



SIEMENS

Documentação de treinamento SCE

Siemens Automation Cooperates with Education
(SCE) | A partir da versão V14 SP1

Módulo TIA Portal 031-100
Fundamentos da Programação FC
com SIMATIC S7-1200

[siemens.com/sce](https://www.siemens.com/sce)

SIEMENS

Global Industry
Partner of
WorldSkills
International



Paquetes SCE apropriados para esta Documentação de treinamento

- **SIMATIC S7-1200 AC/DC/RELÉ 6º "TIA Portal"**
Nº de pedido: 6ES7214-1BE30-4AB3
- **SIMATIC S7-1200 DC/DC/DC 6º "TIA Portal"**
Nº de pedido: 6ES7214-1AE30-4AB3
- **Upgrade SIMATIC STEP 7 BASIC V14 SP1 (para o S7-1200) 6º "TIA Portal"**
Nº de pedido 6ES7822-0AA04-4YE5

Note que os pacotes de treinamento podem ser substituídos por pacotes atualizados quando necessário. Um resumo dos pacotes SCE atualmente disponíveis pode ser encontrado em: [siemens.com/sce/tp](https://www.siemens.com/sce/tp)

Treinamentos avançados

Para treinamentos regionais avançados SCE Siemens, entre em contato com o parceiro SCE da sua região [siemens.com/sce/contact](https://www.siemens.com/sce/contact)

Outras informações sobre SCE

[siemens.com/sce](https://www.siemens.com/sce)

Nota sobre o uso

A Documentação de treinamento SCE para plataforma de engenharia TIA Totally Integrated Automation foi elaborada para o programa "Siemens Automation Cooperates with Education (SCE)" especificamente para fins educacionais em instituições públicas de ensino, pesquisa e desenvolvimento. A Siemens AG não assume responsabilidade sobre o conteúdo.

Este documento só pode ser utilizado para o treinamento inicial em produtos/sistemas da Siemens. Portanto, ele pode ser copiado totalmente ou parcialmente e entregue aos alunos do treinamento para o uso dentro do âmbito do curso. A transmissão e reprodução deste documento, bem como a divulgação de seu conteúdo, são permitidas apenas para fins educacionais.

As exceções demandam a aprovação por escrito do representante da Siemens AG: Sr. Roland Scheuerer roland.scheuerer@siemens.com.

As violações estão sujeitas a indenização por danos. Todos os direitos, inclusive da tradução, são reservados, particularmente para o caso de registro de patente ou marca registrada.

A utilização em cursos para clientes industriais é expressamente proibida. O uso comercial dos documentos não é autorizado.

Agradecemos à Universidade Técnica de Dresden (TU Dresden), especialmente ao Prof. Dr. Eng. Leon Urbas à firma Michael Dziallas Engineering e todos os outros envolvidos pelo auxílio na preparação desta Documentação de treinamento SCE.

Diretório

1	Objetivo	5
2	Requisito.....	5
3	Hardwares e softwares necessários	6
4	Teoria	7
4.1	Sistema operacional e programa de aplicação	7
4.2	Bloco de organização	8
4.3	Imagem de processo e processamento cíclico de programa	9
4.4	Funções	11
4.5	Blocos de funções e blocos de dados de instância.....	12
4.6	Blocos de dados globais	13
4.7	Blocos de código habilitados para biblioteca	14
4.8	Linguagens de programação	15
5	Definição da tarefa	16
6	Planejamento.....	16
6.1	PARADA DE EMERGÊNCIA	16
6.2	Operação manual – motor de correia no comando por impulsos	16
6.3	Esquema de tecnologia	17
6.4	Tabela de atribuição	18
7	Instrução estruturada passo a passo	19
7.1	Extrair arquivo de um projeto atual.....	19
7.2	Criar tabela de variantes.....	20
7.3	Criar variáveis dentro de uma tabela de variáveis	22
7.4	Importação da "Tag table_sorting station" (Tabela de variáveis_sistema de classificação)	23
7.5	Criar função FC1 "MOTOR_manual" para o motor de correia no comando por impulsos.....	27
7.6	Definir interface da função FC1 "MOTOR_manual"	29
7.7	Programação do FC1: MOTOR_MANUAL	32
7.8	Programação do bloco de organização OB1 – controller do movimento da correia para frente na operação manual.....	39
7.9	Salvar e verter programa	44
7.10	Carregar programa	45
7.11	Observar blocos de programa.....	46
7.12	Arquivar projeto	48
7.13	Lista de verificação	49
8	Exercício	50

8.1	Tarefa – exercício	50
8.2	esquema de tecnologia.....	50
8.3	Tabela de atribuição	51
8.4	Planejamento	51
8.5	Lista de verificação – exercício	52
9	Informação adicional	53

Fundamentos da Programação FC

1 Objetivo

Neste capítulo você aprender sobre os elementos básicos de um programa de controller – o **bloco de organização (OB)**, as **funções (FC)**, **blocos funcionais (FB)** e **blocos de dados (DB)**. Adicionalmente, será apresentada a função e programação de bloco funcional **utilizando a biblioteca**. Você irá conhecer a linguagem de programação **Plano de função (FUP)** e aprenderá a usar esta para programação de uma função FC1 e um bloco de organização OB1.

Os comandos SIMATIC S7 listados no capítulo 3 podem ser utilizados.

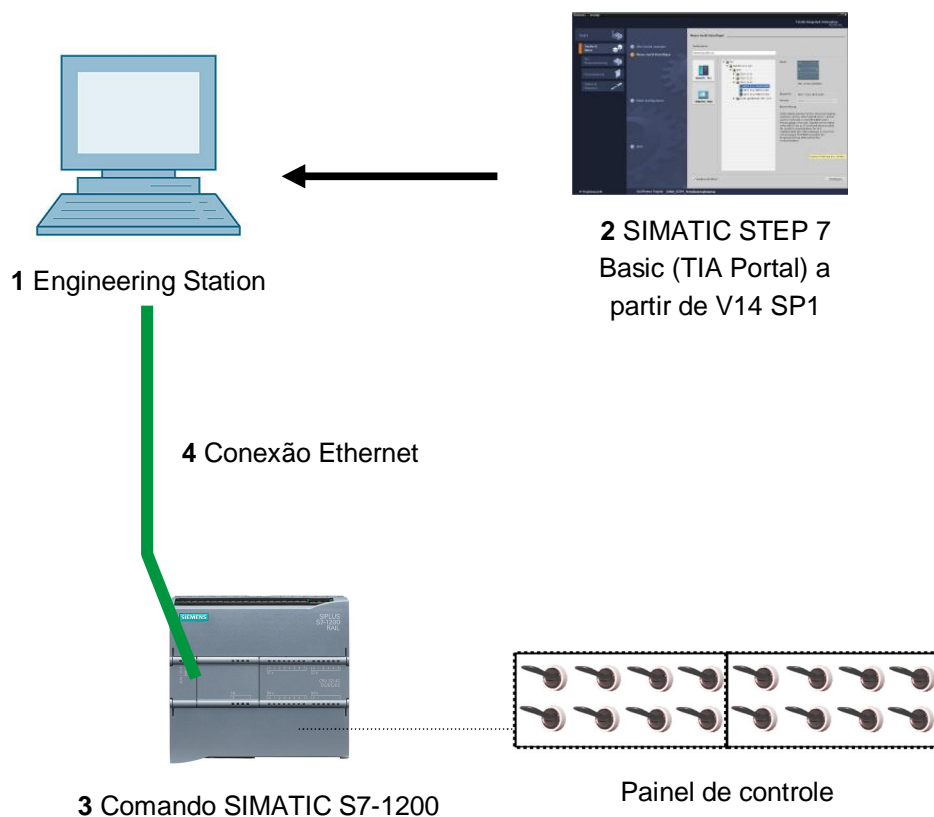
2 Requisito

Este capítulo tem como base a configuração de hardware do SIMATIC S7 CPU1214C, no entanto, também é possível realizar outras configurações de hardware, que tenham entrada digital e saídas de cartão. Para realização deste capítulo, pode-se recorrer ao seguinte projeto:

SCE_PT_011_101_ Configuração do hardware _CPU1214C.zap14

3 Hardwares e softwares necessários

- 1 Engineering Station: Pré-requisitos são hardware e sistema operacional (outras informações, vide Readme nos DVDs TIA Portal Installations)
- 2 Software SIMATIC STEP 7 Basic no TIA Portal – a partir de V14 SP1
- 3 Comando SIMATIC S7-1200, p. ex. CPU 1214C DC/DC/DC com Signalboard ANALOG OUTPUT SB1232, 1 AO – a partir de Firmware V4.2.1 Nota: As entradas digitais deverão ser executadas em um painel de controle.
- 4 Conexão Ethernet entre Engineering Station e comando



4 Teoria

4.1 Sistema operacional e programa de aplicação

O **sistema operacional** está contido em todo controller (CPU) e organiza todas as funções e sequências da CPU, que não estão relacionadas a uma tarefa de controle específico. Às tarefas do sistema operacional pertencem, por ex.:

- Realizar a reinicialização (warm start)
- Atualização da imagem do processo das entradas e representação das saídas
- Solicitação cíclica do programa de usuário
- Captação e solicitação de alarm-OBs
- Detecção e resolução de erros
- Gerenciamento de áreas de armazenamento

O sistema operacional é componente da CPU, que é entregue com este contida nela.

O **programa do usuário** contém todas as funções, necessárias ao processamento de suas tarefas específicas de automação. Às tarefas do programa do usuários pertencem:

- Análise das condições prévias para uma reinicialização (warm start) usando a partida com os OBs
- Processamento dos dados (do processo), do acionamento dos sinais de saída de acordo aos status dos sinais de entrada
- Resposta a alarmes e entradas de alarme
- Processamento de interferências na sequência normal do programa

4.2 Bloco de organização

Os blocos de organização (OB) formam a interface entre o sistema operacional do controller (CPU) e o programa de usuário. Eles são chamados pelo sistema operacional e controlam as seguintes operações:

- Processamento cíclico de programa (por ex., OB1)
- Comportamento de start-up do controller
- Processamento do programa controlado por alarme
- Tratamento de erros

Em um projeto deve ter pelo menos **um bloco de organização para o processamento cíclico do programa**. Um bloco OB será solicitado através de um **evento de inicialização**, como mostrado na Imagem 1. Assim, cada bloco OB têm prioridades definidas, para que, por ex., um OB82 para tratamento de erros possa interromper o OB1 cíclico.

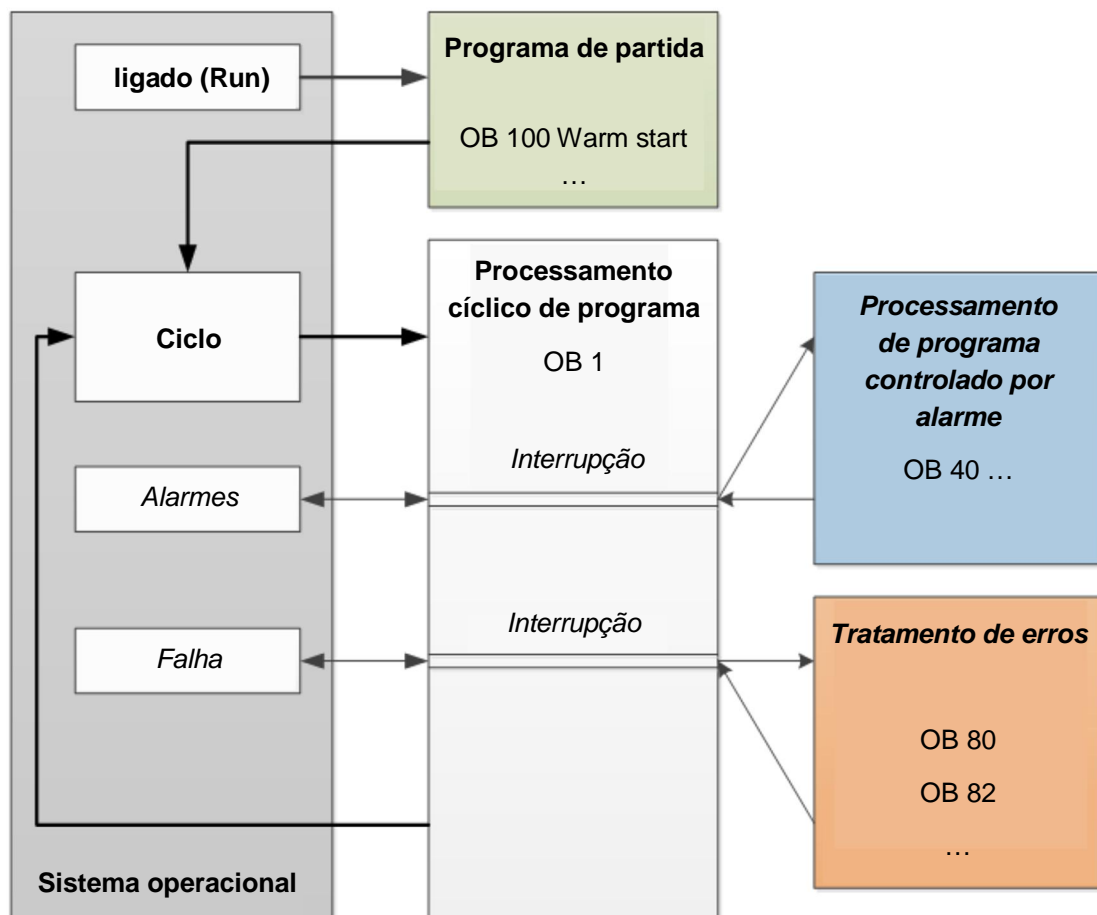


Imagem 1: Evento de inicialização no sistema operacional e solicitação OB

Após surgir um evento de inicialização, as seguintes respostas serão possíveis:

- Se ao evento for associado um OB, este evento aciona a execução do OB associado. Se a prioridade do OB associado for maior do que a prioridade dos OBs em atual execução, ele é acionado imediatamente (interrupção). Não sendo este o caso, espera-se primeiro até o OB com a prioridade mais alta puder ser executado.
- Se você não atribuiu o OB a um evento, a resposta padrão do sistema será efetuada.

A tabela 1 mostra exemplos para um SIMATIC S7-1200 para diversos eventos de inicialização. Serão mostrados também os possíveis número (s) OB e as respostas predefinidas do sistema, que ocorrem quando não há o respectivo módulo de organização (OB) no controller.

Evento de inicialização	Números OB possíveis	Resposta padrão do sistema
Partida	100, ³ 123	Ignorar
Programa cíclico	1, ³ 123	Ignorar
Alarme de tempo	10 a 11	-
Atualização de alarme	56	Ignorar
Ciclo de monitoramento de tempo excedido uma vez	80	Ignorar
Ciclo de monitoramento de tempo excedido duas vezes	80	STOP
Alarme de diagnóstico	82	Ignorar

Tabela 1: Números de OB para diversos eventos de inicialização

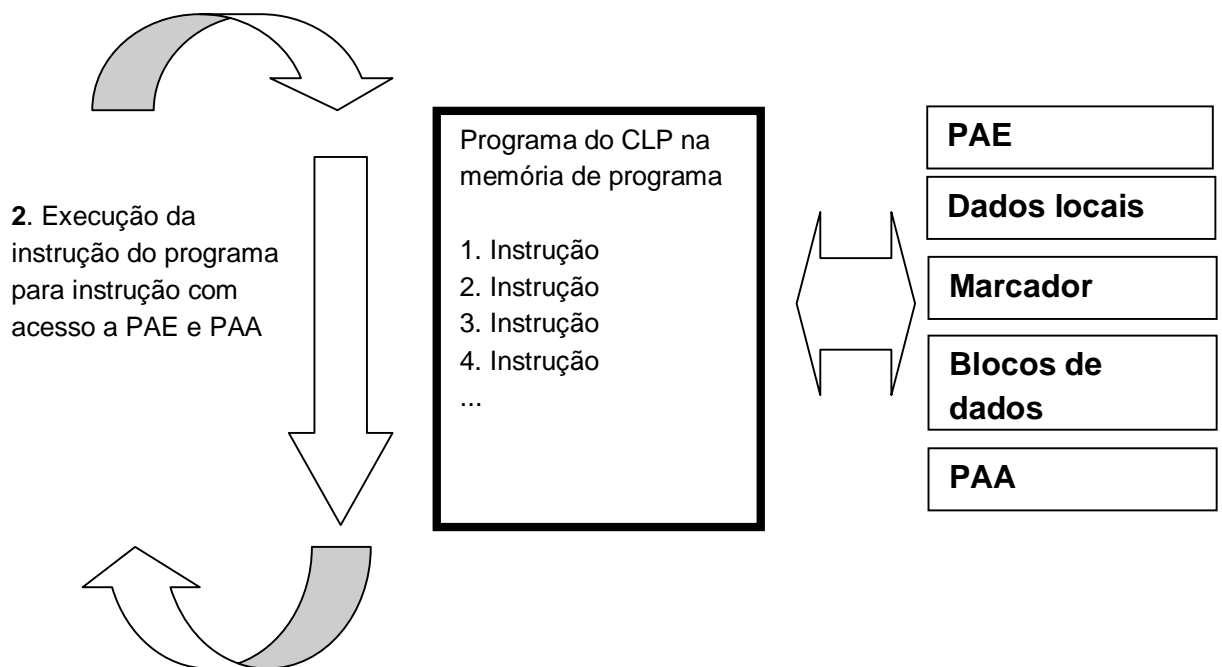
4.3 Imagem de processo e processamento cíclico de programa

Se no programa cíclico do usuário, as entradas (E) e saídas (A) forem requeridas, o status de sinal normalmente não será solicitado direto nos módulos de entrada e saída, mas acessado área de armazenamento da CPU. Esta área de armazenamento contém uma imagem do status de sinais e será designada **Imagem do processo**.

O processamento cíclico de programa ocorre com a seguinte sequência:

1. No início do programa cíclico será perguntado, se cada entrada conduzem tensão ou não. Este status das entradas é salvo na imagem de **processo das entradas (PAE)**. Nisto, a informação 1 ou "High" é definida para as entradas conduzindo tensão e a informação 0 ou "Low" é definida para as entradas não conduzindo tensão.
2. Este processador executa o programa salvo no bloco cíclico de organização. Para a informação de entrada, é acessada a **imagem do processo das entradas (PAE)** anteriormente lida e os resultados das operações são gravados em uma assim chamada **imagem de processo das saídas (PAA)**.
3. No fim do ciclo a **Imagem do processo das saídas (PAA)** será transmitida como status de sinal aos módulos de saída e liga e desliga estes. Em seguida, como continuação retorna-se ao item 1 novamente.

1. Salvar o status das entradas na PAE.



3. Transferir o status da PAA às saídas.

Imagem 2: Processamento cíclico do programa

Indicação: O tempo que o processador necessita para esta sequência é chamado de tempo de ciclo. Este, por sua vez, depende da quantidade e do tipo de instruções e do desempenho do processador do controller.

4.4 Funções

Funções (FCs) são blocos de código sem memória. Eles **não têm memória de dados**, em qual possa se armazenar valores dos parâmetros do bloco. Portanto, todos os parâmetros de interface devem ser ligados ao solicitar uma função. Para armazenar dados permanentemente, os blocos de dados anteriormente compartilhados precisam ser criados.

Uma função contém um programa que é executado sempre que a função for chamada por outro bloco de código.

As funções podem ser aplicadas, por exemplo, para as seguintes finalidades:

- Funções matemáticas – que dão um resultado dependendo dos valores de entrada.
- Funções tecnológicas – como acionamento individual com conexões binárias.

Uma função também pode ser chamada diversas vezes em diferentes locais dentro de um programa.

