



UNIVERSIDADE DE ÉVORA
ESCOLA DE CIÊNCIAS E TECNOLOGIA

Mestrado em Engenharia Mecatrónica

Dissertação

Automação na Maximização de Produção de Energia Elétrica de um Painel Solar

Eliseu António Calado Fernandes

Orientador:

Professor Doutor João Manuel Gouveia Figueiredo

Setembro de 2012

Mestrado em Engenharia Mecatrónica

Dissertação

Automação na Maximização da Produção de Energia Elétrica de um Painel Solar

Eliseu António Calado Fernandes

Orientador:

Professor Doutor João Manuel Gouveia Figueiredo

Resumo

Apresenta-se a implementação de uma plataforma para um sistema de seguimento solar de dois eixos, que utiliza como método de seguimento a avaliação da potência elétrica produzida por uma célula fotovoltaica.

O protótipo é motorizado por dois motores de passo, controlados automaticamente por um controlador lógico programável (Siemens S7-300) com módulos funcionais FM353 e FM350-1.

O dispositivo é constituído por dois subsistemas. O subsistema principal é responsável pela execução dos movimentos pré-programados de pesquisa e colocação do painel na posição que maximiza a potência elétrica. O subsistema secundário tem a função de selecionar o varrimento pré-programado mais indicado, de acordo com as medições obtidas pelos sensores de irradiância programáveis, próximos da célula.

O protótipo laboratorial foi desenvolvido em colaboração com o Centro de Engenharia Mecatrónica da Universidade de Évora.

Palavras-chave: Seguimento solar, Painel fotovoltaico, Controlador lógico programável; Maximização de potência elétrica

Automation to Maximise Electricity Generation Through Solar Panel

Abstract

This dissertation presents the implementation of a platform for a two axis solar tracking system, which utilizes a small photovoltaic cell as a sensing element.

Two stepper motors controlled automatically by a programmable logic controller (S7-300) motorize the system. Specific function modules control the stepper motors (FM353 and FM 350-1).

The prototype consists of two subsystems. The main subsystem preforms the programmed movements to positioning the photovoltaic cell in order maximize the electric power output, according to the measurements of programmable light sensors placed close to the cell.

The laboratorial prototype was developed in the University of Évora, Portugal in cooperation with the Mechatronics Engineering Centre of this institution.

Keywords: Solar tracking systems; Photovoltaic panel; Programmable logic controller; Electric power maximization

Agradecimentos

Em primeiro lugar aos meus pais, porque lhes devo tudo e porque são uma enorme fonte de motivação. Na mesma medida, agradeço à minha irmã de quem tenho um imenso orgulho. Este agradecimento estende-se a todos os meus familiares.

Agradeço ao Professor João Figueiredo, meu orientador, pela disponibilidade e apoio prestado, muito importante sobretudo durante o período em que estive ausente do país.

Ao meu colega David Bruno, pela companhia no laboratório, que foi fundamental durante as estadias para “férias” em Portugal, que me permitiram concluir a componente laboratorial.

O apoio inicial prestado pelo colega Bruno Robalo.

A todos os amigos, que sempre compreenderam quando não podia estar com eles.

Índice geral

Capítulo 1 – Introdução.....	1
1.1 Panorama energético internacional.....	1
1.1.1 Recurso solar em Portugal	3
1.2 Motivação	4
1.3 Objetivos.....	4
1.4 Estrutura da dissertação	5
Capítulo 2 - Tecnologia solar	6
2.1 Central solar fotovoltaica.....	6
2.1.1 Sistemas de seguimento solar em dois eixos.....	6
2.2 Sistema solar térmico.....	9
2.3 Central solar térmica de concentração.....	9
Capítulo 3 – Descrição do sistema.....	12
3.1 Descrição dos componentes e do funcionamento	12
3.1.1 Aspetos gerais do funcionamento do sistema.....	14
3.2 Descrição técnica do equipamento – Subsistema principal	17
3.2.1 Descrição do controlador.....	17
3.2.2 Módulos constituintes do PLC.....	18
3.2.3.1 Fonte de alimentação	19
3.2.3.2 CPU Siemens 315-2 DP.....	20
3.2.3.3 Módulo de comunicação CP-343 Advanced-IT.....	23
3.2.3.4 Módulo de entradas e saídas digitais	23
3.2.3.5 Módulo de entradas e saídas analógicas	24
3.2.3.6 Módulo funcional de contagem FM350-1	26
3.2.3.7 Motores passo-a-passo.....	31
3.2.3.8 Drivers – Cartas de controlo dos motores PaP.....	35
Fig. 39. Configuração do número de passos (esquerda) e configuração da corrente [17]	35
3.2.3.8 Módulo Funcional FM 353 para comando de motores PaP.....	36
3.3 Descrição técnica do equipamento – Subsistema secundário	40
3.3.1 Elemento sensor.....	40
3.3.2 Microcontrolador – Arduino Mega 2560	44
Capítulo 4 – Descrição do software do subsistema secundário.....	48

4.1 Instalação e utilização do software de programação	48
4.2 Classificação face ao conjunto de instruções.....	48
4.2.1 Funções base	49
Capítulo 5 – Descrição do software do subsistema principal.....	51
5.1 Criação do projeto em STEP-7.....	51
5.2 Configuração de hardware	54
5.2.1 Parametrização dos módulos FM 350-1	57
5.2.2 Parametrização dos módulos FM 353	58
5.3 Programa executável.....	59
5.3.1 Endereçamento de memória em PLCs S7	61
5.3.2 Estrutura do programa de utilizador.....	63
Capítulo 6 – Descrição da aplicação HMI	71
Capítulo 7 – Descrição do Sistema de Supervisão e Controlo	75
7.1 Configuração da comunicação (SCADA – PLC).....	77
7.2 Descrição da aplicação SCADA	79
Capítulo 8 - Resultados	84
Capítulo 9 – Conclusões.....	92
Bibliografia.....	94

Índice de figuras

Fig. 1. Consumo energético mundial (10^{15} Btu) (adaptado de [5])	2
Fig. 2. Produção mundial de eletricidade (10^{12} kWh) (adaptado de [5]).....	2
Fig. 3. Distribuição geográfica mundial da produção elétrica a partir de fontes renováveis (adaptado de [6]).....	2
Fig. 4. Irradiação global anual na Europa em (kWh/m^2), numa superfície orientada para sul e com inclinação ótima [7]	3
Fig. 5. Ilustração da variação azimutal e zenital (ângulos solares)	7
Fig. 6. Diferença de captação de radiação entre uma instalação com sistema de dois eixos e uma instalação sem sistema de seguimento [26].....	8
Fig. 7. Detalhe de um tubo recetor, esquema de um coletor parabólico (adaptado de [10])	10
Fig. 8. Esquema simplificado do princípio de funcionamento de um LFR (adaptado de [10]).....	10
Fig. 9. Esquema simplifica do princípio de funcionamento de um coletor de disco parabólico (adaptado de [10]).....	10
Fig. 10. Esquema do princípio de funcionamento de uma Torre de Concentração de Energia Solar (adaptado de [10]).....	11
Fig. 11. Fotografia do protótipo do sistema principal.....	12
Fig. 12. PLC e painel HMI constituintes do sistema principal.....	13
Fig. 13. Principais componentes do sistema secundário.....	14
Fig. 14. Diagrama de blocos geral do sistema.....	15
Fig. 15. Curva I-V da célula FV	16
Fig. 16. Ciclo de varrimento genérico de um PLC	18
Fig. 17. Fotografia da fonte de alimentação do PLC	19
Fig. 18. Adaptador RS232/MPI e dispositivos conectados em rede MPI [11]	19
Fig. 19. Esquema da sinalética e das portas de interface da CPU do PLC S7-300 [12].....	20
Fig. 20. Transferência de informação entre software PC, cartão de memória e CPU do PLC [12]....	22
Fig. 21. Módulo de comunicação (CP) do PLC S7-300 utilizado [Siemens]	23
Fig. 22. Módulo de entradas e saídas digitais do PLC S7-300 utilizado [Siemens].....	23
Fig. 23. Ligações e diagrama de bloco – SM 323 DI8/DO8 x DC 24 V/0.5A [13]	23
Fig. 24. Módulo de entradas analógicas utilizado no PLC S7-300 [Siemens].....	24
Fig. 25. Marcação dos módulos para diferentes faixas de medição [12].....	24

Fig. 26. Esquema de ligações: transdutores de tensão a entradas analógicas eletricamente isoladas (esquerda); transdutores de corrente de dois fios a entradas analógicas eletricamente isoladas (direita) [13].....	25
Fig. 27. Diagrama de blocos de um módulo de entradas AI (SM 331 AI8 x12 bit) [13]	25
Fig. 28. Esquema com sinalética identificativa do módulo FM 350-1 (direita) [14]	26
Fig. 29. Modo de funcionamento do módulo AI, de acordo com o posicionamento do conector (A/D) [14].....	27
Fig. 30. Esquema de ligações / PIN-OUT dos sensores de ângulo utilizados [14]	28
Fig. 31. Colocação dos módulos contadores numa rack com CPU da gama S7-300. Notar na conexão blindada dos fios de ligação dos encoders aos módulos FM 350-1 [14]	28
Fig. 32. Terminais do conector frontal de 20 pinos do módulo FM 350-1 (impulsos dos encoders; entradas e saídas digitais; tensão de alimentação dos encoders; tensão de alimentação auxiliar); PIN-OUT dos encoders [14].....	29
Fig. 33. Abertura e fecho de porta para contagem de pulsos [14].....	29
Fig. 34. Distinção entre os diferentes métodos de contagem do módulo FM 350-1 [14].....	30
Fig. 35. Diagrama temporal de sinais de um encoder incremental de 5V	30
Fig. 36. Diagrama temporal exemplificativo de avaliação singular [14].....	31
Fig. 37. Fotografia motor do eixo 1 (esquerda) e do motor do eixo 2 (direita) [17].....	32
Fig. 38. Elementos constituintes do motor HB PaP. Vista em corte longitudinal (esquerda) e transversal (direita) [16].....	33
Fig. 39. Configuração do número de passos (esquerda) e configuração da corrente [17].....	35
Fig. 40. Fotografia (esquerda) e sinalética exterior do módulo FM 353 (direita) [18].....	37
Fig. 41. Sinalética: interfaces e conector frontal [17]. Atribuição dos terminais do conector frontal de 20 pinos do módulo FM 353. I: Entrada; Q: Saída; VI: Entrada de tensão; DI1 – 4: Entrada digital 1 – 4; DQ1 – 4: Saída digital 1 – 4; RM_P: Entrada positiva de sinal de controlo; RM_N: Entrada negativa de sinal de controlo; L+: alimentação de 24 V; M: terra.....	37
Fig. 42. Ligações entre os módulos FM 353, os drives SMC32 e os motores passo-a-passo (eixo 1)	38
Fig. 43. Ligações entre os módulos FM 353, os drives SMC32 e os motores passo-a-passo (eixo 2)	39
Fig. 44. Imagem de um circuito integrado TSL230R-LF [19].....	40
Fig. 45. Diagrama de blocos do circuito integrado TSL230R-LF [19]	41
Fig. 46. Representação das portas do circuito integrado (cima) e do microcontrolador (esquerda), adaptado de [19], [20].....	41
Fig. 47. Representação gráfica da irradiação incidente em ordem à saída em frequência [19].....	43
Fig. 48. Diagrama de um microcontrolador genérico e exemplos de funcionalidades [22]	44

Fig. 49. Microcontrolador Arduino Mega 2560, com destaque para os diferentes tipos de portas de interface [21]	45
Fig. 50. Forma de sinal PWM	46
Fig. 51. Representação de um impulso PWM.....	47
Fig. 52. Escolha da porta série e do modelo de microcontrolador, no software de desenvolvimento Arduino	48
Fig. 53. Estrutura base de um programa para execução num microcontrolador Arduino [22].....	49
Fig. 54. Relação entre as funções do programa executável (sketch) do microcontrolador.....	50
Fig. 55. Papel do gestor de projetos STEP 7 na administração de projetos.....	53
Fig. 56. Árvore hierárquica do Projeto, com detalhe para o conteúdo da pasta projeto e da estação SIMATIC 300.....	54
Fig. 57. Configuração de rede PROFIBUS com endereço 2 e velocidade de transmissão 1.5 Mbps. 55	
Fig. 58. Introdução dos módulos na tabela que representa a rack do PLC	55
Fig. 59. Criação da rede Ethernet (TCP/IP)	56
Fig. 60. Tabela da aplicação <i>HW Config</i> com colunas: de endereço MPI, endereços de entrada e endereços de saída	56
Fig. 61. Aplicação “NetPro” do STEP 7, com destaque para rede Ethernet.....	57
Fig. 62. Diagrama com a estrutura da configuração de um FM 353, com recurso ao pacote de software “Parameterize FM353”, adaptado de [18]	59
Fig. 63. Panorâmica dos blocos de organização em programação STEP 7 [Siemens].....	61
Fig. 64. Diagrama para diferenciação entre endereçamento direto e indireto em programação STEP 7 [26].....	62
Fig. 65. Criação do G-code	65
Fig. 66. Detalhe da seleção do DB que corresponde ao varrimento a executar	65
Fig. 67. Processo de atribuição do DB com o G-code do varrimento selecionado	66
Fig. 68. Sentido dos eixos do sistema principal e representação dos varrimentos pré-programados..	66
Fig. 69. Sequência de movimentos de um varrimento pré-programado	66
Fig. 70. Organigrama do funcionamento do subsistema principal.....	67
Fig. 71. Diagrama ilustrativo da comunicação entre o programa de utilizador e os módulos FM 350-1 [14].....	68
Fig. 72. Diagrama ilustrativo da interligação entre o módulo FM 353 e o programa de utilizador [18]	68
Fig. 73. Organigrama de blocos constituintes do programa de utilizador do sistema principal	70
Fig. 74. Tela modelo “TEMPLATE” (1)	71

Fig. 75. Tela inicial “HOME” (2).....	71
Fig. 76. Tela de varrimentos “RESEARCH MENU” (3)	71
Fig. 77. Tela de seleção do tempo de pausa entre varrimentos “RESEARCH PERIOD SELECTION” (4).....	71
Fig. 78. Painel de comando manual “JOG MODE” (5).....	71
Fig. 79. Painel HMI OP 270 (Siemens)	73
Fig. 80. Aplicação “NetPro” do STEP 7, com destaque para comunicação entre PLC e painel HMI	73
Fig. 81. Tipo de conexão estabelecida entre o painel HMI e o controlador S7-300 [28].....	74
Fig. 82. Tabela com as configurações possíveis de estabelecer entre o OP e um PLC, associadas à configuração do switch da interface IF1B [27].....	74
Fig. 83. Pirâmide da hierarquia entre os diferentes níveis de automação numa estrutura industrial [Siemens].....	76
Fig. 84. Sub-redes em sistemas SIMATIC [Siemens]	76
Fig. 85. Definição da interface PG/PC: STEP 7 TCP/IP	78
Fig. 86. Atribuição do IP da rede Ethernet TCP/IP	78
Fig. 87. Lista completa das “tags” criadas para interação com o controlador.....	79
Fig. 88. Diagrama com as diferentes telas/ecrãs da aplicação SCADA	80
Fig. 89. Desenvolvimento da tela principal, na aplicação “Graphics Designer”. Tela de sistema secundário “PICTURE WINDOW” para telas do sistema principal	81
Fig. 90. Tela com instruções de operação (“RESEARCH”) e “PICTURE WINDOW” para tela do sistema secundário	81
Fig. 91. Tela do subsistema secundário e tela com instruções de operação.....	82
Fig. 92. Tela para posicionamento manual dos eixos do sistema principal	82
Fig. 93. Portal Web para monitorização remota do sistema.....	83
Fig. 94. Resultado de posicionamento ótimo obtido – ensaio 1	84
Fig. 95. Resultado de posicionamento ótimo obtido – ensaio 2.....	85
Fig. 96. Representação gráfica no sistema SCADA do ensaio número três	86
Fig. 97. Interface amigável SCADA.....	86
Fig. 98. Comparação do posicionamento do eixo 1: varrimento e posição final	88
Fig. 99. Diferenças entre o valor de posição registado no varrimento e o valor de posição final, com relação ao eixo 1	88
Fig. 100. Comparação do posicionamento do eixo 2: varrimento e posição final	89
Fig. 101. Diferenças entre o valor de posição registado no varrimento e o valor de posição final, com relação ao eixo 2.....	89

Fig. 102. Comparação de valores de potência elétrica: varrimento e posição final 90

Fig. 103. Gráfico das diferenças entre o valor de potência eléctrica no varrimento e o valor de potência elétrica final..... 90

Índice de tabelas

Tabela 1. Classificação de coletores solares [10].....	9
Tabela 2. Identificação dos componentes principais do protótipo	13
Tabela 3. Características da célula FV	16
Tabela 4. Configuração de hardware do PLC Siemens ® S7-300	18
Tabela 5. Componentes necessários para implementar o varrimento solar	18
Tabela 6. Principais características da CPU do PLC S7-300 [12].....	20
Tabela 7. Designação das portas de interface da CPU.....	21
Tabela 8. Descrição das áreas de endereçamento da memória de sistema (RAM) [12].....	22
Tabela 9. Métodos e faixas de medição do módulo de entradas analógicas [13].....	24
Tabela 10. Resoluções em módulos AI em PLCs da gama S7-300 [12].....	25
Tabela 11. Representação dos valores digitalizados referentes aos valores analógicos medidos [13]	26
Tabela 12. Sinalização do painel frontal do FM 350-1 [14]	27
Tabela 13. Tipos de encoders e sinais suportados pelo contador [14].....	27
Tabela 14. Características técnicas principais dos motores passo-a-passo [16].....	32
Tabela 15. Descrição das principais características dos motores BH PaP [16], [17]	32
Tabela 16. Características técnicas das cartas de comando dos motores PaP [17]	35
Tabela 17. Ligações do módulo FM 353 com o drive SMC32 (motor 1) [18]	38
Tabela 18. Ligações do módulo FM 353 com o drive SMC32 (motor 2) [18]	39
Tabela 19. Parâmetros elétricos dos sinais de saída do módulo FM 353 para o driver [18].....	40
Tabela 20. Características do sensor (TSL230R-LF) [19].....	40
Tabela 21. Apresentação das portas do circuito integrado e correspondência com as ligações ao microcontrolador [19], [20].....	41
Tabela 22. Algumas considerações importantes relativas à montagem do sistema secundário (TSL230R-LF) [19], [20]	42
Tabela 23. Numeração e funções dos pinos do circuito integrado [19].....	42
Tabela 24. Correspondência entre as saídas analógicas do microcontrolador e as entradas analógicas do PLC	42
Tabela 25. Seleção de opções disponíveis (TSL230R-LF) [19].....	42
Tabela 26. Condições de operação recomendadas (TSL230R-LF) [19].....	42
Tabela 27. Características de operação quando se verifica: $V_{DD} = 5\text{ V}$; $T_A = 25\text{ }^{\circ}\text{C}$; $E_e = 130\text{ }\mu\text{W}/\text{cm}^2$; $\lambda_p = 640\text{ nm}$ (TSL230R-LF) [19]	43
Tabela 28. Características do microcontrolador Arduino Mega [22].....	45

Tabela 29. Alimentação elétrica, interface de entradas/saídas e comunicação do microcontrolador [22].....	46
Tabela 30. Condições programadas para indicação ao subsistema primário do varrimento a efetuar	49
Tabela 31. Variáveis mais utilizadas na programação em STEP 7	51
Tabela 32. Apresentação de tipos de dados simples.....	52
Tabela 33. Definição das variantes de dados simples.....	52
Tabela 34. Indicação e descrição dos blocos do utilizador	72

Lista de siglas

AREF	Analog Reference
HB	Hybrid
CAN	Controller Area Network
CP	Communication Module
DB	Data Block
DB-MD	Machine Data
DB-NC	Traversing Programs
DB-SM	Increments
DB-TO	Tool Offset
DP	Decentralized Peripherals
EEPROM	Electrically-Erasable Programmable Read-Only Memory
FM	Function Module
FMS	Field Message Specification
FTDI	Future Technology Devices International (company)
FV	Fotovoltaico
GND	Ground
HMI	Human Machine Interface
MD	Machine Data
MPI	Multi-Point Interface
MRES	Memory Reset
OP	Operator Panel
PA	Process Automation
PaP	Passo-a-Passo
PLC	Programmable Logic Controller
PROFIBUS	Process Field Bus
PWM	Pulse-Width Modulation
SCADA	Supervisory Control and Data Acquisition
SM	Signal Module
SRAM	Static Random Access Memory
STL	Statement List
TCP/IP	Transmission Control Protocol/Internet Protocol
TIA	Totally Integrated Automation
TP	Touch panel
TTL	Transistor-transistor logic
UART	Universal Asynchronous Receiver/Transmitter
UDT	User Defined Datatype

Capítulo 1 – Introdução

1.1 Panorama energético internacional

Por inúmeras razões, sejam elas ambientais, económicas, geopolíticas ou técnicas, têm-se verificado esforços para reduzir a dependência de utilização de combustíveis fósseis. Em particular, as crises energéticas periódicas, desde o choque petrolífero da década de 70, e a crescente sensibilização acerca dos efeitos nocivos inerentes ao seu consumo, são fatores promotores do desenvolvimento da produção de energia a partir de fontes renováveis [1], [2].

No contexto dos avanços gerados pela Revolução Industrial, em Inglaterra no século XVIII, o carvão tornou-se o primeiro e mais importante combustível fóssil. Mais tarde juntou-se a este o petróleo e depois o gás natural. Constituem fontes de energia altamente concentradas, abundantes e baratas, que representam, na atualidade, aproximadamente três quartos da energia consumida globalmente. Contudo, face ao elevado ritmo de extração, inerente ao desenvolvimento das sociedades, passou a haver mais consciência da natureza finita dos combustíveis fósseis [3].

Desde o final da Segunda Guerra Mundial, por razões económicas, de segurança e dos riscos associados aos resíduos tóxicos, ainda não se desenvolveram formas de extrair toda a potencialidade da Energia Nuclear, razão porque muitas regiões escolhem não produzir eletricidade por esta via.

Mesmo nos países produtores e ou exploradores de petróleo, o investimento no desenvolvimento, fabrico e instalação de sistemas de produção de energia elétrica baseados em fontes de energia renováveis tem aumentado, fazendo parte de um conjunto de estratégias para assegurar a sustentabilidade desses países a médio e longo prazo, através do uso de fontes de energia sustentáveis, i.e., que não se comprometem pelo uso continuado e não emitem gases poluentes [4].

Segundo previsões do *International Energy Outlook* de 2011 [5], o consumo de energia, a nível mundial, compreendido entre 2008 e 2035 deverá aumentar cerca de 53%. O maior crescimento (2.8% ao ano) surgirá nas energias renováveis, atingindo aproximadamente os 15% no ano de 2035. Contudo, os combustíveis fósseis continuarão a ser a principal fonte energética mundial, com uma percentagem de contribuição de 80%.

A descoberta de reservas de gás natural não convencional suportará um crescimento do seu uso de cerca de 1.6% ao ano. Os combustíveis líquidos continuarão a representar a principal fonte energética mas a percentagem associada ao petróleo descerá para os 28%. O preço permanecerá elevado, tornando-se um incentivo à substituição de combustível, onde for possível, e ao uso, embora modesto, de biocombustíveis.

Apesar do crescimento das energias renováveis e do gás natural, o carvão continuará a ser a maior fonte utilizada para produção de energia elétrica, em 2035. O relatório (2008-2035) indica um aumento de 43% das emissões de dióxido de carbono.

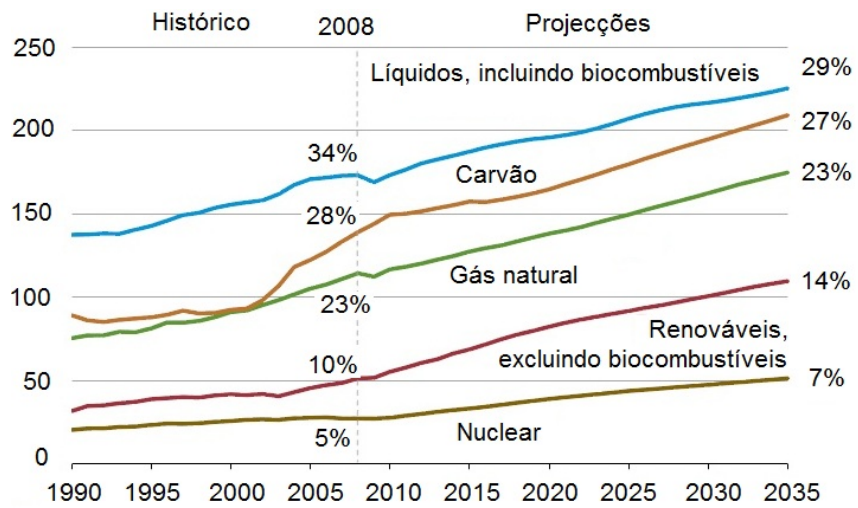


Fig. 1. Consumo energético mundial (10^{15} Btu) (adaptado de [5])

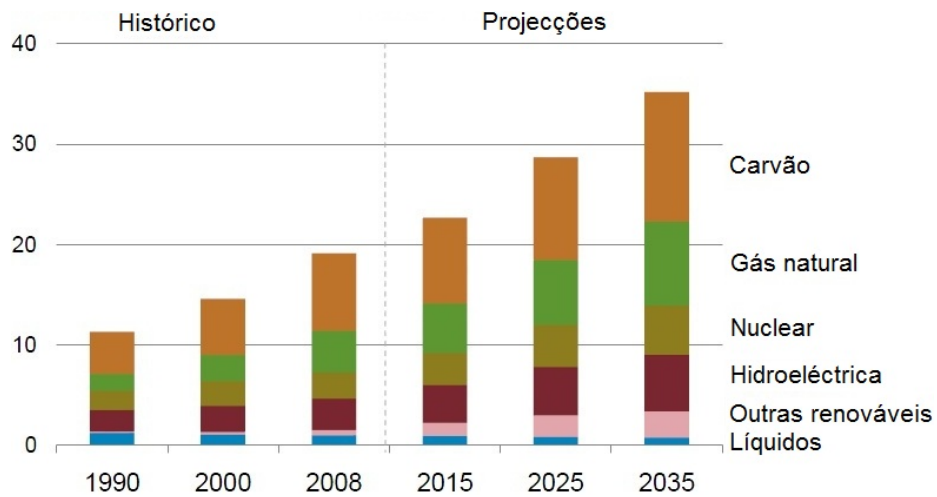


Fig. 2. Produção mundial de eletricidade (10^{12} kWh) (adaptado de [5])

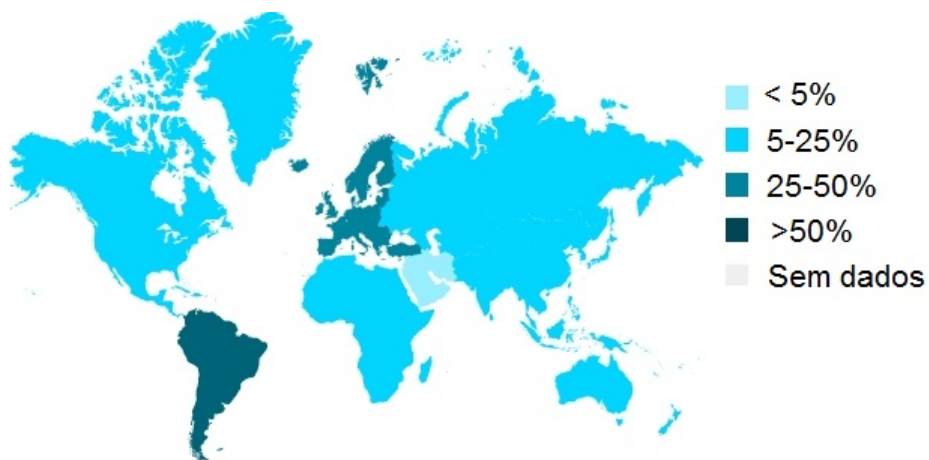


Fig. 3. Distribuição geográfica mundial da produção elétrica a partir de fontes renováveis (adaptado de [6])

1.1.1 Recurso solar em Portugal

Na fig. 1 pode ler-se o valor da irradiação global anual (kWh/m^2), medida numa superfície orientada para sul e com a inclinação ótima que permite captar o máximo de radiação solar. Os dados utilizados foram recolhidos entre 1981 e 1990. É também possível ler o valor anual da potência-pico (kWh/kWp) prevista para painéis fotovoltaicos (FV) fixos, com inclinação ótima para maximizarem a produção de energia elétrica. Observa-se ainda que o recurso solar disponível na Europa varia significativamente com a latitude.

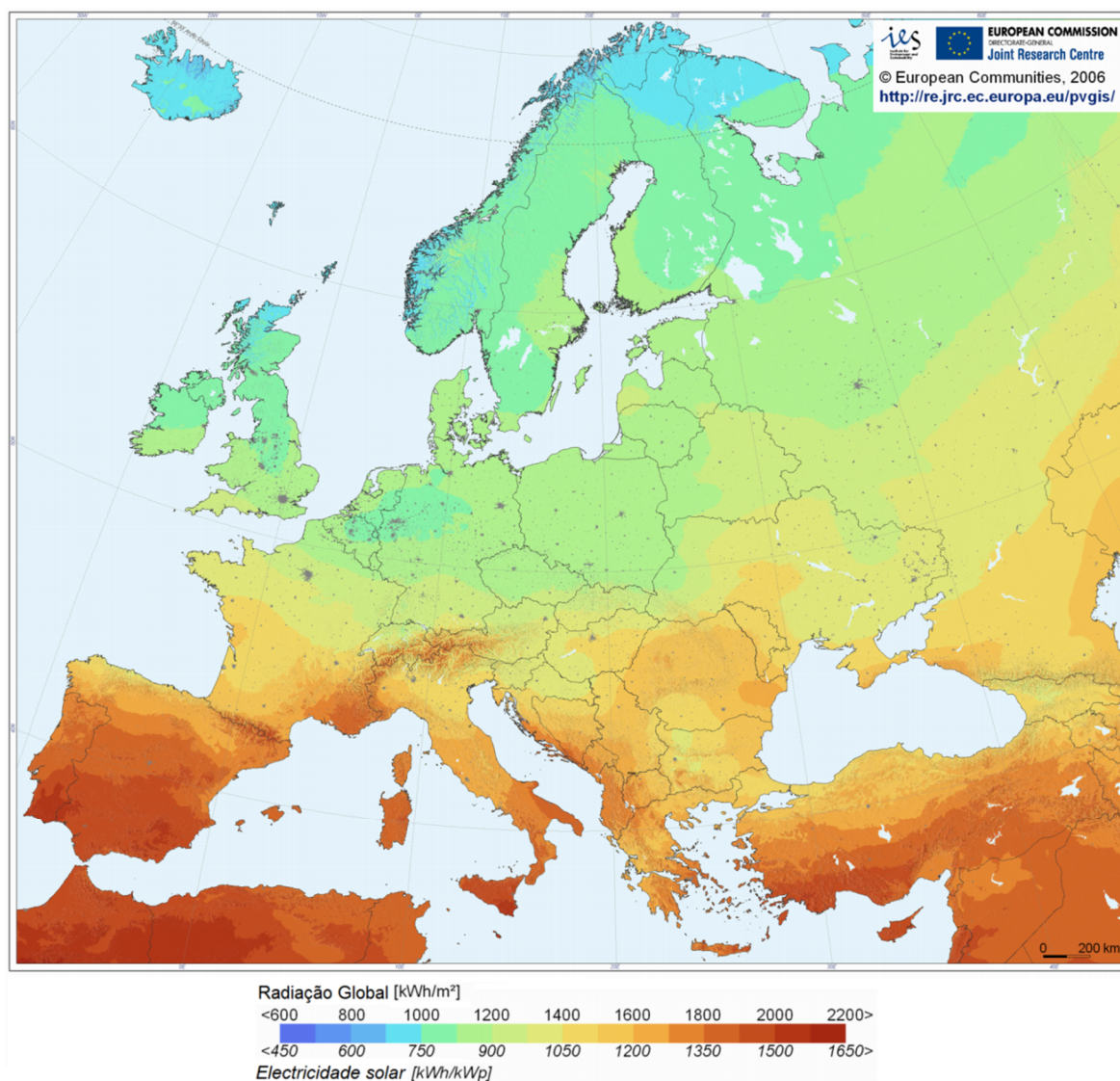


Fig. 4. Irradiação global anual na Europa em (kWh/m^2), numa superfície orientada para sul e com inclinação ótima [7]

Em Portugal a irradiação solar anual chega a atingir quase o dobro da média disponível nos países do norte da Europa. Observa-se que em Portugal, numa superfície com orientação fixa ótima, são atingidos valores de irradiação solar que variam entre 1.700 kWh/m^2 no Norte e 2000 kWh/m^2 no Sul. Analisando a figura, torna-se evidente que a região sudeste de Portugal tem um elevado potencial para aproveitamento de energia solar, designadamente para a produção de energia elétrica por via fotovoltaica.

O portal PVGIS-Europe disponibiliza os ganhos obtidos com a implementação de sistemas de seguimento solar a dois eixos em painéis FV, relativamente a um sistema fixo com orientação e inclinação ótima. Para Portugal, o estudo concluiu que é possível atingir ganhos de cerca de 30%, sendo que na região sudeste do Alentejo podem atingir os 40%. Nestas condições e nesta região seria possível obter valores anuais de potência-pico da ordem das 2100 horas.

Ganhos de 40% justificam desde logo o interesse no estudo e desenvolvimento de sistemas de seguimento solar, contudo há que ter em atenção o facto dos sistemas FV com seguimento solar necessitarem de uma maior área de terreno por unidade de potência-pico e terem uma manutenção mais dispendiosa, dado que possuem partes mecânicas móveis. O sobrecusto deve ser ponderado, face ao aumento de produção elétrica [3], [7], [8].

1.2 Motivação

A localização geográfica e as características climáticas tornam Portugal um dos países europeus com maior exposição solar, tanto em intensidade como em número de horas de exposição.

Estudos apresentados no portal PVGIS-Europe, sobre certas regiões do território português, apresentam ganhos, na ordem dos 40%, de produção de energia elétrica, quando se transita de um sistema FV fixo com orientação ótima para um sistema FV com seguimento solar a dois eixos.

É um dado adquirido o facto da irradiância incidente ser um dos principais fatores que condicionam a produção FV de uma instalação. A orientação que é definida para a instalação solar determina diferentes níveis de radiação captada. Maximiza-se a captação de radiação solar numa determinada superfície quando esta está posicionada perpendicularmente à radiação [7], [9].

Sabe-se que sistemas mecânicos de seguimento da posição do Sol possibilitam um aumento de energia produzida e que os ganhos podem atingir até 50% no verão e 300% no inverno, quando comparados com sistemas FV instalados na horizontal. É nas localizações de maior radiação que a contribuição destes sistemas no aumento de energia produzida tende a ser maior [9].

Compreendendo a vantagem na utilização de sistemas mecânicos de seguimento solar que suportam, em alguns casos, vários módulos de painéis FV, considera-se oportuno desenvolver um sistema que realize a tarefa de busca pela posição ótima para produção de energia elétrica, transmitindo essa posição ao referido sistema de seguimento. Pretende-se uma diminuição dos movimentos dos componentes mecânicos destes seguidores de grandes dimensões.

1.3 Objetivos

O protótipo [2] que serviu de base de trabalho para esta dissertação é composto essencialmente por um sistema eletromecânico de dois eixos motorizados com controlador, desenvolvido em ambiente Siemens STEP 7, com o objetivo de movimentar uma célula FV montada na estrutura. Nesse trabalho, foi implementada a configuração e comunicação base entre diversos componentes e o autómato, por forma a controlar o posicionamento angular dos motores passo-a-passo.

Os desenvolvimentos agora introduzidos têm o objetivo de tornar o protótipo que resulta desta dissertação, na base de trabalho para o desenvolvimento de um sistema de seguimento solar de dois eixos. Como tal, propôs-se a continuação do desenvolvimento da programação em ambiente STEP 7 de forma a explorar a capacidade de utilização das cartas SIMATIC FM 353 que controlam os movimentos dos motores passo-a-passo.

Assim, a presente dissertação tem os seguintes objetivos:

- Desenvolver uma aplicação em autómato programável Siemens S7-300 para fazer o varrimento no espaço dos dois eixos, armazenando os valores da potência elétrica produzida pelo painel solar, em cada localização;
- Detetar a posição que maximiza a produção do painel solar;
- Capacitar o sistema de deslocar o painel para o posicionamento ótimo, relativamente à irradiação incidente;
- Desenvolver uma aplicação para controlar e monitorizar o sistema localmente, através de um painel Siemens HMI SIMATIC OP 270, em ambiente Siemens WinCC Flexible;
- Criar um sistema de supervisão e controlo que permita monitorizar remotamente o posicionamento e a potência à saída da célula FV.

1.4 Estrutura da dissertação

A presente dissertação encontra-se dividida em nove capítulos.

O primeiro capítulo descreve brevemente o panorama energético mundial, destacando a energia solar como recurso energético a considerar em Portugal. Em sequência, refere-se a importância da implementação de sistemas de seguimento solar na maximização da produção de energia elétrica de um painel solar.

O segundo capítulo contém uma descrição essencial das tecnologias solares fotovoltaica e solar térmica e dos diferentes sistemas de seguimento solar.

No terceiro capítulo encontra-se a descrição do funcionamento do sistema desenvolvido e dos principais componentes que o constituem.

No quarto capítulo é apresentada a descrição do software utilizado para elaborar a programação de um microcontrolador. Este constitui na plataforma de desenvolvimento de um sistema secundário desenvolvido para auxiliar o sistema principal, ou seja, aquele que executa a movimentação da célula FV.

O quinto capítulo contém a criação do projeto STEP 7, são apresentadas noções básicas importantes para o desenvolvimento da programação, contém a estrutura do programa de utilizador e é descrita a implementação da funcionalidade de varrimento.

No sexto capítulo encontra-se a descrição da aplicação desenvolvida para o painel de monitorização e controlo local.

O sétimo capítulo apresenta o sistema de supervisão e controlo do protótipo.

No oitavo capítulo são apresentados resultados obtidos e o nono capítulo contém as conclusões finais.

Capítulo 2 - Tecnologia solar

A conversão ou utilização da energia associada à radiação solar pode classificar-se em ativa ou passiva. Diz-se ativa quando a energia solar é diretamente transformada em energia térmica ou elétrica. Por sua vez, denomina-se passiva quando a energia solar é aproveitada para aquecimento passivo de edifícios através de elementos de arquitetura e de construção [3], [8].

2.1 Central solar fotovoltaica

Numa central solar fotovoltaica usam-se painéis fotovoltaicos (FV), constituídos por exemplo por um material com base em silício a que se juntam substâncias dopantes, que depois de adequadamente tratado, apresenta a propriedade de libertar eletrões quando exposto à radiação solar. O movimento orientado dos eletrões estabelece a criação de uma corrente elétrica. Estabelece-se assim o efeito fotovoltaico, ou seja, a conversão direta da potência associada à radiação solar em potência elétrica (corrente contínua, DC).

O elemento mais pequeno de um sistema fotovoltaico designa-se por célula e gera potências na ordem de 1.5 W. Para obter maiores potências ligam-se células em série e/ ou em paralelo, formando módulos tipicamente com potências DC entre 100 e 200 W. A agregação destes módulos dá origem a painéis fotovoltaicos. Da mesma forma, para aumentar a potência, podem-se conectar vários painéis solares constituindo-se uma central fotovoltaica. Para injetar a potência DC gerada na rede elétrica AC (corrente alternada) é necessário um inversor (conversor DC/AC).

Contudo, o rendimento da referida conversão é baixo, cerca de 15% [3]. Atualmente num conjunto de equipamentos otimizados, 1000 W de energia solar incidente originam cerca de 190 W, o que representa 19% de eficiência. Este baixo desempenho implica a utilização de uma vasta área para a instalação de centrais fotovoltaicas com potências significativas (50 MW – 100 MW) [2].

Devido à alteração da quantidade de energia proveniente da radiação solar e à variação da temperatura das células FV, a potência de saída de um sistema de painéis solares não é constante.

Por forma a diminuir custos e a aumentar o rendimento de conversão energética, realizam-se estudos nas tecnologias de filmes finos. Por outro lado, implementam-se centrais fotovoltaicas com sistemas de seguimento solar, desenvolvendo-se estudos para otimizar a localização do ponto de potência máxima (MPP – Maximum Power Point). Sistemas que funcionam com base neste princípio apresentam melhorias de rendimento e possibilitam a integração de sistemas fotovoltaicos em zonas para além de terrenos aplanados, como por exemplo em edifícios [1], [2], [3].

2.1.1 Sistemas de seguimento solar em dois eixos

A eficiência dos sistemas fotovoltaicos depende de vários aspetos podendo ser enumerada a tecnologia e a qualidade das células FV, a qualidade dos reguladores e dos conversores, as condições atmosféricas (p.e. a sombra das nuvens, a sujidade sobre os painéis, a temperatura e o vento e a qualidade dos seguidores solares).

Existem três tipos de seguidores:

- Com dois eixos (azimutal e zenital);
- Com um eixo azimutal;
- Com um eixo zenital.

Ao longo do dia ocorre uma variação azimutal, relacionada com a orientação horizontal do Sol do Leste até o Oeste. Ao longo do ano ocorre a variação zenital, com um máximo de altura no Solstício de verão e um mínimo no Solstício de inverno, devido ao facto de que a Terra possui uma inclinação de 23.27° .

Os seguidores de um eixo possuem componentes mais simples e por isso mais baratos. Contudo, não apresentam uma eficiência tão alta para instalações FV, pois apenas permitem que estas coloquem os módulos numa posição perpendicular relativamente a um dos planos da irradiação solar.

Os sistemas fotovoltaicos com seguimento solar a dois eixos são projetados para orientar as superfícies coletoras para que a sua normal fique exatamente alinhada com a posição do Sol no céu. Assim, processam sempre a componente máxima da irradiância direta incidente e maximizam a energia eléctrica produzida, durante todo o dia e todo o ano.

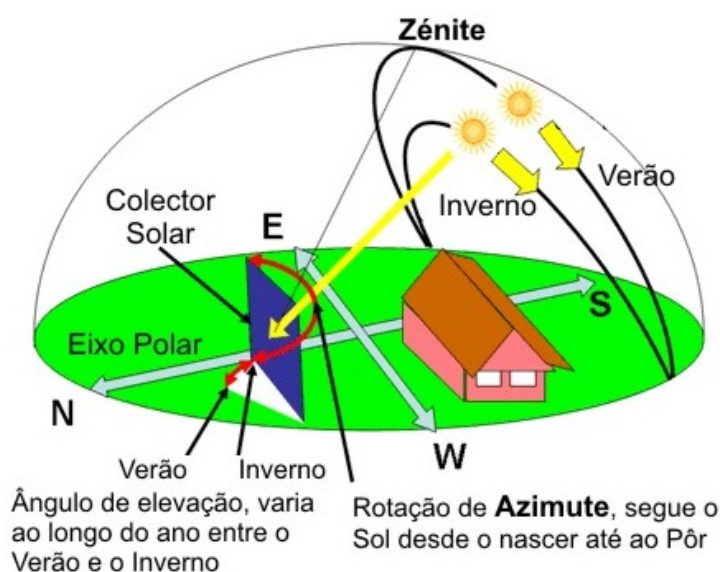


Fig. 5. Ilustração da variação azimutal e zenital (ângulos solares)

Como já foi referido nesta dissertação, vários estudos indicam que a implementação de um sistema de seguimento solar em dois eixos pode resultar num aumento de produção de energia, na ordem dos 40%, em regiões do Sul da Europa. Usando um modelo matemático para determinar o ângulo de incidência [3] e conhecendo os dados de irradiância direta numa superfície fixa, medidos p.e. com um piranómetro, é possível calcular a respetiva componente máxima.

A figura 6 mostra a diferença entre a captação de radiação entre uma instalação com sistema de dois eixos e uma instalação sem sistema de seguimento, ao longo do dia 20 de dezembro e ao longo do dia 21 de junho, ambos sem nuvens, para uma latitude de 50° .

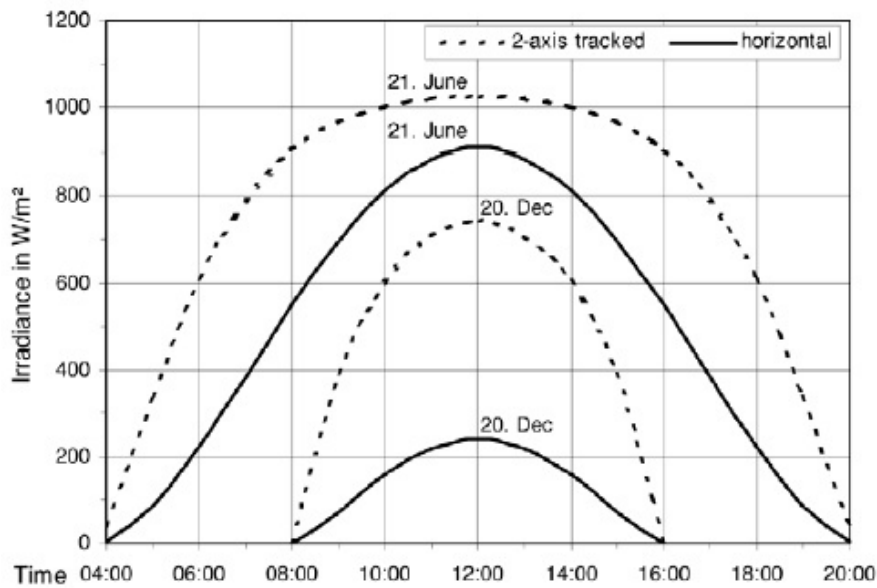


Fig. 6. Diferença de captação de radiação entre uma instalação com sistema de dois eixos e uma instalação sem sistema de seguimento [26]

Para além de se classificar os sistemas de seguimento solar relativamente ao número de eixos, também estes também se distinguem entre sistemas do tipo passivo, ativo, cronológico ou misto:

- *Tipo passivo* – É o tipo de seguidor mais básico e caracteriza-se por não apresentar qualquer tipo de compensação relativamente à posição solar. Apenas se move de uma forma pré-determinada e sem nenhum tipo de entrada externa que o compense. Atualmente investiga-se sobre a aplicação da termo-expansão de materiais a este tipo de seguidores, dado que a sua vantagem em termos da simplicidade do dispositivo poderá ser determinante;
- *Tipo ativo* – É o tipo de sistema mais comum. Fazem parte da sua constituição um ou mais sensores, para determinação da posição do Sol. Caracterizam-se também por possuírem a capacidade de se deslocarem em função dessa determinação. Alguns sistemas até tendem a compensar a posição do módulo quando existem nuvens e estes não recebem radiação direta;
- *Tipo cronológico* – O seu princípio de funcionamento baseia-se na memorização das posições solares para todo o ano, dado que de dia para dia a sua inclinação vai variando, bem como a hora a que começa a seguir o Sol e o período de seguimento;
- *Tipo misto* – É a melhor solução por fazer a combinação entre um sistema que observa o seu calendário interno, mas que em caso de necessidade utiliza a leitura dos sensores para fazer a compensação.

2.2 Sistema solar térmico

Destina-se a aproveitar a energia solar para aquecer água. O princípio de funcionamento é simples: a radiação solar atinge um coletor solar, de vidro, para criar efeito de estufa, aquecendo um fluido térmico, em geral água com anticongelante; um permutador de calor transmite o calor armazenado no fluido térmico para a água de consumo. Este tipo de sistemas possui rendimento na ordem dos 30% e encontra-se bastante difundido [3].

2.3 Central solar térmica de concentração

Nestes sistemas a radiação solar é focada, com o auxílio de superfícies espelhadas orientadas por sistemas de seguimento da posição solar, sobre um recetor, de modo a obter calor de alta temperatura, suscetível de produzir vapor. Neste sentido, têm-se verificado esforços para desenvolvimento da robótica de seguimento solar. A partir deste ponto, a produção de energia elétrica segue o princípio das centrais térmicas convencionais. A disseminação destes sistemas ainda é diminuta, pelo rendimento que se compara ao dos painéis fotovoltaicos e o seu custo ainda ser elevado [3], [8].

Tabela 1. Classificação de coletores solares [10]

Movimento	Tipo de coletor	Absorvedor	Rácio de concentração	η (%) médio	Temperatura (°C)
Estacionário	Coletor Plano (FPC)	Plano	1		30-80
	Coletor de Tubo de Vácuo (ETC)	Plano	1		50-200
Um eixo	Coletor Parabólico (CPC)	Tubular	1-5		60-240
			5-15		60-300
	Sistema de Canal Fresnel (LFR)	Tubular	10-40	[9; 11]	60-250
	Sistema de Canal Cilíndrico (CTC)	Tubular	15-50		60-300
Dois eixos	Sistema de Canal Parabólico (PTC)	Tubular	10-85	[10; 15]	60-400
	Sistema de Disco Parabólico (PDR)	Ponto	600-2000	[16; 18]	100-1500
	Sistema de Torre Solar (HFC)	Ponto	300-1500	10	150-2000

Nota: O rácio de concentração é definido como a razão entre a área de abertura e a área de absorção do coletor

Vantagens:

- Trata-se de uma tecnologia solar que não acarreta o elevado custo da utilização de materiais semicondutores;
- É alcançada uma maior eficiência termodinâmica, dado que o fluido de trabalho pode atingir uma temperatura mais elevada, conseguida através da redução da área onde podem ocorrer perdas de calor. Verifica-se uma diminuição da área do recetor por unidade de energia solar recolhida;

- Esse aumento de eficiência pode tornar economicamente viável o tratamento de superfícies e o isolamento por vácuo, necessários neste tipo de sistemas.

Desvantagens:

- As superfícies refletoras podem tornar-se menos eficientes com o passar do tempo, requerendo limpeza e substituições periódicas;
- É necessária a implementação de um sistema de seguimento solar (custo; manutenção) [10].

