

# PLC500 MOTION CONTROLLER

## PLC500MC

### Nota de Aplicação



# Nota de Aplicação

**PLC500MC**

Documento: 10010339510

Revisão: 00

Data de publicação: 05/2023

## SUMÁRIO DAS REVISÕES

---

A informação abaixo descreve as revisões ocorridas neste manual.

Versão	Revisão	Descrição
-	R00	Primeira edição.

<b>1</b>	<b>INTRODUÇÃO</b>	<b>1-1</b>
1.1	ABREVIACÕES E DEFINIÇÕES	1-1
1.2	SOBRE O PLC500MC	1-2
1.3	TECNOLOGIA ETHERCAT	1-3
1.3.1	Interfaces EtherCAT PLC500MC	1-3
1.3.2	Abrangência EtherCAT	1-3
1.4	CONTROLE DE MOVIMENTO	1-4
1.4.1	Editor came	1-5
1.4.2	Editor CNC 3D	1-5
1.4.3	Abrangência das bibliotecas SoftMotion + CNC Robotic	1-6
<b>2</b>	<b>CRIAR E CONFIGURAR REDE ETHERCAT + SOFTMOTION</b>	<b>2-1</b>
2.1	COMPONENTES UTILIZADOS	2-1
2.2	ARQUITETURA DE REDE	2-1
2.3	CONFIGURAÇÃO DO SERVOCONVERSOR SCA06	2-1
2.4	CRIAR UM PROJETO NO CODESYS	2-2
2.4.1	Adicionar EtherCAT Master SoftMotion	2-3
2.4.2	Adicionar SCA06_SoftMotion como escravo na rede EtherCAT	2-3
2.4.3	Configurar EtherCAT Master SoftMotion	2-4
2.4.4	Configurar SCA06_SoftMotion	2-5
2.4.5	Configurar SM_Drive_ETC_WEG_SCA	2-5
2.5	MONITORAÇÃO	2-7
2.5.1	Estado da comunicação EtherCAT	2-7
2.5.2	Verificar variação na posição atual do servomotor	2-7
2.6	COMISSIONAMENTO	2-8
<b>3</b>	<b>APLICAÇÃO SOFTMOTION</b>	<b>3-1</b>
3.1	CRIAR APLICAÇÃO	3-1
3.2	CRIAR VISUALIZAÇÃO	3-3
<b>4</b>	<b>INFORMAÇÕES ADICIONAIS DA REDE ETHERCAT</b>	<b>4-1</b>
4.1	ATRIBUIR UM ENDEREÇO ESTÁTICO PARA O SCA06 NA REDE ETHERCAT	4-1
4.2	LER E EDITAR PARÂMETROS NO SCA06 PELA REDE ETHERCAT	4-3
4.3	EDITAR PDOS NA REDE ETHERCAT	4-5
4.4	CONFIGURAR REDUNDÂNCIA ETHERCAT	4-6
4.5	ARQUIVO XML	4-7
<b>5</b>	<b>INFORMAÇÕES ADICIONAIS SOFTMOTION</b>	<b>5-1</b>
5.1	PRIORIDADE DE TAREFAS	5-1
5.2	CONFIGURAÇÕES DE ESCALA PARA O SM_DRIVE_ETC_WEG_SCA	5-1
5.2.1	Motor Type: Rotary	5-1
5.2.2	Motor Type: Linear	5-2
5.3	ADICIONAR EIXO VIRTUAL	5-3
5.4	ADICIONAR EIXO ENCODER	5-4
5.5	SINCRONIZAÇÃO CAME	5-8
5.5.1	Criar aplicação came	5-8
5.5.2	Importar tabela came	5-9
5.5.3	Executar tabela came	5-10
5.6	INTERPRETAR E EXECUTAR ARQUIVOS CNC	5-13
5.6.1	Abrangência dos comandos (G-Code) suportados	5-13
5.6.2	Criar aplicação CNC	5-13
5.6.3	Importar arquivos CNC	5-16
5.6.4	Executar caminho CNC	5-17
5.6.5	Eixo tangencial em caminhos CNC	5-20
5.7	ALTERAR MODO DE CONTROLE	5-22



<b>6</b>	<b>CRIAR E CONFIGURAR REDE CAN + SOFTMOTION</b> .....	<b>6-1</b>
6.1	CONFIGURAÇÃO DO SERVOCONVERSOR SCA06 CAN .....	6-1
6.2	CRIAR UM PROJETO NO CODESYS .....	6-1
6.2.1	Adicionar CANopen Manager SoftMotion .....	6-2
6.2.2	Adicionar SCA06 como escravo na rede CANopen .....	6-2
6.2.3	Configurar objeto CAN .....	6-4
6.2.4	Configurar objeto CANopen Manager SoftMotion .....	6-5
6.2.5	Configurar SCA06 como escravo SoftMotion CAN .....	6-5
6.2.6	Configurar SM_Drive_GenericDSP402 .....	6-7
6.3	MONITORAÇÃO .....	6-7
6.3.1	Estado da comunicação CAN .....	6-7
6.3.2	Verificar variação na posição atual do servomotor .....	6-8
6.4	COMISSIONAMENTO .....	-9
<b>A</b>	<b>APLICAÇÃO CAME</b> .....	<b>A-1</b>
<b>B</b>	<b>APLICAÇÃO CNC</b> .....	<b>B-1</b>
<b>C</b>	<b>APLICAÇÃO CNC TANGENCIAL</b> .....	<b>C-1</b>

# 1 INTRODUÇÃO

Esta Nota de Aplicação apresenta as principais características e informações necessárias para a configuração e utilização do PLC500MC juntamente com o servoconversor SCA06.

Para o controle de movimento, é essencial a correta configuração da rede e dos dispositivos envolvidos. Por favor, siga as etapas descritas neste documento para uma configuração apropriada.

Para mais informações a respeito do hardware do produto, interfaces e protocolos de comunicação, consulte o Manual do Usuário do PLC500, disponível em <http://www.weg.net>.

## 1.1 ABREVIATÓES E DEFINIÇÕES

**CNC:** Comando Numérico Computadorizado, é um método que controla os movimentos de máquinas pela interpretação direta de instruções codificadas na forma de números e letras.

**Codesys:** Plataforma de programação que permite desenvolver, configurar e monitorar soluções para automação industrial e integração de sistemas.

**CoE:** CANopen sobre EtherCAT (CANopen over EtherCAT).

**EDS:** Arquivo de configuração que contém informações sobre os objetos, serviços e configurações de um escravo de rede.

**EEPROM:** Memória Somente de Leitura Programável Apagável Eletronicamente (Electrically-Erasable Programmable Read-Only Memory)

**Ethernet:** Arquitetura de interconexão para redes locais (IEEE 802.3).

**EtherCAT:** Tecnologia para comunicação de tempo real baseada em Ethernet (Ethernet for Control Automation Technology).

**FB:** Bloco de função.

**MC:** Controlador de movimento.

**PDO:** Dados de processo.

**PLC:** Controlador lógico programável.

**PLCopen:** Organização que promove o controle industrial com base na norma IEC61131-3.

**POU:** Unidade de organização do programa (Program Organization Unit).

**SCA06:** Servoconversor WEG - SCA06.

**SoftMotion:** Controle suave de movimento.

**SDO:** Dados de serviço.

**u:** Unidade de aplicação.

**XML:** Arquivo de configuração que contém informações sobre os objetos, serviços e configurações de um escravo EtherCAT.

# INTRODUÇÃO

## 1.2 SOBRE O PLC500MC

O PLC500 Motion Controller (PLC500MC) é um Controlador Lógico Programável com funcionalidades SoftMotion que possibilita o controle de até **32 eixos** reais ou virtuais, viabilizando uma extensa variedade de controles de movimento, como o posicionamento de eixos simples, sincronização de múltiplos eixos (cames eletrônicas e engrenagens eletrônicas), interpolação de múltiplos eixos (linear, circular e helicoidal), controle de velocidade, controle de torque, leitura e interpretação de Código-G, controle de máquinas CNC, controle para máquinas de corte e robôs industriais, entre outras funcionalidades.

É desenvolvido para atender aplicações de médio e grande porte. Possui alta velocidade de processamento devido a sua CPU composta por um processador Dual-core ARM Cortex-A7 rodando a 1 GHz, um coprocessador Real-time ARM Cortex-M4 de 200 MHz, memória RAM de 1 GByte e Flash de 4 GBytes.

Possui um total de 8 saídas digitais, sendo 3 destas com funcionalidade PWM até 300 kHz, e 8 entradas digitais, das quais 4 podem operar até 150 kHz.

Como interfaces de comunicação, estão disponíveis duas portas Ethernet independentes, porta CAN, serial RS485, USB OTG, USB device e Micro SD Card.

São utilizados supercapacitores internos para o Relógio de Tempo Real (RTC) e também para salvar dados retentivos em memória Flash durante o Power Off, dispensando assim o uso de baterias.

O PLC500MC permite a conexão de cartões de expansão de entradas e saídas digitais, analógicas, termopar, PT100, PT1000, célula de carga, relés, etc., dando mais flexibilidade às aplicações. Possui conectores plug-in e a fixação pode ser feita em trilho DIN 35 ou diretamente no painel.

A programação do PLC500MC é realizada pelo software CODESYS, amplamente difundido no meio industrial, possibilitando a utilização de uma infinidade de aplicações, e funções já desenvolvidas no mercado, bem como a importação de aplicações de outros produtos.

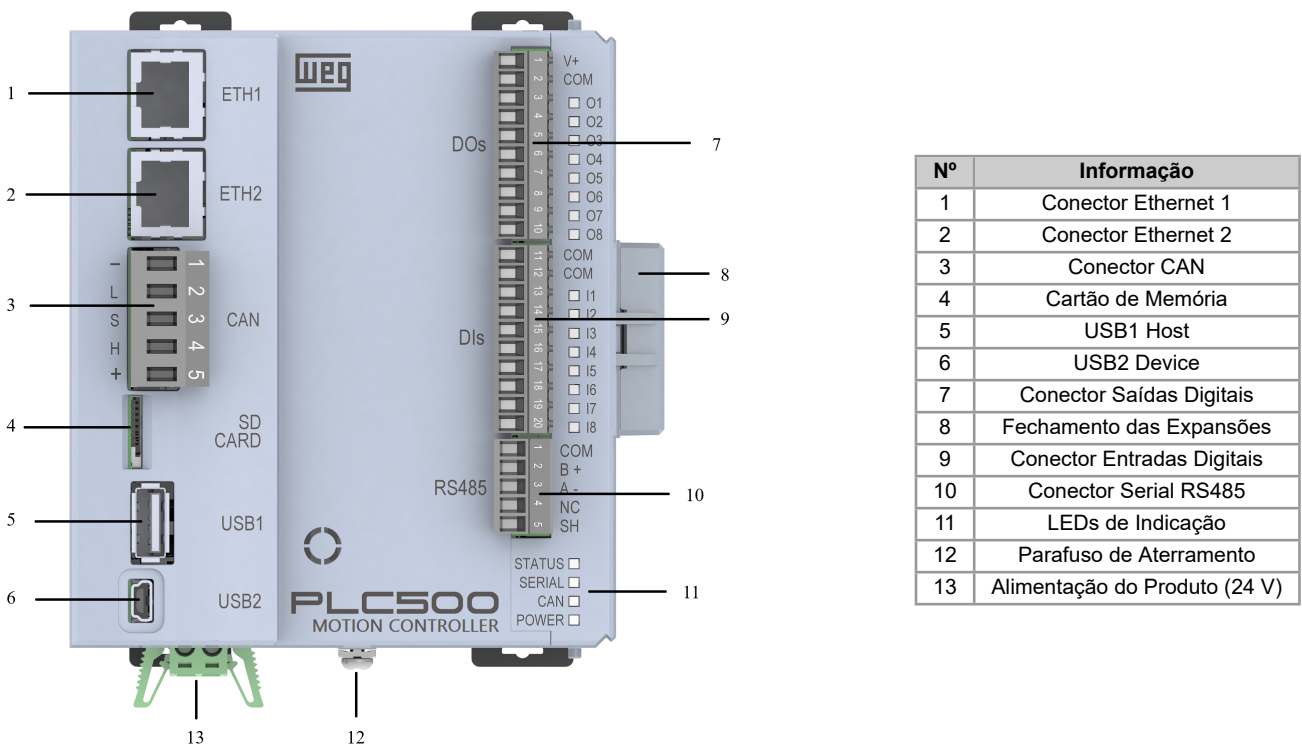


Figura 1.1: PLC500 Motion Controller.

O PLC500MC possui grande área de memória disponível para o usuário. O uso da memória de uma aplicação pode ser visualizado através do codesys em: View->View memory usage.

A memória do PLC500MC é dividida conforme tabela abaixo.

Memória	Capacidade	Descrição
Área 0 (DATA)	128M Bytes	Armazena todos os dados locais e globais (variáveis, blocos de função, instâncias, etc).
Área 1 (CODE)	32M Bytes	Armazena todo código gerado pela aplicação e também os dados constantes.
Área 2 (RETAIN)	64k Bytes	Armazena as variáveis do tipo retain (mantém o valor após o reboot do controlador).
Área 3 (PERSISTENT)	16k Bytes	Armazena as variáveis do tipo persistent (mantém o valor após o reboot e também após o download, se o layout delas se manter idêntico).

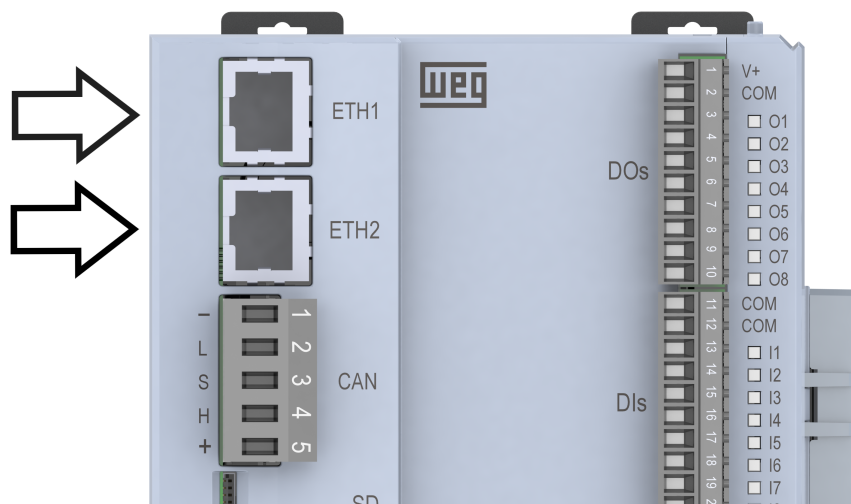
**Tabela 1.1:** Áreas de memória.

## 1.3 TECNOLOGIA ETHERCAT

EtherCAT (**E**thernet for **C**ontrol **A**utomation **T**echnology) é uma poderosa tecnologia para comunicação de tempo real baseada em Ethernet. Com seus tempos de ciclo curtos, baixo valores de jitter e diferentes topologias de rede, o sistema é padrão em muitas aplicações de automação industrial atualmente.

### 1.3.1 Interfaces EtherCAT PLC500MC

O PLC500MC possui duas interfaces independentes (**ETH1** e **ETH2**) que podem ser utilizadas para a comunicação EtherCAT. A Figura 1.2 mostra o PLC500MC e suas duas interfaces possíveis para a comunicação EtherCAT.



**Figura 1.2:** Interfaces EtherCAT

### 1.3.2 Abrangência EtherCAT

As funcionalidades suportadas pelo protocolo EtherCAT disponível no PLC500MC incluem:

- Diferentes topologias de barramento (linha e estrela).
- Grande flexibilidade com conexão quente.
- Clocks distribuídos.
- Diagnóstico de barramento: pelo editor e pela aplicação.
- Varredura de rede: reconhecer e insirir escravos conectados.
- Redundância EtherCAT.
- Camadas de protocolos suportados:

# INTRODUÇÃO

---

- CoE (CANopen over EtherCAT) / Comunicação SDO.
- EoE (Ethernet over EtherCAT).
- SoE (Servodrive over EtherCAT).
- FoE (File over EtherCAT).
- VoE (Vendor over EtherCAT).
- Suporte para escravos MDP (Modular Device Profile).
- Diversos blocos de função para uso na aplicação.

Para as configurações da rede, o PLC500MC possui uma interface que facilita as configurações do mestre da rede EtherCAT e seus escravos. Através desta interface é possível:

- Configurar a rede de forma automática ou utilizar o modo especialista.
- Adicionar e configurar escravos utilizando arquivos XML EtherCAT (ESI).
- Configurar unidades de sincronização. (Sync Unit)
- Configurar PDOs (dados de processo).
- Configurar parâmetros de inicialização para CoE e SoE.
- Configurar escravos EoE.
- Visualizar objetos CoE de forma online e suporte para upload de SDOinfo.
- Visualizar histórico de diagnóstico da rede de forma online.
- Ler e escrever na memória EEPROM dos dispositivos.

## 1.4 CONTROLE DE MOVIMENTO

O PLC500MC possibilita o controle de movimento para eixos únicos e múltiplos eixos sincronizados (cames eletrônicos e engrenagens eletrônicas) além de possibilitar o controle de máquinas CNC e robôs industriais.

Os servoconversores compatíveis com a CiA402 podem ser operados facilmente pelo PLC500MC sem que os usuários se preocupem com palavra de status, palavra de controle, modo de operação e outros parâmetros necessários para o controle de movimento.

O PLC500MC apresenta diversas funcionalidades específicas para o controle de movimento, entre elas:

- Extensa biblioteca com blocos para controle de eixos, manipulação e processamento de caminhos CNC, grupos de eixo, além de transformações cinemáticas populares.
- Editor came integrado.
- Editor CNC 3D integrado de acordo com DIN 66025 (G-Code).
- Configurador de grupos de eixos para diferentes cinemáticas (constumizável).
- Fácil comissionamento de eixos (utilizando **Online Configuration Mode**).
- Blocos de função certificados de acordo com **PLCopen MotionControl, Part 1 (V20)**.
- Decodificador de **G-code**, incluindo suporte para subprogramas e expressões em **G-code**.
- Blocos de função para testar velocidades de transição.
- Blocos de função para leitura e processamento de caminhos CNC de arquivos (para caminhos criados e processados externamente).
- Blocos de função certificados de acordo com **PLCopen MotionControl Part 4 (movimento coordenado)**.

### 1.4.1 Editor came

O PLC500MC possui um editor de tabelas came que facilita a visualização e implementação para este fim.

Abrangência do editor came:

- Planejamento gráfico e numérico para o came usando qualquer base em representação de distância, velocidade, aceleração e jerk.
- Interpolação linear ou polinomial (polinômio de 5ª ordem).
- Configuração dos tuchos e seu comportamento de chaveamento no came.
- Configuração do came relativos a requisitos de dimensão, período e continuidade.
- Possibilidade de importar e exportar tabelas came.

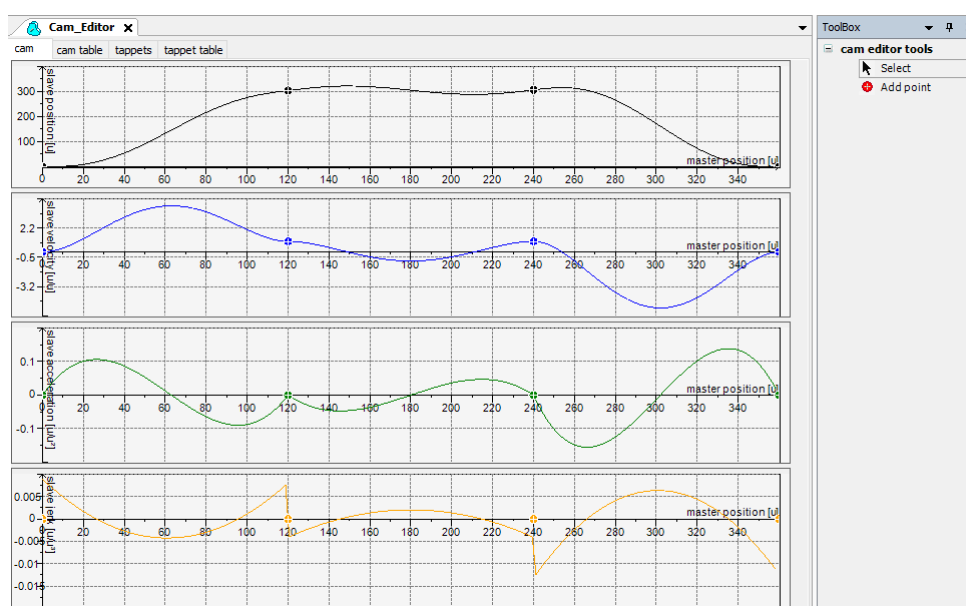


Figura 1.3: Editor came.

Informações adicionais referentes ao Editor Came podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > Basic Motion > Cams).

### 1.4.2 Editor CNC 3D

O PLC500MC possui a capacidade de interpretar e executar programas G-Code de acordo com DIN 66025.

Abrangência do editor CNC 3D de acordo com DIN 66025 (G-Code):

- Editor gráfico e textual simultâneo.
- Pré-processamento de caminho (visualização offline dos efeitos, por exemplo, suavização de ângulo).
- Pré-interpolação de caminho (pré-visualização (offline) da posição resultante, velocidade, aceleração e curvas jerk de todos os eixos suportados).
- Importar arquivos DXF e ASCII (.cnc, .gcode, .txt).
- Ler e salvar em arquivo.
- Transformações do programa (girar, deslocar e redimensionar o código G).

# INTRODUÇÃO

- Conversão para tabelas.
- Informações do programa (comprimento do caminho, duração do caminho, número de objetos, etc.)

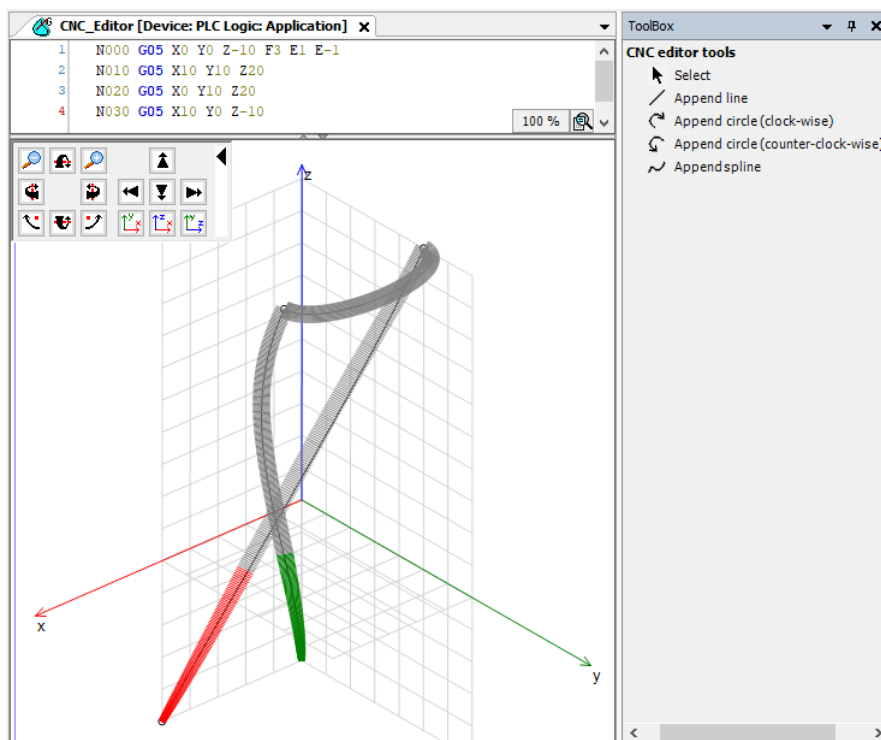


Figura 1.4: Editor CNC 3D.

Informações adicionais referentes ao Editor CNC 3D podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > CNC > Editor).

## 1.4.3 Abrangência das bibliotecas SoftMotion + CNC Robotic

As instruções para o controle de movimento são definidas como blocos de função (FB) e podem ser utilizadas durante a aplicação para realizar uma extensa variedade de movimentos. As instruções para o controle de movimento são desenvolvidas com base nas especificações dos blocos de função PLCopen<sup>1</sup>. Além das instruções baseadas na PLCopen, também estão disponíveis blocos adicionais que facilitam a implementação do controle de movimento.

- Blocos de função certificados de acordo com PLCopen MotionControl, Parte 1 (V20):
  - Posicionamento absoluto e relativo (**MC\_MoveAbsolute**, **MC\_MoveRelative**).
  - Posicionamento sobreposto (**MC\_MoveSuperimposed**).
  - Movimento em velocidade constante (**MC\_MoveVelocity**).
  - Suporte consistente de perfis com limitação de jerk (aceleração contínua para qualquer tipo de interrupção de o movimento atual).
  - Retorno guiado por drive (**MC\_Home**).
  - Parada de bloqueio (**MC\_Stop**).
  - Liberação de controle (**MC\_Power**).
  - Leitura e gravação em parâmetros (**MC\_Read/WriteParameter**).
  - Leitura da posição real (**MC\_ReadActualPosition**).
  - Perfis de posição, velocidade e aceleração (**MC\_\*Profile**).
  - Definir e mover a posição (**MC\_SetPosition**).

<sup>1</sup>A PLCopen é uma organização que promove o controle industrial com base na norma IEC61131-3. Para mais informações sobre o PLCopen, consulte o site oficial em: <http://www.plcopen.org/>.

- Leitura da velocidade real e o torque real (**MC\_ReadActualVelocity**, **MC\_ReadActualTorque**).
  - Chaveamento de came (**MC\_DigitalCamSwitch**).
  - Engrenagem eletrônica com posição de sincronização (**MC\_GearInPos**).
  - Parada completa (**MC\_Halt**).
  - Rastreamento de sinais mestres respeitando os limites de velocidade, aceleração e jerk (**SMC\_TrackSetValues**).
  - Blocos adicionais.
  - Controle e consulta do freada.
  - Monitoramento do erro de lag, uma janela de posição ou valores máximos.
  - Medição de distância percorrida.
  - Gerenciamento de erros nos blocos de função.
  - Retorno guiado pelo controlador (**MC\_Homing**).
  - Comissionamento de dispositivos.
  - Posicionamento absoluto e relativo com velocidade de transição (**SMC\_MoveContinuousAbsolute** e **SMC\_MoveContinuousRelative**)
  - Configuração do modo de controle (posição, velocidade ou torque).
- Modelos de visualização para os blocos de função mais importantes utilizados para o comissionamento rápido integrado no software Codesys.
  - Decodificador de código G.
  - Suporte para subprogramas e expressões em código G.
  - Limitador para restringir os valores de dinâmicas de velocidade e aceleração para um ou mais eixos.
  - Blocos para testar velocidades nas transições.
  - Interpolador para calcular os pontos do caminho CNC com base no perfil de velocidade.
  - Blocos para transformação de coordenadas (**SMC\_ScaleQueue3D** e **SMC\_CoordinateTransformation3D**).
  - Blocos de transformação (incluindo inversa) para cinemáticas populares:
    - Sistemas pórticos (gantry) 2D / 3D.
    - Sistemas pórticos (gantry) com eixos de orientação e compensação de ferramenta.
    - Sistemas pórticos (gantry) com acionamento por correia (portais H e portais T).
    - Transformação polar.
    - SCARA de 2/3 braços.
    - Bipod.
    - Tripod com eixos lineares e articulados.
    - Cinemática de 5 eixos para portal de 3 eixos com ferramenta rotativa e basculante.
    - Cinemática de 4 eixos para robôs de paletização.
    - Cinemática de 6 eixos para robôs de braço articulado.
  - Blocos para leitura e processamento de caminhos CNC (para caminhos criados e processados externamente).
  - Modos de velocidade de caminho trapezoidal/sigmoidal/quadrático/quadrático suave.
  - Função hodômetro.
  - Transformação de coordenadas 3D parametrizável (incluindo inversa).
  - Biblioteca de funções certificada com blocos de função de acordo com PLCopen Motion, Part 4 (Movimento coordenado).
    - Blocos administrativos: **MC\_GroupEnable/Disable/Reset/ReadError**, etc.
    - Comandos de movimento: **MC\_MoveDirectAbsolute**, **MC\_MoveDirectRelative**, **MC\_MoveCircular\***, **MC\_MoveLinear\***, **MC\_GroupHalt**, **MC\_GroupStop**.



## INTRODUÇÃO

---

- Acompanhamento: **MC\_TrackConveyorBelt**, **MC\_TrackRotaryTable**, **MC\_SetDynCoordTransform**.
- Modo jog em qualquer sistema de coordenadas: **SMC\_GroupJog2**.
- Suporte de diferentes sistemas de coordenadas: coordenadas globais (WCS), coordenadas de máquina (MCS), diversas coordenadas do produto (PCS\_1, PCS\_2), coordenadas da ferramenta (TCS) e coordenadas do eixo (ACS).
- Suporte para espera no caminho com tempo de espera (**SMC\_GroupWait**).
- Interface pública e documentada para criar cinemática específica do usuário na linguagem IEC 61131-3.
- Cinemática de orientação adicional, que pode ser combinada com as outras cinemáticas.
- Ferramentas com orientação e deslocamento de posição.

## 2 CRIAR E CONFIGURAR REDE ETHERCAT + SOFTMOTION

Nesta seção são descritas as etapas necessárias para realizar uma comunicação EtherCAT entre o PLC500MC e o servoconversor SCA06 através do software Codesys. Informações adicionais e configurações avançadas serão apresentadas nas demais seções desta nota de aplicação ou podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com>.

