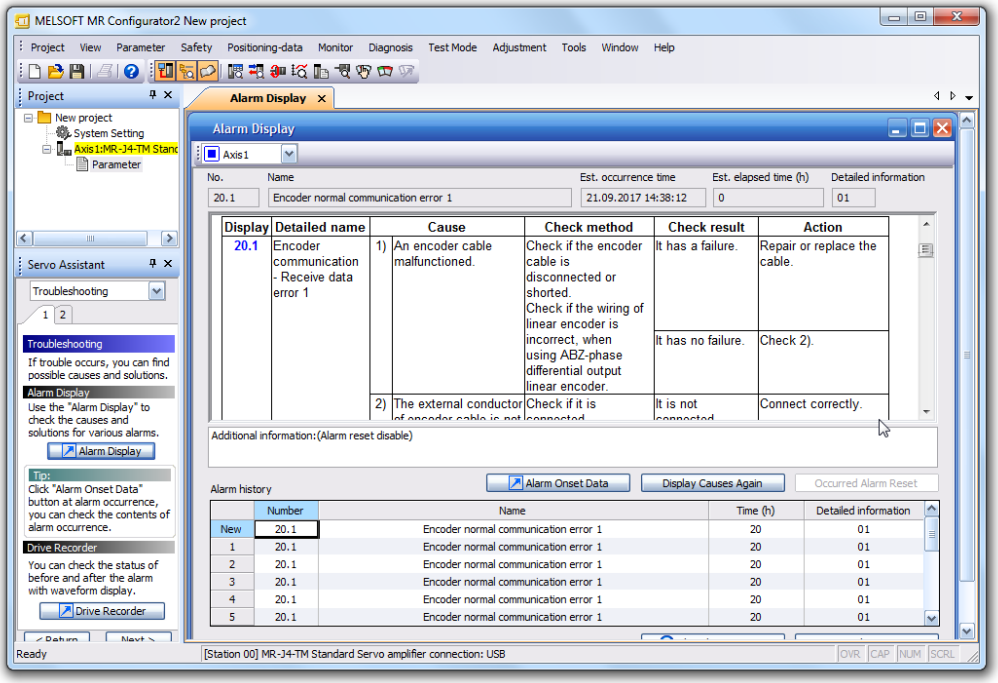




Zum einsehen des Alarmdisplay, kann im Servo Assistent Menü der Punkt **Troubleshooting** → **Alarm Display** ausgewählt werden. Das Alarm Display kann auch über **Diagnosis** → **Alarm Display** gestartet werden.