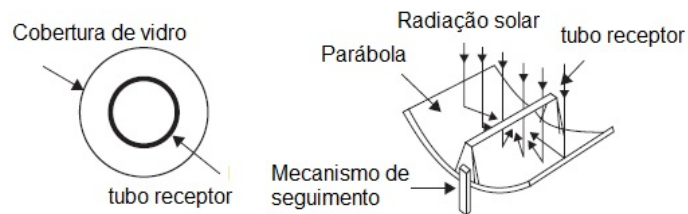


Fig. 7. Detalhe de um tubo receptor, esquema de um coletor parabólico (adaptado de [10])

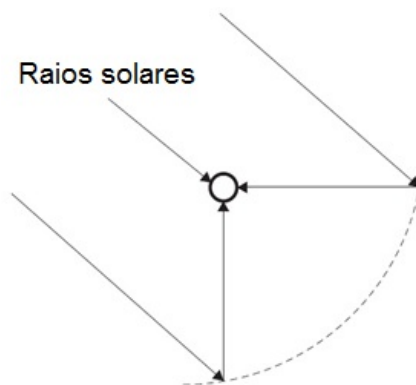


Fig. 8. Esquema simplificado do princípio de funcionamento de um LFR (adaptado de [10])

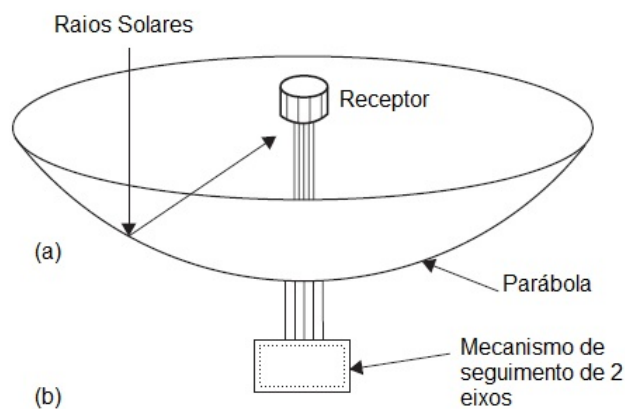


Fig. 9. Esquema simplifica do princípio de funcionamento de um coletor de disco parabólico (adaptado de [10])

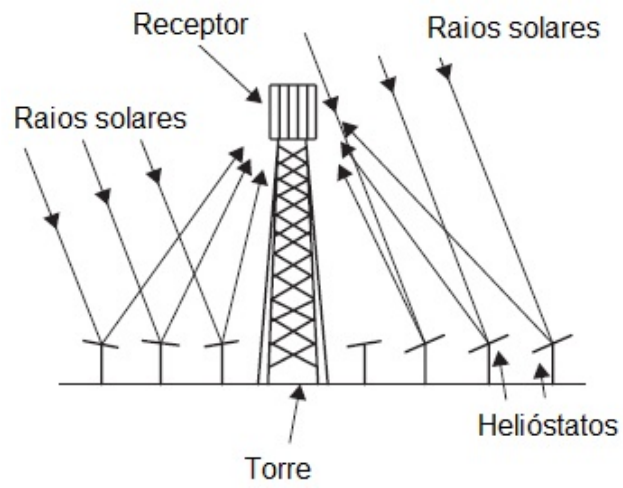


Fig. 10. Esquema do princípio de funcionamento de uma Torre de Concentração de Energia Solar (adaptado de [10])

Capítulo 3 – Descrição do sistema

3.1 Descrição dos componentes e do funcionamento

O protótipo desenvolvido é constituído por dois subsistemas.

O primeiro consiste num mecanismo em alumínio normalizado, com dois graus de liberdade, que suporta uma célula FV (150 mm x 150 mm). Possui dois eixos motorizados por motores passo-a-passo, com sensores de ângulo incorporados, ligados a módulos funcionais (FM353). O posicionamento dos eixos é aferido por encoders externos também ligados a cartas específicas (FM350-1). Estes módulos funcionais fazem parte da constituição de um autómato programável (PLC) que controla os motores passo-a-passo e o funcionamento de todo o sistema. Na figura 11 pode observar o mecanismo.

Ao nível do subsistema principal foi criada uma rede para controlo local e uma rede para controlo remoto. São dadas instruções de funcionamento ao sistema a partir do um painel de interface homem máquina (figura 12), onde também é possível monitorizar o posicionamento e a potência elétrica da célula FV (capítulo 6). Este encontra-se conectado ao PLC por rede PROFIBUS. O PLC possui um módulo de comunicação o que permitiu a criação um sistema de supervisão e o mecanismo pode assim ser monitorizado e controlado remotamente (capítulo 7).

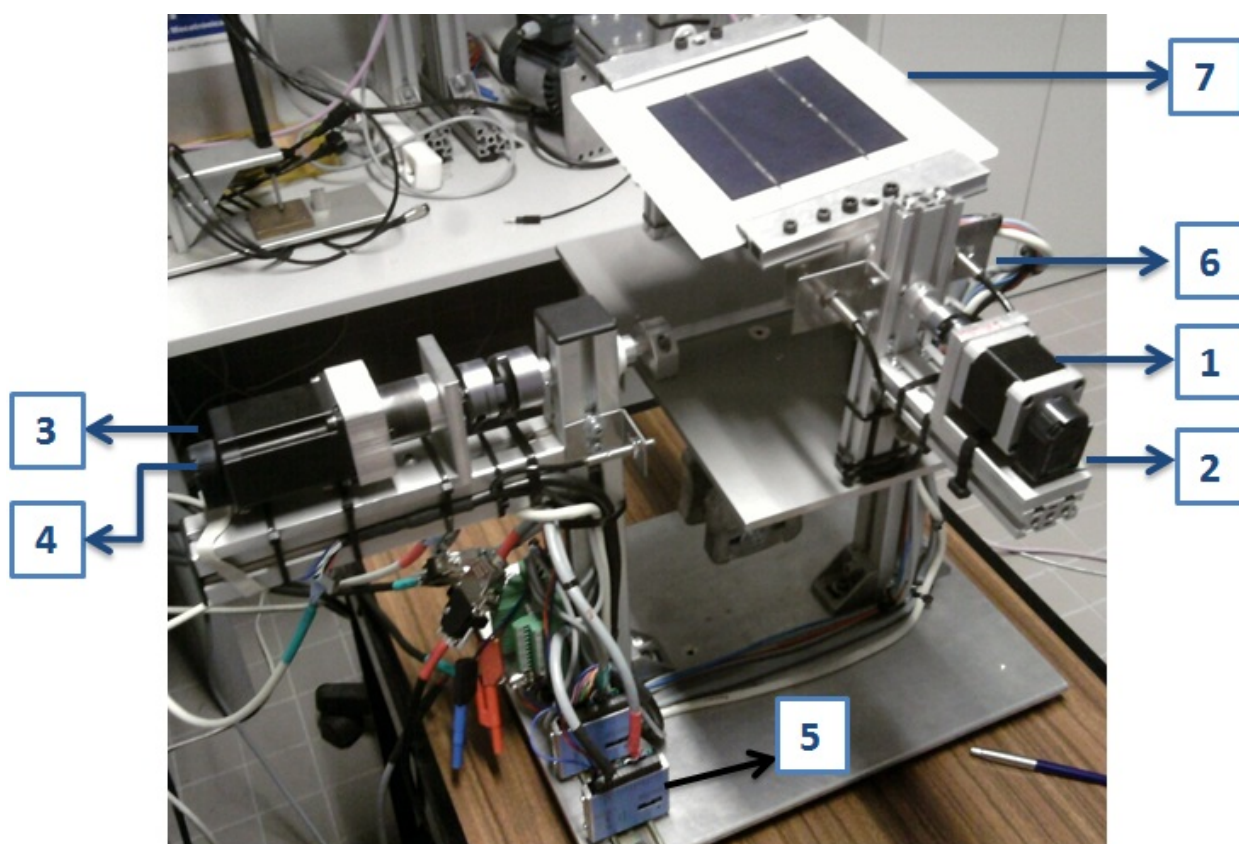


Fig. 11. Fotografia do protótipo do sistema principal

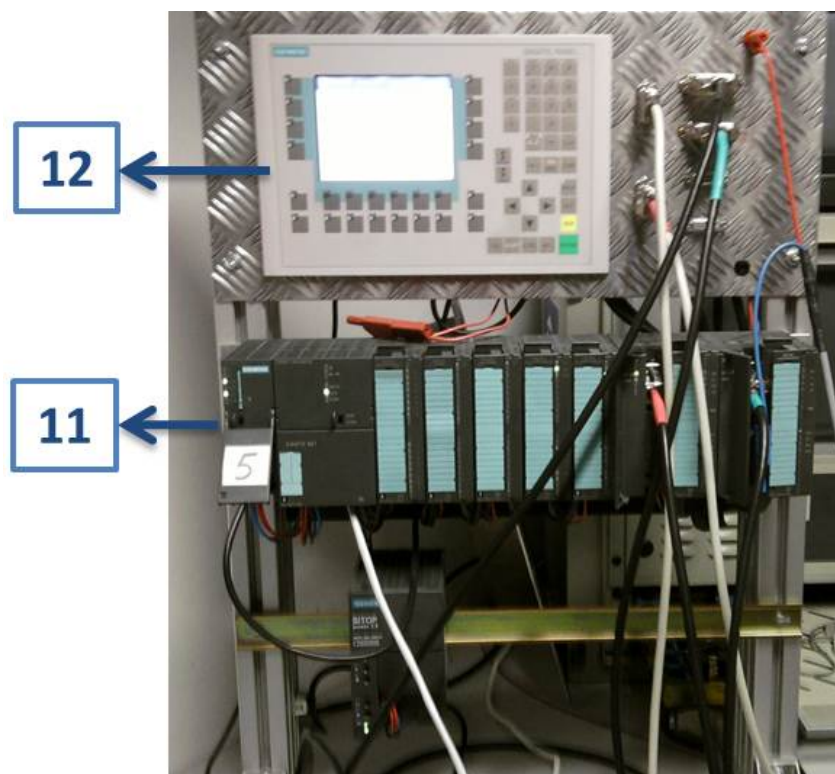


Fig. 12. PLC e painel HMI constituintes do sistema principal

Tabela 2. Identificação dos componentes principais do protótipo

N.º	Componente
1	Motor passo-a-passo n.º 1 (eixo 1)
2	Sensor de ângulo n.º 1
3	Motor passo-a-passo n.º 2 (eixo 2)
4	Sensor de ângulo n.º 2
5	Drivers de controlo de motor passo-a-passo
6	Sensores de proximidade (posição inicial)
7	Célula fotovoltaica
8	Conversor de irradiância para frequência
9	Microcontrolador
10	Saídas do microcontrolador
11	PLC
12	Painel HMI

A figura 13 apresenta o subsistema secundário. É constituído por quatro dispositivos optoelectrónicos programáveis para medição de irradiância (sensores de luz), conectados a um microcontrolador.

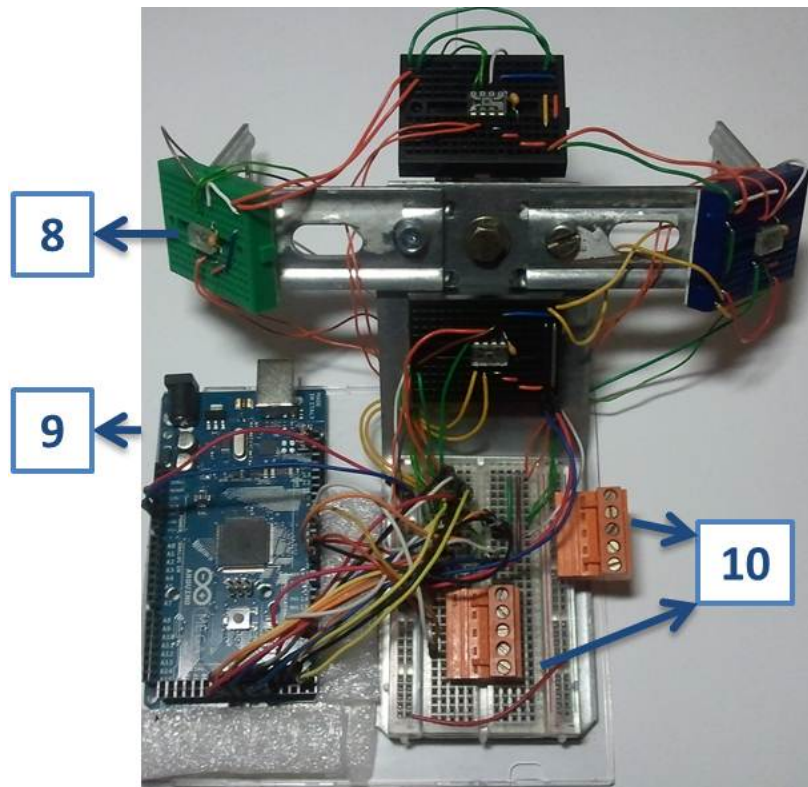


Fig. 13. Principais componentes do sistema secundário

3.1.1 Aspetos gerais do funcionamento do sistema

- O subsistema principal executa um varrimento no espaço dos dois eixos, enquanto regista o valor máximo de potência elétrica à saída da célula fotovoltaica;
- Genericamente, o subsistema secundário consiste em quatro conversores programáveis de irradiância para frequência, fixos numa estrutura, em quatro posições e com diferentes orientações, na proximidade da célula FV;
- Estes estão alimentados e programados por meio de um microcontrolador, cuja saída, em tensão elétrica do tipo PWM, está conecta a uma entrada do PLC. A sua função é seleccionar o varrimento solar mais indicado de acordo com a localização da fonte luminosa;
- O varrimento no espaço dos dois eixos foi dividido de forma a tornar o sistema mais rápido e energeticamente eficiente, através da diminuição do tempo de funcionamento dos motores passo-a-passo;
- A plataforma do subsistema de varrimento (principal) assenta num PLC que possui os módulos necessários à realização das seguintes tarefas:
 - Determinação da posição inicial dos eixos motorizados, por meio de sensores de proximidade indutivos;

- Determinação do posicionamento angular dos motores passo-a-passo (eixo 1 e eixo 2), por meio de sensores de ângulo;
 - Leitura da saída em tensão elétrica do microcontrolador;
 - Comando de movimento dos motores passo-a-passo, que acionam os dois eixos motorizados.
- Criou-se uma aplicação para interface homem máquina local: operação do sistema e leitura de variáveis;
 - Desenvolveu-se uma aplicação SCADA Web enable, para monitorizar os dados do subsistema principal, nomeadamente o posicionamento ótimo e a potência máxima de saída da célula FV;
 - Com a implementação da monitorização, é possível verificar a cada momento se a posição em que o mecanismo se encontra é a mais favorável à produção de energia eléctrica.

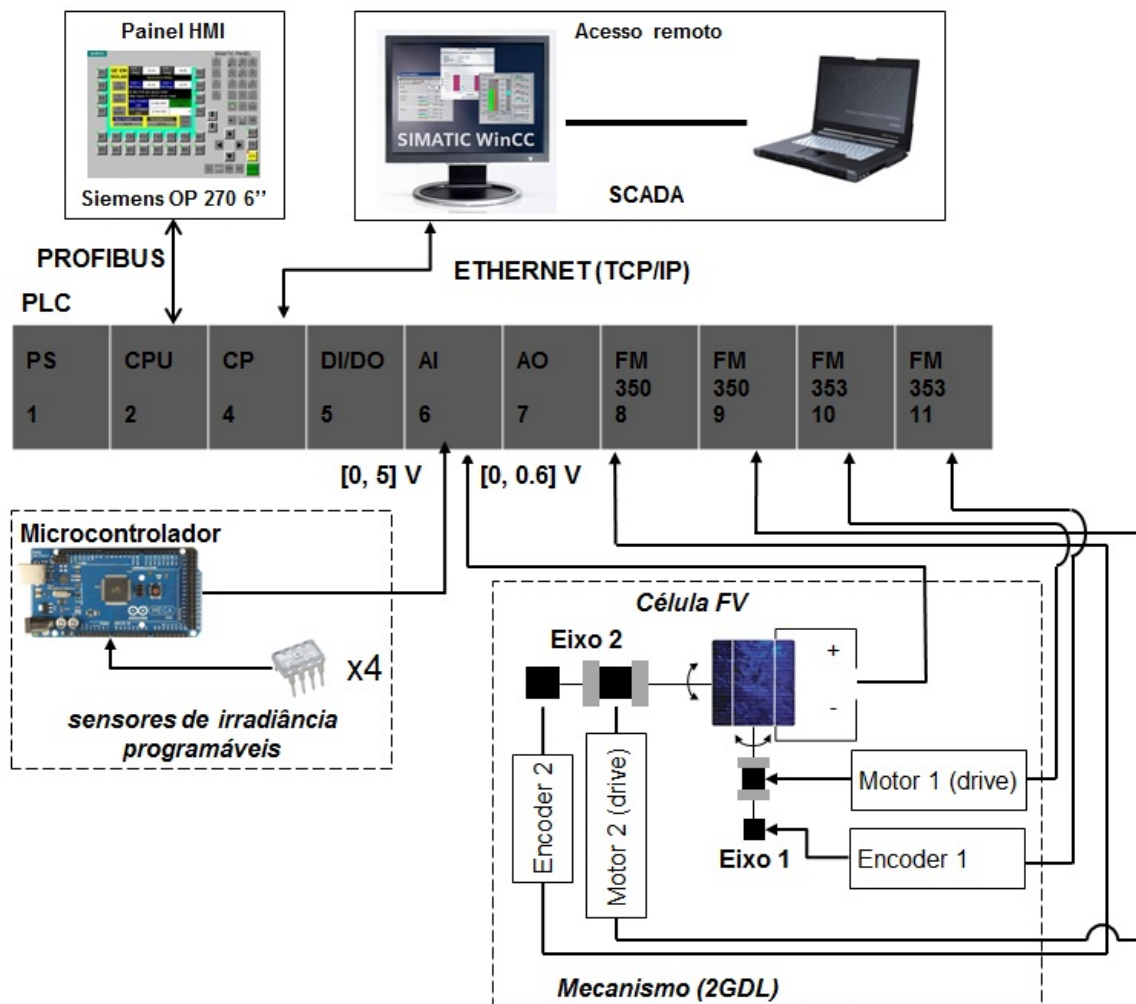


Fig. 14. Diagrama de blocos geral do sistema

Caraterísticas técnicas da célula FV

O elemento sensor do sistema é uma célula FV de silício policristalino com as caraterísticas a seguir representadas. Importa salientar que a resistência colocada em paralelo na saída da célula é de 0,1 Ohm.

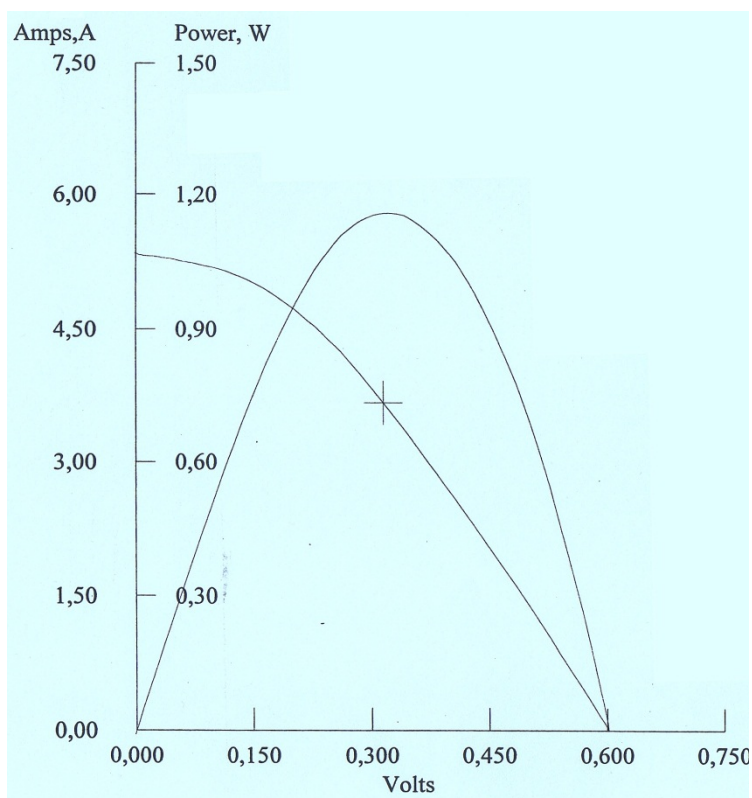


Fig. 15. Curva I-V da célula FV

Tabela 3. Caraterísticas da célula FV

Característica	Valor
Irradiância	100 mW/cm ²
Temperatura	25 °C
Tensão de circuito aberto	0,60 V
Corrente de curto-circuito	5,342 A
Potência máxima	1,12 W
Resistência em série	0,073 Ohm
Resistência em paralelo	0,601 Ohm
Tensão no ponto de potência máxima	0,32 V
Corrente no ponto de potência máxima	3,560 A
FF	0,349

3.2 Descrição técnica do equipamento – Subsistema principal

3.2.1 Descrição do controlador

A comunicação com os *drivers* de controlo dos motores passo-a-passo, a leitura dos seus movimentos angulares, da tensão elétrica produzida pela célula FV, dos resultados provenientes do microcontrolador, a comunicação com o painel de operação, bem como a comunicação com o sistema de supervisão, estão interligadas por meio da programação desenvolvida, transferida e armazenada na unidade de controlo deste sistema, o PLC. Este dispositivo, de estrutura modular, é constituído pelos módulos necessários à realização das tarefas mencionadas.

Por definição, um controlador lógico programável (PLC) é um dispositivo eletrónico digital, baseado num microcontrolador, possui uma memória programável para armazenamento interno de instruções e implementação de funções lógicas, sequenciamento, temporização, contagem e aritméticas. O controlador usado no projeto é apto para tarefas de controlo, de integração de tecnologias e de armazenamento de informações. Trata-se de um controlador modular SIMATIC S7 Siemens ® com CPU 315-2DP, que possui dois tipos de interfaces:

- **MPI (Multipoint Interface)**: tem função de entrada para estação de programação (PG/PC com software STEP 7); serve para estabelecer comunicação entre o PLC e diversos dispositivos Siemens ®, como por exemplo painéis HMI; a velocidade de transmissão por defeito é 187.5 kbit/s; os endereços dos nós MPI têm de ser únicos definidos na estação de programação; RS 485 é o meio de transmissão elétrico padrão [11].
- **PROFIBUS DP (Process Field Bus – Decentralized Peripherals)**: é um protocolo livre; aplica-se para estabelecer comunicação entre a unidade central (CPU do PLC) e dispositivos descentralizados, como unidades distribuidoras de I/Os; painéis HMI (OP/TP); válvulas comutadas eletricamente; drives, etc...; usa-se também com uma forma prática de estabelecer comunicação em rede com outros PLCs (DP Master e DP Slave); no caso específico do PLC utilizado, surge no meio físico RS-485, todavia também existe em fibra ótica; suporta velocidades de transmissão compreendidas entre 9 kbit/s e 12 Mbit/s; é determinístico e garante curtos tempos de resposta [11].

O princípio de funcionamento de um PLC baseia-se na execução de um ciclo de varrimento sequencial das etapas programadas e transferidas para a sua memória interna. O tempo necessário para executar um ciclo é denominado *clock*. A não simultaneidade das operações justifica a implementação de processadores com velocidades cada vez mais elevadas.

Quando um PLC é colocado em funcionamento (início) executa uma verificação da CPU, memórias, entradas e saídas, existência de programa de utilizador (blocos), existência de configuração de *hardware*, se existir alguma falha emite um aviso e desativa todas as saídas.

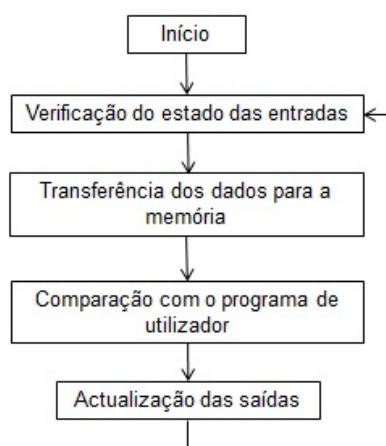


Fig. 16. Ciclo de varrimento genérico de um PLC

3.2.2 Módulos constituintes do PLC

Tabela 4. Configuração de hardware do PLC Siemens ® S7-300

Posição	Função	Módulo	Número de ordem
N.D.	Fonte de alimentação	SITOP power 2.5	6EP1 332-1SH12
2	CPU	CPU 315-2DP (128	6ES7315-2AG10-0AB0
3	Reservado (módulos de interface): para aumento do número de módulos utilizados		
4	Comunicação (<i>Ethernet</i>)	CP 343 Advanced-IT	6GK7 342-5DA02-0XE0
5	Entradas e saídas digitais	DI8/DO8xDC 24V/0.5A	6ES7 323-1BH01-0AA0
6	Entradas analógicas	AI 8x12 Bit	6ES7 331-7KF02-0AB0
7	Saídas analógicas	AO 4x12 Bit	6ES7 332-5HD01-0AB0
8	Módulo funcional para contador	FM 350 Counter	6ES7 350-1AH03-0AE0
9	Módulo funcional para contador	FM 350 Counter	6ES7 350-1AH03-0AE0
10	Módulo funcional para motor passo-a-passo	FM 353 Stepper	6ES7 353-1AH01-0AE0
11	Módulo funcional para motor passo-a-passo	FM 353 Stepper	6ES7 353-1AH01-0AE0

Tabela 5. Componentes necessários para implementar o varrimento solar

Dispositivo	Eixo	Referência	Fabricante
Motor passo-a-passo	1	ST4118L0804	Nanotec
	2	ST5918L1008	
Drive (motor)	1	SMC32	
	2	SMC32	
Encoder	1	HEDS-5540-A14	Hewlet Packard
	2	HEDS-5540-H06	

3.2.3 Descrição técnica dos componentes do PLC

3.2.3.1 Fonte de alimentação

Designação: Siemens SITOP Modular 2.5A

Função: Fonte de alimentação universal, comutada
Alimentação elétrica estabilizada dos módulos constituintes

Principais características:

Entrada: 120-230V AC (110-350V DC); 50/60Hz

Saída: 24V DC/ 2.5 A



Fig. 17. Fotografia da fonte de alimentação do PLC

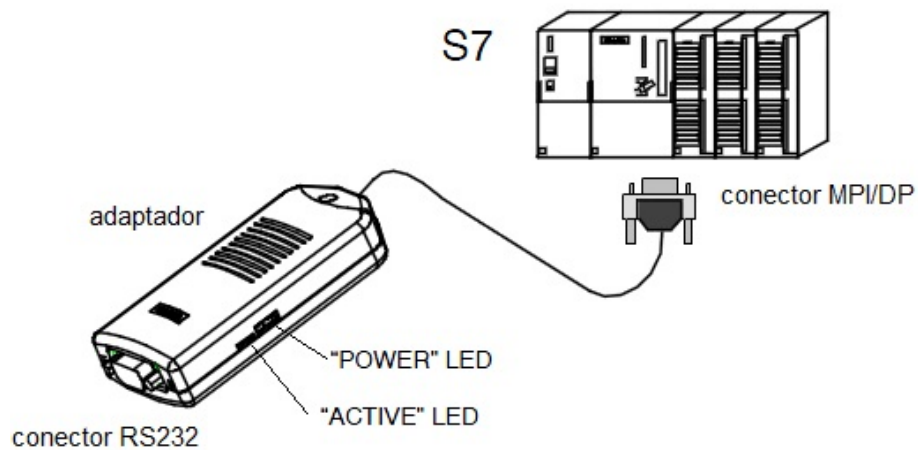


Fig. 18. Adaptador RS232/MPI e dispositivos conectados em rede MPI [11]

3.2.3.2 CPU Siemens 315-2 DP

Tabela 6. Principais características da CPU do PLC S7-300 [12]

Característica	Dados técnicos
<i>Firmware</i>	V2.6
Alimentação elétrica	24 VDC
Memória RAM integrada	128 kB + micro cartão de memória (máx.8 MB)
<i>Bit memory</i>	2048 bytes
<i>Data blocks</i>	1 – 1023 (16 KB)
Blocos (DBs, FCs, FBs)	1024
Apenas FBs	1024 (16 KB)
Apenas FCs	1024 (16 KB)
Linguagens de programação	LAD/FBD/STL
Tempo de processamento – operações bit	Min 0.1 μ s
Tempo de processamento – instruções word	Min. 0.2 μ s
Tempo de processamento – vírgula flutuante	Min. 3.0 μ s
Contadores	Max. 256; Janela de valores: 0 a 999
Temporizadores	Max. 256; Janela de valores: 10ms a 9990s
Entradas (<i>process image</i>)	128
Entradas (<i>digital channels</i>)	Max. 16384
Saídas (<i>process image</i>)	128
Saídas (analog channels)	Max. 1024
Comunicação – Interface 1	MPI (RS 458)
Comunicação – Interface 2	PROFIBUS DP (RS 458)

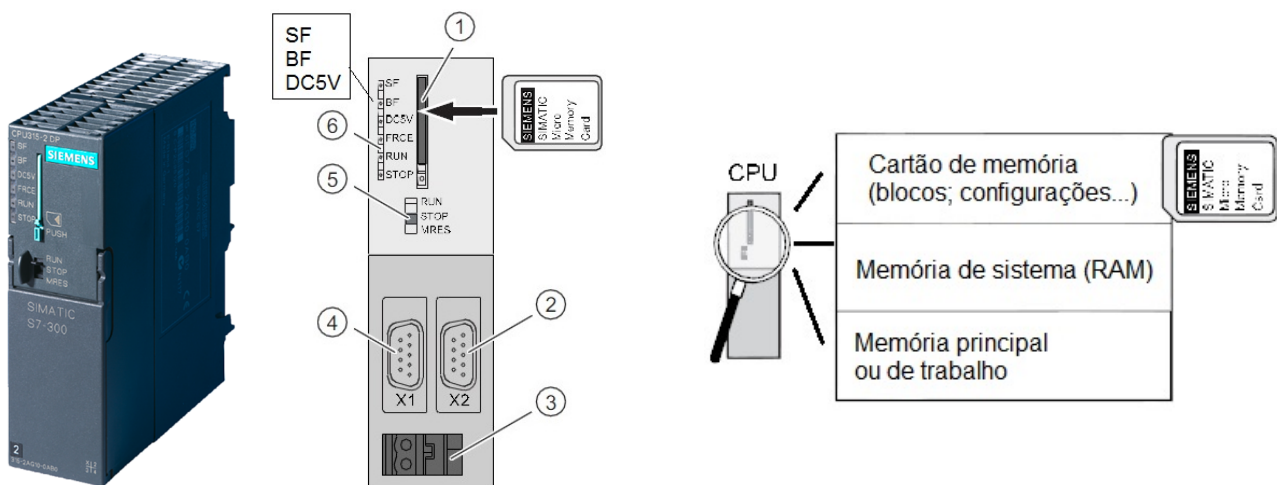


Fig. 19. Esquema da sinalética e das portas de interface da CPU do PLC S7-300 [12]

Tabela 7. Designação das portas de interface da CPU

Numeração	Designação
1	Ranhura para micro cartão de memória
2	Interface (X1) PROFIBUS DP
3	Conexão para alimentação
4	Interface (X2) MPI
5	Seletor de modo de funcionamento: MRES / STOP / RUN
6	LEDs sinalizadores de estado e erros
SF	Vermelho (configuração/programação): erro de hardware ou software
BF	Vermelho: Erro de interface (X2) PROFIBUS DP
DC5V	Verde: Alimentação para CPU e BUS S7-300

Descrição dos diferentes tipos de memória da CPU

Cartão de memória (*load memory*):

Tem como função armazenar blocos de programação e informação de sistema (*system data*), ou seja, configuração de hardware (módulos existentes), parametrização de módulos, etc... Desta forma, para este modelo de CPU, o cartão de memória é essencial para a transferência do código realizado no STEP 7 para o PLC. A memória aqui armazenada é do tipo rententiva, ou seja, permanece mesmo com o desligar da alimentação elétrica da CPU [12].

Memória de sistema, ou memória RAM, (*system memory*):

Contém a área de endereços (bits de memória, temporizadores, contadores, etc...).

Alguns endereços permitem armazenamento retentivo. Da mesma forma, a retentividade do endereço MPI e a velocidade de transmissão, permitem que dispositivos continuem a comunicar mesmo após uma reinicialização do PLC ou limpeza da informação do cartão de memória. Processamento da informação das entradas e saídas (I/Os); memória temporária (*local data*) [12].

Memória de trabalho:

Onde o código é executado e onde se processa o programa de utilizador (blocos programados no STEP 7). Os “programas” apenas correm nas memórias de sistema e de trabalho [12].

O programa de utilizador elaborado no STEP 7, armazenado no disco da estação de programação (PC), transfere-se por MPI para a CPU ficando guardado no cartão de memória. No modo de execução (PLC ligado), o conteúdo relevante, armazenado no cartão de memória, é transferido para a memória de trabalho, para a execução das tarefas programadas.

Tabela 8. Descrição das áreas de endereçamento da memória de sistema (RAM) [12]

Áreas de endereçamento	Descrição
Processamento de imagem de entradas (0 a 2048 bytes)	No início da execução de um ciclo do bloco OB1, a CPU lê os valores das entradas de cada módulo do PLC e guarda-os numa área de memória denominada <i>process image of inputs</i> .
Vantagens	Os valores dos sinais de entrada permanecem os mesmos durante a execução de um ciclo do programa de utilizador, mesmo que o valor dos sinais de entrada se modifiquem durante esse processo, apenas sendo alterados na memória de imagem de entrada após a realização completa do ciclo. Há um rápido acesso ao valor das entradas pelo facto destes estarem já armazenados na memória da CPU.
Processamento de imagem de saídas (0 a 2048 bytes)	Durante um ciclo, o programa de utilizador calcula os valores das saídas e escreve-os numa área de memória denominada <i>process image of outputs</i> . No fim do ciclo a CPU disponibiliza esses valores nas saídas dos módulos existentes.
Memória bit	Espaço de memória para armazenamento de valores intermédios obtidos durante a execução de um ciclo.
Temporizadores	Espaço de memória disponível para temporizadores.
Contadores	Espaço de memória disponível para contadores.
Dados temporários (locais)	Espaço de memória temporária, destinado a blocos (OB, FC e FB) enquanto são editados, durante a execução do programa de utilizador.
Blocos de dados	Espaço de memória para guardar valores nas diversas variáveis utilizadas no programa de utilizador.

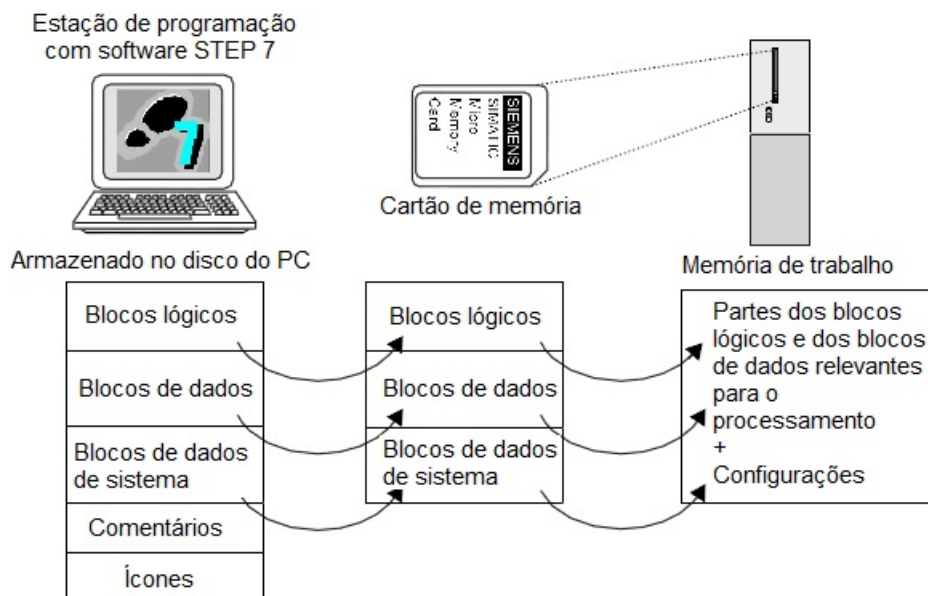


Fig. 20. Transferência de informação entre software PC, cartão de memória e CPU do PLC [12]

3.2.3.3 Módulo de comunicação CP-343 Advanced-IT

Módulo de comunicação da família S7-300:
Ethernet Industrial

- PROFINET
- TCP/IP
- SEND/RECEIVE
- S7-COMM, etc...
- Velocidade de transmissão: 10/100 Mbit/s
- Conector RJ45



Fig. 21. Módulo de comunicação (CP) do PLC S7-300 utilizado [Siemens]

3.2.3.4 Módulo de entradas e saídas digitais

- Entradas (8) e saídas (8) eletricamente isoladas
- Características elétricas (24 VCD/0.5 A)
- Sinal “1”: tensão de entrada compreendida entre 13V e 30V
- Sinal “0”: tensão de entrada compreendida entre -30V até 5V
- Tensão de saída 24 VDC
- Entradas: interruptores e sensores de proximidade (2, 3 e 4 fios)
- Saídas: comutação de válvulas solenoides, contactores DC, luzes, p.e.



Fig. 22. Módulo de entradas e saídas digitais do PLC S7-300 utilizado [Siemens]

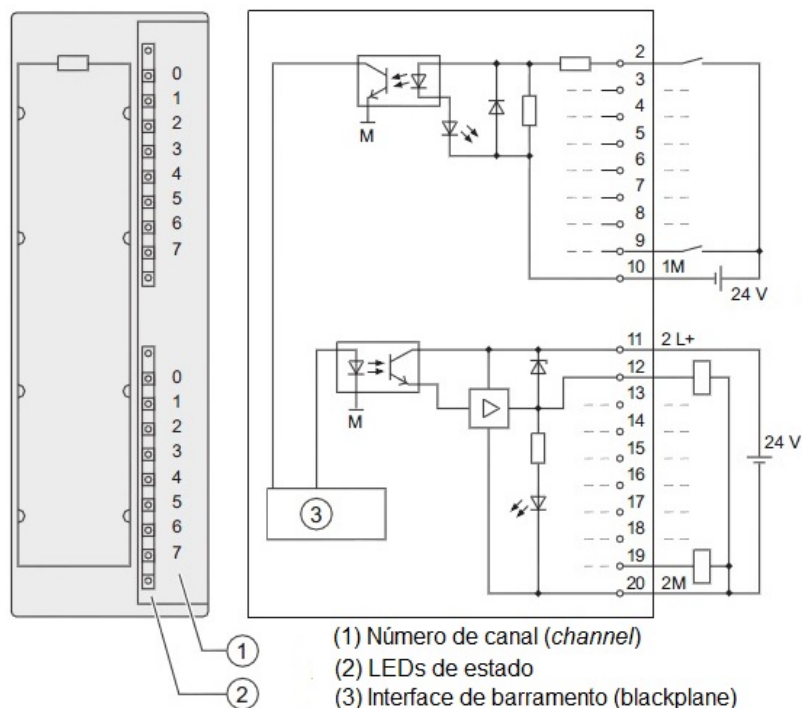


Fig. 23. Ligações e diagrama de bloco – SM 323 DI8/DO8 x DC 24 V/0.5A [13]

3.2.3.5 Módulo de entradas e saídas analógicas

- N.º de entradas: 8 entradas em 4 grupos de canais
 - Características elétricas 24VDC/0.5A
 - Resolução programável para cada grupo de canais:
 - 9 bits + sinal
 - 12 bits + sinal
 - 14 bits + sinal
 - Tipo de medição: Tensão; Corrente; Resistência e Temperatura
 - Isolamento galvânico à CPU
- Alimentação: 24 VDC



Fig. 24. Módulo de entradas analógicas utilizado no PLC S7-300 [Siemens]

Tabela 9. Métodos e faixas de medição do módulo de entradas analógicas [13]

Designação	Método de medição	Faixa de medição
A	Tensão	+/- 1000 mV
B	Tensão	+/- 10 V
C	Corrente, transdutor de 4 fios	4 a 20 mA
D	Corrente, transdutor de 2 fios	4 a 20 mA

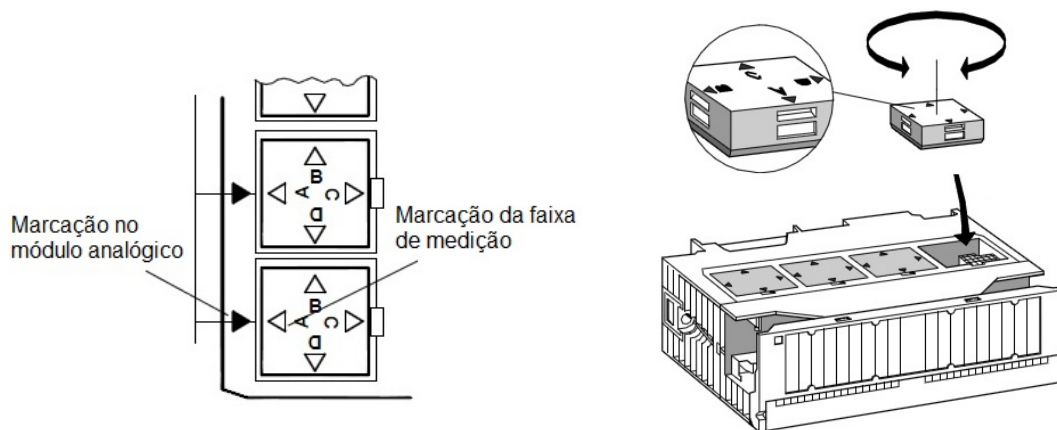


Fig. 25. Marcação dos módulos para diferentes faixas de medição [12]

A fig. 25 ajuda a compreender o funcionamento dos módulos que recebem/enviam sinais analógicos de/para outros elementos do sistema. Contudo, há que ter em atenção que o PLC é um dispositivo que apenas processa dados na forma digital.

Os módulos de entrada analógicos convertem o sinal analógico a processar num sinal digital. Assim, a CPU processa os valores analógicos num formato binário. Os valores analógicos resultam em valores binários fixos de complemento para dois. Para resoluções inferiores as 15 bits, os bits identificados por “x” tornam-se “0” (não utilizados).

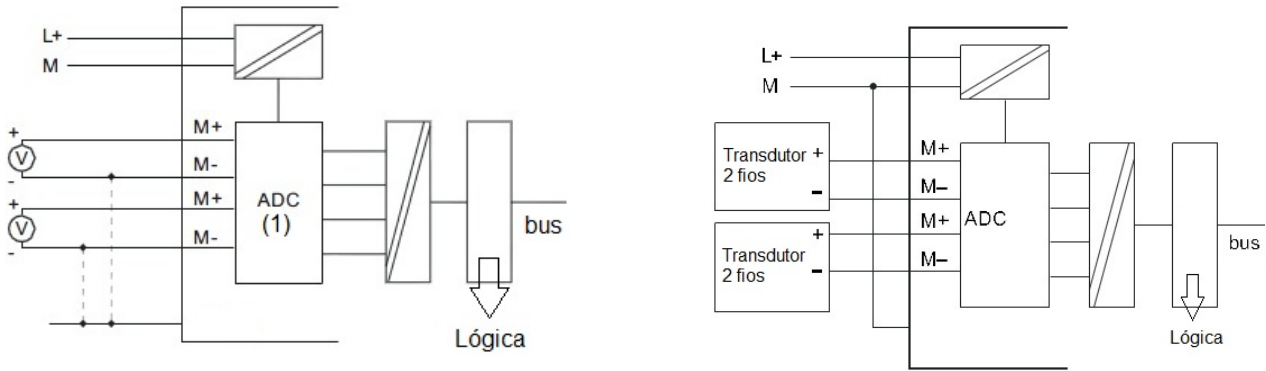


Fig. 26. Esquema de ligações: transdutores de tensão a entradas analógicas eletricamente isoladas (esquerda); transdutores de corrente de dois fios a entradas analógicas eletricamente isoladas (direita) [13]

Tabela 10. Resoluções em módulos AI em PLCs da gama S7-300 [12]

Resolução	Valor analógico															
	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Nº de bits	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
15 bits (+S)	0	1	0	0	0	1	1	0	0	1	1	1	0	0	1	1
12 bits (+S)	0	1	0	0	0	1	1	0	0	1	1	1	0	0	0	0
8 bits (+S)	0	1	0	0	0	1	1	0	0	0	0	0	0	0	0	0

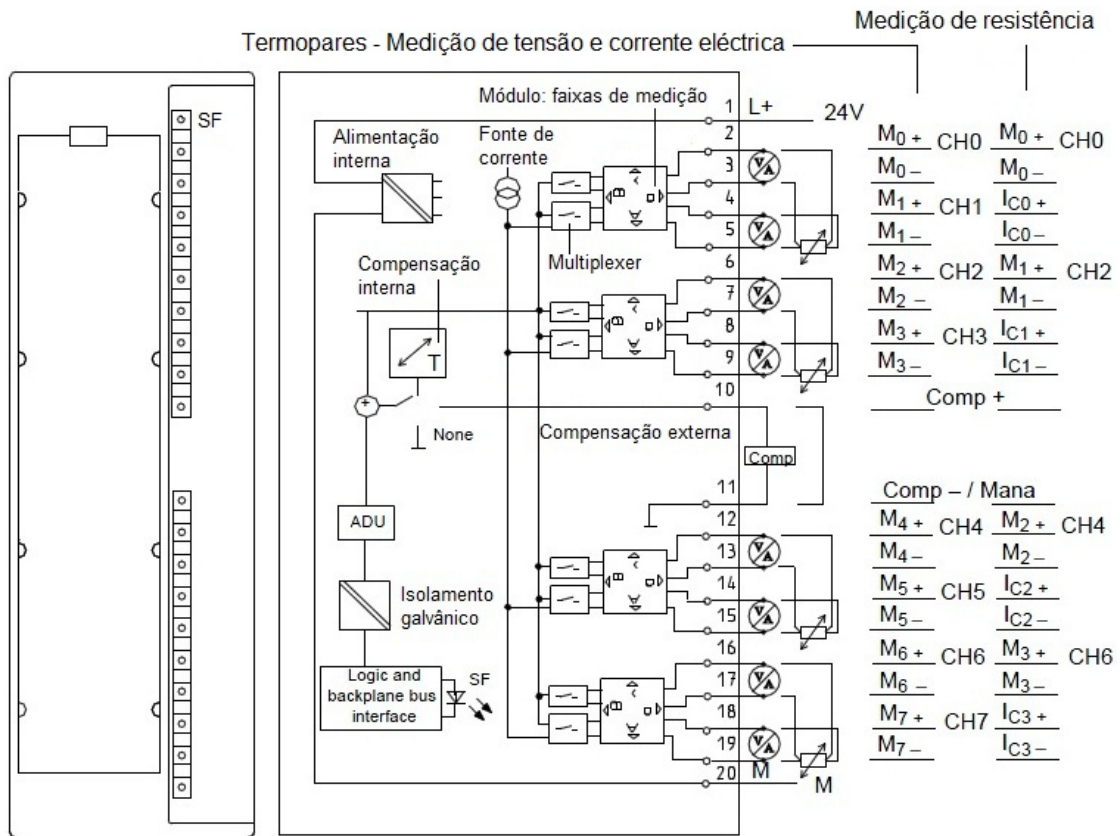


Fig. 27. Diagrama de blocos de um módulo de entradas AI (SM 331 AI8 x12 bit) [13]

Tabela 11. Representação dos valores digitalizados referentes aos valores analógicos medidos [13]

$\pm 5 \text{ V}$	$\pm 10 \text{ V}$ $\pm 10 \text{ mA}$	$\pm 3.2 \text{ mA}$	$\pm 20 \text{ mA}$	Deci- mal	Range
> 5.8794	> 11.7589	> 3.7628	> 23.515	32767	Overflow
5.8794	11.7589	3.7628	23.515	32511	Ovrange
⋮	⋮	⋮	⋮	⋮	
5.0002	10.0004	3.2001	20.0007	27649	
5.00	10.00	3.200	20.000	27648	Nominal range Faixa de medição nominal
3.75	7.50	2.400	14.998	20736	
⋮	⋮	⋮	⋮	⋮	
-3.75	-7.50	-2.400	-14.998	-20736	
-5.00	-10.00	-3.200	-20.000	-27648	
-5.0002	-10.0004	-3.2001	-20.0007	-27649	Underrange
⋮	⋮	⋮	⋮	⋮	
-5.8796	-11.759	-3.7629	-23.516	-32512	
<- 5.8796	<- 11.759	<- 3.7629	<- 23.516	-32768	Underflow

3.2.3.6 Módulo funcional de contagem FM350-1

Aplicações:

- Contagem de objetos
- Medição do período de rotação
- Medição da frequência de rotação
- Medição de velocidade angular
- Determinação de posicionamento angular

Outras ligações no painel frontal:

- Sinais de saída digitais (Q0 e Q1)
- Tensão elétrica auxiliar (1L+) para alimentação de encoders
- Tensão elétrica auxiliar (2L +) para disponibilização das saídas digitais

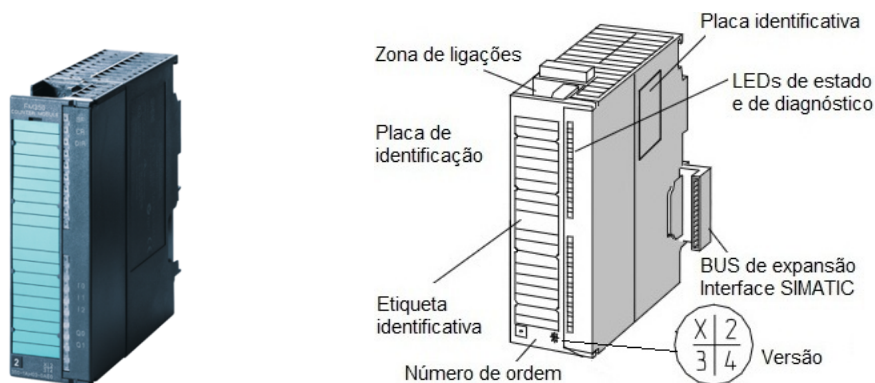


Fig. 28. Esquema com sinalética identificativa do módulo FM 350-1 (direita) [14]

Tabela 12. Sinalização do painel frontal do FM 350-1 [14]

Etiqueta	Cor	Função
SF	Vermelho	Erro de grupo / sistema
CR	Verde	Em contagem; estado do bit menos significativo do contador
DIR	Verde	Direção da contagem; LED ON caso seja no sentido decrescente (baixo)
I0	Verde	Estado da entrada digital Start
I1	Verde	Estado da entrada digital Stop
I2	Verde	Estado da entrada digital Set
Q0	Verde	Estado da saída digital Q0
Q1	Verde	Estado da saída digital Q1

Tipos de encoders suportados pelo módulo FM 350-1:

Tabela 13. Tipos de encoders e sinais suportados pelo contador [14]

Encoder	Sinais
Encoder incremental de 5V	Diferenciais (A, /A; B, /B; N, /N)
Encoder incremental de 24V	A*, B* e N*
Encoder de pulso de 24V	Sinal (com direção) de 24V
Sensor de proximidade de 24V	Sinal (sem direção) de 24V

Conectores	Sinais dos encoders
A	5 V (diferenciais)
D	24 V

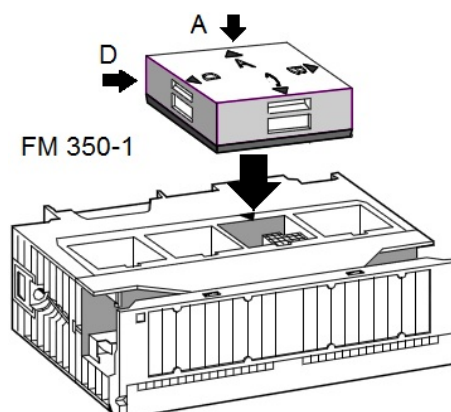
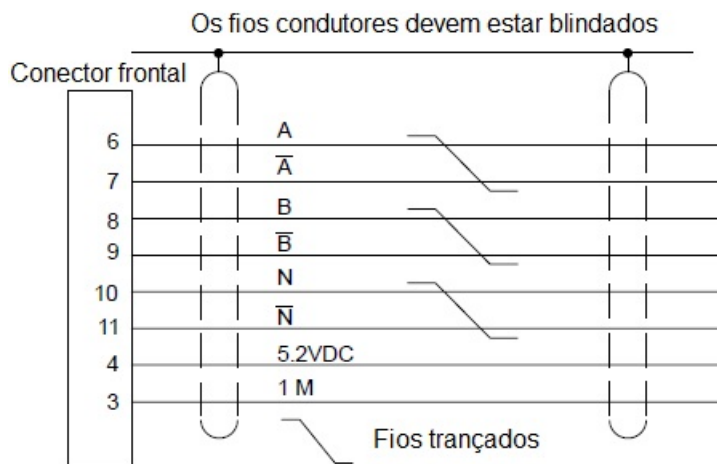


Fig. 29. Modo de funcionamento do módulo AI, de acordo com o posicionamento do conector (A/D) [14]

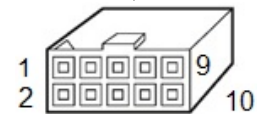
Características principais dos sensores de ângulo (incrementais) (eixo 1 e eixo 2):

Através da digitalização de um código de barras, os encoders incrementais geram pulsos elétricos retangulares. A distinção observa-se na amplitude dos impulsos e do número de sinais.