### 2.1 COMPONENTES UTILIZADOS

Os componentes necessários para este manual de aplicação:

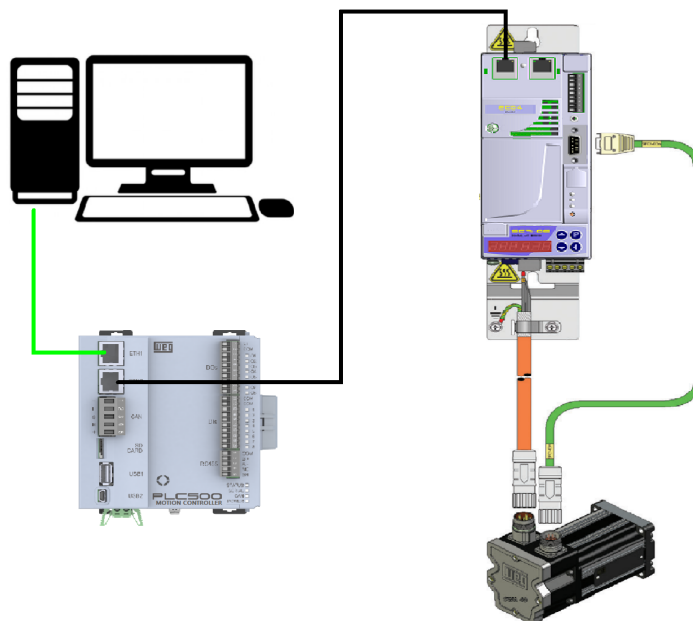
Componente	Versão FW
PLC500MC	1.2.0 ou superior
Servoconversor SCA06	2.11 ou superior
Servomotor	Compatível com o servoconversor
Acessório EtherCAT ECO4	Rev. 2436 ou superior

**Tabela 2.1:** Componentes necessários.

Para os componentes passivos de rede (cabos, conectores e fonte de alimentação), utilize somente componentes certificados para aplicações industriais. Consulte a documentação dos produtos para mais informações sobre a instalação adequada do servoconversor SCA06 e do servomotor utilizado.

### 2.2 ARQUITETURA DE REDE

A Figura 2.1 mostra a topologia da rede utilizada, o computador deve estar conectado ao PLC500MC através da interface ETH1 ou USB2. A comunicação EtherCAT com o servoconversor SCA06 utilizará a interface ETH2 do PLC500MC.



**Figura 2.1:** Arquitetura de rede.

### 2.3 CONFIGURAÇÃO DO SERVOCONVERSOR SCA06

Conecte corretamente o acessório EtherCAT ECO4 e o servomotor ao servoconversor SCA06.

Partindo dos parâmetros de padrão de fábrica do SCA06:

- Altere o parâmetro **P0202** para **5** (controle via rede CAN/EtherCAT).

## CRIAR E CONFIGURAR REDE ETHERCAT + SOFTMOTION

- Altere o parâmetro **P0385** para configurar o modelo do motor conforme placa do mesmo e tabela de motores.

Siga as recomendações descritas no manual do usuário do servoconversor SCA06 para programar parâmetros de ajuste do equipamento, relativos à parametrização do motor, funções desejadas para os sinais de I/O, etc...

Em caso de dúvida, consulte o Manual de Programação do servoconversor SCA06.

- Reinicie o servoconversor.

Com isso, o servoconversor SCA06 estará pronto para ser acessado através da rede EtherCAT.

### 2.4 CRIAR UM PROJETO NO CODESYS

- Faça o download do software Codesys e a instalação do **WEG Package** conforme o manual do PLC500.
- Após a instalação abra o Codesys e crie um novo projeto em **File > New Project**. Selecione **Standard Project**, defina um diretório e o nome da aplicação. Selecione o Device **PLC500MC** e a linguagem de programação desejada, conforme a Figura 2.2.

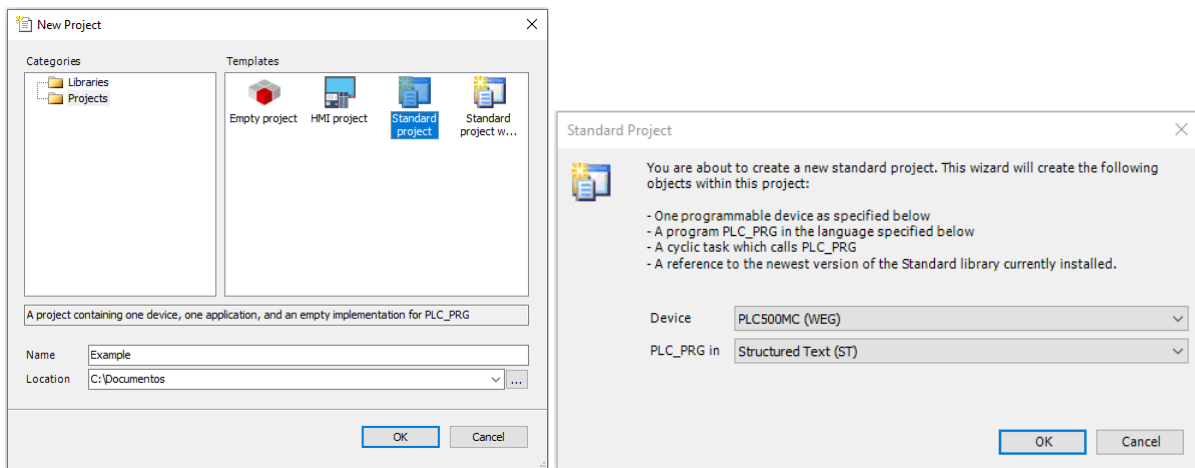


Figura 2.2: Configuração do projeto no Codesys.

Criando uma aplicação para o Device PLC500MC as interfaces de redes padrão serão pré-configuradas automaticamente, como indicado na Figura 2.3.

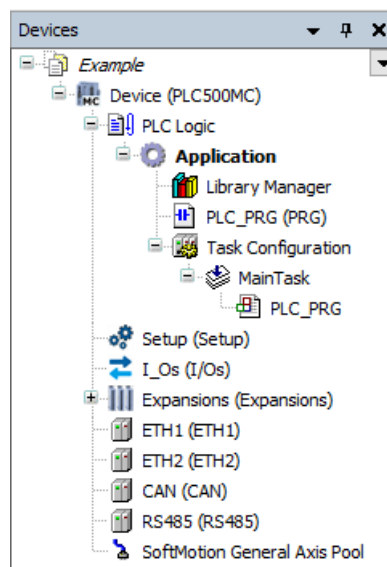
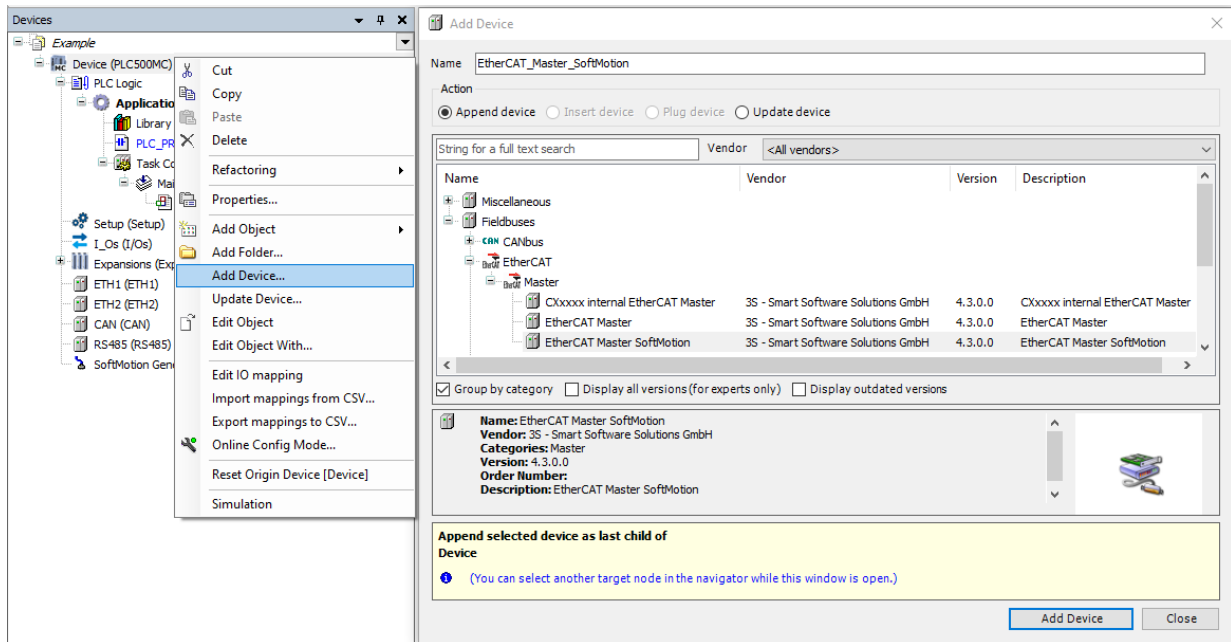


Figura 2.3: Interfaces PLC500MC

## 2.4.1 Adicionar EtherCAT Master SoftMotion

- Para adicionar uma nova interface de comunicação **EtherCAT Master SoftMotion** clique com o botão direito em cima do **Device (PLC500MC)**, clique em **Add Device**, na caixa de diálogo selecione **Fieldbuses > EtherCAT > Master > EtherCAT Master SoftMotion** clique em **Add Device** para adicionar à árvore de dispositivos, conforme a Figura 2.4.



**Figura 2.4:** Adicionando **EtherCAT Master SoftMotion** à árvore de dispositivos.

Ao adicionar a interface de comunicação **EtherCAT Master SoftMotion** automaticamente será criada uma tarefa chamada **EtherCAT\_Task<sup>2</sup>**.

## 2.4.2 Adicionar SCA06\_SoftMotion como escravo na rede EtherCAT

- Para adicionar o dispositivo **SCA06\_SoftMotion** como escravo da rede EtherCAT clique com o botão direito no dispositivo **EtherCAT Master Softmotion** criado anteriormente e selecione a opção **Add Device**.)
- Na seção **Action**, da caixa de diálogo aberta, certifique-se de que a opção **Append device** esteja selecionada. Busque pelo dispositivo **SCA06\_SoftMotion**, ele se encontra dentro da pasta **WEG > Servo Drives**.
- Clique em **Add Device**.

A Figura 2.5 apresenta os passos anteriores diretamente no software Codesys.

<sup>2</sup>Tarefa utilizada para os comandos de controle de movimento **SoftMotion**.

## CRIAR E CONFIGURAR REDE ETHERCAT + SOFTMOTION

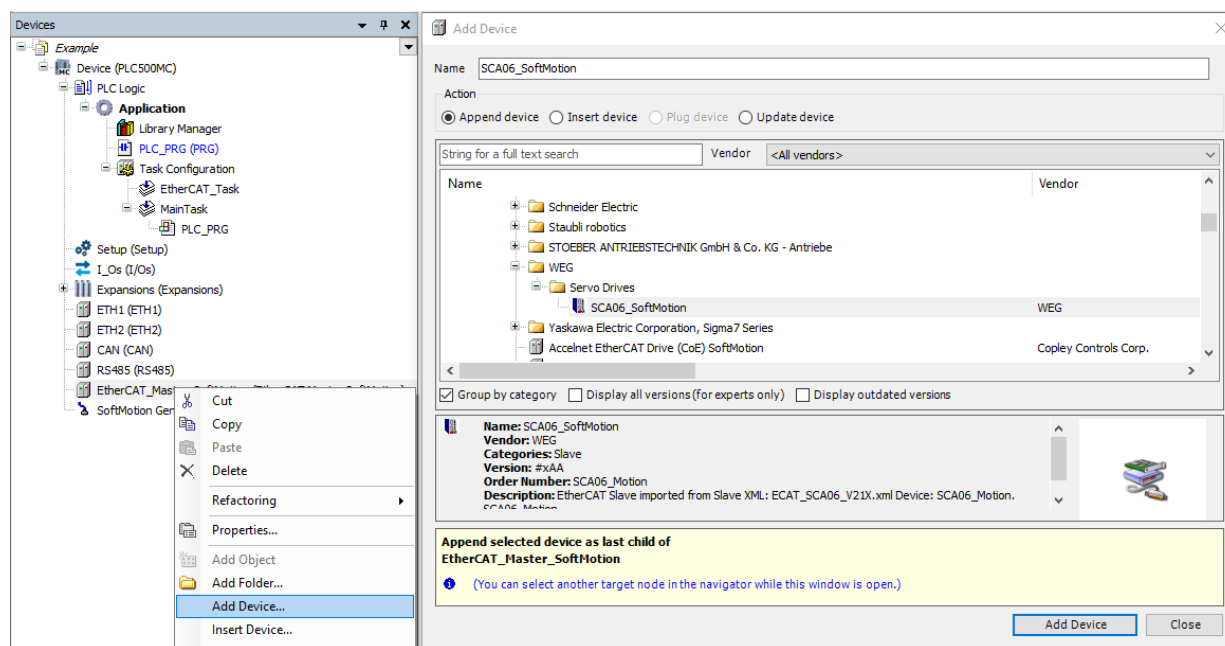


Figura 2.5: Adicionando SCA06\_SoftMotion como escravo na rede EtherCAT.



### NOTA!

É importante adicionar a unidade SoftMotion, pois a unidade padrão, importada por um XML, não contém um eixo SoftMotion associado (o **SCA06\_SoftMotion** é instalado juntamente com o WEG Package).

Após estas configurações, à árvore de dispositivos deverá conter os ícones apresentados na Figura 2.6.

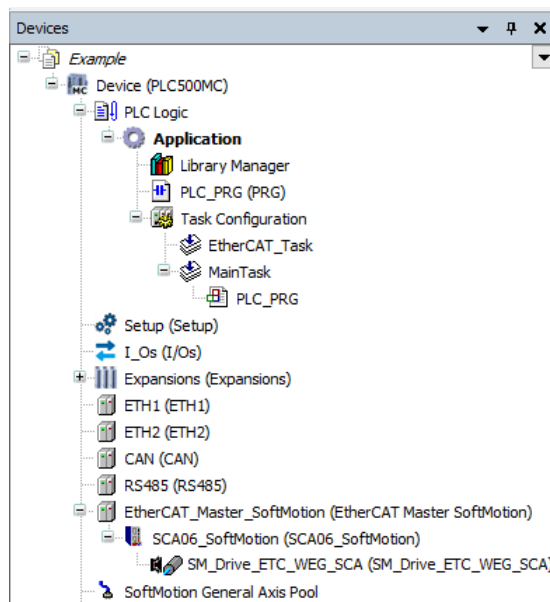
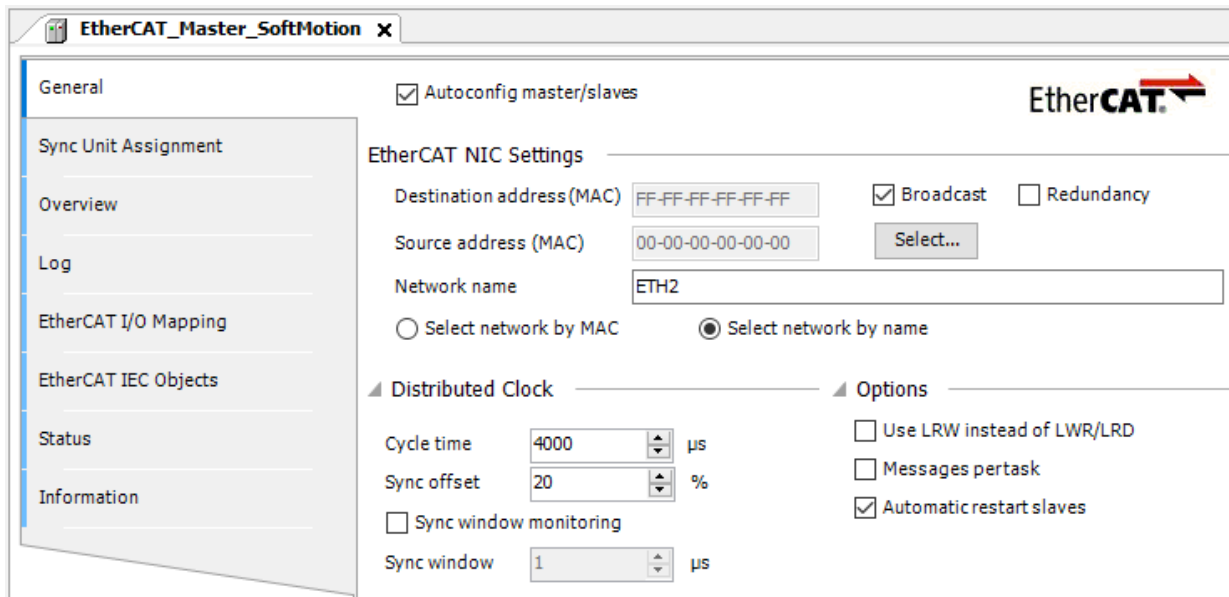


Figura 2.6: Árvore de dispositivos para utilização do SoftMotion.

### 2.4.3 Configurar EtherCAT Master SoftMotion

- Abra as configurações do dispositivo **EtherCAT Master SoftMotion**, na aba **General**, selecione a opção **Autoconfig Master/Slave**. Com isso, as principais configurações de mestre/escravo serão feitas automaticamente com base no arquivo de descrição do dispositivo.
- Configure as demais opções da página conforme a Figura 2.7



**Figura 2.7:** Configuração padrão EtherCAT Master SoftMotion.

Informações sobre configurações avançadas serão apresentadas na Seção 4 ou podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Fieldbus Support > EtherCAT > Configuration).

### 2.4.4 Configurar SCA06\_SoftMotion

Utilizando a opção **Autoconfig Master/Slave** no **EtherCAT Master Softmotion** a configuração do servoconversor **SCA06\_SoftMotion** será feita automaticamente.

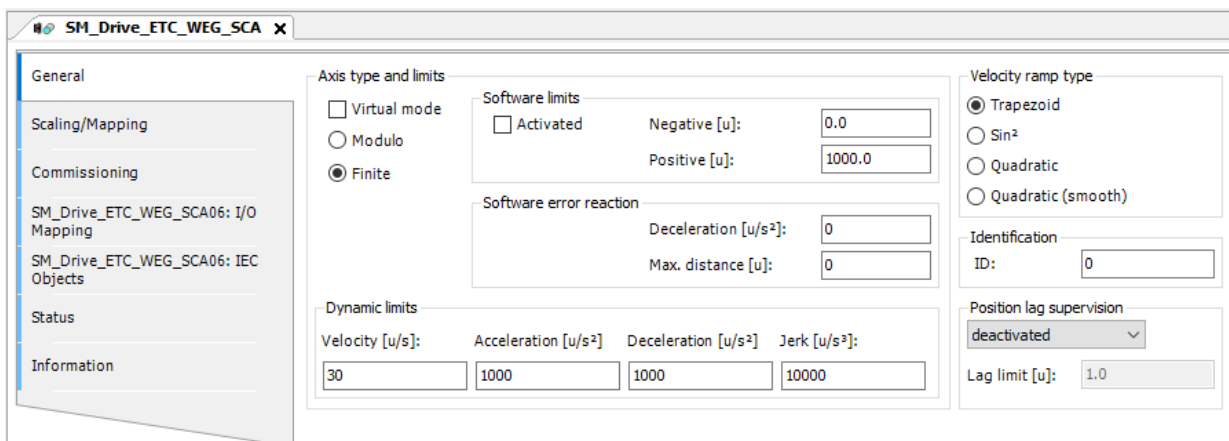
Informações sobre configurações avançadas serão apresentadas na Seção 4 ou podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Fieldbus Support > EtherCAT > Configuration).

### 2.4.5 Configurar SM\_Drive\_ETC\_WEG\_SCA

- Abra as configurações do SM\_Drive\_ETC\_WEG\_SCA.

Na aba **General** encontram-se as configurações referente ao tipo e limites do eixo, tipo da rampa de velocidade e supervisão de lag.

- Configure a página conforme a Figura 2.8.



**Figura 2.8:** Configuração padrão SM\_Drive\_ETC\_WEG\_SCA.

## CRIAR E CONFIGURAR REDE ETHERCAT + SOFTMOTION

Esta configuração define o eixo como finito, limitação por software desabilitada, rampa de velocidade do tipo trapezoidal, ID do eixo igual a 0, sem supervisão de lag e com os limites de dinâmica<sup>3</sup> definidos no campo *Dynamic limits*.

- Clique na aba **Scaling/Mapping**.

Na aba **Scaling/Mapping** pode-se definir a relação entre as unidades de aplicação (por exemplo, milímetros ou graus) e a unidade do servoconversor (pulsos).

- Configure a página conforme a Figura 2.9.



### NOTA!

É possível mapear manualmente as variáveis do SM\_Drive\_ETC\_WEG\_SCA. Para isso, no campo **Mapping**, desmarque a opção **Automatic mapping**.

Cyclic object	Object number	Address	Type
in.wStatusWord	16#6041:16#00	'%IW18'	'UINT'
diActPosition	16#6064:16#00	'%ID10'	'DINT'
diActVelocity	16#606C:16#00	'%ID11'	'DINT'

Figura 2.9: Configuração padrão de escala SM\_Drive\_ETC\_WEG\_SCA.

Esta configuração define que 65536 pulsos do servomotor equivalerá a uma unidade de aplicação, ou seja, cada unidade de aplicação será exatamente igual a um giro no servomotor<sup>4</sup>



### ATENÇÃO!

É imprescindível a correta configuração destes valores, pois os blocos de função SoftMotion utilizarão a **unidade de aplicação** como parâmetro para o movimento.

Informações sobre configurações avançadas serão apresentadas na Seção 5 ou podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > Reference > User Interface > Objects > SoftMotion Drives).

- Após aplicadas as configurações desta seção faça o download do programa para o **PLC500MC** e monitore no modo **Online**.

<sup>3</sup>Estes limites são levados em consideração quando utilizado grupo de eixos (PLCopen Parte 4). Além disso, são usados pelos blocos de função **SMC\_ControlAxisBy\*** para detectar saltos.

<sup>4</sup>O servoconversor SCA06 possui a resolução de 65536 pulsos por volta, consulte o manual EtherCAT do SCA06 para mais informações..

## 2.5 MONITORAÇÃO

### 2.5.1 Estado da comunicação EtherCAT

O estado da rede EtherCAT pode ser monitorado no modo **Online** do Codesys, indicando o estado de cada uma das etapas de comunicação e reportando o estado (Status). Ao encontrar problemas de conexão como mostrado na Figura 2.10, verifique novamente se os cabos estão devidamente conectados e revise as configurações feitas na Seção 2.

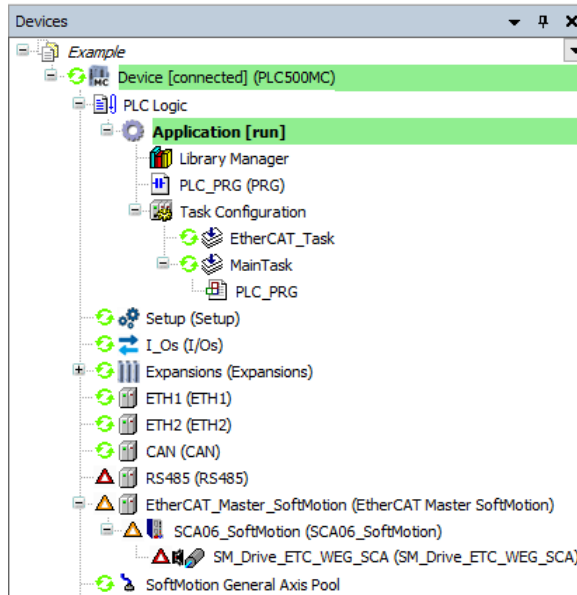


Figura 2.10: Indicação de erro na comunicação EtherCAT.

Quando as configurações estiverem corretas e os dispositivos estiverem comunicando adequadamente todos os itens da comunicação EtherCAT estarão em verde, como indicado na Figura 2.11.

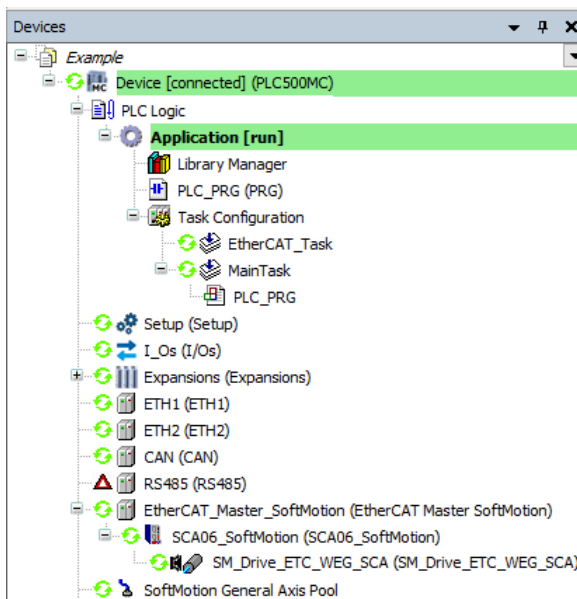


Figura 2.11: Comunicação corretamente configurada e dispositivos comunicados.

### 2.5.2 Verificar variação na posição atual do servomotor

- Após uma correta configuração da rede EtherCAT e ainda no modo **Online** abra as configurações do **SM\_Drive\_ETC\_WEG\_SCA**.



## CRIAR E CONFIGURAR REDE ETHERCAT + SOFTMOTION

Quando o PLC estiver no modo **Online**, na aba **General**, será habilitado um campo para visualização do eixo, conforme a Figura 2.12.

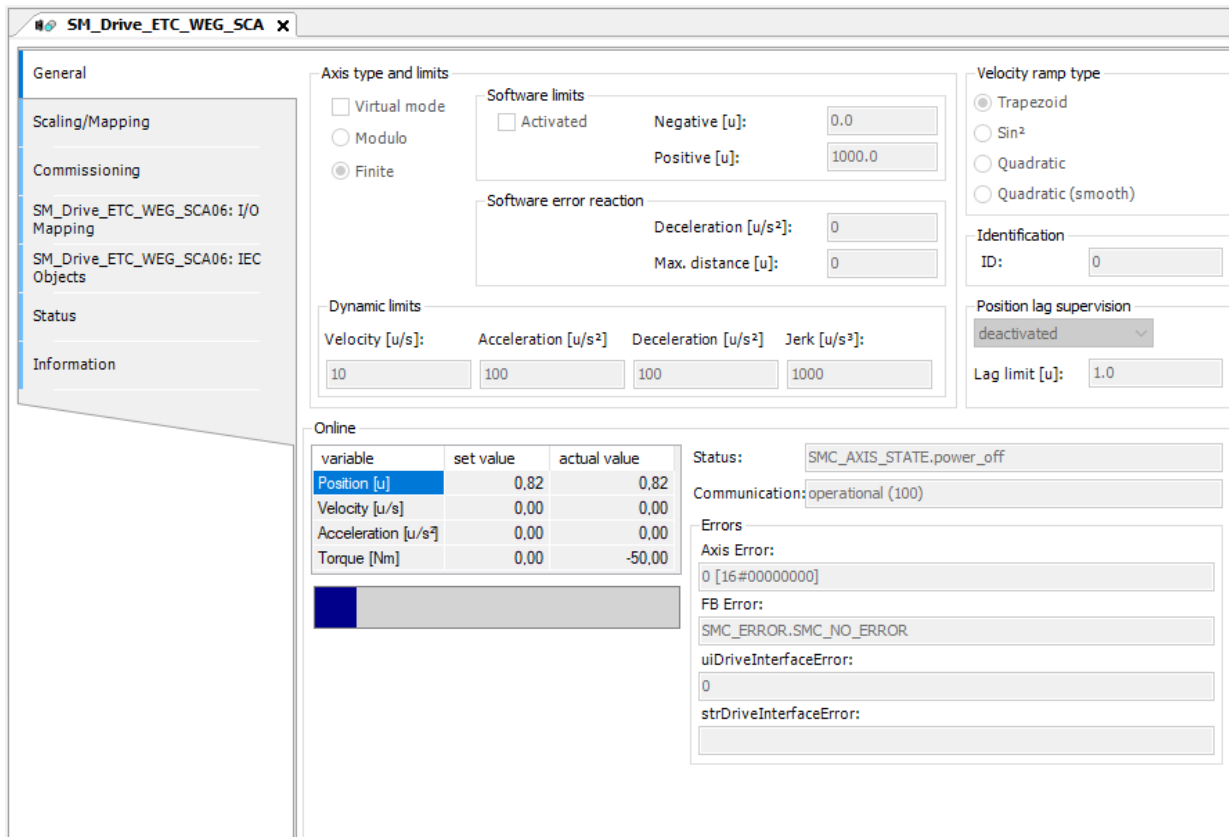


Figura 2.12: Monitoração online do servomotor.

Neste campo é possível observar o estado do eixo e da comunicação, variáveis de posição, velocidade, aceleração e torque, com suas referências e valores atuais.

- Movimente o eixo do servomotor manualmente e observe o valor da posição alterando em **Position [u] - actual value**.

## 2.6 COMISSIONAMENTO

É possível testar as configurações aplicadas para o servoconversor SCA06 através dos passos apresentados nesta subseção.

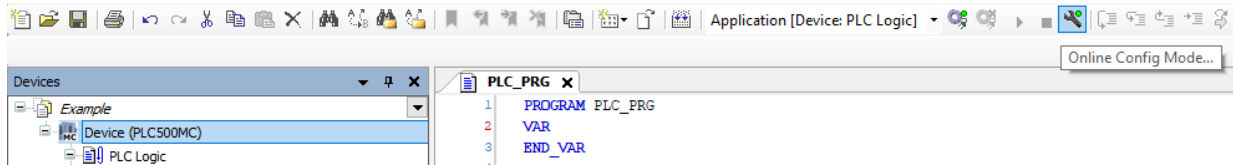
- Saia do modo **Online** e entre novamente no PLC utilizando a opção **Online Config Mode**. Este é o modo para configuração do PLC, através dele é possível testar e validar as configurações aplicadas para o servoconversor.



### ATENÇÃO!

Ao utilizar a opção **Online Config Mode** a aplicação presente no PLC será automaticamente apagada.