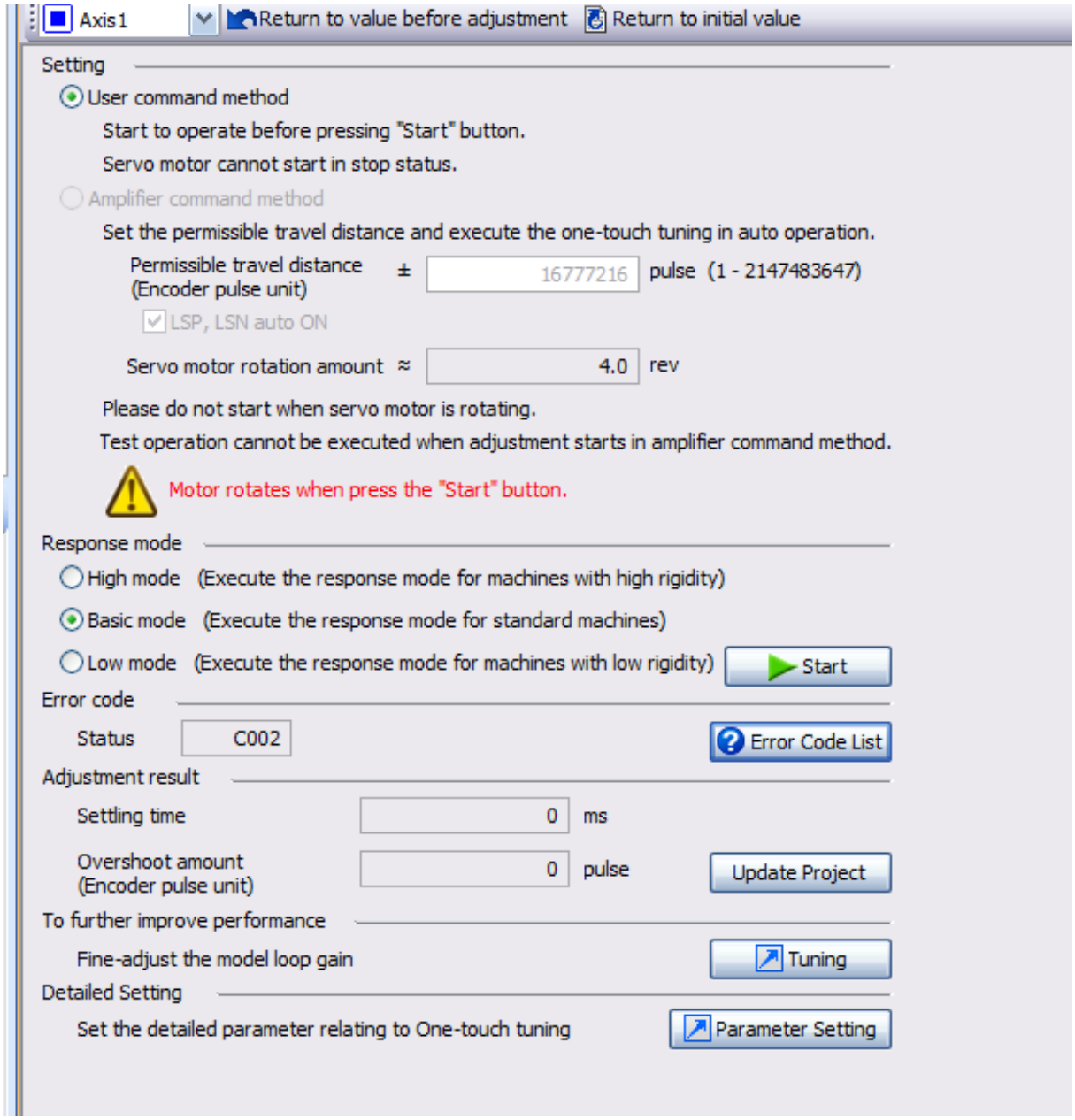
Im Alarm Display werden aktuell anliegende Fehler angezeigt, sowie eine **Alarm history**. Hier ist nun abzulesen, was die Fehlernummer bedeutet und was zu machen ist, um den Fehler zu beheben.



5 Tuning Möglichkeiten

5.1 One touch tuning

Es ist möglich den Verstärker zu tunen um bessere Ergebnisse zu erlangen. Dazu kann die One-touch Tuning Funktion genutzt werden, welche unter **Adjustment** → **One-touch Tuning** erreicht wird. Im Gegensatz zum „Auto tuning“, welches später beschrieben wir, ist das **One touch tuning** eine abgeschlossene Prozedur, ermittelte Parameter werden im Anschluss „eingefrohren“.



Mit einem Klick auf **Start** wird das Tuning aktiviert. Der bei der Anwendung gewöhnliche Verfahr Zyklus muss anschließend so lange ausgeführt werden, bis der Fortschrittsbalken 100% erreicht und das Tuning beendet ist. Das Tuning kann einige Minuten in Anspruch nehmen. Erhöhte Geräuschentwicklung durch Schwingungen ist hierbei normal. Mit einem Klick auf **Update Project** bzw. auf das erschienene Fenster werden die Tuningparameter übertragen.
Tritt während des Tuning Vorgangs ein Fehler auf, ist dieser im **Error Code Fenster** zu sehen.
(C002 bedeutet hier, der „Servo On“ Befehl wird benötigt, bevor das Tuning ausgeführt werden kann). Die entsprechende Fehlerbeschreibung ist unter **Error Code List** aufgeführt.