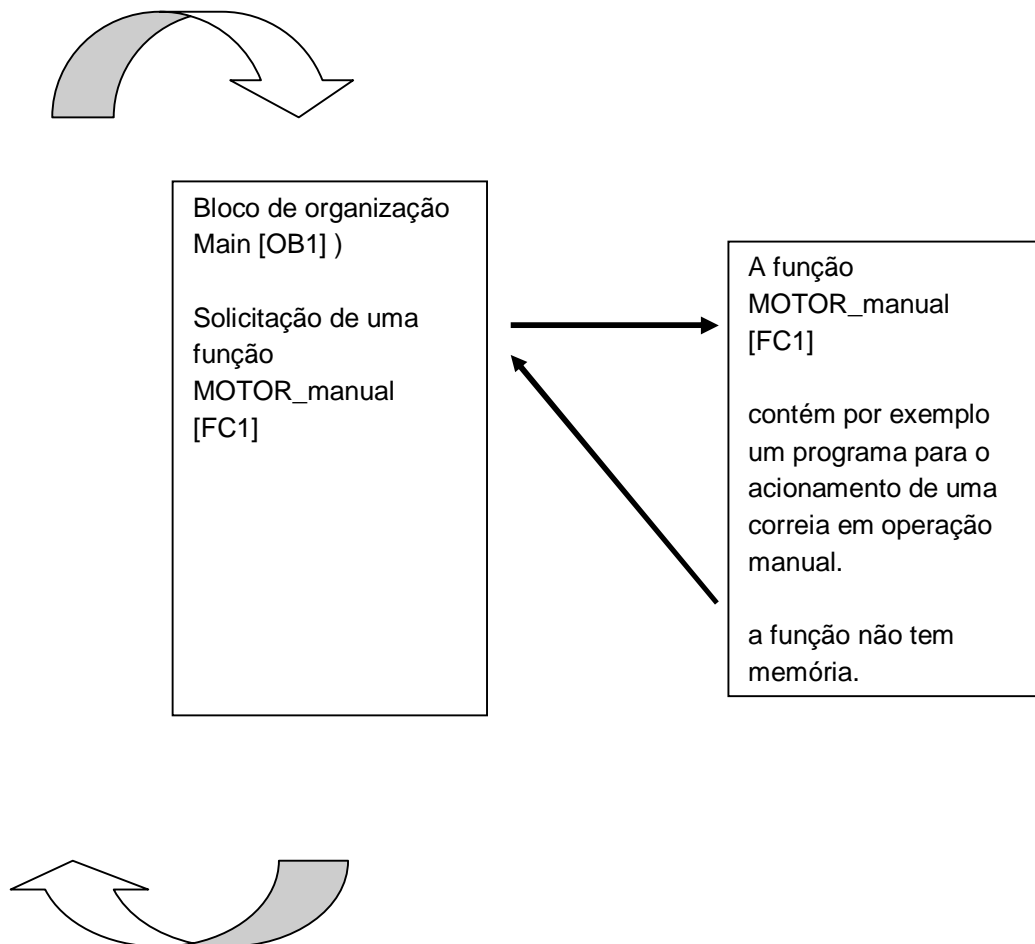


Imagem 3: Função com solicitação do bloco de organização Main[OB1]

4.5 Blocos de funções e blocos de dados de instância

Blocos de funções são blocos de código, que colocam suas variáveis de entrada, variáveis de saída, variáveis de passagem e também a variável estática permanentemente em módulos de dados de instância, para que elas **fiquem disponíveis após o processamento bloco**. É por isto que eles também são chamados de blocos com "memória".

Blocos de funções podem trabalhar com variáveis temporários. As variáveis temporárias não são salvas no bloco DB de instância, mas só ficam disponíveis por longo tempo para um ciclo.

Os blocos de função são usados em tarefas que não podem ser realizadas com funções:

- sempre que os tempos e contadores forem requeridos nos módulos ou
- quando uma informação tiver que ser armazenada no programa. Por exemplo, uma pré-seleção do modo de operação com uma tecla.

Blocos de funções serão sempre executados, se um bloco de função for solicitado por outro bloco de código. Um bloco de função também pode ser chamado diversas vezes em diferentes locais dentro de um programa. Desta forma, elas facilitam a programação de funções complexas que se repetem com frequência.

A chamada de um bloco de função é denominada como instância. A cada instância de um bloco de função é atribuído um espaço de memória, que irá conter os dados com os quais o bloco de função irá trabalhar. Esta memória é disponibilizada pelos blocos de dados, que são automaticamente criados pelo software.

Também é possível disponibilizar a memória para diversas instâncias em um bloco de dados na forma de **múltipla instância**. O tamanho máximo do blocos de dados de instância variam dependendo da CPU. As variáveis declaradas no bloco de função determinam a estrutura do bloco de dados de instância.

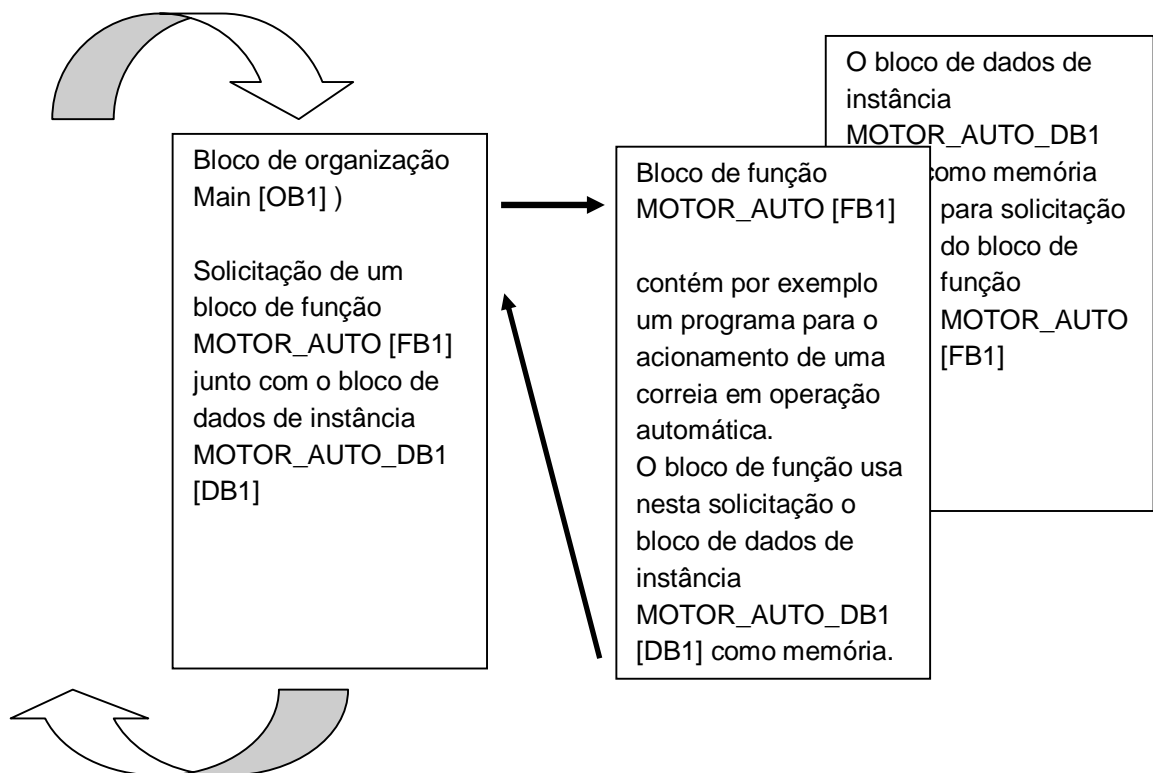


Imagem 4: Bloco de função e instância com solicitação do bloco de organização Main[OB1]

4.6 Blocos de dados globais

Ao contrário dos blocos de código, os blocos de dados não contêm nenhuma instrução, mas são usados para o armazenamento dos dados de usuário.

Nos blocos de dados, portanto, estão contidos dados variáveis com os quais o programa de usuário trabalha. A estrutura dos blocos de dados de função pode ser livremente definida.

Blocos de dados universais registram dados, que podem ser usados **por todos os outros blocos** (veja a imagem 5). Aos blocos de dados de instância, só deve acessar o bloco de função correspondente. O tamanho máximo dos blocos de dados varia dependendo da CPU.

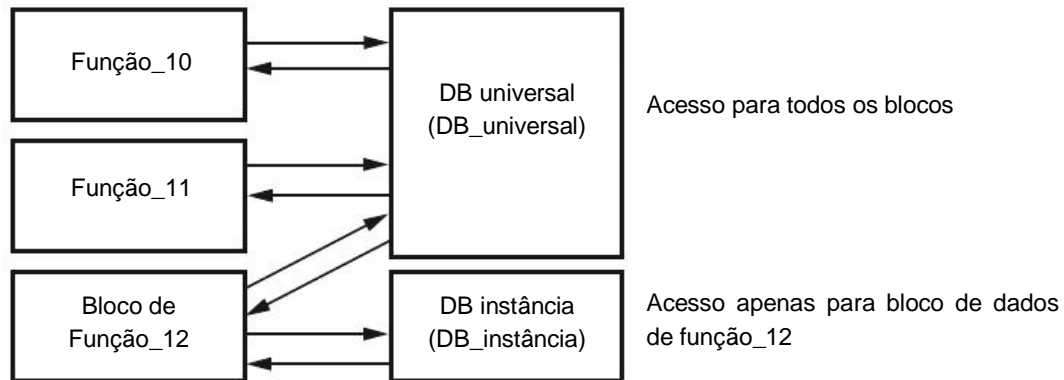


Imagem 5: Diferença entre DB universais e DB de instância.

Exemplos de aplicação para **Blocos de dados universais** são:

- armazenamento das informações de um sistema de armazenamento. "Qual produto está onde?"
- armazenamento de receitas referentes a determinados produtos.

4.7 Blocos de código habilitados para biblioteca

A criação de um programa do usuário pode ocorrer de forma linear ou estruturada. A **programação linear** escreve todo o programa do usuário no ciclo-OB, é adequada porém apenas para programas simples, em quais outros sistemas de controle mais baratos, por ex., LOGO! são usados.

Para programas mais complexos sugere-se sempre uma **programação estruturada**. Aqui toda a tarefa de automação pode ser dividida em pequenas tarefas, a fim de resolvê-las em funções e blocos de função.

Os blocos de código habilitados para biblioteca devem ser criados. Isto significa que os parâmetros de entrada e saída de uma função ou de um bloco de função serão definidos no geral e só fornecidos na utilização do módulo com a variante universal atual (entradas/saídas).

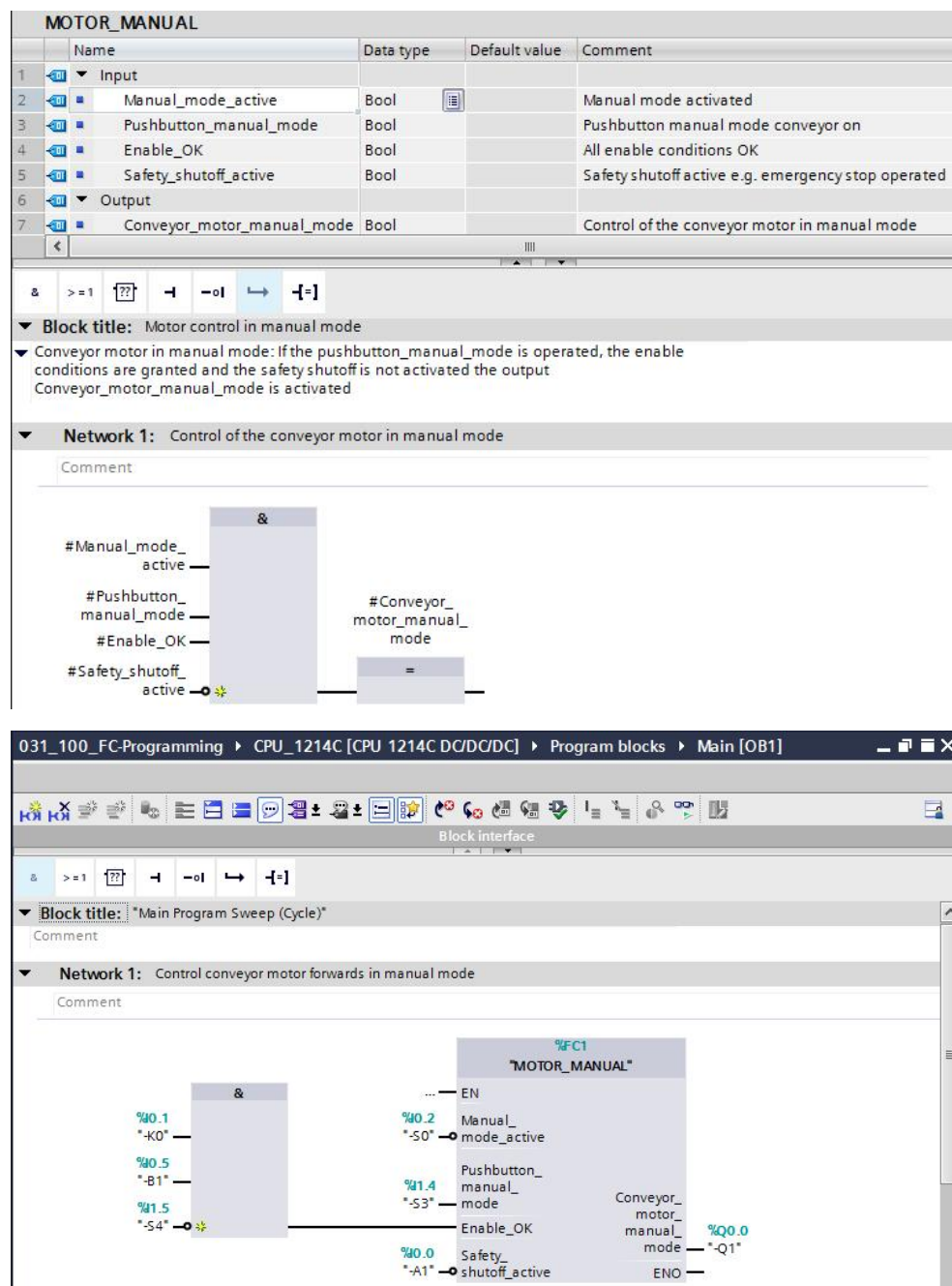


Imagem 6: Bloco de função habilitados com solicitação no OB1

4.8 Linguagens de programação

As linguagens (de programação) dispostas para programação de funções e blocos de função para o SIMATIC S7-1200 são o plano de função (FUP), plano de contato (KOP) e Structured Control Language (SCL).

A seguir, a linguagem de programação **plano de função(FUP)** será apresentada.

O FUP é uma linguagem de programação gráfica. A apresentação é modelada em sistemas de circuitos eletrônicos. O programa é mostrado nas redes. Uma rede contém um ou mais caminhos de ligação. Sinais binários e analógicos serão interligados através de boxes. Para apresentação da lógica binária serão utilizados os símbolos de lógica gráfica conhecidos a partir da álgebra booleana.

Com funções binárias você pode consultar os operandos binários e ligar os seus estados de sinal. Exemplos de funções binárias são as instruções "link E", "link OU" e "link OU EXCLUSIVO" como mostrado na Imagem 7.

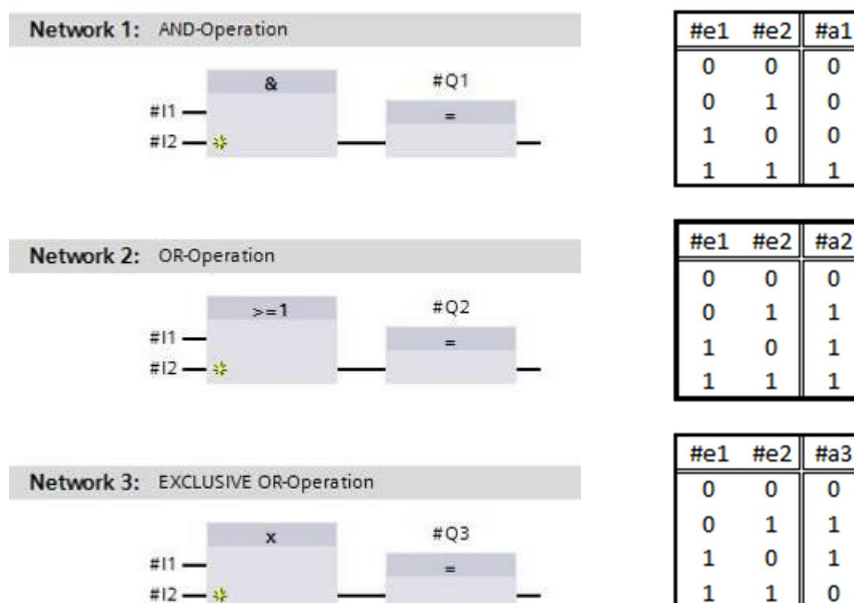


Imagem 7: Funções binárias no FUP e tabela de lógica correspondente

Com instruções simples, você pode controlar, por exemplo, as saídas binárias, avaliar flancos ou funções de salto executados no programa.

Instruções complexas dispõem elementos de programa, como por ex., os temporizadores IEC e contadores IEC.

O box vazio serve como marcador de posição, no qual você pode selecionar a instrução desejada.

Mecanismo de entrada de liberação EN (enable)/ saída de liberação ENO (enable output):

- Uma instrução sem mecanismo EN/ENO será efetuada independente do status de sinal nos box de entrada.
- Instruções com mecanismo EN/ENO só serão efetuadas se a entrada de liberação "EN" emitir o status de sinal "1". Com o processamento apropriado, o box emite à saída de liberação "ENO", o status de sinal "1". Se ocorrer um erro durante o processamento, a saída de liberação "ENO" será redefinida. Se a entrada de liberação EN não estiver conectada, o box será sempre executado.

5 Definição da tarefa

Neste capítulo, as seguintes funções da descrição do processo de classificação devem ser planejadas, programadas e testadas:

- operação manual – acionamento do movimento da correia para frente manualmente/ comando por impulsos

6 Planejamento

A programação de todas as funções no OB1 não é recomendável por razões de clareza e de reutilização. Por isto, o código de programa é complementado em grande parte nas funções (FCs) e blocos de funções (FBs). A decisão, sobre quais funções em FCs serão complementadas e quais terão lugar no OB1, será planejada desta forma.

6.1 PARADA DE EMERGÊNCIA

A PARADA DE EMERGÊNCIA não requer função própria. Bem como o modo de operação, o estado atual do relé de PARADA DE EMERGÊNCIA é usado diretamente sobre os blocos.

6.2 Operação manual – motor de correia no comando por impulsos

O comando por impulsos do motor de correia deve ser encapsulado em uma função (FC) "MOTOR_MANUAL". Assim, a clareza no OB1 é preservada e por outro lado, a reutilização é possível em uma extensão do sistema a uma outra esteira. Na Tabela 2 estão listados os parâmetros planejados.

Input	Tipo de dados	Comentário
operação manual_ativa	BOOL	Tipo de operação manual ativada
Tecla_comando por impulsos	BOOL	Tecla para ligar o motor de correia no comando por impulsos
Liberação_OK	BOOL	Todas as condições para liberação preenchidas
Desligamento de proteção_ativo	BOOL	Desligamento de proteção ativo, por ex., PARADA DE EMERGÊNCIA acionada
Output		
motor de correia_comando por impulsos	BOOL	acionar motor de correia no comando por impulsos

Tabela 2: parâmetro para o FC "MOTOR_manual"

A saída motor de correia_comando por impulsos está LIGADA, enquanto a tecla_comando por impulsos for pressionada, o tipo de operação manual estiver ativado, a liberação concedida e o desligamento de proteção não ativo.