Para utilizar a opção **Online Config Mode**, na árvore de dispositivos, clique em **PLC500MC** e em seguida clique na opção **Online Config Mode**, como apresentado na Figura 2.13



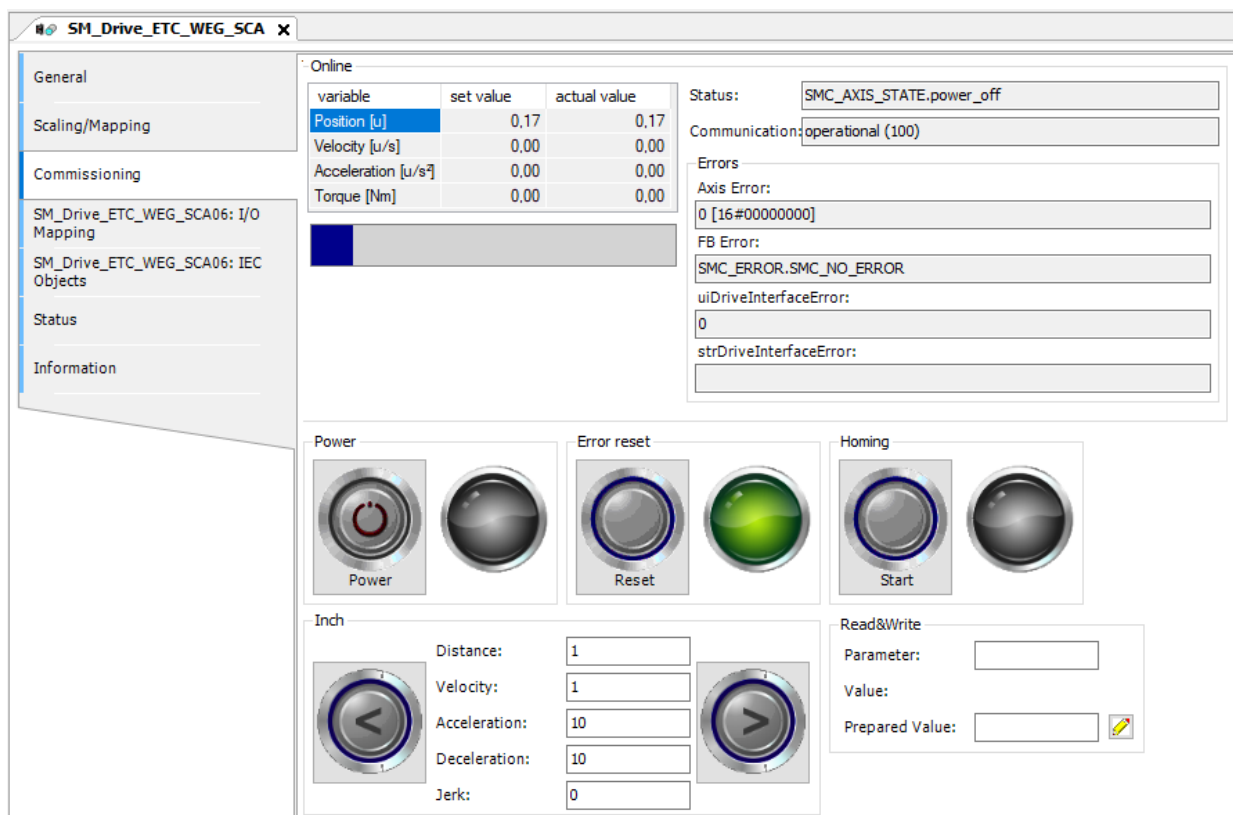
**Figura 2.13: Online Config Mode.**

- Abra as configurações do servomotor (**SM\_Drive\_ETC\_WEG\_SCA**) e clique na aba **Commissioning**. Nesta aba, além das variáveis e status do eixo, ficam disponíveis alguns botões para o acionamento do servomotor, como apresentado na Figura 2.14.



**NOTA!**

Esta página apenas é habilitada utilizando a opção **Online Config Mode**.



**Figura 2.14: Commisionamento SCA06\_SoftMotion.**



**ATENÇÃO!**

Pode-se mover o eixo utilizando os botões desta página. O eixo pode executar movimentos inesperados caso as configurações não estejam adequadas. Tome todas as precauções de segurança necessárias.

Elementos de operação	Descrição
Power	O Drive é habilitado (equivalente ao bloco de função <b>MC_Power</b> ).
Error reset	Reinicia Drive após um erro (equivalente ao bloco de função <b>MC_Reset</b> ).
Start homing	O Drive executa o comando <i>homing</i> com os parâmetros definidos internamente no servoconversor (equivalente ao bloco de função <b>MC_Home</b> )*.
Jogging mode	Utilizando os botões < e > pode-se mover o eixo para frente e para trás de acordo com os valores especificados para <b>Distance</b> , <b>Velocity</b> , <b>Acceleration</b> , <b>Deceleration</b> e <b>Jerk</b> (equivalente ao bloco de função <b>MC_Inch</b> ).
ReadWrite	Para o parâmetro do inversor especificado, o valor ( <b>Value</b> ) atual é lido pelo PLC e exibido. Em <b>Prepared value</b> , pode-se especificar um novo valor e escreve-lo no parâmetro do drive (equivalente aos blocos de função <b>MC_ReadParameter</b> e <b>MC_WriteParameter</b> ).

*Tabela 2.2: Elementos de comissionamento.*

- Clique no botão **Power** para habilitar o servoconversor, em seguida segure pressionado no botão >. O servomotor deverá realizar uma volta completa e parar.
- Se desejar, teste mais alguns comandos e saia do modo **Online Config Mode**.

### 3 APLICAÇÃO SOFTMOTION

Esta seção apresenta os passos necessários para a criação de uma aplicação SoftMotion para o controle de um eixo simples.

#### 3.1 CRIAR APLICAÇÃO

Para uma aplicação SoftMotion, é necessário criar um **POU** específico que será utilizado para o movimento dos eixos.

- Utilize como base as configurações apresentadas na Seção 2.4.
- Na árvore de dispositivos, clique com o botão direito no objeto **Application > Add Object > POU...**
- Crie um **POU** do tipo **Program** com o nome **MyMotion**.
- No campo **Implementation language**, selecione a opção **Structured Text (ST)**.
- Clique em **Add**.

A Figura 3.1 apresenta os passos anteriores diretamente pelo Codesys.

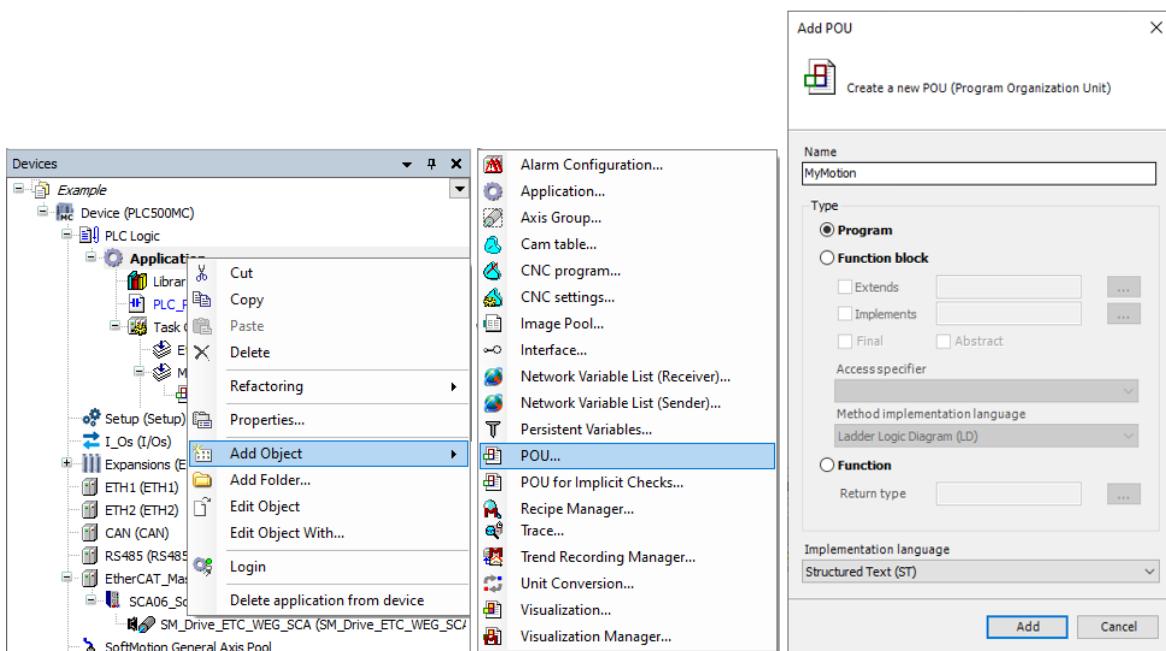


Figura 3.1: Criar POU SoftMotion.

Este **POU** deve ser chamado sob a tarefa **EtherCAT\_Task**.



**NOTA!**

Todos os blocos de função relativos ao movimento dos eixos devem ser declarados e chamados na **EtherCAT\_Task**. Demais funcionalidades devem ser utilizadas em tarefas diferentes, com uma menor prioridade.

- Arraste o **POU MyMotion** sob a tarefa **EtherCAT\_Task**, como apresentado na Figura 3.2.

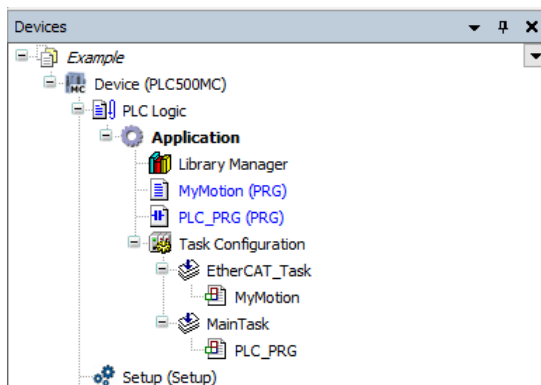


Figura 3.2: Adicionar o POU MyMotion à tarefa EtherCAT\_Task.

- Abra o **POU MyMotion**.
- Crie uma instância **MC\_Power** e outra **MC\_MoveRelative** e referencie a entrada **Axis** dos blocos de função com o nome do eixo criado, conforme apresentado na Figura 3.3.

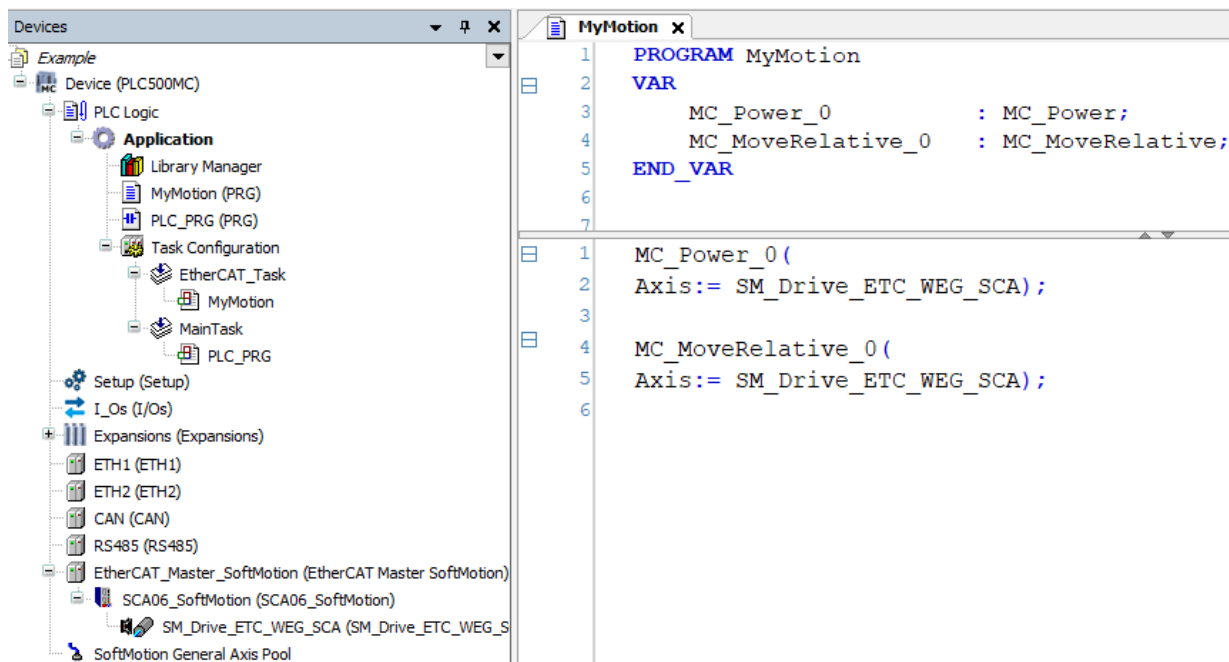


Figura 3.3: Adicionar o POU MyMotion à tarefa EtherCAT\_Task.

Aplicação MyMotion:
PROGRAM MyMotion
VAR
MC_Power_0 : MC_Power;
MC_MoveRelative_0 : MC_MoveRelative;
END_VAR
MC_Power_0(
Axis:= SM_Drive_ETC_WEG_SCA);
MC_MoveRelative_0(
Axis:= SM_Drive_ETC_WEG_SCA);

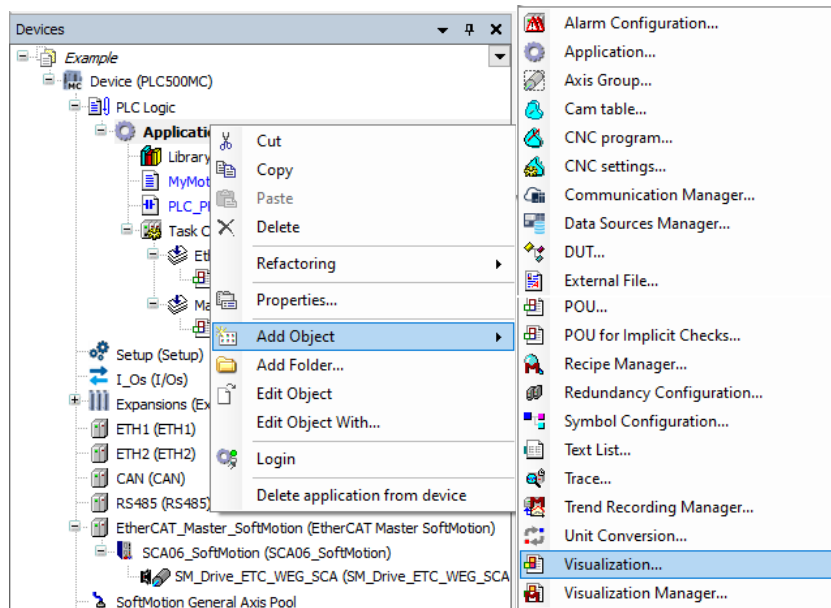
### 3.2 CRIAR VISUALIZAÇÃO



**NOTA!**

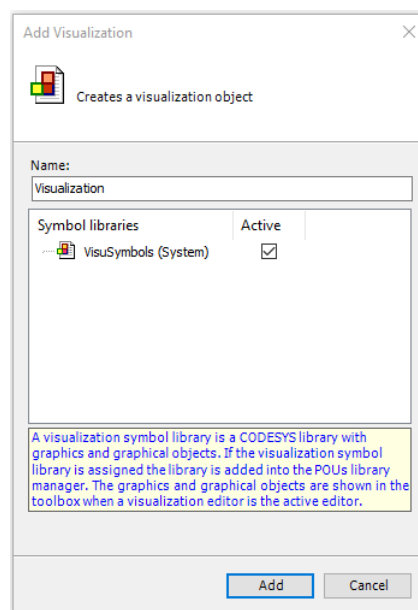
A biblioteca **SM3\_Basic** possui diversos modelos de visualização integrados que podem ser usados para testar a funcionalidade de um bloco de função de maneira simplificada.

- Adicione um objeto do tipo **Visualization** na árvore de dispositivos, conforme a Figura 3.4.



**Figura 3.4:** Adicionar o objeto do tipo **Visualization**.

Ao adicionar um objeto do tipo **Visualization**, uma caixa de diálogo será aberta como na Figura 3.5.



**Figura 3.5:** Caixa de diálogo aberta ao adicionar um objeto do tipo **Visualization**.

- Marque a opção **Active** e clique em **OK**.
- Abra o objeto **Visualization** criado.
- No campo **Visualization Toolbox**, localizado no lado direito da tela, selecione a aba **SM3\_Basic**. No

campo de busca digite **MC\_Power** e selecione o modelo **VISU\_NEW\_MC\_Power**, conforme apresentado na Figura 3.6.

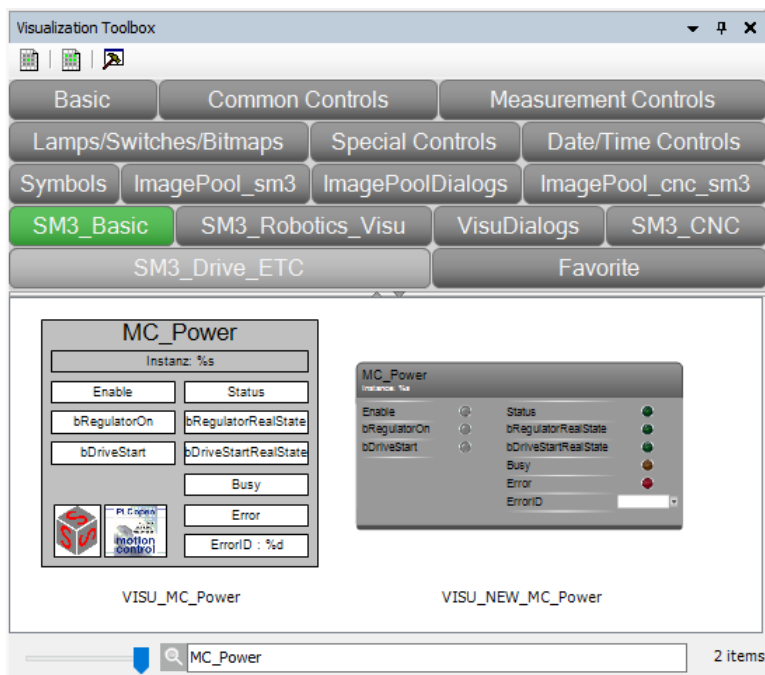


Figura 3.6: Buscar modelo de visualização MC\_Power.

- Arraste e solte o modelo na visualização.

Ao soltar o objeto, uma caixa de diálogo **Assign parameters** será aberta para o modelo de visualização.

- Realize um duplo clique em **Value** e clique em ...

Com isso, uma nova caixa de diálogo **Input Assistant** será aberta.

- Busque a instância do bloco de função **MC\_Power\_0** criada no POU **MyMotion** e clique em **OK**.

A Figura 3.7 apresenta os passos anteriores diretamente pelo Codesys.

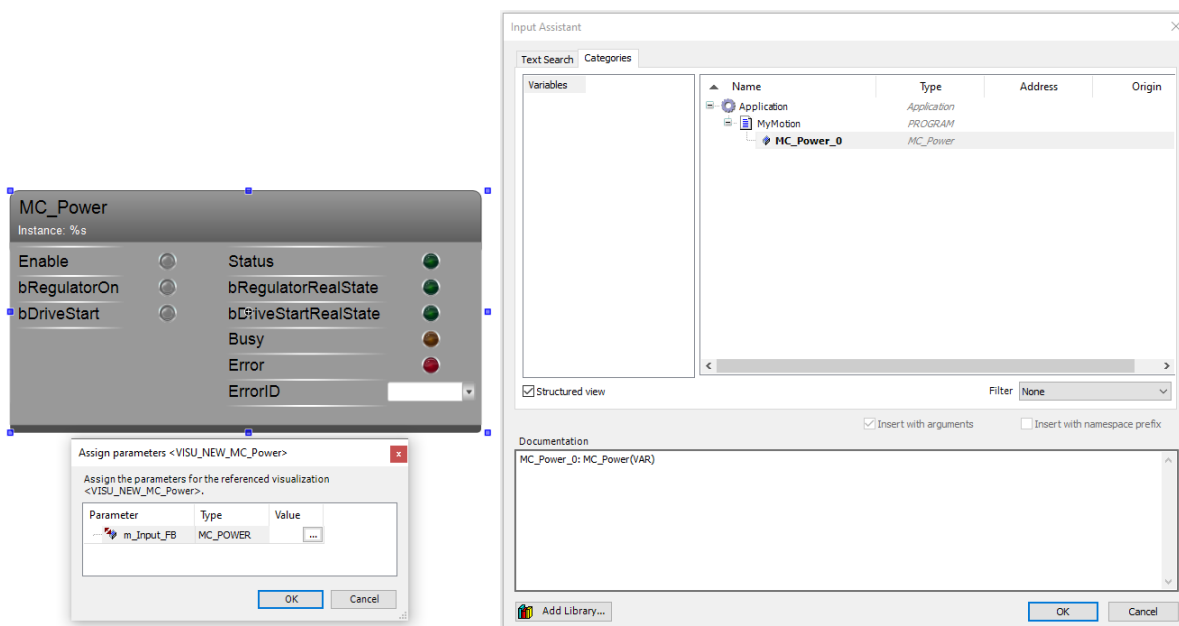


Figura 3.7: Adicionar o objeto do tipo Visualization.

Com isso, as entradas e saídas do modelo de visualização são mapeadas automaticamente para a instância do bloco de função.



**NOTA!**

Outra forma de mapear o modelo de visualização ao bloco de função criado pode ser feita selecionando o modelo e usando a aba **Properties > References > m\_Input\_FB**.

- Faça o mesmo procedimento agora utilizando o modelo **VISU\_NEW\_MC\_MoveRelative**, referenciando a instância do bloco de função **MC\_MoveRelative\_0**.

Após a configuração, a página de visualização deverá conter estes dois modelos de visualização mapeados nos blocos de função criados anteriormente, como na Figura 3.8.

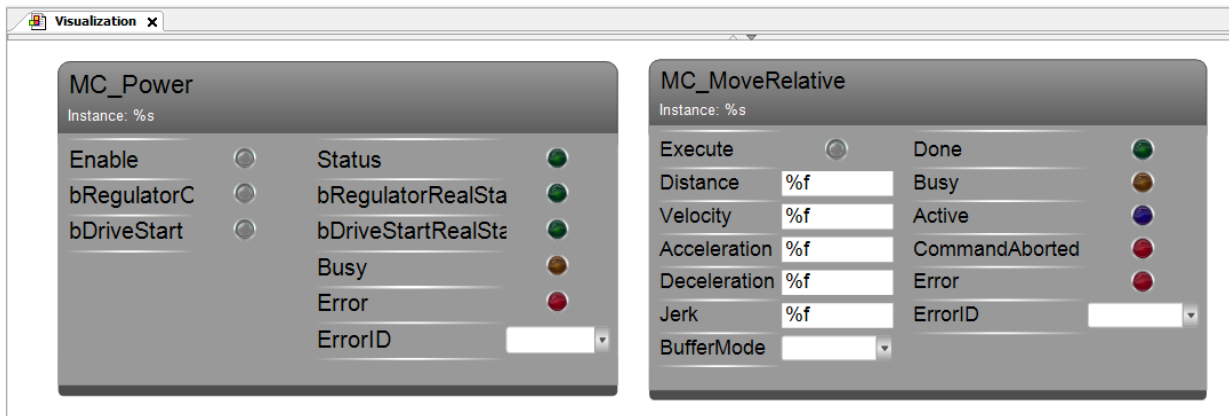


Figura 3.8: Adicionar o objeto do tipo Visualization.

Através destes modelos será possível controlar o eixo do servomotor.

- Faça o download do programa para o **PLC500MC**.
- No modo de monitoração **Online**, abra o objeto **Visualization**.



**ATENÇÃO!**

Pode-se mover o eixo por meio de botões nesta página. O eixo pode executar movimentos inesperados caso as configurações não estejam adequadas. Tome todas as precauções de segurança necessárias.

- No modelo **MC\_Power**, clique nos botões **bDriveStart**, **bRegulatorOn** e **Enable** respectivamente.

Observe as saídas do bloco, elas mostrarão o estado do servoconversor. Para o correto acionamento as saídas **Status**, **bRegulatorOnRealState** e **bRegulatorOnRealState** deverão estar em verde, como na Figura 3.9, indicando a habilitação do **Servoconversor** para o movimento.

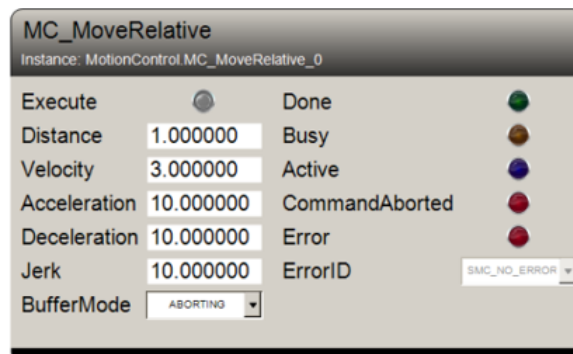


Figura 3.9: Exemplo servomotor habilitado.



## APLICAÇÃO SOFTMOTION

- No modelo **MC\_MoveRelative**, ajuste as variáveis relativas ao movimento (**Distance**, **Velocity**, **Acceleration**, **Deceleration** e **Jerk**) como apresentado na Figura 3.10. Após isso, clique no botão **Execute** para iniciar o movimento.



*Figura 3.10: Exemplo configuração de movimento relativo para o servomotor.*

Se desejar, realize mais alguns testes.

Outros exemplos de aplicação podem ser encontrados diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > Application Examples).

## 4 INFORMAÇÕES ADICIONAIS DA REDE ETHERCAT

Neste capítulo serão apresentadas algumas informações adicionais e configurações avançadas utilizadas na rede EtherCAT.

### 4.1 ATRIBUIR UM ENDEREÇO ESTÁTICO PARA O SCA06 NA REDE ETHERCAT

É possível definir um endereço estático para o **SCA06** como escravo na rede EtherCAT utilizando uma memória EEPROM interna do acessório **ECO4**.

- Utilize como base as configurações apresentadas na Seção 2.4.
- Abra as configurações do **SCA06\_Motion**, na aba **General**, habilite a opção **Expert settings**, após isso diversas configurações avançadas ficarão disponíveis.
- No campo **Identification**, selecione a opção **Configured station alias (ADO 0x012)**, conforme a Figura 4.1.

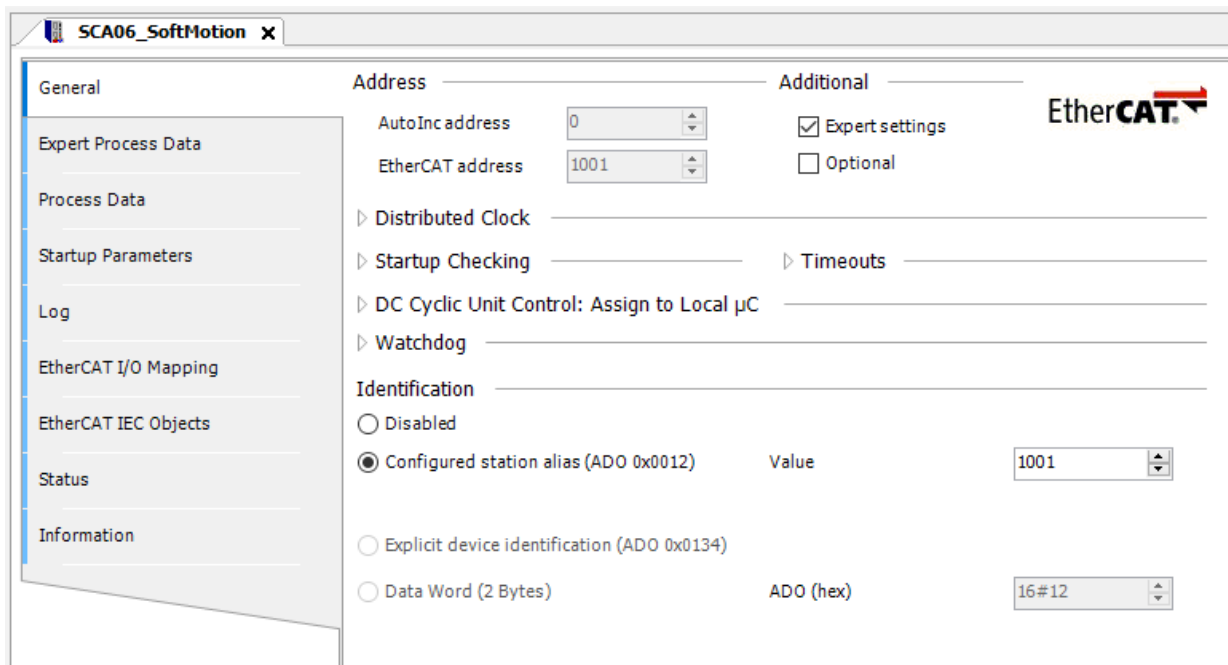


Figura 4.1: Habilitando identificação.

- Após aplicadas as configurações desta seção faça o download do programa para o **PLC500MC** e monitore no modo **Online**.

No modo **Online**, ao estabelecer uma comunicação com o **SCA06\_Motion**, no campo **Identification**, a variável **Actual adress** aparecerá informando o valor atual do endereço. A opção **Write to EEPROM** também estará disponível, como na Figura 4.2.

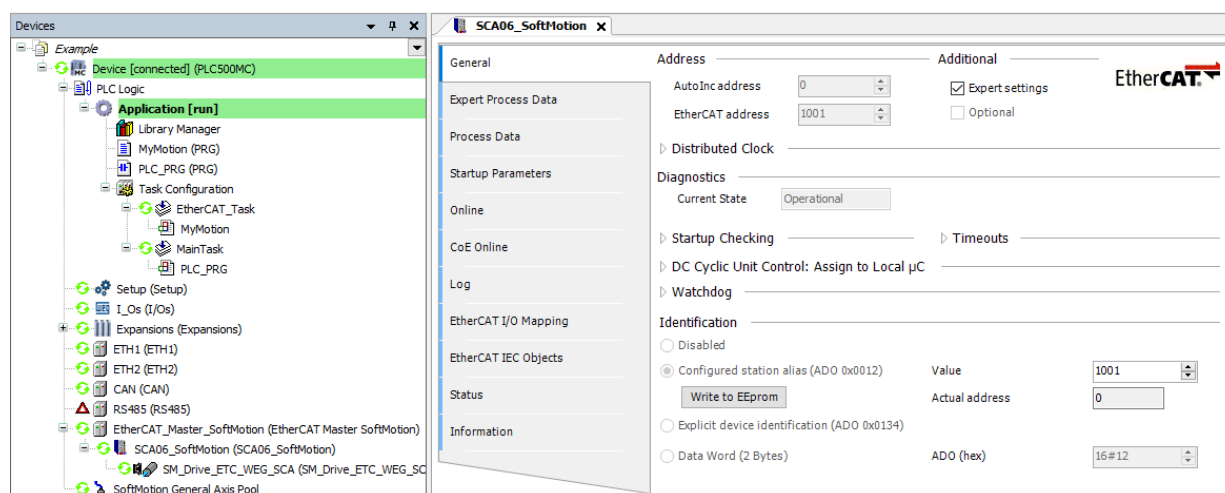


Figura 4.2: Endereço atual na memória EEPROM.