- Tensão de funcionamento: 4.5 VDC a 5.5 VDC
- Largura de pulso: 180 +/- 35 graus
- Mudança de fase: (canal A para B) 90 +/- 15 graus
- Sinais de saída: 2 ondas quadradas + 1 impulso nulo
- Frequência de corte: 100 kHz
- Pulsos por revolução: 500 (eixo 1); 400 (eixo 2)



Conector (PIN-OUT)



N.º	Cor	Parâmetro
1	Vermelho	NC
2	Cinzento	Vcc (+5V)
3	Cinzento	GND
4	Cinzento	NC
5	Cinzento	NOT A
6	Cinzento	A
7	Cinzento	NOT B
8	Cinzento	B
9	Cinzento	NOT I
Pin 10 = I (INDEX)		

Fig. 30. Esquema de ligações / PIN-OUT dos sensores de ângulo utilizados [14]

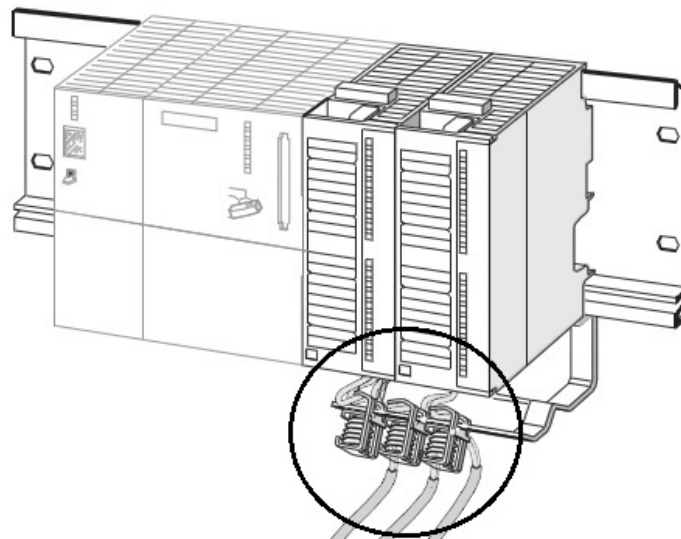
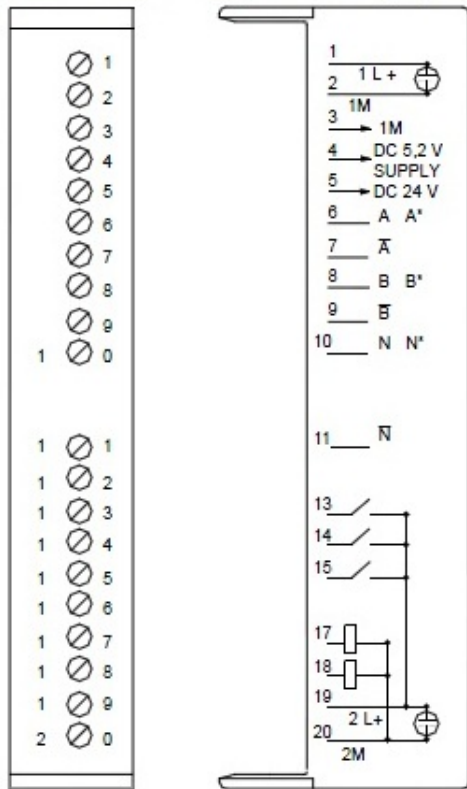


Fig. 31. Colocação dos módulos contadores numa rack com CPU da gama S7-300. Notar na conexão blindada dos fios de ligação dos encoders aos módulos FM 350-1 [14]

Conector frontal Esquema de ligações



Pino	Nome	Tipo	Função	Pino	Encoder
1	1L+	Input	24V - auxiliar		
2	1M	Input	GND - auxiliar	3	GND
3	1M	Output	GND encoder		
4	5.2VDC	Output	+ V encoder	2	Vcc(+5V)
5	24VDC	Output	+ V encoder		
6	AA*	Input	Sinal A encoder	6	A
7	NOT A	Input	Sinal A invertido	5	NOT A
8	BB*	Input	Sinal B encoder	8	B
9	NOT B	Input	Sinal B invertido	7	NOT B
10	NN*	Input	Sinal N encoder	10	I
11	NOT N	Input	Sinal N invertido	9	NOT I
12	Não utilizado				
13	I0	Input	Start – digital		
14	I1	Input	Stop – digital		
15	I2	Input	Set – digital		
16	Não utilizado				
17	Q0	Output	Saída 0 – digital		
18	Q1	Output	Saída 1 – digital		
19	2L+	Input	24V - cargas		
20	2M	Input	GND (I/Os)		

Fig. 32. Terminais do conector frontal de 20 pinos do módulo FM 350-1 (impulsos dos encoders; entradas e saídas digitais; tensão de alimentação dos encoders; tensão de alimentação auxiliar); PIN-OUT dos encoders [14]

Características do módulo funcional FM 350-1

O módulo funcional FM 350-1 é um módulo contador de impulsos de alta velocidade para uso num controlador lógico programável (S7-300 / M7-300), que pode operar nas gamas: $(0 \text{ a } 2^{32}-1)$; $(-2^{31} \text{ a } 2^{31})$.

A frequência máxima dos sinais do contador é 500 kHz, dependendo do sinal do encoder.

É possível utilizar o módulo FM 350-1 para contagem contínua, individual ou periódica.

A contagem pode ser iniciada ou terminada por via de um programa de utilizador (*software gate*) ou por via de sinais externos (*hardware gate*).

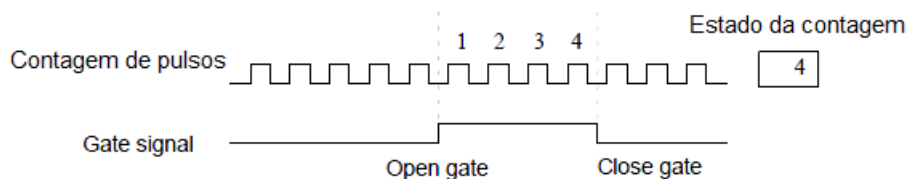
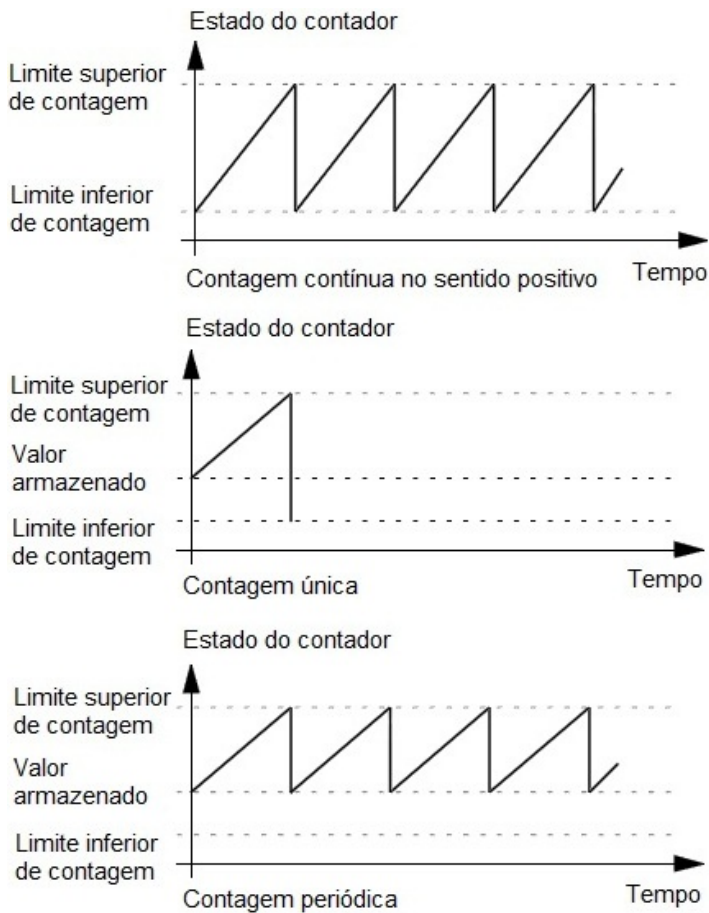


Fig. 33. Abertura e fecho de porta para contagem de pulsos [14]

Descrição dos tipos de contagem – FM 350-1



Contagem contínua:

Quando o contador alcança o valor limite superior e surge outro pulso, há um salto para o valor limite inferior do contador e os pulsos seguintes adicionam-se sobre valor, repetindo-se consecutivamente o processo de contagem.

Contagem única / singular / individual:

Neste método a contagem inicia-se a partir do valor armazenado. Quando o contador alcança o valor limite superior (contagem positiva) e surge outro pulso, há um salto para o valor limite inferior, permanecendo o contador neste valor mesmo que surjam outros pulsos.

Se a contagem for para baixo (negativa), quando o contador alcança o valor limite inferior e surge outro pulso, ocorre um salto para o valor limite superior, permanecendo o contador neste valor mesmo se existir a receção de novos pulsos.

Contagem periódica:

Neste método a contagem inicia-se a partir do valor armazenado. Quando o contador atinge o valor limite superior de contagem e surge outro pulso, há um salto para o valor armazenado no início da contagem, iniciando-se um novo processo de contagem com o aparecimento de novos pulsos.

Fig. 34. Distinção entre os diferentes métodos de contagem do módulo FM 350-1 [14]

Encoder incremental de sinais diferenciais de 5V

Os sinais /A (NOT A), /B (NOT B) e /N (NOT N) são sinais invertidos dos sinais A, B e N.

Os sinais A e B têm uma diferença de fase de 90°.

Os sinais A e B recebidos são usados para contagem e os sinais N para iniciar a contagem a partir de um valor previamente carregado/definido.

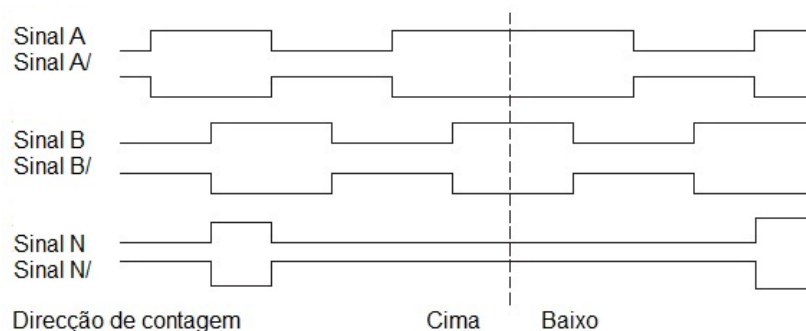


Fig. 35. Diagrama temporal de sinais de um encoder incremental de 5V
O conector do módulo tem de estar colocado na posição A (ver tabela 13) [14]

Avaliação de sinais – FM 350-1

O módulo tem a capacidade de contar os flancos dos sinais que recebem. Por norma, avalia o flanco do sinal A (avaliação singular) mas note que existem outras opções por forma a aumentar a resolução de leitura: avaliação dupla e quadrupla. A avaliação dupla só é possível quando o encoder fornece sinais A e B com diferença de fase de 90°.

No modo de avaliação singular, o módulo avalia apenas o flanco do sinal A. Conta de forma ascendente, quando ocorre um flanco positivo do sinal A (high) e o não exista sinal B (low). Conta de forma descendente se ocorrer um flanco negativo do sinal A e não exista sinal B.

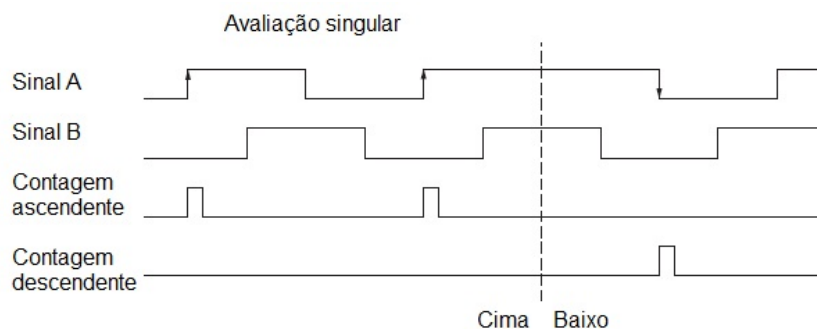


Fig. 36. Diagrama temporal exemplificativo de avaliação singular [14]

Condições de funcionamento iniciais – FM 350-1:

Após os módulos estarem devidamente conectados ao PLC, colocando-o em funcionamento, antes de efetuar qualquer transferência de parametrização, as condições são as seguintes:

- Porta de contagem de pulsos (gate) – aberta;
- Contador de impulsos definido para sinais diferenciais de 5V;
- Sinal B não invertido;
- Avaliação de sinal singular;
- Modo de contagem contínua;
- Faixa de medição (0 a +32 bit);
- Valor inicial do contador a 0;
- Contagem com sinais digitais bloqueada;
- Outputs bloqueados.

3.2.3.7 Motores passo-a-passo

O movimento dos eixos (1 e 2) é garantido por dois motores passo-a-passo. Este tipo de motor consiste num actuador eletromecânico que converte impulsos elétricos de comando em deslocamentos precisos tipicamente angulares, designados por passos. A cada impulso o eixo roda um determinado ângulo, que depende do passo do motor, e permanece estável nessa posição se mais nenhum impulso for aplicado. Ao manter uma série de impulsos, o eixo do motor roda por saltos contínuos com uma velocidade proporcional à frequência dos impulsos [16].

Tabela 14. Características técnicas principais dos motores passo-a-passo [16]

Motor (eixo 1)	Características Técnicas	Motor (eixo 2)
7.4 VDC	Tensão Nominal	8.8 VDC
0.80 A/Enrolamento	Corrente Nominal	1.0 A/Enrolamento
9.30 Ohm/Enrolamento	Resistência/Enrolamento	8.80 Ohm/Enrolamento
17.0 mH/Enrolamento	Indutância/Enrolamento	19.0 mH/Enrolamento
0.5 N.m	Binário (detenção)	1.2 N.m
1.37×10^{-2} N.m	Binário (retenção)	0.068 N.m
1.8	Passo (ângulo)	1.8
0.82 g.cm ²	Inércia (binário rotor)	480 g.cm ²
0.34 kg	Peso	1.0 kg
49 mm	Comprimento	76 mm
Bipolar	Modo de alimentação	Bipolar
4	Contactos	8



Fig. 37. Fotografia motor do eixo 1 (esquerda) e do motor do eixo 2 (direita) [17]

Tabela 15. Descrição das principais características dos motores BH PaP [16], [17]

Tensão Nominal	Tensão de funcionamento do motor
Corrente Nominal	Corrente que os enrolamentos do motor podem suportar sem problemas
Resistência do Enrolamento	Valor ôhmico que o enrolamento possui
Indutância do Enrolamento	Valor da indutância do enrolamento
Binário	A força de rotação do eixo do motor que depende: da frequência dos passos, da corrente de excitação dos enrolamentos e do tipo de excitação
Passo	Mais pequeno deslocamento angular realizado pelo eixo do motor
Ângulo de Passo	O ângulo de passos obtém-se dividindo 360° pelo número de passos
Binário de Detenção	Máxima força que é possível aplicar ao eixo de um motor alimentado sem provocar a sua rotação
Binário de Retenção	Máxima força que é possível aplicar ao eixo de um motor não alimentado sem provocar a sua rotação
Inércia do Rotor	Valor que indica a maior ou menor dificuldade de o rotor iniciar a sua rotação
Fases	Os enrolamentos eletricamente separados que dão origem a fluxos magnéticos.

O motor passo-a-passo do tipo híbrido (HB PaP) combina os princípios do motor de ímã permanente e do motor de relutância variável. O ângulo de passo é tipicamente pequeno (1.8°) e normalmente apresentam um binário elevado a baixa rotação, característica importante para aplicações de posicionamento.

O motor HB PaP possui tipicamente dois enrolamentos com várias bobinas e pólos magnéticos com dentes. Estes pólos estão dispostos de frente dos dentes das massas polares do rotor. Por apresentar um rotor magnético, distingue-se do motor de relutância variável, e por apresentar dentes no estator e no rotor, distingue-se do motor de ímã permanente [16].

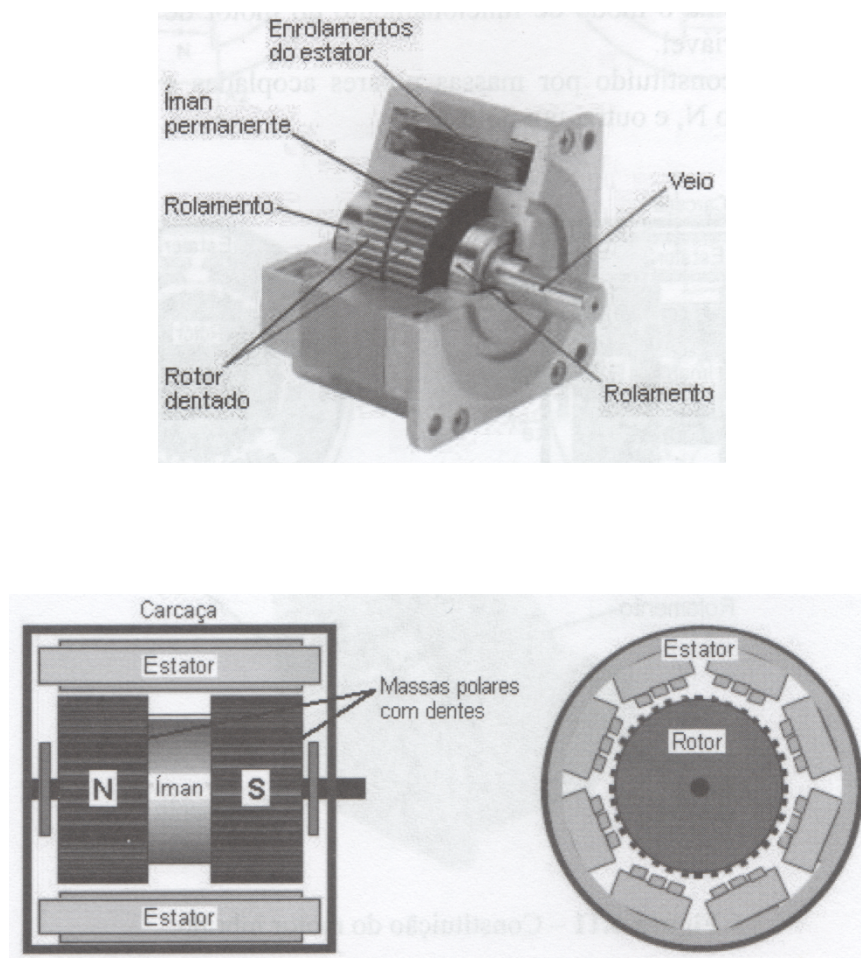


Fig. 38. Elementos constituintes do motor HB PaP. Vista em corte longitudinal (esquerda) e transversal (direita) [16]

Modos de alimentação de motores PaP

Alimentação Unipolar

Tipo de alimentação em que a corrente de excitação circula sempre no mesmo sentido nos enrolamentos do estator. Dentro deste género, as mais comuns são as de cinco e as de seis condutores. São formadas por dois enrolamentos iguais com ponto médio comum (cinco condutores) ou separado (seis condutores).

Alimentação Bipolar

Neste modo de alimentação o estator do motor possui dois enrolamentos iguais, sendo que a corrente de excitação os percorre ora num sentido, ora no sentido contrário. Desta forma, a corrente cria um campo magnético ora com uma polaridade, ora com a polaridade inversa. O motor com alimentação bipolar é de construção mais simples, que o motor unipolar, e, para o mesmo tamanho, possui um binário mais elevado.

Passo Completo (Full Step)

Neste modo de funcionamento, em cada passo, duas fases são alimentadas em simultâneo. Num motor bipolar, os dois enrolamentos recebem corrente em simultâneo. Uma sequência completa de impulsos tem quatro passos. Uma volta completa é realizada em quatro passos de 90° cada. Por passo, os enrolamentos são utilizados em 100%. O rotor roda em sentido anti-horário e, mantendo-se a sequência aplicada ao motor, este roda continuamente. Para inverter o sentido de rotação, inverte-se a ordem da sequência.

Meio Passo (Half Step Mode)

Neste modo de funcionamento, em passos consecutivos, é alimentada uma fase e de seguida duas. Com este tipo de alimentação, o rotor executa deslocamentos de meios passos. Com passos de 45°, uma sequência completa de impulsos tem oito passos.

Esta modalidade de funcionamento fornece um binário ligeiramente menor que o funcionamento em passo completo, mas o funcionamento do motor é mais suave, uma vez que a velocidade se reduz a metade.

Com este tipo de alimentação consegue-se um movimento do rotor a metade do passo real. Um motor de 1.8° alimentado em modo de meio passo, em vez dos 200 passos por volta, tem 400 passos, sendo o deslocamento de 0.9° [16].

Considerações gerais sobre os motores passo-a-passo

- Um motor parado e alimentado durante muito tempo pode aquecer demasiado;
- Os motores PaP são dispositivos mecânicos e, para rodarem, têm de vencer a inércia, portanto, os impulsos de comando têm de ser aplicados a uma determinada frequência;
- Em velocidades de funcionamento elevadas, para se conseguir um arranque suave, é recomendável iniciar o movimento com uma frequência de impulso baixa e ir aumentando gradualmente até à velocidade desejada, sem ultrapassar a máxima permitida. Também se deve baixar previamente a velocidade, antes de se inverter o sentido de rotação;
- Os motores têm um tempo mínimo entre passos, este tempo depende da velocidade e do binário necessário para movimentar a carga aplicada ao veio. Se o comando do motor lhe aplicar uma velocidade superior, considerando a carga que o veio possui, pode acontecer que o eixo vibre ou reaja de forma errática [16].

Vantagens dos motores PaP

- Construção simples (baixo custo);
- Fácil manutenção (não possuem escovas) e alta fiabilidade;

- Tamanho reduzido;
- Fácil controlo (comando digital);
- Binário elevado em velocidades baixas;
- O veio pode estar fixo numa determinada posição sem necessidade de freio;
- Pode rodar de forma contínua e com velocidade variável [16].

Desvantagens dos motores PaP

- Fenómenos de ressonância podem ocorrer a baixa velocidade;
- Binário diminui com o aumento da velocidade;
- A velocidade elevada pode perder o sincronismo com o sistema de controlo [16].

3.2.3.8 Drivers – Cartas de controlo dos motores PaP

As seqüências de comando dos motores PaP alcançam-se recorrendo à utilização de circuitos integrados, microcontroladores ou PLCs, dependendo a simplicidade da aplicação. Existem mesmo módulos funcionais para PLCs, cuja finalidade é possibilitar multiconfigurações de formas de comando para vários tipos de motores PaP. As cartas de controlo de motores PaP (drives) têm por finalidade alimentar os enrolamentos com as correntes necessárias ao funcionamento do motor [16].

Tabela 16. Características técnicas das cartas de comando dos motores PaP [17]

Características (drives)	Dados técnicos
Tensão de operação	24 VDC até 48 VDC
Máx. corrente por fase	0.4 – 2 A/fase
Modos de operação	Completo (1/1); 1/2; 1/4; 1/5; 1/8; 1/10
Frequência de step	0 a 50 kHz
Sinais de entrada	Via optoacoplador 5 V (24 V)

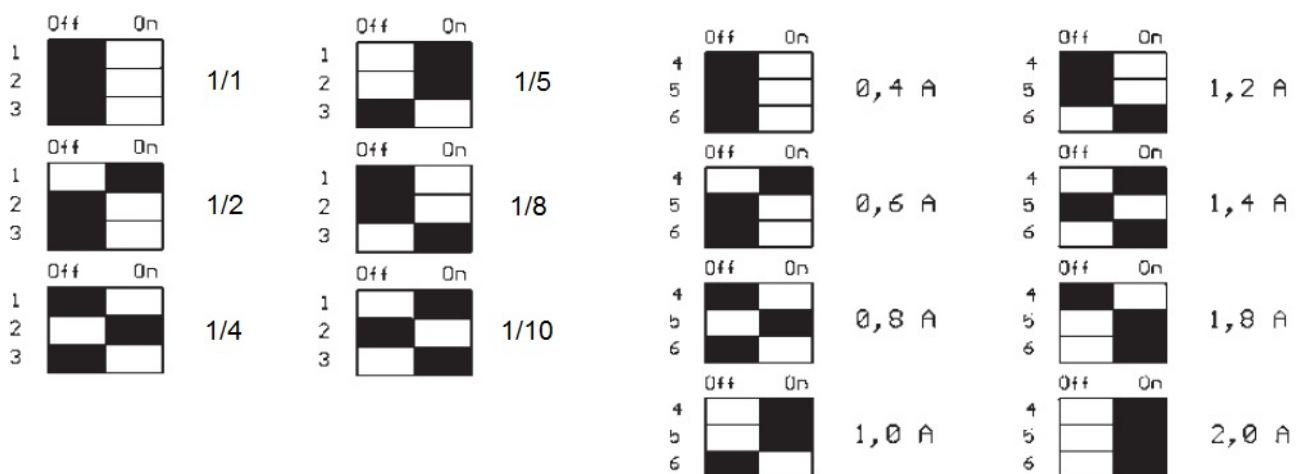


Fig. 39. Configuração do número de passos (esquerda) e configuração da corrente [17]

3.2.3.8 Módulo Funcional FM 353 para comando de motores PaP

Descrição geral das funções do módulo FM 353

Modos de controlo disponíveis

- “*Jogging*”;
- “*Open-loop control*”;
- “*Reference point approach*”;
- “*Incremental mode, relative*”;
- “*MDI-Manual Data Input*”;
- “*Automatic*”;
- “*Automatic single block*”.

Determinação de posição

A frequência dos pulsos emitidos pelo módulo FM 353 é utilizada internamente para determinar um valor de posição angular do actuador.

Controlo de motor PaP

- Comando do acionamento de acordo com a velocidade, durante a sequência de movimentos, por meio de operações ajustáveis de *start*, *stop*, aceleração e atraso;
- Saída de pulsos de passos por meio de controlo de frequência;
- Saída de sinais direcionais;
- Posicionamento preciso, através da programação da posição alvo;
- Disponibilização de sinais de comando para driver de controlo de corrente nos enrolamentos do motor PaP.

Entradas e saídas digitais

Estão disponíveis quatro entradas digitais e quatro saídas digitais para conexão:

- Interruptores de ponto de referência e inicialização externa;
- Sensores de posição.

A função é atribuída ao dispositivo com um determinado endereço de I/O através do STEP 7.

Interrupção de processo

A interrupção de acionamento surge com os seguintes acontecimentos:

- Alcance de posição ou medição de distância concluída;
- Alteração programada num bloco em execução.

A função de interrupção é ativada por meio de *software*, através do STEP 7.

Blocos de sequência de comandos e armazenamento de dados

Funcionamento automático através da leitura de dados programados em blocos para o efeito. Os dados da parametrização são transferidos e ficam armazenados no próprio módulo [18].

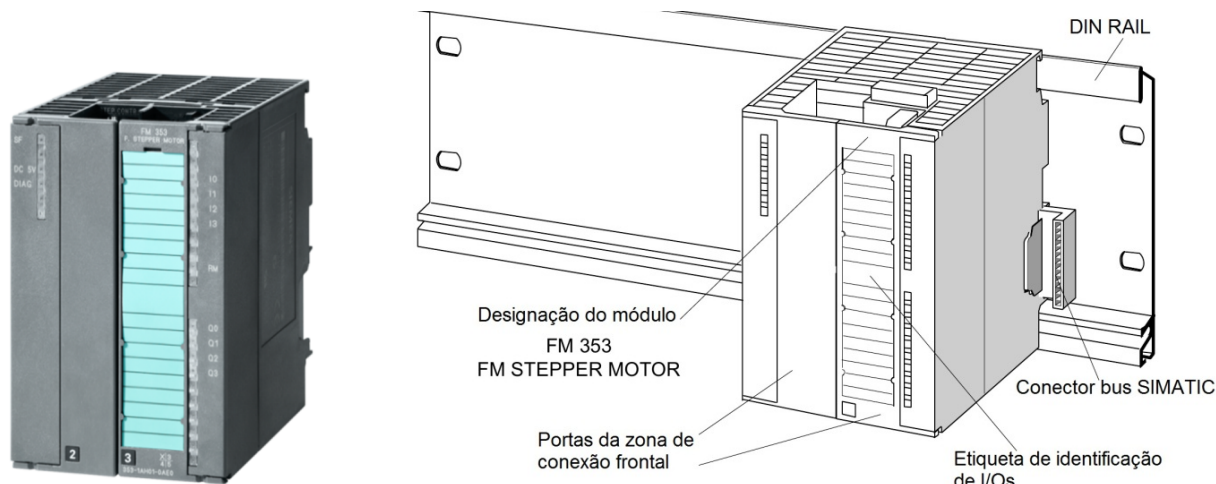


Fig. 40. Fotografia (esquerda) e sinalética exterior do módulo FM 353 (direita) [18]

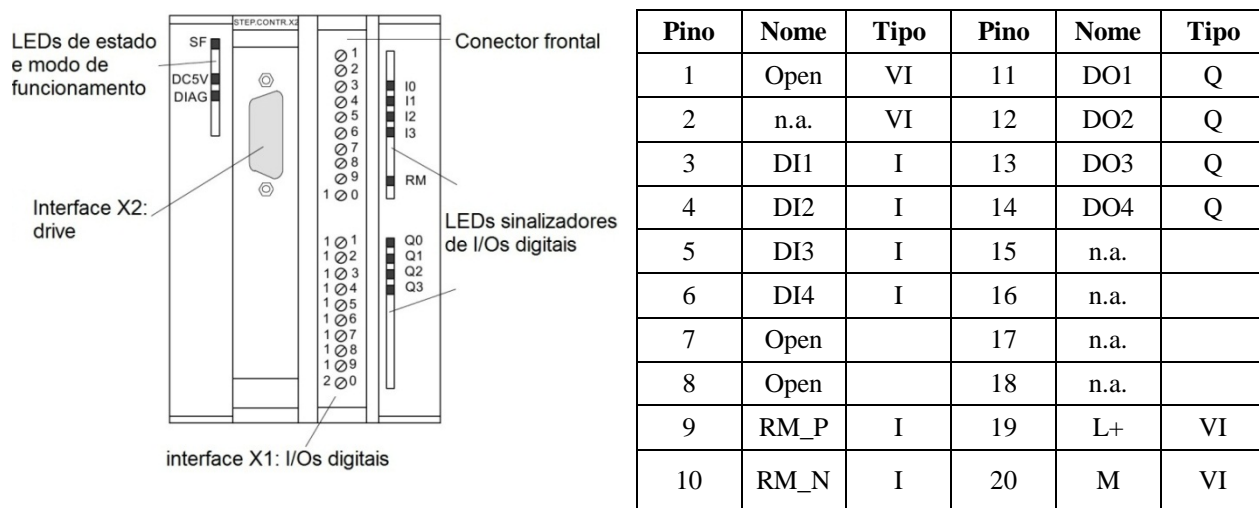


Fig. 41. Sinalética: interfaces e conector frontal [17]. Atribuição dos terminais do conector frontal de 20 pinos do módulo FM 353. I: Entrada; Q: Saída; VI: Entrada de tensão; DI1 – 4: Entrada digital 1 – 4; DQ1 – 4: Saída digital 1 – 4; RM_P: Entrada positiva de sinal de controlo; RM_N: Entrada negativa de sinal de controlo; L+: alimentação de 24 V; M: terra

Descrição dos sinais de saída do módulo FM 353

- **Pulso (*pulse*):** Com pulsos de relógio controla-se o motor PaP. O motor executa um passo em resposta a cada flanco de pulso positivo. Isso significa que o número de pulsos de saída determina o ângulo de rotação e a frequência dos pulsos determina a velocidade de rotação.
- **Direção (*direction*):** O nível do sinal de saída determina a direção da rotação que se pretende dar ao motor; sinal “0” para rotação no sentido horário e sinal “1” para rotação no sentido anti-horário.
- **Enable:** O módulo FM 353 ativa este sinal sempre que se deteta ativo o modo de controlo cíclico; sinal ON: ativação energética disponível; em sinal OFF: pode desativar entrada de pulso; cortar a alimentação do motor; o reset ao contador e apaga mensagens de erro [18].

Tabela 17. Ligações do módulo FM 353 com o drive SMC32 (motor 1) [18]

Porta X2 (Pino)	Nome	Tipo	Drive SMC32 (Motor 1)	
1	PULSE	Output	Vermelho	CLK+5
2	DIR	Output	Branco	DIR+5
3	ENABLE	Output	Verde	EN+5
4	PWM/BOOST	Output		
5	GND	Output		
6	GND	Output		
8	GND	Output	Azul	CLK-
9	PULSE_N	Output	Preto	DIR-
10	DIR_N	Output	Roxo	EN-
11	ENABLE_N	Output		
12	PWM_N/BOOST_N	Output		
13	GND	Output		
14	GND	Output		
15	READY1_N	Input		

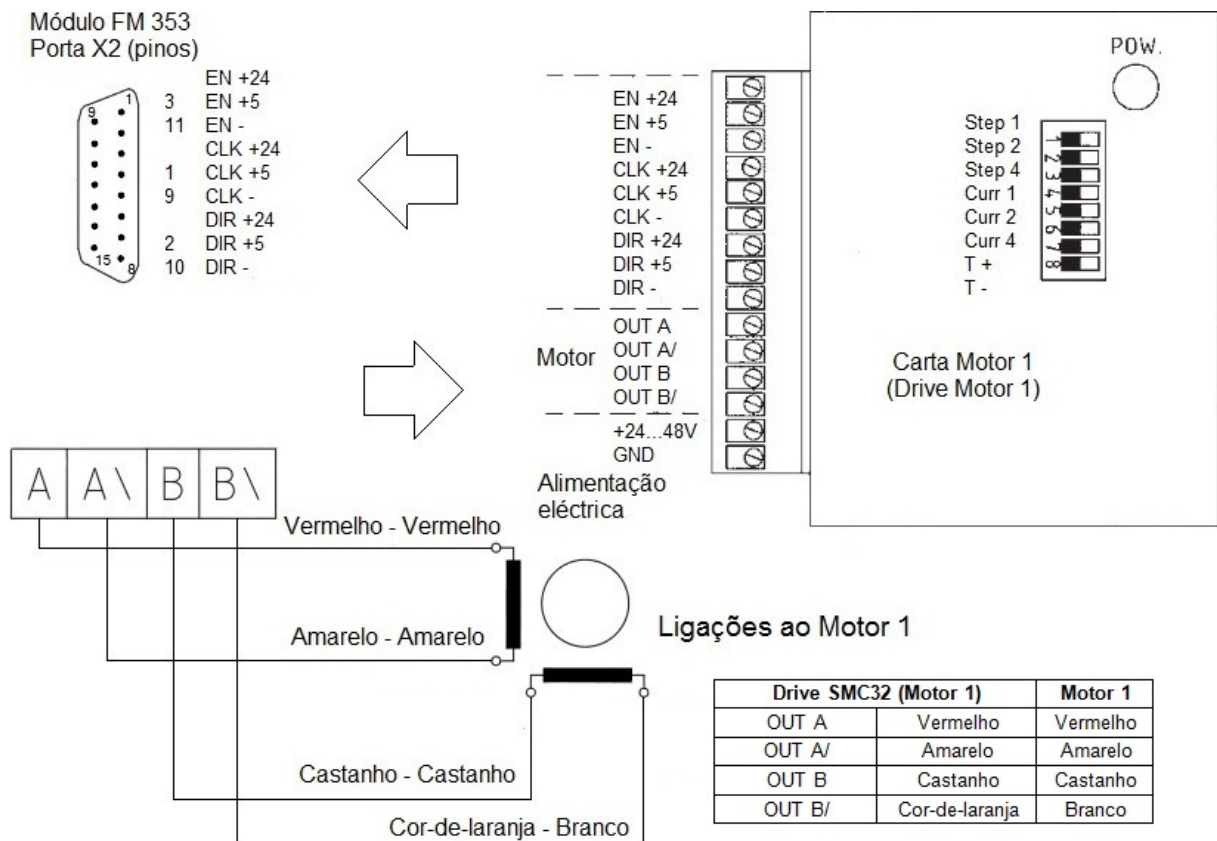


Fig. 42. Ligações entre os módulos FM 353, os drives SMC32 e os motores passo-a-passo (eixo 1)

Tabela 18. Ligações do módulo FM 353 com o drive SMC32 (motor 2) [18]

Porta X2 (Pino)	Nome	Tipo	Drive SMC32 (Motor 1)	
1	PULSE	Output	Vermelho	CLK+5
2	DIR	Output	Branco	DIR+5
3	ENABLE	Output	Verde	EN+5
4	PWM/BOOST	Output		
5	GND	Output		
6	GND	Output		
8	GND	Output	Azul	CLK-
9	PULSE_N	Output	Preto	DIR-
10	DIR_N	Output	Roxo	EN-
11	ENABLE_N	Output		
12	PWM_N/BOOST_N	Output		
13	GND	Output		
14	GND	Output		
15	READY1_N	Input		

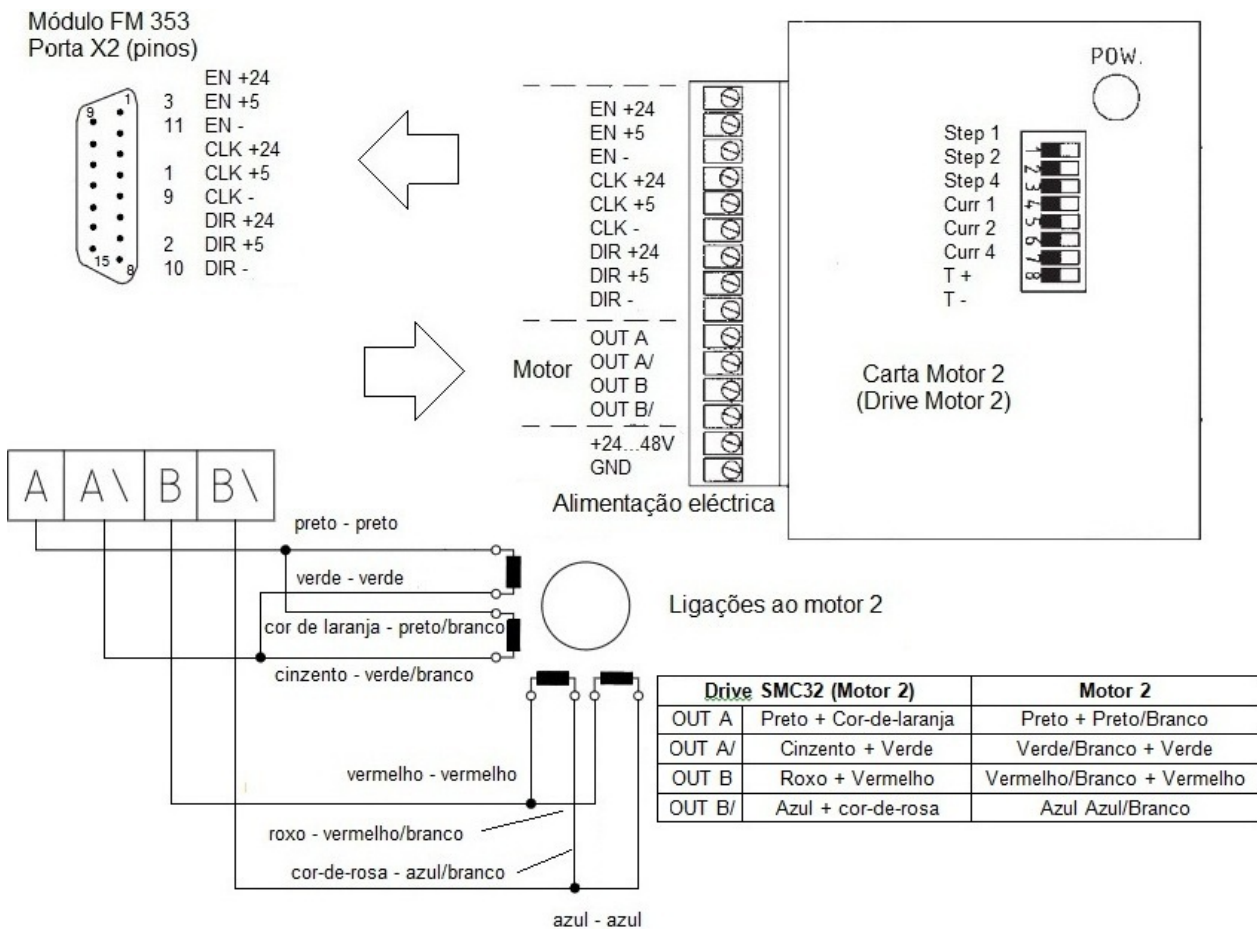


Fig. 43. Ligações entre os módulos FM 353, os drives SMC32 e os motores passo-a-passo (eixo 2)

Tabela 19. Parâmetros elétricos dos sinais de saída do módulo FM 353 para o driver [18]

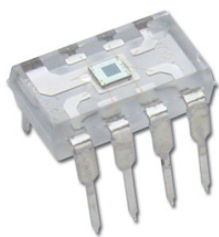
Parâmetros	min	máx	Unidade	Quando
Tensão de saída diferencial V_{OD}	2		V	$R_L = 100 \text{ Ohm}$
Tensão de saída "1" V_{OH}	3.7		V	$I_o = -20 \text{ mA}$
	4.5		V	$I_o = -100 \text{ micro A}$
Tensão de saída "0" V_{OL}		1	V	$I_o = 20 \text{ mA}$
Resistência de carga R_L	55		Ohm	
Corrente de saída I_o		+/- 60	mA	
Frequência de pulso f_p		200	kHz	

3.3 Descrição técnica do equipamento – Subsistema secundário

3.3.1 Elemento sensor

O componente base do sistema secundário é um sensor que consiste basicamente num conversor de irradiância-frequência. Os produtores de transdutores e sensores desenvolvem cada vez mais produtos que têm como vantagem a utilização de frequência como parâmetro informativo. O sensor utilizado combina um fotodíodo configurável com um conversor corrente-frequência, num único circuito integrado. Como saída podemos extrair um trem de impulsos ou, como foi implementado no projeto, uma onda quadrada, com 50% de ciclo de trabalho (*duty cycle*), cuja frequência é proporcional à irradiância. É possível trabalhar, sem a necessidade de implementar filtros, com níveis de irradiância compreendidos entre os $0.001 \mu\text{W}/\text{m}^2$ e os $100000 \mu\text{W}/\text{m}^2$. Este tipo de sensor pode ser conectado diretamente a um microcontrolador.

Tabela 20. Características do sensor (TSL230R-LF) [19]



Característica	TSL230R-LF
Max. Frequência de saída	1.0 MHz
Resposta espectral	350 – 1000 nm
Erro de não-linearidade	0.2 % a 100 kHz
Tolerância absoluta (saída em frequência)	+/- 20%
Programável	Sim

Fig. 44. Imagem de um circuito integrado TSL230R-LF [19]

A frequência à saída, no caso do sensor utilizado, é dada pela expressão:

$$fO = fD + Re \cdot Ee, \quad (1)$$

onde fO é a frequência de saída; fD é a frequência resultante de corrente elétrica de fuga (na ausência de luz); Re é a responsividade para radiação com um determinado comprimento de onda, apresentada em $\text{kHz}/(\text{mW}/\text{cm}^2)$; Ee é a irradiação incidente em mW/cm^2 .

A partir da expressão (1) compreende-se que, no caso de baixo nível de irradiância, o valor total da frequência de saída tem uma componente com peso significativo, que provem da frequência resultante da corrente de fuga (em ausência de luz), valor esse que depende da temperatura.

A sensibilidade do dispositivo seleciona-se em três níveis (baixa, normal e alta).

Todas as entradas e saídas do circuito integrado são compatíveis com arquitetura TTL e possui um ativador de saída (*enable output*) que coloca a saída num estado de alta impedância, sendo possível colocar o sensor a comunicar diretamente com um microcontrolador para programação e criação de interface de saída, [19], [20].

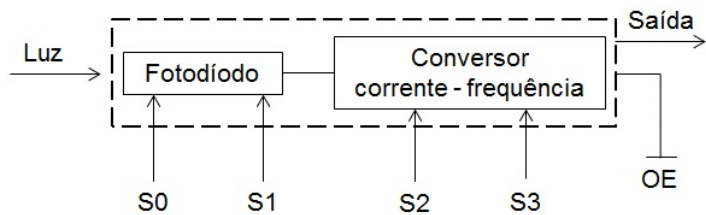
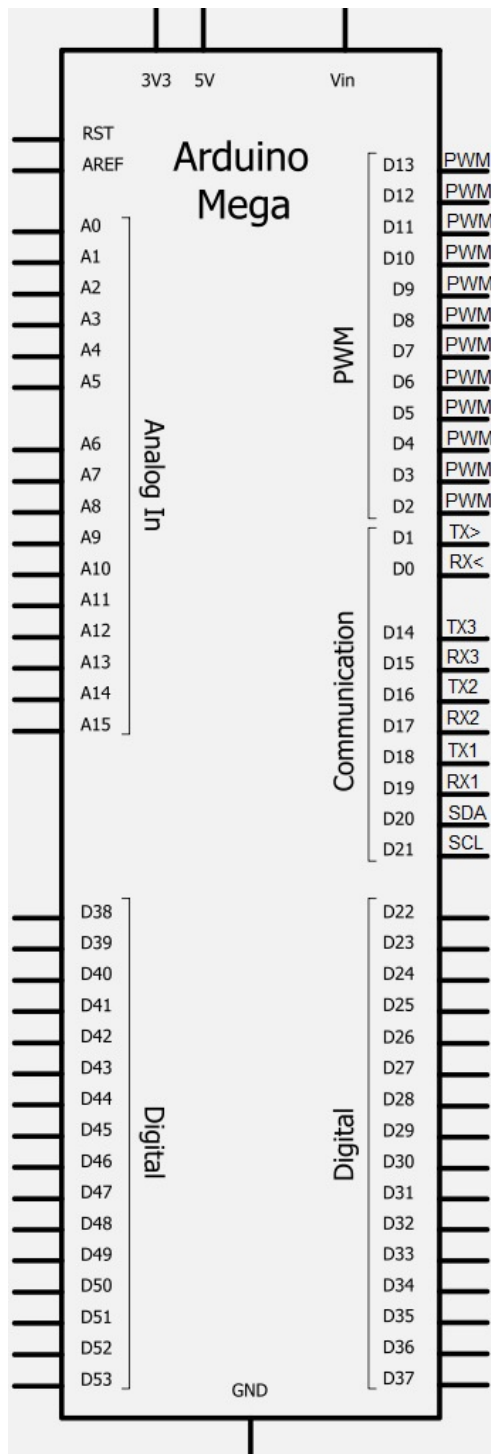


Fig. 45. Diagrama de blocos do circuito integrado TSL230R-LF [19]

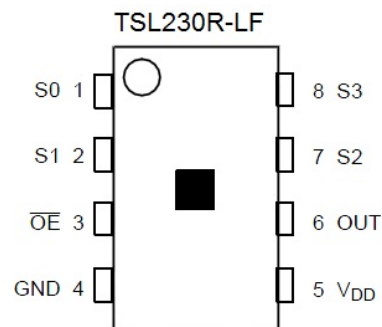


Fig. 46. Representação das portas do circuito integrado (cima) e do microcontrolador (esquerda), adaptado de [19], [20]

Tabela 21. Apresentação das portas do circuito integrado e correspondência com as ligações ao microcontrolador [19], [20]

Sensor 1		Sensor 2		Sensor 3		Sensor 4	
S0	D22	S0	D30	S0	D34	S0	D38
S1	D24	S1	D31	S1	D35	S1	D39
OE	GND	OE	GND	OE	GND	OE	GND
GND	GND	GND	GND	GND	GND	GND	GND
S2	D26	S2	D32	S2	D36	S2	D40
S3	D28	S3	D33	S3	D37	S3	D41
OUT	D2	OUT	D3	OUT	D18	OUT	D19
V _{DD}	5 V	V _{DD}	5 V	V _{DD}	5 V	V _{DD}	5 V

Tabela 22. Algumas considerações importantes relativas à montagem do sistema secundário (TSL230R-LF) [19], [20]

Alimentação elétrica	Conexão de um condensador (0.01 mF; 0.1mF) entre V_{DD} e GND para diminuir o ruído proveniente da alimentação elétrica
Interface de entrada	Deve existir uma ligação de baixa impedância entre o pino OE e o GND para garantir mais imunidade ao ruído.
Interface de saída	Garantir que os fios conectados à saída não possuem mais do que 30 cm, para não haver necessidade de usar <i>buffer</i> .
Ajustamento de sensibilidade	A sensibilidade é controlada a partir de duas entradas (S0 e S1). Alterações da sensibilidade provocam alterações, na mesma proporção, da área de recepção de luz do fotodíodo. Desta forma garante-se que a responsividade do dispositivo é otimizada para um dado nível de luz incidente, mantendo o intervalo de saída (escala completa) da saída em frequência.

Tabela 23. Numeração e funções dos pinos do circuito integrado [19]

Pino	Porta	Função – TSL230R-LF
1	S0	Entradas de seleção de sensibilidade
2	S1	
3	OE	Ativador de saída
4	GND	Terra
5	V_{DD}	Alimentação elétrica
6	OUT	Saída em frequência (em escala)
7	S2	Entradas para seleção da escala da saída em frequência
8	S3	

Tabela 24. Correspondência entre as saídas analógicas do microcontrolador e as entradas analógicas do PLC

Saídas Microcontrolador	Módulo de entradas Analógicas (PLC)	Endereço (PLC)
AO 4 (PWM)	AI 3	PIW 292

Tabela 25. Seleção de opções disponíveis (TSL230R-LF) [19]

S1	S0	Sensibilidade
L	L	Desligado
L	H	1X
H	L	10X
H	H	100X

S3	S4	Escala f_0
L	L	1
L	H	2
H	L	10
H	H	100

Tabela 26. Condições de operação recomendadas (TSL230R-LF) [19]

Condições de operação recomendadas	Mín	Normal	Max	Unidade
Alimentação elétrica, V_{DD}	2.7	5	5.5	V
Tensão de entrada (para valor lógico "1" = H) (V_{DD} =4.5 V até 5.5 V)	2		V_{DD}	V
Tensão de entrada (para valor lógico "0" = L) (V_{DD} =4.5 V até 5.5 V)	0		0.8	V
Temperatura ambiente	-25		70	°C

Tabela 27. Características de operação quando se verifica: $V_{DD} = 5\text{ V}$; $T_A = 25\text{ }^\circ\text{C}$; $E_e = 130\text{ }\mu\text{W}/\text{cm}^2$; $\lambda_p = 640\text{ nm}$ (TSL230R-LF) [19]

	S3	S2	S1	S0	Min	Normal	Max	f _o em kHz
	L	L	H	H	80	100	120	
	L	L	H	L	8	10	12	
	L	L	L	H	0.8	1	1.2	
	L	H	H	H	40	50	60	
	H	L	H	H	8	10	12	
	H	H	H	H	0.8	1	1.2	
E _e =0	L	L	H	H		0.4	10	

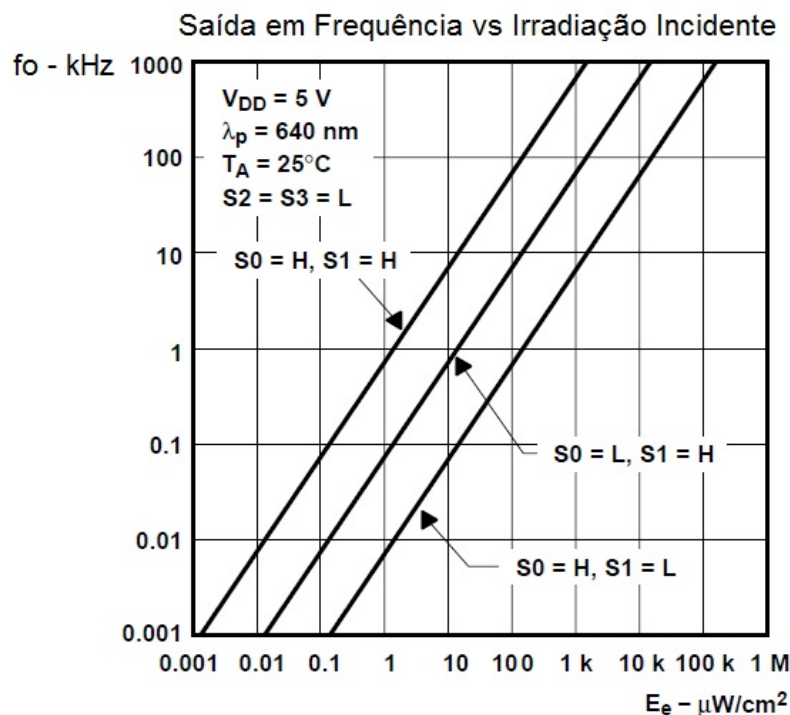


Fig. 47. Representação gráfica da irradiação incidente em ordem à saída em frequência [19]

Atribuição da escala da saída em frequência

A seleção da escala da saída em frequência é controlada a partir de duas entradas lógicas (S2 e S3) e consegue-se por meio de uma conexão interna da saída do conversor (trem de pulsos) a um subcircuito de divisores de frequência.