6.3 Esquema de tecnologia

Aqui você pode ver o esquema de tecnologia para a tarefa.

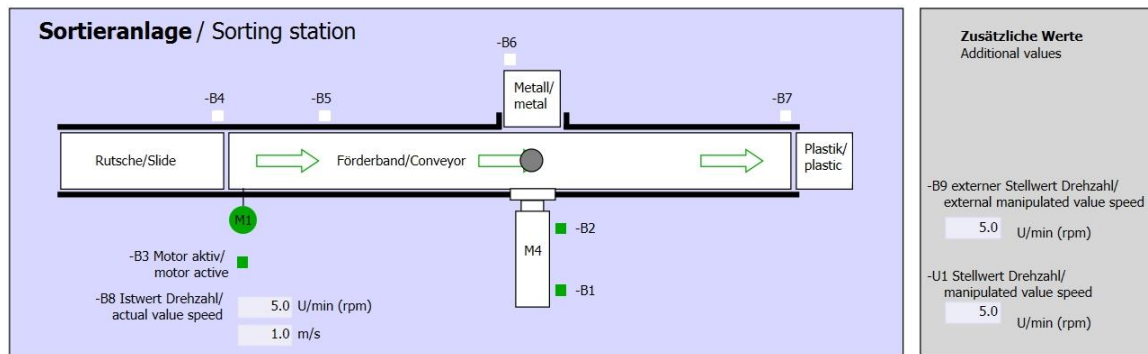


Imagem 8: esquema de tecnologia



Imagem 9: painel de controle

6.4 Tabela de atribuição

Os sinais a seguir serão utilizados como operando nesta tarefa.

DE	Tipo	Identificação	Função	NC/NO
E 0.0	BOOL	-A1	Mensagem PARADA DE EMERGÊNCIA ok	NC
E 0.1	BOOL	-K0	Equipamento "Ligado"	NO
E 0.2	BOOL	-S0	Chave seletora operação manual (0)/ automático(1)	manual = 0 auto=1
E 0.5	BOOL	-B1	Cilindro de sensor -M4 retraído	NO
E 1.4	BOOL	-S3	Tecla comando por impulsos correia -M1, movimento para frente	NO
E 1.5	BOOL	-S4	Tecla comando por impulsos correia -M1, movimento para trás	NO

DA	Tipo	Identificação	Função	
A 0.0	BOOL	-Q1	motor de correia -M1 velocidade fixa para frente	

Legenda da lista de atribuição

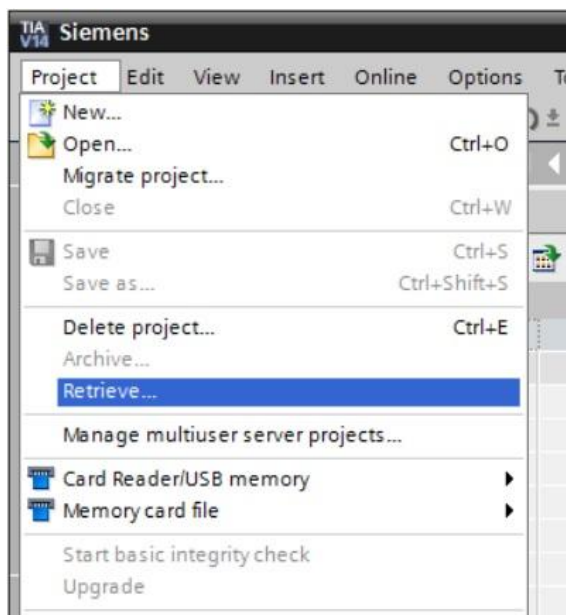
DE	entrada digital	DA	saída digital
AE	entrada analógica	AA	saída analógica
E	entrada	A	saída
NC	normally closed (contato normalmente fechado)		
NO	normally open (contato normalmente aberto)		

7 Instrução estruturada passo a passo

A seguir, você verá uma instrução de como implementar o planejamento. Se você já lida bem com assunto, os passos enumerados já serão suficientes para o procedimento. Se isto não for o caso, basta os passos detalhados seguir da instrução.

7.1 Extrair arquivo de um projeto atual

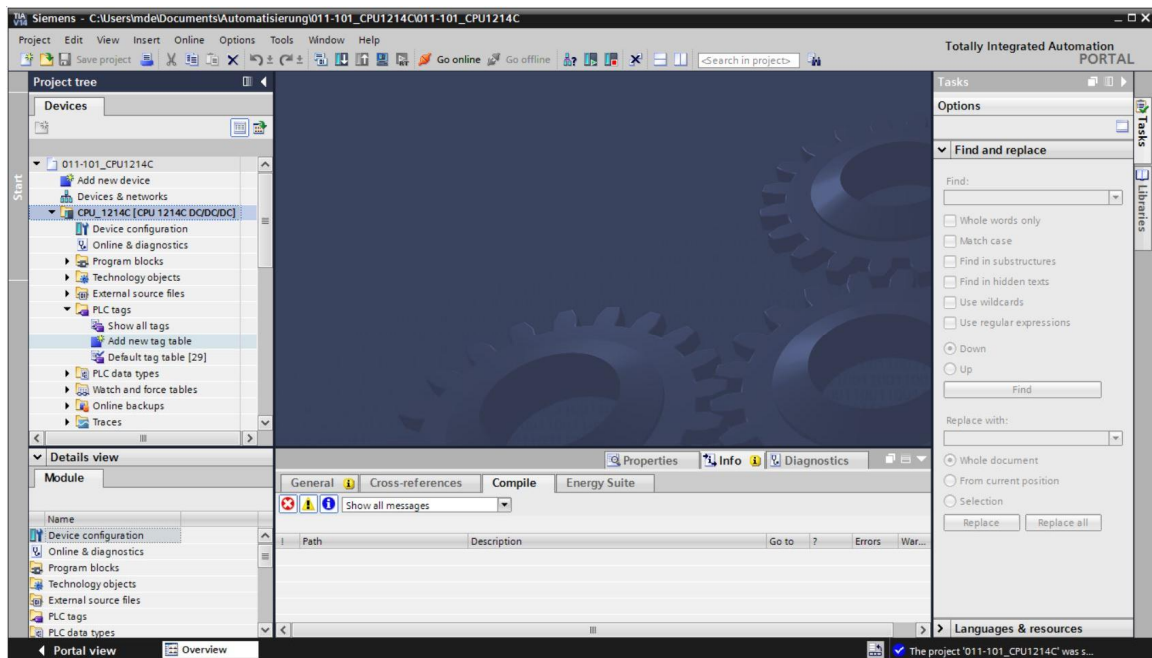
- ® Antes de iniciarmos com a programação da função (FC) "MOTOR_manual", precisaremos de um projeto com uma configuração do hardware. (por ex., SCE_PT_011_101_ Configuração do hardware _CPU1214C.zap14). Para extrair do arquivo de um projeto atual, você deve procurar na visualização do projeto em ® Project (Projeto) ® Retrieve (Extrair) o arquivo correspondente. Confirme sua seleção em seguida com Open (Abrir). (® Project (Projeto) ® Retrieve (Extrair) ® Selection of a .zap archive (Seleção de uma arquivo .zap) ® Open (Abrir).)



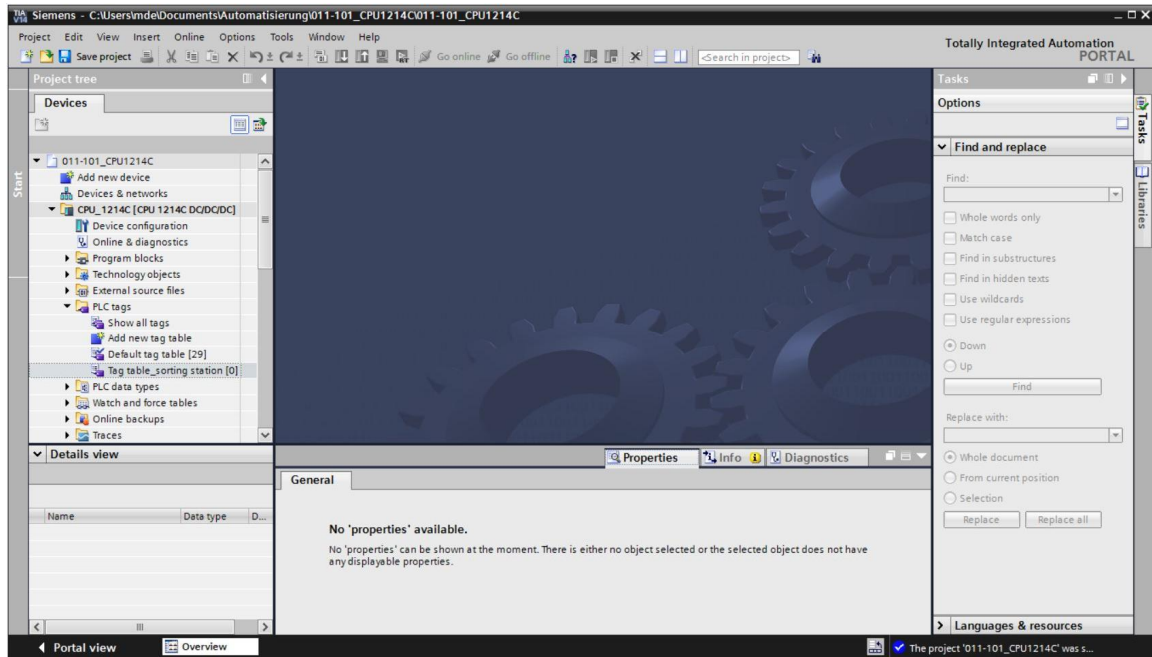
- ® Depois, é possível escolher o diretório de destino, em qual o projeto descompactado deverá ser salvo. Confirme sua seleção com "OK". (® Target directory (Diretório de destino) ® OK)

7.2 Criar tabela de variantes

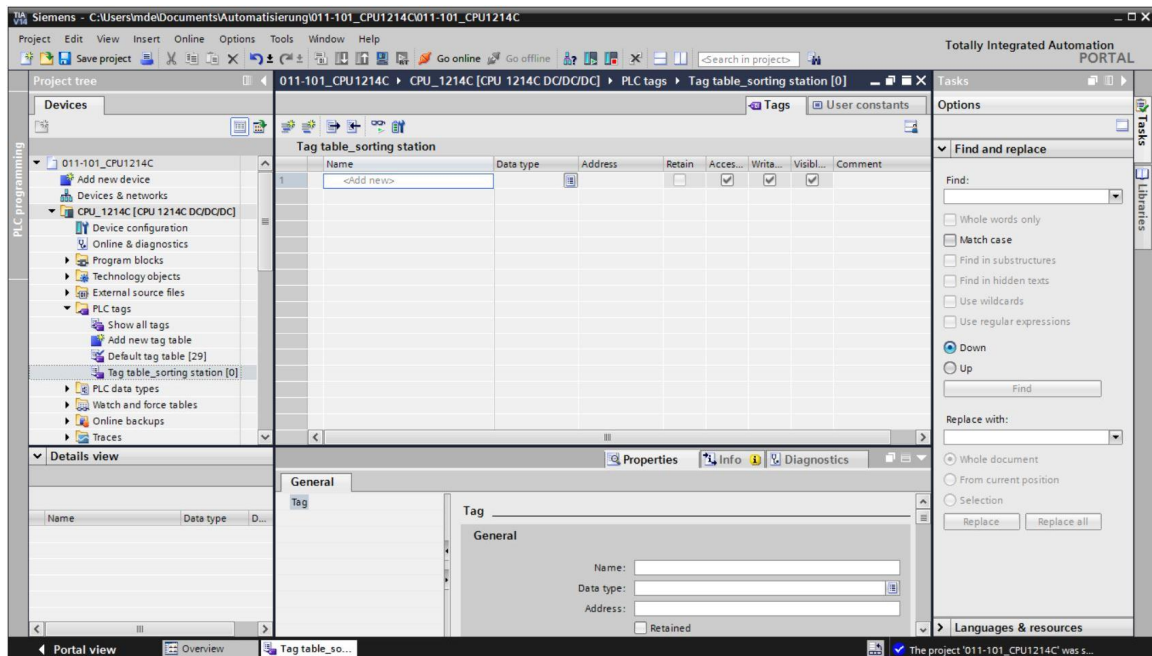
- ® Navegue na visualização do projeto até variantes ® PLC de seu controller e crie uma tabela de variantes, clicando duas vezes em ® Add new tag table (Adicionar nova tabela de variantes) "clique duplo".



- ® Nomeie a tabela de variantes recém-criada como "Tag table_sorting station" (Tabela de variantes_sistema de classificação). (® Clique no lado direito em "Tag table_1" (Tabela de variantes_1) ® "Rename" (Renomear) ® Tag table_sorting station (Tabela de variantes_sistema de classificação))

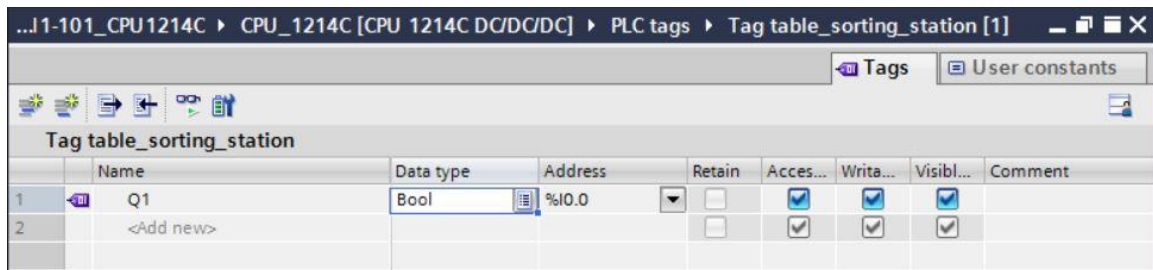


- ® Abra esta e com duplo clique. (® Tag table_sorting station (Tabela de variantes_sistema de classificação))

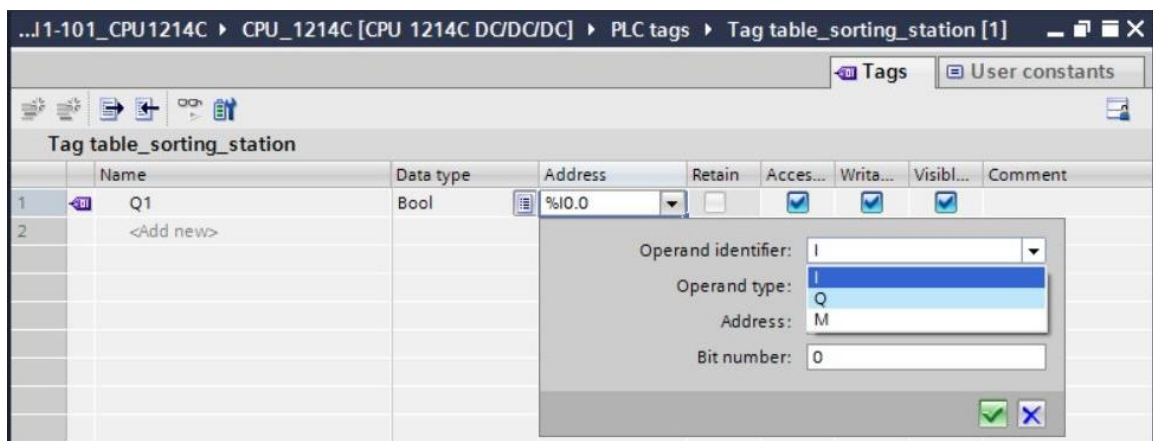


7.3 Criar variáveis dentro de uma tabela de variáveis

- ® Adicione o nome Q1 e confirme a entrada com a tecla Enter. Se você não criar nenhuma outra variante, o TIA Portal irá fornecer automaticamente o tipo de dado "Bool" e o endereço %E0.0 (I 0.0). (® <Add new> (<Adicionar>) ® Q1 ® Enter)



- ® Mude o endereço para %A0.0 (Q0.0), inserindo diretamente ou rolando a seta para baixo e clicando para abrir o menu para endereçamento. Mude a identificação de operando para Q e confirme com Enter ou clicando na marca de seleção. (® %E0.0 ® Operand identifier (Indicação de operação) ® Q ® ☒)

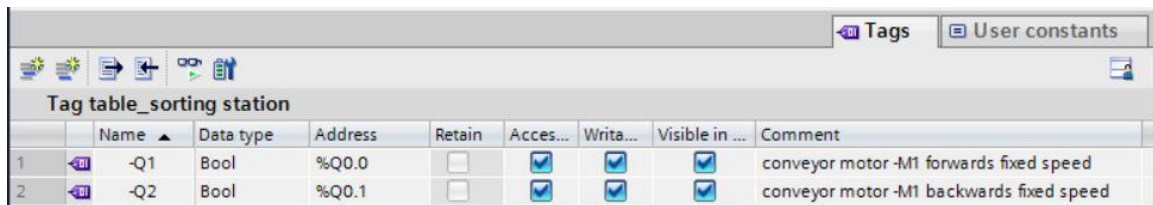


- ® Faça o comentário para a variante "conveyor motor -M1 forwards fixed speed" (motor de correia -M1 velocidade fixa de movimento para frente).



- ® Adicione na coluna 2 uma nova variante Q2. TIA Portal atribuiu automaticamente o mesmo tipo de dados, como na coluna 1, e o endereço de 1 aumentado para %A0.1 (Q0.1). Faça o comentário "conveyor motor -M1 backwards fixed speed" (motor de correia -M1 velocidade fixa de movimento para trás).

(® <Add new> (<Adicionar>) ® Q2 ® Enter ® Comentário ® conveyor motor -M1 backwards fixed speed (motor de correia -M1 velocidade fixa de movimento para trás)

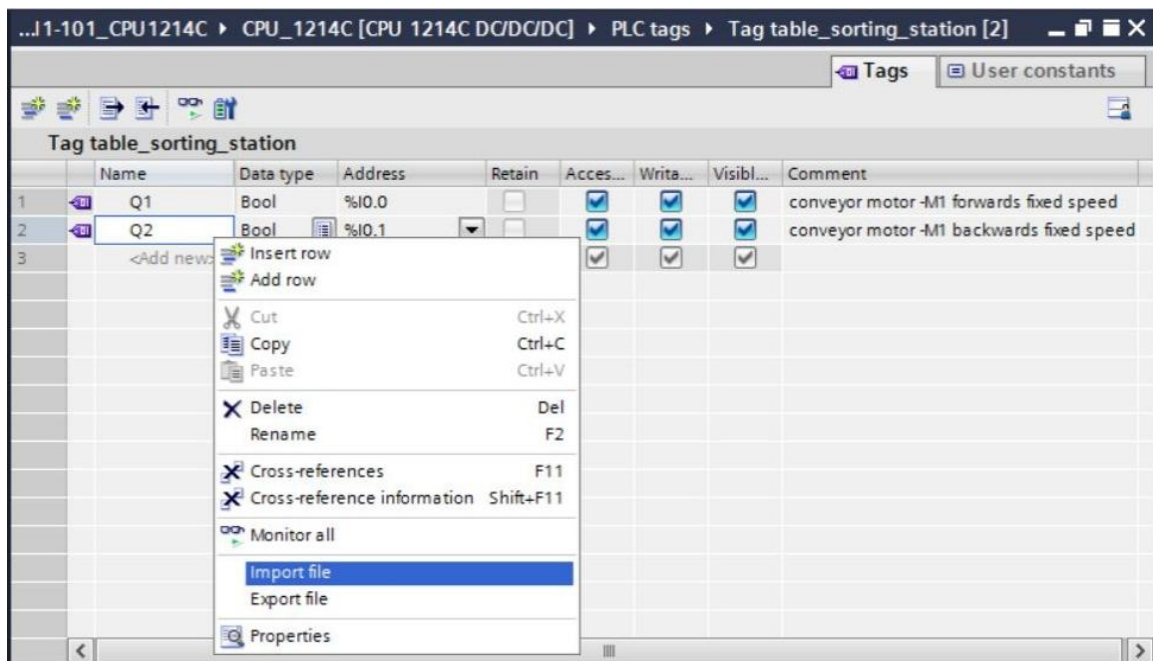


	Name	Data type	Address	Retain	Acces...	Writa...	Visible in ...	Comment
1	-Q1	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	conveyor motor -M1 forwards fixed speed
2	-Q2	Bool	%Q0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	conveyor motor -M1 backwards fixed speed

7.4 Importação da "Tag table_sorting station" (Tabela de variáveis_sistema de classificação)

- ® Para inserir uma tabela de símbolos existente, clique com o botão direito do mouse sobre um espaço vazio da "Tag table_sorting station" (Tabela de variáveis_sistema de classificação) aplicada. No menu de contexto, selecione "Import file" (Arquivo de importação).