- Digite o endereço desejado no campo **Value** e clique na opção **Write to EEPROM**, como apresentado na Figura 4.3.



Figura 4.3: Escrevendo novo endereço na memória EEPROM.

Uma mensagem, como na Figura 4.4, aparecerá na tela solicitando que o servoconversor seja reiniciado para aplicar o novo endereço de rede.

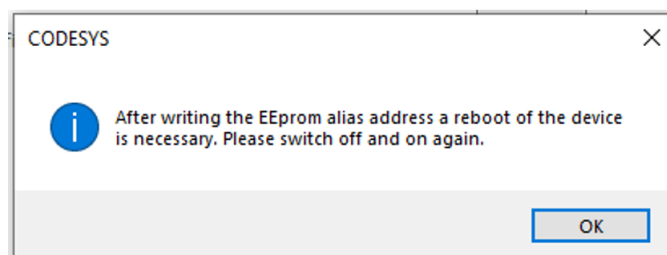


Figura 4.4: Mensagem de aviso para aplicar a escrita EEPROM.

- Reinicie o servoconversor.
- Saia do modo **Online**, no campo **Additional**, selecione a opção **Optional**. No campo **Configured station alias (ADO 0x0012)**, como apresentado na Figura 4.5. Certifique-se de que endereço é o mesmo que foi escrito na memória EEPROM anteriormente.

The screenshot shows the configuration interface for EtherCAT. On the left is a sidebar with menu items: General, Expert Process Data, Process Data, Startup Parameters, Log, EtherCAT I/O Mapping, EtherCAT IEC Objects, Status, and Information. The main area is divided into sections: Address, Additional, Identification, and Timeouts. Under the 'Additional' section, the 'Optional' checkbox is checked. Other settings include 'AutoInc address' (0), 'EtherCAT address' (1001), and 'Expert settings' (checked). The 'Identification' section has 'Configured station alias (ADO 0x0012)' selected with a value of 3. The 'ADO (hex)' field is set to 16#12. The EtherCAT logo is visible in the top right corner.

Figura 4.5: Habilitar o campo *Optional*.

- Faça o download do programa para o **PLC500MC** e monitore no modo **Online**.

Observe que agora o endereço atual será o endereço escrito na EEPROM do dispositivo, como apresentado na Figura 4.6.

This screenshot shows the 'Identification' section of the configuration interface. The 'Configured station alias (ADO 0x0012)' option is selected. A 'Write to EEPROM' button is visible. The 'Value' field is set to 3, and the 'Actual address' field also displays 3. The 'ADO (hex)' field remains at 16#12.

Figura 4.6: Novo endereço EEPROM.



**NOTA!**

Para utilizar o endereço escrito na memória EEPROM em uma rede com mais de um SCA06, é necessário que os dispositivos estejam marcados com a opção **Optional**, caso contrário, a rede será configurada automaticamente pelo mestre da rede sem utilizar o endereço da memória EEPROM.

## 4.2 LER E EDITAR PARÂMETROS NO SCA06 PELA REDE ETHERCAT

Utilizando este método é possível modificar parâmetros de configuração do **SCA06** remotamente através da rede EtherCAT, sem a necessidade de utilizar sua IHM.

- Utilize as configurações apresentadas na Seção 2.4.
- No modo **Online**, abra as configurações do escravo EtherCAT (**SCA06\_SoftMotion**), na aba **General**, habilite a opção **Expert setting**.
- Abra a aba **CoE Online** e selecione a opção **Auto Update**, conforme a Figura 4.7.

# INFORMAÇÕES ADICIONAIS DA REDE ETHERCAT

Index/Subindex	Name	Flags	Type	Value
16#24DE:16#00	P1246 - User Parameter	RW	INT	0
16#24DF:16#00	P1247 - User Parameter	RW	INT	0
16#24E0:16#00	P1248 - User Parameter	RW	INT	0
16#24E1:16#00	P1249 - User Parameter	RW	INT	0
16#6040:16#00	Controlword	RW	UINT	0
16#6041:16#00	Statusword	RO	UINT	592
16#6060:16#00	Modes of operation	RW	SINT	8
16#6061:16#00	Modes of operation display	RO	SINT	8
16#6063:16#00	Position actual internal value	RO	DINT	38691
16#6064:16#00	Position actual value	RO	DINT	38689
16#6069:16#00	Velocity sensor actual value	RO	DINT	38689
16#606B:16#00	Velocity demand value	RO	DINT	0
16#606C:16#00	Velocity actual value	RO	DINT	0
16#6071:16#00	Target torque	RW	INT	0
16#6077:16#00	Torque actual value	RO	INT	-50
16#607A:16#00	Target position	RW	DINT	38691
16#6081:16#00	Profile velocity	RW	UDINT	0
16#6083:16#00	Profile acceleration	RW	UDINT	0
16#6084:16#00	Profile deceleration	RW	UDINT	0
16#6086:16#00	Motion profile type	RW	INT	0
16#6087:16#00	Torque slope	RW	UDINT	0
16#6088:16#00	Torque profile type	RW	INT	0
16#6081:16#00	Velocity offset	RW	DINT	0
16#60C2:16#00	Interpolation time period			---
16#60FF:16#00	Target velocity	RW	DINT	0
16#6502:16#00	Supported drive modes	RO	UDINT	0

Figura 4.7: CoE Online

Elementos de operação	Descrição
Read Objects	O diretório de objetos é lido uma vez.
Auto update	Os objetos são lidos em ciclos.
Offline from ESI file	A aba mostra o conteúdo do diretório de objetos da descrição do dispositivo.
Online from Device	A aba mostra o conteúdo do diretório de objetos do dispositivo (Não disponível para o SCA06).
Flags	<b>RO:</b> O valor é protegido contra escrita. <b>RW:</b> O valor pode ser modificado.
Value	É possível clicar duas vezes no campo de texto para editar esse valor. O novo valor será escrito diretamente no SCA06.

Tabela 4.1: Elementos CoE Online.

Através desta aba é possível ler e modificar alguns parâmetros internos do servoconversor SCA06.

Faça um teste modificando o parâmetro P1249, para o valor 15.

- Encontre o parâmetro **P1249 - User Parameter** na lista, clique duas vezes no campo **Value**, digite **15** e pressione **Enter**, como apresentado na Figura 4.8.



### NOTA!

As variáveis serão atualizadas em ciclos, aguarde a leitura dos parâmetros, isso pode levar alguns instantes.

The screenshot shows the 'Read Objects' window in the SCA06\_SoftMotion software. The window has a sidebar on the left with navigation options like 'General', 'Expert Process Data', 'Process Data', 'Startup Parameters', 'Online', 'CoE Online', 'Log', 'EtherCAT I/O Mapping', 'EtherCAT IEC Objects', 'Status', 'Information', and another 'Log' section. The main area contains a table with columns for 'Index:Subindex', 'Name', 'Flags', 'Type', and 'Value'. The table lists various parameters such as 'P1246 - User Parameter', 'Controlword', 'Statusword', and 'Modes of operation'. The value for 'P1249 - User Parameter' is highlighted as '15'.

Index:Subindex	Name	Flags	Type	Value
16#24DE:16#00	P1246 - User Parameter	RW	INT	0
16#24DF:16#00	P1247 - User Parameter	RW	INT	0
16#24E0:16#00	P1248 - User Parameter	RW	INT	0
16#24E1:16#00	P1249 - User Parameter	RW	INT	15
16#6040:16#00	Controlword	RW	UINT	0
16#6041:16#00	Statusword	RO	UINT	592
16#6060:16#00	Modes of operation	RW	SINT	8
16#6061:16#00	Modes of operation display	RO	SINT	8
16#6063:16#00	Position actual internal value	RO	DINT	38691
16#6064:16#00	Position actual value	RO	DINT	38691
16#6069:16#00	Velocity sensor actual value	RO	DINT	38691
16#606B:16#00	Velocity demand value	RO	DINT	0
16#606C:16#00	Velocity actual value	RO	DINT	0
16#6071:16#00	Target torque	RW	INT	0
16#6077:16#00	Torque actual value	RO	INT	-50
16#607A:16#00	Target position	RW	DINT	38691
16#6081:16#00	Profile velocity	RW	UDINT	0
16#6083:16#00	Profile acceleration	RW	UDINT	0
16#6084:16#00	Profile deceleration	RW	UDINT	0
16#6086:16#00	Motion profile type	RW	INT	0
16#6087:16#00	Torque slope	RW	UDINT	0
16#6088:16#00	Torque profile type	RW	INT	0
16#6081:16#00	Velocity offset	RW	DINT	0
16#60C2:16#00	Interpolation time period			---
16#60FF:16#00	Target velocity	RW	DINT	0
16#6502:16#00	Supported drive modes	RO	UDINT	0

**Figura 4.8:** Editar parâmetros Online

Com isso, o valor será modificado.

- Confirma a escrita deste valor diretamente pela IHM do **SCA06**, no parâmetro **P1249**.

### 4.3 EDITAR PDOS NA REDE ETHERCAT

É possível editar os **PDOS** definidos como padrão do **SCA06** na comunicação EtherCAT.

- Utilize as configurações apresentadas na Seção 2.4.
- Abra as configurações do **SCA06\_Motion**, na aba **General**, habilite a opção **Expert settings**, após isso diversas configurações avançadas ficarão disponíveis, como apresentado na Figura 4.9.

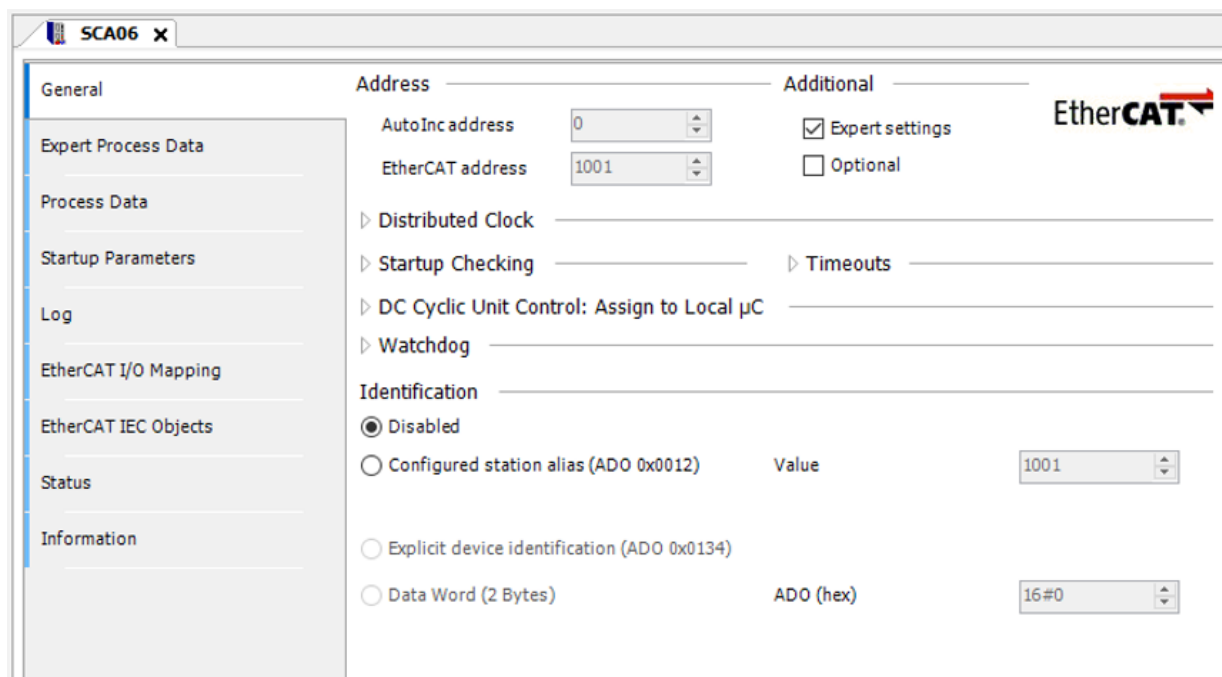


Figura 4.9: Habilitar configurações avançadas SCA06\_SoftMotion.

- Acesse a aba **Expert Process Data**, nesta aba será possível modificar os **PDOs** da comunicação.

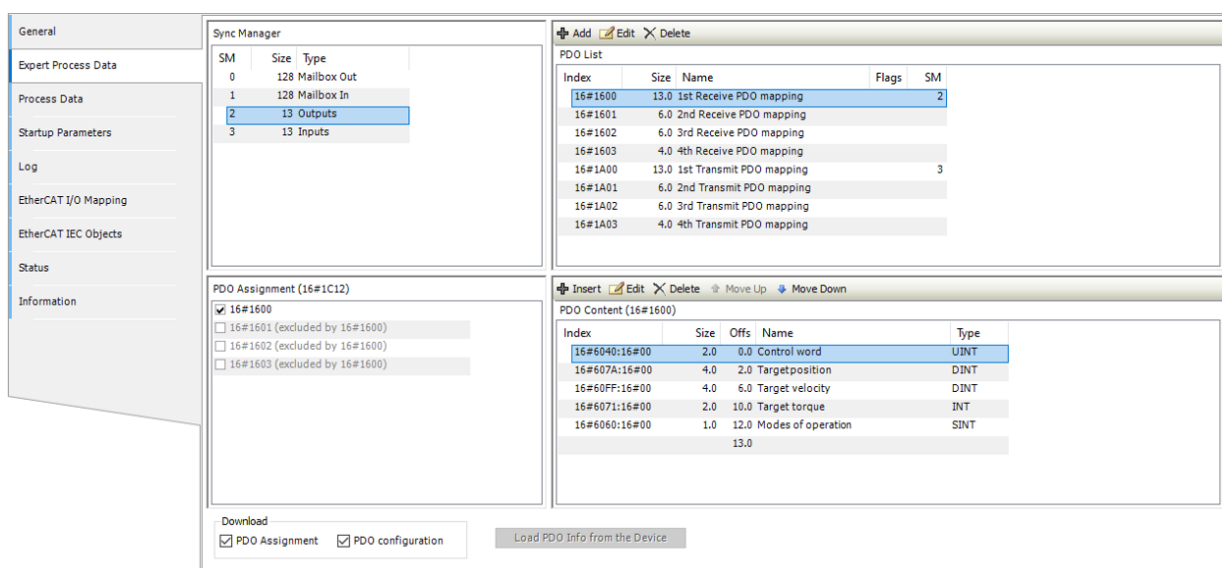


Figura 4.10: Editar PDOs da rede EterCAT.

- Selecione o **PDO** que você deseja modificar no campo **PDO List** e em seguida edite-o no campo **PDO Content**.
- Para aplicar a nova configuração de **PDOs** certifique-se que no campo **Download** as opções **PDO Assignment** e **PDO configuration** estejam selecionadas.

Utilizando este procedimento, ao fazer o **Download** do programa para o **PLC500MC** e iniciar a comunicação EtherCAT com o servoconversor SCA06, a lista será automaticamente modificada para a nova configuração de PDOs.

## 4.4 CONFIGURAR REDUNDÂNCIA ETHERCAT

É possível configurar uma rede EtherCAT com redundância utilizando o PLC500MC.

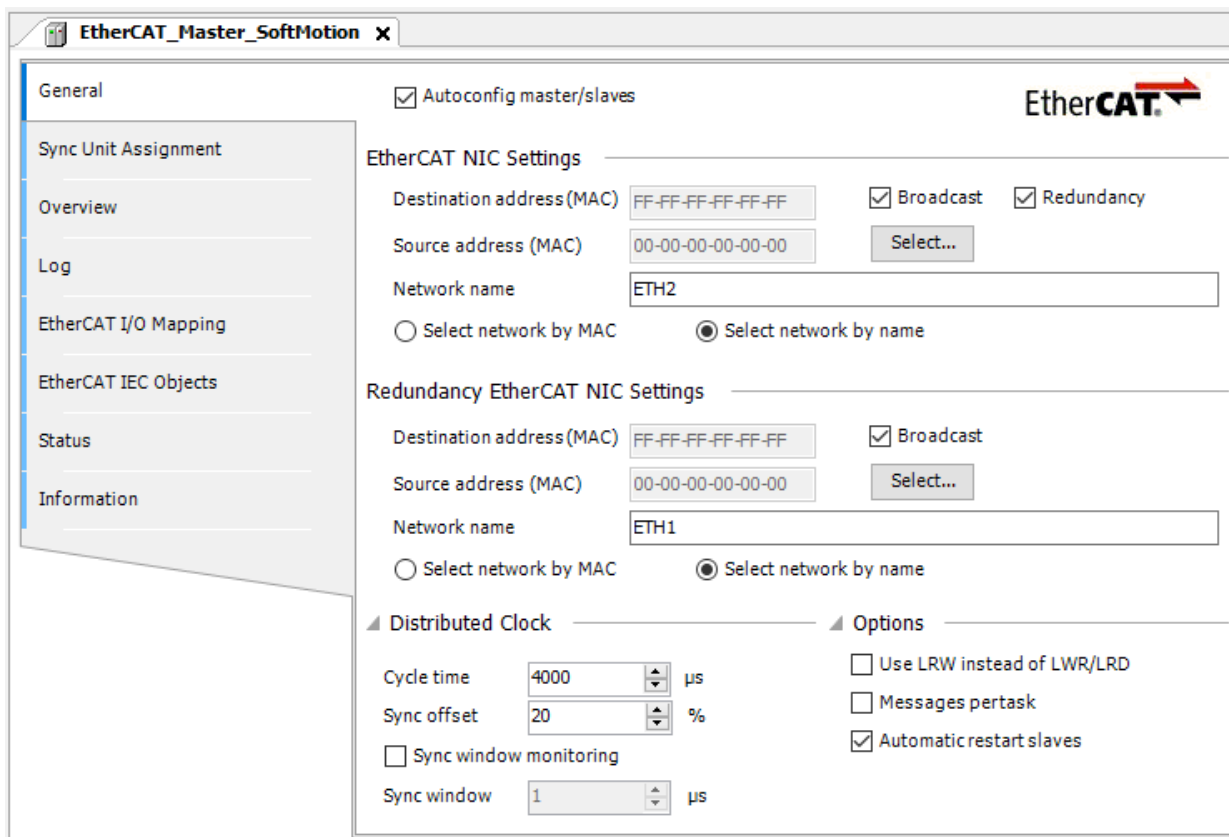


**NOTA!**

As portas **ETH1** e **ETH2** são portas independentes, desta forma, não é possível realizar uma comunicação EtherCAT em anel, entretanto, é possível realizar uma comunicação EtherCAT com redundância.

- Utilize as configurações apresentadas na Seção 2.4.
- Abra as configurações do dispositivo **EtherCAT Master SoftMotion**.
- Marque a opção **Redundancy**.
- No campo **Redundancy EtherCAT NIC Settings**, marque a opção **Select network by name**.
- No campo **Network name**, digite **ETH1**.

A Figura 4.11 apresenta as configurações anteriores já realizadas.



**Figura 4.11:** Configurar redundância EtherCAT.

Com isso, a redundância da rede já está configurada e pronta para ser utilizada.

## 4.5 ARQUIVO XML

Cada dispositivo em uma rede EtherCAT possui um arquivo de configuração XML que contém informações sobre o funcionamento do dispositivo na rede EtherCAT, bem como a descrição de todos os objetos existentes para comunicação. Em geral, este arquivo é utilizado por um mestre ou software de configuração para a programação dos dispositivos presentes na rede.



**ATENÇÃO!**

É possível adicionar dispositivos escravos EtherCAT ao software Codesys utilizando arquivos do tipo XML. Entretanto, para o controle de movimento é recomendado que sejam utilizados os dispositivos já instalados e específicos para este fim. Você poderá adicionar um eixo genérico seguindo a CiA402 porém, algumas funcionalidades SoftMotion podem não estar disponíveis.



## 5 INFORMAÇÕES ADICIONAIS SOFTMOTION

Neste capítulo serão apresentadas algumas informações adicionais e configurações avançadas utilizadas para o controle de movimento.

### 5.1 PRIORIDADE DE TAREFAS

O controle de movimento necessita de uma alta prioridade para seu correto funcionamento. Desta maneira, é imprescindível uma correta configuração da prioridade das tarefas.

Deve-se definir 1 para a tarefa responsável pelo controle de movimento. Quando adicionado um dispositivo mestre EtherCAT, ele criará uma tarefa (com prioridade 1) automaticamente. A aplicação responsável pelo controle de movimento deve ser executada sob esta tarefa.

Demais aplicações, além do controle de movimento, ou que possuem um alto consumo computacional devem ser executados em uma tarefa diferente e com uma prioridade menor. Recomenda-se a prioridade 10 ou menor (10 - 31) para estas tarefas.



**NOTA!**

Quanto menor o número maior será a sua prioridade, sendo 0 a tarefa mais prioritária e 31 a menos prioritária.

A Figura 5.1 apresenta um exemplo de configuração de tarefas, onde o controle de movimento é executado no programa **MyMotion** e as demais funcionalidades são executadas no programa **PLC\_PRG**.

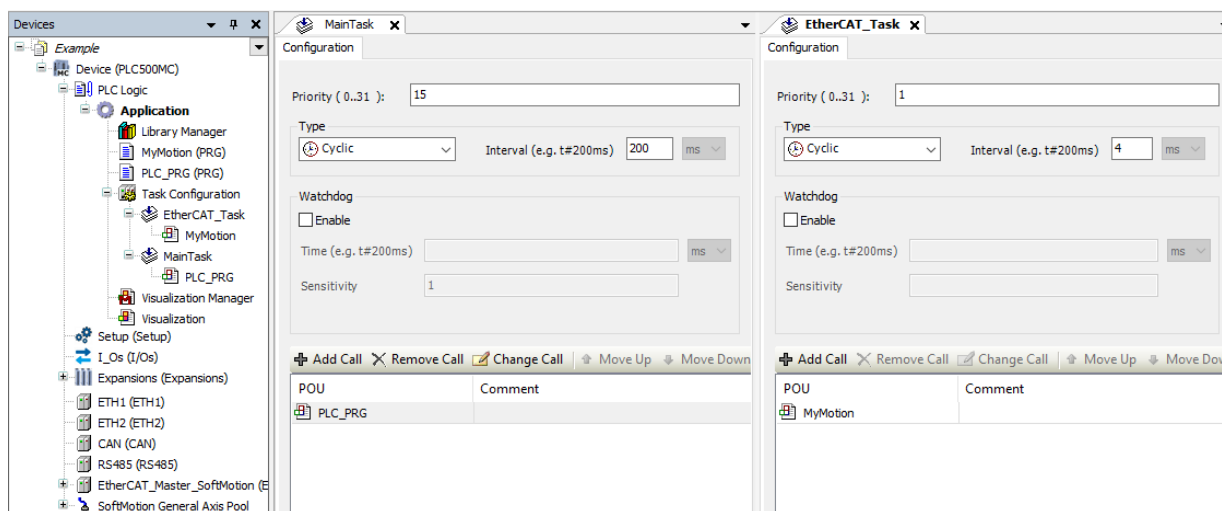


Figura 5.1: Editar prioridade de tarefas.

### 5.2 CONFIGURAÇÕES DE ESCALA PARA O SM\_DRIVE\_ETC\_WEG\_SCA

Nesta subseção será apresentada as possíveis configurações de escala aplicadas ao eixo **SM\_Drive\_ETC\_WEG\_SCA**.

- Utilize como base as configurações apresentadas na Seção 2.4.
- Abra as configurações do **SM\_Drive\_ETC\_WEG\_SCA**, na aba **Scaling/Mapping**.

Para as configurações de escala é possível utilizar dois tipos motores, escolhida através do campo **Motor Type**. Dependendo do tipo de motor selecionado o campo Scaling as configurações disponíveis serão diferentes.

#### 5.2.1 Motor Type: Rotary

**Rotary:** Geralmente utilizada para configurações de eixos rotativos pois possui uma configuração mais completa, podendo adicionar relação de engrenagens ou polias. A Figura 5.2 apresenta um exemplo de

configuração utilizando o **Motor Type: Rotary**.

Figura 5.2: Exemplo de configuração utilizando Motor Type: Rotary.

Cada valor do campo **Scaling** pode ser alterado de acordo com a mecânica envolvida na aplicação.

Elementos de operação	Descrição
increments <=> motor turns	Número de incrementos que correspondem a um determinado número de voltas do motor.
motor turns <=> gear output turns	Número de voltas do motor que correspondem a um determinado número de voltas na saída da engrenagem.
gear output turns <=> units in application	Número de voltas na saída da engrenagem que correspondem a unidades de aplicação.

Tabela 5.1: Elementos Scaling.

Para esta configuração cada unidade de aplicação equivalerá a 1/6 de volta do servomotor.

## 5.2.2 Motor Type: Linear

**Linear:** Geralmente utilizada para configurações de eixos lineares pois possui uma configuração mais simplificada e direta. A Figura 5.3 apresenta um exemplo de configuração utilizando o **Motor Type: Linear**.

Figura 5.3: Exemplo de configuração utilizando Motor Type: Linear.

Elementos de operação	Descrição
increments <=> units in application	Número de incrementos que correspondem a unidades de aplicação

Tabela 5.2: Elementos Scaling.

Para esta configuração cada unidade de aplicação equivalerá a 1 de volta do servomotor.



**NOTA!**

Ao selecionar a opção **Invert direction** o sentido de giro será invertido. O servoconversor receberá os valores de referência com sinais opostos.

Mais informações sobre as configurações podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > Reference > User Interface > Objects > SoftMotion Drives).

### 5.3 ADICIONAR EIXO VIRTUAL

Virtual Drive são unidades simuladas em software. Com isso, você pode testar seus programas sem um hardware conectado ou implementar funcionalidades estendidas utilizando eixos virtuais.

Para adicionar um eixo virtual em uma aplicação siga os passos apresentados a seguir.

- Clique com o botão direito em **SoftMotionGeneral axis pool** na árvore de dispositivos, selecione a opção **Add device**.
- Selecione o dispositivo **SoftMotionDrives > virtual drives > SM\_Drive\_Virtual** na caixa de diálogo **Add Device**.
- Clique em **Add Device**.

A Figura 5.4 apresenta os passos anteriores diretamente pelo Codesys.

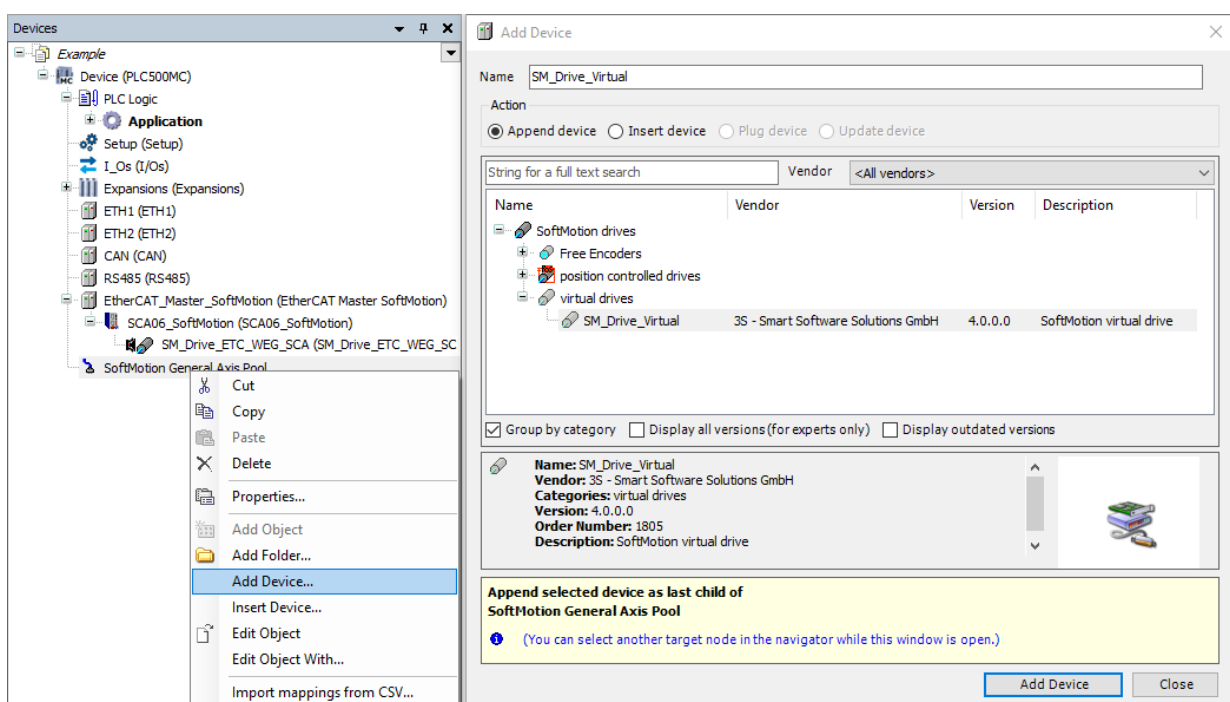


Figura 5.4: Adicionar eixo virtual.

Com isso, um eixo virtual será adicionado a baixo do objeto **SoftMotionGeneral axis pool**. A Figura 5.5 apresenta a árvore de dispositivos com um eixo virtual adicionado.

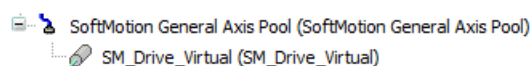


Figura 5.5: Árvore de dispositivos com eixo virtual adicionado.

- Abra as configurações do **SM\_Drive\_Virtual**.
- Na aba **General**, as configurações de tipo de eixo, limites, rampa de aceleração e dinâmica limite podem ser configuradas.
- Configure a aba **General**, de acordo com a Figura 5.6.