As saídas disponíveis são divididas por 2, 10, 100 e 1 (sem divisão). Apresentam-se na forma de onda quadrada com ciclo de trabalho de 50%, enquanto a saída direta (sem divisão) na forma de um trem de pulsos com largura de pulso fixa.

Por a divisão da saída em frequência se obter através de contagem de pulsos da frequência principal (sem divisão), o período da saída final representa a média de 2, 10 ou 100 períodos da frequência principal.

Em contraste com o ajuste de sensibilidade, o uso de saídas divididas possibilita a diminuição da frequência (com ou sem incidência de luz), de acordo com o fator de escala selecionado.

A saída principal pode ser usada juntamente com um contador de frequência, acumulador de pulsos, ou temporizador (medição de período).

As saídas divididas podem ser usadas por dispositivos que têm disponível um contador mais lento de frequência, como por exemplo um microcontrolador de baixo custo [19].

Medição da frequência

Somente quando se exigem taxas elevadas de aquisição de dados, é que se implementam técnicas de medição de período de sinal. Não é o caso da aplicação desenvolvida para este projeto.

O uso de saída em frequência, dividida por dois permite a recolha de dados a uma taxa de duas vezes a saída em frequência principal. A seleção de escala de saída pode ser usada para aumentar a resolução de uma determinada taxa de relógio ou para maximizar a resolução, à medida que a luz incidente varia. Podem ser obtidas resolução e precisão máximas por meio do uso de medição de frequência, acumulação de pulsos ou integração.

A medição de frequência disponibiliza o benefício de ponderação de variações aleatórias ou de alta frequência, resultantes de ruído no sinal resultante da incidência de luz ou a partir do ruído resultante da fonte de alimentação.

O método de medição de frequência utilizado é adequado para variação lenta, ou nível de irradiação constante, e para leitura de níveis médios de irradiação ao longo de períodos de tempo curtos. O modo de integração (acumulação de pulsos por um grande período de tempo) pode ser usado para medir a quantidade de luz presente numa área sobre um dado período de tempo [19], [20].

3.3.2 Microcontrolador – Arduino Mega 2560

Como plataforma para desenvolver a integração e programação dos quatro sensores com o PLC, utilizou-se o microcontrolador Arduino ® Mega 2560. Trata-se de uma plataforma de *hardware* livre, projetada com um microcontrolador de placa única, com suporte embutido de entradas e saídas e uma linguagem de programação padrão, de nome *Processing*, com origem em *wiring*, sendo essencialmente C/C++ [20].

Um microcontrolador integra diversos componentes num único circuito integrado, evitando a necessidade de adicionar componentes externos para lhe darem acesso às suas funcionalidades, [22].

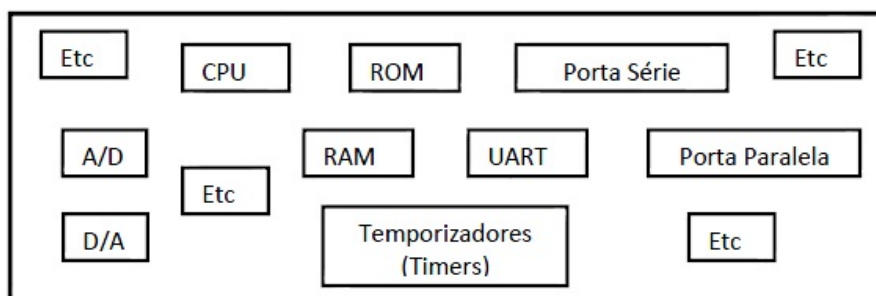


Fig. 48. Diagrama de um microcontrolador genérico e exemplos de funcionalidades [22]

Descrição dos diferentes tipos de memória

- **Bootloader:** *software* gravado no chip que permite a transferência de um programa a ser executado (*sketch*) entre o PC e o microcontrolador, em ambos os sentidos;
- **Memória SRAM** (*Static Random Access Memory*): com funcionamento semelhante à RAM de um PC, é na SRAM que o programa, ao ser executado, cria e modifica o tipo de variáveis necessárias à sua execução. Este tipo de memória apenas mantém os dados enquanto se encontra alimentada eletricamente, o mesmo não acontece com a memória EEPROM;
- **Memória EEPROM** (*Electrically-Erasable Programmable Read-Only Memory*): do tipo volátil, é usada para guardar informação;
- **Memória Flash:** armazena o *bootloader* e o *sketch*);
- Para poupar memória SRAM, necessária para a execução do programa, por vezes são guardadas constantes nas memórias EEPROM e *Flash*, recorrendo-se à bibliotecas “<EEPROM.h>”, ou “<pgmspace.h>”, “<Flash.h>” respetivamente.

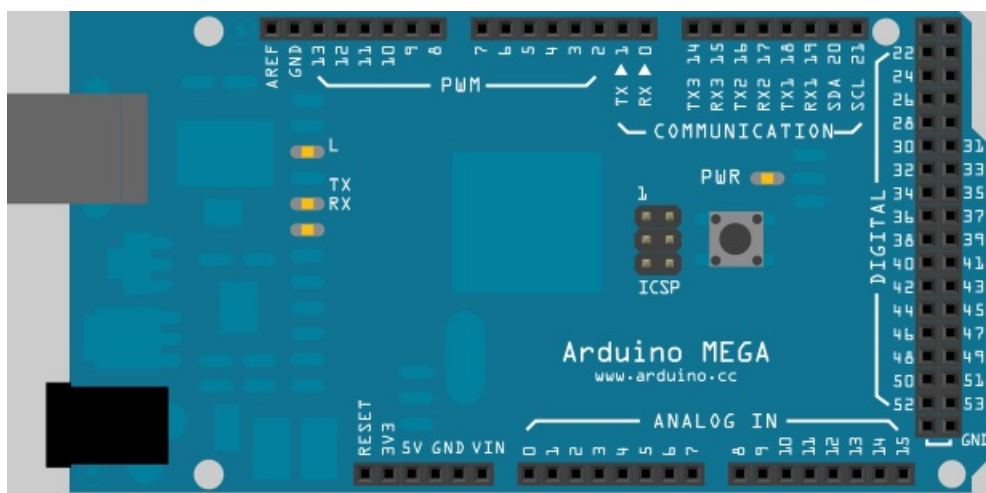


Fig. 49. Microcontrolador Arduino Mega 2560, com destaque para os diferentes tipos de portas de interface [21]

Tabela 28. Características do microcontrolador Arduino Mega [22]

Característica - Arduino Mega 2560	Valores
Tensão elétrica de operação	5 V
Tensão elétrica de entrada	7 – 12 V
Limites de tensão elétrica de entrada	6 – 20 V
Entradas e saídas digitais	54
Entradas analógicas	16
Corrente elétrica DC por pino	40 mA
Memória <i>Flash</i>	256 KB
SRAM	8 KB
EEPROM	4 KB
Velocidade de Relógio	16 MHz

Tabela 29. Alimentação elétrica, interface de entradas/saídas e comunicação do microcontrolador [22]

Alimentação elétrica	Por ligação USB (5 VDC) ou através de uma conexão externa (bateria ou fonte com conversor AC-DC, que disponibilize uma tensão à saída de 7 a 20 V). Existem dois pinos para alimentação externa: Vin e DC <i>power jack</i>
Pinos de alimentação	5 V: Disponibilizados através do regulador da placa do microcontrolador 3.3 V: Disponibilizados pelo regulador da placa (corrente max. de 50 mA) GND: pino de ligação à terra
Entradas e saídas	Possui 54, das quais 15 são saídas do tipo PWM (2 a 13 e 44 a 46); operam a 5 V; cada pino pode receber um max. de 40 mA; possuem resistência interna <i>pull-up</i> (desconectadas por defeito) de 20-50 kOhms
Entradas e saídas especializadas	Os pinos de entrada (2, 3, 18, 19, 20 e 21) podem ser configurados para gerar um disparo de interrupção na existência de um valor baixo de entrada, na existência de um flanco positivo ou negativo, ou numa alteração no valor da entrada.
Comunicação	Possui dispositivos que permitem a comunicação com um computador pessoal ou outro microcontrolador: 4 <i>hardware</i> UARTs (conversão para comunicação serial TTL – 5 V); interface USB que disponibiliza uma porta COM virtual, por <i>software</i> ; o <i>software</i> Arduino ® disponibiliza monitorização serial, que permite a transferência de dados simples de texto entre o microcontrolador e o PC.

Conceito de saída PWM

A capacidade de utilizar *Pulse Width Modulation* (PWM), ou seja, Modelação com Largura de Impulso permite o microcontrolador fornecer uma tensão elétrica analógica a partir de um sinal digital, ou seja, com dois valores lógicos “0” (0 V) e “1” (5 V). O conceito de PWM é utilizado para referir um sinal que possua uma frequência constante e um *duty cycle* variável, [22], [23], [24].

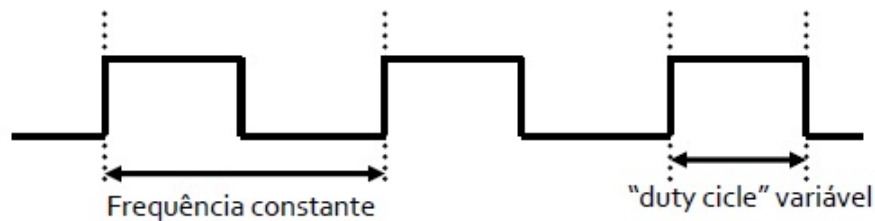


Fig. 50. Forma de sinal PWM

A teoria por detrás da PWM descreve-se a partir da seguinte expressão:

$$V_{DC} = \frac{1}{T} \int_0^T V(t) dt, (2)$$

onde T representa o período do sinal e $V(t)$ representa a tensão elétrica em função do tempo.

Aplicando o conceito de PWM à equação descrita anteriormente, obtemos o seguinte:

$$V(t) = \begin{cases} V_{pulso}, & 0 \leq t \leq t_p \\ 0, & t_p \leq t \leq T \end{cases}, (3)$$

onde t_p representa a duração do impulso e V_{pulso} representa a tensão do impulso do sinal PWM.

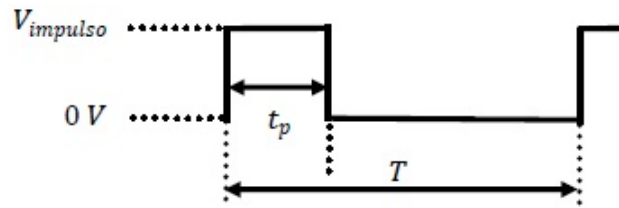


Fig. 51. Representação de um impulso PWM

Aplicando os conceitos referidos anteriormente:

$$V_{DC} = \frac{1}{T} \left(\int_0^{t_p} V_{impulso} dt + \int_{t_p}^T 0 dt \right) = \frac{t_p}{T} \cdot V_{impulso}. \quad (4)$$

A tensão média V_{DC} é diretamente proporcional ao *duty cycle* do sinal PWM, facto que permite fazer variar a tensão, no caso específico do microcontrolador em utilização, entre 0 e 5 V.

Descrição da conversão A/D no microcontrolador

O microcontrolador utilizado possui um conversor analógico-digital de 10 bits ($2^{10}=1024$). Como a tensão máxima de referência, se encontra nos 5 V, correspondendo ao valor 1023, obtemos a seguinte resolução: $5/1024 \sim 0.00488 \sim 5 \text{ mV}$. O que significa que o valor lido pelo microcontrolador só se altera a cada 5 mV de variação do sinal de entrada analógico.

Existe a possibilidade de utilizar uma resolução, de todas as entradas analógicas, de 2 mV. Para isso há que utilizar o pino correspondente à entrada *AREF*. Para obter mais informação sobre este tema sugere-se a consulta da bibliografia [22], [24].

Capítulo 4 – Descrição do software do subsistema secundário

4.1 Instalação e utilização do software de programação

De seguida vão ser apresentados os passos mais importantes para efetuar a instalação do *software* de desenvolvimento e dos *drivers*, que permitem a transferência de programas de utilizador entre o PC e o microcontrolador, bem como os passos fundamentais para utilizar o software Arduino:

1. Obter o software a partir do site (Arduino), descompactar a pasta transferida;
2. Conectar o cabo USB ao microcontrolador e ao PC;
3. Instalar o *driver* FTDI “*Arduino MEGA 2560 REV3*” (pasta referida no ponto 1);
4. Correr o software Arduino, configurar a porta série a ser utilizada e seleccionar o modelo de microcontrolador Arduino, como é possível observar na fig. 52;

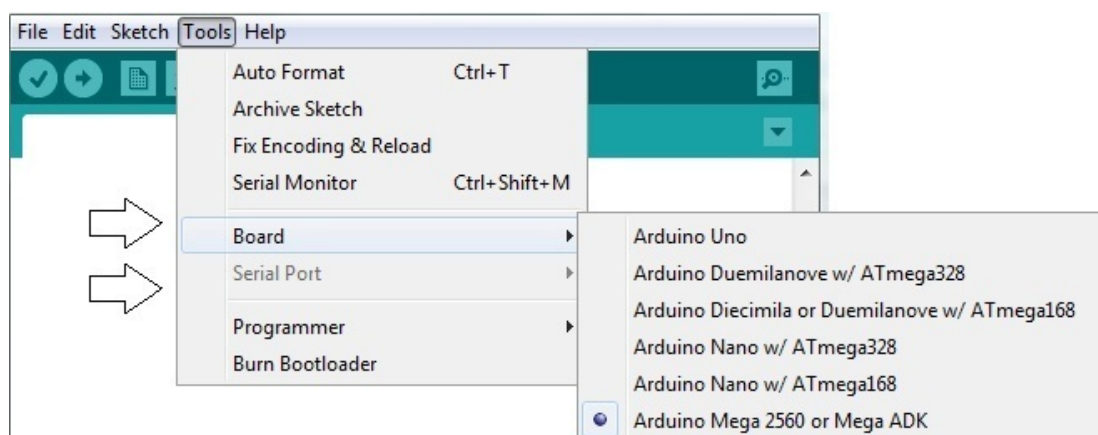


Fig. 52. Escolha da porta série e do modelo de microcontrolador, no software de desenvolvimento Arduino

5. Elaborar o programa a executar no microcontrolador (*sketch*), verificar a existência de erros  e transferir para o Arduino .

4.2 Classificação face ao conjunto de instruções

Os sistemas baseados em arquiteturas RISC (*Reduced Instruction Set Computers*) apresentam um reduzido conjunto de instruções, possibilitando obter simplicidade e tempo de execução por instrução menor que sistemas CISC (*Complex Instruction Set Computers*). Por possuir arquitetura RISC o Arduino possui elevada velocidade de processamento. Na sua versão mais simples, os sistemas baseados na arquitetura RISC possuem os seguintes tipos de instruções:

- Instruções lógicas e aritméticas sobre registos;
- Instruções de transferência de dados entre memória e registos;
- Instruções de controlo.

4.2.1 Funções base

“void setup”

Esta função é executada uma vez, para a inicialização de variáveis, a inicialização da utilização de bibliotecas, a definição dos pinos (como *input* ou *output*), o início do uso de comunicação série, etc... Esta função apenas volta a ser executada ao ser efetuado *reset* ou quando se desligar e volta a ligar o microcontrolador.

“void loop”

Todos os comandos existentes no interior desta função são sucessivamente repetidos, o que pode permitir a leitura sucessiva de portas, a leitura sucessiva de parâmetros provenientes de sensores externos e atuar de acordo com as condições estabelecidas.

As funções atrás descritas têm de ser necessariamente do tipo “*void*”, ou seja, não podem retornar qualquer valor depois da sua execução. As funções “*void setup()*” e “*void loop()*” são de carácter obrigatório, ou seja, mesmo que não seja necessária a sua utilização deverão constar no código utilizado. Apenas serão chamadas funções externas que constem na função “*void loop()*” [24].

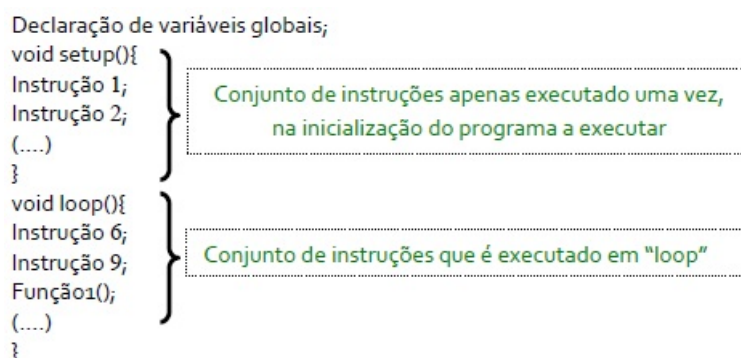


Fig. 53. Estrutura base de um programa para execução num microcontrolador Arduino [22]

A variável da saída, do tipo PWM, pode receber valores entre 0 e 255 (resolução de 8 bits) e, desta forma, fornecer ao sistema principal uma tensão elétrica compreendida entre 0 e 5 V. O valor que recebe é de acordo com a verificação das condições programadas, de forma a indicar quais os integrados TSL230R-LF expostos a uma maior irradiação.

Tabela 30. Condições programadas para indicação ao subsistema primário do varrimento a efetuar

Varrimento	Condição (comparação das saídas dos sensores)	Valor digital	Variável de saída [AO4]
1	S1 > S3 & S1 > S4 & S2 > S3 & S2 > S4	102	2.0 V
2	S2 > S1 & S2 > S4 & S3 > S1 & S3 > S4	128	2.5 V
3	S3 > S1 & S3 > S2 & S4 > S1 & S4 > S2	153	3.0 V
4	S4 > S2 & S4 > S3 & S1 > S2 & S1 > S3	179	3.5 V
5	S1 > S2 & S1 > S4 & S3 > S2 & S3 > S4	204	4.0 V
Total	Quando nenhuma das condições anteriores se verifica	255	5.0 V

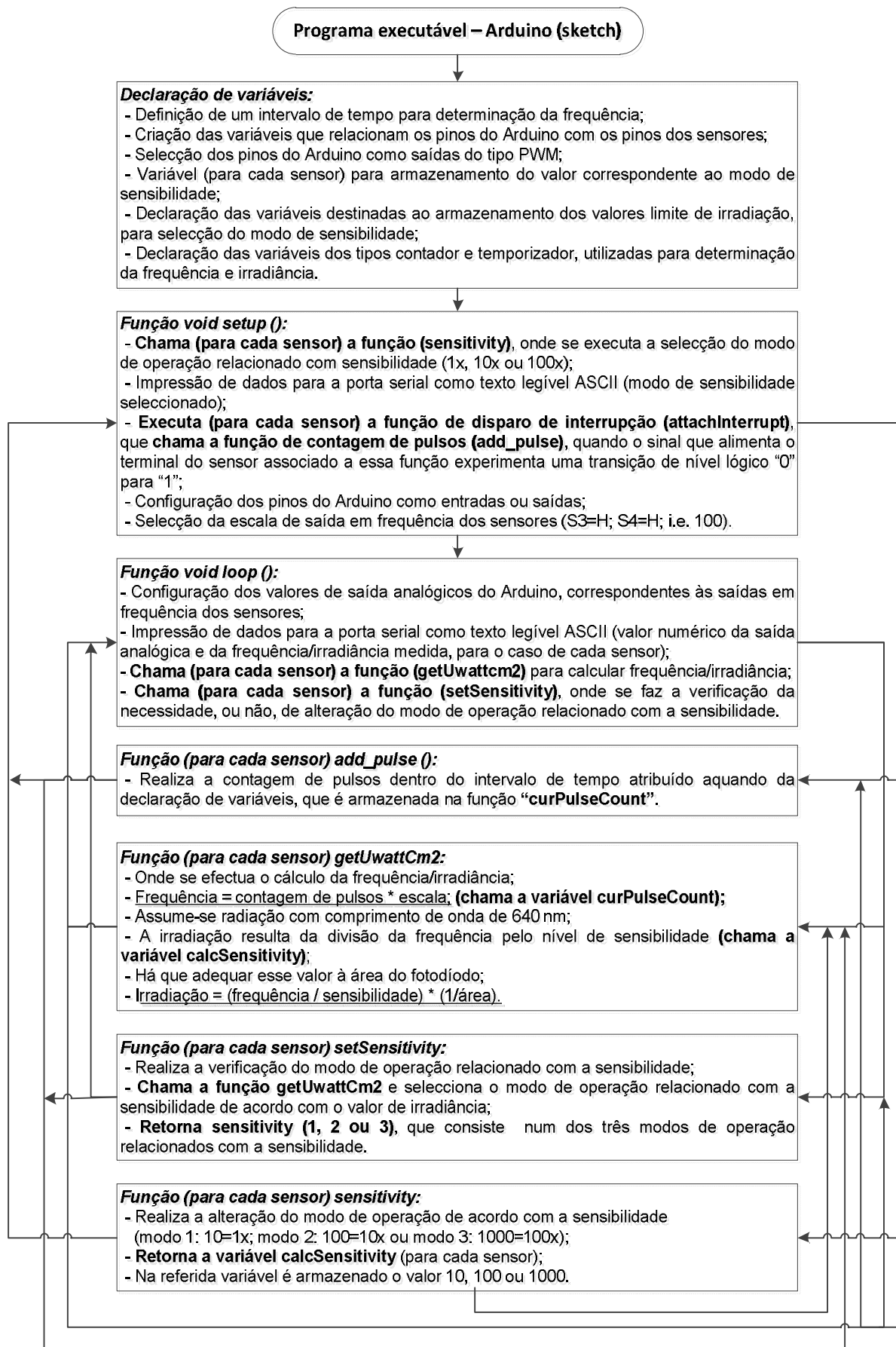


Fig. 54. Relação entre as funções do programa executável (sketch) do microcontrolador

Capítulo 5 – Descrição do software do subsistema principal

Etapas da implementação da componente de *software* do sistema principal:

- Projeto em STEP 7 V5.5;
- Configuração de hardware;
- Programa executável para PLC Siemens S7-300;
- Aplicação para painel operacional (HMI) em WinCC Flexible 2008;
- Sistema de supervisão em WinCC V 7.0.

5.1 Criação do projeto em STEP-7

Antes de desenvolver a programação das instruções que gerem o funcionamento do sistema através do PLC, já conhecendo a sua constituição modular (capítulo IV) e sabendo que a plataforma de programação é o *software* STEP-7, há que conhecer os tipos de variáveis e de dados disponíveis.

É possível relacionar as variáveis de acordo com a função, tipo de dados que armazenam e zona de atuação. Existem variáveis que se identificam pelo endereço de memória do PLC (M, MB, MW e MD), outras que representam endereços de memória específicos para armazenar resultados de contagem (C) e temporização (T). Existem também variáveis cuja função é armazenar o valor lógico das entradas e saídas digitais (I e Q), outras o valor numérico das entradas e saídas analógicas (PIB, PIW, PID e PQB, PQW, PQD), de acordo com o espaço de memória necessário [12].

Tabela 31. Variáveis mais utilizadas na programação em STEP 7

Tipo de variável	Designação	Disponibilidade de acordo com a configuração de hardware	Endereços obtidos com a configuração de hardware
Memory Bit	M	2048	M0.0 até M255.0
Timer	T	128	T0 até T127
Counter	C	64	C0 até C63
Memory Byte	MB	256	MB0 até MB255
Memory Word	MW	128	MW0 até MW127
Memory Double Word	MD	64	C0 até C63
Input Bit	I	1x8	I4.0 até I4.7
Output Bit	Q	1x8	Q4.0 até Q4.7
Peripheral Input Byte	PIB	16	PIB288 até PIB303
Peripheral Output Byte	PQB	8	PQB304 até PQB311
Peripheral Input Word	PIW	8	PIW288 até PIW302
Peripheral Output Word	PQW	4	PQW304 até PQW310
Peripheral Input Double Word	PID	4	PID288 até PID300
Peripheral Output Double Word	PQD	2	PQD304 até PQD308

Tabela 32. Apresentação de tipos de dados simples

Bit	Byte	Word	Double Word
BOOL	BYTE CHAR	WORD INT DATE S5TIME	DWORD DINT REAL TIME TIME OF DAY

Tabela 33. Definição das variantes de dados simples

Dado simples	Descrição
BOOL	Dado binário TRUE ou FALSE, “0” ou “1”
BYTE	Ocupa 8 bits e pode assumir um número em base binária (2#00000000 a 2#11111111) ou hexadecimal (B#16#0 até B#16#FF)
CHAR	Ocupa 8 bits e pode assumir um caractere ASCII (“A” ou “a”, p.e.)
WORD	Ocupa 16 bits e pode assumir um número em base binária (2#0000000000000000) a 2#1111111111111111) ou hexadecimal (W#16#0 a W#16#FFFF)
INT	Ocupa 16 bits e pode assumir um número inteiro com sinal ($\pm 2^{15} = \pm 32768$), já que o bit mais significativo corresponde ao sinal. Desta forma, pode representar um número entre L#-32768 e L#32767
DATE	A data apresenta-se como um inteiro sem sinal (0 a 65535), em que o primeiro dia é 1 de janeiro de 1990. Apresenta-se no formato yyyy-MM-dd e consegue representar datas desde D#1990-01-01 até D#2168-12-31
S5TIME	Os temporizadores podem assumir valores entre S5T#10ms e S5T#2h46m30s
DWORD	Idêntico ao WORD mas com a diferença de possuir 32 bits. Pode assumir um número em base binária (2#00000000000000000000000000000000) a 2#11111111111111111111111111111111) ou hexadecimal (DW#16#0 a DW#16#FFFFFFFF)
DINT	Idêntico ao INT mas com a diferença de possuir 32 bits. Desta forma, consegue representar números desde L#-2147483648 até L#+2147483647
REAL	Representa números reais de 32 bits sob a forma de vírgula flutuante (norma ANSI/IEEE 754-1985). O menor número em vírgula flutuante possível de ser utilizado é 1,17595E-38 e o maior 3,402823E+38
TIME	Consiste numa representação, em 32 bits, de um intervalo de tempo. Intervalo de representação: T#-24d20h31m23s647ms e T#+24d20h31m23s647ms
TIME OF DAY	Representa uma determinada hora do dia no formato: TOD#23:59:59:999

O STEP 7 possui uma interface de utilizador gráfica para configurar todo o hardware do PLC, bem como iniciar um projeto de programação. A partir deste software acede-se a qualquer função instalada no sistema (sistema padrão e todas as aplicações que funcionam em conjunto com este). Em termos funcionais é possível criar/abrir projetos, configurar e introduzir parâmetros de hardware (*HW Config*), estabelecer as configurações de comunicação (*NET Pro*), desenvolver a programação (Editor LAD/STL/FBD), testar a programação, iniciar a sua execução, aceder ao projeto armazenado no disco duro da estação de programação (modo *offline*) ou aceder ao projeto armazenado no PLC (modo *online*) e trabalhar com os módulos de memória [12].

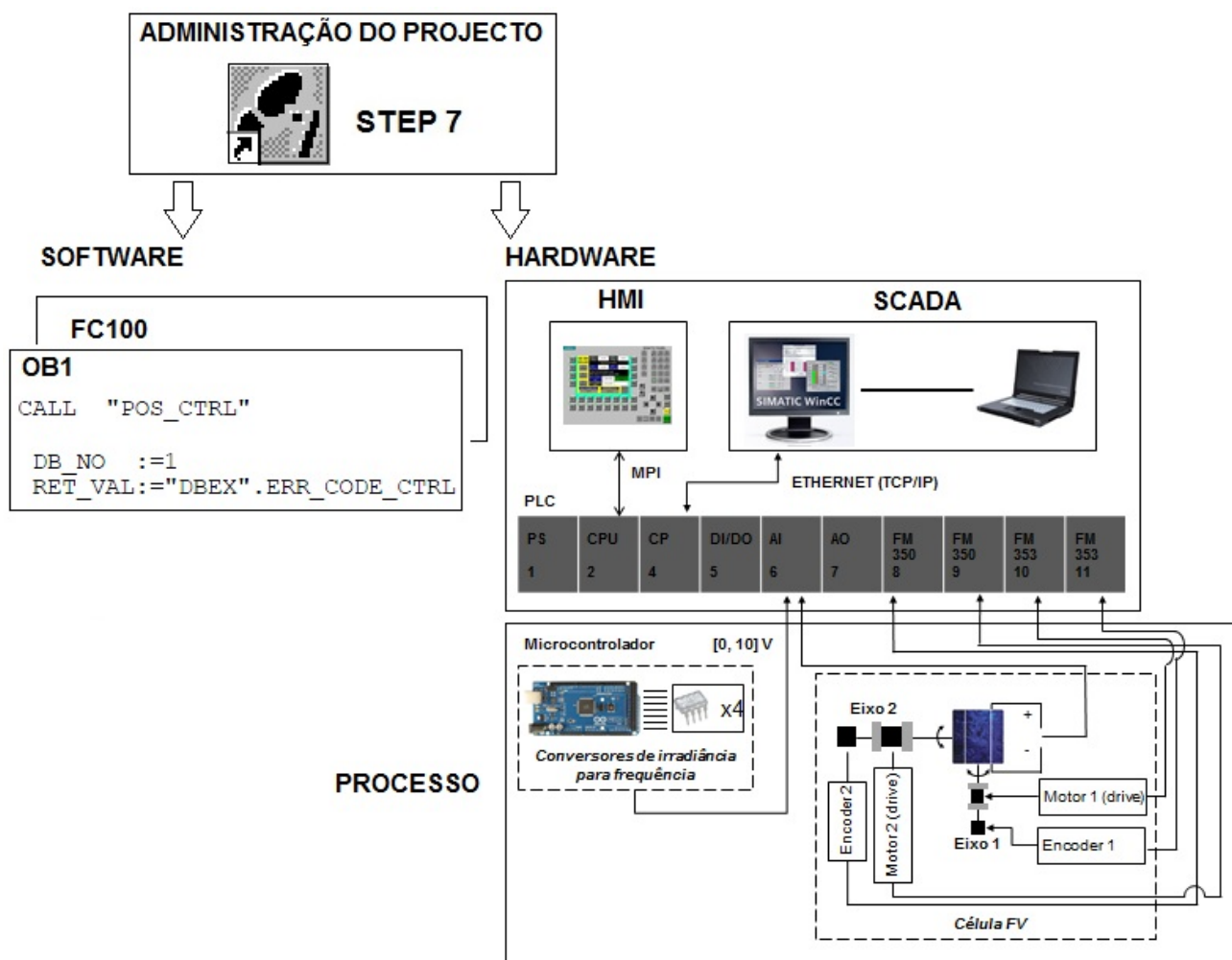


Fig. 55. Papel do gestor de projetos STEP 7 na administração de projetos

Para criar um novo projeto há que iniciar o software SIMATIC MANAGER STEP 7, na barra de ferramentas seleccionar o separador ficheiro e clicar em novo. Na janela que surge, definir o nome do projeto e a diretoria onde fica guardado.

Os dados são armazenados num projeto sob a forma de objetos. Os objetos são organizados numa estrutura hierárquica em árvore. No primeiro nível encontra-se um ícone com o nome do projeto. Cada projeto representa uma base de dados onde são armazenados todos os dados relevantes. No segundo nível há que introduzir estações e outros objetos necessários.

Quando se cria um novo projeto, este surge já com o objeto "MPI(1)", porque o STEP 7 vem configurado, por defeito, para que a comunicação com o autómato seja feita por MPI. O passo seguinte é introduzir uma estação S7-300, de acordo com o modelo de PLC utilizado. Aqui fica armazenada toda a informação sobre a configuração do hardware e atribuição de parâmetros aos módulos. A estação S7-300 é o ponto de partida para a configuração de hardware. Após esta ter sido executada ficam disponíveis as subestações "CPU 315-2 DP", "FM 353 STEPPER M1" (motor 1), "FM 353 STEPPER M2" (motor 2) e "CP 343-1 Advanced" e o ícone "Hardware" para aplicação de configuração de hardware (*HW Config*). Nas subestações acabadas de referir ficam armazenadas as pastas da componente de software para atribuição de parâmetros aos módulos da gama S7.

A subestação “CPU 315-2DP” é criada contendo no seu interior dois objetos: a pasta “S7 Program(1)” e o ícone “Connections” para a aplicação de configuração de comunicação “NetPro”.

A pasta “S7 Program” é o ponto de partida para a introdução da programação do sistema. É criada contendo no seu interior as subpastas “Blocks” e “Source Files” e o ícone “Symbols” para a aplicação onde se faz a atribuição dos símbolos aos endereços de memória utilizados na programação (tabela de símbolos). A subpasta “Blocks” é criada contendo unicamente no seu interior o bloco OB1 (vazio). Um bloco numa CPU consiste numa secção do programa que tem uma função e estrutura particulares. O bloco OB1 é chamado ciclicamente pelo sistema operativo (do PLC), pode ter instruções e/ou chamada de outros blocos/funções.

As subestações “FM 353 STEPPER M1” e “FM 353 STEPPER M2” são criadas depois de introduzidos os dois módulos funcionais FM 353, aquando da configuração de hardware, e contêm no seu interior uma pasta, para armazenamento dos dados resultantes da sua configuração. Esses dados serão posteriormente transferidos para os referidos módulos e acedidos a partir do programa de utilizador, transferido e armazenado na CPU do PLC.

No mesmo nível hierárquico da estação SIMATIC 300 e depois de efetuada a configuração de hardware, há que introduzir a estação relacionada com a aplicação desenvolvida para o painel operacional do sistema (“SIMATIC HMI Station”).

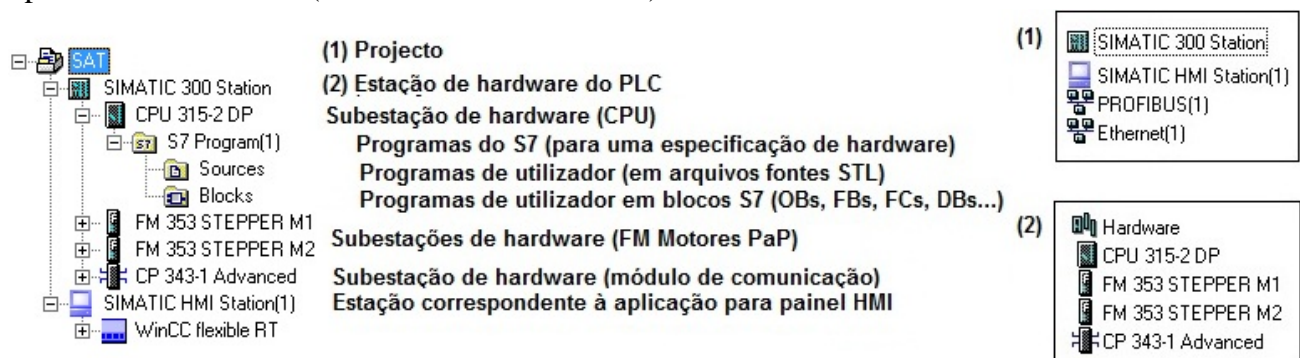


Fig. 56. Árvore hierárquica do Projeto, com detalhe para o conteúdo da pasta projeto e da estação SIMATIC 300

5.2 Configuração de hardware

Antes de se iniciar a configuração de hardware há que efetuar, em primeiro lugar, a instalação de dois pacotes de software que complementam as aplicações do STEP 7 e permitem a configuração, por intermédio de uma interface gráfica, dos módulos FM 350-1 e FM 353 (“*Configuration Package for FM 350-1 and FM 450-1 Counter Modules*” e “*FM 353 and FM 354 Configuration Package V4.03.07*”). Os ficheiros de instalação das referidas aplicações estão disponíveis para transferência no portal Web da Siemens.

Estando a correr a aplicação “HW Config” o primeiro objeto a introduzir é uma rack, que fisicamente representa a estrutura metálica de suporte dos módulos do PLC e na interface gráfica *HW Config* consiste numa tabela com onze linhas, que corresponde ao número máximo de módulos possível de conectar à CPU utilizada. A adição dos módulos é bastante intuitiva, bastando selecionar a linha da tabela que corresponde à posição (*slot*) que um módulo ocupa na *Rack* e arrastar para esta

a versão adequada do módulo, a partir da zona de catálogo de módulos, de acordo com as especificações indicadas na tabela 4. Notar que a introdução de módulos tem regras que se coadunam com as suas características técnicas, como já foi referido no capítulo III.

O primeiro módulo a ser introduzido corresponde à CPU. A versão em utilização dispõe de uma interface de comunicação PROFIBUS-DP e há que efetuar a configuração de rede (figura 57), que estabelece a comunicação com o painel HMI.

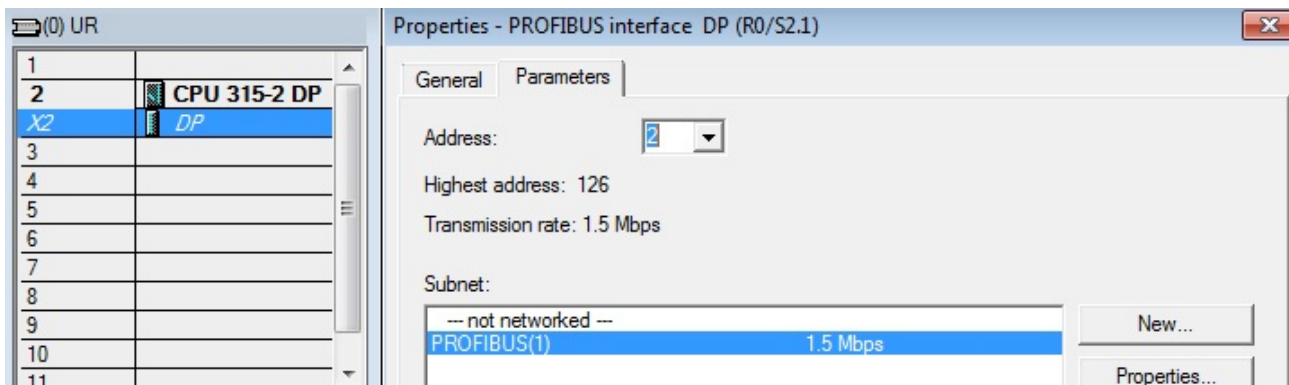


Fig. 57. Configuração de rede PROFIBUS com endereço 2 e velocidade de transmissão 1.5 Mbps

Na tabela de configuração, o STEP 7 automaticamente atribui um endereço para cada módulo. Contudo, um endereço pode ser escolhido para cada canal do módulo, independentemente do seu *slot*.

Introduzida a CPU na tabela da Rack do PLC surge, na árvore do projeto, a subestação “CPU 315-2 DP”. Segue-se a colocação na pasta “Blocks” (fig. 56) dos blocos de dois projetos, um fornecido com o pacote de instalação para o FM 350-1 (zEn23_01_FMx50-1) e outro fornecido com o pacote de instalação para o FM 353 (zEn13_02_FM353_Ex). São acedidos a partir do separador “Sample Projects”, da janela “Open Project”. Notar que a programação apenas contempla um encoder e um motor passo-a-passo e assim sendo fez-se a duplicação dos referidos blocos. É a partir destes que se começa a desenvolver o programa de utilizador, uma vez que contêm funções desenvolvidas especificamente para os modelos FM em utilização.

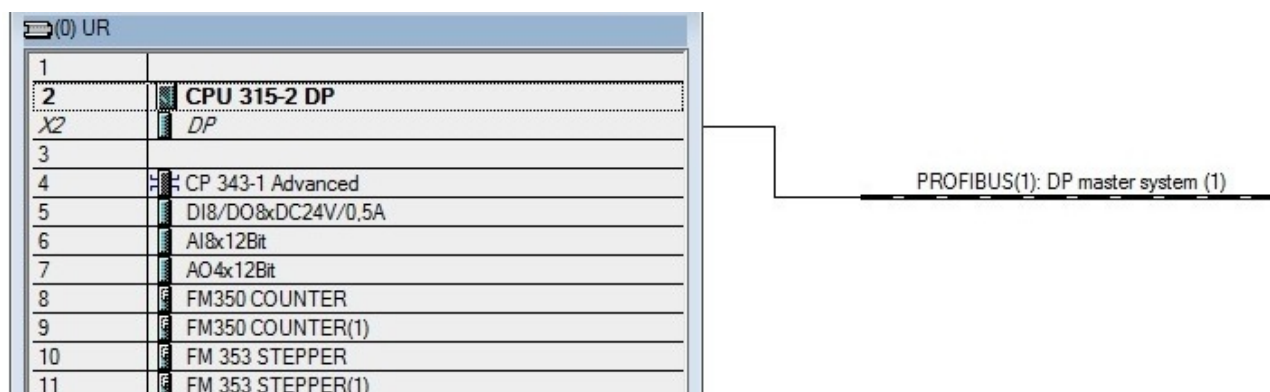


Fig. 58. Introdução dos módulos na tabela que representa a rack do PLC

O autômato possui um módulo CP 343-1 Advanced. Este permite a comunicação entre o PLC e o sistema de supervisão. Para tal, houve a necessidade de criar uma rede ethernet (TCP/IP). Estando a rede criada, caso se selecione corretamente a interface de comunicação entre PC e PLC, na aplicação “*Set PG/PC Interface*”, que se acede a partir do Painel de Controlo, é possível transferir as configurações e os blocos de programação para o PLC, a partir da rede criada previamente.

A rede ethernet cria-se no menu de propriedades do módulo CP 343-1 Advanced, tendo sido escolhido um IP da mesma gama do IP da ligação de área local em utilização.

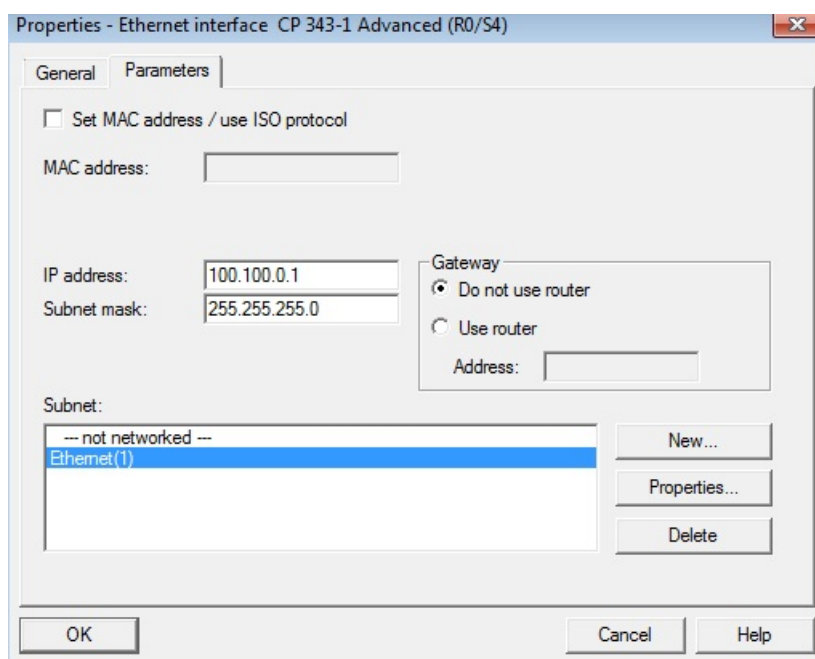


Fig. 59. Criação da rede Ethernet (TCP/IP)

As figuras que se seguem resumem as configurações de hardware e redes de comunicação implementadas.

Slot	Module	Order number	Firmware	MPI address	I address	Q address	Comment
1							
2	CPU 315-2 DP	6ES7 315-2AG10-0AB0	V2.6	2			
	<i>DP</i>				<i>204,7"</i>		
3							
4	CP 343-1 Advanced	6GK7 343-1GX21-0XE0	V1.1	3	256...271	256...271	
5	DI8/DO8xDC24V/0,5A	6ES7 323-1BH01-0AA0			4	4	
6	AI8x12Bit	6ES7 331-7KF02-0AB0			288...303		
7	AO4x12Bit	6ES7 332-5HD01-0AB0				304...311	
8	FM350 COUNTER	6ES7 350-1AH03-0AE0			320...335	320...335	
9	FM350 COUNTER(1)	6ES7 350-1AH03-0AE0			336...351	336...351	
10	FM 353 STEPPER	6ES7 353-1AH01-0AE0		4	352...367	352...367	
11	FM 353 STEPPER(1)	6ES7 353-1AH01-0AE0		5	368...383	368...383	

Fig. 60. Tabela da aplicação *HW Config* com colunas: de endereço MPI, endereços de entrada e endereços de saída

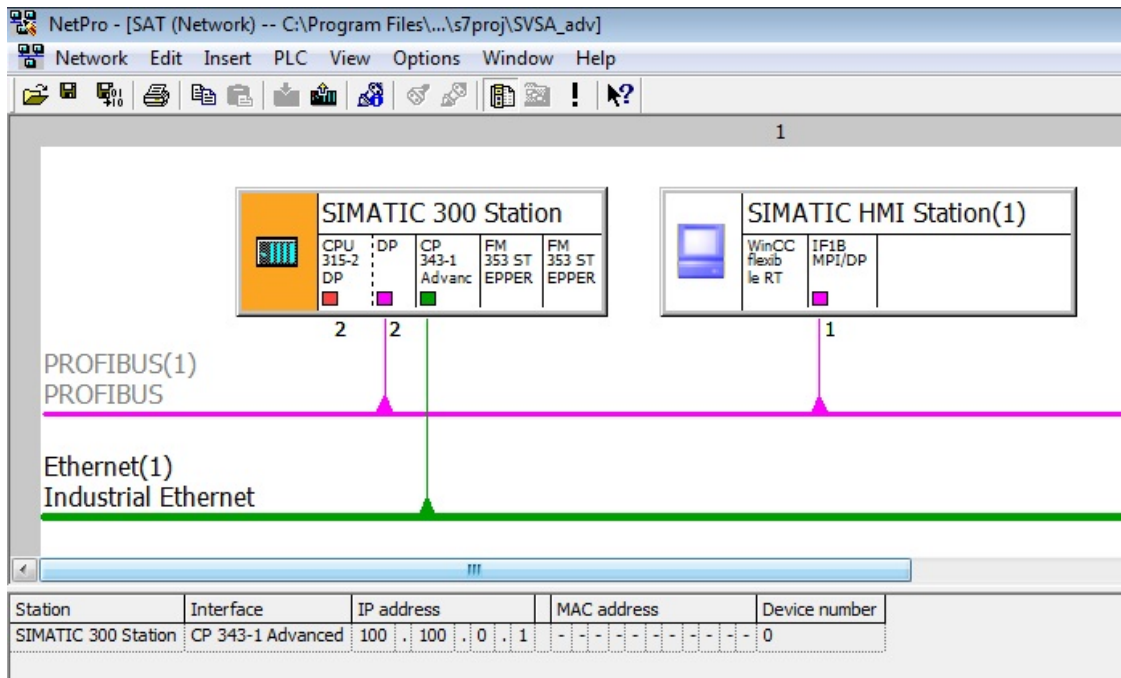


Fig. 61. Aplicação “NetPro” do STEP 7, com destaque para rede Ethernet

Finalizado o processo de configuração de hardware, executa-se a transferência dessa informação para o PLC, “*Download to PLC*”, na aplicação “*HW Config*”.

5.2.1 Parametrização dos módulos FM 350-1

Para integrar o funcionamento dos FM 350-1 com o PLC recorre-se a um pacote de software, que constitui um upgrade ao software de programação do PLC (STEP 7).

- Software para integração/comunicação do FM com a CPU;
- Aplicação para configuração/parametrização.

A partir do STEP 7, o programador pode efetuar a parametrização, transferi-la se estiver conectado ao PLC ou guardá-la num bloco de sistema (SDB). Acede-se a essa aplicação de parametrização a partir da configuração de hardware.

A partir da instalação do pacote de software tem-se acesso à função “FC CNT_CTRL”, que foi colocada no programa de utilizador (blocos de programação desenvolvidos). Esta função é primordial à comunicação entre CPU e FM 350-1.

O módulo FM 350-1 pode ser colocado na rack como qualquer módulo de sinal (SM) em qualquer *slot* compreendido entre os números 4 e 11 inclusive. Há que ter em atenção que na constituição do PLC utilizado, numa única rack, apenas podem coexistir oito SMs ou FMs em simultâneo. A restrição deve-se ao consumo energético de cada dispositivo e à memória requerida pelo software na CPU. Há que atribuir um endereço de comunicação entre a CPU e o FM350-1. Esse endereço é introduzido num DB (*data block*), por meio da uma parametrização já referida.

O endereço inicial (a partir do qual se obtêm os outros) pode ser fixo, de acordo com as regras de colocação de módulos nos *slots* disponíveis de uma *rack* ou escolhido pelo programador, através

do STEP 7. Quando se faz a configuração e hardware o programador obtém automaticamente os endereços, contudo pode-se utilizar a seguinte fórmula: $Endereço = 256 + (n.º \text{ de rack} * 128) + (n.º \text{ de slot} - 4) * 16$ [14].

A configuração dos módulos FM 350-1 está discriminada passo-a-passo no Anexo C.

5.2.2 Parametrização dos módulos FM 353

Para integrar o funcionamento dos FM 353 com o PLC, de acordo com as características e funcionalidade requeridas para os motores, também se recorre a um pacote de software que constitui um upgrade ao software de programação (STEP 7).

- Software para integração/comunicação do FM com a CPU;
- Aplicação para configuração/parametrização.

Após a instalação do upgrade, fica habilitado o acesso a uma aplicação de configuração dos FM 353, a partir da aplicação “*HW Config*” do STEP 7. A configuração dos módulos FM 353 está discriminada na totalidade e passo-a-passo no Anexo D.