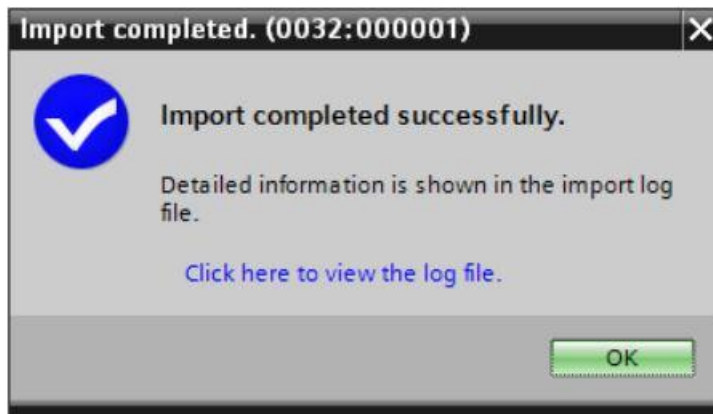
(® Clique no lado direito em um espaço vazio da tabela de variáveis ® Import file (Arquivo de importação))



- ® Selecione a tabela de símbolo desejada (por ex., no formato .Xlsx) e confirme a seleção com "Open" (Abrir).

(® SCE_PT_020-100_tabela de variáveis_sistema de classificação...® Open (Abrir))

- ® Com a importação concluída, você receberá uma janela de confirmação que permite ver o arquivo de protocolo de importação. Clique aqui em ® OK.



- ® Você verá que alguns endereços foram destacados em laranja. Eles estão duplicados e os nomes das suas variáveis são numerados automaticamente, para evitar ambiguidade.
- ® Delete a variável duplicada selecionando a coluna e pressione a tecla Delete no teclado ou selecione a opção "Delete" (Excluir) no menu de contexto.
- (® Clique no lado direito nas variáveis marcadas ® Delete (Excluir))

...1-101_CPU1214C > CPU_1214C [CPU 1214C DC/DC/DC] > PLC tags > Tag table_sorting station [30]

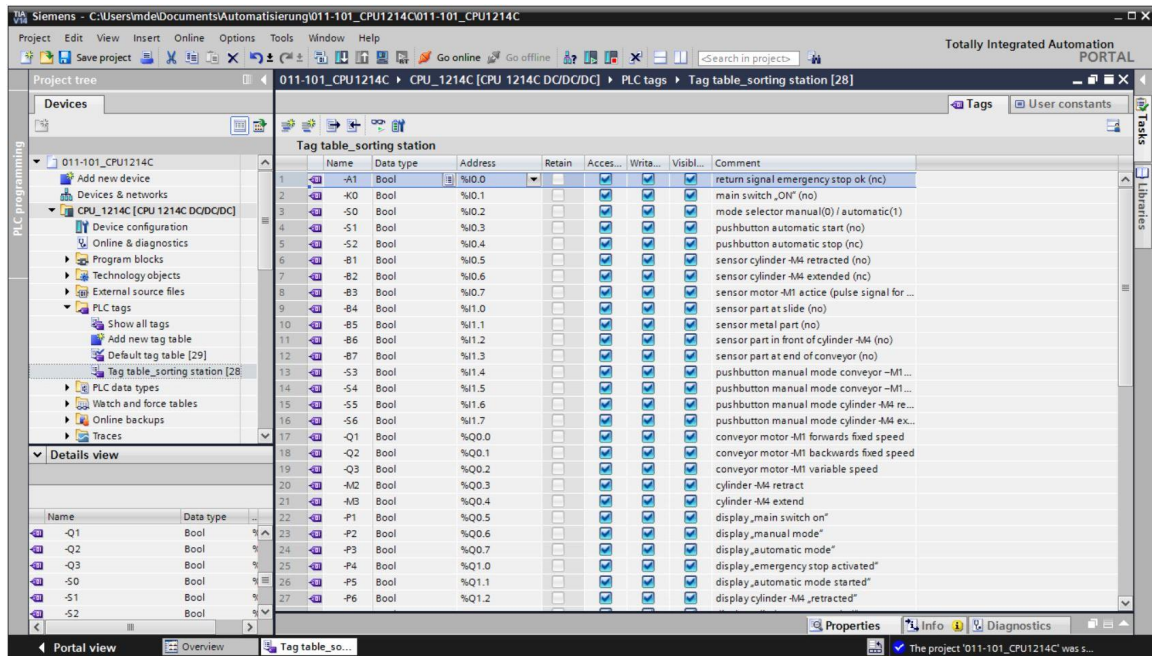
Tags User constants

Tag table_sorting station

	Name	Data type	Address	Retain	Acces...	Writa...	Visibl...	Comment
1	Q1	Bool	%Q0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	conveyor motor -M1 forwards fixed speed
2	Q2	Bool	%Q0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	conveyor motor -M1 forwards fixed speed
3	-A1	Bool	%I0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	return signal emergency stop ok (nc)
4	-K0	Bool	%I0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	main switch „ON“ (no)
5	-S0	Bool	%I0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	mode selector manual(0) / automatic(1)
6	-S1	Bool	%I0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	pushbutton automatic start (no)
7	-S2	Bool	%I0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	pushbutton automatic stop (nc)
8	-B1	Bool	%I0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	sensor cylinder -M4 retracted (no)
9	-B2	Bool	%I0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	sensor cylinder -M4 extended (nc)
10	-B3	Bool	%I0.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	sensor motor -M1 actice (pulse signal for ...
11	-B4	Bool	%I1.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	sensor part at slide (no)
12	-B5	Bool	%I1.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	sensor metal part (no)
13	-B6	Bool	%I1.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	sensor part in front of cylinder -M4 (no)
14	-B7	Bool	%I1.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	sensor part at end of conveyor (no)
15	-S3	Bool	%I1.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	pushbutton manual mode conveyor -M1...
16	-S4	Bool	%I1.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	pushbutton manual mode conveyor -M1...
17	-S5	Bool	%I1.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	pushbutton manual mode cylinder -M4 re...
18	-S6	Bool	%I1.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	pushbutton manual mode cylinder -M4 ex...
19	-Q1	Bool	%Q0.0		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	conveyor motor -M1 forwards fixed speed
20	-Q2	Bool	%Q0.1		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	conveyor motor -M1 backwards fixed speed
21	-Q3	Bool	%Q0.2		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	conveyor motor -M1 variable speed
22	-M2	Bool	%Q0.3		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	cylinder -M4 retract
23	-M3	Bool	%Q0.4		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	cylinder -M4 extend
24	-P1	Bool	%Q0.5		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	display „main switch on“
25	-P2	Bool	%Q0.6		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	display „manual mode“
26	-P3	Bool	%Q0.7		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	display „automatic mode“


® Você tem agora uma tabela de símbolos completa de entradas e saídas digitais. salve seu projeto sob o nome 031-100_ Programação-FC.

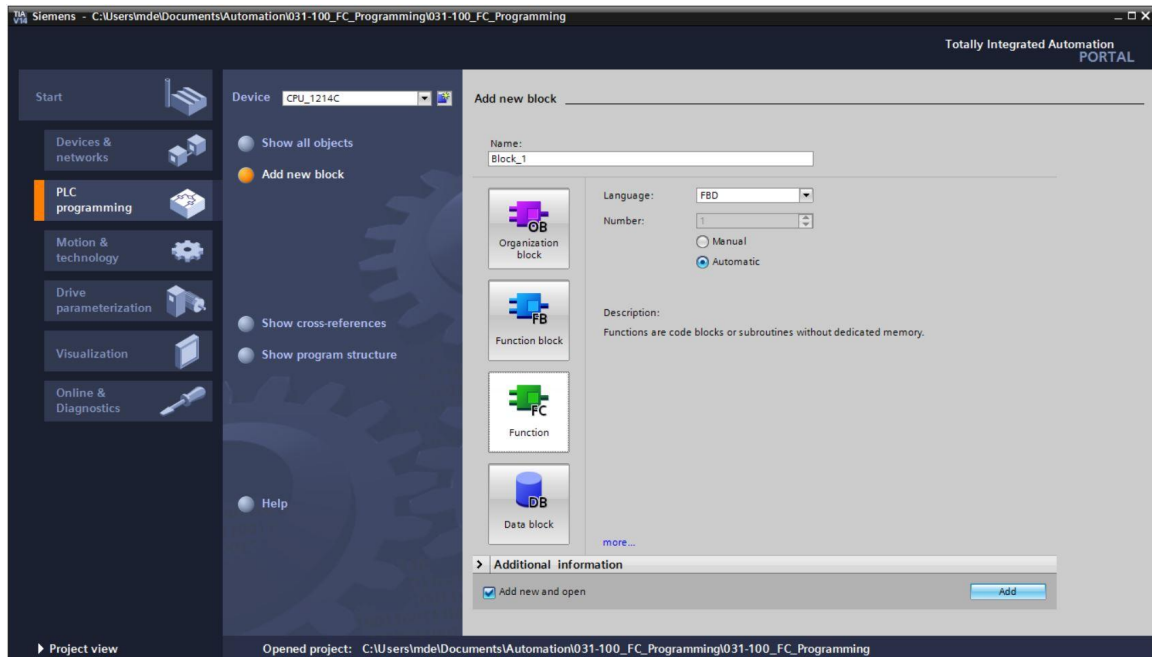
(® Project (Projeto) ® Save as ... (Salvar em ...) ® 031-100_ Programação-FC ® Save (Salvar))



7.5 Criar função FC1 "MOTOR_manual" para o motor de correia no comando por impulsos

- ® Clique no Portal, na seção Programação PLC em "Add new block" (Adicionar novo bloco), para criar aqui uma função.

(® PLC programming (Programação PLC) ® Add new block (Adicionar novo bloco) ® )



- ® Nomeie seu novo bloco com o nome: "MOTOR_manual", defina o idioma no FBD e deixe que o número seja atribuído automaticamente. Ative a marca de verificação "Add new and open" (Adicionar novo bloco e abrir), e automaticamente você chegará na visualização do projeto do seu bloco de função criado. Clique em "Add" (Adicionar). (® Name (Nome): MOTOR_manual® Language (Idioma): FBD ® Number (Número): automaticamente, ® ☒ Add new and open (Adicionar novo bloco e abrir) ® Add (Adicionar))

Add new block

Name:

Language:

Number:

☐ Manual
☒ Automatic

Description:
Functions are code blocks or subroutines without dedicated memory.

[More...](#)

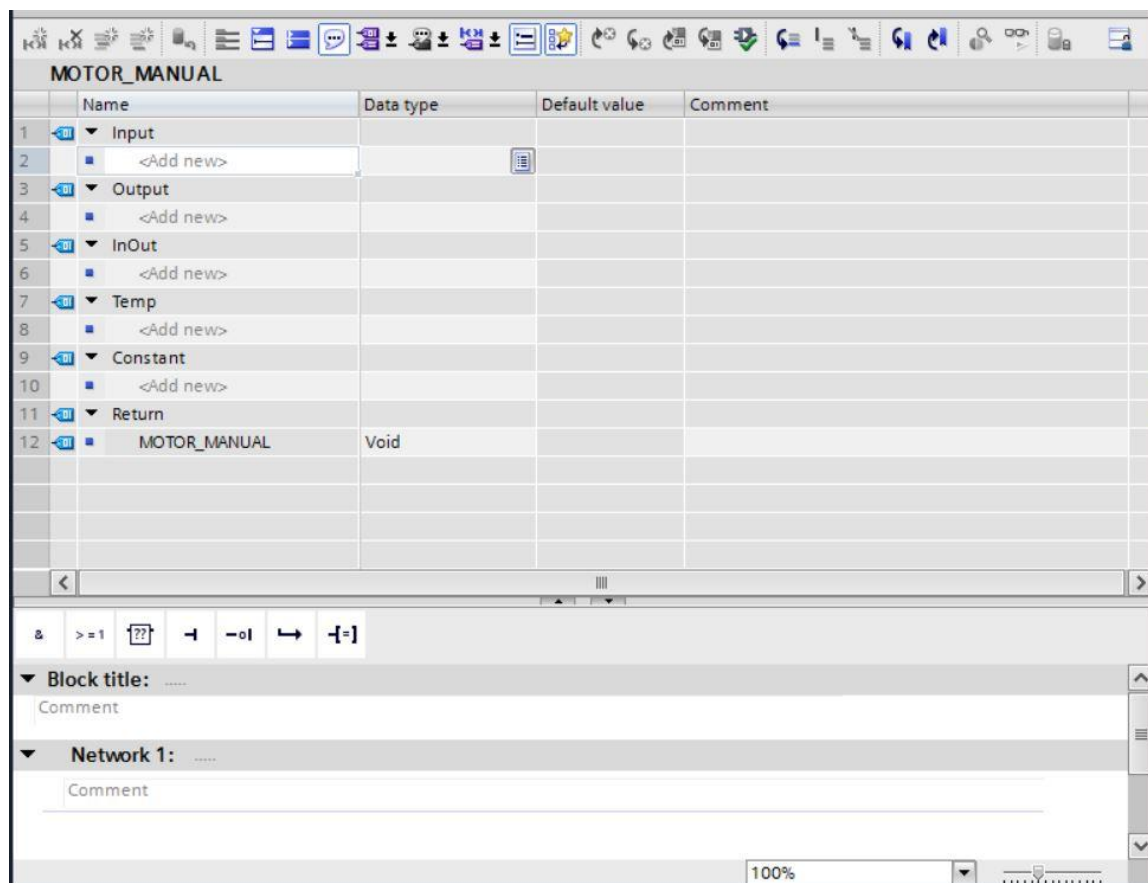
> **Additional information**

☒ Add new and open

7.6 Definir interface da função FC1 "MOTOR_manual"

Clicando em "Add new and open" (Adicionar novo bloco e abrir)", a visualização do projeto abre com uma janela para configuração do bloco recém-criado.

- ® Na parte superior da visualização do programa, você encontrará a descrição da interface da devida função.



- ® Para o acionamento do motor de correia será necessário um sinal de saída binário. Por isso, criamos primeiro a variável output #Conveyor_motor_manual_mode (Motor de correia_comando por impulsos) do tipo "Bool". Para o parâmetro, faça o comentário "acionar motor de correia no comando por impulsos".

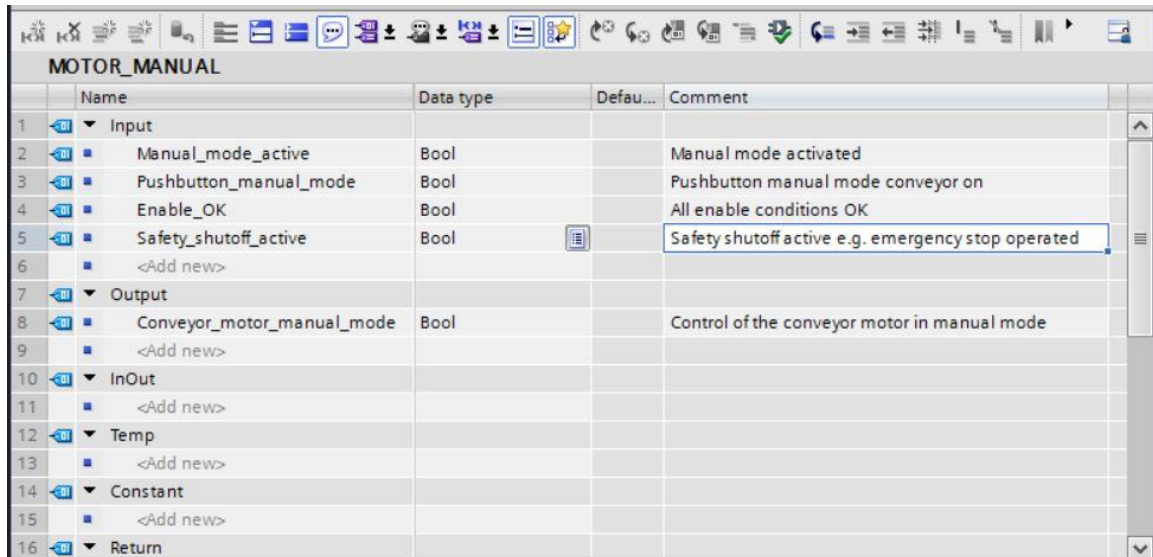
(® Output: Conveyor_motor_manual_mode (Motor de correia_comando por impulsos ® Bool ® Control of the conveyor motor in manual mode (Acionar motor de correia no comando por impulsos))

MOTOR_MANUAL				
	Name	Data type	Default value	Comment
1	▼ Input			
2	■ <Add new>			
3	▼ Output			
4	■ Conveyor_motor_manual_mode	Bool		Control of the conveyor motor in manual mode
5	■ <Add new>			
6	▼ InOut			
7	■ <Add new>			
8	▼ Temp			
9	■ <Add new>			
10	▼ Constant			
11	■ <Add new>			
12	▼ Return			
13	■ MOTOR_MANUAL	Void		

- ® Adicione primeiro como interface de entrada em Input o parâmetro #Manual_mode_active (Operação_manual_ativo) e confirme a entrada com Enter ou deixando o campo de entrada. O tipo de dados "Bool" é atribuído automaticamente. Este é mantido. Faça em seguida o devido comentário "Manual mode activated" (Tipo de operação manual ativado).