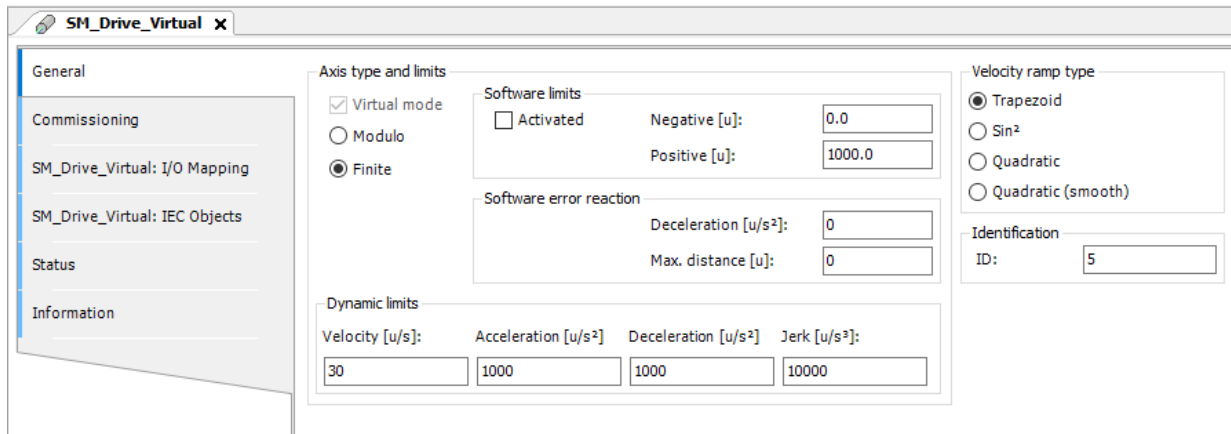


Figura 5.6: Exemplo configurações de eixo virtual.

Após essas configurações o eixo virtual poderá ser usado em suas aplicações.

Mais informações sobre eixos virtuais podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > Reference > User Interface > Objects > SoftMotion Drives > Tab 'Logical Axes').

## 5.4 ADICIONAR EIXO ENCODER

É possível utilizar as duas entradas do tipo encoder do **PLC500MC** como **Drives SoftMotion**. Para isso, configure a entrada **DI1** do **PLC500MC** como encoder (**I\_Os > DI1 / Encoder1 > Pin type > Pulse/Direction** ou **Quadrature**).

A Figura 5.7 apresenta os passos anteriores diretamente pelo Codesys.

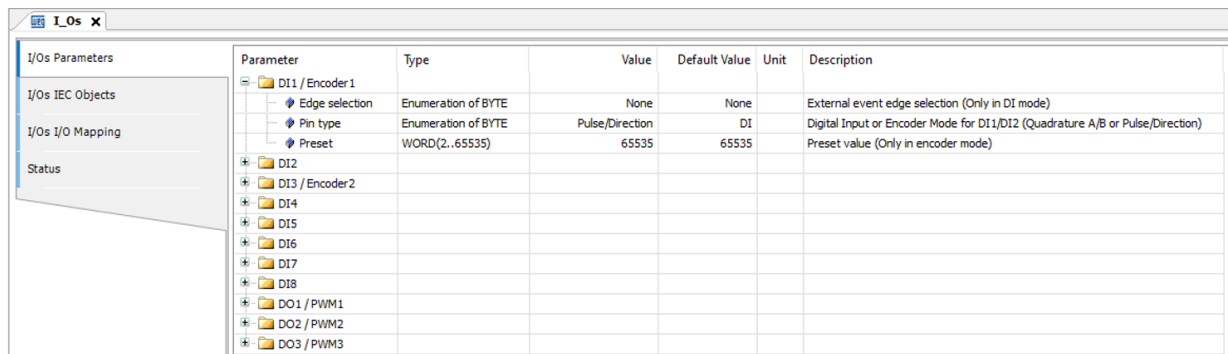


Figura 5.7: Configurar DI como encoder.

Desta forma as entradas **DI1** e **DI2** do **PLC500MC** deixam de ser entradas digitais e passam a ser entradas para encoder.

- Utilize como base as configurações apresentadas na Seção 3.
- Clique com o botão direito em **SoftMotion General Axis Pool** na árvore de dispositivos.
- Clique em **Add Device...**
- Na aba **Add Device**, no campo **Action**, selecione a opção **Append device**.
- Selecione o dispositivo **SoftMotion drives > Free Encoder > SMC\_FreeEncoder** na caixa de diálogo.
- Clique em **Add Device**.

A Figura 5.8 apresenta os passos anteriores diretamente pelo Codesys.

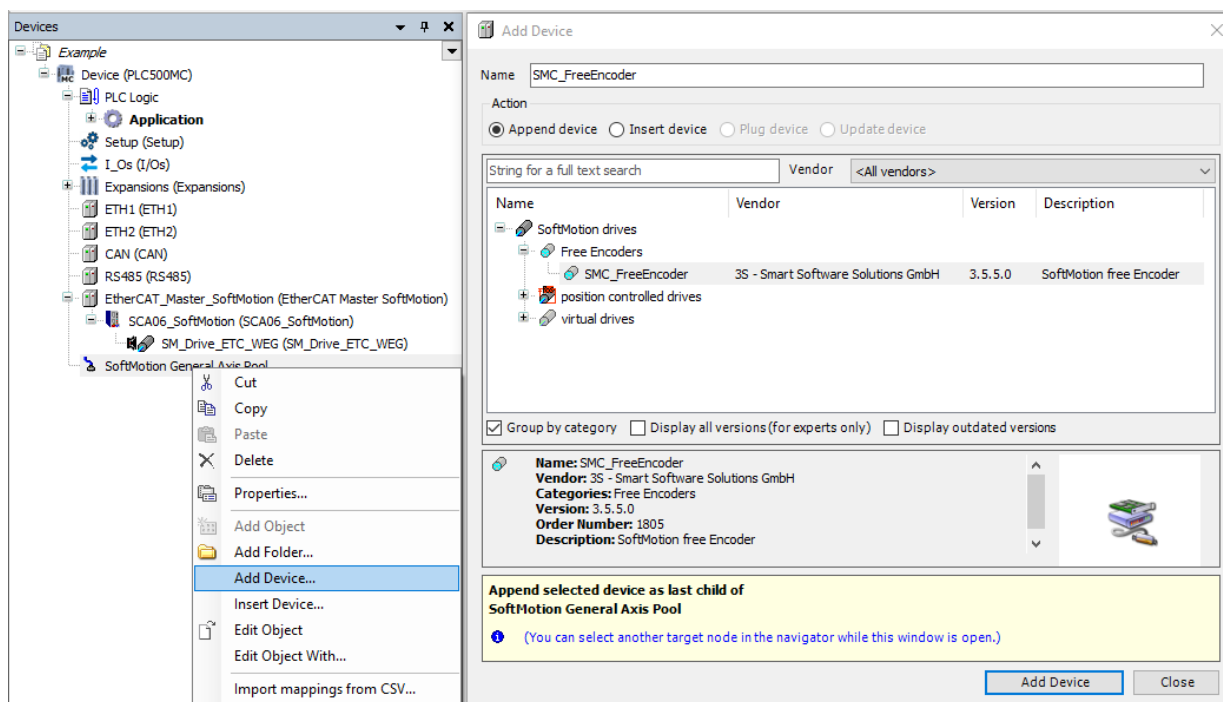


Figura 5.8: Adicionando eixo encoder.

Com isso, será adicionado à árvore de dispositivos em eixo do tipo **SMC\_FreeEncoder**, conforme a Figura 5.9.

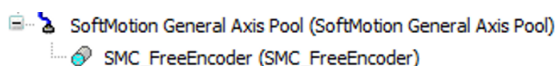


Figura 5.9: Drive Encoder adicionado à árvore de dispositivos.

- Abra as configurações do **SMC\_FreeEncoder**, na aba **Scaling**, faça a configuração adequada para o tipo de encoder utilizado em sua aplicação.

A Figura 5.10 apresenta um exemplo de configuração onde cada mil pulsos no encoder corresponderão a uma unidade de aplicação.

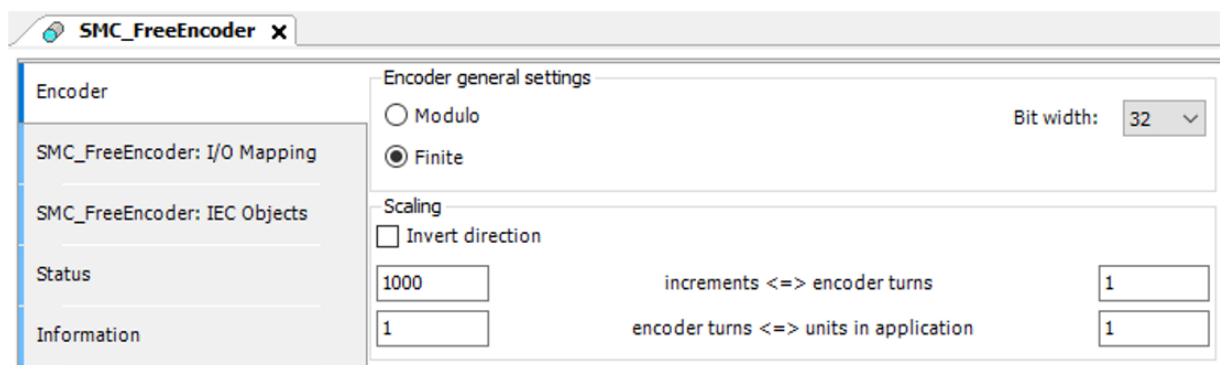


Figura 5.10: Drive Encoder adicionado à árvore de dispositivos.

Para que o valor atual do Drive adicionado seja atualizado com o valor da entrada de encoder do PLC500MC é necessário atribuir o seu valor para variável **<FREE\_ENCODER\_AXIS>.diEncoderPosition**, isso deve ocorrer na tarefa responsável pelo movimento (**EtherCAT\_Task**). É necessário ainda a conversão do tipo da variável de **LINT** para **DINT**, utilize a função **LINT\_TO\_DINT()** para isso.

O campo a baixo apresenta um exemplo do comando que deve ser utilizado para atribuir o valor de

## INFORMAÇÕES ADICIONAIS SOFTMOTION

**counter\_Encoder1** para a variável do drive **SMC\_FreeEncoder**.

```
SMC_FreeEncoder.diEncoderPosition := LINT_TO_DINT(counter_Encoder1);
```



### NOTA!

Também é possível utilizar os blocos de função disponíveis na biblioteca **IoDrvGPIO (WEG)** para atualizar os valores da posição do encoder.

- Abra o POU **MyMotion**, adicione o comando para atribuir o valor do encoder ao drive **SMC\_FreeEncoder**, como apresentado na Figura 5.11.

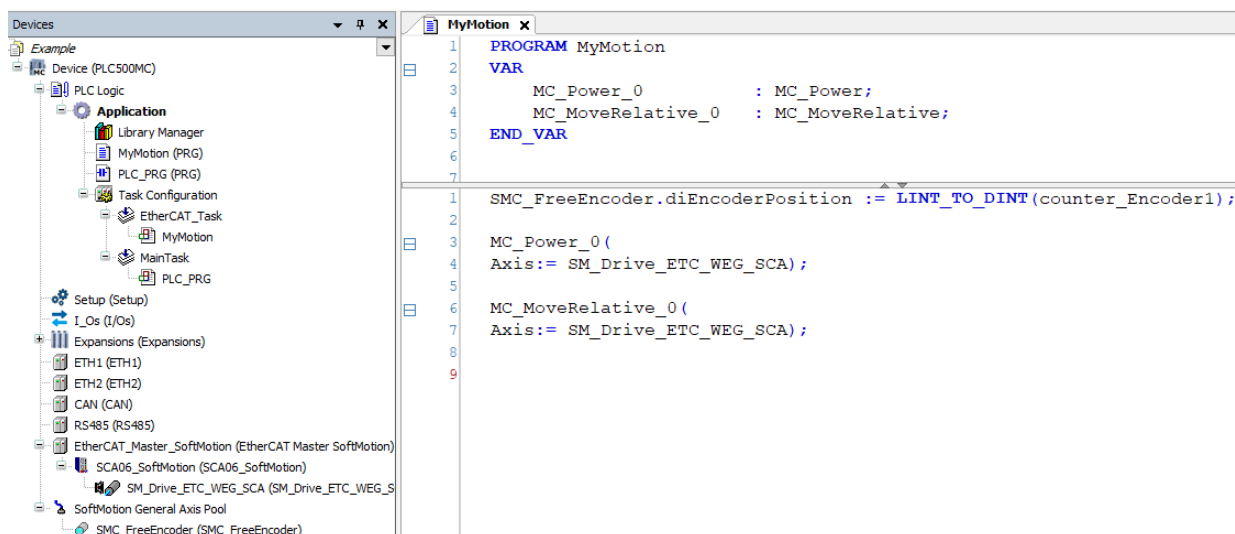


Figura 5.11: Adicionando o comando ao POU associado ao movimento.

Aplicação MyMotion:
<pre>PROGRAM MyMotion VAR   MC_Power_0      : MC_Power;   MC_MoveRelative_0 : MC_MoveRelative; END_VAR</pre>
<pre>SMC_FreeEncoder.diEncoderPosition := LINT_TO_DINT(counter_Encoder1);</pre>
<pre>MC_Power_0(   Axis:= SM_Drive_ETC_WEG_SCA);</pre>
<pre>MC_MoveRelative_0(   Axis:= SM_Drive_ETC_WEG_SCA);</pre>

- Conecte um encoder às entradas DI1 e DI2.
- Faça o download do programa para o PLC500MC e monitore no modo **Online**.
- Abra as configurações do **SMC\_FreeEncoder** na aba **Encoder**, como apresentado na Figura 5.12.

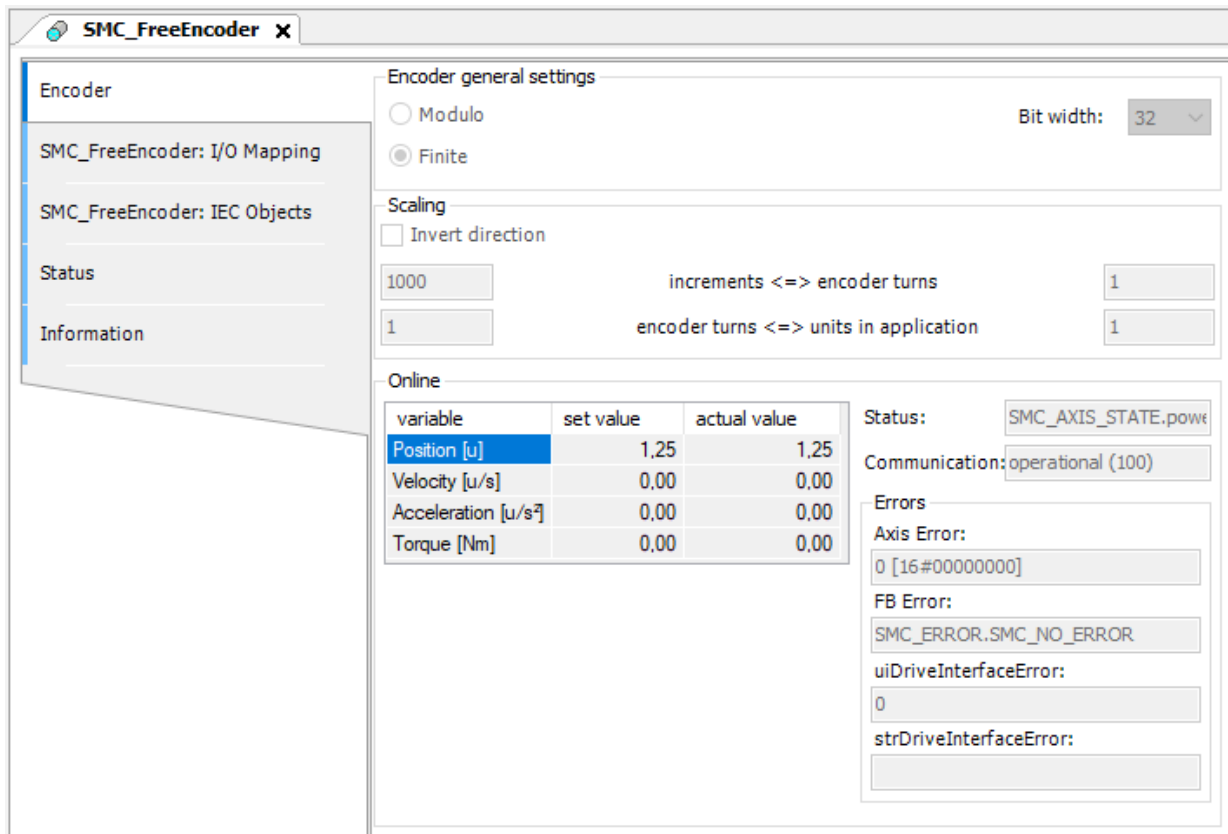


Figura 5.12: Monitorando encoder.

- Movimente o eixo encoder e observe o valor da posição alterando em **Position [u] - actual value**.

Mais informações sobre eixos de encoder podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > Reference > User Interface > Objects > SoftMotion Drives > Tab 'Encoder').

### 5.5 SINCRONIZAÇÃO CAME

Um came descreve a dependência funcional de movimento de uma unidade (escravo) em relação a outra unidade (mestre). A relação é descrita por uma função contínua (ou curva) que mapeia uma faixa definida de valores do mestre para valores do escravo.

#### 5.5.1 Criar aplicação came

Nesta subseção serão apresentadas as configurações necessárias e os blocos de função utilizados para executar um movimento came utilizando eixos virtuais.

- Crie um projeto novo em **File > New Project**. Selecione **Standard Project**, defina um diretório e o nome da aplicação (Example\_Cam). Selecione o dispositivo **PLC500MC** e a linguagem de programação **Contínuos Function Chart (CFC)**.
- Na árvore de dispositivos, clique com o botão direito no objeto **Application > Add Object > Cam table...**
- Na caixa de diálogo aberta, defina o nome como apresentado na Figura 5.13.
- Clique em **Add**.

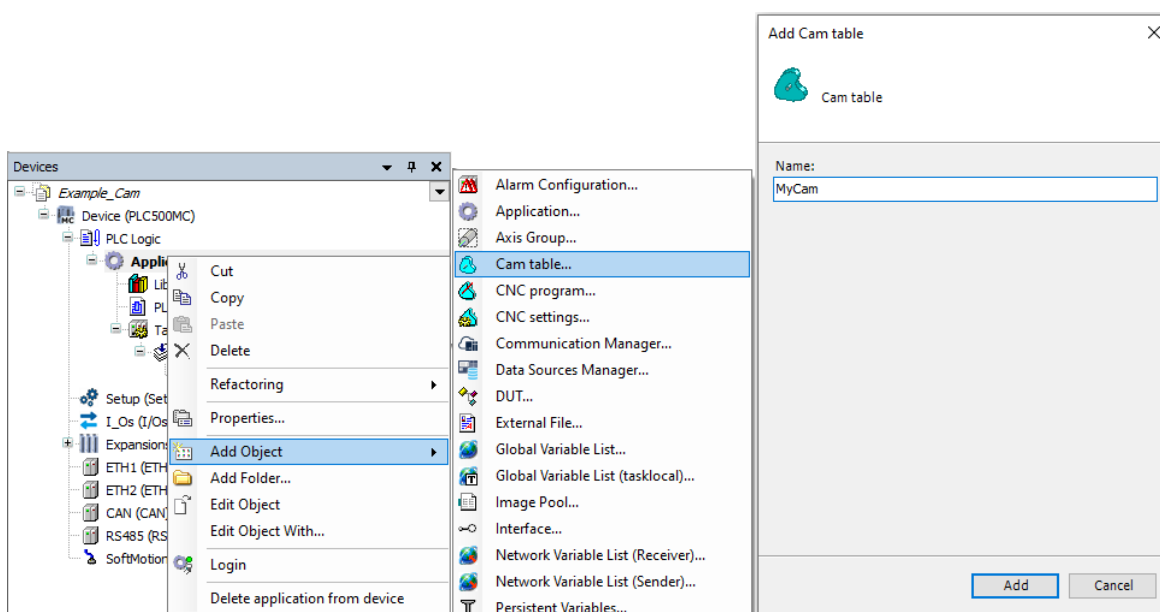


Figura 5.13: Criar tabela came.

- Abra o objeto **MyCam** criado anteriormente.

O software Codesys possui um editor gráfico came integrado que permite a criação e a edição rápida das tabelas comes.

Neste objeto, as tabelas de comes são definidas. Você pode alternar entre o editor gráfico (aba **cam**) e o editor de tabela alternativo (aba **cam table**) a qualquer momento.



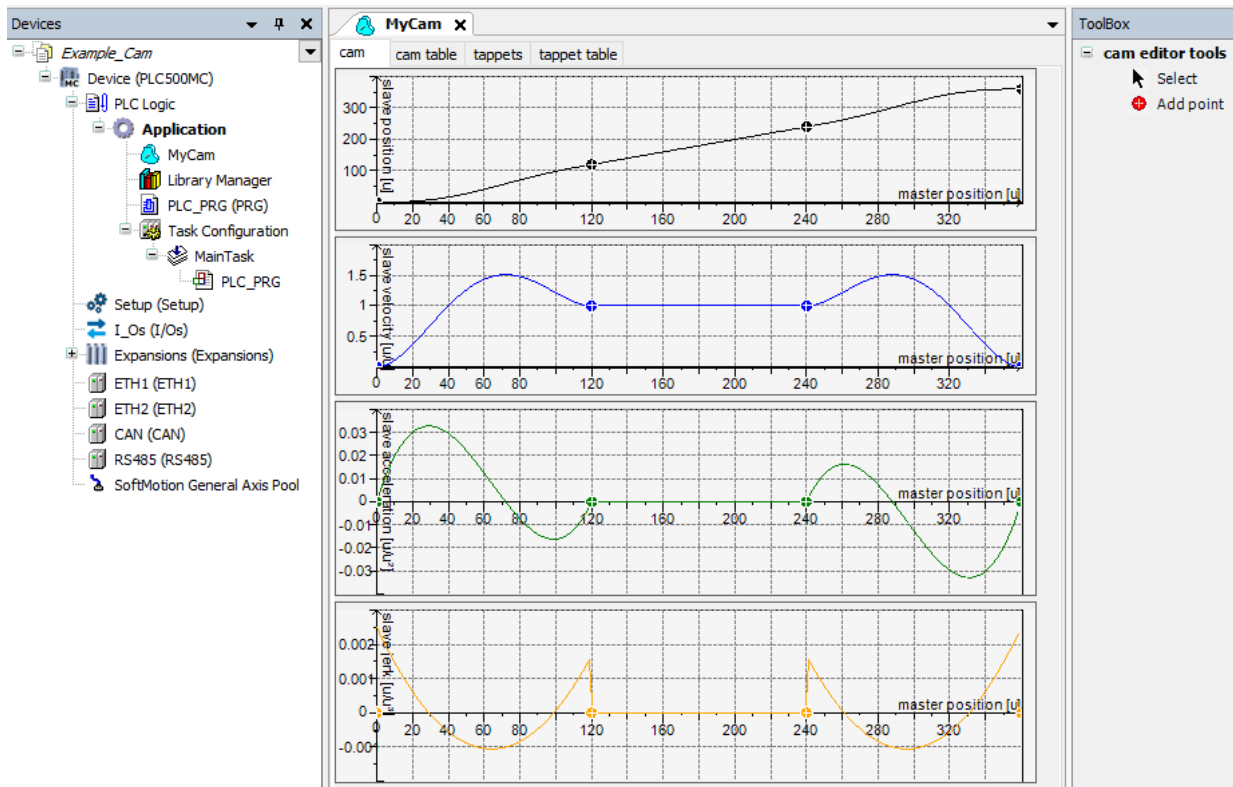


Figura 5.14: Editor came.

### 5.5.2 Importar tabela came

Além de criar uma tabela came, através do editor, também é possível importar e exportar estas tabelas.

- Para importar ou exportar uma tabela came, abra o objeto **MyCam**.

Com o objeto aberto, uma nova opção chamada **Cam** é habilitada no menu superior do software Codesys, nesta aba ficam localizadas as opções para importar e exportar tabelas came.

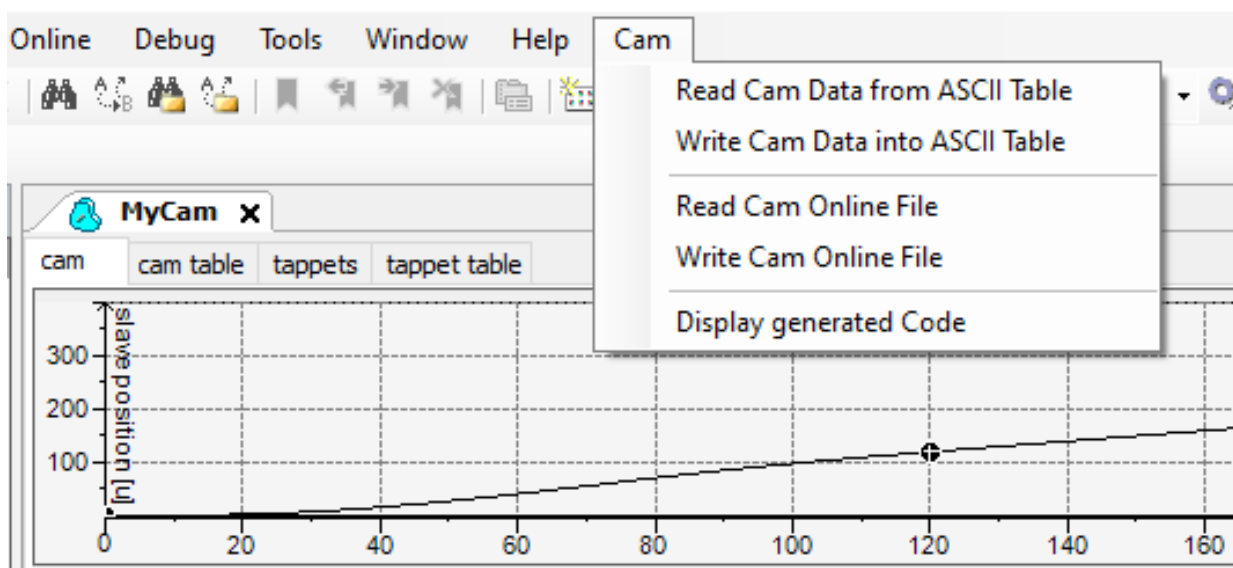


Figura 5.15: Importar/exportar tabelas came.

Mais informações sobre tabelas came podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > Reference > User Interface > Objects > Object 'Cam Table').

## 5.5.3 Executar tabela came

Para executar uma tabela came é necessário configurar os eixos que farão parte do movimento.

- Adicione dois eixos virtuais nessa aplicação (**Axis\_A** e **Axis\_B**), como apresentado na Subseção 5.3.
- Configure a aba **General** de ambos os eixos criados conforme a Figura 5.16.

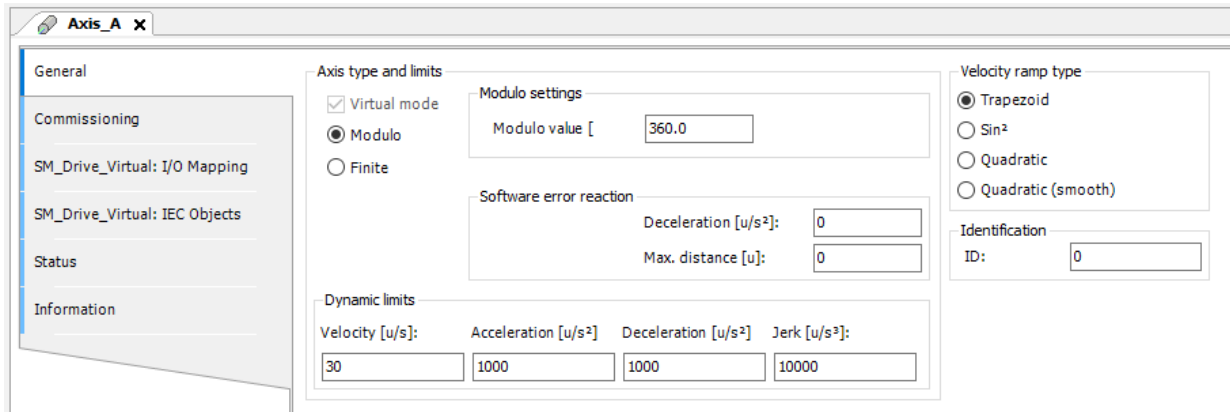


Figura 5.16: Configurações tarefa came.

- Modifique a prioridade da tarefa **MainTask** para 1 e a defina com intervalo cíclico de 4ms.

A Figura 5.17 apresenta as configurações da tarefa e os objetos já adicionados.