5.2 Andere tuning Möglichkeiten

Eine weitere Möglichkeit ist, unter **Adjustment** → **Tuning** manuell in den Tuning Prozess einzugreifen. Die wichtigsten Modi sind:

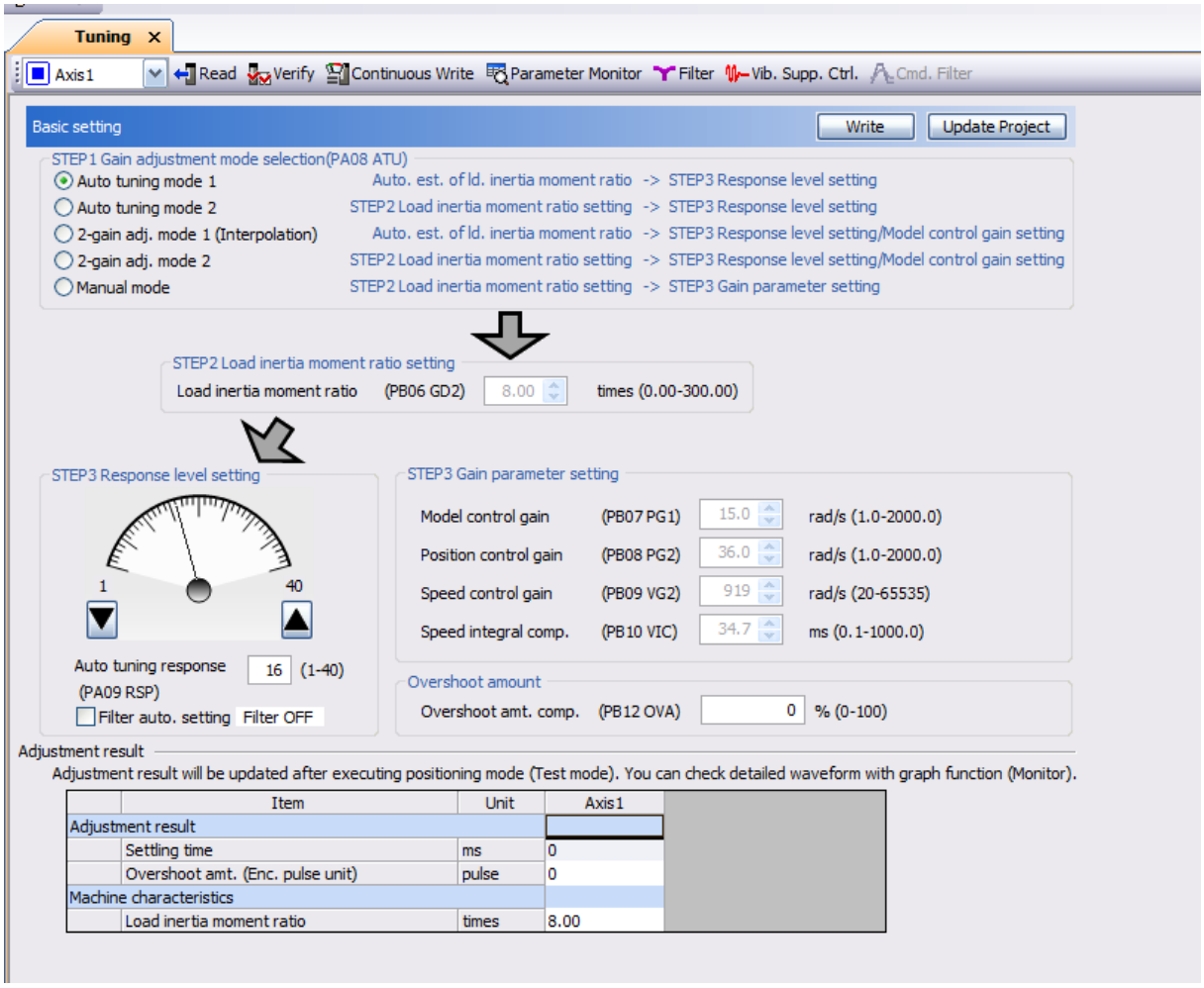
Auto tuning mode 1
Massenträgheitsverhältnis wird automatisch ermittelt

Auto tuning mode 2
Massenträgheitsverhältnis wird manuell eingetragen

Manual mode
Alle Tuning Parameter werden von Hand eingetragen

Mit **STEP3 Response level setting** kann die „Härte" des Motors eingestellt werden.

1 ist sehr weich (Welle lässt sich von Hand bewegen, bevor eine Gegenreaktion kommt) 40 ist extrem hart (Gegenreaktion kommt sofort)

Bitte hier möglichst in 1er Schritten verfahren und das Testergebnis beobachten.

Um die Tuningoptionen zu übernehmen kann mit **Continuous Write** permanent jede Änderung sofort auf den Verstärker gespielt werden oder mit **Write** einmalig die Veränderungen übernommen werden.