(® Manual_mode_active (Operação_manual_ativo ® Enter ® Bool ® Manual mode activated (Tipo de operação manual ativado))

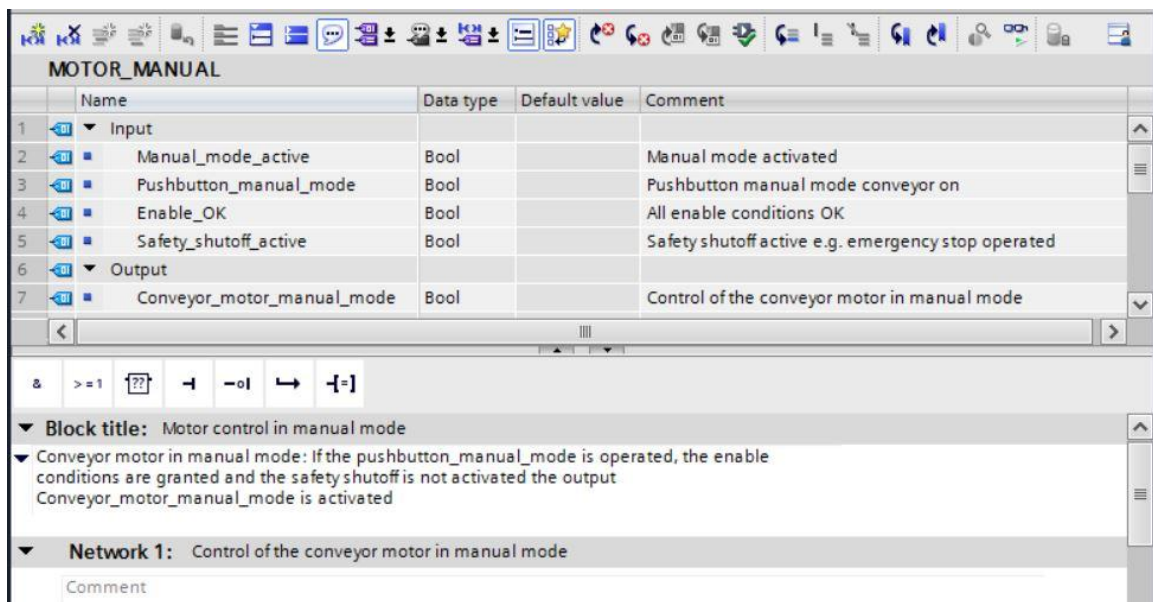
- ® Em seguida, adicione como Input outros parâmetro de entrada binários #Pushbutton_manual_mode (Tecla_comando por impulsos), #Enable_OK (Liberação_OK) e #Safety_shutoff_active (Desligamento de proteção_ativo) e verifique os tipos de dados deles. Complemente com comentários significativos.



	Name	Data type	Defau...	Comment
1	Input			
2	Manual_mode_active	Bool		Manual mode activated
3	Pushbutton_manual_mode	Bool		Pushbutton manual mode conveyor on
4	Enable_OK	Bool		All enable conditions OK
5	Safety_shutoff_active	Bool		Safety shutoff active e.g. emergency stop operated
6	<Add new>			
7	Output			
8	Conveyor_motor_manual_mode	Bool		Control of the conveyor motor in manual mode
9	<Add new>			
10	InOut			
11	<Add new>			
12	Temp			
13	<Add new>			
14	Constant			
15	<Add new>			
16	Return			

- ® Atribua à documentação do programa o título do bloco, comentário de bloco e para a rede 1 um título rede útil.

(® Block title (Título do bloco): Conveyor motor in manual mode (Acionamento de motor em operação manual) ® Network 1 (Rede 1): acionar motor de correia no comando por impulsos)



	Name	Data type	Default value	Comment
1	Input			
2	Manual_mode_active	Bool		Manual mode activated
3	Pushbutton_manual_mode	Bool		Pushbutton manual mode conveyor on
4	Enable_OK	Bool		All enable conditions OK
5	Safety_shutoff_active	Bool		Safety shutoff active e.g. emergency stop operated
6	Output			
7	Conveyor_motor_manual_mode	Bool		Control of the conveyor motor in manual mode

& >=1 [??] -o| -| =]

▼ **Block title:** Motor control in manual mode

▼ Conveyor motor in manual mode: If the pushbutton_manual_mode is operated, the enable conditions are granted and the safety shutoff is not activated the output Conveyor_motor_manual_mode is activated

▼ **Network 1:** Control of the conveyor motor in manual mode

Comment

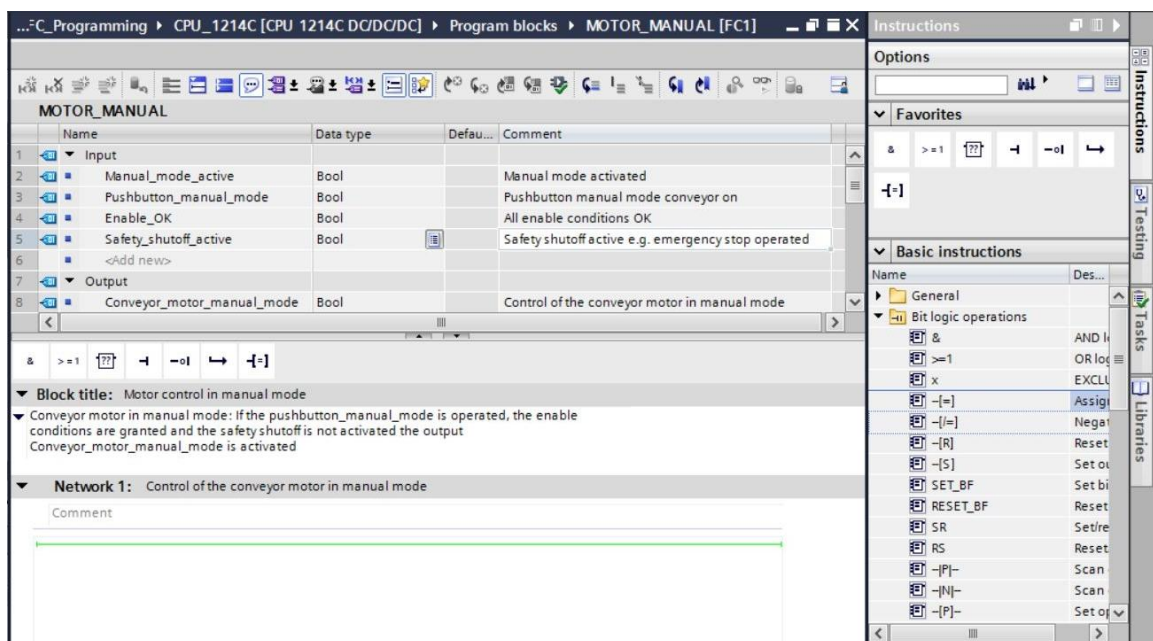
7.7 Programação do FC1: MOTOR_MANUAL

® Abaixo da descrição de interface, você verá uma barra de ferramentas com funções de lógica diferente na janela de programação inclusive uma área de redes. Neste local já definimos o título de bloco e o título da primeira rede. Dentro da rede ocorre a programação com a utilização de cada bloco de lógica. Uma distribuição de várias redes serve para manter a clareza. A seguir você vai aprender as diferentes maneiras de inserir blocos de lógica.



® No lado direito da sua janela de programação há uma lista de instruções que você pode usar no programa. Busque em ® Basic instructions (Instruções simples) ® conexões de bit conforme a função := (Atribuição) e arraste estas para sua rede 1 (linha verde aparece, mouse com símbolo +).

(® Instructions (Instruções) ® Basic instructions (Instruções simples) ® Bit logic operation (Conexão bit) ® :=)



- ® Arraste agora seu parâmetro de output #Conveyor_motor_manual_mode (Motor de correia_comando por impulsos) para **<???.?>** sobre seu bloco recém-adicionado. Você pode selecionar melhor um parâmetro na descrição de interface acionando no símbolo azul.

(® Conveyor_motor_manual_mode (Motor de correia_comando por impulsos))

	Name	Data type	Defau...	Comment
1	Input			
2	Manual_mode_active	Bool		Manual mode activated
3	Pushbutton_manual_mode	Bool		Pushbutton manual mode conveyor on
4	Enable_OK	Bool		All enable conditions OK
5	Safety_shutoff_active	Bool		Safety shutoff active e.g. emergency stop operated
6	<Add new>			
7	Output			
8	Conveyor_motor_manual_mode	Bool		Control of the conveyor motor in manual mode


Block title: Motor control in manual mode

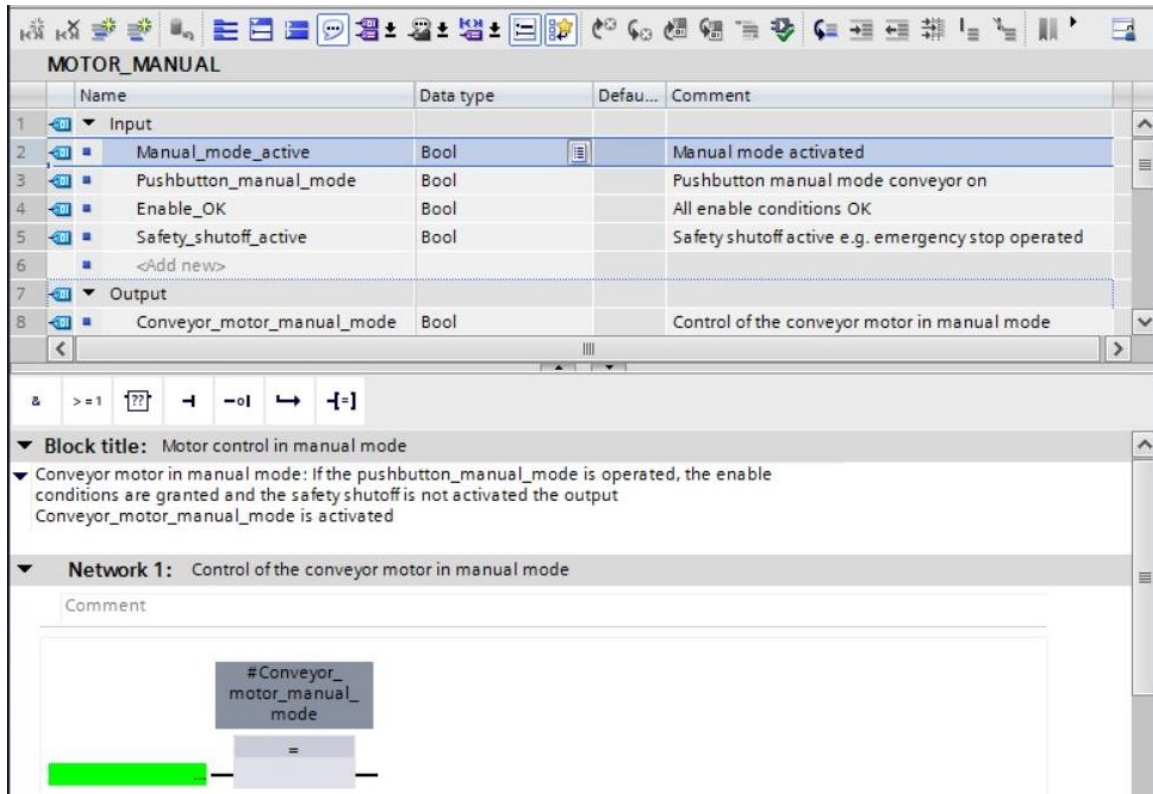
Conveyor motor in manual mode: If the pushbutton_manual_mode is operated, the enable conditions are granted and the safety shutoff is not activated the output Conveyor_motor_manual_mode is activated

Network 1: Control of the conveyor motor in manual mode

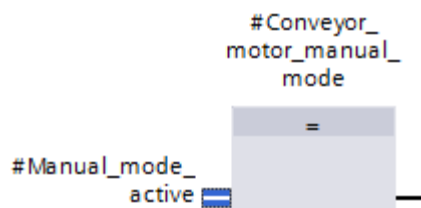
Comment


- ® Isto determina que o parâmetro #Conveyor_motor_manual_mode (Motor de correia_comando por impulsos) seja escrito por este bloco. Faltam ainda as condições de entrada para isso realmente acontecer. Arraste o parâmetro de input #Manual_mode_active (Operação manual_ativo) e solte "... " no lado esquerdo do bloco de alocação.

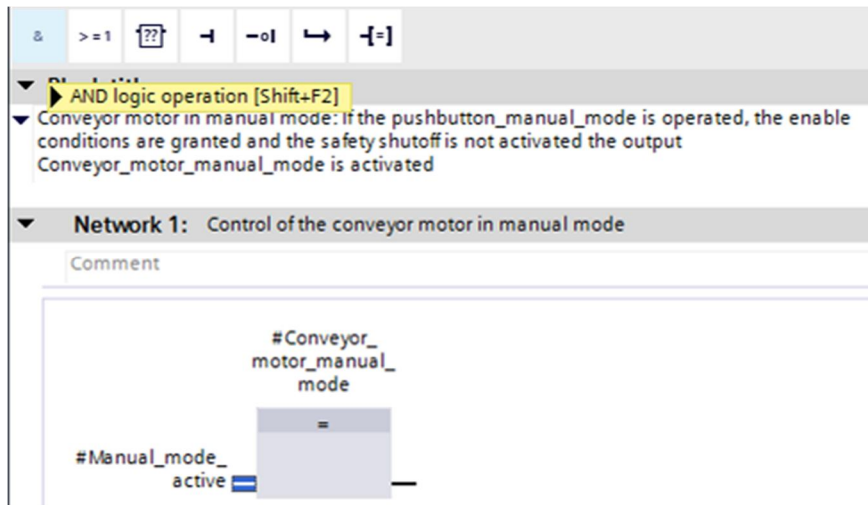
(®  Manual_mode_active (Operação manual_ativo))



- ® A entrada do bloco de alocação deve ser adicionalmente interligada com outros parâmetros. E. Clique primeiro na entrada do bloco, onde a # operação manual_ativa está conectada, para que a linha de entrada fique azul.

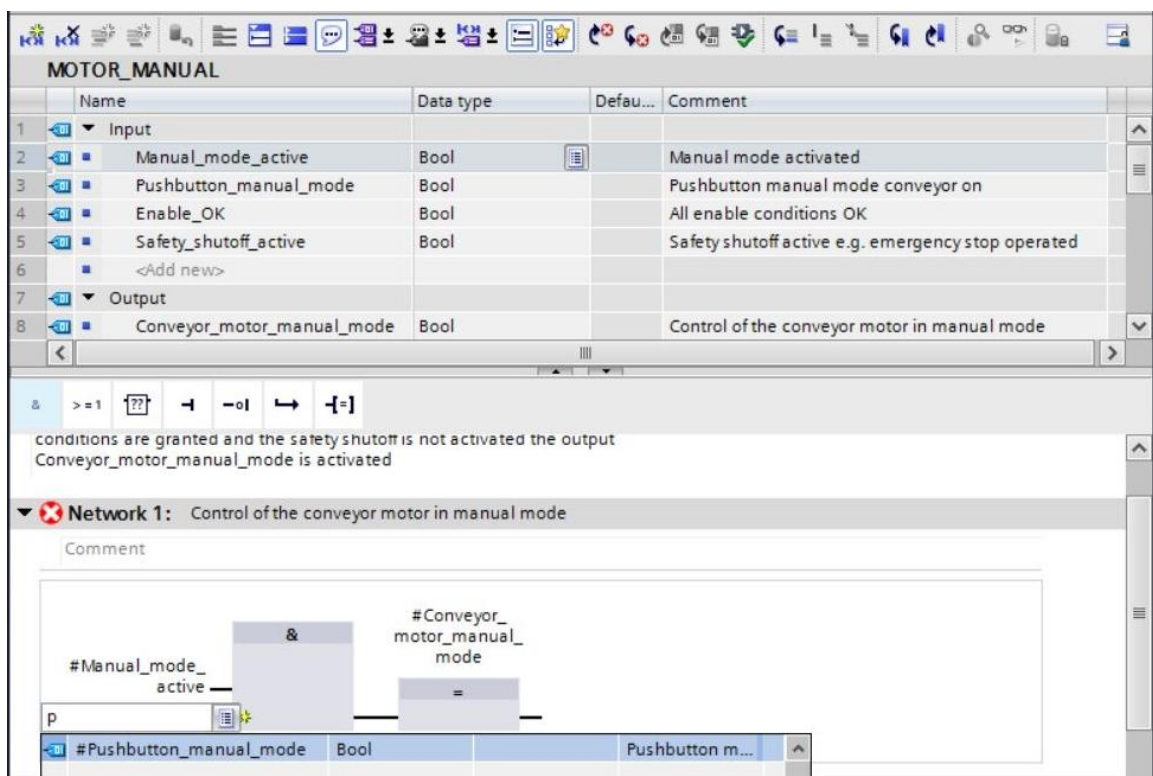


- ④ Clique no símbolo  sua barra de ferramentas lógica, para adicionar uma conexão E entre a variável #Manual_mode_active (Operação manual_ativo) e seu bloco de alocação.




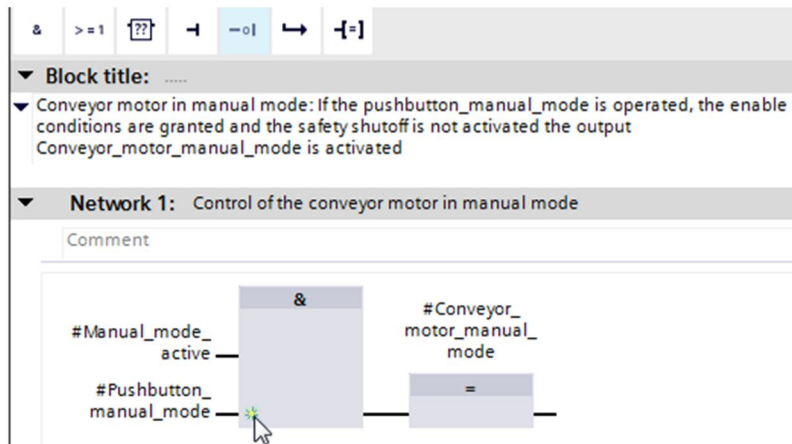
- ④ Clique duas vezes na segunda entrada da conexão & <??.> e digite a letra "T" na caixa que aparece, para ver uma lista de variáveis disponíveis, que começam com "T". Clique na variável #Pushbutton_manual_mode (Tecla_comando por impulsos) e confirme com ④ Enter.

(④ Block & (Bloco &) ④ <??.> ④ T ④ #Pushbutton_manual_mode (Tecla_comando por impulsos) ④ Enter)

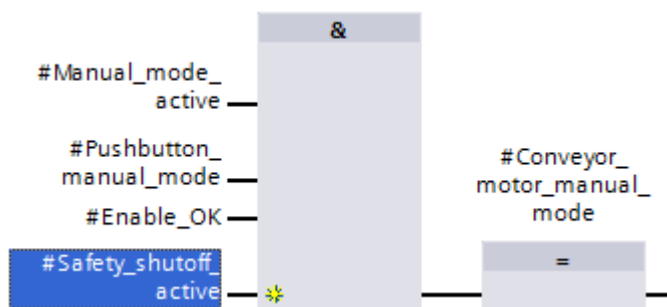



Indicação: Nesta variante da ordem de variáveis, há o risco de uma troca com a variável global da tabela de variáveis. Portanto, a versão mostrada anteriormente ao arrastar e soltar da descrição da interface deve ser preferida.

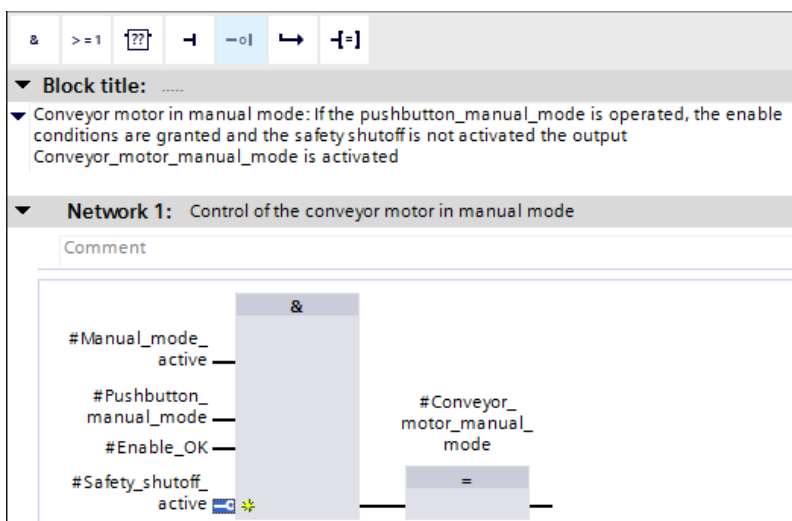
- ® Para que a saída só seja acionada se a liberação for concedida e o desligamento de proteção não fique ativo, as variáveis de entrada #Enable_OK (Liberação_OK) e #Safety_shutoff_active (Desligamento de proteção_ativo) devem ser conectadas adicionalmente com o E. Clique duas vezes sobre a estrela amarela  para adicionar ao seu elemento E mais duas entradas.