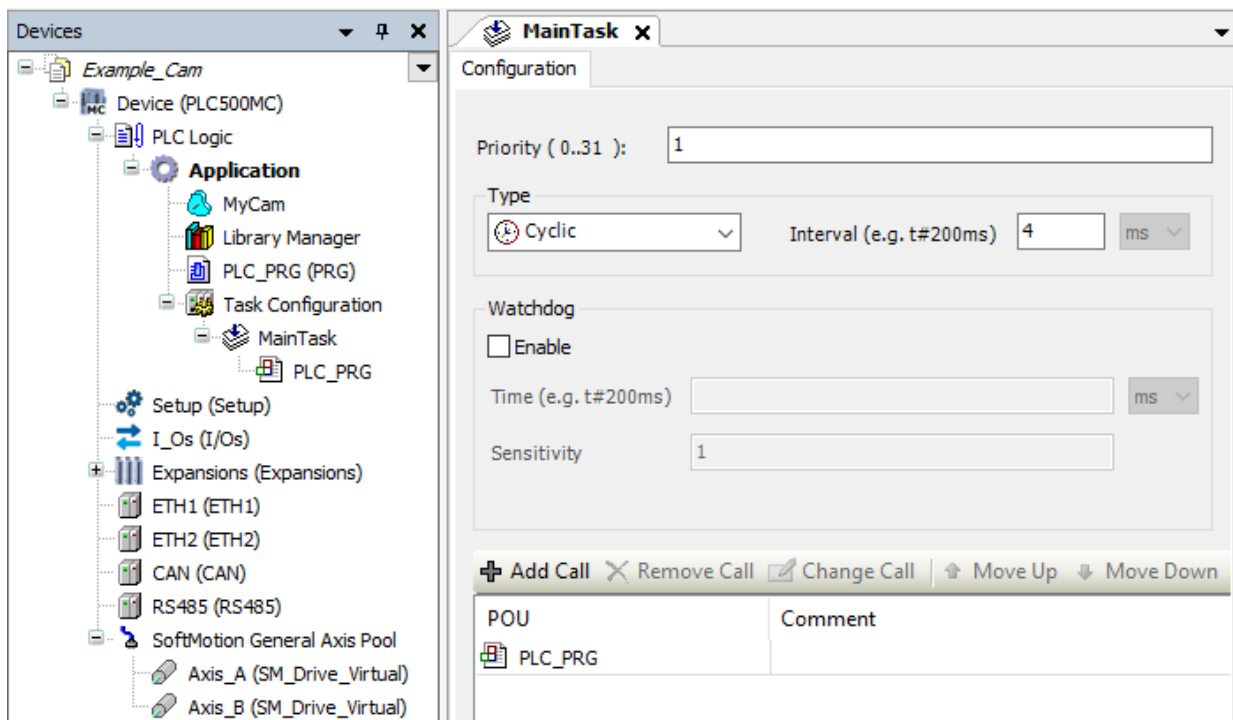


Figura 5.17: Configurações tarefa came.

O programa padrão para executar uma tabela came é apresentado na Figura 5.18.

- Abra as configurações do programa **PLC\_PRG(PRG)**.
- Declare as instancias dos blocos de função e faça as ligações dos blocos como apresentado na Figura 5.18

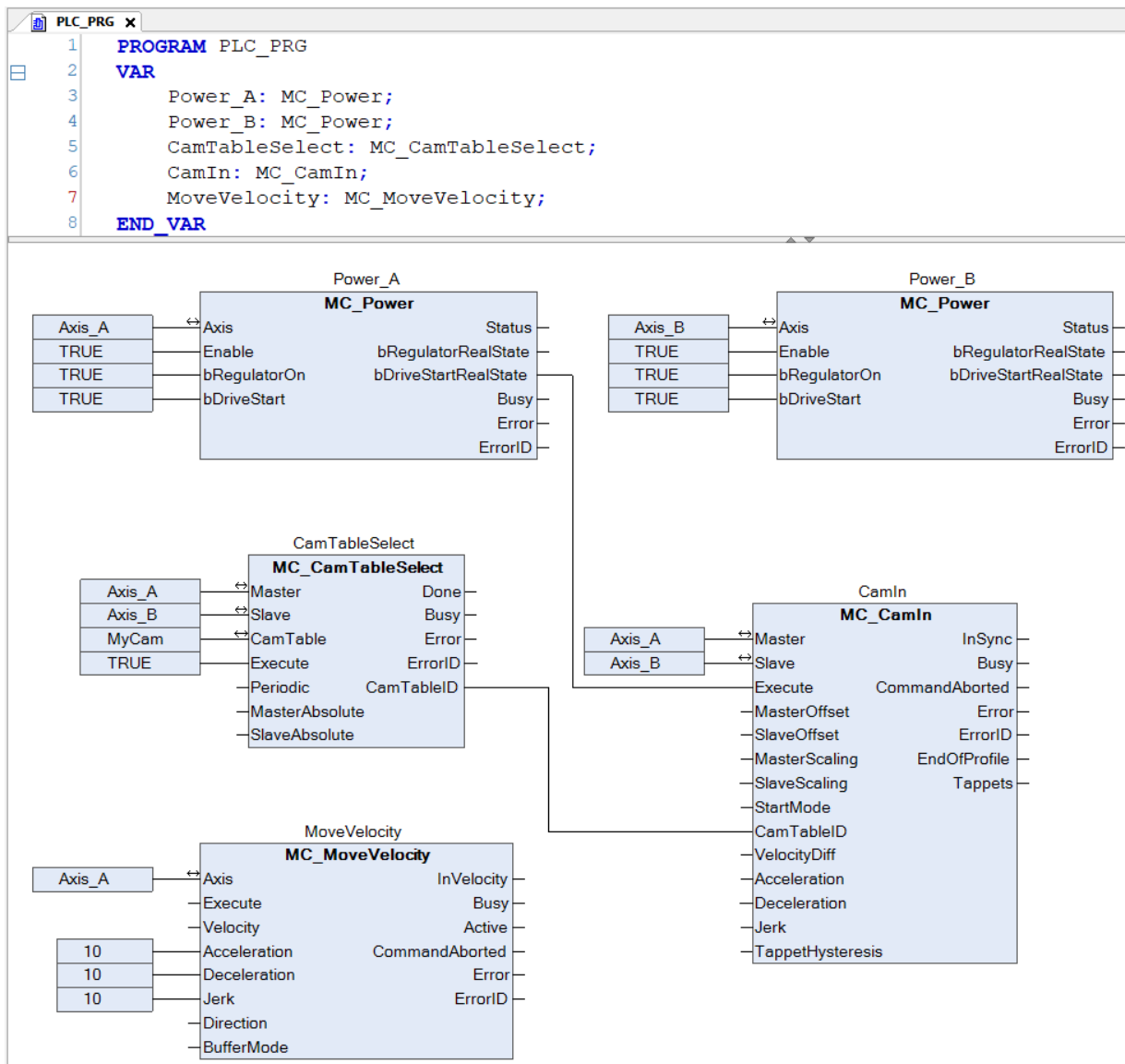


Figura 5.18: Programa para executar tabelas came.



**NOTA!**

O Apêndice A apresenta este mesmo programa utilizando a linguagem ST.

A seguir, serão apresentadas algumas informações referente a cada bloco do programa e suas conexões.

Os blocos de função do tipo **MC\_Power** são responsáveis por habilitar os eixos.

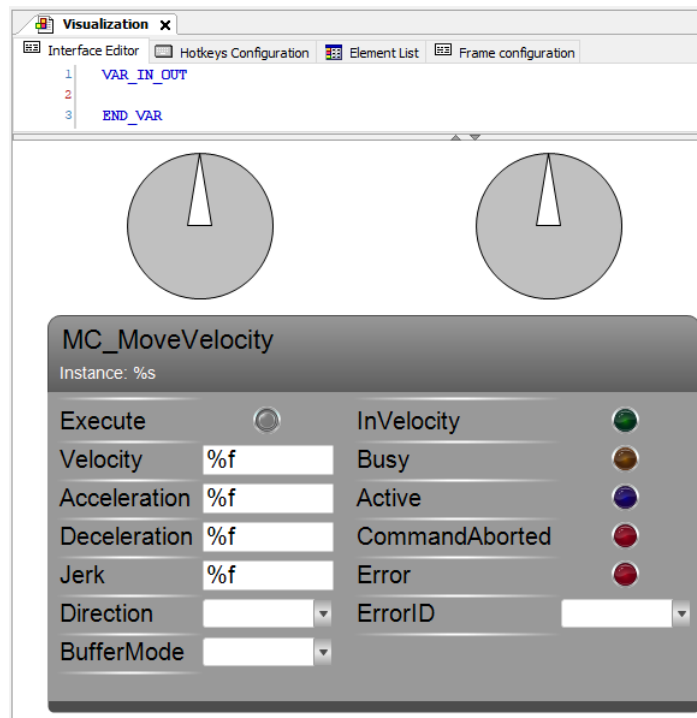
O bloco de função **MC\_CamTableSelect** seleciona a tabela came a ser executada. A entrada **CamTable** deve referenciar a tabela came da árvore de dispositivos e a saída **CamTableID** deve estar conectada a entrada **CamTableID** do bloco de função **MC\_CamIn**

O bloco de função **MC\_CamIn** implementa a tabela de came selecionada.

O bloco de função **MC\_MoveVelocity** controla a velocidade do eixo mestre.

- Crie um objeto do tipo **Vizualization**.
- Adicione e referencie o modelo de visualização do tipo **VISU\_NEW\_MC\_MoveVelocity** ao bloco de função **MC\_MoveVelocity**.
- Adicione e referencie um modelo de visualização do tipo **RotDrive** para cada eixo **Axis\_A** e **Axis\_B**.

A Figura 5.19 mostra o objeto **Visualization** com os modelos adicionados.



**Figura 5.19:** Visualização came.

- Faça o download do programa para o **PLC500MC**.
- No modo de monitoração **Online**, abra o objeto **Visualization**.
- No modelo de visualização do **VISU\_NEW\_MC\_MoveVelocity** selecione a velocidade de giro para o eixo mestre e clique em **Execute**.
- O movimento dos eixos podem ser observados pelos modelos de visualização **RotDrive**.

Modifique a tabela came pelo editor e realize mais alguns testes.

Outros exemplos de aplicação utilizando tabelas came podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > Application Examples).

## 5.6 INTERPRETAR E EXECUTAR ARQUIVOS CNC

O PLC500MC possui a capacidade de interpretação para Código-G (de acordo com a DIN 60025) utilizando o editor CNC 3D presente no software Codesys.

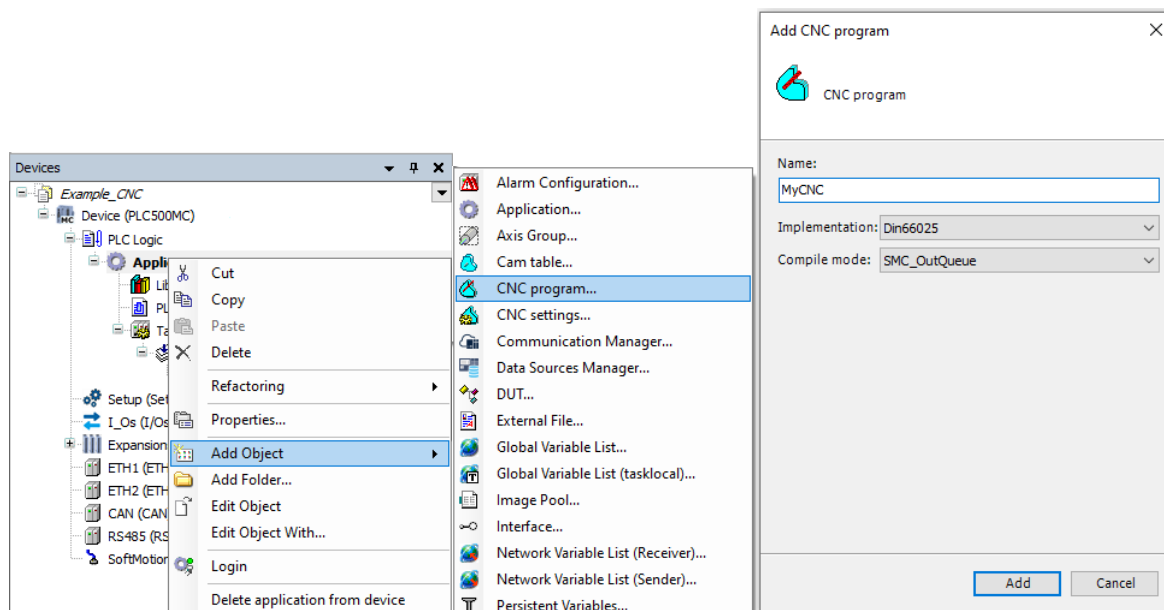
### 5.6.1 Abrangência dos comandos (G-Code) suportados

- Posicionamento rápido (G0).
- Interpolação linear (G1), interpolação circular (G2/G3).
- Temporização (G4).
- Interpolação helicoidal (G5, G10).
- Interpolação parabólica (G6), interpolação elíptica (G8, G9).
- Seleções de plano de interpolação para arcos circulares (G16 - G19).
- Saltos condicionais (G20).
- Gravação/incremento de variável IEC (G36, G37).
- Compensação do raio da ferramenta (G40 - G42).
- Arredondamento e suavização de ângulos (G50, G51, G52).
- Deslocamento do sistema de coordenadas (G53 a G56).
- Supressão de loop (G60, G61).
- Sincronização de tempo com interpolador (G75).
- Coordenadas absolutas e relativas (G90, G91).
- Configuração de posição (G92).
- Coordenadas absolutas e relativas (G98, G99).
- Funções M (M), Tuchos de caminho (H).
- Definição de velocidade e aceleração (F, E).
- Dimensões suportadas: X, Y, Z (eixos de interpolação primários).
- A, B, C (eixos de orientação).
- P, Q, U, V, W (eixos adicionais).

### 5.6.2 Criar aplicação CNC

Nesta subseção serão apresentadas as configurações necessárias e os blocos de função utilizados para executar um caminho CNC para uma planta do tipo pórtico 2D.

- Crie um projeto novo em **File > New Project**. Selecione **Standard Project**, defina um diretório e o nome da aplicação (Example\_CNC). Selecione o dispositivo **PLC500MC** e a linguagem de programação **Continuous Function Chart (CFC)**.
- Na árvore de dispositivos, clique com o botão direito no objeto **Application > Add Object > CNC program...**
- Na caixa de diálogo aberta, configure-a como apresentado na Figura 5.20.
- Clique em **Add**.



**Figura 5.20:** Criar programa CNC.

Ao adicionar o objeto, observe na árvore de dispositivos que além do programa CNC (MyCNC) é adicionado um objeto chamado **CNC Settings**. As configurações deste objeto são válidas para todos os objetos CNC da aplicação. Nas configurações do **CNC Settings**, podem ser especificadas configurações para os módulos de pré-processamento de trajetória, pré-interpolação e editor de tabelas CNC.

As configurações de pré-processamento disponíveis são apresentadas na Tabela 5.3.

Bloco de função	Descrição
<b>SMC_CheckVelocities</b>	Reduz a velocidade a zero caso existam curvas fechadas.
<b>SMC_AvoidLoop</b>	Desconsidera <i>loop</i> no código.
<b>SMC_ExtendedVelocityChecks</b>	Verifica a velocidade dos eixos adicionais.
<b>SMC_LimitCircularVelocity</b>	Limita a velocidade em movimentos circulares.
<b>SMC_ObjectSplitter</b>	Divide uma curva em vários pontos.
<b>SMC_RotateQueue2D</b>	Rotaciona o caminho 2D no plano.
<b>SMC_RoundPath</b>	Arredonda cantos utilizando arcos circulares.
<b>SMC_ScaleQueue3D</b>	Ajusta o fator de escala do caminho.
<b>SMC_SmoothAddAxes</b>	Suaviza movimentos dos eixos adicionais.
<b>SMC_SmoothPath</b>	Suaviza as arestas de um determinado caminho.
<b>SMC_SmoothMerge</b>	Aproxima um número pontos por um polinômio.
<b>SMC_ToolCorr   SMC_ToolRadiusCorr</b>	Corrige do raio da ferramenta.
<b>SMC_TranslateQueue3D</b>	Desloca o caminho em X, Y e Z.
<b>SMC_SmoothBSpline</b>	Suaviza segmentos de elementos <b>G1</b> consecutivos com um <b>B-Spline</b> de quinto grau.
<b>SMC_RecomputeABCSlopes</b>	Recalcula as inclinações dos eixos adicionais A,B,C, para executar um movimento suave.
<b>SMC_ReduceVelEndAtCorner</b>	Reduz a velocidade final se houver uma aresta entre dois elementos de caminho consecutivos.

**Tabela 5.3:** Descrição dos blocos de função de pré-processamento.

Mais informações sobre as configurações de pré-processamento de trajetória do objeto **CNC Settings** podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Libraries > SM3\_CNC Library Documentation > SM\_CNC\_POUs > SoftMotion CNC > SoftMotion Function Blocks).

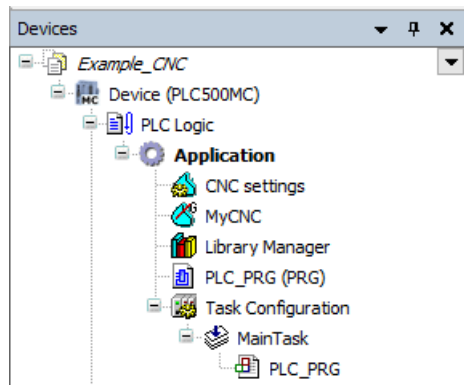


Figura 5.21: Árvore de dispositivos CNC.

- Abra as configurações do programa CNC (MyCNC).
- No editor CNC, escreva os comandos da Figura 5.22.

Observe que ao digitar os comandos, o caminho CNC será apresentado no editor gráfico.

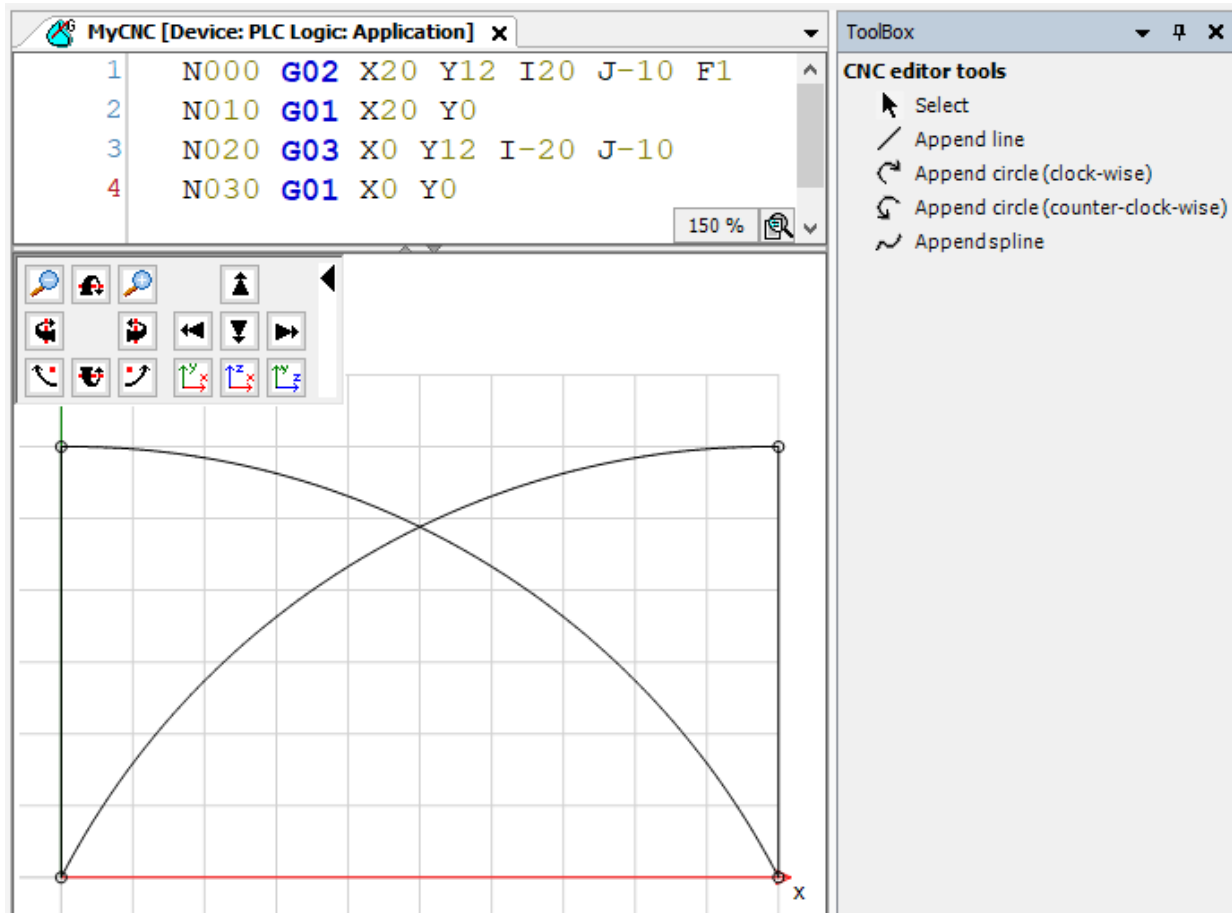


Figura 5.22: Programa CNC Básico.

<b>Código G utilizado:</b>
N000 G02 X20 Y12 I20 J-10 F1
N010 G01 X20 Y0
N020 G03 X0 Y12 I-20 J-10
N030 G01 X0 Y0

### 5.6.3 Importar arquivos CNC

Além de criar um caminho CNC, através do editor, também é possível importar arquivos do tipo DXF ou ASCII (.cnc, .gcode, .txt).

- Para importar um arquivo, abra o objeto **CNC** (MyCNC) na árvore de dispositivos.

Com o objeto aberto, uma nova opção chamada **CNC** é habilitada no menu superior do software Codesys, nesta aba ficam localizadas as opções para importar arquivos.

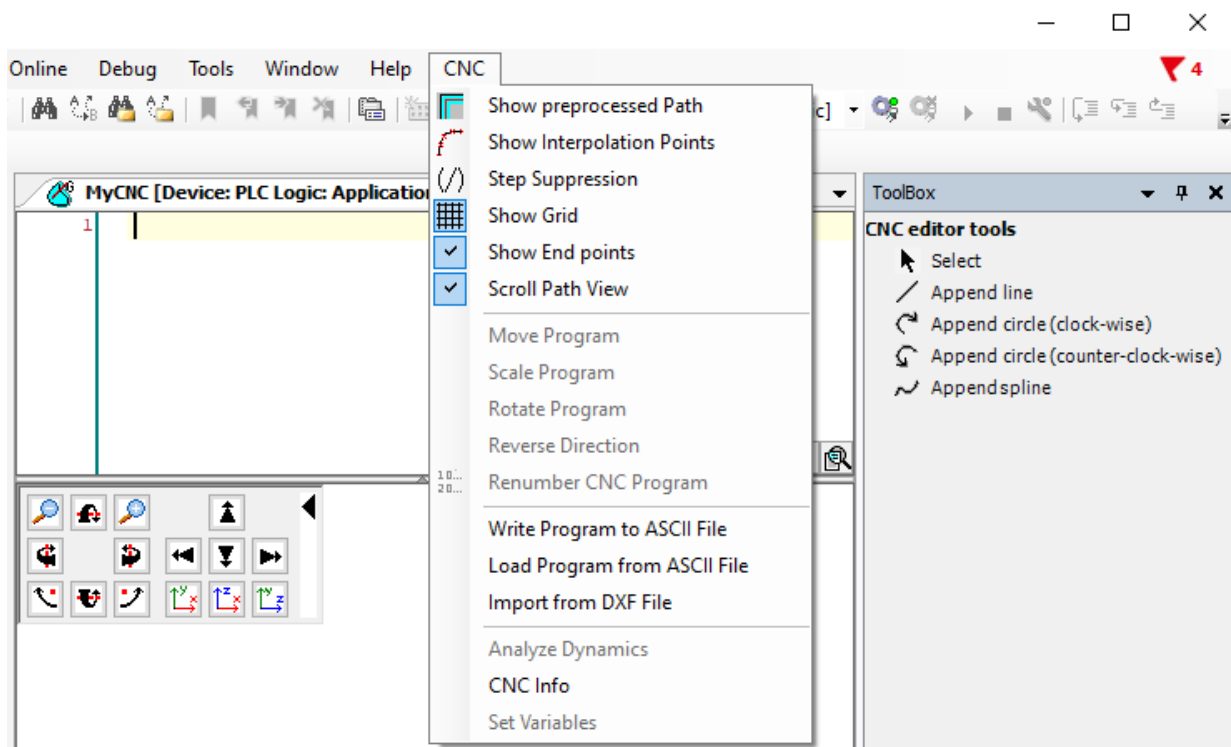


Figura 5.23: Importar/exportar caminhos CNC.

- Clique em **Import from DXF File** ou **Load Program from ASCII File** e selecione o arquivo.

Com isso, o arquivo será importado e poderá ser visualizado no editor gráfico, como na Figura 5.24.



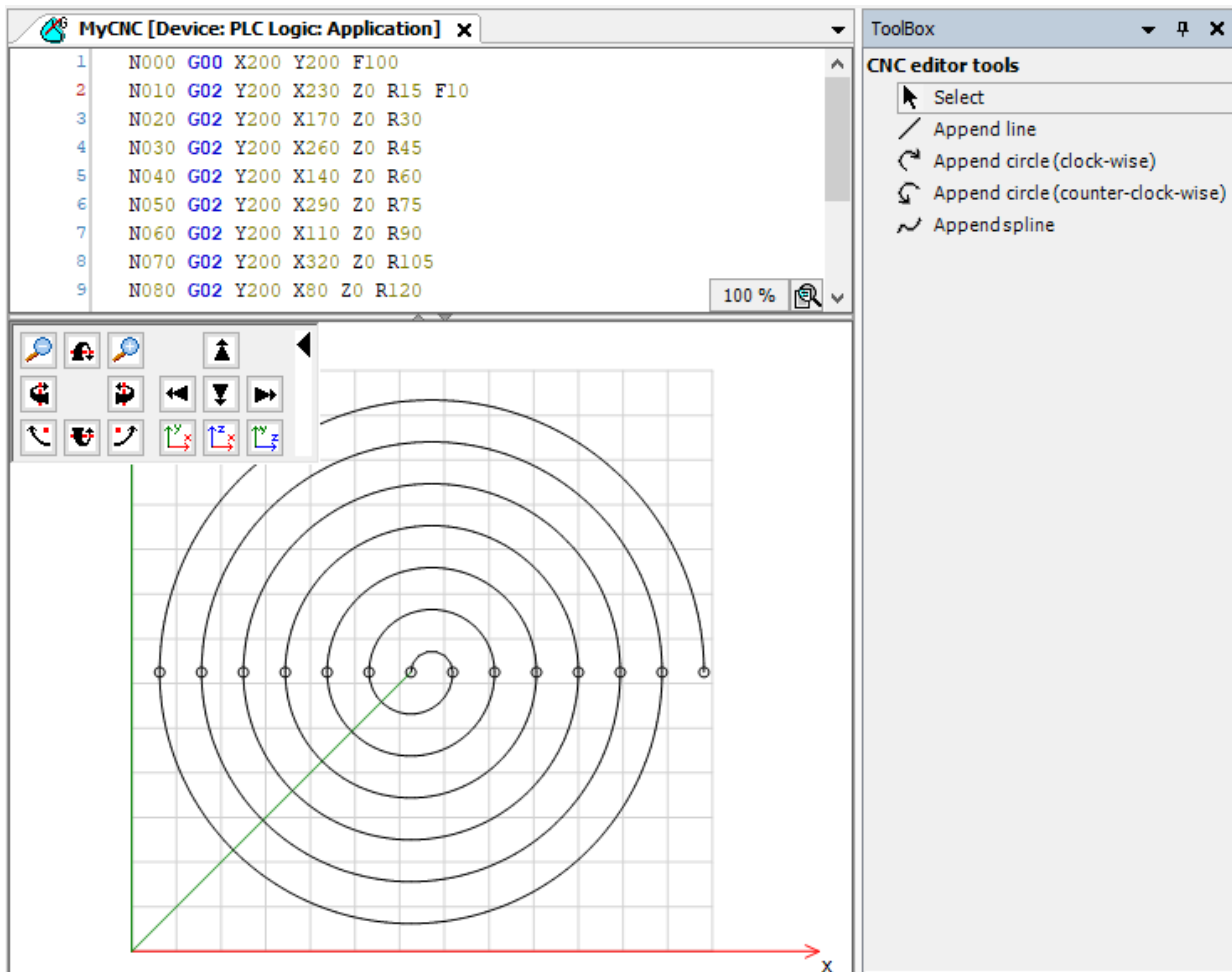


Figura 5.24: Caminho CNC importado.



**ATENÇÃO!**

As unidades utilizadas no caminho CNC são unidades de aplicação, realize uma correta configuração das escalas para os eixos.

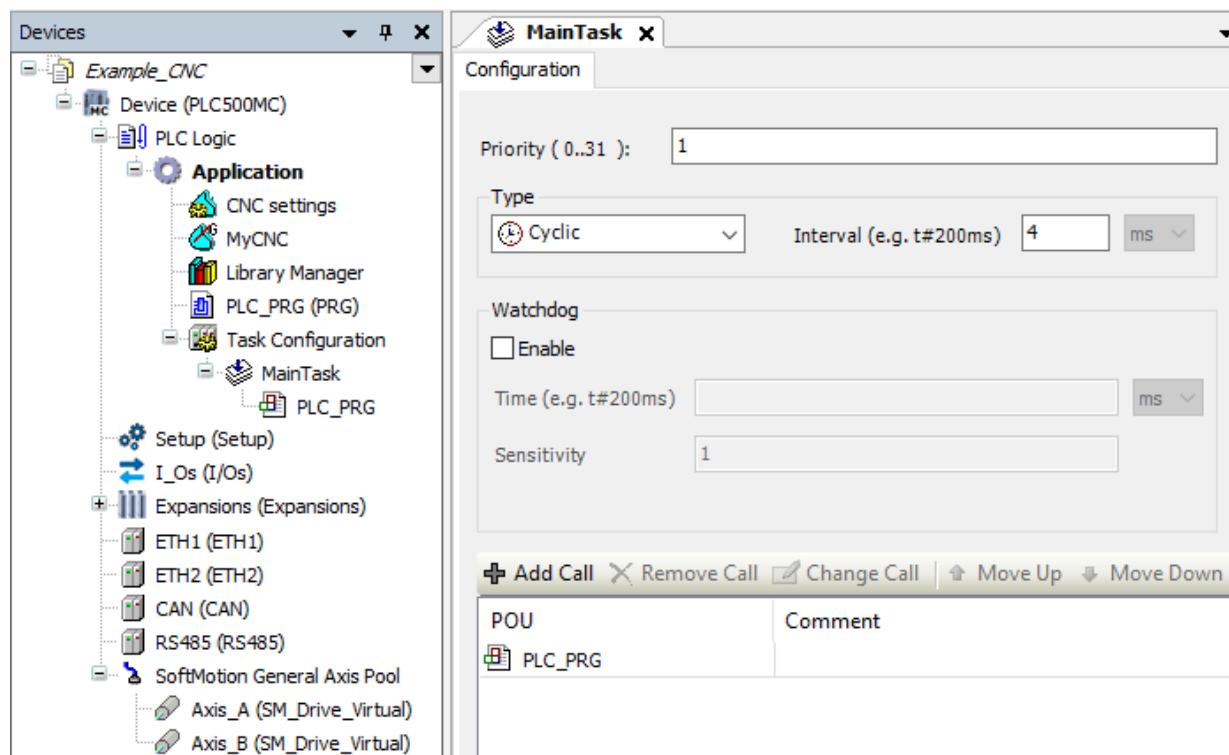
Mais informações sobre arquivos CNC podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > Reference > User Interface > Commands > CNCCommand).