O módulo FM 353 é parametrizado indiretamente a a partir da configuração de blocos de dados (“*data block*” – ver sub-capítulo 5.3), que posteriormente são transferidos e armazenados na memória do módulo, sendo-lhes atribuído um número identificativo do seu endereço de memória compreendido entre 1001 e 1239. O pacote de software instalado também contém funções tecnológicas sobre as quais se constroi o programa de utilizador (ver sub-capítulo 5.3).

Para um módulo FM 353 existem os seguintes DBs, associados à sua parametrização:

- “*Machine Data*” (MD);
- “*Increment Sizes*” (SM);
- “*Tool Offset Data*” (TO);
- “*Traversing Programs*” (NC);
- *User Data (user data blocks)* – único armazenado na memória da CPU.

Dos DBs referidos destaca-se o DB-MD (“*Machine Data*”), dado que é o único que é sempre solicitado em qualquer forma de utilização do FM 353, por ser aquele que recebe os dados relativos aos motores e os disponibiliza nas funções tecnológicas existentes no código de programação, definindo, p.e, as características de movimento (linear ou angular), a amplitude dos movimentos, ou estabelecendo como se encontra a posição inicial do sistema, ou determinado como este para numa posição desejada (ver Anexo D). Destaca-se ainda o DM-NC (“*Traversing Programs*”), que possibilita uma fácil programação (G-code) de movimento sequencial do motor, para execução automática dos movimentos do sistema.

Após terminar a configuração é necessário enviar os DBs para a memória de cada módulo FM 353 (eixo 1 e eixo 2), em “*Transfer data to FM*” na aplicação “*FM353 Parameterization*” (ver anexo D).

A informação da parametrização de um módulo FM é guardada sob a forma de um bloco de dados de sistema (SDB), útil no caso de necessidade de reenvio da configuração do PLC e do programa de utilizador.

O endereço do FM 353 é atribuído automaticamente, mas pode ser selecionado pelo programador. Essa atribuição faz-se através da identificação do “*User Data Block*”, que consiste

num bloco gerado a partir de um “*User-defined Data Type*” (UDT) que estabelece a interligação entre os DBs do FM 353 [18].

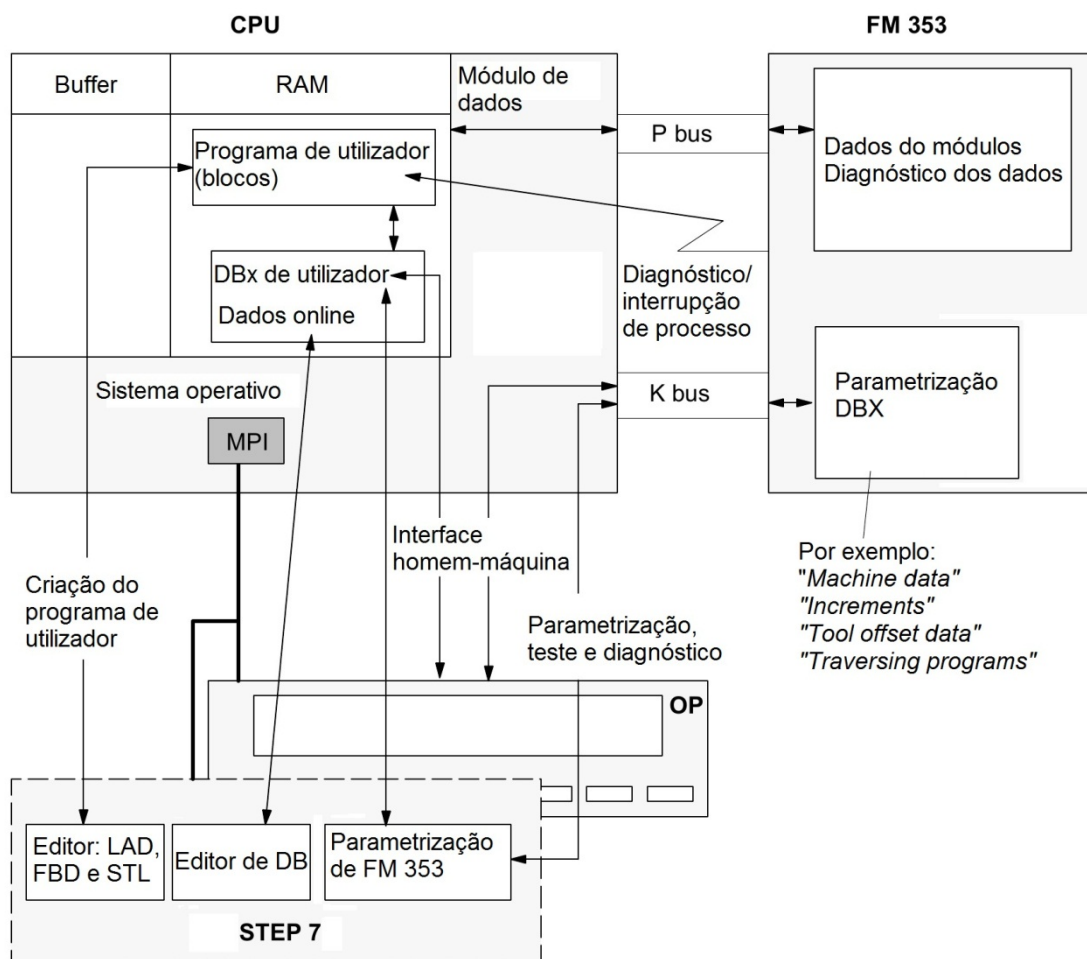


Fig. 62. Diagrama com a estrutura da configuração de um FM 353, com recurso ao pacote de software “Parameterize FM353”, adaptado de [18]

5.3 Programa executável

Os blocos desenvolvidos pelo programador ficam armazenados na subpasta “*Blocks*” da subestação “*CPU 315-2 DP*”, para serem transferidos para a memória da CPU. Os blocos que resultam da configuração/parametrização dos módulos FM 350-1 e FM 353 ficam armazenados nas subpastas que correspondem às respetivas subestações, para quando finalizada a programação serem transferidos para os módulos a que correspondem, na forma de DBs.

Para melhor se compreenderem as diferenças entre os blocos possíveis de usar na programação do sistema, elaborou-se a tabela que se segue.

Tabela 34. Indicação e descrição dos blocos do utilizador [25]

Blocos do Utilizador	
OB	Bloco de Organização (“<i>Organization Block</i>”) : realiza a interface entre a CPU e o programa de utilizador. OB1 é o bloco de organização principal, como o menor grau de prioridade. É possível escrever/programar apenas no OB1 ou pode-se dividir a programação por diferentes blocos, que são acedidos a partir do OB1. O sistema operativo do PLC pode chamar outros OBs, com diferentes prioridades, que reagem a eventos: interrupção data programada; interrupção de diagnóstico; interrupção de erro; interrupção de tempo de ciclo; interrupção de hardware; start-up de hardware.
FB	Bloco de Função (“<i>Function Block</i>”) : função ou sequência de comandos armazenados num bloco, onde os parâmetros podem ser arquivados numa memória. Esta memória surge na forma de um bloco de dados (DB) do tipo <i>instance</i> , e possui endereço de memória. Parâmetros e alguns dados locais podem ser retidos neste DB associado a esta FB, de forma a não se perderem quando a FB deixar de ser acedida.
FC	Função (“<i>Function Call</i>”) : A função é um bloco, similar a uma FB, ao qual não é atribuída uma área de memória e, como tal, não necessita da associação a um DB do tipo <i>instance</i> . Desta forma, deixando-se de aceder à FC os seus dados/parâmetros perdem-se.
DB	<p>Bloco de dados (“<i>Data Block</i>”): são utilizados para armazenar valores em variáveis (endereços de memória do tipo DB), que não se perdem quando o DB deixa de ser acedido.</p> <p>DBs partilhados (“<i>Shared</i>”): podem ser acedidos, para leitura ou escrita, por todos os blocos do programa de utilizador (OBs, FCs e/ou FBs). Estes podem ser criados de duas formas:</p> <ul style="list-style-type: none"> • Aquando da criação do DB, por meio da criação e edição da ordem das variáveis, da sua representação em símbolo (nome) e da atribuição do tipo de dado de cada uma; • Através da criação de um UDT (“<i>User-Defined Data Type</i>”), cuja sua estrutura define a do DB. <p>DBs <i>instance</i>: relacionam-se com uma determinada FB e retêm os seus dados locais.</p>

Tabela 35. Indicação e descrição dos blocos de sistema [25]

Blocos de Sistema	
SFC	Função de Sistema: é arquivada e integrada no Sistema Operativo da CPU. Suportam tarefas de parametrização de módulos, redes de comunicação, funções de cópia, etc... Pode ser acedida pelo programa de utilizador, porém sem fazer parte dele (não ocupa memória de trabalho).
SFB	Bloco de Função de Sistema: Sem ser carregada, pode ser acedida pelo programa de utilizador, fazendo parte do Sistema Operativo do PLC. Devem ser associadas a um DB, este sim fazendo parte do programa de utilizador.
SDB	Bloco de Dados de Sistema: consiste numa área de memória para armazenamento de dados necessários para a gestão de operações (dados de configuração/parametrização; conexões/redes de comunicação).

A programação foi desenvolvida em arquitetura estruturada, em códigos reutilizáveis separados, por funcionalidade, em blocos individuais. Desta forma, o bloco OB1 acede os diversos blocos (FCs, FBs, DBs...), ocorrendo troca de dados/parâmetros relevantes entre vários blocos, de acordo com o código programado.

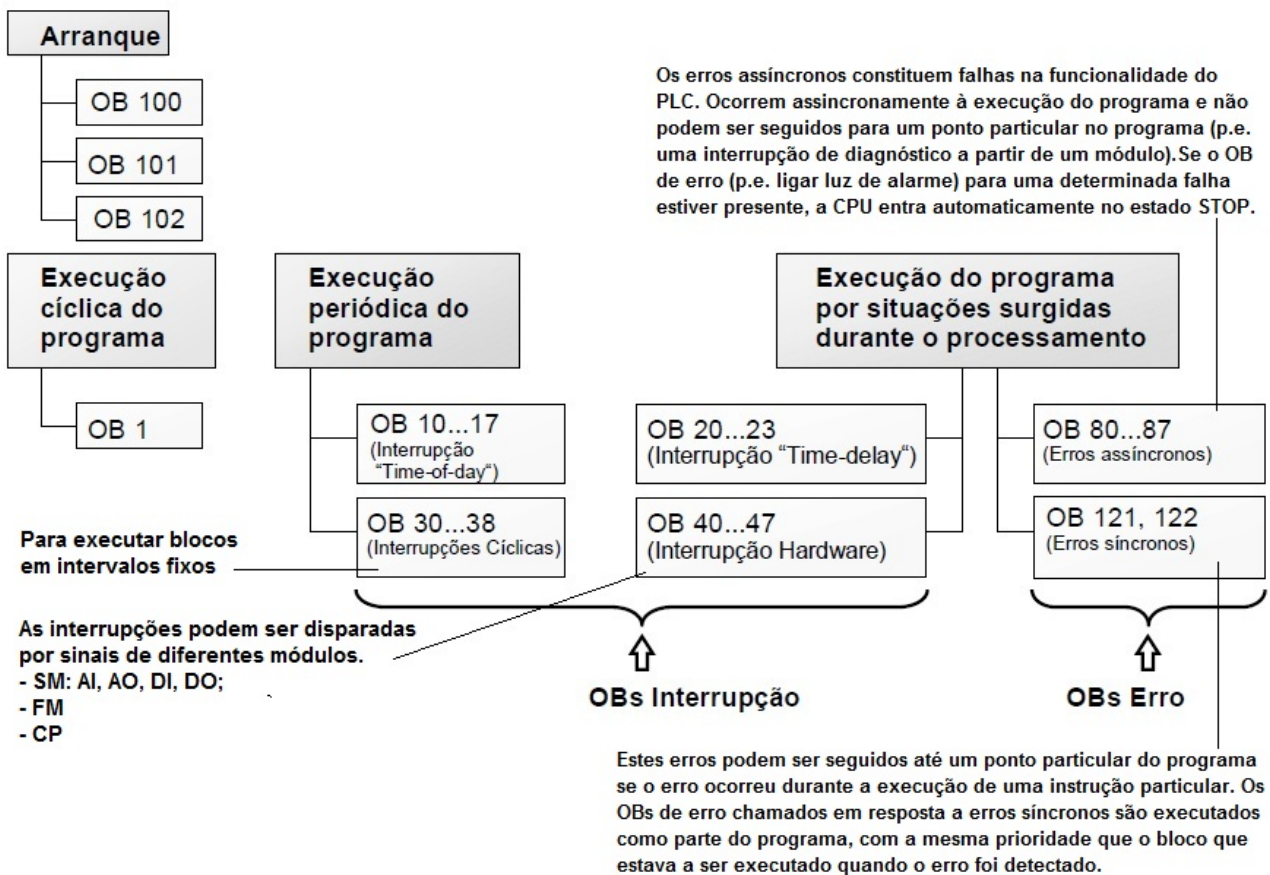


Fig. 63. Panorâmica dos blocos de organização em programação STEP 7 [Siemens]

5.3.1 Endereçamento de memória em PLCs S7

Uma instrução STL é o constituinte de menor dimensão de um programa de utilizador. Define uma tarefa específica para a CPU executar, como p.e. carregar o conteúdo de uma variável no registo ou realizar uma operação matemática.

Tabela 36. Estrutura de uma instrução STL

Etiqueta (“Label”)	Operação	Operando	Comentário
Loop:	L	MB 12	// Armazena no acumulador 1

- *Etiqueta* – é um nome opcional atribuído à instrução, necessário quando se realizam operações do tipo “Loop” ou “Jump”;
- *Operação* – é a tarefa que a CPU tem de executar (p.e. “Load”, “Transfer”, “Add”);
- *Operando* – é a variável que armazena o dado a ser manipulado pela instrução. A variável pode ser um valor, um endereço de memória que contém dados ou pode ser um apontador “pointer” para o endereço de memória que contém os dados.

Existem três métodos de endereçamento de memória em programação STEP 7, utilizados na elaboração do programa de utilizador:

- *Endereçamento Imediato* – refere-se ao uso de um valor numérico como operador;
 - L 14 // Carrega no acumulador 1 o valor inteiro 14;
- *Endereçamento Direto* – refere-se ao uso de memória ou de um objeto de endereçamento para especificar a localização do operando;
 - L MW46// Carrega no acumulador 1 o conteúdo de MW46;
- *Endereçamento Indireto* – refere-se ao uso de um apontador para indicar onde é que o endereço do operador pode ser encontrado;
 - L MW [Localização_do_apontador] // Armazena no acumulador o conteúdo de uma *memory word* (MW), cujo endereço consta num apontador;

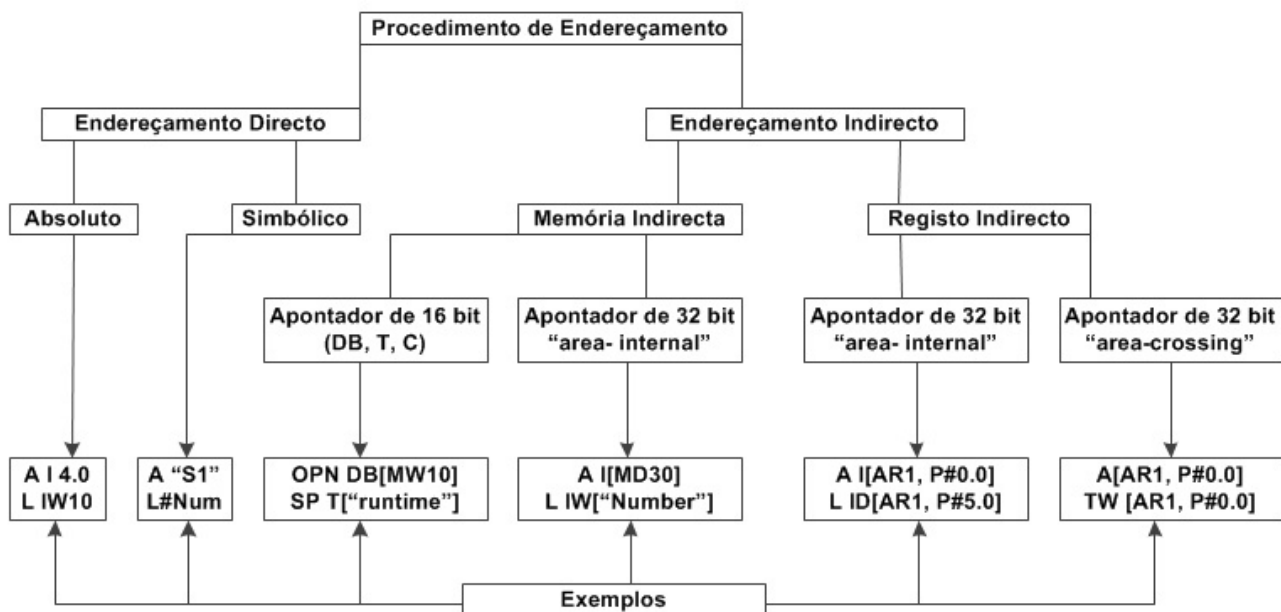


Fig. 64. Diagrama para diferenciação entre endereçamento direto e indireto em programação STEP 7 [26]

Endereçamento indireto em memória

Neste tipo de endereçamento o apontador para o endereço do operando é armazenado na memória da CPU, numa determinada localização. Os apontadores podem ser guardados em “*Marker*” MD, “*Local*” LD, “*Global Data Block*” DBD ou “*Instance Data Block*” DID. É indicada a utilização de endereçamento indireto de memória quando vários apontadores estão a ser utilizados na programação.

Exemplo:

“L MW[DBD12]” – A instrução carrega o acumulador com o conteúdo da MW, cujo endereço está armazenado no DBD12. Se DBB12 contiver o apontador P#122.0, esta instrução irá carregar o conteúdo de MW122 no acumulador.

Endereçamento indireto em registo

Um endereço de registo é usado para formar o efetivo endereço dos dados a armazenar. O apontador do endereço dos referidos dados é armazenado num dos dois registos da CPU que, quando combinado com um determinado valor de *offset*, origina o endereço da variável onde esses dados ficam armazenados. Operações com endereçamentos desta natureza tendem a ser executadas mais rapidamente, do que as operações com os endereçamentos referidos anteriormente.

Exemplo:

“LAR1 P#20.0”

“L MB[AR1,P#0.0] – A instrução pré-carrega o endereço de registo 1 (AR1) da CPU com o apontador P#20.0. De seguida carrega o acumulador com o conteúdo da MB endereçada pela soma de AR1 com o offset P#0.0. Neste caso o acumulador será carregado com o conteúdo de MB20.

Endereçamento indireto de objetos bloco

Alguns objetos bloco são identificados por endereços do tipo inteiro em vez de endereços do tipo byte. Por exemplo:

- “*Global Data Blocks*” DBxxx e “*Instanced Data Blocks*” DIxxx;
- “*Timers*” Txxx e “*Counters*” Cxxx;
- “*Functions*” FCxxx e “*Function Blocks*” FBxxx.

Exemplo:

“OPN DB[MW22]” – A instrução abre o DB cujo endereço (número do DB) está armazenado, como dado do tipo inteiro, em MW22. Se a MW contiver o valor inteiro 126, então abre o DB126.

Apontadores

Apontadores são tipos de dados estruturados para receber um endereço de memória. Podem ser armazenados numa localização de memória ou de registo da CPU. Existem três classes de apontadores em S7:

- “*Area Pointer*” – estrutura de 32 bit que especifica um determinado endereço;
- “*DB Pointer*” – estrutura de 48 bit que combina um “*area pointer*” e um n.º DB de 16 bit;
- “*Any Pointer*” – estrutura de 80 bit, com especificações adicionais para o endereço da variável [26].

5.3.2 Estrutura do programa de utilizador

A base do programa de utilizador consiste na junção e adaptação de num conjunto de blocos desenvolvidos e disponibilizados pela própria Siemens, uns para o módulo FM 350-1 e outros para o módulo FM 353. Estes ainda se podem dividir em blocos com programação essencial para a comunicação entre PLC, módulo FM e programa de utilizador, designados por funções tecnológicas, e blocos com programação direcionada para os diferentes modos de funcionamento dos módulos FM

(exemplos de funcionamento), criados para que se possa aproveitar toda a potencialidade do hardware. Estes blocos, no fim do processo, ficam armazenados na memória da CPU do PLC.

No ponto 5.2 foi abordada a importância dos DBs que armazenam a parametrização dos módulos FM 353, com destaque para o DB-MD (“*Machine Data*”) e o DB-NC (“*Traversing Programs*”). Contudo, há ainda que referir os modos de funcionamento utilizados, alcançados a partir da união destas parametrizações com os blocos de programação do programa de utilizador:

- Posicionamento manual dos eixos “*Jogging*” (1);
- Posicionamento inicial, ou em ponto zero, (0°) “*Reference Point Approach*”(2);
- Posicionamento a partir da introdução da posição desejada “*Manual Data Input*” (6);
- Posicionamento sequencial pré-programado, a partir do DB-NC, “*Automatic*” (8).

A partir destes modos de funcionamentos, após a sua atualização e fusão no programa de utilizador, compreende-se o funcionamento do subsistema principal do projeto:

- Sempre que o utilizador pressiona o botão “*RESET*”, do painel de operação, é utilizado o modo 2 e a célula FV é colocada na sua posição original, paralela à superfície do chão;
- Por defeito está selecionado o modo 1, e o utilizador pode alterar manualmente a posição dos eixos;
- O utilizador inicia o processo de varrimento selecionando o modo 8, quando um dos posicionamentos sequenciais pré-programados, num DB-NC, é acionado;
- Para colocar o sistema na posição de máxima potência registada é utilizado o modo 6. Ao invés do operador introduzir um determinado valor e os eixos girarem para essa posição, após a modificação do código, este direciona a posição registada aquando da máxima potência, para o posicionamento dos eixos a efetuar, neste modo de funcionamento;
- Por forma a deixar o sistema a funcionar automaticamente, executando varrimentos intervalados no tempo, são utilizados alternativamente os modos 8 e 6.

Varrimentos

O sistema secundário foi desenvolvido por forma a dar capacidade ao sistema principal de iniciar um varrimento menor, de entre vários, tornando o processo mais rápido, mas garantido que o varrimento é efetuado onde existe efetivamente mais iluminação. O diagrama que se segue ilustra um varrimento pré-programado. Para escrever a programação dos varrimentos utilizou-se a aplicação de parametrização do módulo FM353 “*Traverse Programs*” (Fig. 65).

A fig. 66 apresenta um detalhe da programação (FC9) para validação de um dos varrimentos pré-programados, selecionado pelo subsistema secundário. Verifica-se o armazenamento do número do respetivo DB numa variável, para posterior utilização (p.e. “*m_byte_db*” – eixo 1).

A fig. 67 representa o processo de atribuição do DB, correspondente ao varrimento selecionado (FC103 – eixo 1). É realizada a transferência do número do DB que contém o G-code, em hexadecimal, para o DB1.DBD242.

```

%1
N1 G90 X310.00 F400.00
N2 G04 X6000

```

% "x" = n.º do programa (DB100x)
 N"x" = n.º da linha do programa
 G90 = movimento em coordenadas absolutas
 G04 = pausa (em ms)
 Quando se segue a G90, X = posição angular (em graus)
 F = velocidade (em graus/min)
 Quando se segue a G04, X = tempo (em ms)

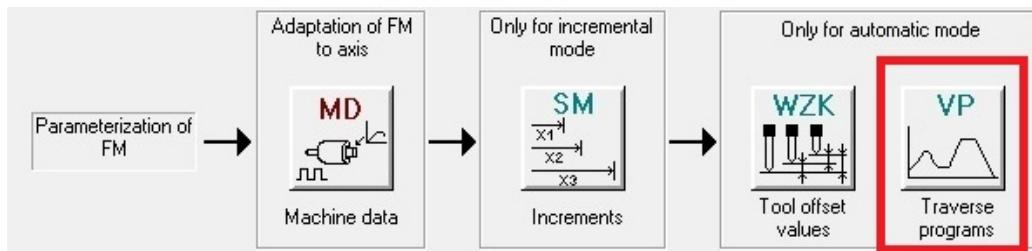


Fig. 65. Criação do G-code

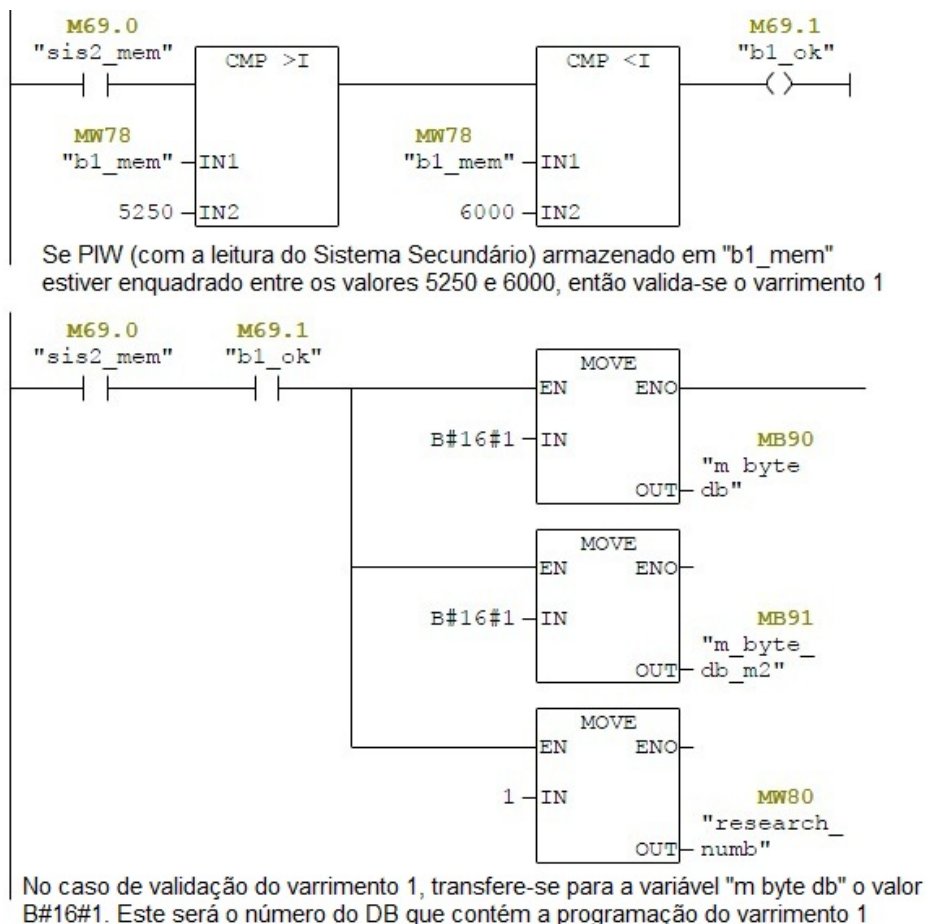


Fig. 66. Detalhe da seleção do DB que corresponde ao varrimento a executar


```
A "DB_UDT_M1".PROGS_D // Transfer program and DB1.DBX49.5 -- part program selection
JC AUST
```

L	"m byte db"		MB90
T	"DB_UDT_M1".PROG_NO	Set request bit	DB1.DBB242 -- part program number
S	"DB_UDT_M1".PROGS_EN	// for program selection	DB1.DBX39.5 -- part program selection
JU	MA01		

Fig. 67. Processo de atribuição do DB com o G-code do varrimento selecionado

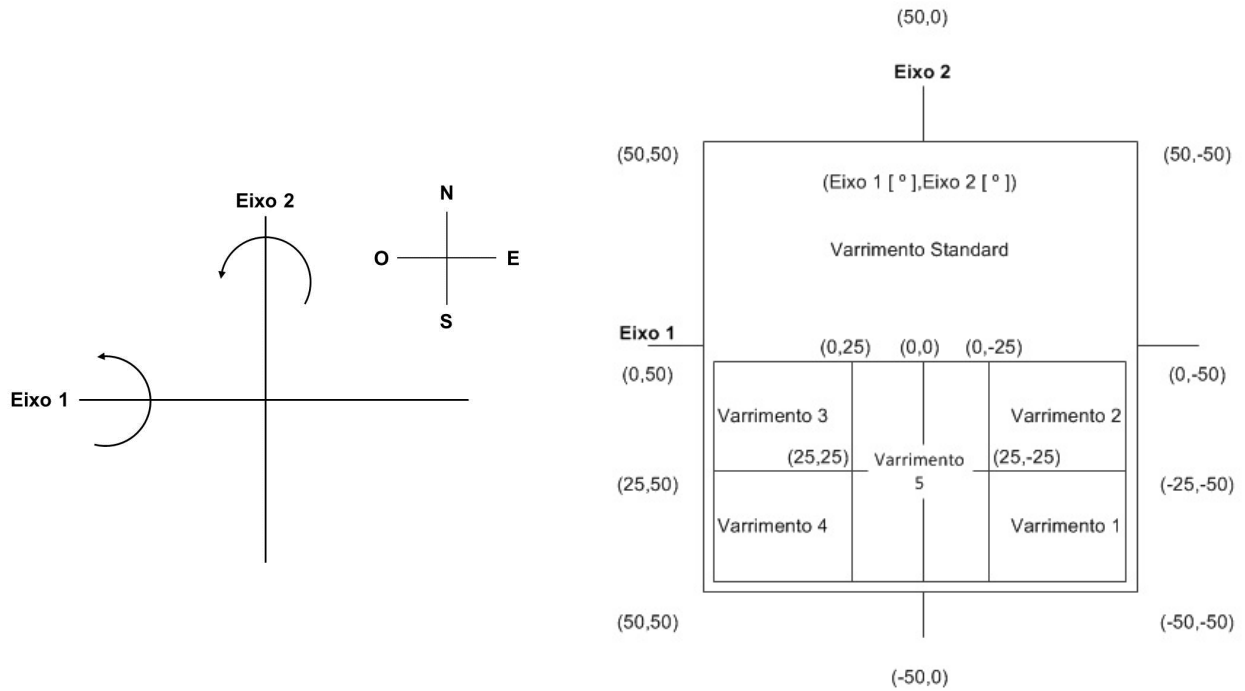


Fig. 68. Sentido dos eixos do sistema principal e representação dos varrimentos pré-programados

Sequência de movimentos do Varrimento 1				
N.º	Eixo 1 (graus)	Eixo 2 (graus)	N.º	
1	-50		1	-50
			2	0
2	-40			
			3	-50
3	-30			
			4	0
4	0			

Fig. 69. Sequência de movimentos de um varrimento pré-programado

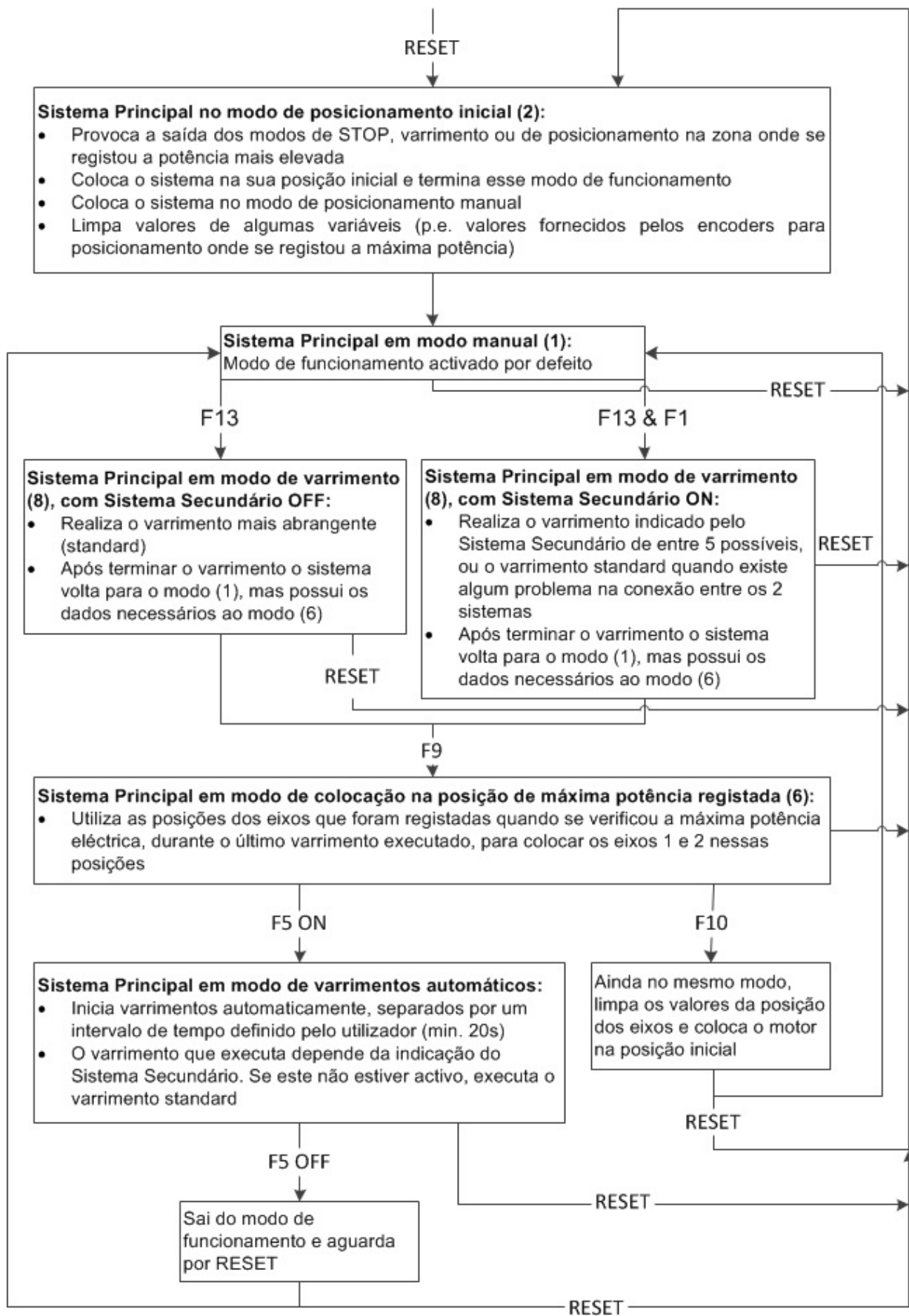


Fig. 70. Organigrama do funcionamento do subsistema principal

Programa de utilizador (CPU)

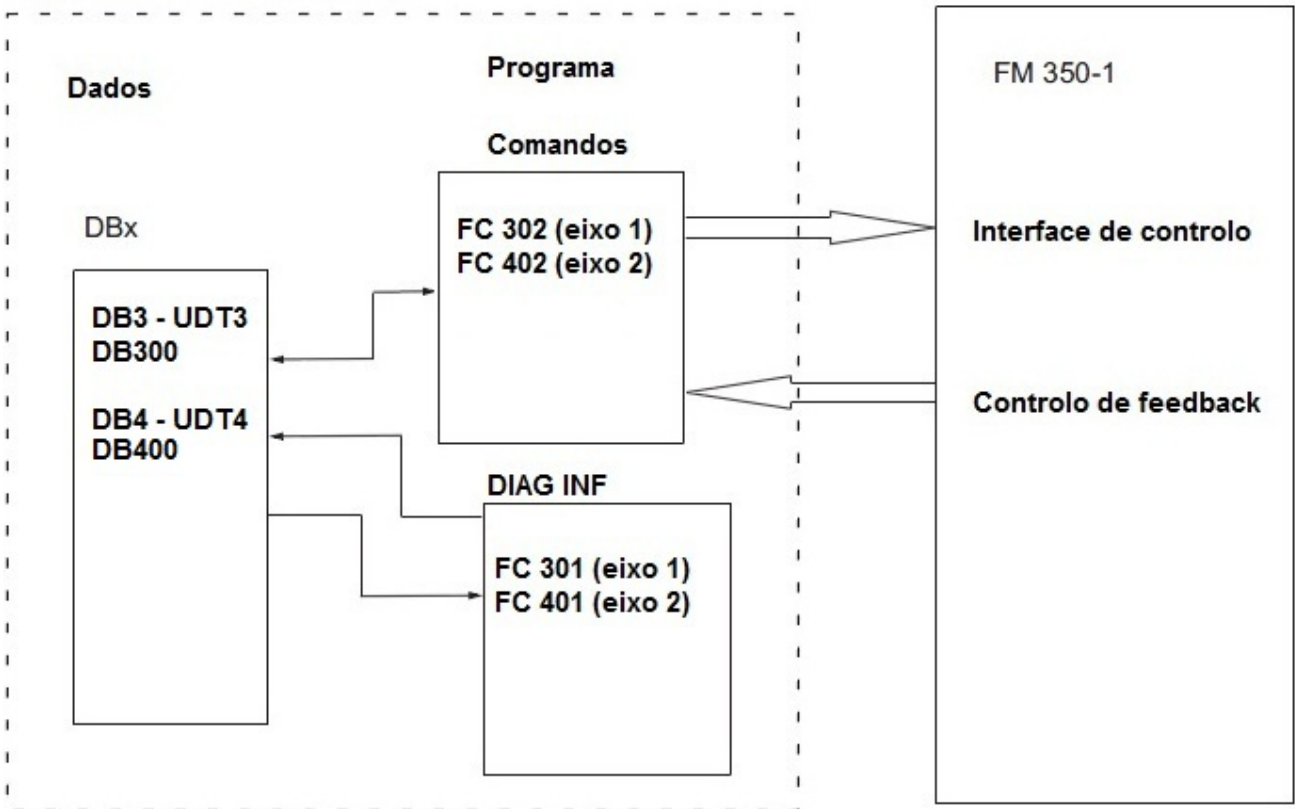


Fig. 71. Diagrama ilustrativo da comunicação entre o programa de utilizador e os módulos FM 350-1 [14]

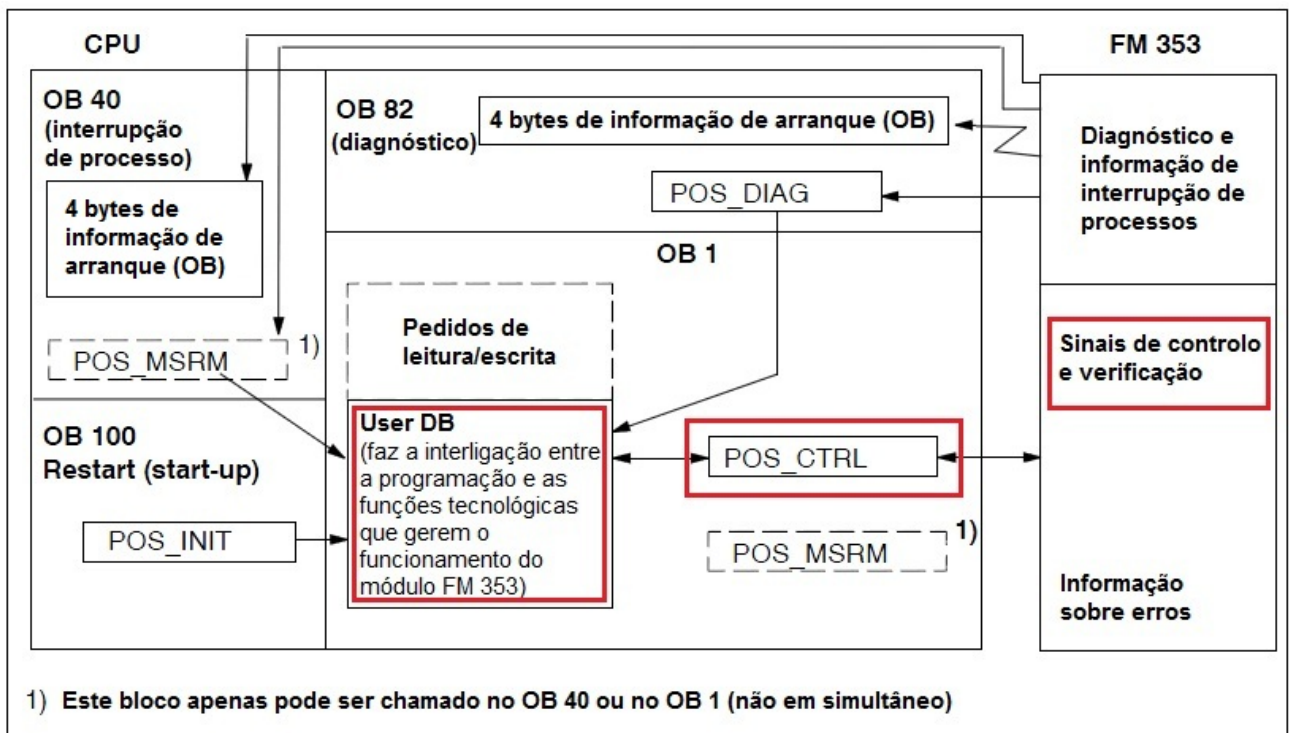
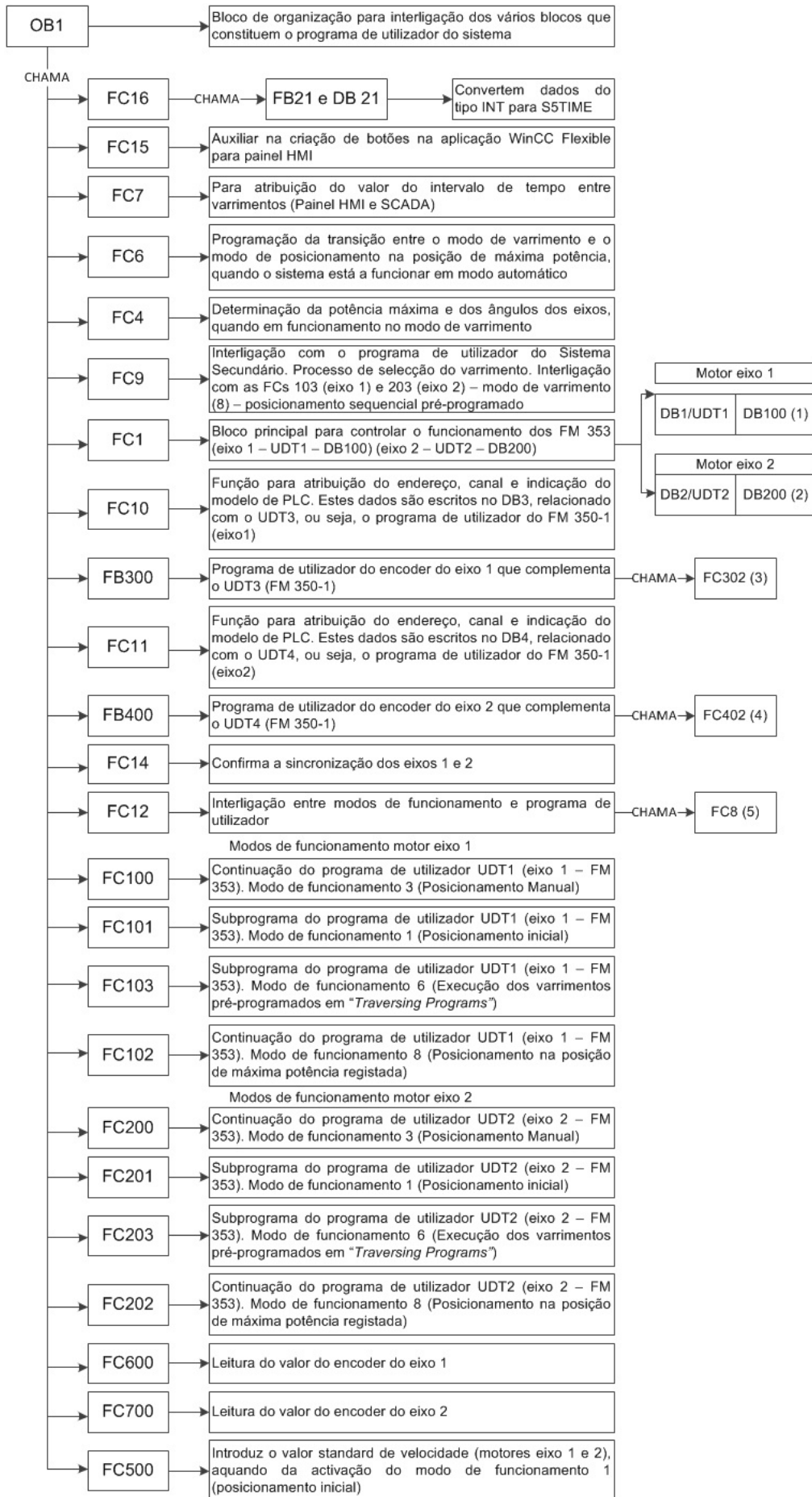


Fig. 72. Diagrama ilustrativo da interligação entre o módulo FM 353 e o programa de utilizador [18]



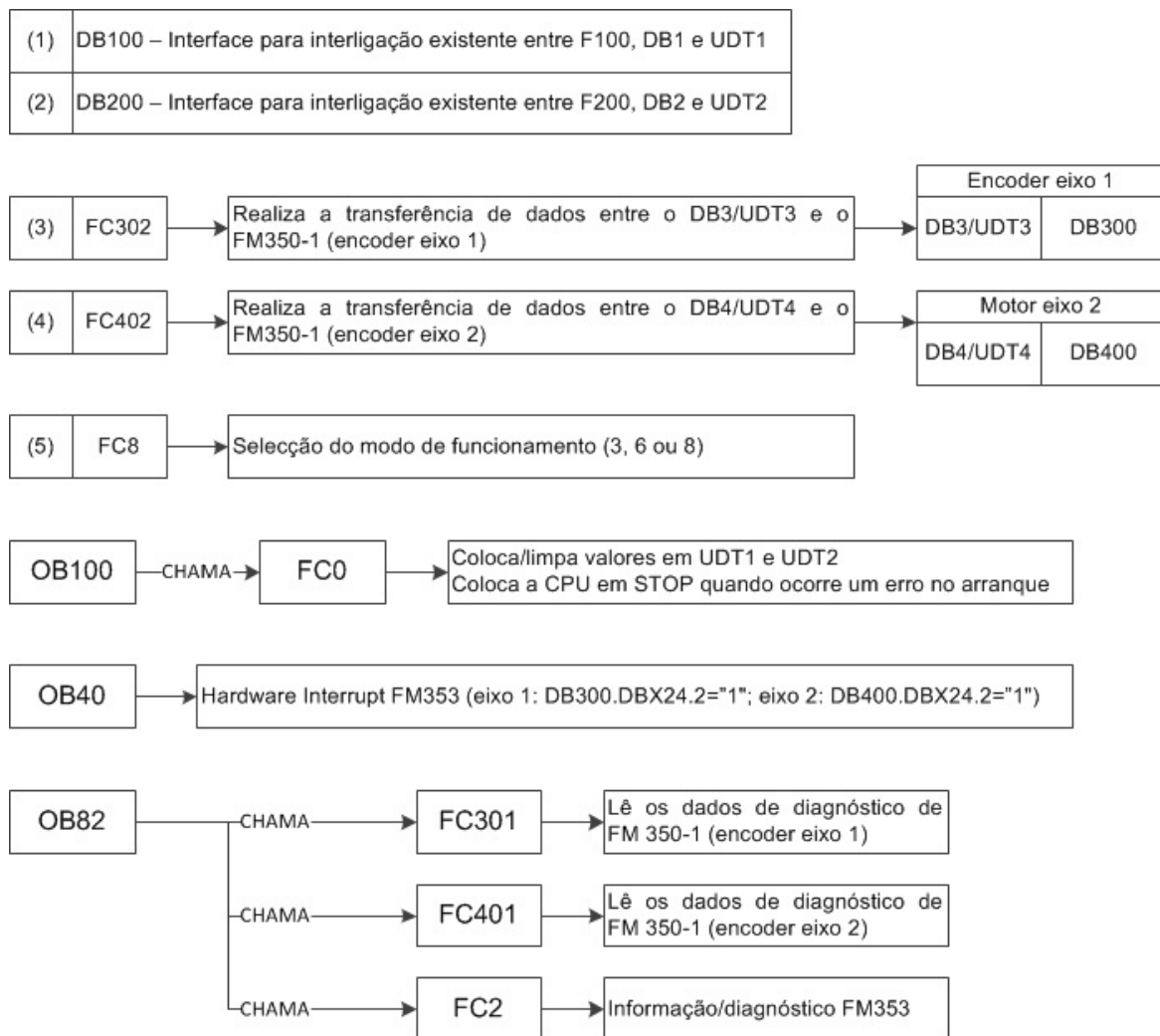


Fig. 73. Organograma de blocos constituintes do programa de utilizador do sistema principal

Dada a importância da função “POS_CTRL” (FC1), entre o conjunto das funções utilizadas/desenvolvidas para o programa de utilizador, faz-se uma breve descrição da função:

- É o bloco principal para controlar o módulo FM353 (leitura e escrita/transferência);
- Controla os modos de operação, por meio de sinais de controlo e checkback (DB1/DB100). Estes são lidos e armazenados no UDT1;
- Transfere os sinais de controlo do UDT1 para o FM 353;
- Interpreta erros e funcionamento.

Capítulo 6 – Descrição da aplicação HMI

As ações do sistema principal são controladas pela CPU 315-2 DP, uma vez que é a partir do controlador que surgem as diversas seqüências de trabalho. Contudo, existem variáveis que podem ser alteradas através do painel de operações OP270.