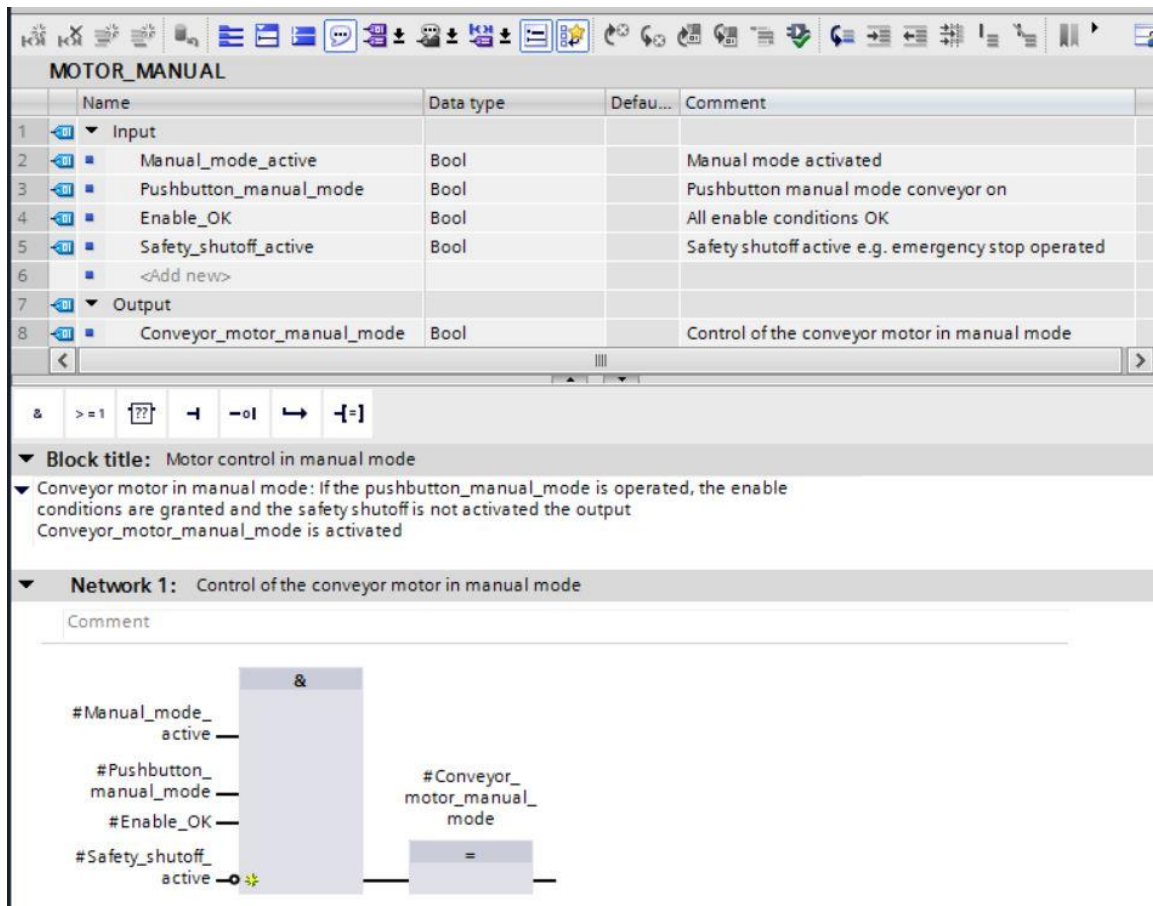
- ® Adicione às variáveis de entrada do elemento E, as variáveis de entrada #Enable_OK (Liberação_OK) e #Safety_shutoff_active (Desligamento de proteção_ativo).



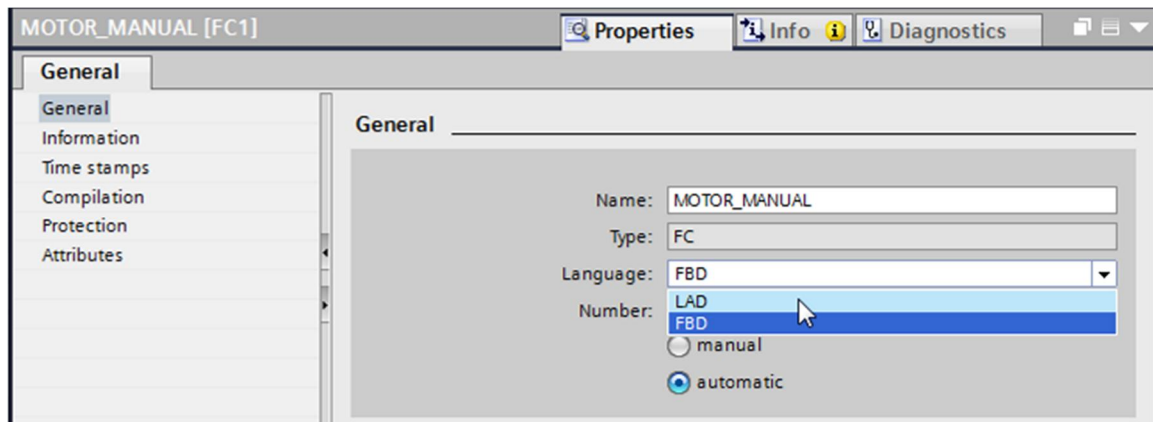
- ® Anule a entrada ligada com o parâmetro #Safety_shutoff_active (Desligamento de proteção_ativo), marcando e em seguida clicando em .



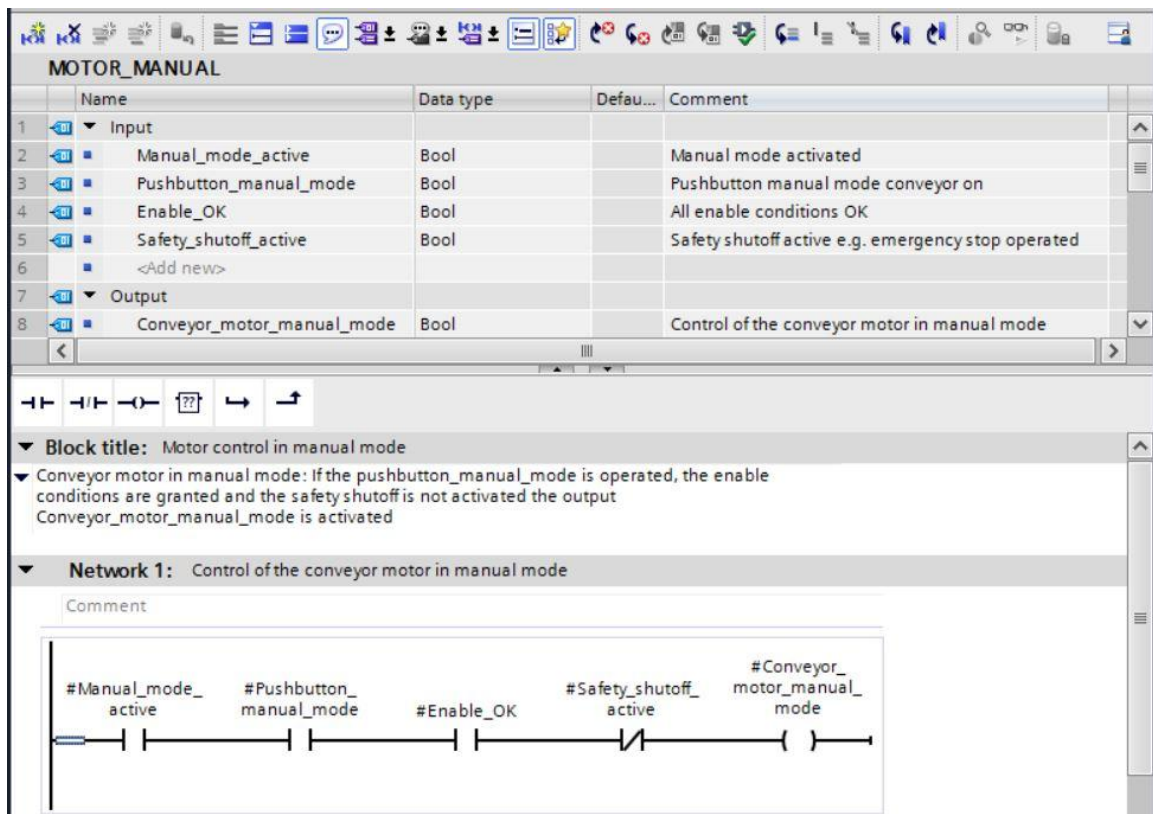
- ® Não se esqueça de clicar regularmente em  **Save project**. A função concluída "MOTOR_manual [FC1] no FBD" será mostrada desta forma.



- ® Em propriedades do bloco, você pode mudar no item "General" (Gerais) o "Language" (Idioma) em LAD (plano de contato). (® Properties (Propriedades) ® General (Gerais) ® Language (Idioma): KOP)



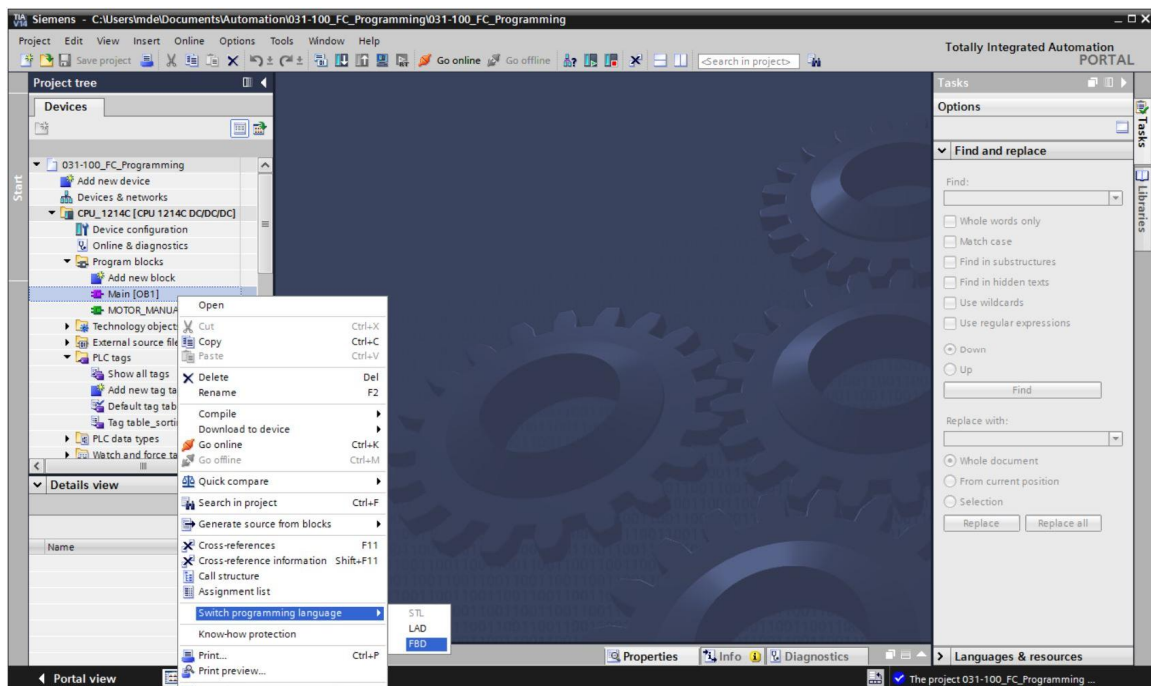
- ® No KOP, o programa é mostrado desta forma.



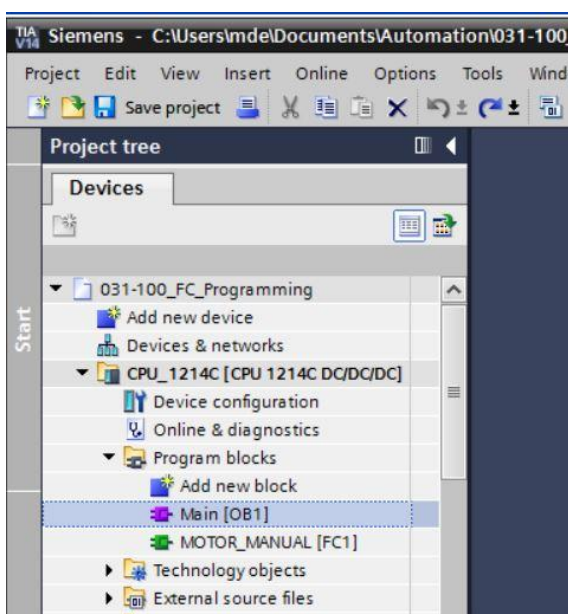
7.8 Programação do bloco de organização OB1 – controller do movimento da correia para frente na operação manual

- ® Antes da programação do bloco de organização "Main[OB1]", iremos configurar a linguagem de programação no FBD (plano de função). Clique primeiro aqui no lado esquerdo do mouse, na pasta "Program blocks" (Blocos de programa) em "Main[OB1]".

(Mudar® CPU_1214C [CPU 1214C DC/DC/DC] ® Program blocks" (Blocos de programa) ® Main [OB1] ® Switch programming language (Linguagem de programação) ® FBD)

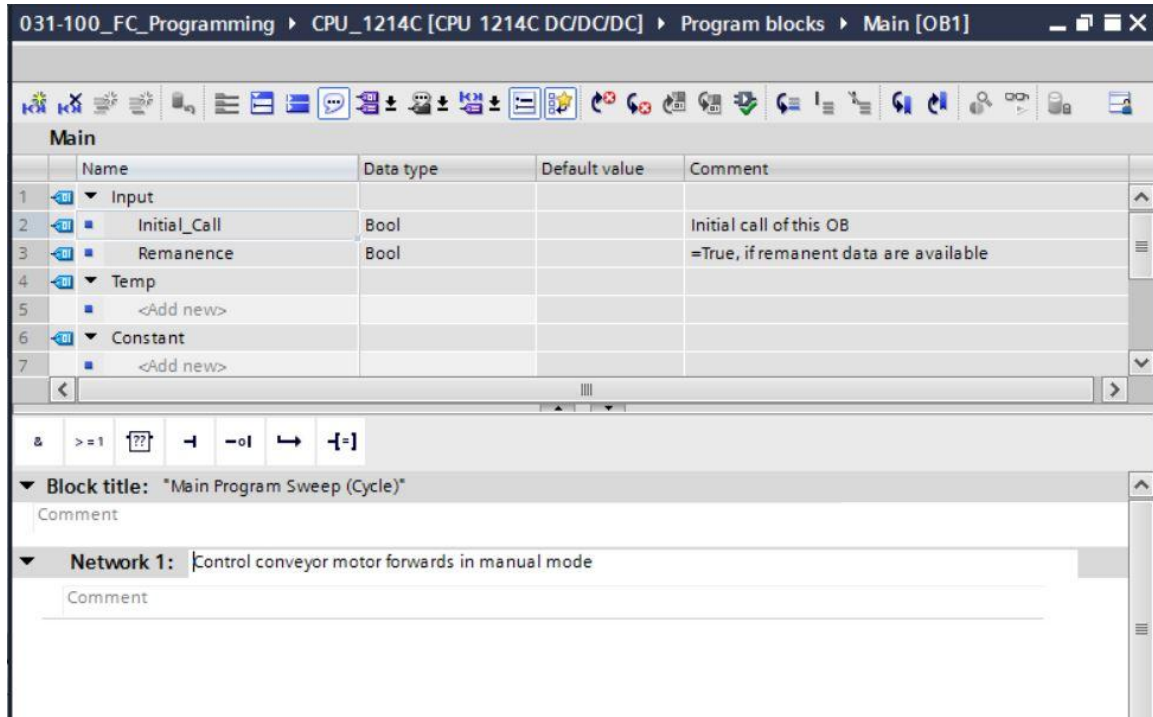


- ® Abra o bloco de organização "Main [OB1]" com duplo clique.

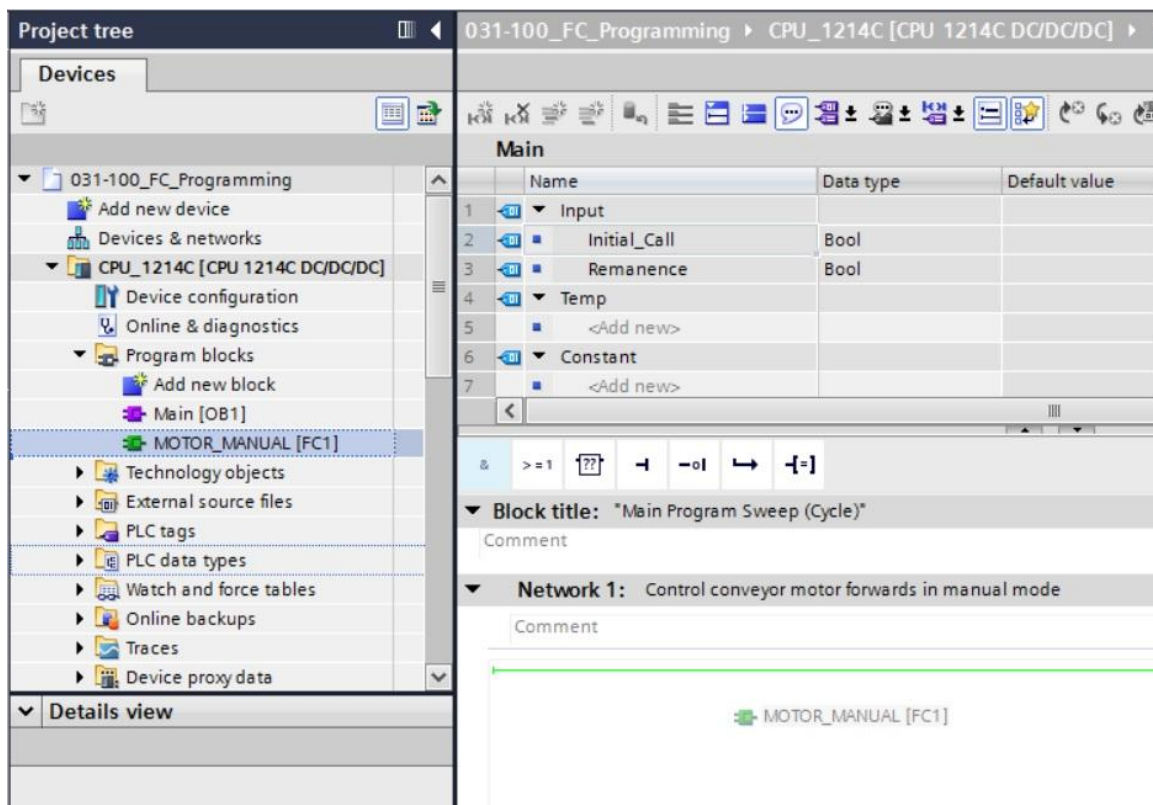


- ® Nomeie a rede 1 como "Control conveyor motor forwards in manual mode" (Acionamento movimento da correia para frente no manual-/comando por impulsos).