**5.6.4 Executar caminho CNC**

Para executar um caminho CNC é necessário configurar os eixos que farão parte do movimento.

- Adicione dois eixos virtuais nessa aplicação (**Axis\_A** e **Axis\_B**), como apresentado na Subseção 5.3.
- Modifique a prioridade da tarefa **MainTask** para 1 e a defina com intervalo cíclico de 4ms.

A Figura 5.25 apresenta as configurações da tarefa e os objetos já adicionados.



**Figura 5.25:** Configurações tarefa CNC.

O programa padrão para executar um caminho CNC controlando um sistema do tipo pórtico 2D é apresentado na Figura 5.26.

- Na árvore de dispositivos, abra o programa **PLC\_PRG(PRG)**.
- Declare as instancias dos blocos de função e faça as ligações dos blocos como apresentado na Figura 5.26

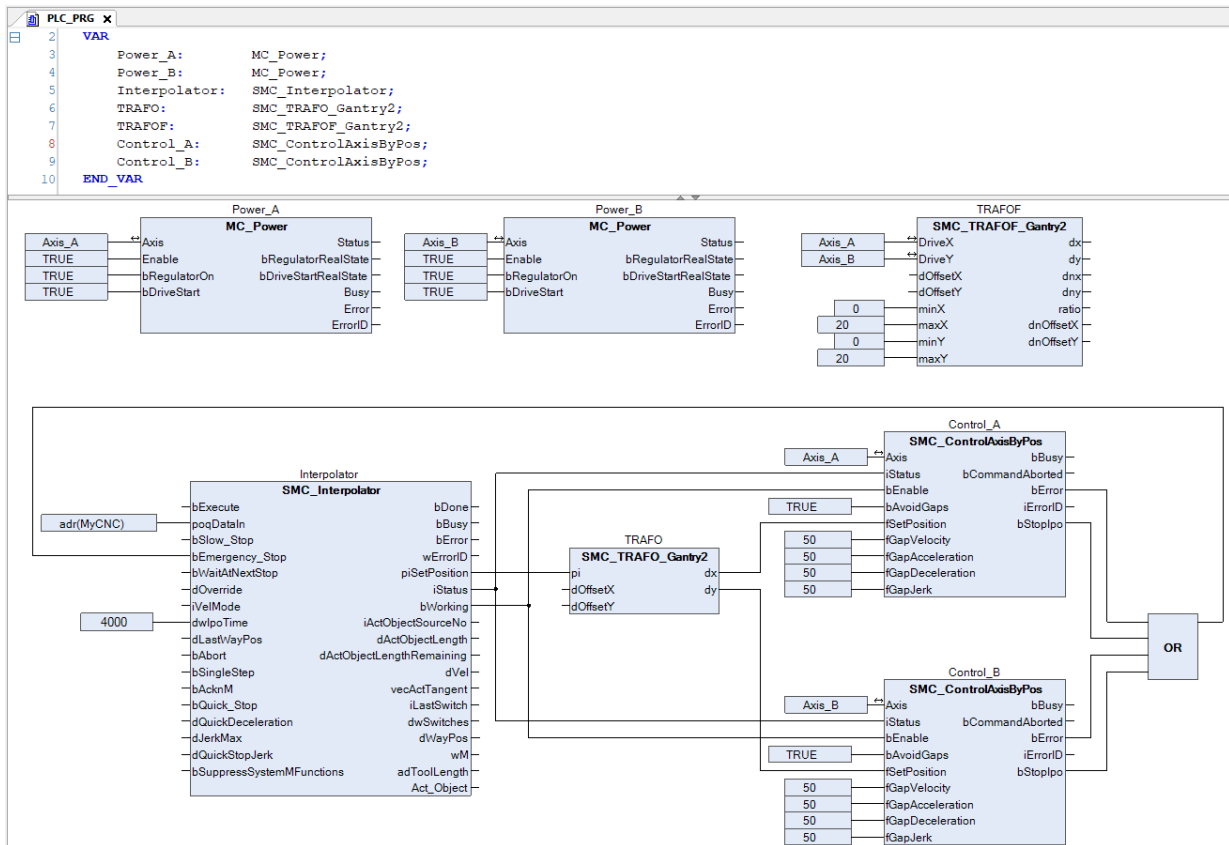


Figura 5.26: Programa para executar caminhos CNC.



**NOTA!**

O Apêndice B apresenta este mesmo programa utilizando a linguagem ST.

A seguir, serão apresentadas algumas informações referente a cada bloco do programa e suas conexões.

Os blocos de função do tipo **MC\_Power** são responsáveis por habilitar os eixos.

O bloco de função **SMC\_Interpolator** converte um caminho definido por objetos GEOINFO em pontos de caminho discretos. O bloco de função recebe o endereço do programa CNC na entrada **poqDataIn** e o tempo de ciclo da tarefa IEC em que ele será executado na entrada **dwlpoTime**.

O bloco de função do tipo **SMC\_TRAFOF\_Gantry2** corresponde a transformada direta do sistema **Gantry2** e é necessário apenas para visualização.

O bloco de função do tipo **SMC\_TRAFOF\_Gantry2** corresponde a transformada inversa do sistema **Gantry2** e é responsável por gerar a referência para cada eixo em sua saída.

O bloco de função do tipo **SMC\_ControlAxisByPosition** controla a posição do eixo conectado a entrada **Axis**. Como a aplicação não garante que as saídas do interpolador sejam constantes (por exemplo, o caminho termina em um ponto diferente de onde começou), é necessário ativar a prevenção de lacunas (**bAvoidGaps**, **fGapVelocity**, **fGapAcceleration**, **fGapDeceleration**).

- Crie um objeto do tipo **Visualizaton**.
- Adicione e referencie os modelos de visualização do tipo **VISU\_NEW\_SMC\_Interpolator** e **SMC\_VISU\_Gantry2** aos blocos de função **SMC\_Interpolator** e **SMC\_TRAFOF\_Gantry2** respectivamente.

A Figura 5.27 mostra o objeto **Visualization** com os modelos adicionados.

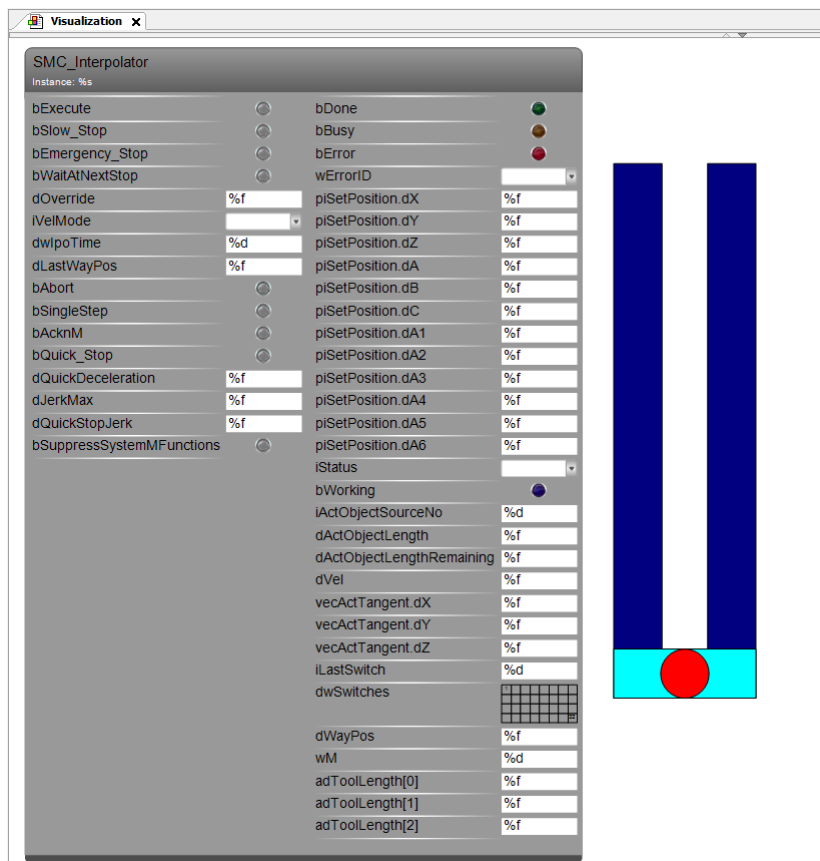


Figura 5.27: Visualização CNC.

- Faça o download do programa para o **PLC500MC**.
- No modo de monitoração **Online**, abra o objeto **Visualization**.
- O programa executa o movimento CNC assim que a entrada **Execute** do interpolador for acionada.
- Após a execução completa do programa, você pode reinicia-lo por meio de uma nova borda de subida na entrada **Execute** do interpolador.
- O movimento pode ser observado pelo modelo de visualização **SMC\_VISU\_Gantry2**.

Se desejar, realize mais alguns testes.

Exemplos de aplicação utilizando caminhos CNC podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > Application Examples).

## 5.6.5 Eixo tangencial em caminhos CNC

O PLC500MC possibilita aplicações que necessitem de um eixo que tangencie o caminho CNC durante o movimento. Este tipo de aplicação geralmente é utilizado em máquina de corte.

- Crie uma nova aplicação como apresentado na Subseção 5.6.2.

Para executar um caminho CNC com eixo tangencial é necessário configurar os eixos que farão parte do movimento.

- Adicione dois eixos virtuais nessa aplicação (**Axis\_A** e **Axis\_B**), como apresentado na Subseção 5.3.
- Adicione um terceiro eixo virtual (**Axis\_R**), como apresentado na Subseção 5.3, porém, na aba **General**, modifique o **Axis type** para **Modulo**. Este será o eixo tangencial.
- Modifique a prioridade da tarefa **MainTask** para 1 e a defina com intervalo cíclico de 4ms.

O programa padrão para executar um caminho CNC controlando um sistema do tipo pórtico 2D com um eixo tangencial é apresentado na Figura 5.28.

- Na árvore de dispositivos, abra o programa **PLC\_PRG(PRG)**.
- Declare as instancias dos blocos de função e faça as ligações dos blocos como apresentado na Figura 5.28.

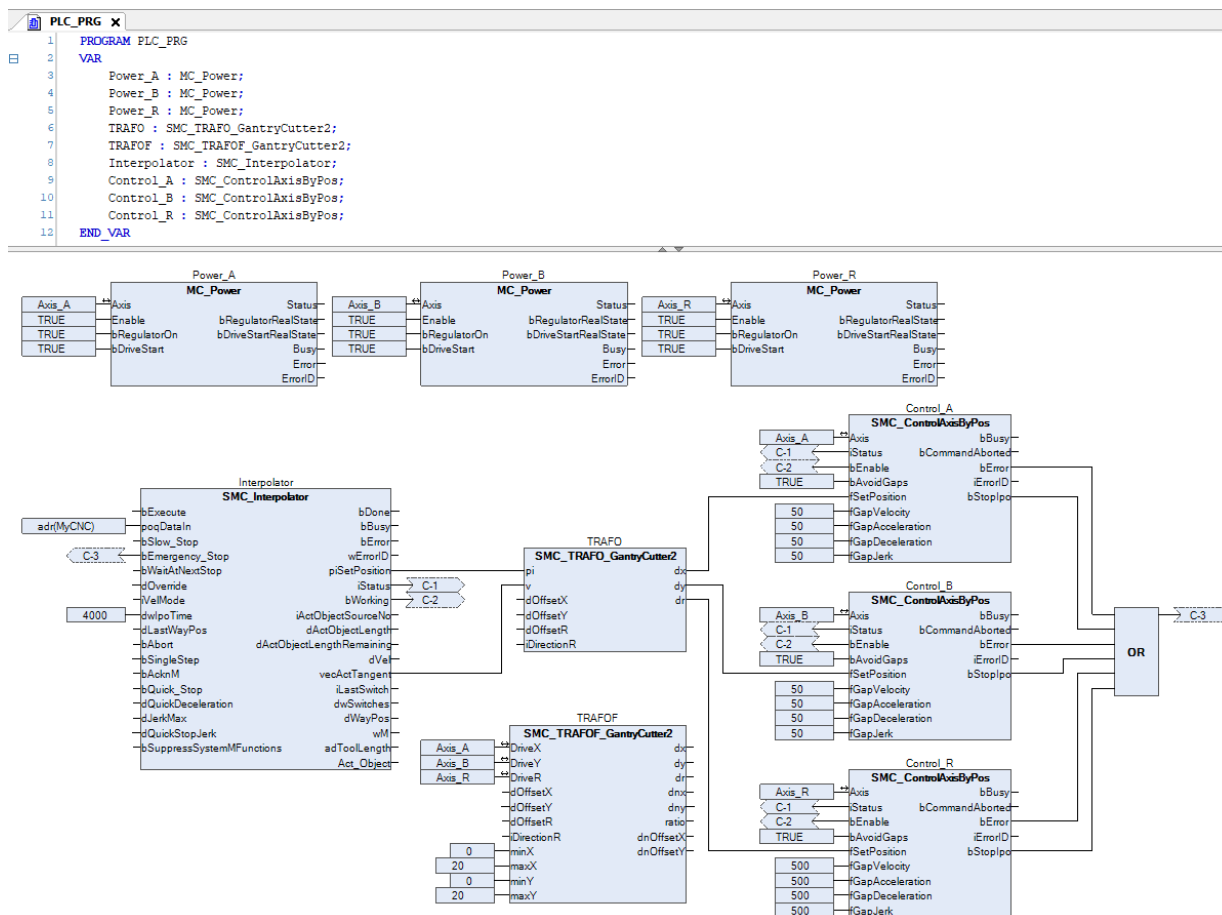


Figura 5.28: Programa para executar caminhos CNC.



**NOTA!**

O Apêndice C apresenta este mesmo programa utilizando a linguagem ST.

O bloco de função do tipo **SMC\_TRAFOF\_GantryCutter2** corresponde a transformada direta do sistema **GantryCutter2** e é necessário apenas para visualização.

O bloco de função do tipo **SMC\_TRAFO\_GantryCutter2** corresponde a transformada inversa do sistema **GantryCutter2** e é responsável por gerar a referência para cada eixo em sua saída.



**NOTA!**

A referência do eixo tangencial é calculada diretamente pelo bloco de função **SMC\_Interpolator** e interpretada pelo bloco de função **GantryCutter2**, dispensando desta forma a necessidade de sua referência no Código-G.

- Crie um objeto do tipo **Visualization**.
- Adicione e referencie os modelos de visualização do tipo **VISU\_NEW\_SMC\_Interpolator** e **SMC\_VISU\_GantryCuuter2** aos blocos de função **SMC\_Interpolator** e **SMC\_TRAFOF\_GantryCutter2** respectivamente.

A Figura 5.29 mostra o objeto **Visualization** com os modelos adicionados.

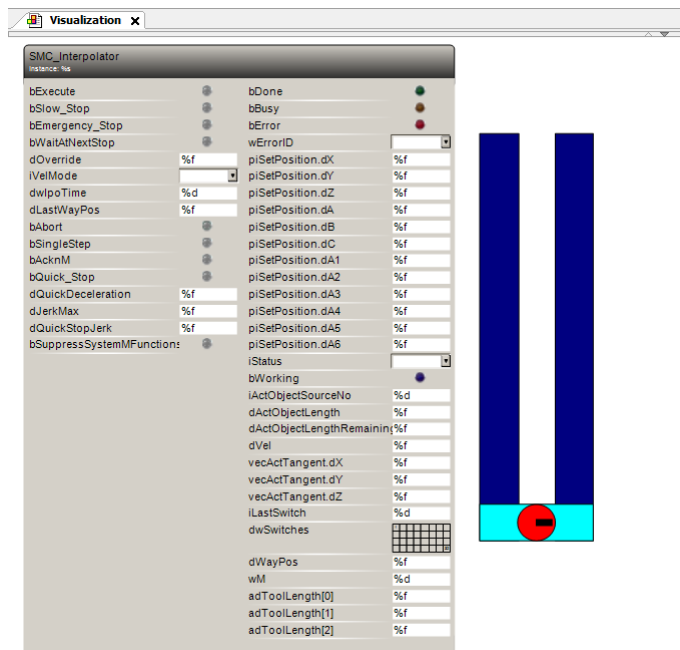


Figura 5.29: Visualização CNC.

- Faça o download do programa para o **PLC500MC**.
- No modo de monitoração **Online**, abra o objeto **Visualization**.
- O programa executa o movimento CNC assim que a entrada **Execute** do interpolador for acionada.
- Após a execução completa do programa, você pode reinicia-lo por meio de uma nova borda de subida na entrada **Execute** do interpolador.
- O movimento pode ser observado pelo modelo de visualização **SMC\_VISU\_GantryCutter2**.

Se desejar, realize mais alguns testes.

Exemplos de aplicação utilizando caminhos CNC podem ser encontradas diretamente no site da Codesys, disponível em: <https://help.codesys.com> (Add-ons > CODESYS SoftMotion > Application Examples).

### 5.7 ALTERAR MODO DE CONTROLE

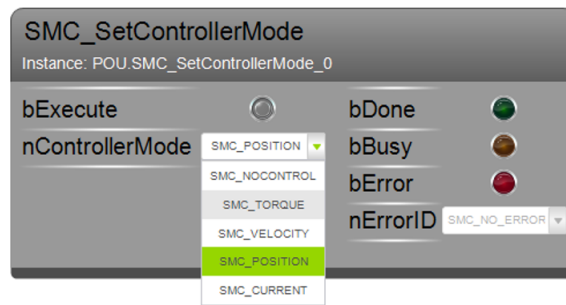
Atualmente o servoconversor SCA06 suporta dois tipos de modo de operação: modo de posição de sincronização cíclica (csp) e modo de velocidade de sincronização cíclica (csv).

O bloco de função **SMC\_SetControllerMode**, pode ser usado para alternar o modo de controle do **SCA06\_Motion**.

Pré-condições:

1. O servoconversor deve suportar o modo de controle desejado.
2. Os **PDOS** de transmissão e recepção necessários devem ser mapeados.
3. O eixo não deve estar no estado **errorstop**, **stop** ou **homing** quando o bloco de função **SMC\_SetControllerMode** for executado.
4. O **SCA06\_Motion** apenas aceitará o novo modo de controle quando desabilitado.

A Figura 5.30 apresenta o modelo de visualização do bloco **SMC\_SetControllerMode**.



**Figura 5.30:** Modelo de visualização do bloco de função **SMC\_SetControllerMode**.

Para trocar o modo de controle:

- Adicione o bloco de função **SMC\_SetControllerMode** em sua aplicação.
- Com a aplicação em modo **Online**. Certifique-se de que o eixo esteja desabilitado (bloco de função **MC\_Power**).
- No bloco de função **SMC\_SetControllerMode** selecione o modo de controle desejado.
- Ative a entrada do bloco de função **bExecute**. A saída **bBusy** do bloco de função ficará ativa durante 1000 ciclos.
- Durante este período, habilite o Eixo.

Com isso, a saída **bDone** do bloco de função **SMC\_SetControllerMode** ficará ativa, indicando que o modo de controle foi modificado.

## 6 CRIAR E CONFIGURAR REDE CAN + SOFTMOTION

Nesta seção são descritas as etapas necessárias para realizar o controle de movimento utilizando uma comunicação CAN entre o PLC500MC e o servoconversor SCA06 através do software Codesys.



### ATENÇÃO!

Para o controle de movimento utilizando a rede CANopen, utilize um eixo CiA402 genérico.

### 6.1 CONFIGURAÇÃO DO SERVOCONVERSOR SCA06 CAN

Conecte corretamente o cabo de comunicação CAN e o servomotor ao servoconversor SCA06.

Partindo dos parâmetros de padrão de fábrica do SCA06:

- Altere o parâmetro **P0202** para **5** (controle via rede CAN/EtherCAT).
- Altere o parâmetro **P0385** para o valor correspondente ao modelo de servomotor utilizado.
- Altere o parâmetro **P0700** para **1** (configura o protocolo de comunicação CAN como sendo o CANopen.)
- Altere o parâmetro **P0701** para **3** (configura o endereço do servo na rede CAN como 3).
- Altere o parâmetro **P0702** para **0** (configura a taxa de comunicação da interface CAN como 1Mbit/s).

Siga as recomendações descritas no manual do servoconversor SCA06 para programar parâmetros de ajuste do equipamento, relativos à parametrização do motor, funções desejadas para os sinais de I/O, etc...

Em caso de dúvida, consulte o Manual de Programação do servoconversor SCA06.

- Reinicie o servoconversor.

Com isso, o servoconversor SCA06 estará pronto para ser acessado através da rede CAN.

### 6.2 CRIAR UM PROJETO NO CODESYS

- Crie um novo projeto em **File > New Project**. Selecione **Standard Project**, defina um diretório e o nome da aplicação. Selecione o Device **PLC500MC** e a linguagem de programação desejada.
- Adicione uma nova tarefa responsável pelo controle de movimento (**Motion\_Task**) nesta aplicação. Aplique as configurações da Figura 6.1.

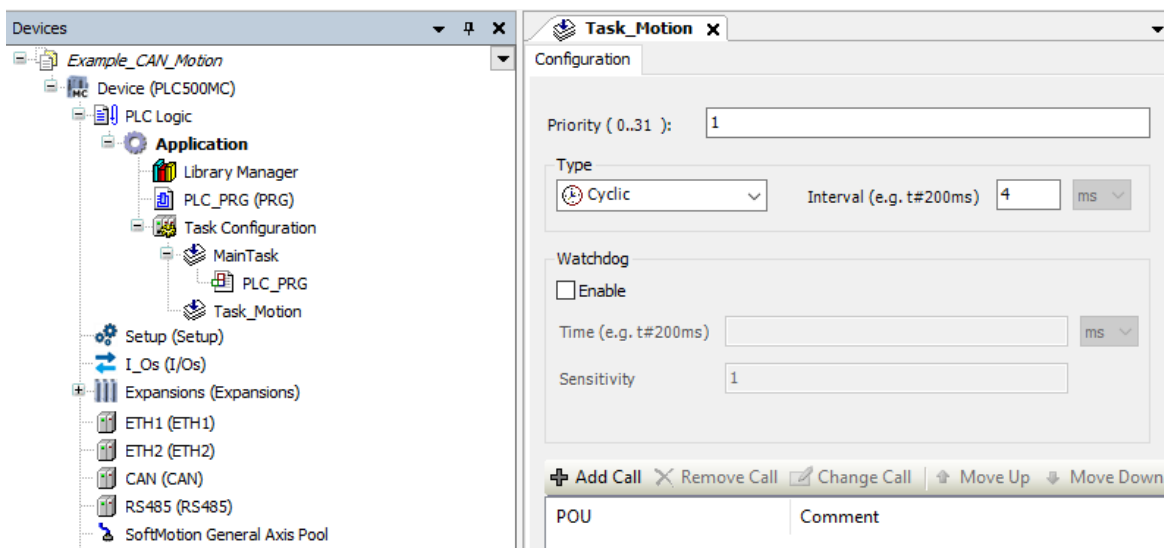


Figura 6.1: Configurações de prioridade.



### 6.2.1 Adicionar CANopen Manager SoftMotion

- Para adicionar uma nova interface de comunicação **CANopen Manager SoftMotion** clique com o botão direito em cima do objeto **CAN** na árvore de dispositivos, clique em **Add Device**, na caixa de diálogo selecione a opção **Append Device**, e então **Fieldbuses > CANopen > CANopen\_Manager\_SoftMotion**, clique em **Add Device** para adicionar à árvore de dispositivos, conforme a Figura 6.2.

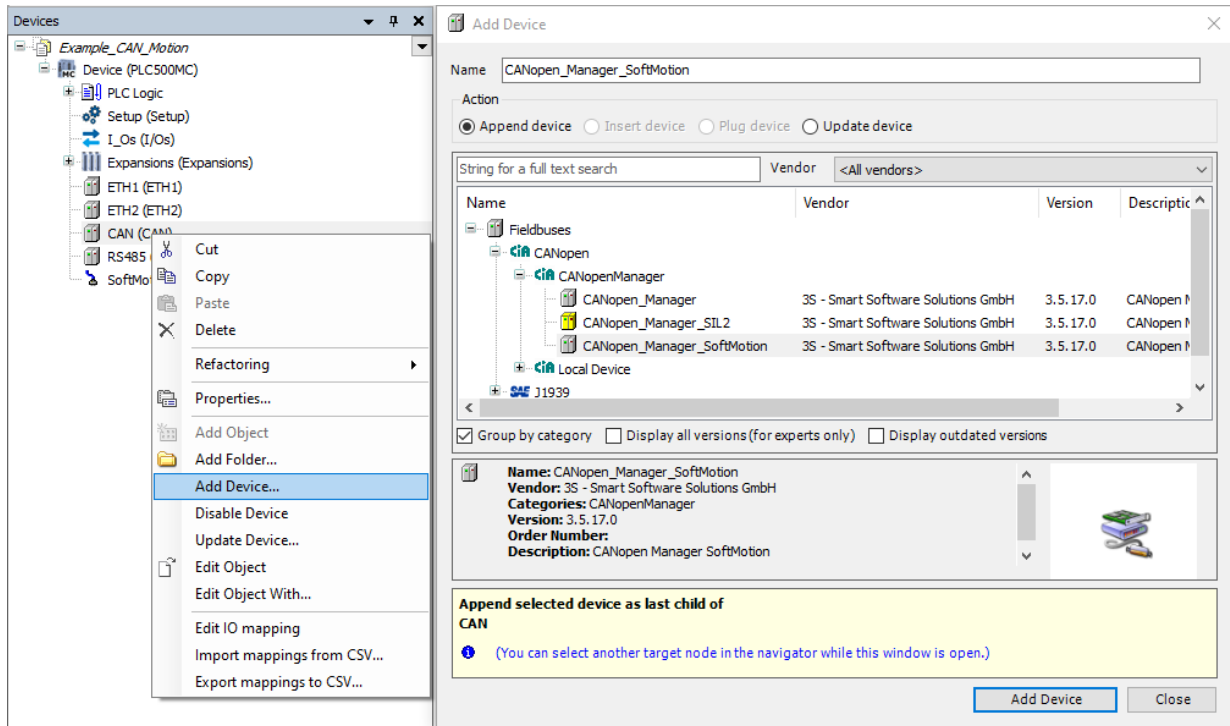
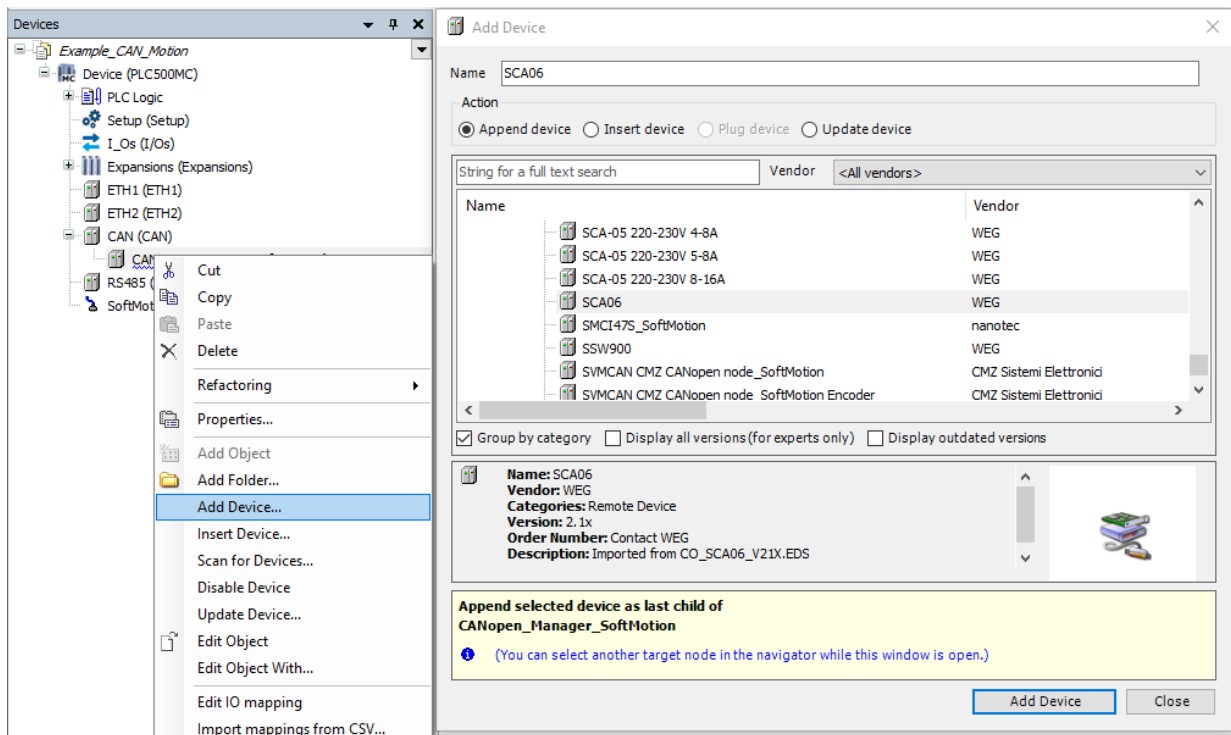


Figura 6.2: Adicionando **CANopen Manager SoftMotion** à árvore de dispositivos.

### 6.2.2 Adicionar SCA06 como escravo na rede CANopen

- Para adicionar o dispositivo **SCA06** como escravo da rede CANopen clique com o botão direito no dispositivo **CANopen Manager SoftMotion** criado anteriormente e selecione a opção **Add Device**.)
- Na seção **Action**, da caixa de diálogo aberta, certifique-se de que a opção **Append device** esteja selecionada. Busque pelo dispositivo **SCA06** **Fieldbuses > CANopen > CANopen Remote Devices > SCA06**.
- Clique em **Add Device**.