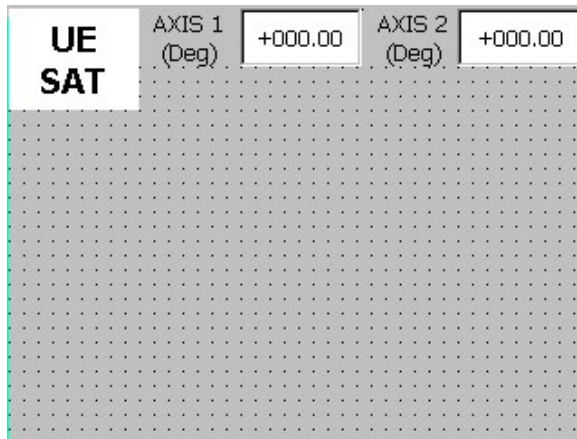


Fig. 74. Tela modelo "TEMPLATE" (1)

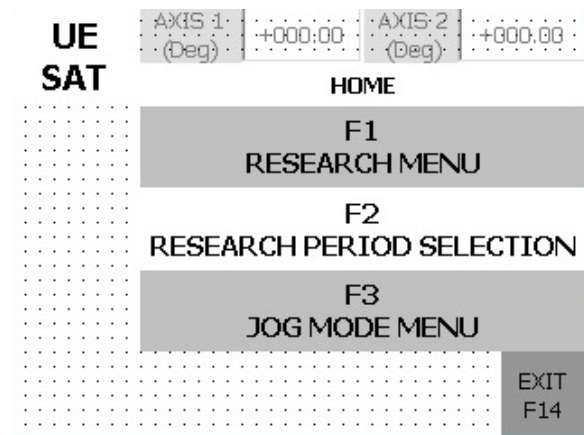


Fig. 75. Tela inicial "HOME" (2)

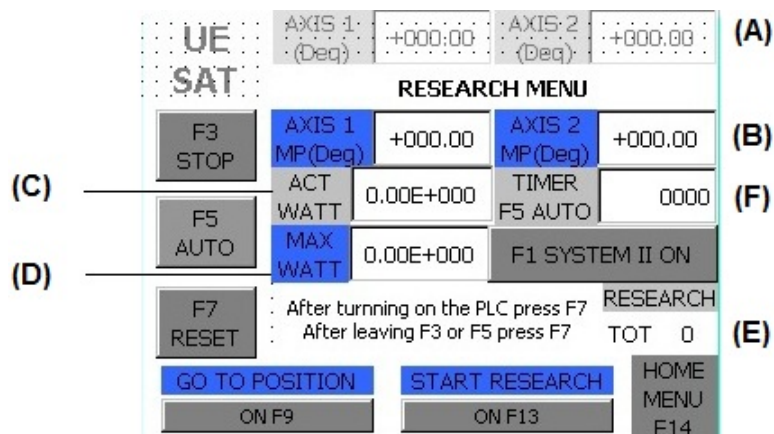


Fig. 76. Tela de varrimentos "RESEARCH MENU" (3)

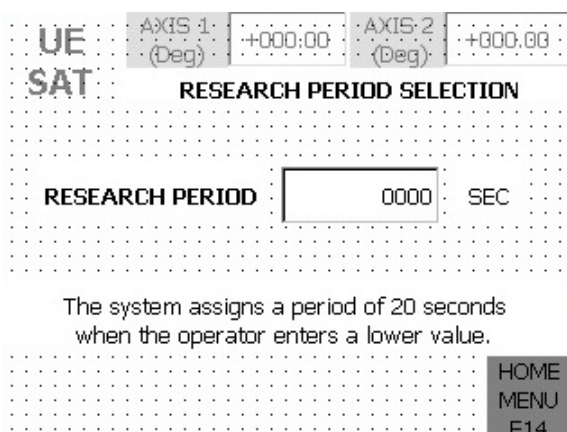


Fig. 77. Tela de seleção do tempo de pausa entre varrimentos "RESEARCH PERIOD SELECTION" (4)

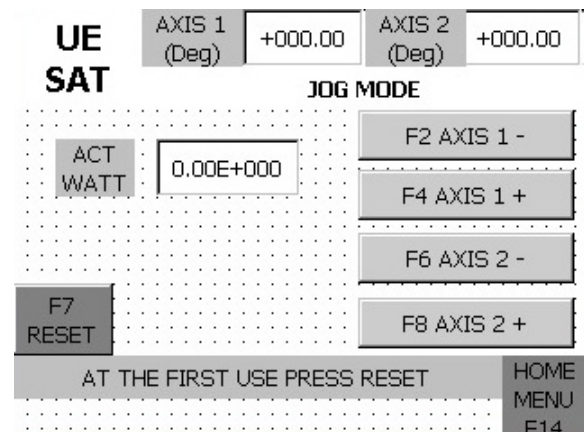


Fig. 78. Painel de comando manual "JOG MODE" (5)

A aplicação para o painel foi desenvolvida em ambiente WinCC Flexible 2008 (SIEMENS).

Tabela 34. Indicação e descrição dos blocos do utilizador

1	<i>TEMPLATE</i>	Os elementos e funcionalidades introduzidos nesta tela surgem automaticamente em todas as outras telas do projeto. Assim, o posicionamento dos eixos motrizes (1 e 2) está disponível no topo de todas as telas.
2	<i>HOME</i>	A partir daqui estão disponíveis os acessos para outras telas (F1 , F2 e F3), bem como a opção de terminar a aplicação (F14).
3	<i>RESEARCH MENU</i>	Disponibiliza os valores de posicionamento atual dos eixos (1 e 2) (A) e o posicionamento aquando do valor de potência máxima registada (B), depois de terminada uma sessão de varrimento. Disponibiliza também a potência elétrica produzida pela célula FV em cada momento (C) e o valor máximo registado no último efetuado (D). A partir do botão F1 o utilizador pode ativar ou desativar o sistema secundário. O botão F3 , quando pressionado, faz parar o sistema. Desprendido o botão F3 , há que colocar o sistema na sua posição inicial. Para tal, deve-se pressionar o botão F7 . Estando o sistema secundário ativo, é possível observar qual o varrimento selecionado (E). O botão F14 disponibiliza o menu principal. Estando o sistema na sua posição inicial (eixo 1 e eixo 2 em 0°), ao pressionar o botão F13 , este começa a executar um varrimento. Quando este chega ao fim, o utilizador deve pressionar o botão F9 para que o sistema se desloque para a posição onde se registou a potência elétrica mais elevada. Nestas condições, ao pressionar o botão F5 , o sistema entra no modo de varrimentos automáticos. O tempo que falta para que se inicie o varrimento seguinte também está disponível num campo de saída (F). Para sair deste modo, solta-se o botão F5 e pressiona-se F7 . O botão F10 tem a função de colocar o sistema na posição inicial, quando este previamente estava estacionado na posição de máxima potência registada (F9), sem fazer “Reset” ao sistema.
4	<i>SELECT PERIOD RESEARCH</i>	O utilizador pode introduzir o tempo de intervalo entre varrimentos. Notar que, o tempo mínimo entre varrimentos está definido em 20 s.
5	<i>JOG MODE</i>	Estão disponíveis quatro botões para posicionar manualmente os dois eixos. F2 : para deslocar o eixo 1 no sentido negativo; F4 : para deslocar o eixo 1 no sentido positivo. F6 : para deslocar o eixo 2 no sentido negativo; F8 : para deslocar o eixo 2 no sentido positivo.

Principais características do sistema de interface homem máquina Simatic HMI OP 270:

- Tensão fornecida: 24V DC;
- Memória: Flash/RAM – 2048 KB utilizáveis para ficheiros do utilizador;
- Ferramenta de configuração: ProTool versão 6 SP3 ou superior, WinCC Flexible Standard Versão 2004 ou superior;
- Display: tipo STN, tamanho 5,7”, resolução 320 x 240, a cores;
- Modo de operação: 24 teclas de função, 18 teclas (com LED) programáveis;

- Interfaces: 2 x RS232, 1 x RS422, 1 x RS485 (máximo 12Mbit/s), 1 x USB;
- Sistema operativo: Windows CE, [27].

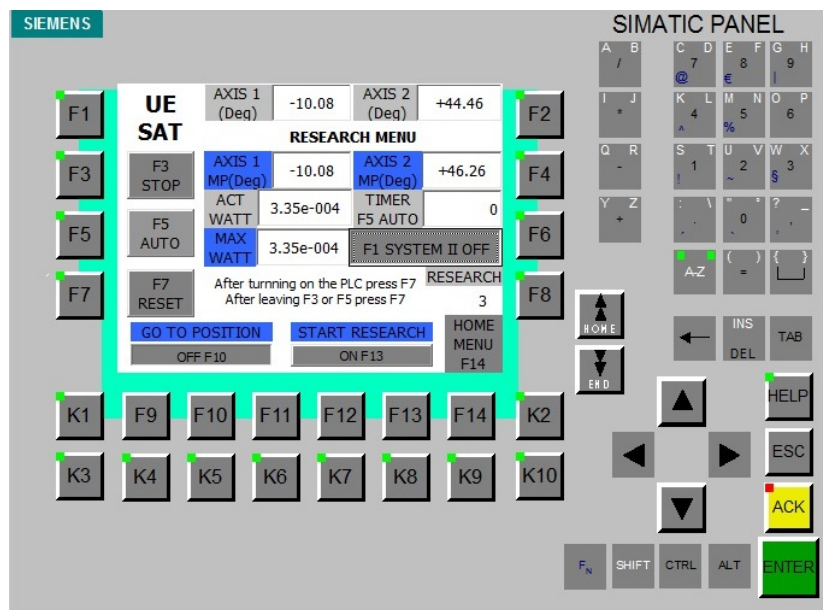


Fig. 79. Painel HMI OP 270 (Siemens)

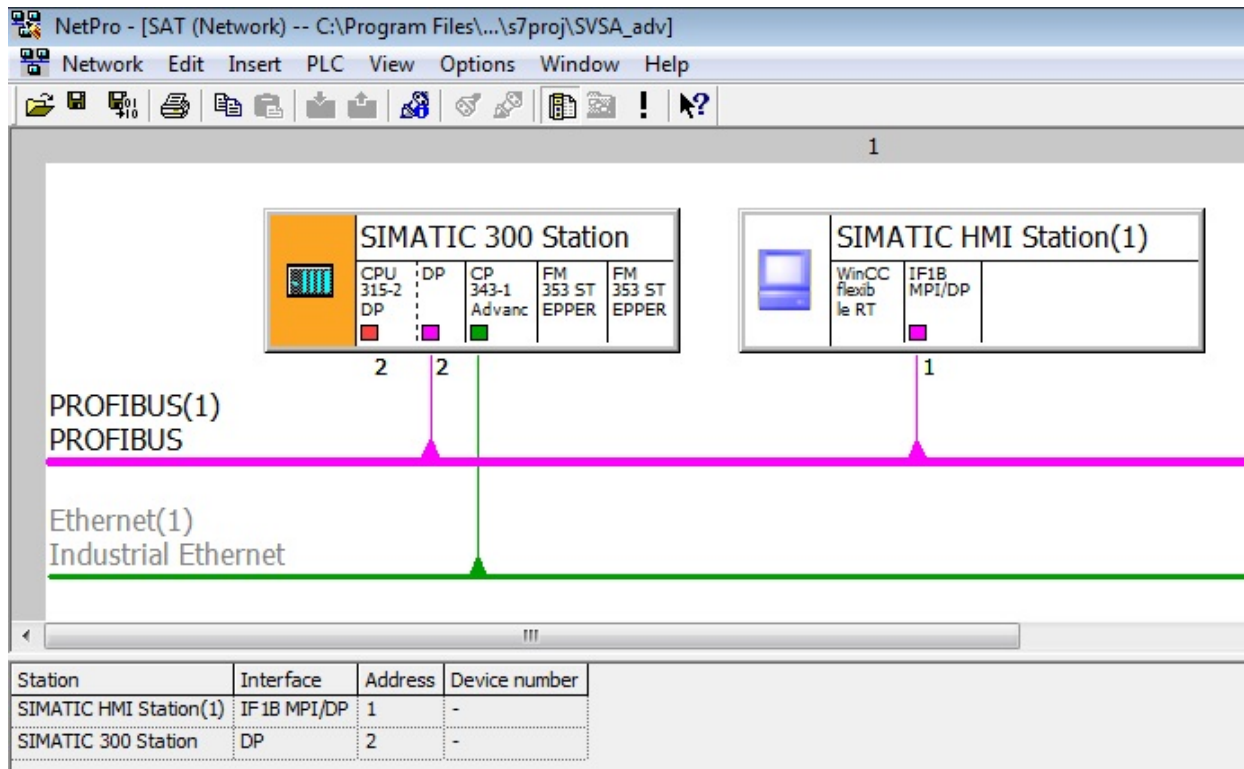


Fig. 80. Aplicação “NetPro” do STEP 7, com destaque para comunicação entre PLC e painel HMI

A comunicação entre o OP e o PLC é suportada por meio da ligação entre a interface do PLC (DP) e a interface IF1B do OP270 (RS 422/485), após prévia configuração da conexão.

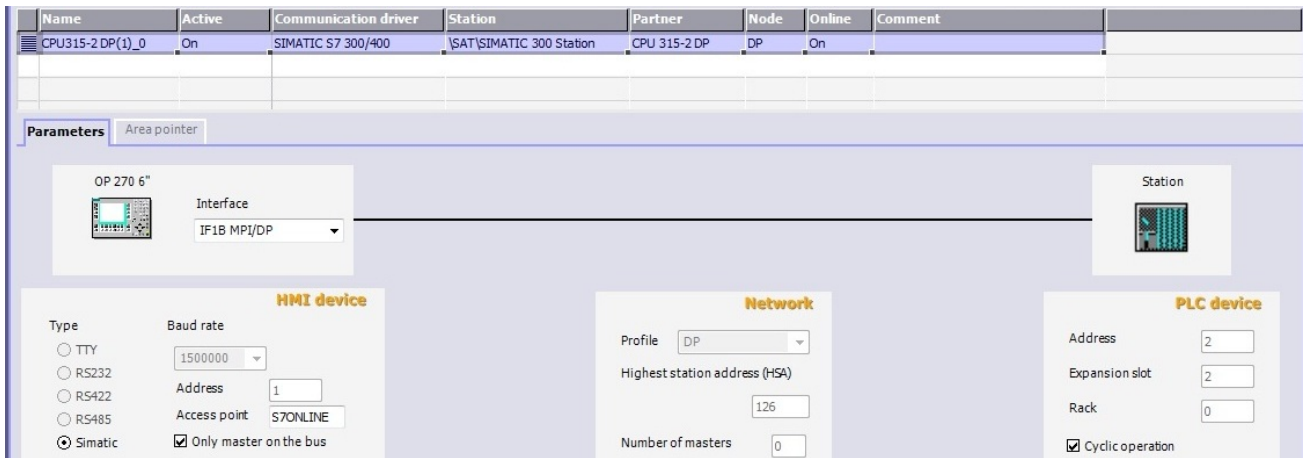


Fig. 81. Tipo de conexão estabelecida entre o painel HMI e o controlador S7-300 [28]

A interface IF1B (RS 422/485) configura-se por meio de um switch, existente nesta gama de OPs. Por defeito, o OP está configurado para a comunicação com controladores SIMATIC S7 [27].

Configurações de Comunicação	Configuração do Switch
	<p>Configuração por defeito</p>

Fig. 82. Tabela com as configurações possíveis de estabelecer entre o OP e um PLC, associadas à configuração do switch da interface IF1B [27]

Capítulo 7 – Descrição do Sistema de Supervisão e Controle

Atualmente verificam-se progressos, em várias áreas da automação, no sentido na integração total dos sistemas que constituem um complexo industrial/empresarial. Tal acontece porque nos últimos anos verifica-se a tendência de substituição de sistemas centralizados, por sistemas distribuídos de arquitetura modular e expansíveis, de fácil instalação e manutenção, viáveis pela possibilidade de comunicação entre os vários integrantes.

A crescente necessidade de comunicação entre os sistemas de automação e entre estes e o nível de planeamento/gestão industrial impulsiona os desenvolvimentos que se verificam no campo das tecnologias em redes de comunicação. Define-se rede de comunicação industrial como o conjunto de hardware e software utilizado para disponibilizar o trânsito de informações entre os diversos níveis hierárquicos participantes de um processo industrial.

As estruturas empresariais procuram a integração da informação dos sistemas industriais desde a produção até à gestão, interligando os níveis de Operação, Manutenção, Supervisão e Gestão. O objetivo é alcançar um melhor controlo dos processos, mais qualidade, eficiência e segurança. Para tal, contribuem as arquiteturas distribuídas, devidos aos progressivos avanços dos mecanismos de comunicação.

Existem diferentes níveis num sistema industrial genérico que se distinguem através dos tipos de transações (horizontais/verticais) que ocorrem e dos diferentes requisitos temporais, mais exigentes ao nível da comunicação com sensores e actuadores.

Esta forma de organização estimula a coexistência de múltiplos fluxos de informação, dada a necessidade de compatibilizar diferentes requisitos em termos de quantidade de informação a transferir e frequência/pontualidade das transações. Assim, surgem vários protocolos com qualidade de serviço adequada para cada nível: *Ethernet (type-II, 802.3, PROFINET, EtherCAT, Ethernet/IP); Foundation Fieldbus; PROFIBUS (FMS, DP, PA), CAN (DeviceNET, CANopen, CANkingdom, TT-CAN...); modbus; AS-Interface, etc...* Alguns fabricantes fornecem soluções integradas: *Totally Integrated Automation (TIA) - Siemens; Control Information Protocol (CIP) – RockWell* [29].

De forma a garantir uma maior segurança de operação e uma diminuição dos custos do processo, os SCADA permitem a visualização de eventos, alarmes, mensagens e gráficos, associados a um elevado nível de segurança. Desta forma, os seus componentes são projetados e construídos de forma a resistirem em ambientes agressivos (temperatura, corrosão, interferências eletromagnéticas, vibrações mecânicas, p.e.), [30].

Nos dias de hoje caminha-se para a normalização entre componentes de sistemas de supervisão e controlo, contudo os primeiros avanços deram-se ao nível dos fabricantes, pelo que cada um concebeu a sua estrutura de redes, [29].

Por forma a comandar e monitorizar à distância o funcionamento do sistema objeto desta dissertação, criou-se uma aplicação *SCADA Web Enable*, implementada em ambiente WinCC V7.0 (Siemens®), integrada com o controlador através de uma ligação Ethernet (TCP/IP). Para além das funções disponíveis no sistema local de interface homem-máquina, este possui duas funcionalidades adicionais: visualização gráfica das potências (atual e máxima) no tempo e emissão de relatório de operação com o referido gráfico.

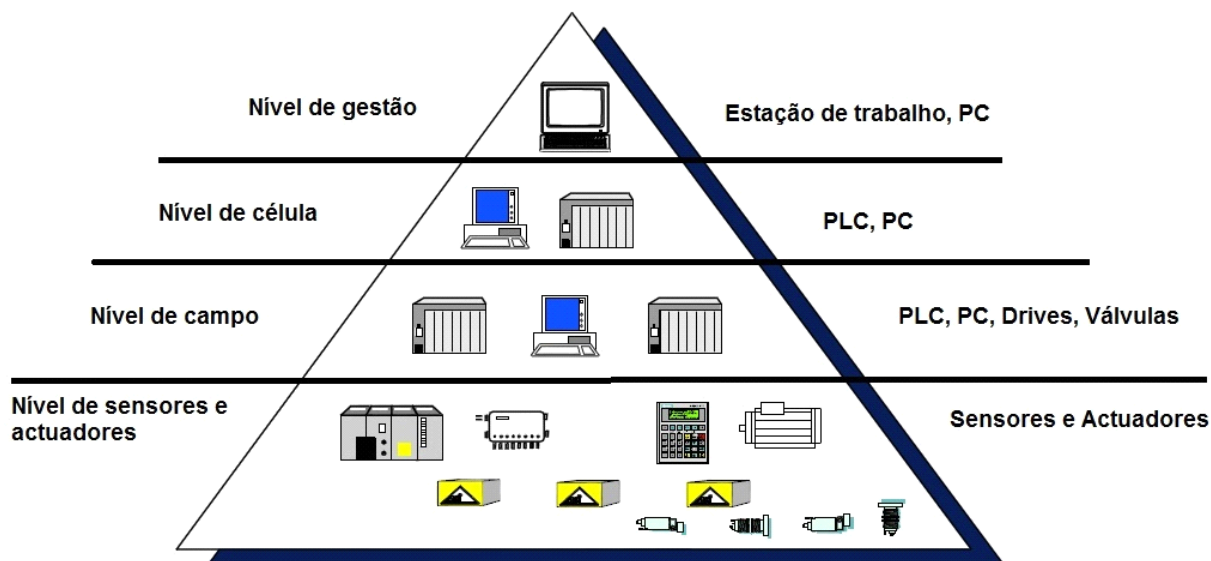


Fig. 83. Pirâmide da hierarquia entre os diferentes níveis de automação numa estrutura industrial [Siemens]

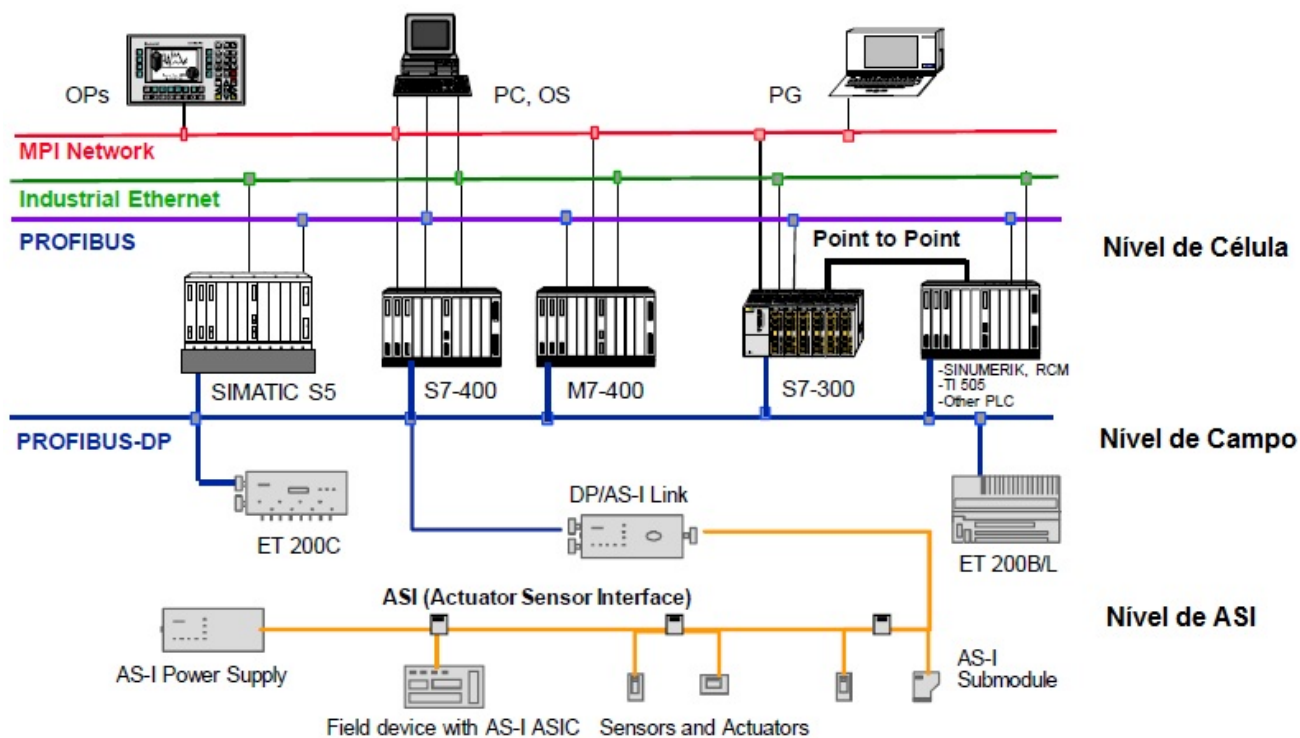


Fig. 84. Sub-redes em sistemas SIMATIC [Siemens]

A interface gráfica de utilização amigável SCADA implementada assenta num conjunto de telas que apresentam informações provenientes do controlador e opções de comando sobre o programa de utilizador nele em execução, através de campos de entrada/saída dos valores das

variáveis a controlar, botões/ícones de seleção dos modos de funcionamento do sistema e botões de navegação entre telas.

Tratando-se de um protótipo implementado em laboratório e considerando as suas características técnicas, em termos de segurança, criou-se apenas um utilizador do sistema (*Log In* e *Palavra-Passe*). Para além disso, limitou-se à visualização de variáveis e à navegação entre telas, no modo de acesso remoto, não sendo possível o comando do sistema.

7.1 Configuração da comunicação (SCADA – PLC)

O canal "*SIMATIC S7 Protocol Suite*" é usado para conectar os controladores SIMATIC S7-300 aos sistemas desenvolvidos em ambiente WinCC. Este suporta um vasto leque de protocolos e tipos de redes de comunicação. No caso da aplicação desenvolvida, utiliza-se este canal para estabelecer a comunicação com uma rede ethernet por TCP/IP.

Seguem-se os passos para configuração do canal "*SIMATIC S7 Protocol Suite*":

1. Criar um projeto WinCC do tipo "*Server, Single User*";
2. Garantir que os nomes do PC e do Servidor WinCC são iguais;
3. Instalar o canal ("*SIMATIC S7 Protocol Suite*");
 - i. Fazer "*Add New Driver*" no atalho para o menu "*Tag Management*";
 - ii. Selecionar "*SIMATIC S7 Protocol Suite.chn*";
4. Selecionar a unidade do canal: "*TCP/IP channel unit*";
5. Criar uma conexão nova;
 - i. Selecionar "*New Connection*";
 - ii. Introduzir o nome da conexão;
6. Realizar a configuração das "*tags*", ou seja, endereços de memória das variáveis do programa de utilizador modificadas/visualizadas no sistema SCADA (ver fig.87);
 - i. Definir nomes e tipos de "*tags*";
 - ii. Selecionar as propriedades de cada "*tag*", de acordo com o tipo de dado da variável que representam.

A unidade "TCP/IP" do canal "*SIMATIC S7 Protocol Suite*", foi usada para estabelecer a comunicação entre o sistema desenvolvido em WinCC com a programação executada no controlador S7-300, via Ethernet Industrial por protocolo "*ISO-on-TCP Transport*". Dado que o controlador utilizado é da família S7-300, a comunicação estabelece-se através de um módulo de comunicação, neste caso: CP 343 Advanced-IT.

Dado que a comunicação entre a aplicação SCADA e o PLC está estabelecida por TPC/IP, há que definir a interface de comunicação da estação que corre o sistema WinCC. Para isso, a partir do Painel de Controlo do Windows, na aplicação "*Set PG/PC Interface*", seleciona-se para "*Access Point of Application*": S7ONLINE (STEP 7); TCP/IP; Placa de Rede disponível, [31].

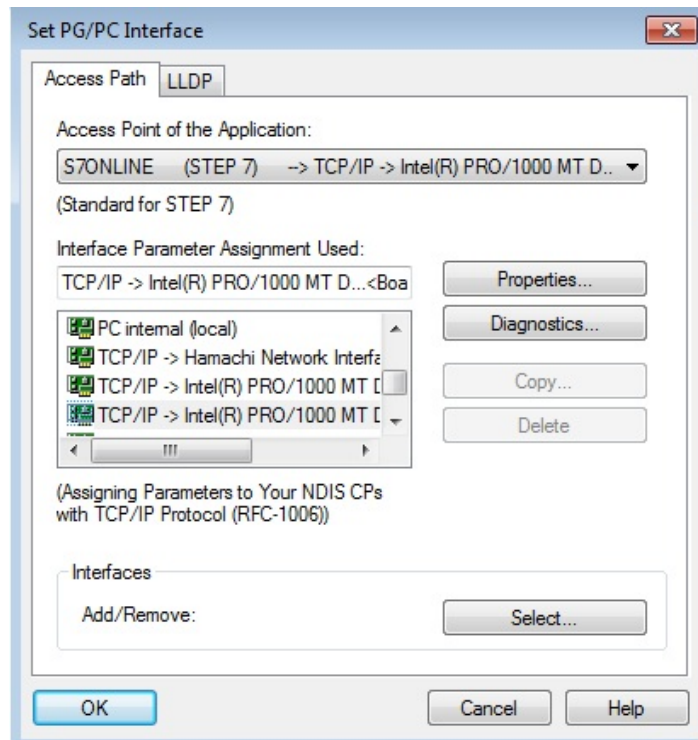


Fig. 85. Definição da interface PG/PC: STEP 7 TCP/IP

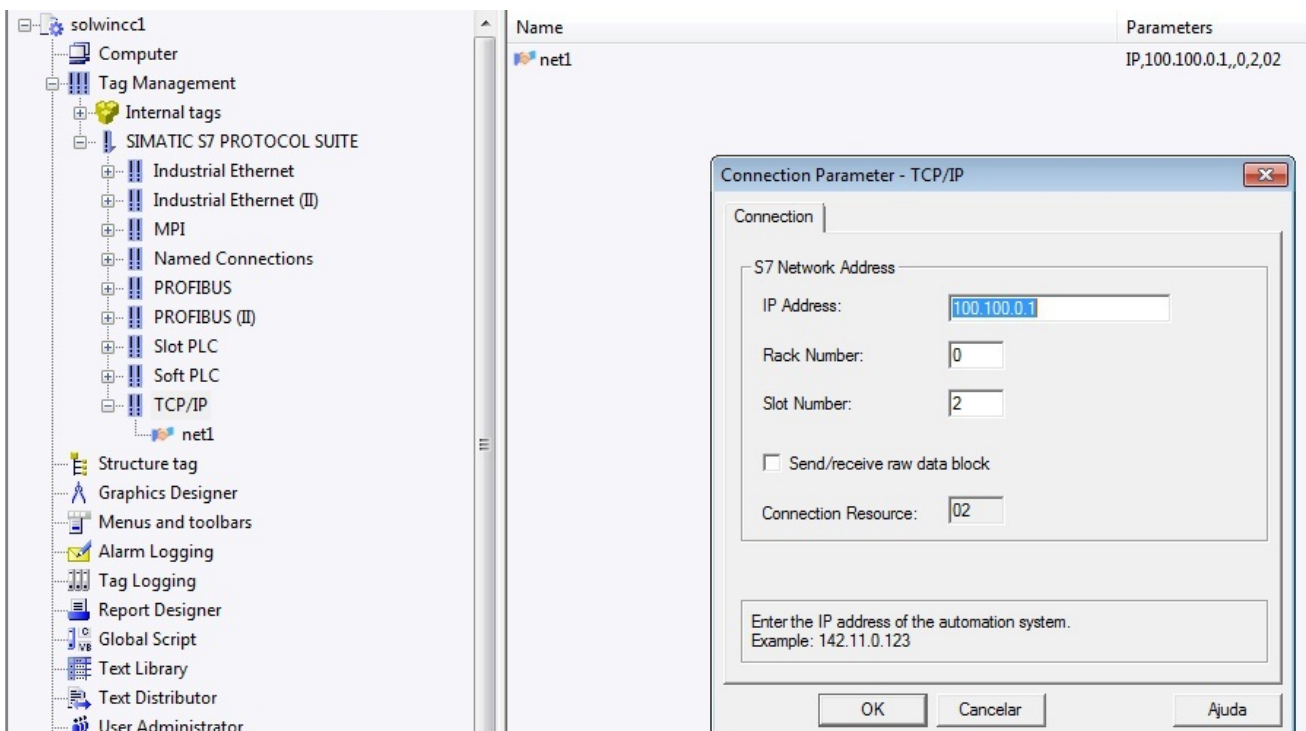


Fig. 86. Atribuição do IP da rede Ethernet TCP/IP

Name	Type	Parameters
auto_go_wcc	Binary Tag	M250.0
auto_pulse_wcc	Binary Tag	M245.0
b1_ok_wcc	Binary Tag	M69.1
b2_ok_wcc	Binary Tag	M69.2
b3_ok_wcc	Binary Tag	M69.3
b4_ok_wcc	Binary Tag	M69.4
b5_ok_wcc	Binary Tag	M69.5
b7_ok_wcc	Binary Tag	M69.7
bi_mem_wcc	Unsigned 16-bit value	MW78
counter_v11_wcc	Signed 32-bit value	MD158
counter_v22_wcc	Signed 32-bit value	MD166
counter_value_m2_wcc	Signed 32-bit value	MD48
counter_value_wcc	Signed 32-bit value	MD44
go_zero_busca_wcc	Binary Tag	M250.1
mem_auto_wcc	Binary Tag	M248.1
mem_research_wcc	Binary Tag	M248.0
mf9_wcc	Binary Tag	M8.7
mw_volt_wcc	Unsigned 16-bit value	MW100
pot_wcc	Floating-point number 32-bit IEEE 754	MD35
potact3_wcc	Floating-point number 32-bit IEEE 754	MD65
research_mode_wcc	Binary Tag	M26.0
research_numb_wcc	Unsigned 16-bit value	MW80
reset_m_wcc	Binary Tag	M7.1
sis2_mem_wcc	Binary Tag	M69.0
stop_m_wcc	Binary Tag	M7.0
t_hmi_wcc	Unsigned 16-bit value	MW128
t_in_wcc	Unsigned 16-bit value	MW120
time_auto_wcc	Binary Tag	M6.4
time_pulse_wcc	Binary Tag	M6.7
travel_m_m1_wcc	Binary Tag	M7.2
travel_m_m2_wcc	Binary Tag	M7.4
travel_p_m1_wcc	Binary Tag	M7.3
travel_p_m2_wcc	Binary Tag	M7.5
zero_pulse_wcc	Binary Tag	M245.1

Fig. 87. Lista completa das “tags” criadas para interação com o controlador

7.2 Descrição da aplicação SCADA

Após completar a configuração da comunicação entre WinCC e PLC e introduzir todas as “tags”, segue-se a criação das telas que compõem a interface amigável SCADA.

No software WinCC, recorre-se à aplicação “*Graphics Designer*” para criar as diferentes telas e o seu conteúdo, ou seja, botões para transição entre telas, campos para visualização de valores das variáveis do projeto, campos para introdução/modificação de valores, entre outros objetos que possam auxiliar no entendimento do funcionamento do sistema, como conjunto de luzes, textos informativos, etc...

Existe a possibilidade de tornar a aplicação SCADA o superior hierárquico do sistema local. Significa isto que, o utilizador pode remotamente atribuir ou cancelar as funcionalidades do painel HMI. Para tal, há que introduzir na programação variáveis auxiliares como as mesmas funções das variáveis criadas para controlar o sistema localmente, mas modificáveis apenas a partir da aplicação SCADA. Por fim, há que criar um botão, numa tela do projeto WinCC, associado a um *memory bit* com a função específica de anular as operações no sistema, executadas a partir do painel. Essa

variável é introduzida em todas as linhas de código criadas para controlo do sistema a partir do painel HMI, como um “contacto normalmente fechado”, que ao tomar valor lógico “1” torna qualquer condição lógica falsa.

Por forma a não aumentar o código de programação e desta forma tornar mais fácil a análise do programa de utilizador em trabalhos futuros, os comandos local e remoto estão no mesmo patamar hierárquico.

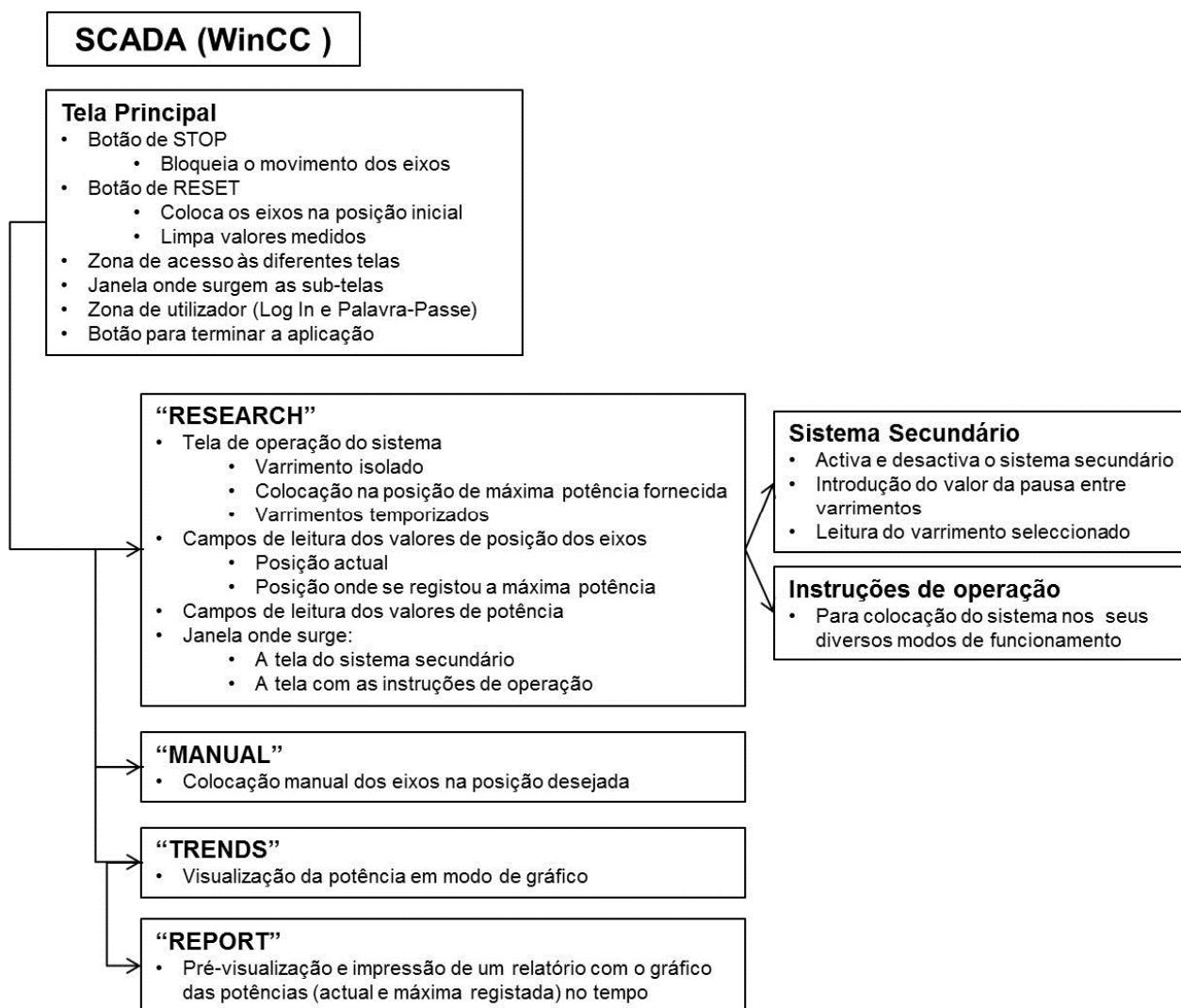


Fig. 88. Diagrama com as diferentes telas/ecrãs da aplicação SCADA

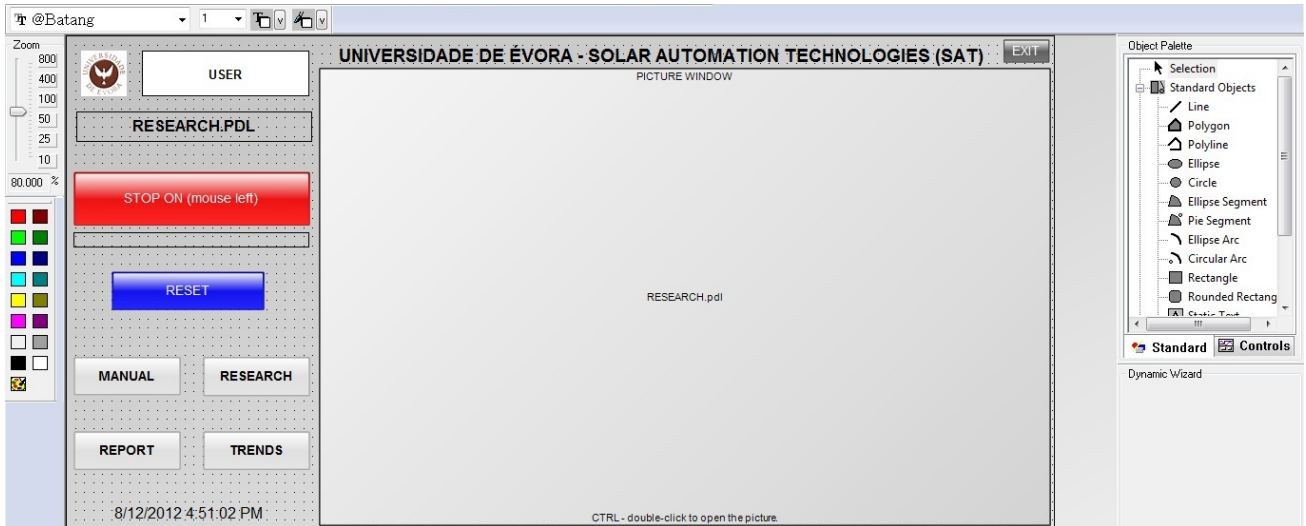


Fig. 89. Desenvolvimento da tela principal, na aplicação “Graphics Designer”. Tela de sistema secundário “*PICTURE WINDOW*” para telas do sistema principal

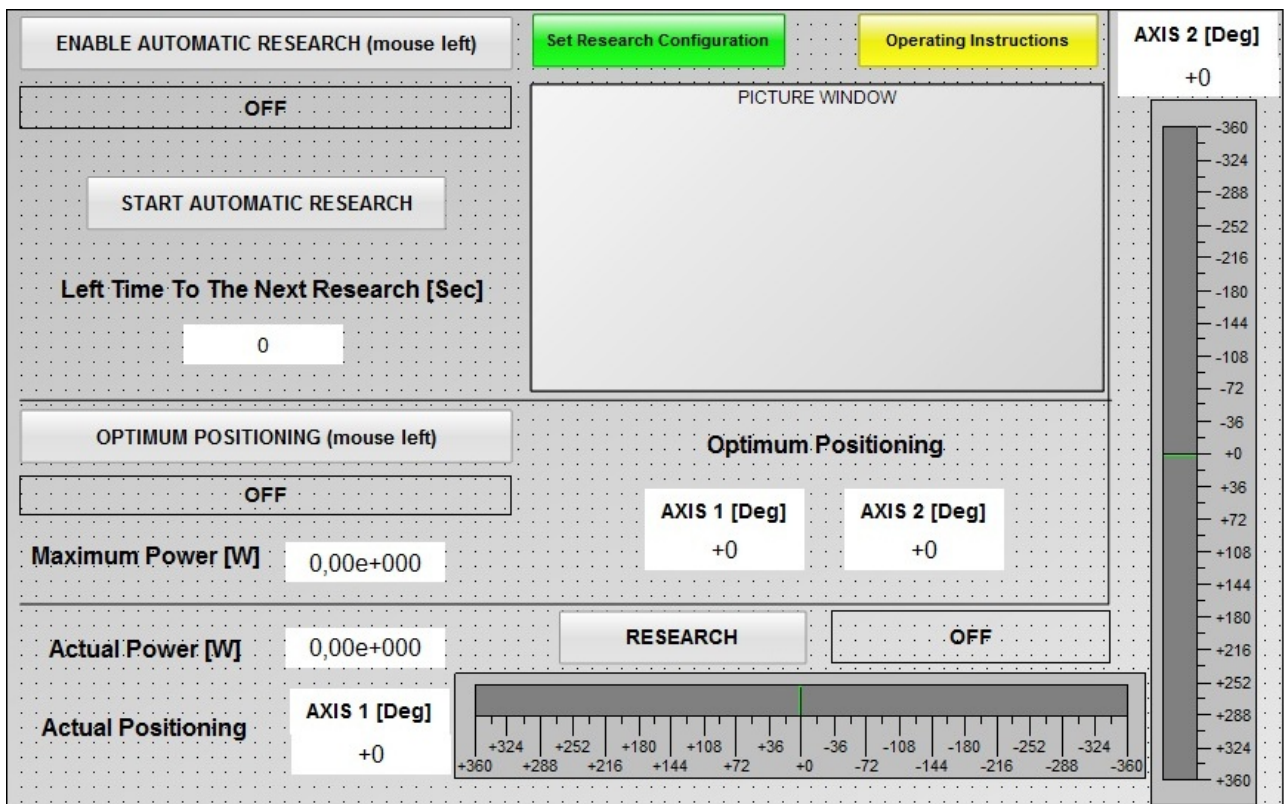


Fig. 90. Tela com instruções de operação (“*RESEARCH*”) e “*PICTURE WINDOW*” para tela do sistema secundário

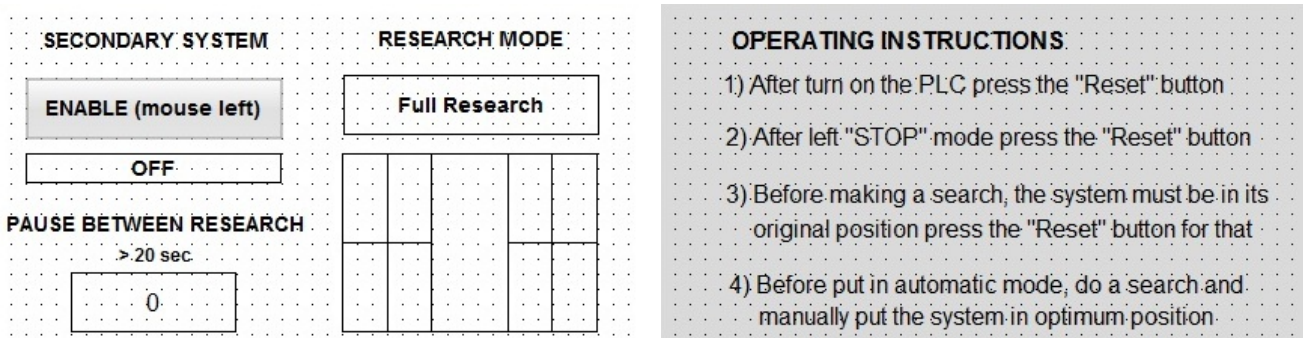


Fig. 91. Tela do subsistema secundário e tela com instruções de operação

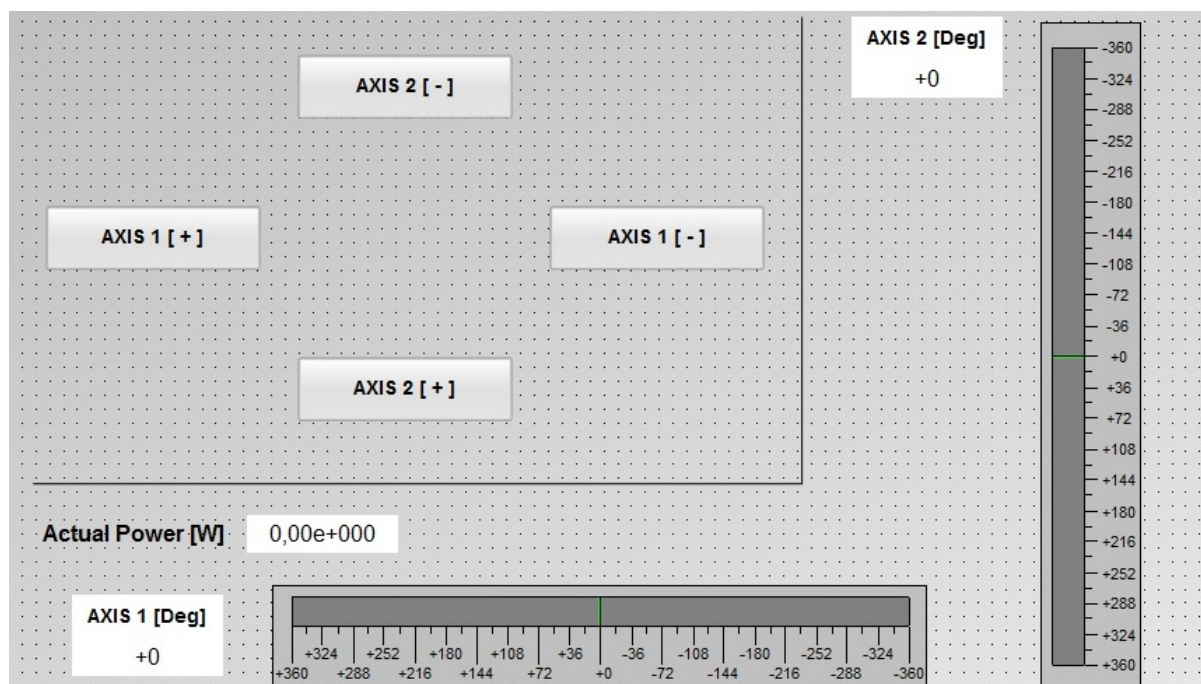


Fig. 92. Tela para posicionamento manual dos eixos do sistema principal

A descrição passo-a-passo da criação da aplicação SCADA está disponível em anexo. Este contém informação sobre a criação do modo de visualização gráfica das potências (atual e máxima registada durante a execução do último varrimento). O mesmo contém as instruções para criação do portal Web para monitorização do sistema, de acesso a partir de uma rede local.

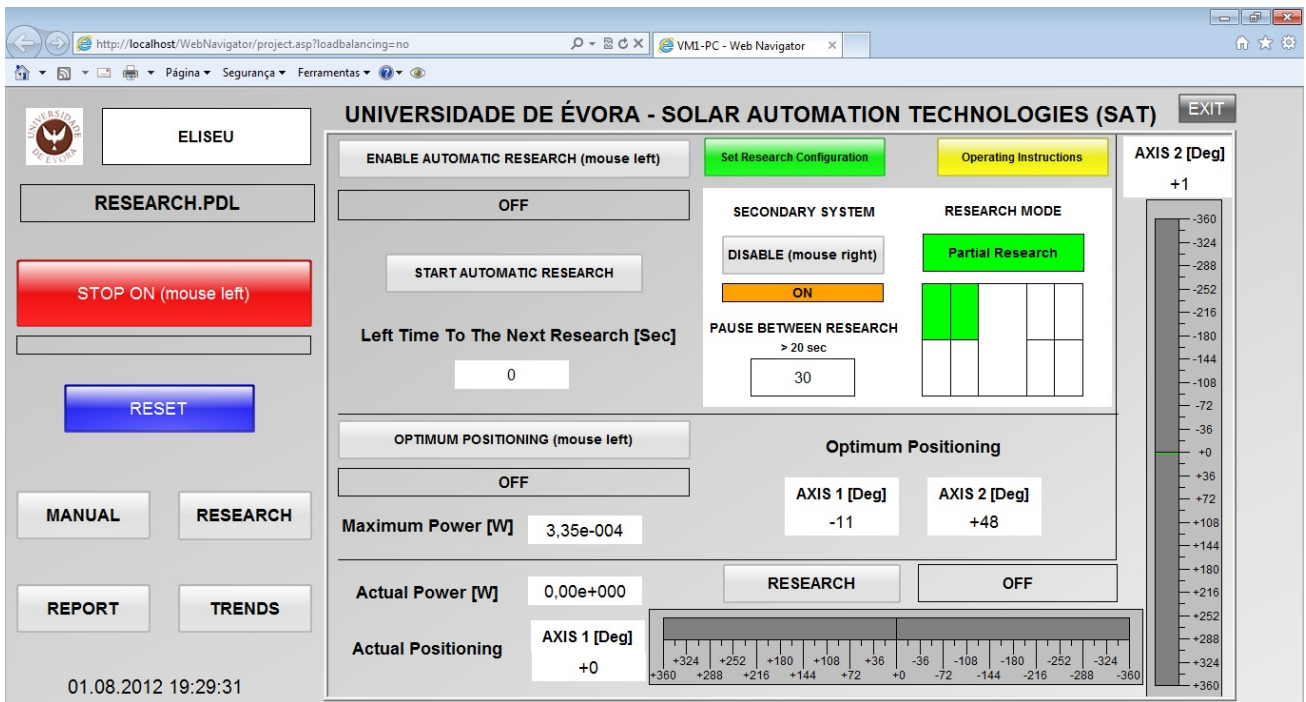


Fig. 93. Portal Web para monitorização remota do sistema

Capítulo 8 - Resultados

O programa de utilizador desenvolvido funciona corretamente, pelo que foi possível realizar os ensaios apresentados neste capítulo. A sua análise comprova que foram alcançados os desafios propostos no projeto de dissertação.

O projeto foi implementado em laboratório tendo sido utilizada uma lâmpada, móvel de halogénio (150 W) como fonte luminosa. Na saída do painel foi colocada uma resistência de 0.1Ω em paralelo, como carga. Assim, se explica a gama de valores de potência elétrica obtidos.

Foram realizados vários ensaios para recolha de dados de posicionamento e potência elétrica.