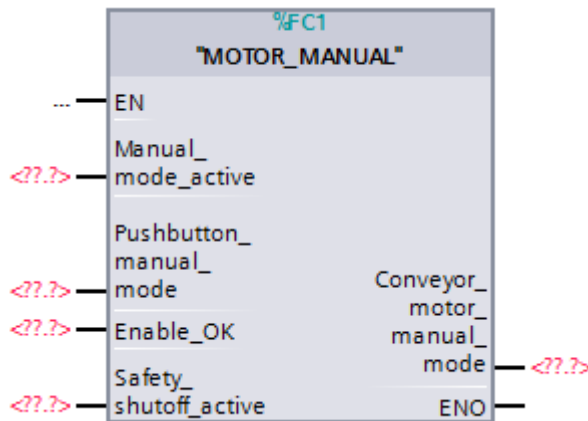
(® Network 1:... (Rede 1:...)
 ® Control conveyor motor forwards in manual mode
 (Acionamento movimento da correia para frente no manual-/comando por impulsos))




- ® Arraste sua função "MOTOR_manual [FC1]" e solte na rede 1, na linha verde.

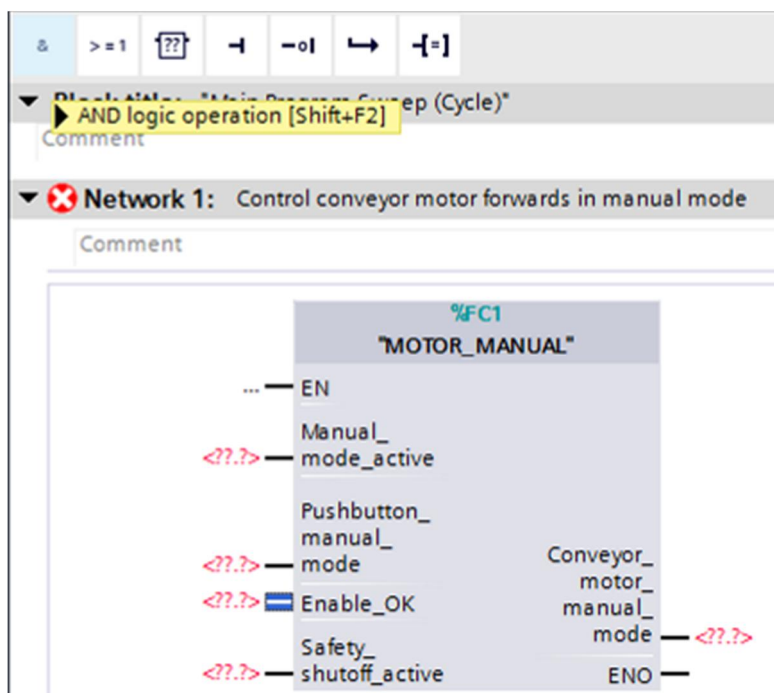




- ® Um bloco será adicionado com interface e conexões EN e ENO definida por você na rede 1.

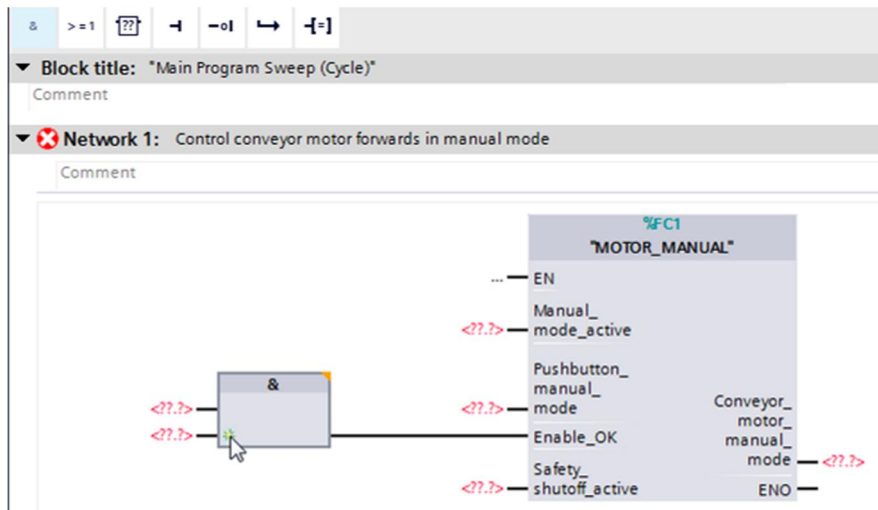


- ® Para adicionar um E antes do parâmetro de entrada "Enable_OK" (Liberação_OK), marque esta entrada e adicione com um clique no símbolo  in na sua barra de ferramentas lógica.

(® )

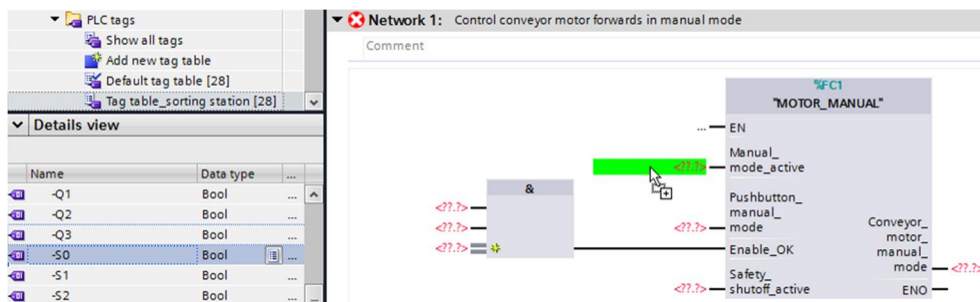


® Clique na estrela amarela  do elemento E para adicionar mais uma entrada. (® )

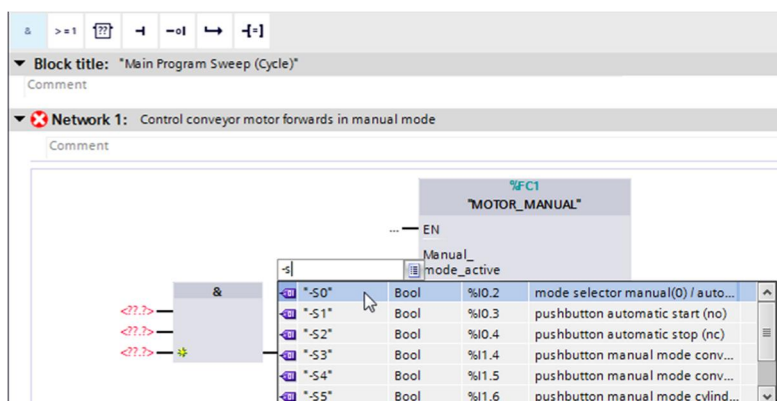


® Para ligar o bloco com o variáveis globais da "Tag table_sorting station" (Tabela de variantes_sistema de classificação), temos 2 possibilidades:

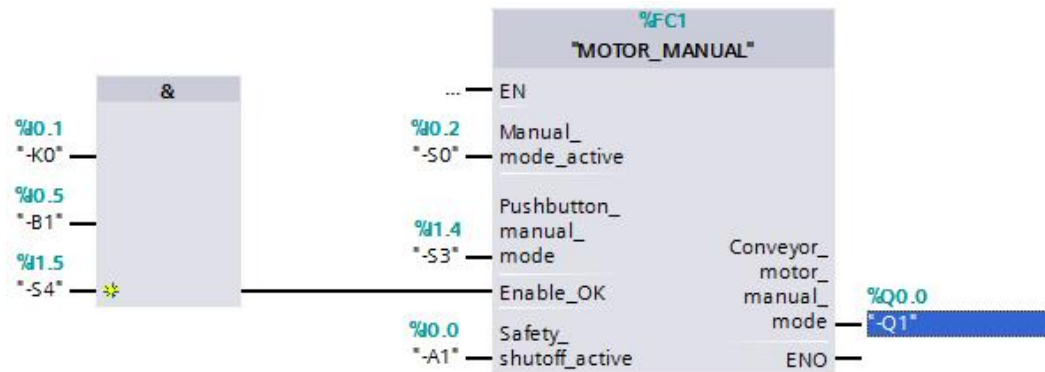
® Ou selecionado a "Tag table_sorting station" (Tabela de variantes_sistema de classificação) na árvore do projeto e arrastando a variável global da visualização detalhada na interface do FC1 (® Tag table_sorting station (Tabela de variantes_sistema de classificação) ® Details view (Visualização detalhada) ® -S0® Manual_mode_active (Operação manual_ativo))



® Ou inserindo em <??.?> as letras iniciais (ex.: "-S") da variável global desejada e escolhendo da lista exibida, a variável de entrada global "-S0" (% E0.2). (® Manual_mode_active (Operação manual_ativo) ® -S ® -S0)

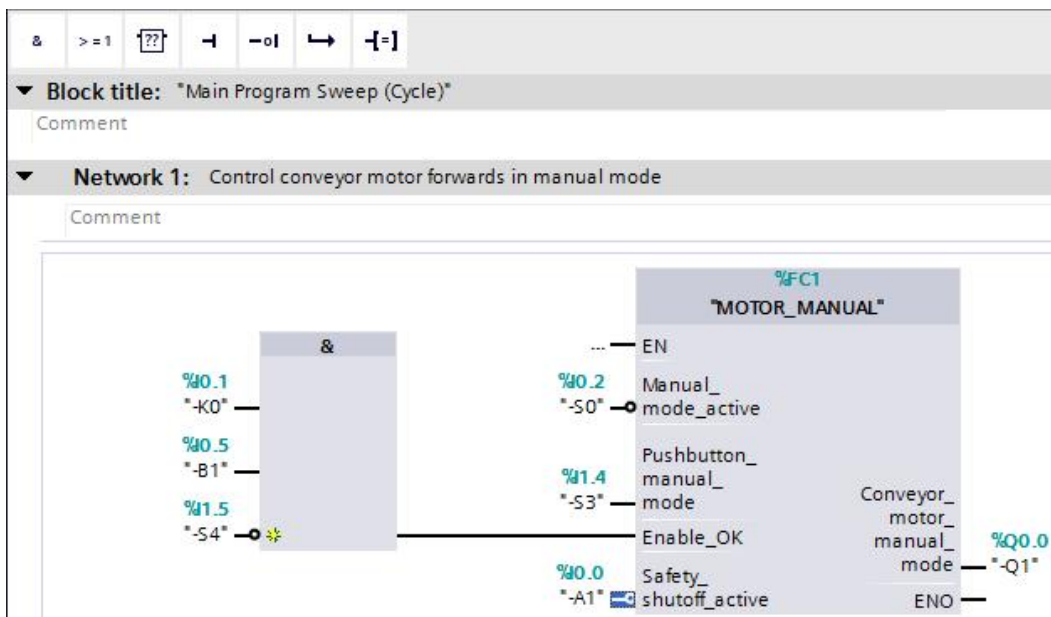


- ® Adicione as outras variáveis de entrada "-S3", "-K0", "-B1", "-S4" e "-A1" e na saída "Conveyor_motor_manual_mode" (Motor de correia_comando por impulsos) a variável de saída "-Q1" (%A0.0).

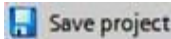

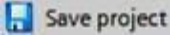



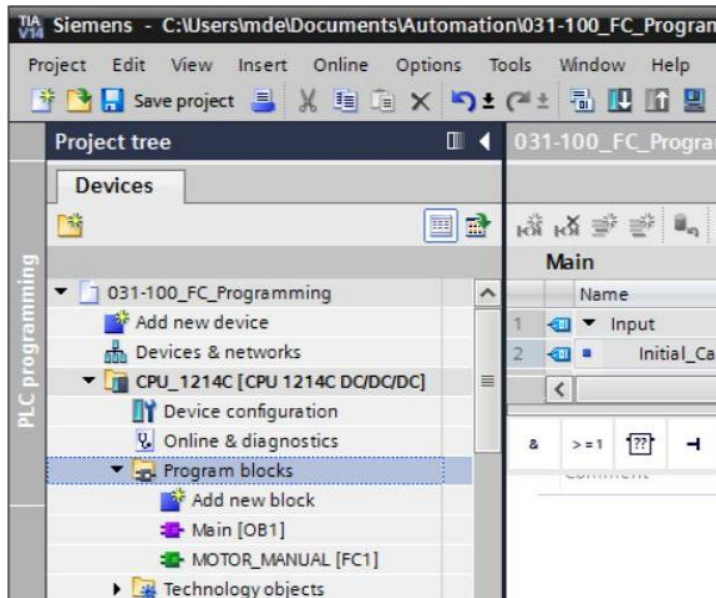
- ® Negue as perguntas das variáveis de entrada "-S0", "-S4" e "-A1" marcando e em seguida

clicando em. (® -S0 ® ® -S4 ® ® -A1 ®)

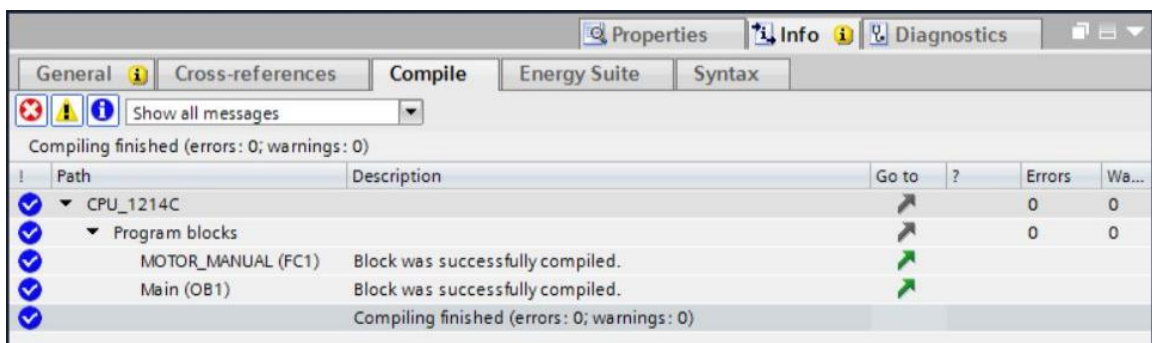


7.9 Salvar e verter programa


- ® Para salvar seu projeto, selecione a função  no menu. Para verter todos os módulos, clique na pasta "Program blocks" (Blocos do programa) e selecione o ícone  no menu. (®  ® Program blocks (Blocos do programa) ® )

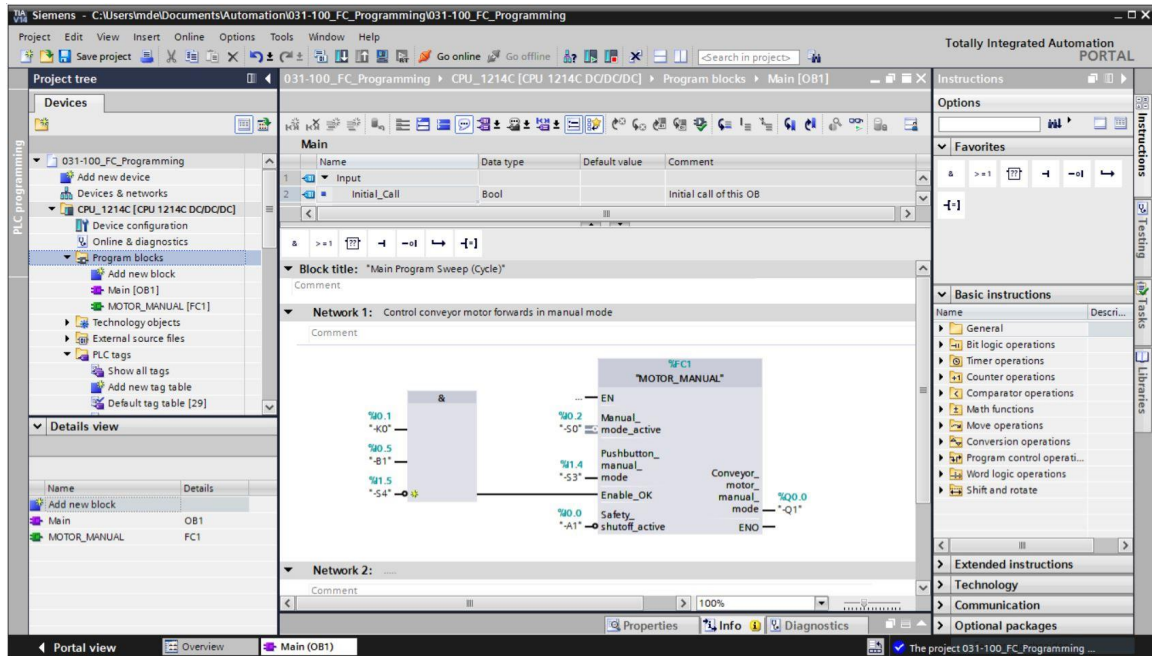


- ® Na área "Info" "Compile" (Verter) será mostrado em seguida, qual bloco pôde ser vertido com êxito.





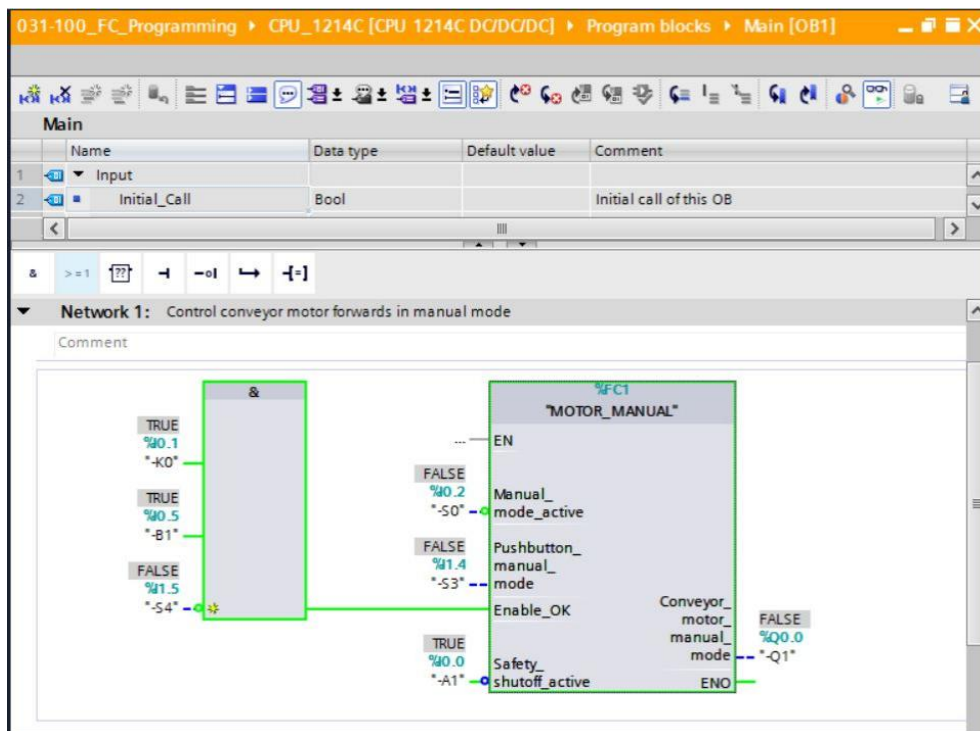
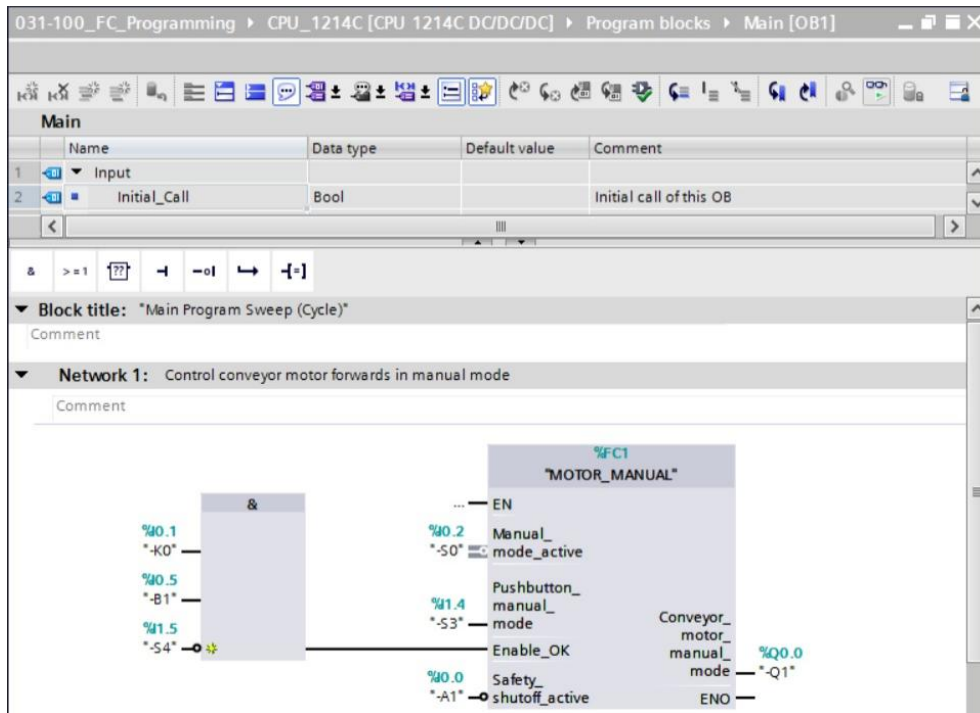
7.10 Carregar programa