A Figura 6.3 apresenta os passos anteriores diretamente no software Codesys.



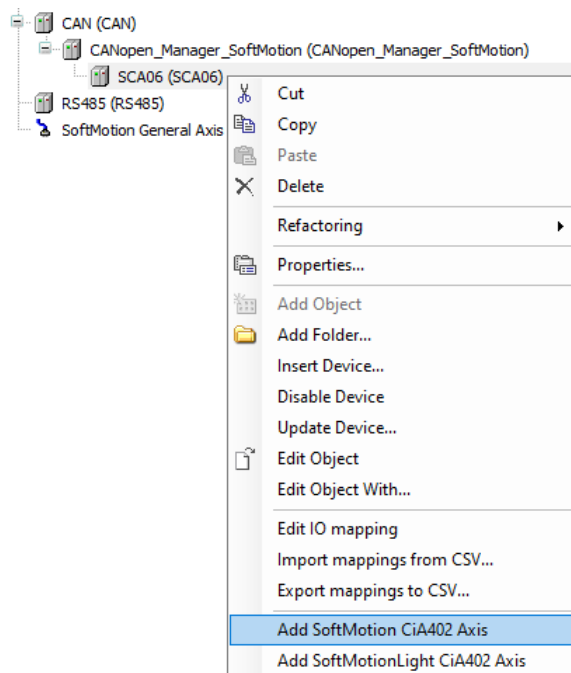
**Figura 6.3:** Adicionando SCA06 como escravo na rede CANopen.



**NOTA!**

Caso o servoconversor SCA06 não esteja disponível, baixe o arquivo .EDS diretamente pelo site da WEG, disponível em: <https://www.weg.net/> e adicione ao repositório de dispositivos do Codesys (**Tools > Device Repository... > Install...**).

- Para adicionar um eixo SoftMotion ao **SCA06**, clique com o botão direito no dispositivo **SCA06** adicionado anteriormente e selecione a opção **Add SoftMotion CiA402 Axis**.



**Figura 6.4:** Adicionando eixo SoftMotion ao SCA06.

## CRIAR E CONFIGURAR REDE CAN + SOFTMOTION

Quando um eixo SoftMotion for adicionado manualmente a caixa de diálogo da Figura 6.5 será exibida.

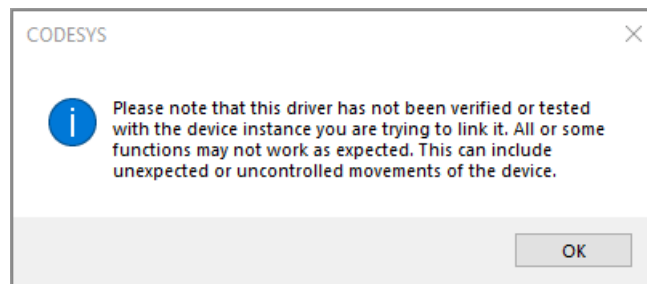


Figura 6.5: Mensagem de alerta ao adicionar um eixo SoftMotion manualmente.

- Leia a mensagem e clique em **OK**.

Após estas configurações a árvore de dispositivos deverá conter os ícones apresentados na Figura 6.6.

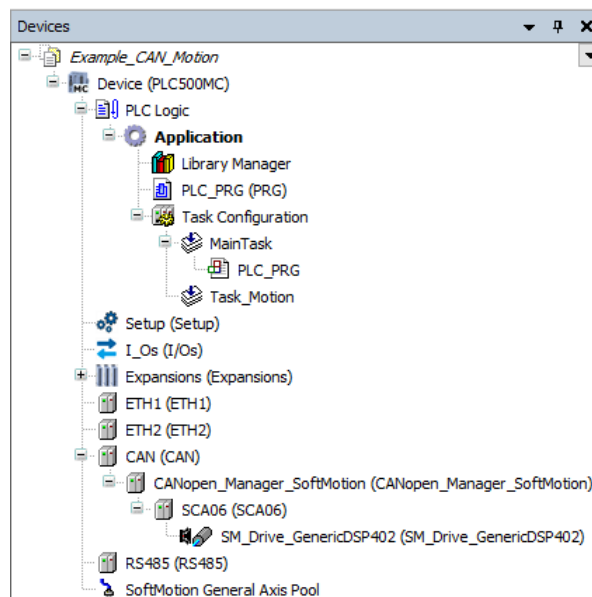


Figura 6.6: Árvore de dispositivos para utilização do SoftMotion.

### 6.2.3 Configurar objeto CAN

- Abra as configurações do dispositivo **CAN**, na aba **General**, configure as opções da página conforme a Figura 6.7.

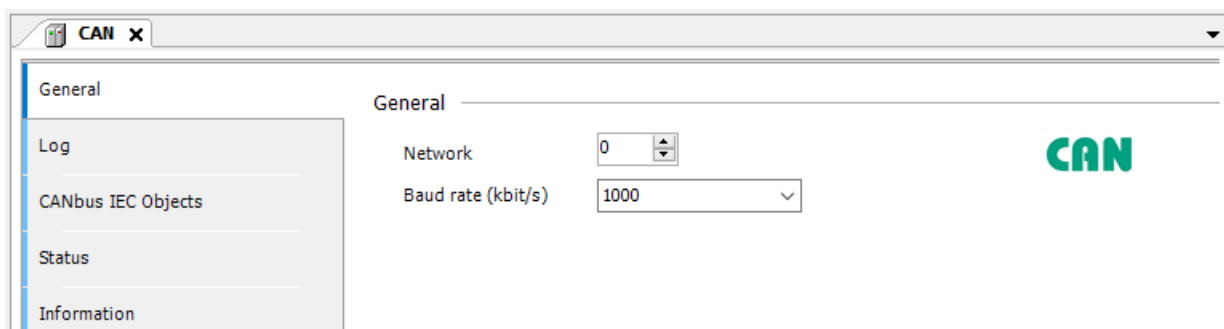


Figura 6.7: Configuração padrão CAN.

## 6.2.4 Configurar objeto CANopen Manager SoftMotion

- Abra as configurações do objeto **CANopen Manager SoftMotion**, na aba **General**, configure as opções da página conforme a Figura 6.8.

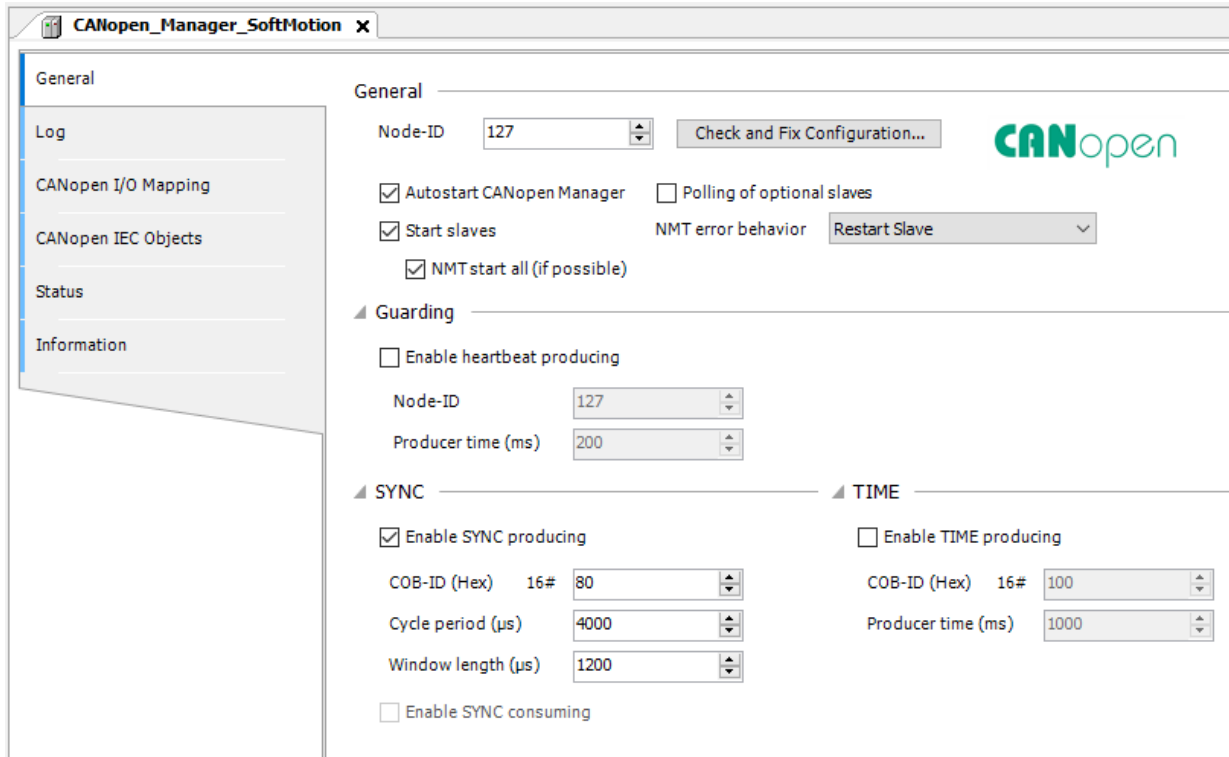


Figura 6.8: Configuração padrão CANopen Manager SoftMotion.

- Ainda nas configurações do objeto **CANopen Manager SoftMotion**, na aba **CANopen I/O Mapping**, selecione a tarefa responsável pelo movimento (**Task\_Motion**), como na Figura 6.9.

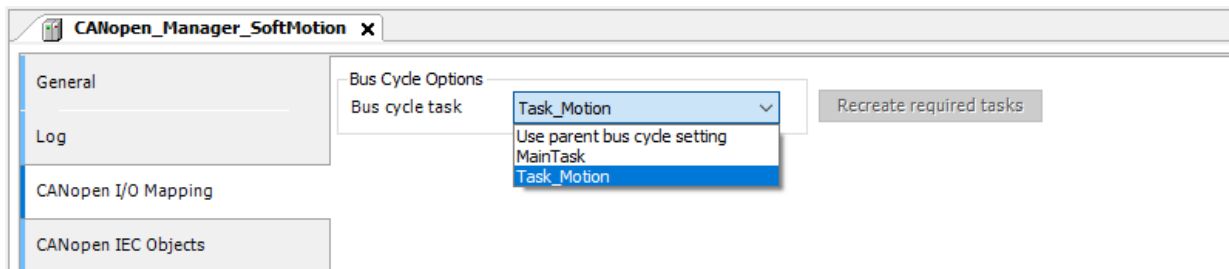


Figura 6.9: Configuração padrão CANopen Manager SoftMotion.

## 6.2.5 Configurar SCA06 como escravo SoftMotion CAN

- Abra as configurações do objeto **SCA06**, na aba **General**, configure as opções da página conforme a Figura 6.10.

## CRIAR E CONFIGURAR REDE CAN + SOFTMOTION

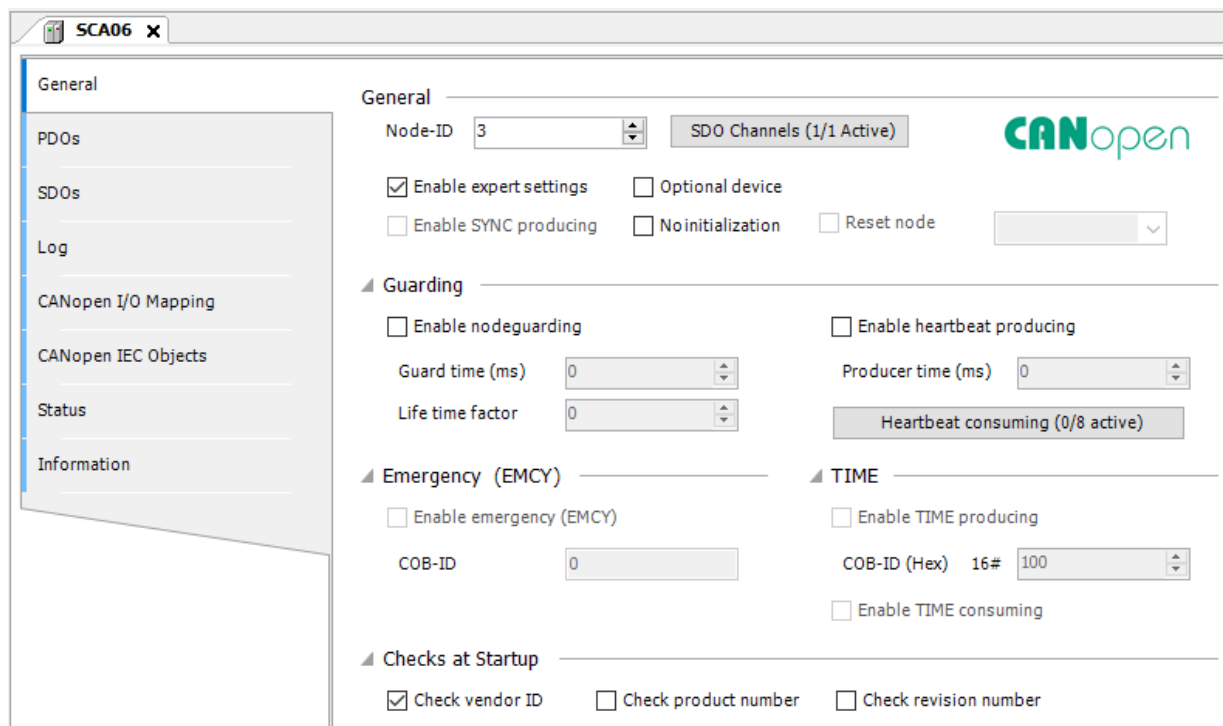


Figura 6.10: Configuração padrão SCA06 na rede CANopen SoftMotion.

- Ainda nas configurações do objeto **SCA06**, na aba **PDOs**, selecione apenas os PDOs da Figura 6.11.

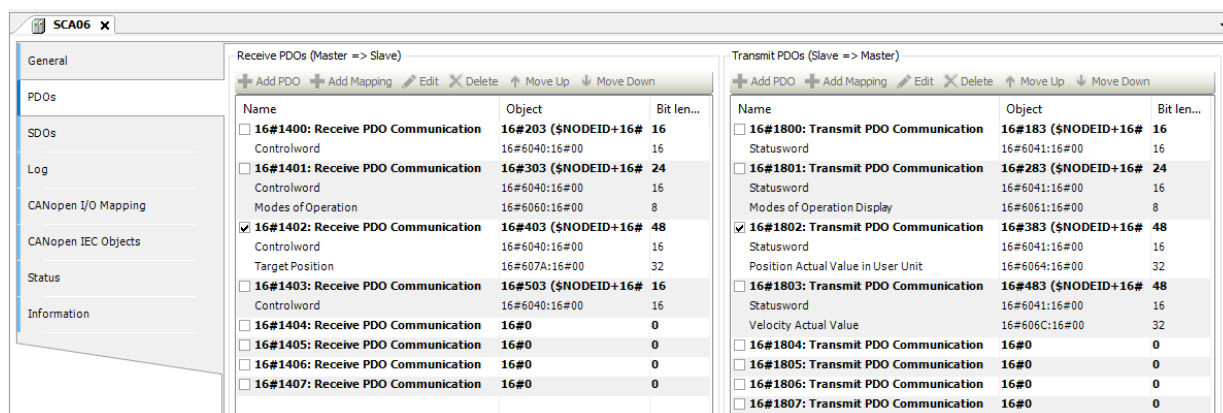
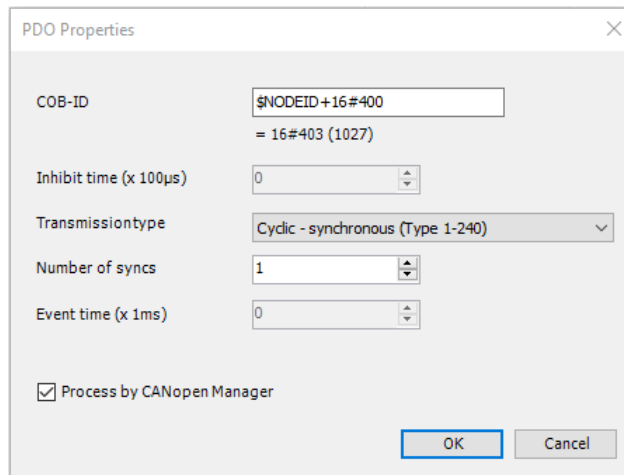


Figura 6.11: Configuração padrão SCA06 na rede CANopen SoftMotion.

- Modifique o **Transmissiontype** dos PDOs de transmissão e recepção para **Cyclic - synchronous (Type 1-240)** (para abrir as propriedades, clique duas vezes no **PDO Communication**).



**Figura 6.12:** Configuração Transmisiontype dos PDOs.

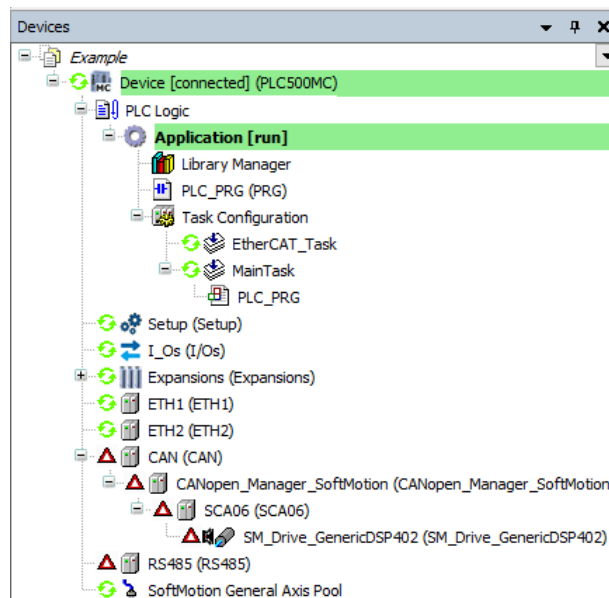
## 6.2.6 Configurar SM\_Drive\_GenericDSP402

- Aplique as mesmas configurações da Subseção 2.4.5.

## 6.3 MONITORAÇÃO

### 6.3.1 Estado da comunicação CAN

O estado da rede CAN pode ser monitorado no modo **Online** do Codesys, indicando o estado de cada uma das etapas de comunicação e reportando o estado (Status). Ao encontrar problemas de conexão como mostrado na Figura 6.13, verifique novamente se os cabos estão devidamente conectados e revise as configurações feitas na Seção 6.



**Figura 6.13:** Indicação de erro na comunicação EtherCAT.

Quando as configurações estiverem corretas e os dispositivos estiverem comunicando adequadamente todos os itens da comunicação CAN estarão em verde, como indicado na Figura 6.14.

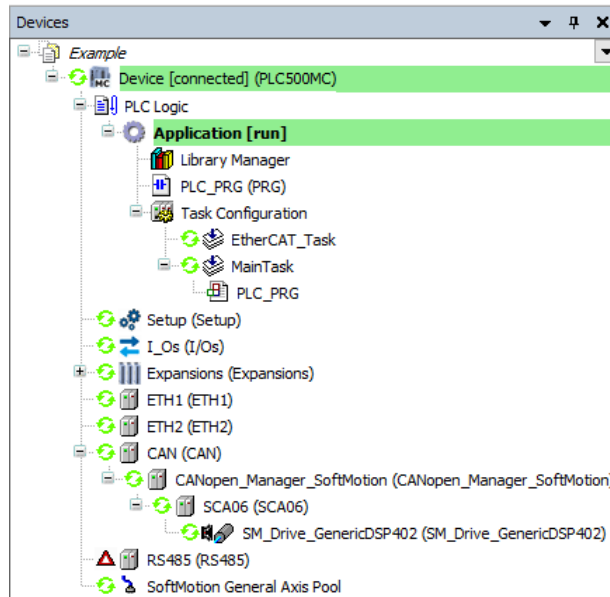


Figura 6.14: Comunicação corretamente configurada e dispositivos comunicados.

### 6.3.2 Verificar variação na posição atual do servomotor

- Após uma correta configuração da rede CAN e ainda no modo **Online** abra as configurações do **SM\_Drive\_GenericDSP402**.

Quando o PLC estiver no modo **Online**, na aba **General**, será habilitado um campo para visualização do eixo, conforme a Figura 6.15.

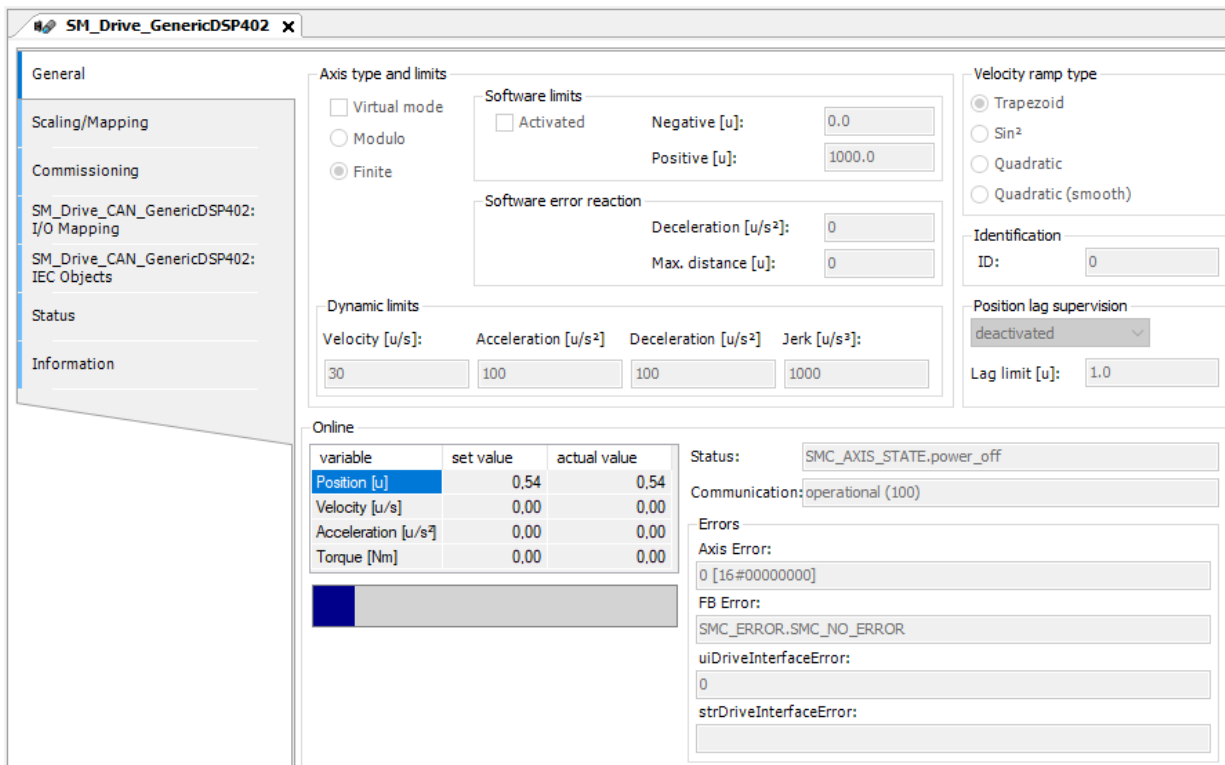


Figura 6.15: Monitoração online do servomotor.

Neste campo é possível observar o estado do eixo e da comunicação, variáveis de posição, velocidade, aceleração e torque, com suas referências e valores atuais.

- Movimente o eixo do servomotor manualmente e observe o valor da posição alterando em **Position [u] - actual value**.

### 6.4 COMISSIONAMENTO

Para testar as configurações aplique as mesmas instruções apresentadas na Subseção 2.6.

Com as configurações aplicadas nesta seção o eixo já pode ser utilizado nas aplicações.



## A APLICAÇÃO CAME

Este apêndice contém a aplicação PLC\_PRG, da Subseção 5.5, em ST.

```

PLC_PRG application:
PROGRAM PLC_PRG
VAR
    Power_A      : MC_Power;
    Power_B      : MC_Power;
    CamTableSelect : MC_CamTableSelect;
    CamIn        : MC_CamIn;
    MoveVelocity  : MC_MoveVelocity;
END_VAR

Power_A(
    Axis:= Axis_A,
    Enable:= TRUE,
    bRegulatorOn:= TRUE,
    bDriveStart:= TRUE);

Power_B(
    Axis:= Axis_B,
    Enable:= TRUE,
    bRegulatorOn:= TRUE,
    bDriveStart:= TRUE);

CamTableSelect(
    Master:= Axis_A,
    Slave:= Axis_B,
    CamTable:= MyCam,
    Execute:= TRUE);

CamIn(
    Master:= Axis_A,
    Slave:= Axis_B,
    Execute:= Power_A.bDriveStartRealState,
    CamtableID:= CamTableSelect.CamTableID);

MoveVelocity(
    Axis:= Axis_A,
    Acceleration:= 10,
    Deceleration:= 10,
    Jerk:= 10);
    
```

## B APLICAÇÃO CNC

Este apêndice contém a aplicação MyMotion, da Seção 5.6, em ST.

```

MyMotion Application:
PROGRAM MyMotion
VAR
    Power_A          : MC_Power;
    Power_B          : MC_Power;
    Interpolator     : SMC_Interpolator;
    Control_A        : SMC_ControlAxisByPos;
    Control_B        : SMC_ControlAxisByPos;
    TRAFO            : SMC_TRAFO_Gantry2;
    TRAFOF           : SMC_TRAFOF_Gantry2;
END_VAR

Power_A(
    Axis:= Axis_A,
    Enable:= TRUE,
    bRegulatorOn:= TRUE,
    bDriveStart:= TRUE);

Power_B(
    Axis:= Axis_B,
    Enable:= TRUE,
    bRegulatorOn:= TRUE,
    bDriveStart:= TRUE);

TRAFOF(
    DriveX:= Axis_A,
    DriveY:= Axis_B,
    minX:= 0,
    maxX:= 20,
    minY:= 0,
    maxY:= 20);

Interpolator(
    poqDataIn:= ADR(MyCNC),
    bEmergency_Stop:= Control_B.bError OR Control_B.bStopIpo OR Control_A.bError OR Control_A.bStopIpo,
    dwIpoTime:= 4000);

TRAFO(
    pi:= Interpolator.piSetPosition);

Control_A(
    Axis:= Axis_A,
    iStatus:= Interpolator.iStatus,
    bEnable:= Interpolator.bWorking,
    bAvoidGaps:= TRUE,
    fSetPosition:= TRAFO.dx,
    fGapVelocity:= 50,
    fGapAcceleration:= 50,
    fGapDeceleration:= 50,
    fGapJerk:= 50);

Control_B(
    Axis:= Axis_B,
    iStatus:= Interpolator.iStatus,
    bEnable:= Interpolator.bWorking,
    bAvoidGaps:= TRUE,
    fSetPosition:= TRAFO.dy,
    fGapVelocity:= 50,
    fGapAcceleration:= 50,
    fGapDeceleration:= 50,
    fGapJerk:= 50);
    
```

## C APLICAÇÃO CNC TANGENCIAL

Este apêndice contém a aplicação MyMotion, da Seção 5.6.5, em ST.

```

MyMotion Application:
PROGRAM MyMotion
VAR
    Power_A          : MC_Power;
    Power_B          : MC_Power;
    Power_R          : MC_Power;
    Interpolator     : SMC_Interpolator;
    Control_A        : SMC_ControlAxisByPos;
    Control_B        : SMC_ControlAxisByPos;
    Control_R        : SMC_ControlAxisByPos;
    TRAFO            : SMC_TRAFO_GantryCutter2;
    TRAFOF           : SMC_TRAFOF_GantryCutter2;
END_VAR

Power_A(
    Axis:= Axis_A,
    Enable:= TRUE,
    bRegulatorOn:= TRUE,
    bDriveStart:= TRUE);

Power_B(
    Axis:= Axis_B,
    Enable:= TRUE,
    bRegulatorOn:= TRUE,
    bDriveStart:= TRUE);

Power_R(
    Axis:= Axis_R,
    Enable:= TRUE,
    bRegulatorOn:= TRUE,
    bDriveStart:= TRUE);

TRAFOF(
    DriveX:= Axis_A,
    DriveY:= Axis_B,
    DriveR:= Axis_R,
    minX:= 0,
    maxX:= 20,
    minY:= 0,
    maxY:= 20);

Interpolator(
    poqDataIn:= ADR(MyCNC),
    bEmergency_Stop:= Control_B.bError OR Control_B.bStoplpo OR Control_A.bError OR Control_A.bStoplpo OR
Control_R.bError OR Control_R.bStoplpo,
    dwlpoTime:= 4000);

TRAFO(
    pi:= Interpolator.piSetPosition,
    v:= Interpolator.vecActTangent );
    
```

```
Control_A(  
  Axis:= Axis_A,  
  iStatus:= Interpolator.iStatus,  
  bEnable:= Interpolator.bWorking,  
  bAvoidGaps:= TRUE,  
  fSetPosition:= TRAF0.dx,  
  fGapVelocity:= 50,  
  fGapAcceleration:= 50,  
  fGapDeceleration:= 50,  
  fGapJerk:= 50);
```

```
Control_B(  
  Axis:= Axis_B,  
  iStatus:= Interpolator.iStatus,  
  bEnable:= Interpolator.bWorking,  
  bAvoidGaps:= TRUE,  
  fSetPosition:= TRAF0.dy,  
  fGapVelocity:= 50,  
  fGapAcceleration:= 50,  
  fGapDeceleration:= 50,  
  fGapJerk:= 50);
```

```
Control_R(  
  Axis:= Axis_R,  
  iStatus:= Interpolator.iStatus,  
  bEnable:= Interpolator.bWorking,  
  bAvoidGaps:= TRUE,  
  fSetPosition:= TRAF0.dr,  
  fGapVelocity:= 500,  
  fGapAcceleration:= 500,  
  fGapDeceleration:= 500,  
  fGapJerk:= 500);
```



WEG Drives & Controls - Automação LTDA.  
Jaraguá do Sul - SC - Brasil  
Fone 55 (47) 3276-4000 - Fax 55 (47) 3276-4020  
São Paulo - SP - Brasil  
Fone 55 (11) 5053-2300 - Fax 55 (11) 5052-4212  
[automacao@weg.net](mailto:automacao@weg.net)  
[www.weg.net](http://www.weg.net)