Ensaio 1

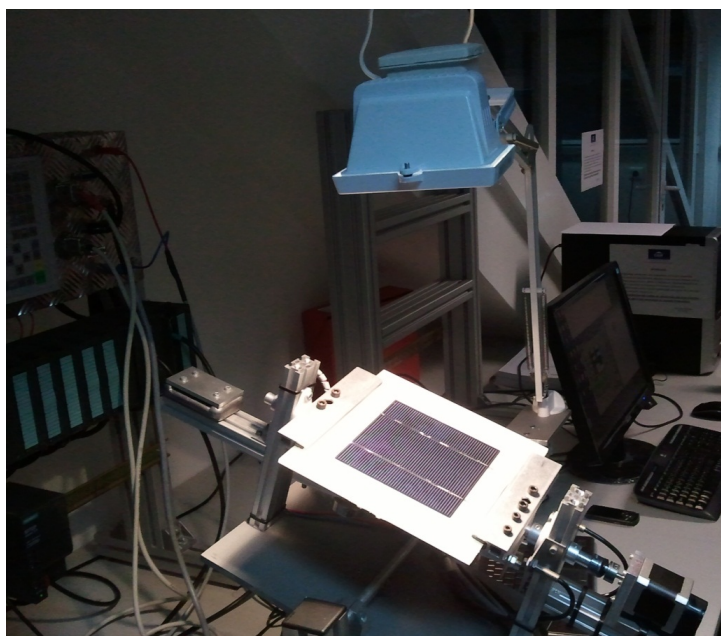


Fig. 94. Resultado de posicionamento ótimo obtido – ensaio 1

Tabela 38. Resultado de um varrimento e colocação na posição ótima – exemplo 1

Durante o varrimento		No posicionamento ótimo	
Posição (°) – eixo 1	10.08	Posição (°) – eixo 1	10.08
Posição (°) – eixo 2	23.40	Posição (°) – eixo 2	24.66
Potência (W) – Célula FV	0.0335	Potência (W) – Célula FV	0.0355

Ensaio 2



Fig. 95. Resultado de posicionamento ótimo obtido – ensaio 2

Tabela 39. Resultado de um varrimento e colocação na posição ótima – ensaio 2

Durante o varrimento		No posicionamento ótimo	
Posição (°) – eixo 1	-19.44	Posição (°) – eixo 1	-46.26
Posição (°) – eixo 2	20.16	Posição (°) – eixo 2	24.66
Potência (W) – Célula FV	0.0121	Potência (W) – Célula FV	0.0121

Ensaio 3

Na fig. 96 representa-se a evolução do valor da potência (painel FV) durante um ensaio, onde se realiza primeiro um processo de varrimento e seguidamente a colocação do painel na posição que maximiza a geração de eletricidade (eixo vertical – potência; eixo horizontal – hora do dia). A azul está representada potência elétrica em cada instante e a vermelho o máximo valor lido. Observa-se que após a sua conclusão o valor da potência do painel FV coincide efetivamente com o valor mais elevado registado durante o varrimento, sendo aproximadamente 5,4 mW.

Os pontos que se seguem identificam as situações numeradas na figura:

- 1) Início do processo de varrimento – posição inicial do sistema;
- 2) Notar que o traçado a vermelho neste ponto ainda representa o máximo valor de potência obtido anteriormente ao início do ensaio;
- 3) É registado o maior valor de potência elétrica;

- 4) O processo de varrimento termina (posição inicial) iniciando-se de seguida a colocação do painel na posição onde a potência foi máxima;
- 5) O sistema foi colocado com sucesso na posição ótima.

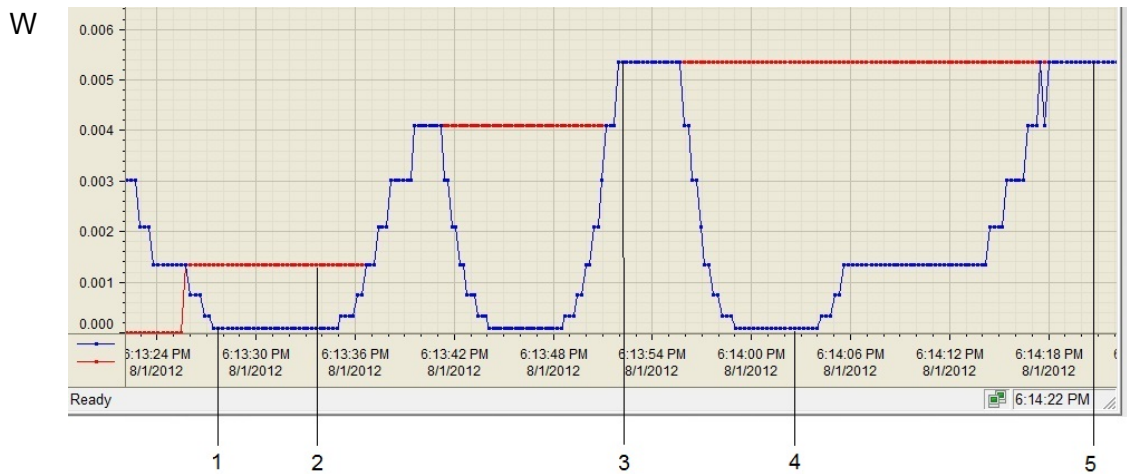


Fig. 96. Representação gráfica no sistema SCADA do ensaio número três

Ensaio 4

A figura que se segue consiste numa captura de imagem da aplicação para monitorização e operação do sistema. Verifica-se que o subsistema secundário está em funcionamento, compreende-se que foi previamente efetuado um processo de varrimento, que está ativado o modo de colocação na posição ótima e que tanto os valores da posição dos eixos como das potências, atual e máxima, estão em conformidade.

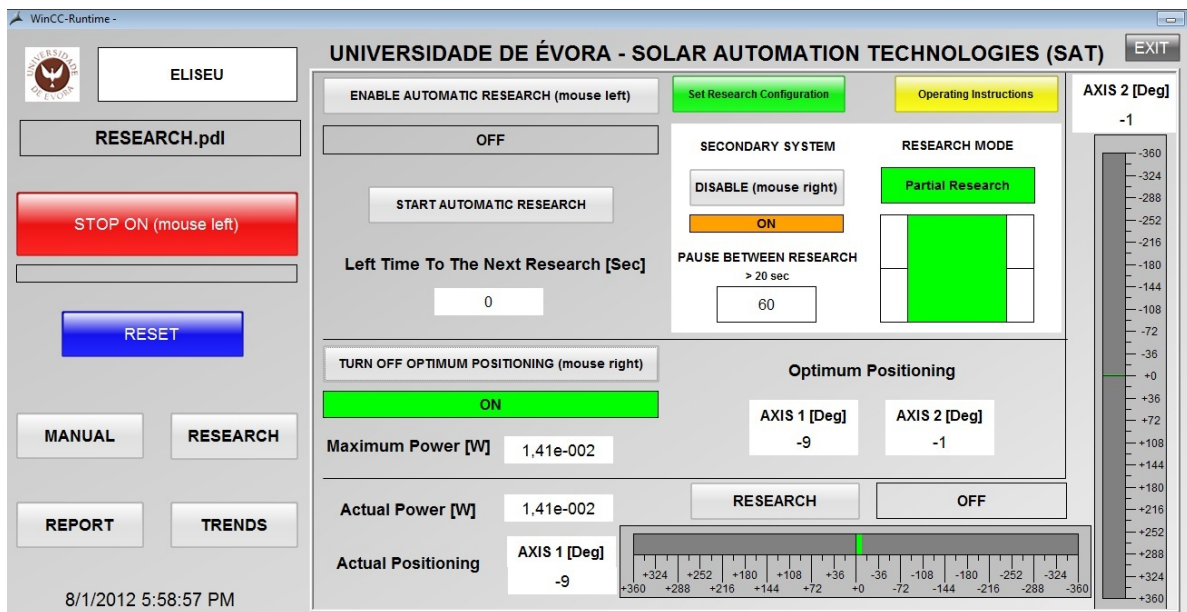


Fig. 97. Interface amigável SCADA

Compilação dos resultados dos ensaios experimentais

Tabela 40. Resultados dos ensaios experimentais

Ensaio	Varrimento	Posição final	Varrimento	Posição final	Varrimento	Posição final
	Eixo 1 (°)		Eixo 2 (°)		Potência elétrica (W)	
1	10,08	10,08	24,66	23,40	3,35E-02	3,35E-02
2	-34,40	-32,40	-48,06	-46,26	4,05E-02	3,69E-02
3	10,08	10,08	5,40	5,40	1,85E-01	1,85E-01
4	29,52	29,52	-45,36	-43,56	3,02E-02	2,71E-02
5	39,60	39,60	19,26	18,54	1,15E-02	1,15E-02
6	-23,04	-23,04	45,90	45,90	1,09E-01	1,09E-01
7	-10,08	-10,08	-12,96	-12,60	1,85E-01	1,85E-01
8	-20,16	-19,44	-48,06	-46,26	1,21E-02	1,21E-02
9	29,52	29,52	5,40	5,40	4,82E-02	4,43E-02
10	-20,16	-19,44	-4,14	-3,96	7,04E-02	5,23E-02
11	-10,08	-10,08	27,36	26,64	6,10E-02	6,10E-02
12	9,36	9,36	-20,70	-19,80	8,05E-02	7,54E-02
13	0,00	0,00	17,10	16,74	9,68E-02	9,68E-02
14	-39,60	-39,60	18,00	17,10	1,03E-01	9,68E-02
15	-50,40	-50,40	-48,06	-46,26	2,14E-02	2,14E-02
16	-29,52	-29,52	-28,80	-27,90	6,56E-02	6,56E-02
17	-28,80	-28,80	48,06	46,26	5,66E-02	5,66E-02
18	29,52	29,52	25,56	24,30	1,15E-01	1,09E-01
19	29,52	29,52	6,84	6,84	1,03E-01	1,03E-01
20	20,16	19,44	-48,06	-46,26	2,42E-02	2,42E-02
21	39,60	39,60	-17,10	-17,82	3,02E-02	3,69E-02
22	18,00	18,00	7,20	6,84	4,43E-02	4,05E-02
23	19,44	19,44	9,00	8,64	3,69E-02	3,69E-02
24	10,08	10,08	19,44	18,90	1,27E-01	1,27E-01
25	0,00	0,00	30,24	29,16	5,66E-02	5,23E-02
26	-29,52	-29,52	27,54	26,64	8,05E-02	8,05E-02
27	0,00	0,00	11,34	10,80	1,15E-01	1,09E-01
28	-10,08	-10,08	-21,24	-21,06	3,35E-02	3,02E-02
29	-20,16	-19,44	-28,26	-27,00	8,57E-02	8,57E-02
30	0,00	0,00	-21,96	-21,96	1,37E-01	1,34E-01
31	0,00	0,00	18,00	17,10	1,48E-01	1,41E-01
32	0,00	0,00	-3,24	-3,06	1,21E-01	1,21E-01
33	10,08	9,36	9,00	8,10	1,62E-01	1,55E-01
34	10,08	10,08	7,02	6,84	1,62E-01	1,62E-01
35	-20,16	-19,44	13,14	12,24	2,01E-01	1,93E-01
36	-29,52	-29,52	44,46	42,46	1,03E-01	1,03E-01
37	-50,40	-50,40	16,20	15,84	1,15E-01	1,15E-01
38	-20,16	-19,44	-6,66	-7,74	4,43E-02	4,43E-02
39	0,00	0,00	-4,50	-4,50	1,88E-02	1,88E-02
40	29,52	28,80	15,84	15,30	2,42E-02	2,42E-02

Comparação entre valores de posicionamento e de potência eléctrica. Determinou-se a diferença entre os valores de posicionamento registados quando se obteve o maior valor de potência eléctrica, durante o varrimento, e os que resultaram quando o mecanismo se deslocou para essa posição (final).

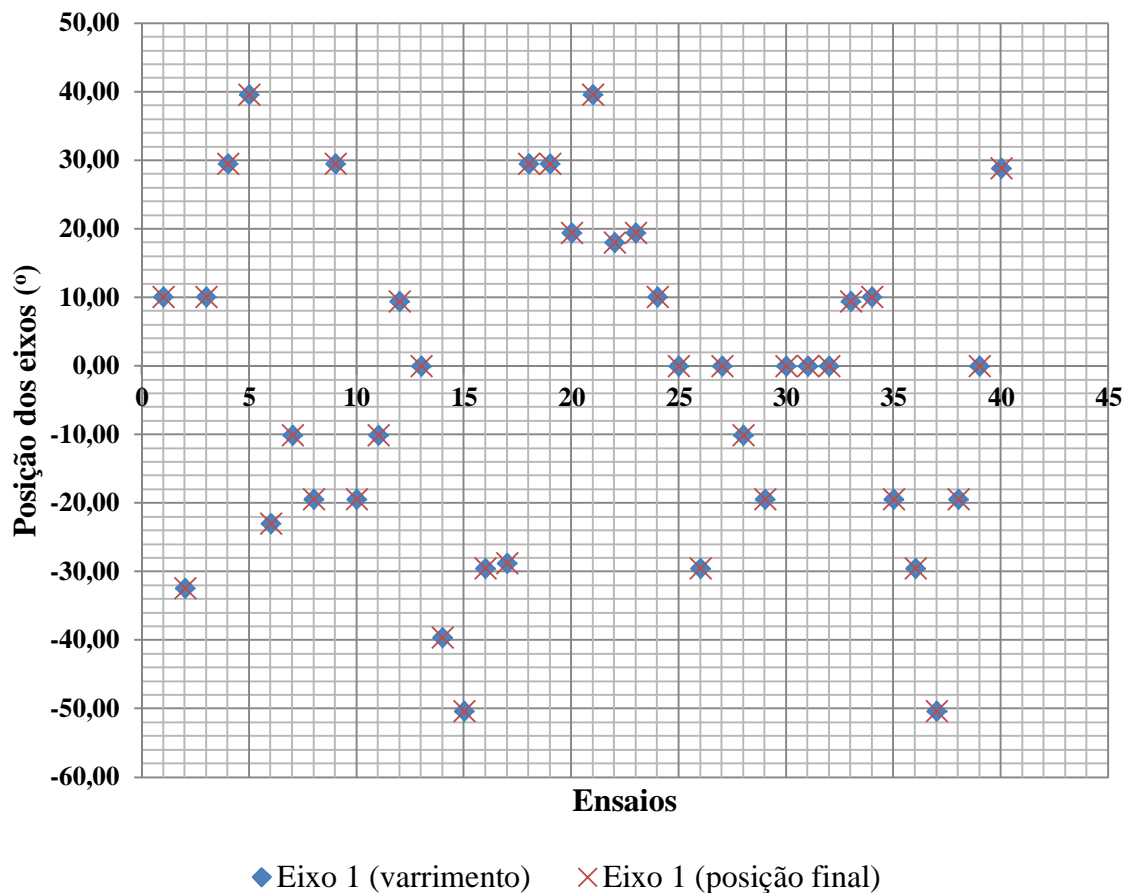


Fig. 98. Comparação do posicionamento do eixo 1: varrimento e posição final

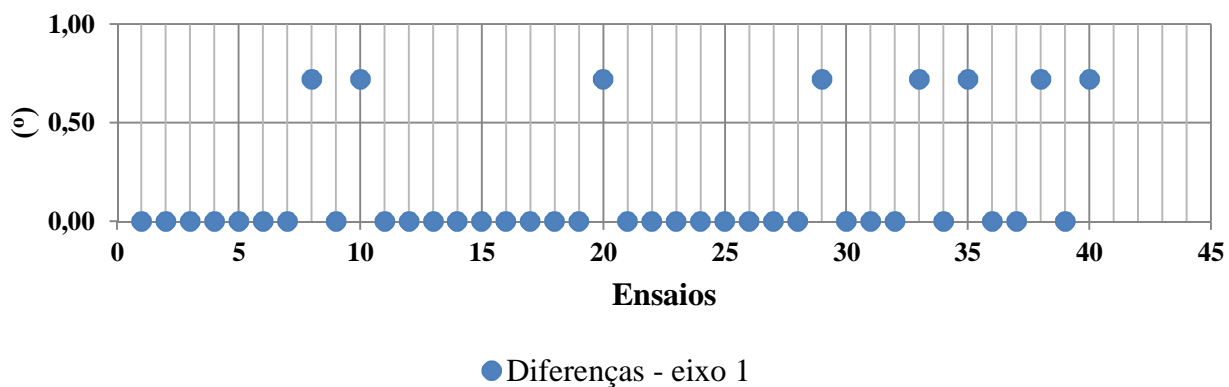


Fig. 99. Diferenças entre o valor de posição registado no varrimento e o valor de posição final, com relação ao eixo 1

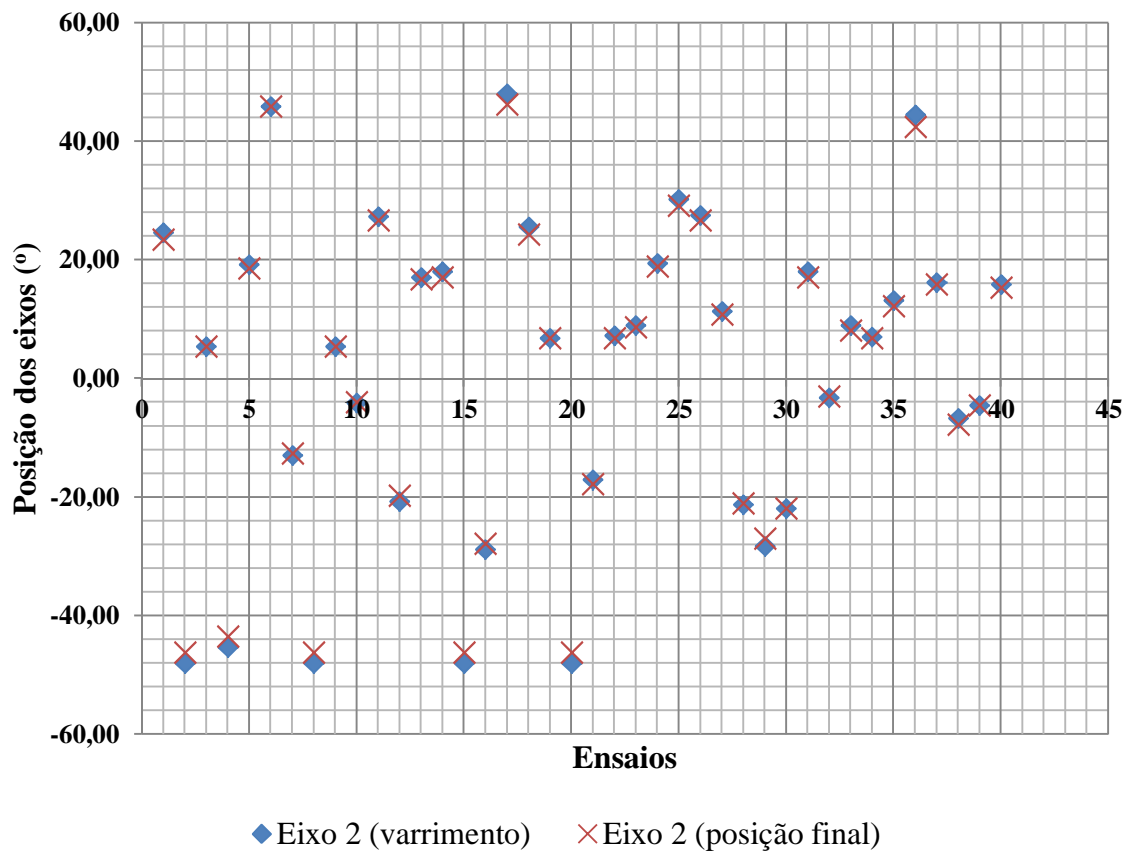


Fig. 100. Comparação do posicionamento do eixo 2: varrimento e posição final

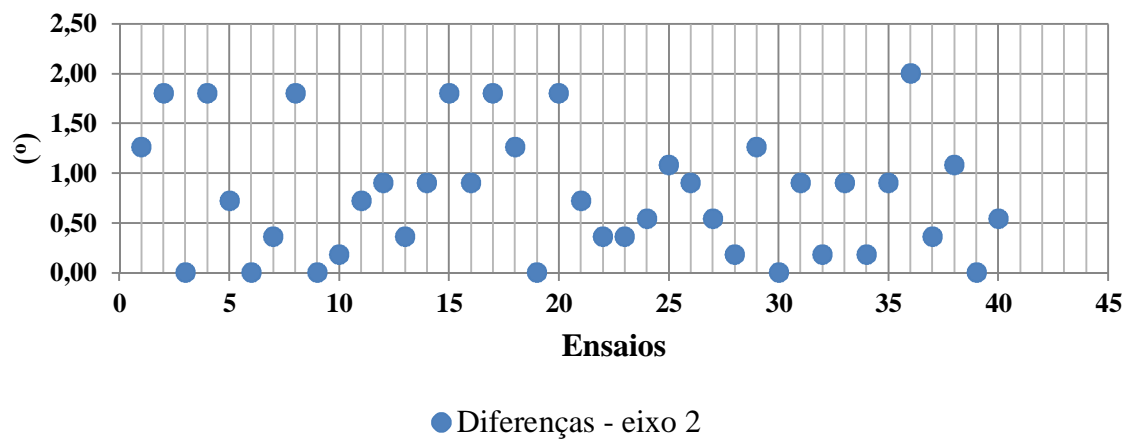


Fig. 101. Diferenças entre o valor de posição registrado no varrimento e o valor de posição final, com relação ao eixo 2

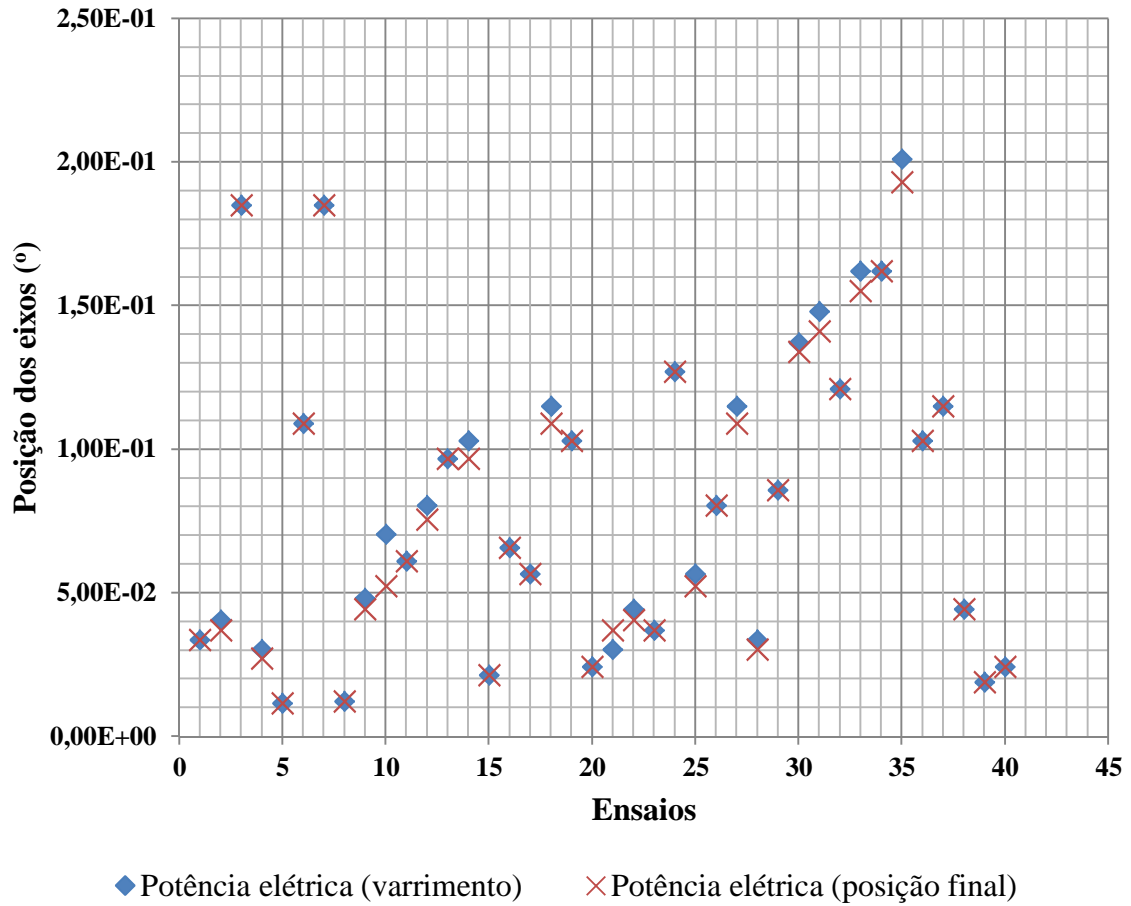


Fig. 102. Comparação de valores de potência elétrica: varrimento e posição final

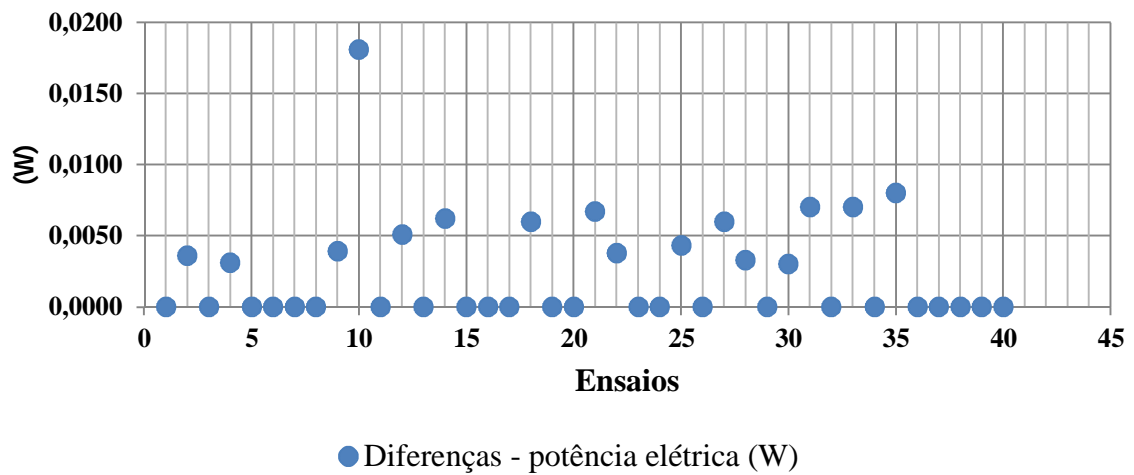


Fig. 103. Gráfico das diferenças entre o valor de potência eléctrica no varrimento e o valor de potência eléctrica final

Análise aos ensaios efetuados

A parametrização do controlo de movimento dos eixos, relativamente ao modo de colocação do sistema na sua posição inicial, foi efectuada de acordo com os resultados obtidos em [2] e alguns testes realizados aquando a implementação deste projeto. Desta forma, para o eixo 1 foi selecionada uma velocidade de 300 graus/min e para o eixo 2 e uma velocidade de 400 graus/min.

Quando o sistema entra no modo de varrimento a velocidade do eixo 1 mantém-se mas a do eixo 2 é alterada para 700 graus/min, por forma a tornar o processo mais rápido.

Após o varrimento, quando a célula FV toma a posição óptima, a velocidade para cada eixo é alterada para 300 graus/min.

Teve-se em consideração as recomendações disponibilizadas em [2], de utilização do eixo 1 no intervalo $(-80^\circ, 80^\circ)$ e do eixo 2 no intervalo $(-50^\circ, 50^\circ)$, por forma a garantir uma utilização segura e eficiente do equipamento.

Tendo em conta o que foi referido e após analisar as figuras (98 a 103), verifica-se que o posicionamento do eixo 1 é mais consistente com o posicionamento óptimo registado no modo de varrimento. O eixo 2 é responsável por movimentar todo o mecanismo, enquanto o eixo 1 apenas movimenta a célula FV. As diferenças verificadas devem-se a oscilações na estrutura mecânica após a paragem dos motores. Notar que o passo do motor também vai influenciar a posição final.

Analisando em concreto os gráficos das diferenças entre valores obtidos durante o varrimento e os valores obtidos na posição final (fig. 99, 101 e 103), verifica-se:

- No eixo 1 a maior diferença de posicionamento é inferior a 1° ;
- No eixo 2 a maior diferença de posicionamento é 2° ;
- A diferença entre os valores de potência elétrica tende a não superar 0,01 W.

Capítulo 9 – Conclusões

O protótipo utilizado para desenvolvimento do sistema apresentado nesta dissertação, foi objecto de apresentação e publicação em atas de conferência internacional [2]. Tendo em conta o nível tecnológico e os resultados obtidos, iniciou-se o trabalho apresentado nesta dissertação com a garantia de bom desempenho do protótipo em termos de hardware.

O trabalho realizado para esta dissertação teve fundamentalmente o objectivo de desenvolver o programa de utilizador, e assim possibilitar a implementação de algoritmos mais refinados que promovam uma orientação otimizada, minimizando o tempo de procura.

O programa executável do controlador do protótipo foi modificado com sucesso. O sistema implementado em laboratório tem agora a capacidade de executar o varrimento no espaço dos dois eixos e de fixar o painel FV na posição em que este registou a potência eléctrica máxima. Destaca-se que o tema da dissertação impõe a utilização de cartas FM353 e FM350-1, interfaces entre motores passo-a-passo e CPU e entre encoders e CPU, respectivamente. Sendo assim, foi necessária a inclusão das funções tecnológicas dos referidos módulos funcionais, sem as quais não é possível realizar a configuração de hardware e colocar o PLC em funcionamento. Notar que estas cartas são aplicáveis em projetos de natureza industrial empresarial, ou seja, em ambientes severamente controlados e, por isso, são colocadas no mercado prontas para executar um vastíssimo leque de funções. Sendo assim, as já referidas funções tecnológicas são de elevada complexidade. Como tiveram obrigatoriamente de ser incluídas no programa de utilizador, este teve de ser desenvolvido de modo a evitar o conflito com o código existente. Ao longo do código desenvolvido, foram introduzidos comentários que explicam como se modificaram algumas das funções e se estabeleceu a interligação com os blocos de programação criados de origem.

Com o objectivo de dar um primeiro passo na redução no consumo dos motores e de minimizar a duração de qualquer varrimento, foi introduzido um sistema complementar, denominado de subsistema secundário. Este revelou ter potencial para pré-determinar a posição de maior exposição à radiação. Foi construído com quatro circuitos integrados (TSL230R-LF), que foram distribuídos espacialmente numa pequena estrutura, de modo a não possuírem idêntica exposição luminosa. O microcontrolador que determina o seu funcionamento, disponibiliza numa saída analógica para o PLC, indicando sobre quais os sensores com maior nível de exposição. A introdução destes elementos externos permitiu estabelecer uma estratégia, muito simples, de criação de vários programas para menores áreas de varrimento, apropriados à resposta do subsistema secundário.

Trabalhos futuros:

- Ao nível do equipamento, sugere-se uma diminuição do peso da estrutura que suporta a célula FV, por forma a melhorar o posicionamento dos eixos;
- Adicionar mais células FV e implementar um algoritmo que utilize os seus sinais de tensão eléctrica para otimizar o varrimento no espaço dos dois eixos;

- Atualizar a programação do subsistema principal ao nível da interligação dos modos de funcionamento. Por exemplo, adicionar um único botão para dar início ao modo de varrimento automático, sem que antes seja necessário realizar uma pesquisa e enviar o sistema para a posição ótima;
- Implementar um algoritmo que indique ao sistema a necessidade de realizar um novo varrimento, por se ter registado uma diminuição significativa da potência elétrica;
- Criar uma função que desligue automaticamente o sistema durante a noite e o volte a ligar a partir do momento em que existe exposição solar;
- Tornar o seguidor solar do tipo misto, introduzindo um algoritmo com base na equação de tempo;
- Tendo em conta a utilização do microcontrolador (subsistema secundário), adicionar um sensor de corrente para medir a corrente à saída da célula FV. Esta ação também se justifica pelo facto do valor óhmico da resistência (carga) poder sofrer alterações ao longo do tempo;
- Implementar a comunicação entre subsistemas via software (p.e. LabView) e desenvolver um sistema de monitorização especificamente para o subsistema secundário.

Bibliografia

- [1] Figueiredo, João; Sá da Costa, J.; “Intelligent Sun-Tracking System for Efficiency Maximization of Photovoltaic Energy Production”, ICREPQ Intl. Conf. On Renewable Energy and Power Quality, Santander, March 2008
- [2] Robalo, Robalo; Figueiredo, João; “Supervisory Control for a Solar Tracking Prototype Based on PV-Technology”; IFAC Conference on Control Methodologies and Technology for Energy Efficiency; 2010
- [3] Castro, Rui; “Uma Introdução às Energias Renováveis”; Ensino da Ciência e da Tecnologia; IST PRESS; 2011
- [4] Jackson, Munzila; Ferreira, Fernando; Quivota, Rui; “Casa-Laboratório do MINEA – Uma Plataforma Avançada para o Estudo Técnico-Económico de Sistemas Fotovoltaicos em Angola”; Ministério da Energia e Águas da República de Angola; 2011; Portal da Internet (consultado em em 2011):
http://www.minerg.gv.ao/index.php?option=com_docman&task=cat_view&gid=76&Itemid=128&lang=pt
- [5] EIA, International Energy Outlook, 2011: <http://www.eia.gov/forecasts/ieo/> (consultado em 2011)
- [6] Portal da Internet, Enerdata: <http://www.enerdata.net/> (consultado em 2012)
- [7] Portal da Internet, Comissão Europeia: <http://re.jrc.ec.europa.eu/pvgis/> (consultado em 2011)
- [8] Canhoto, Paulo; “Energia Solar”: <http://www.parquepeter.org/> (consultado em 2012)
- [9] Quaschnig, V.; “Understanding renewable energy systems”; London: Earthscan, 2005
- [10] Kalogirou, Soteris A.; “Solar Energy Engineering - Processes and Systems”; Elsevier - Academic Press; 2009
- [11] Siemens; SIMATIC; “Communication with SIMATIC”; System Manual; 2006
- [12] Siemens; SIMATIC S7-300 CPU 31xC and CPU 31x: Specifications; Manual; 2011
- [13] Siemens; SIMATIC; S7-300 and M7-300 Programmable Controllers Module Specifications; Reference Manual; 2011
- [14] Siemens; SIMATIC; FM 350-1 Function Module; Manual; 2008
- [15] Hewlett-Packard: “Quick Assembly Two and Three Channel Optical Encoders”; Technical Data; 2002
- [16] Francisco, António; “Motores Eléctricos”; ETEP; 2008
- [17] Nanotec; Technical Data: <http://en.nanotec.com/start.html> (consultado em 2011)
- [18] Siemens; SIMATIC; FM 353 Function Module; Manual; 2008

- [19] Technical Data; TAOS/AMS: <http://www.ams.com/eng> (consultado em 2011)
- [20] Hughes, Mark; “Basic Stamp Experiments Using the TSL230”; Intelligent Opto Sensor Designer’s Notebook; TAOS; N.º5: (consultado em 2011)
<http://www.globalspec.com/reference/5886/tsl230-basic-stamp-experiments>
- [21] Portal da Internet: <http://fritzing.org/> (consultado em 2011)
- [22] Portal da internet: <http://arduino.cc/en/> (consultado em 2011)
- [23] Santos, Nuno; “Arduino Introdução e Recursos Avançados”; Ramo de Armas e Electrónica; Departamento de Engenheiros Navais; Escola Naval; 2009
- [24] Banzi, Massimo; “Getting Started with Arduino”; O’REILLY; 2009
- [25] Siemens; SIMATIC; “Statement List (STL) for S7-300 and S7-400 Programming”; Reference Manual; 2006
- [26] Prince, Eddie; “Pointers and Indirect Addressing in S7 PLCs”; Totally Integrated Automation Application Notes; 2000
- [27] Siemens; SIMATIC HMI; “Multi Panel MP 270B, Touch Panel MP 270B, Touch Panel TP 270, Operator Panel OP 270”; Equipment Manual; 2001
- [28] Siemens; SIMATIC HMI; “WinCC Flexible 2008 Compact/Standard/Advanced”; User’s Manual; 2008
- [29] Fonseca, José; “Redes de Comunicação em Ambientes Industriais”; Apontamentos da Disciplina; Engenharia Electrotécnica; Universidade de Aveiro; 1999
- [30] Figueiredo, João; “Guia Prático para Configuração e Implementação de Redes Industriais Siemens”; Apontamentos da Disciplina de Sistemas de Supervisão e Controlo; Mestrado em Engenharia Mecatrónica; Universidade de Évora; 2008
- [31] Siemens; SIMATIC HMI; WinCC V7.0 SP1 MDM – WinCC: Working with WinCC; System Manual; 2008



UNIVERSIDADE DE ÉVORA
ESCOLA DE CIÊNCIAS E TECNOLOGIA

Mestrado em Engenharia Mecatrónica

Dissertação – Anexos

Automação na Maximização de Produção de Energia Elétrica de um Painel Solar

Eliseu António Calado Fernandes

Orientador:

Professor Doutor João Manuel Gouveia Figueiredo

Setembro de 2012

Mestrado em Engenharia Mecatrónica

Dissertação – Anexos

Automação na Maximização da Produção de Energia Elétrica de um Painel Solar

Eliseu António Calado Fernandes

Orientador:

Professor Doutor João Manuel Gouveia Figueiredo

Anexos

Índice

Anexo A: Características técnicas dos encoders	1
Anexo B: Motores passo-a-passo - características técnicas	4
Anexo C: Parametrização do módulo FM 350-1 – encoders	7
Anexo D: Parametrização do módulo FM 353 – eixo 1	12
Anexo E: Parametrização do módulo FM 353 – eixo 2.....	20
Anexo F: Diagramas dos varrimentos programados.....	27
Anexo G: Programação dos varrimentos (G-Code).....	30
Anexo H: Configuração da rede para sistema de supervisão	33
Anexo I: Aplicação para Painel HMI (WinCC Flexible 2008)	37
Anexo J: Sistema de Supervisão em WinCC V7.....	54
Anexo K: Ficha técnica do circuito integrado TSL230R-LF.....	69
Anexo L: Tabela de sinais de controlo e de verificação de FC1	78
Anexo M: Programação	80
Tópico 4 – Índice de blocos (Programação STEP 7)	81

Anexo A: Características técnicas dos encoders

Optischer Impulsgeber

Die 3-Kanal-Encoder zeichnen sich neben der kleinen Bauform und der äußerst geringen Eigenmasse vor allem auch durch die schnelle und einfache Montage aus.

Technische Daten:

Betriebsspannung:	DC 4,5 V bis 5,5 V
Stromaufnahme max:	(bei 5V) 57 mA
Pulsbreite:	180 ± 35 Grad (HEDM ± 45 Grad)
Signal-Phasenverschiebung:	(Kanal A zu B) 90 ± 15 Grad
Signal-Anstiegs-/Abfallzeit:	0,25 / 0,25 µs
Grenzfrequenz:	bis 100 kHz
Ausgangssignale:	rechteckig 2+1 Nullimpuls (HEDM 2)
Impulse pro Umdrehung:	200, 400, 500, 1000, 2000
Zusatzoption:	mit integriertem Line-Treiber
Betriebstemperatur:	0°C bis +70°C

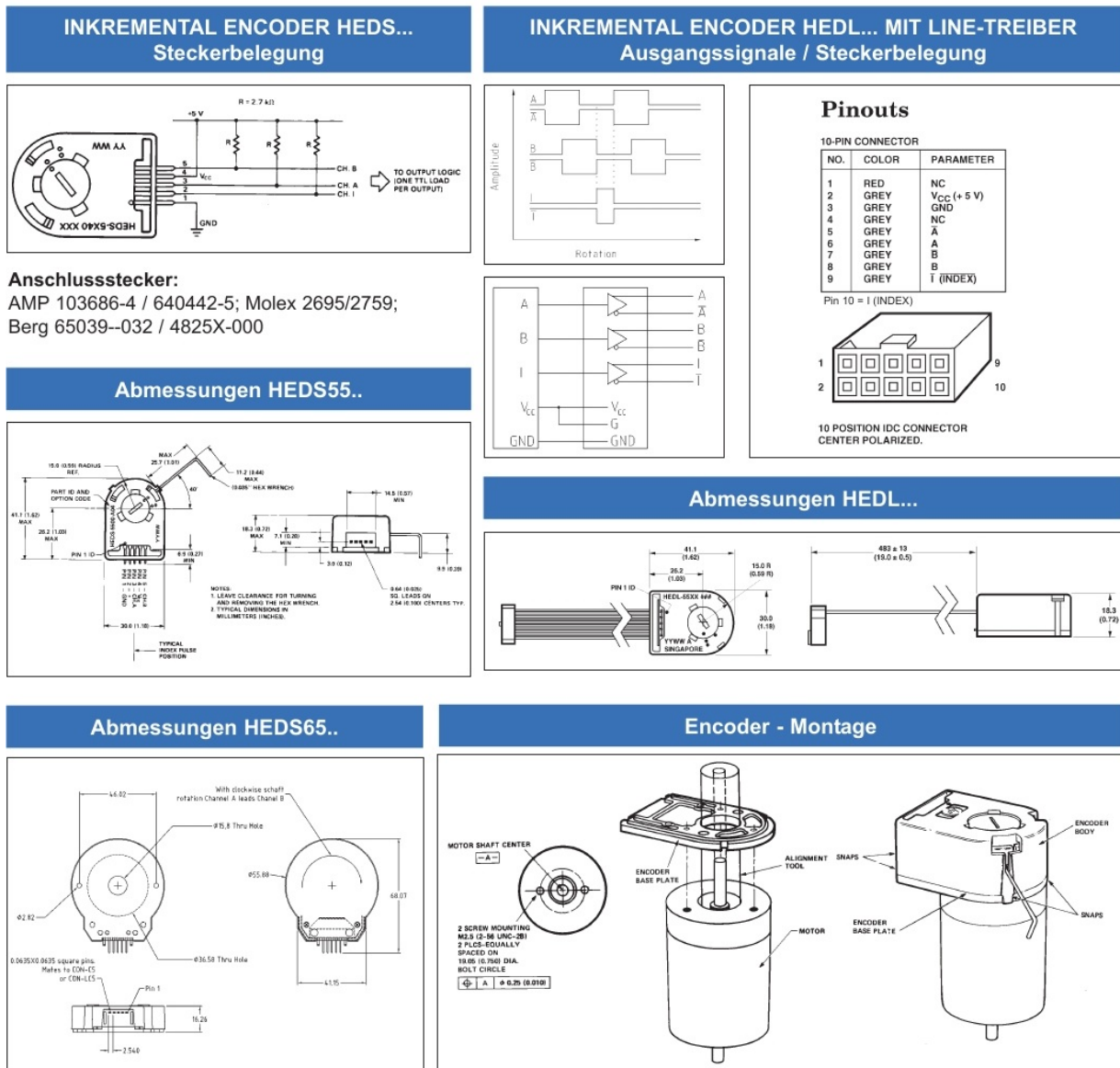


Fig. 1. Ficha técnica (1 de 2) dos encoders

Encoder

Optische Impulsgeber: Standard Encoder für Schrittmotoranbau

Bestellbezeichnung	Impulse / Umdr.	für Well Ø mm
HEDS-5540 E06	200	6,35
HEDS-5540 H14	400	5,0
HEDS-5540 H06	400	6,35
HEDS-5540 A02	500	3,0
HEDS-5540 A14	500	5,0
HEDS-5540 A12	500	6,0
HEDS-5540 A06	500	6,35
HEDS-5540 A13	500	8,0
HEDM-5500 B14 (2 Kanal, ohne I-Ausgang)	1000	5,0
HEDM-5500 B06 (2 Kanal, ohne I-Ausgang)	1000	6,35


Encoder mit Line-Treiber (für extrem störsichere Einsatzbedingungen oder lange Zuleitungen)

HEDL-5540 E06	200	6,35
HEDL-5540 H14	400	5,0
HEDL-5540 H06	400	6,35
HEDL-5540 A02	500	3,0
HEDL-5540 A14	500	5,0
HEDL-5540 A12	500	6,0
HEDL-5540 A06	500	6,35
HEDL-5540 A13	500	8,0

Hohlwellenencoder für Durchgangswelle speziell für Linearaktuatoren:

HEDS-6545 H16	400	16,0
HEDS-6545 C16	2000	16,0

Anschlußkabel (passend für HEDS...)
 Kabel-HEDS-8903
 (Stecker mit Einzelleitern, 150 mm lang)

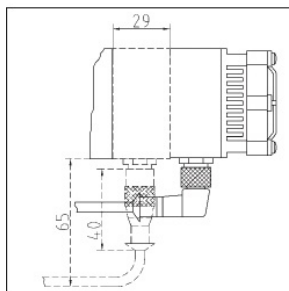


andere Impulse/Umdr. oder für andere Wellendurchmesser auf Anfrage.

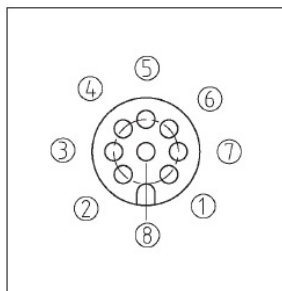
integrierte Encoder-Versionen

- Inkrementalencoder E2 = 200 Imp.; E4 = 400 Imp.; E5 = 500 Imp.; E1000 = 1000 Imp.
- Inkrementalencoder mit Line-Treiber L2 = 200 Imp.; L4 = 400 Imp.; L5 = 500 Imp.

für Plug&Drive Motor



Steckerbelegung 8-pol. Anschlußkabel (Z-Anschlußkabel)



Belegung-HEDS..(E2-E100)

- | | |
|-----------|-------------|
| 1) weiß | = nc |
| 2) braun | = CH B |
| 3) grün | = Vcc + 5 V |
| 4) gelb | = CH A |
| 5) grau | = CH I |
| 6) rosa | = GND 0 V |
| 7) blau | = nc |
| 8) Schirm | = nc |

Passendes Anschlußkabel siehe Seite 92

Belegung-HEDL..(L2-L5)

- | | |
|-----------|-------------|
| 1) weiß | = CH A\ |
| 2) braun | = CH B |
| 3) grün | = Vcc + 5 V |
| 4) gelb | = CH A |
| 5) grau | = CH I |
| 6) rosa | = GND 0 V |
| 7) blau | = CH B\ |
| 8) Schirm | = nc |

Passendes Anschlußkabel siehe Seite 92

Fig. 2. Ficha técnica (2 de 2) dos encoders

Anexo B: Motores passo-a-passo - características técnicas

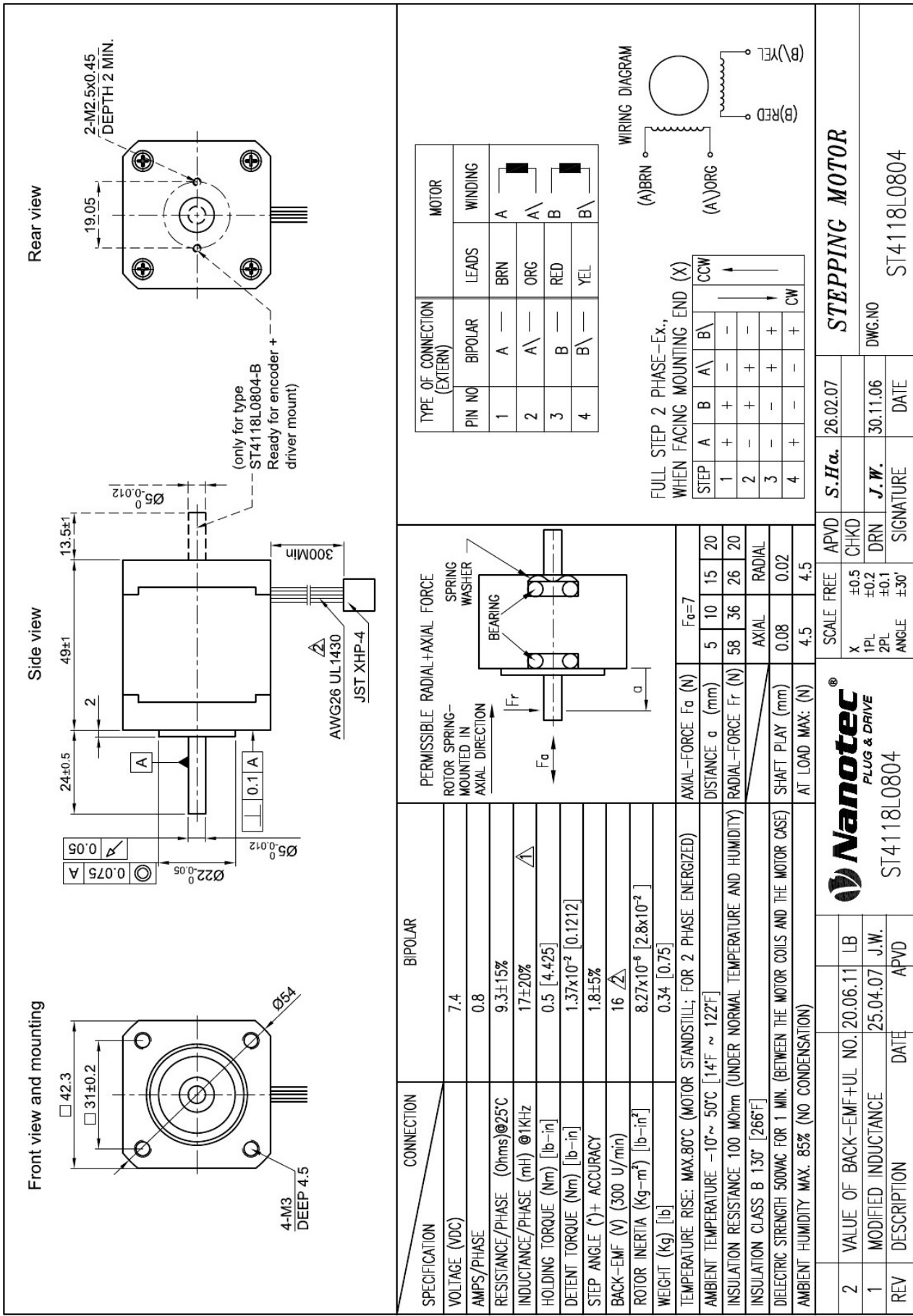
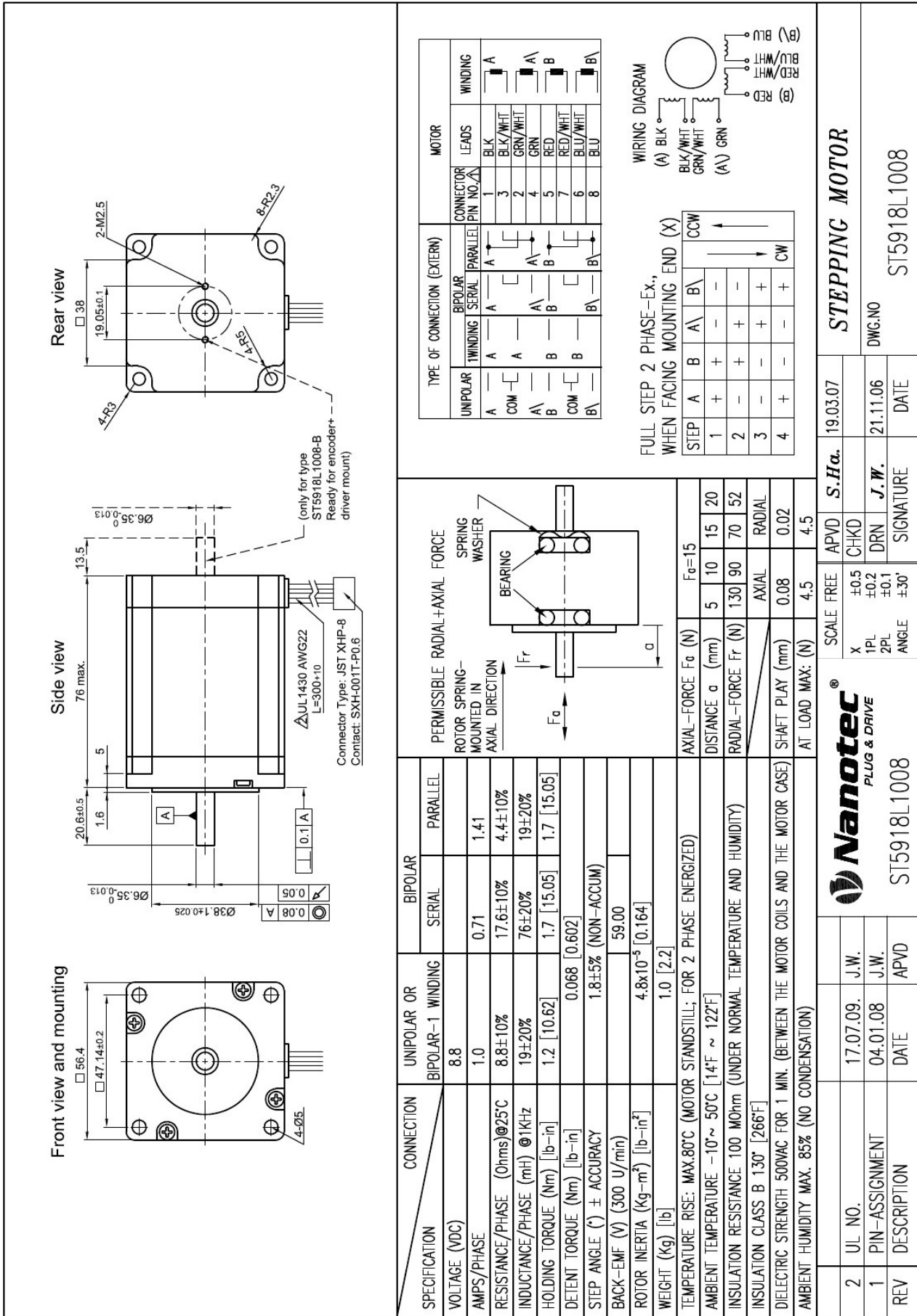


Fig. 3. Ficha técnica do motor do eixo 1



Nanotec
PLUG & DRIVE

ST5918L1008

APVD

ST5918L1008

Fig. 4. Ficha técnica do motor do eixo 2

Anexo C: Parametrização do módulo FM 350-1 – encoders

1. Instalar a aplicação “*Parameterize FMx50*”;
2. Ir para a aplicação de configuração de hardware “*HW Config*”;
3. Fazer duplo clique no módulo FM 350-1 correspondente ao motor do eixo 1;
4. Selecionar a opção “*Encoders*” e realizar a parametrização dos seguintes pontos: “*Signal type*”, “*Signal evaluation*” e “*Monitoring*”;

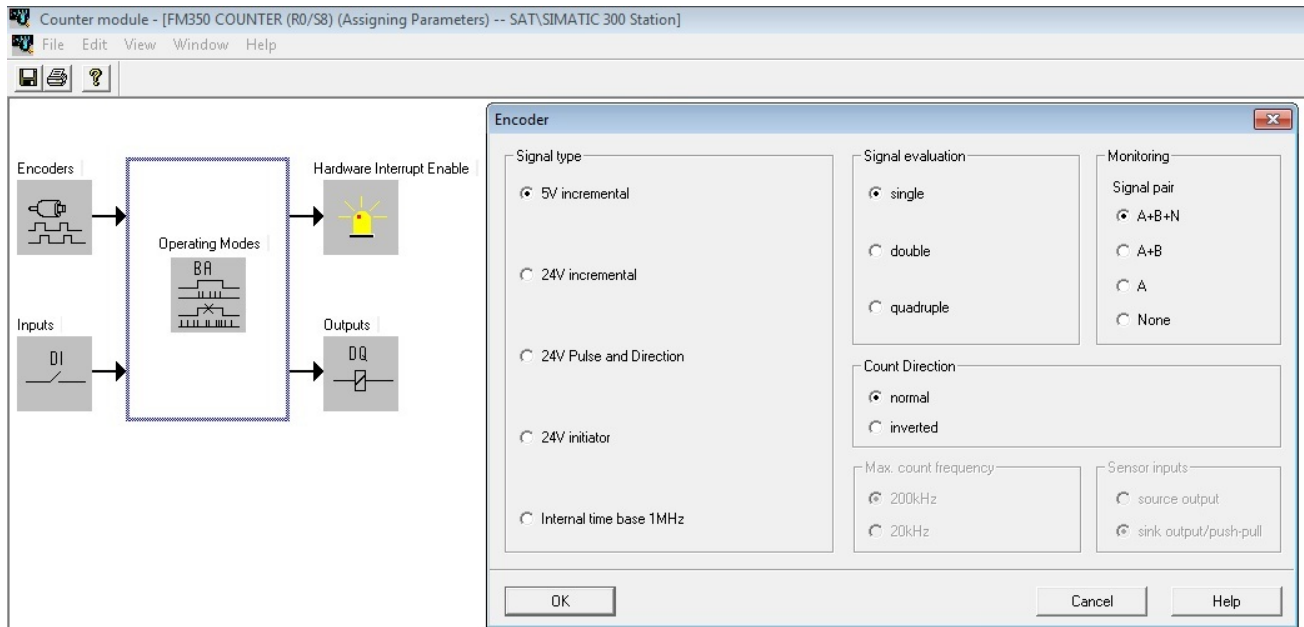


Fig. 5. FM 350-1 Encoder 1: “*Signal type*”, “*Signal evaluation*” e “*Monitoring*”

5. Selecionar a opção “*Inputs*” e parametrizar de acordo com a figura que se segue;

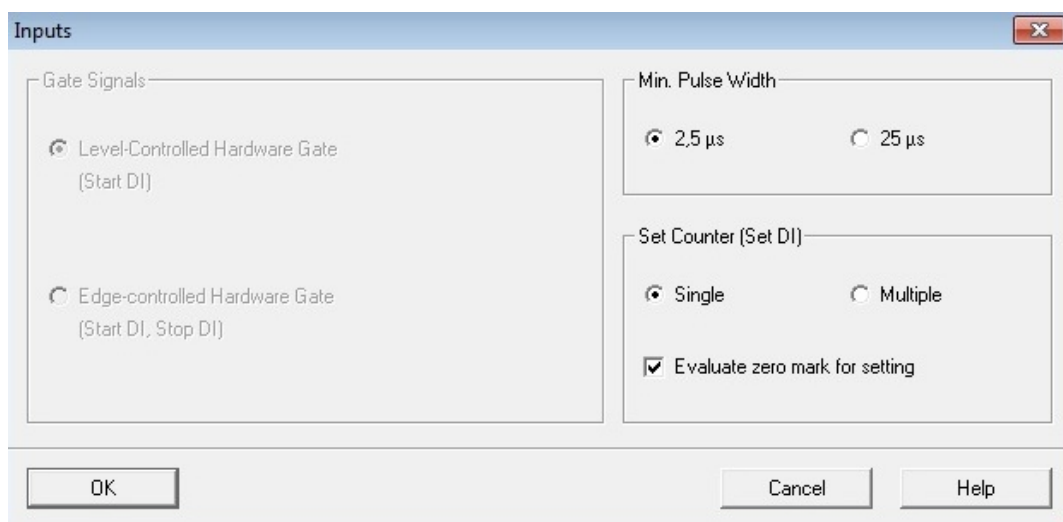


Fig. 6. FM 350-1 Encoder 1: “*Inputs*”

6. Selecionar a opção “*Operating Modes*” e parametrizar de acordo com a figura que se segue;



Fig. 7. FM 350-1 Encoder 1: “*Operating Modes*”

7. Realizar o mesmo processo para o encoder do eixo 2, não esquecendo de inverter o sentido da contagem, com é possível observar na fig. 8.