- ® Ao verter com êxito, o controller todo pode ser carregado com o programa criado, assim como já descrito nos módulos para configuração do hardware. (® )



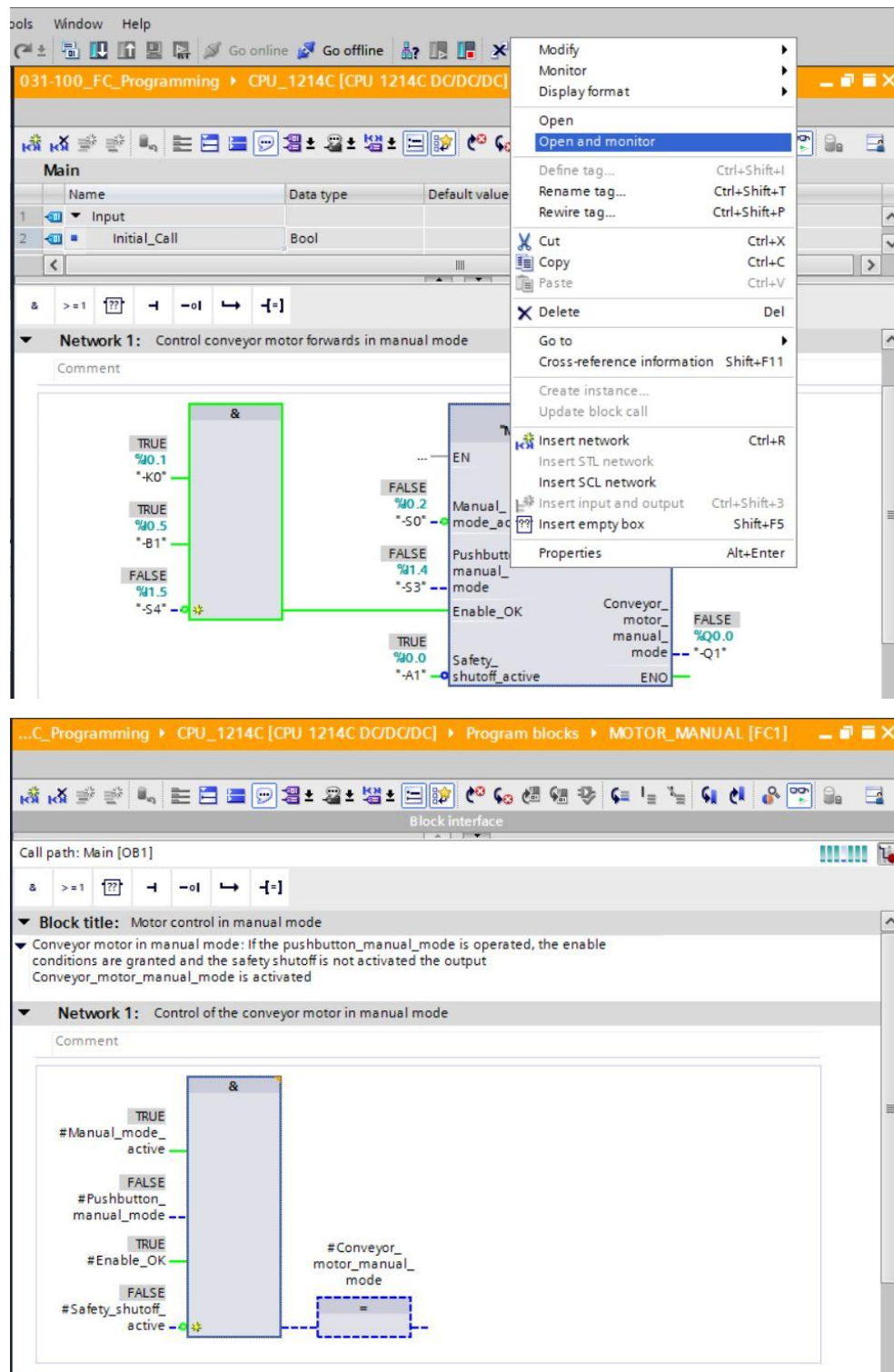
7.11 Observar blocos de programa

- ® Para observar o programa carregado, o bloco desejado deve ser aberto. Depois é possível trocar, clicando no símbolo  a observação em ligada/desligada. (® Main [OB1] ® )





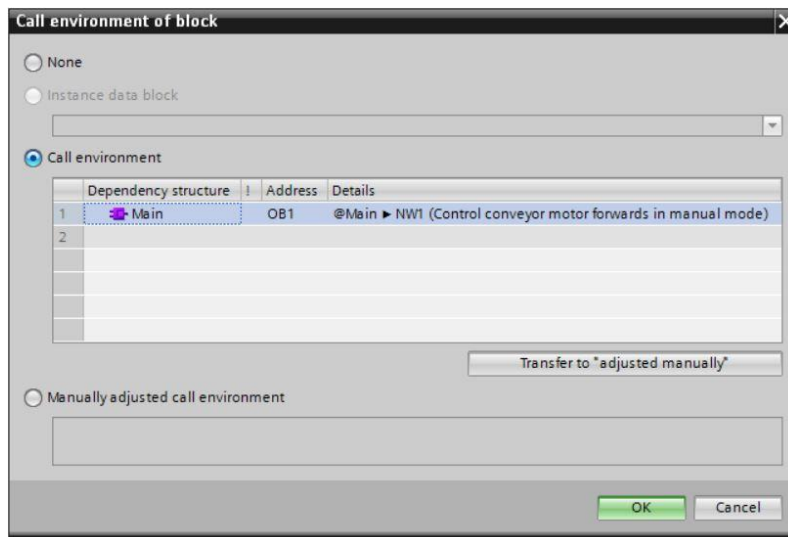
Indicação: A observação ocorre aqui associado a sinais e dependendo do controle. o status do sinal nos terminais é exibido com VERDADEIRO (TRUE) OU FALSO (FALSE).

- ® A função solicitada no bloco de organização "Main [OB1]", "MOTOR_manual" [FC1] pode ser selecionada com um clique no lado do mouse direito em "Open and monitor" (Abrir e observar). (® "MOTOR_manual" [FC1] ® Open and monitor (Abrir e observar))



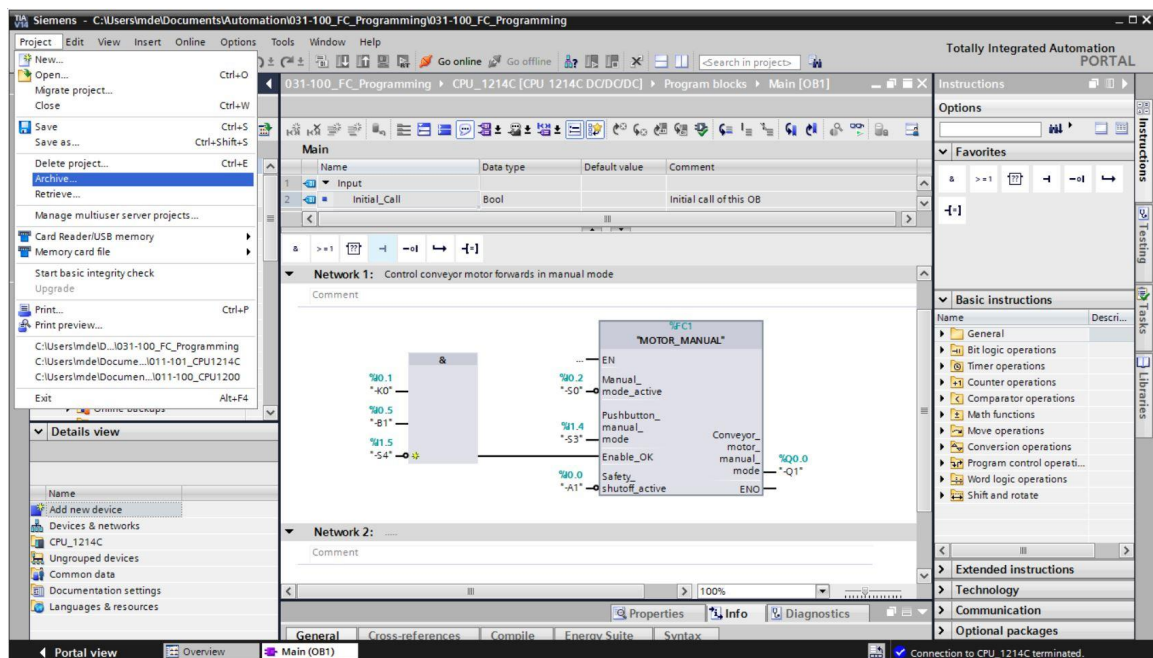
Indicação: A observação ocorre aqui associado à função e dependendo do controle. A confirmação do transmissor ou status do equipamento é exibido com VERDADEIRO (TRUE) OU FALSO (FALSE).

- ® Se um determinado uso da função "MOTOR_MANUAL" [FC1] não for observado, é possível selecionar o ambiente da solicitação através do símbolo . (®  ® Call environment (Ambiente da solicitação) ® OK)



7.12 Arquivar projeto

- ® No fim queremos arquivar o projeto por completo. Por favor, selecione no menu ® "Project" (Projeto), o item ® "Archive ..." (Arquivar ...). Selecione uma pasta, na qual se quer arquivar o projeto e salve como tipo de arquivo "TIA Portal project archives" (TIA Portal-Arquivos de projeto). (® Project (Projeto) ® Archive (Arquivar) ® TIA Portal project archives" (TIA Portal-Arquivos de projeto) ® 031-100_Programação-FC...® Salvar)



7.13 Lista de verificação

N°.	Descrição	Verificado
1	Versão com êxito e sem mensagem de erro	
2	Carregamento com êxito e sem mensagem de erro	
3	Ligar equipamento (-K0 = 1) Cilindro retraído/ retorno ativado (-B1 = 1) DESLIGAMENTO DE EMERGÊNCIA (-A1 = 1) não ativo Tipo de operação manual (-S0 = 0) Ativar comando por impulsos correia para frente (-S3 = 1) Motor de correia velocidade fixa para frente (-Q1 = 1)	
4	Como no 3 mas tendo que ativar DESLIGAMENTO DE EMERGÊNCIA (-A1 = 0) ® -Q1 = 0	
5	Como no 3 mas sendo o tipo de operação AUTO (-S0 = 1) ® -Q1 = 0	
6	Como no 3 mas tendo que desligar equipamento (-K0 = 0) ® -Q1 = 0	
7	Como no 3 mas sendo que Cilindro não está retraído(-B1 = 0) ® -Q1 = 0	
8	Como no 8 mas tendo que ativar também o comando por impulsos, correia com movimento para trás (-S4 = 1) ® -Q1 = 0	
9	projeto arquivado com êxito	

8 Exercício

8.1 Tarefa – exercício

Neste exercício, as seguintes funções da descrição do processo de classificação devem ser planejadas, programadas e testadas:

- Operação manual – acionamento do movimento da correia para trás manualmente/comando por impulsos

8.2 esquema de tecnologia

Aqui você pode ver o esquema de tecnologia para a tarefa.

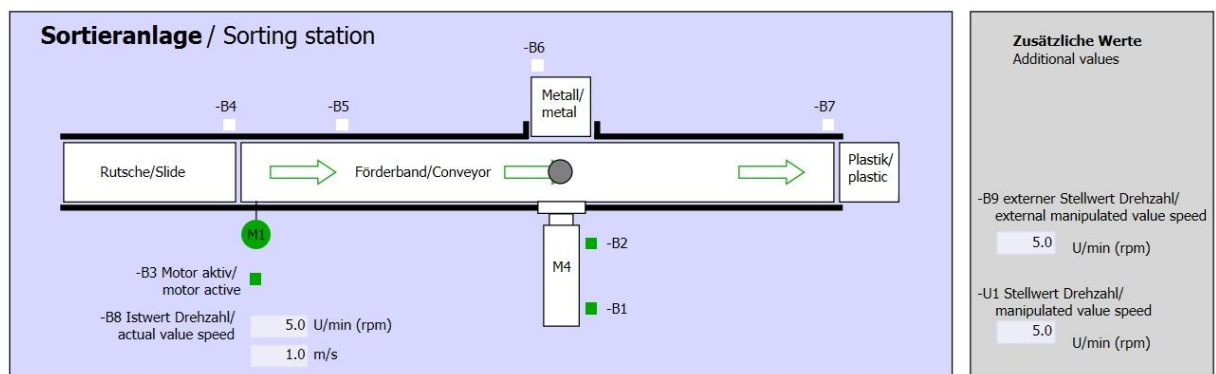


Imagem 10: esquema de tecnologia

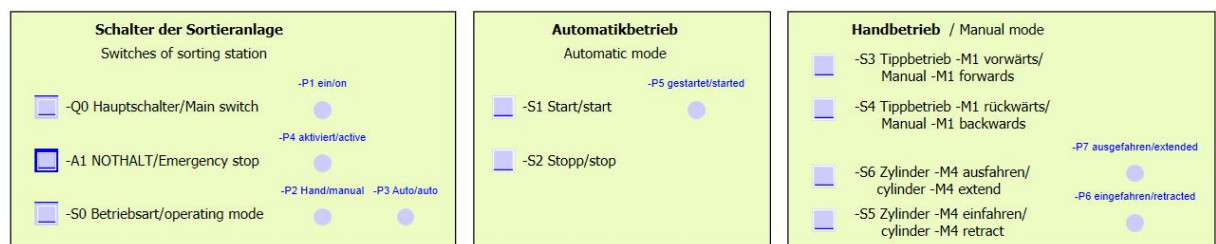


Imagem 11: painel de controle

8.3 Tabela de atribuição

Os sinais a seguir serão utilizados como operando nesta tarefa.

DE	Tipo	Identificação	Função	NC/NO
E 0.0	BOOL	-A1	Mensagem PARADA DE EMERGÊNCIA ok	NC
E 0.1	BOOL	-K0	Equipamento "Ligado"	NO
E 0.2	BOOL	-S0	Chave seletora operação manual (0)/ automático(1)	manual = 0 auto=1
E 0.5	BOOL	-B1	Cilindro de sensor -M4 retraído	NO
E 1.4	BOOL	-S3	Tecla comando por impulsos correia -M1, movimento para frente	NO
E 1.5	BOOL	-S4	Tecla comando por impulsos correia -M1, movimento para trás	NO

DA	Tipo	Identificação	Função	
A 0.1	BOOL	-Q2	motor de correia -M1 velocidade fixa para trás	

Legenda da lista de atribuição

DE	entrada digital	DA	saída digital
AE	entrada analógica	AA	saída analógica
E	entrada	A	saída
NC	normally closed (contato normalmente fechado)		
NO	normally open (contato normalmente aberto)		

8.4 Planejamento

Faça seu planejamento de implementação de tarefa.

8.5 Lista de verificação – exercício

N°.	Descrição	Verificado
1	Versão com êxito e sem mensagem de erro	
2	Carregamento com êxito e sem mensagem de erro	
3	Ligar equipamento (-K0 = 1) Cilindro retraído/ retorno ativado (-B1 = 1) DESLIGAMENTO DE EMERGÊNCIA (-A1 = 1) não ativo Tipo de operação manual (-S0 = 0) Ativar comando por impulsos correia para trás (-S4 = 1) Motor de correia velocidade fixa de movimento para trás (-Q2 = 1)	
4	Como no 8 mas tendo que ativar DESLIGAMENTO DE EMERGÊNCIA (-A1 = 0) ® -Q2 = 0	
5	Como no 8 mas sendo o tipo de operação AUTO (-S0 = 1) ® -Q2 = 0	
6	Como no 8 mas tendo que desligar equipamento (-K0 = 0) ® -Q2 = 0	
7	Como no 8 mas sendo que Cilindro não está retraído(-B1 = 0) ® -Q2 = 0	
8	Como no 8 mas tendo que ativar também o comando por impulsos, correia com movimento para frente (-S3 = 1) ® -Q1 = 0 e também -Q2 = 0	
9	projeto arquivado com êxito	

9 Informação adicional

Para familiarização, respectivamente, aprofundamento maior, você encontra informações adicionais no Guia de Orientação, como por ex.: Getting Started, vídeos, tutoriais, aplicativos, manuais, guias de programação e testes de software/firmware, no link a seguir:

www.siemens.com/sce/s7-1200

Pré-visualização “Informações adicionais”

Getting Started, Videos, Tutorials, Apps, Manuals, Trial-SW/Firmware

- TIA Portal Videos
- TIA Portal Tutorial Center
- Getting Started
- Programming Guideline
- Easy Entry in SIMATIC S7-1200
- Download Trial Software/Firmware
- Technical Documentation SIMATIC Controller
- Industry Online Support App
- TIA Portal, SIMATIC S7-1200/1500 Overview
- TIA Portal Website
- SIMATIC S7-1200 Website
- SIMATIC S7-1500 Website

Mais informações

Siemens Automation Cooperates with Education
[siemens.com/sce](https://www.siemens.com/sce)

Documentos de treinamento SCE
[siemens.com/sce/documents](https://www.siemens.com/sce/documents)

Pacotes para o instrutor SCE
[siemens.com/sce/tp](https://www.siemens.com/sce/tp)

Contatos SCE
[siemens.com/sce/contact](https://www.siemens.com/sce/contact)

Empreendimento digital
[siemens.com/digital-enterprise](https://www.siemens.com/digital-enterprise)

Indústria 4.0
[siemens.com/future-of-manufacturing](https://www.siemens.com/future-of-manufacturing)

Totally Integrated Automation (TIA)
[siemens.com/tia](https://www.siemens.com/tia)

TIA Portal
[siemens.com/tia-portal](https://www.siemens.com/tia-portal)

Controlador SIMATIC
[siemens.com/controller](https://www.siemens.com/controller)

Documentação técnica SIMATIC
[siemens.com/simatic-docu](https://www.siemens.com/simatic-docu)

Suporte industrial online
support.industry.siemens.com

Catálogo de produtos e sistema de solicitação online Industry Mall
mall.industry.siemens.com

Siemens AG
Digital Factory
P.O. Box 4848
90026 Nuremberg
Germany

Sujeito a alterações e erros
© Siemens AG 2018

[siemens.com/sce](https://www.siemens.com/sce)