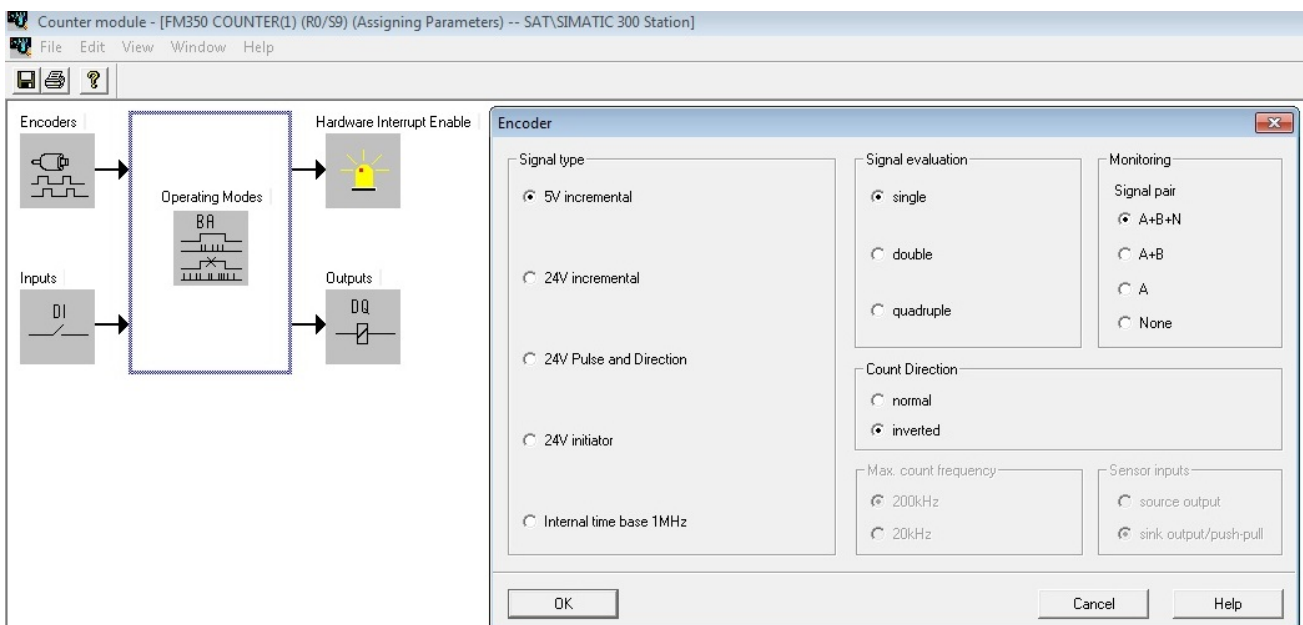


Fig. 8. FM 350-1 Encoder 2: “*Signal type*”, “*Signal evaluation*” e “*Monitoring*”

```

Properties
=====
Module : FM350 COUNTER
Order No. : 6ES7 350-1AH03-0AEO

Description : FM350 COUNTER
Station : SIMATIC 300 Station
Rack (R) : 0
Slot (S) : 8

Addresses
Inputs - Start : 320
Inputs -End : 335
Outputs - Start : 320
Outputs - End : 335

Select Interrupt : None
Interrupt OB: Not relevant
Reaction to CPU STOP : STOP
Reaction to new parameters: Always reset

Operating Modes
=====

Operating mode: Continuous counting
Gate control None
Count range 0 to +32 Bits

Encoders
=====
Encoder Type 5V incremental
Signal evaluation single
Monitoring A+B+N
Count Direction normal

Inputs
=====
Min. Pulse Width 2,5 µs
Set Counter (Set DI) Single
Evaluate zero mark for setting Yes

Hardware Interrupt Enable
=====

Opening of the gate No
Closing of the gate No
Overflow No
Underflow No
Reaching comparison value 1 up No
Reaching comparison value 1 down No
Reaching comparison value 2 up No
Reaching comparison value 2 down No
Zero pass No
Setting the counter No
Latching No

Outputs
=====
Pulse duration: 0 ms
Hysteresis: 0
Substitute value at CPU Stop D00 0
Substitute value at CPU Stop D01 0
Output behavior D00 Inactive
Output behavior D01 Inactive

```

Fig. 9. FM 350-1 Encoder 1: Resumo da parametrização efetuada

```

Properties
=====
Module : FM350 COUNTER
Order No. : 6ES7 350-1AH03-0AEO

Description : FM350 COUNTER(1)
Station : SIMATIC 300 Station
Rack (R) : 0
Slot (S) : 9

Addresses
Inputs - Start : 336
Inputs -End : 351
Outputs - Start : 336
Outputs - End : 351

Select Interrupt : None
Interrupt OB: Not relevant
Reaction to CPU STOP : STOP
Reaction to new parameters: Always reset

Operating Modes
=====

Operating mode: Continuous counting
Gate control None
Count range 0 to +32 Bits

Encoders
=====
Encoder Type 5V incremental
Signal evaluation single
Monitoring A+B+N
Count Direction normal

Inputs
=====
Min. Pulse Width 2,5 µs
Set Counter (Set DI) Single
Evaluate zero mark for setting Yes

Hardware Interrupt Enable
=====

Opening of the gate No
Closing of the gate No
Overflow No
Underflow No
Reaching comparison value 1 up No
Reaching comparison value 1 down No
Reaching comparison value 2 up No
Reaching comparison value 2 down No
Zero pass No
Setting the counter No
Latching No

Outputs
=====
Pulse duration: 0 ms
Hysteresis: 0
Substitute value at CPU Stop D00 0
Substitute value at CPU Stop D01 0
Output behavior D00 Inactive
Output behavior D01 Inactive

```

Fig. 10. FM 350-1 Encoder 2: Resumo da parametrização efetuada

Anexo D: Parametrização do módulo FM 353 – eixo 1

1. Instalar a aplicação “*Parameterize FM 353*”;
2. Ir para a aplicação de configuração de hardware “*HW Config*”;
3. Fazer duplo clique no módulo FM 353 correspondente ao motor do eixo 1;
4. Fazer duplo clique no botão “*Parameter*”;

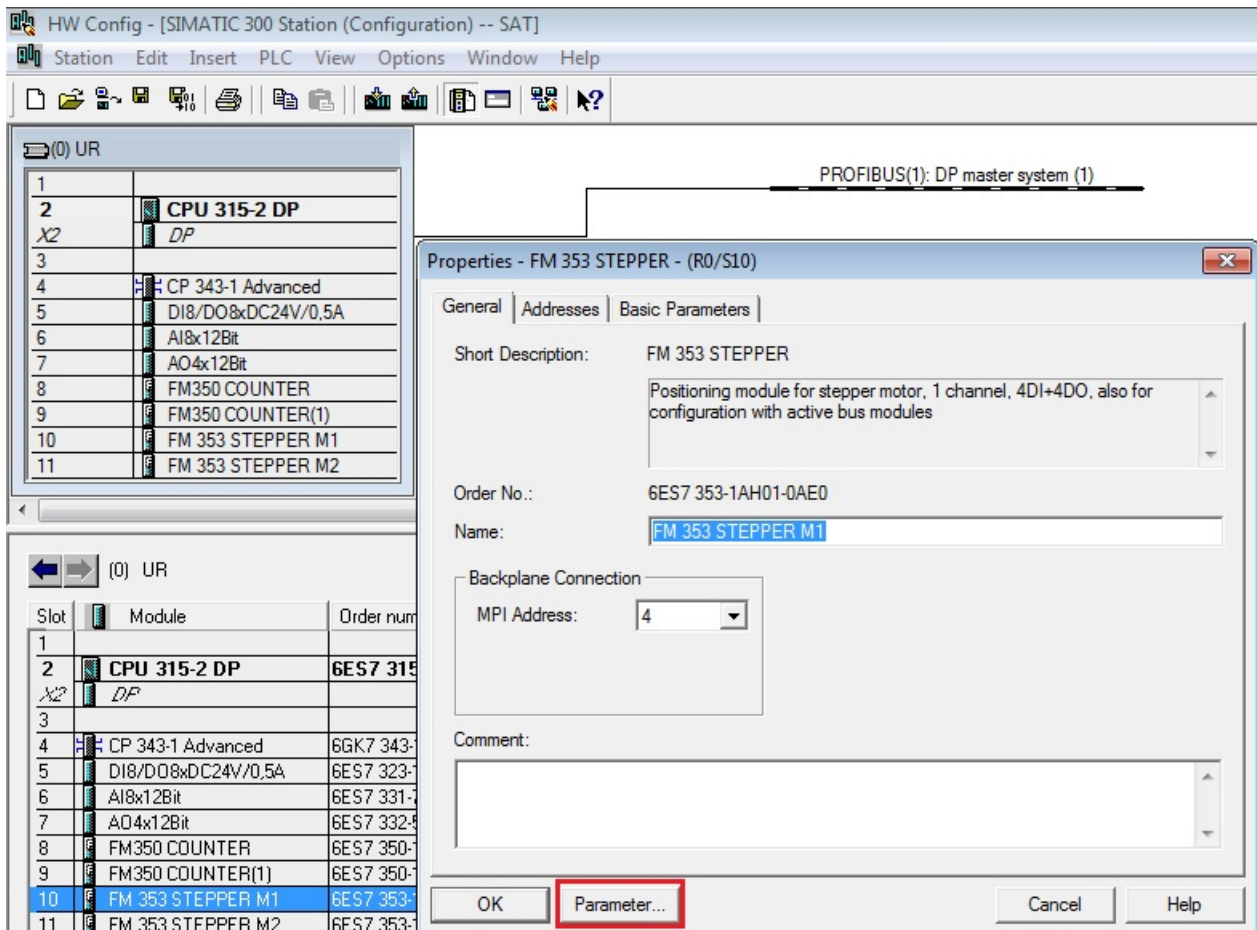


Fig. 11. FM 353 Motor 1: Acesso à aplicação de parametrização

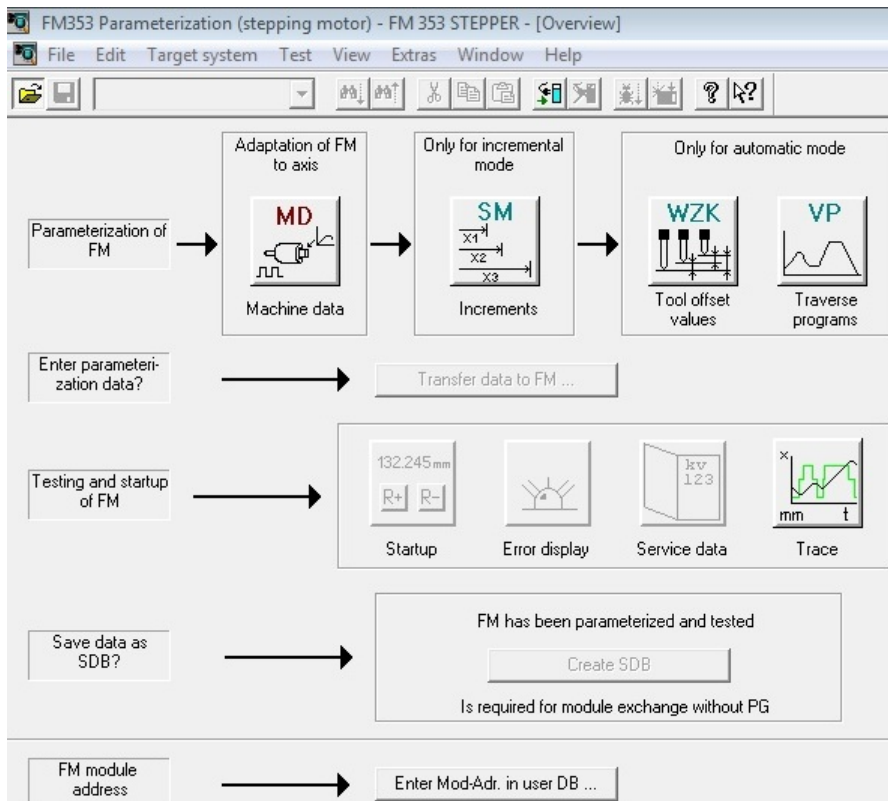


Fig. 12. FM 353 Motor 1: Janela principal da aplicação de parametrização

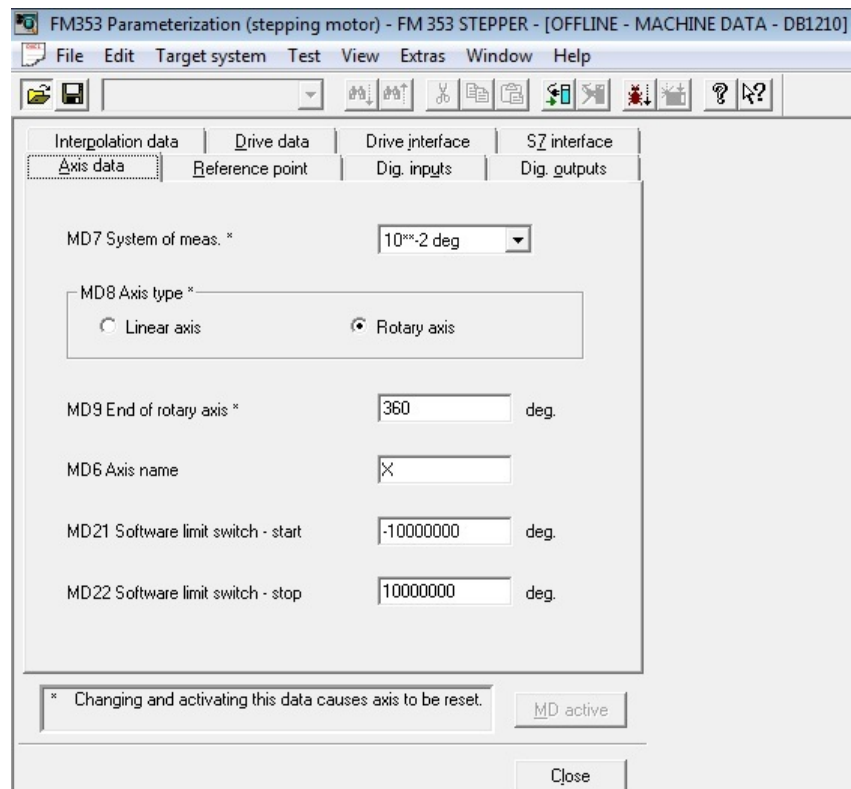


Fig. 13. FM 353 Motor 2: "Reference point"

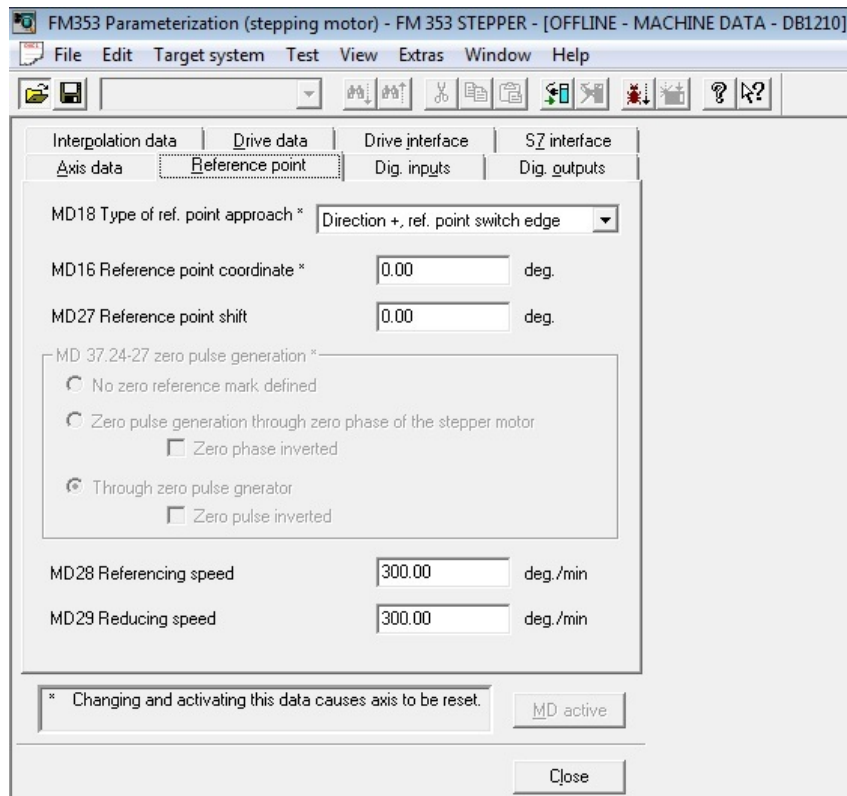


Fig. 14. FM 353 Motor 1: “Axis data”

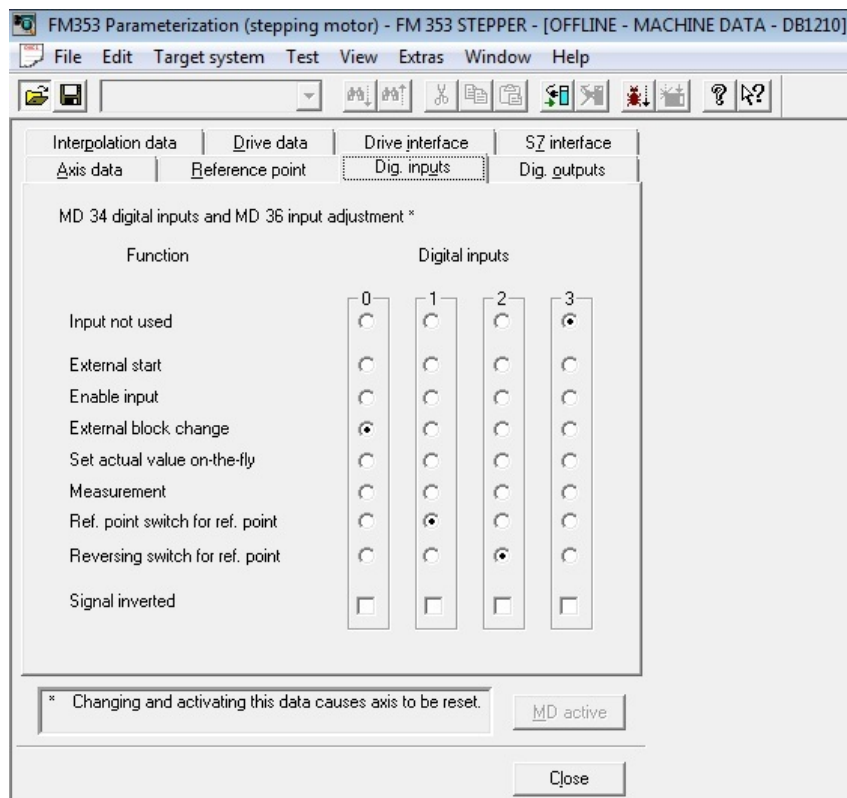


Fig. 15. FM 353 Motor 1: “Digital inputs”

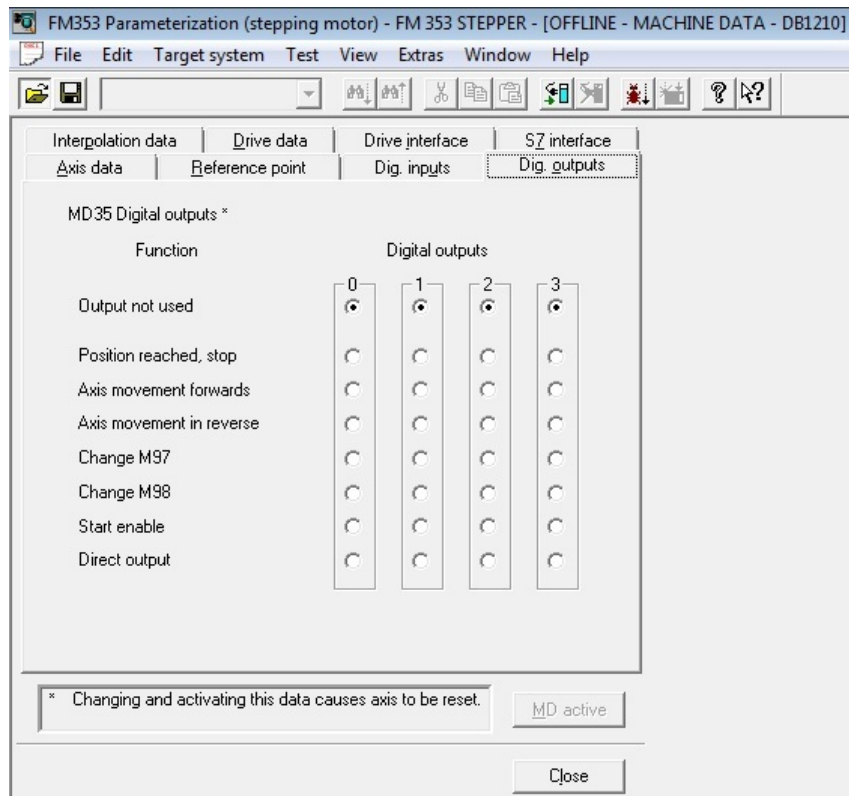


Fig. 16. FM 353 Motor 1: “Digital outputs”

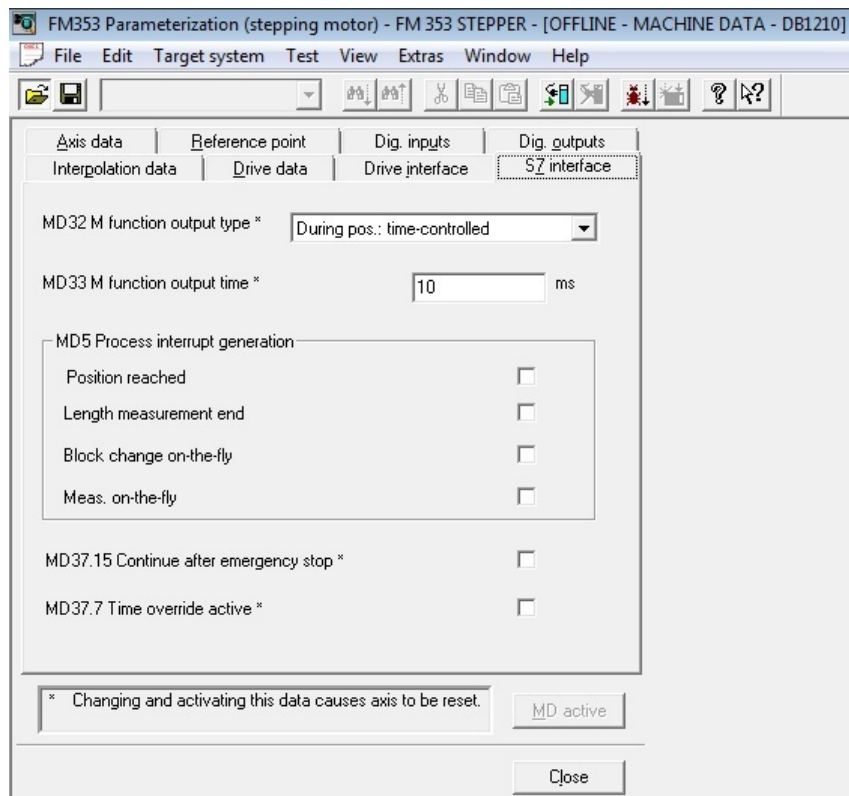


Fig. 17. FM 353 Motor 1: “S7_interface”

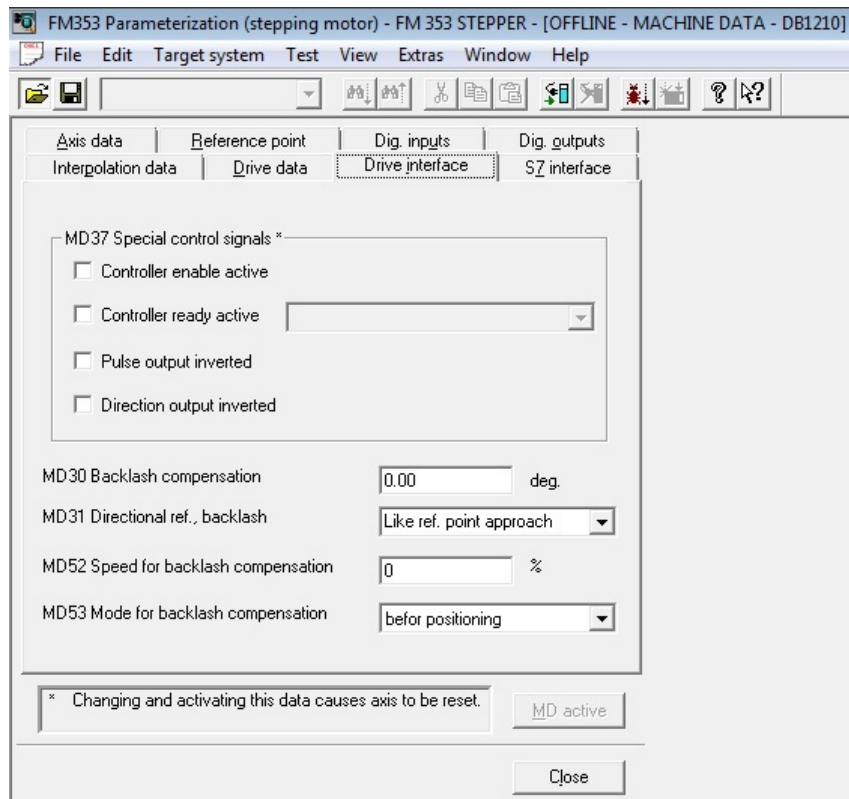


Fig. 18. FM 353 Motor 1: “*Drive interface*”

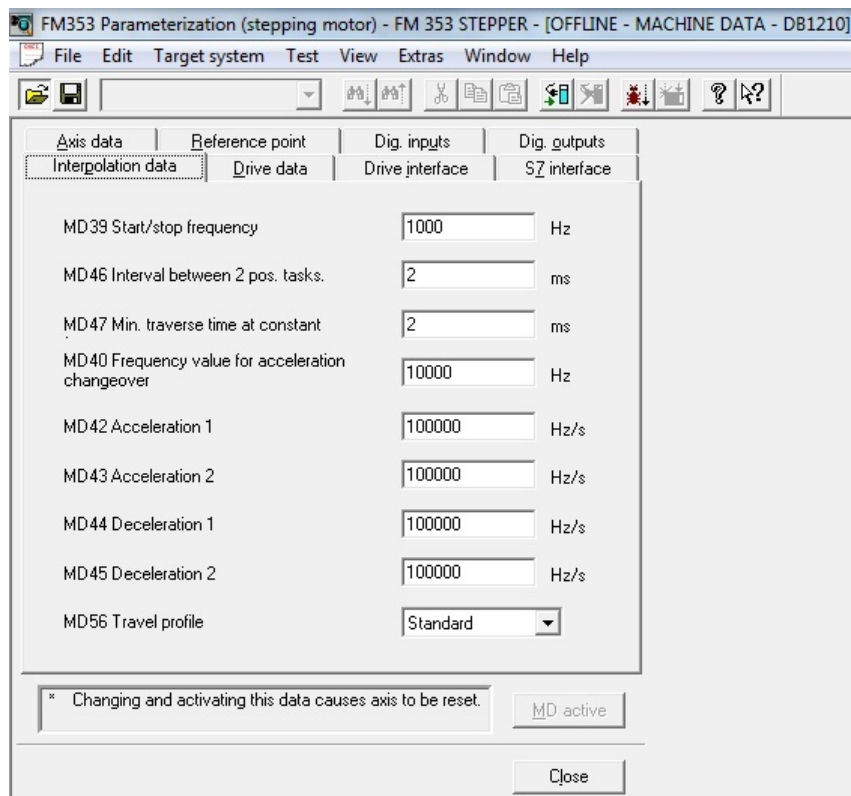


Fig. 19. FM 353 Motor 1: “*Interpolation data*”

FM353 Parameterization (stepping motor)

=====

Properties

MPI address.....4
 Address / input.....352

 Measure.....10**-2 deg

Axis data

MD7 System of meas. * 10**-2 deg
 MD8 Axis type * Rotary axis
 MD9 End of rotary axis * 360 deg.
 MD6 Axis name X
 MD21 Software limit switch - start -10000000 deg.
 MD22 Software limit switch - stop 10000000 deg.

Reference point

MD18 Type of ref. point approach * Direction +, ref. point switch edge
 MD16 Reference point coordinate * 0.00 deg.
 MD27 Reference point shift 0.00 deg.
 MD 37.24-27 zero pulse generation * Through zero pulse gnerator
 Zero pulse inverted No
 MD28 Referencing speed 300.00 deg./min
 MD29 Reducing speed 300.00 deg./min

Dig. inp&uts

MD 34 digital inputs and MD 36 input adjustment *
 Input0 External block change
 Signal inverted No
 Input1 Ref. point switch for ref. point approach
 Signal inverted No
 Input2 Reversing switch for ref. point approach
 Signal inverted No
 Input3 Input not used
 Signal inverted No

Dig. outputs

MD35 Digital outputs *
 Output0 Output not used
 Output1 Output not used
 Output2 Output not used
 Output3 Output not used

Interpolation data

MD39 Start/stop frequency 1000 Hz
 MD46 Interval between 2 pos. tasks. 2 ms
 MD47 Min. traverse time at constant frequency 2 ms
 MD40 Frequency value for acceleration changeover 10000 Hz
 MD42 Acceleration 1 100000 Hz/s
 MD43 Acceleration 2 100000 Hz/s
 MD44 Deceleration 1 100000 Hz/s
 MD45 Deceleration 2 100000 Hz/s
 MD56 Travel profile Standard

Drive data

MD11/12 Displacement per motor revolutions * 360.0000000000000000 deg.
 MD13 Steps per motor rev. (IMS) * 400
 MD38 No. of steps per motor current cycle * 20
 MD41 Maximum frequency 40000 Hz
 MD37.17-19 Phase current control of drive * Pulse-width modulation(PWM) active
 PWM inverted No
 MD48 Absolute boost time 100 ms

Fig. 20. FM 353 Motor 1: Resumo da parametrização efetuada (1 de 2)

MD49 Relative boost time	100 %
MD50 Phase current in operation	100 %
MD51 Phase current, idling	100 %

Drive interface

MD37 Special control signals *	
Controller enable active	No
Controller ready active	No
Pulse output inverted	No
Direction output inverted	No
MD30 Backlash compensation	0.00 deg.
MD31 Directional ref., backlash	Like ref. point approach
MD52 Speed for backlash compensation	0 %
MD53 Mode for backlash compensation	befor positioning

S7 interface

MD32 M function output type *	During pos.: time-controlled
MD33 M function output time *	10 ms
MD5 Process interrupt generation	
Position reached	No
Length measurement end	No
Block change on-the-fly	No
Meas. on-the-fly	No
MD37.15 Continue after emergency stop *	No
MD37.7 Time override active *	No

* Changing and activating this data causes axis to be reset.

Fig. 21. FM 353 Motor 1: Resumo da parametrização efetuada (2 de 2)

Anexo E: Parametrização do módulo FM 353 – eixo 2

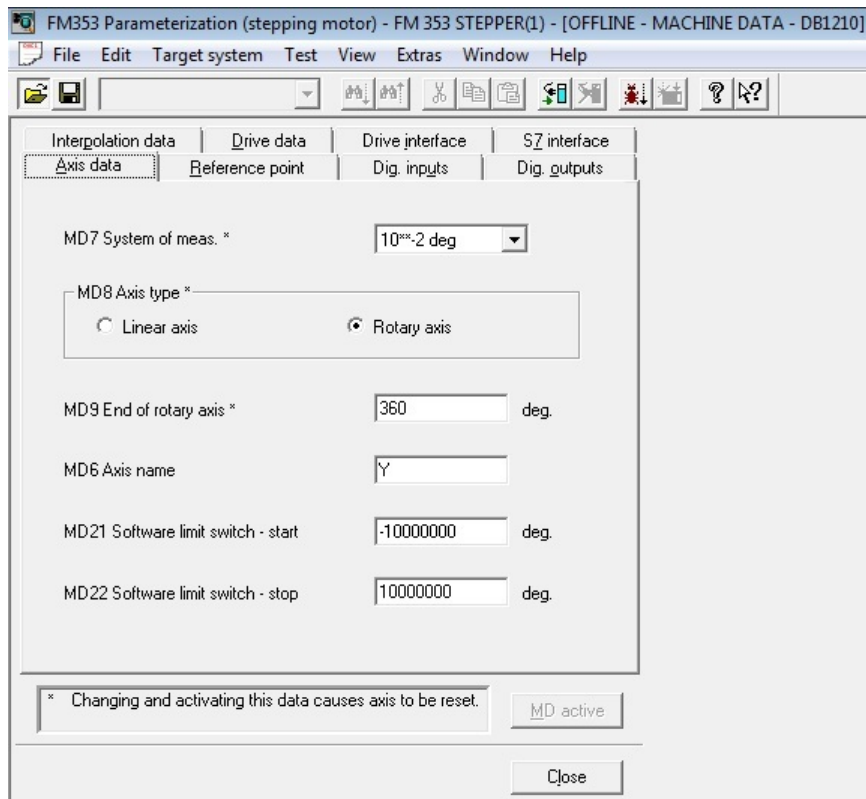


Fig. 22. FM 353 Motor 2: “Axis data”

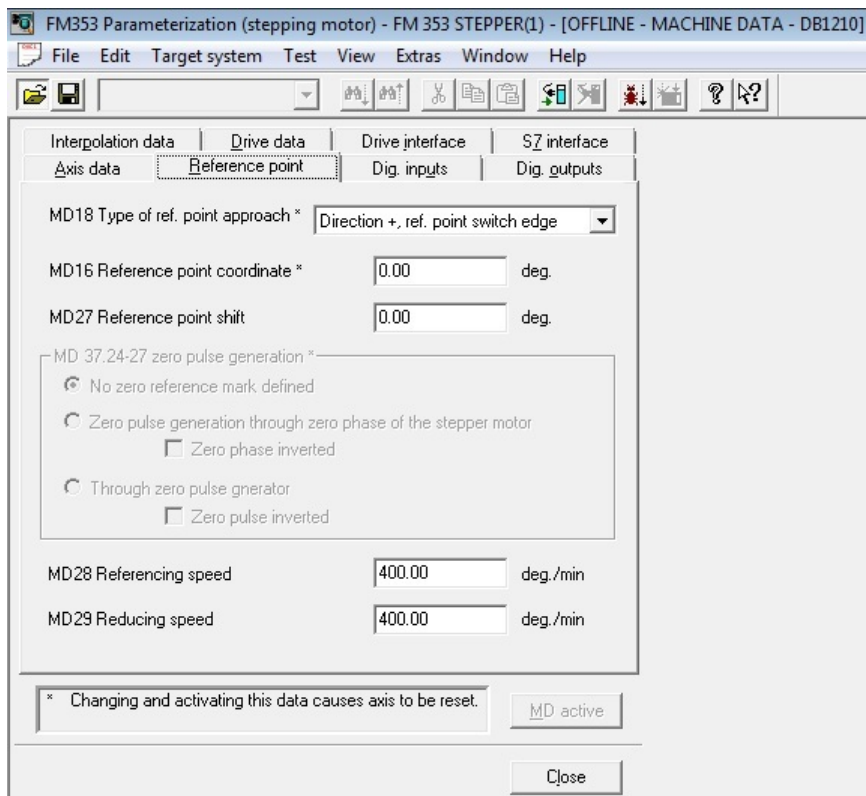


Fig. 23. FM 353 Motor 2: “Reference point”

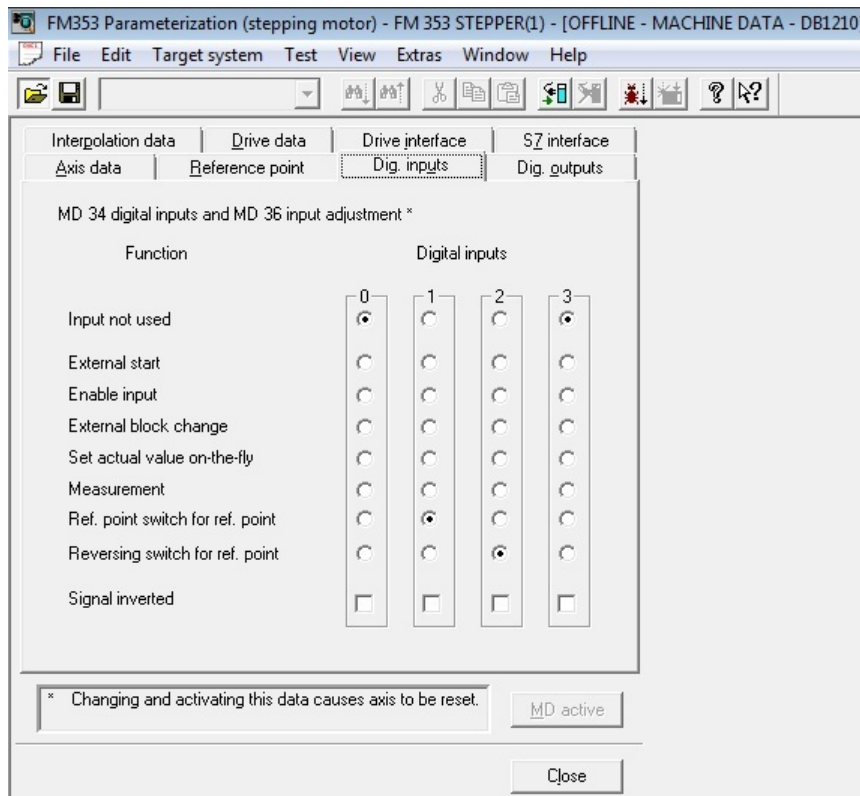


Fig. 24. FM 353 Motor 2: "Digital inputs"

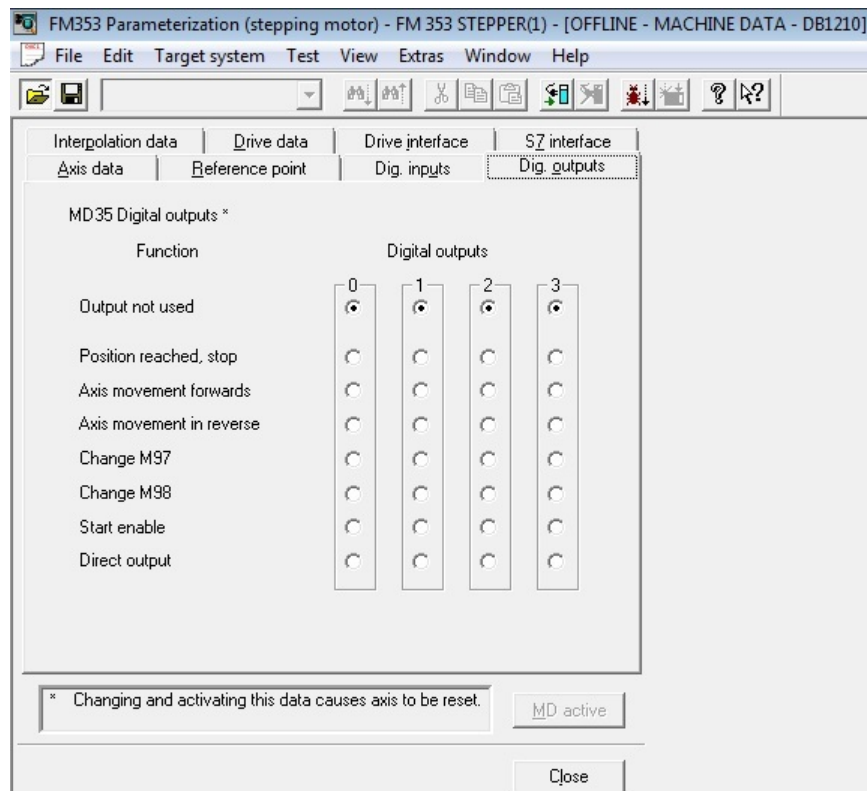


Fig. 25. FM 353 Motor 2: "Digital outputs"

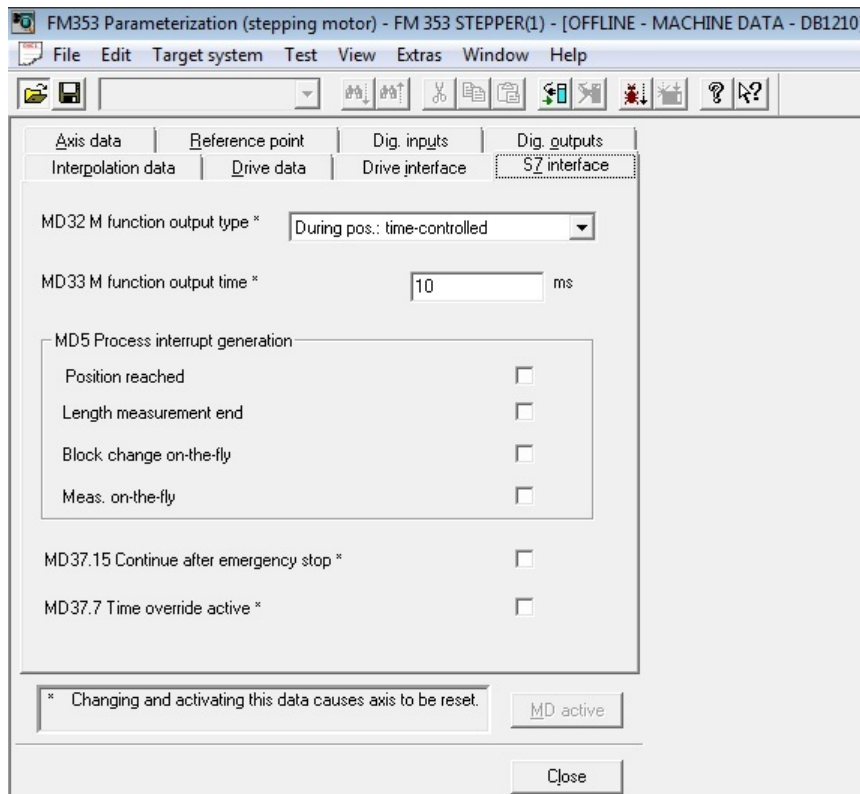


Fig. 26. FM 353 Motor 2: “SZ_interface”

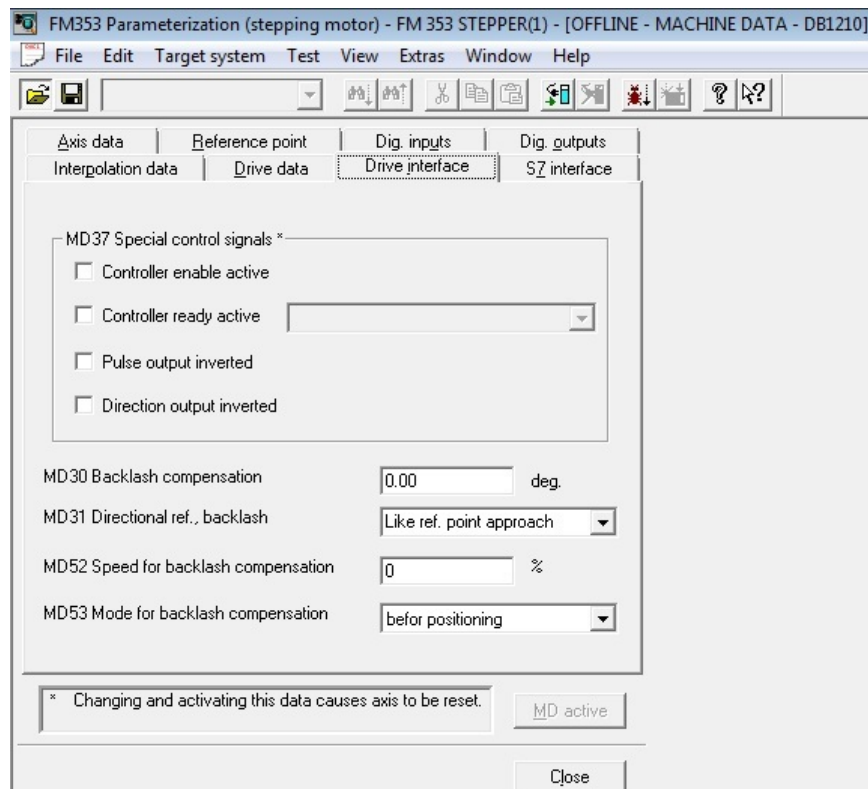


Fig. 27. FM 353 Motor 2: “Drive interface”

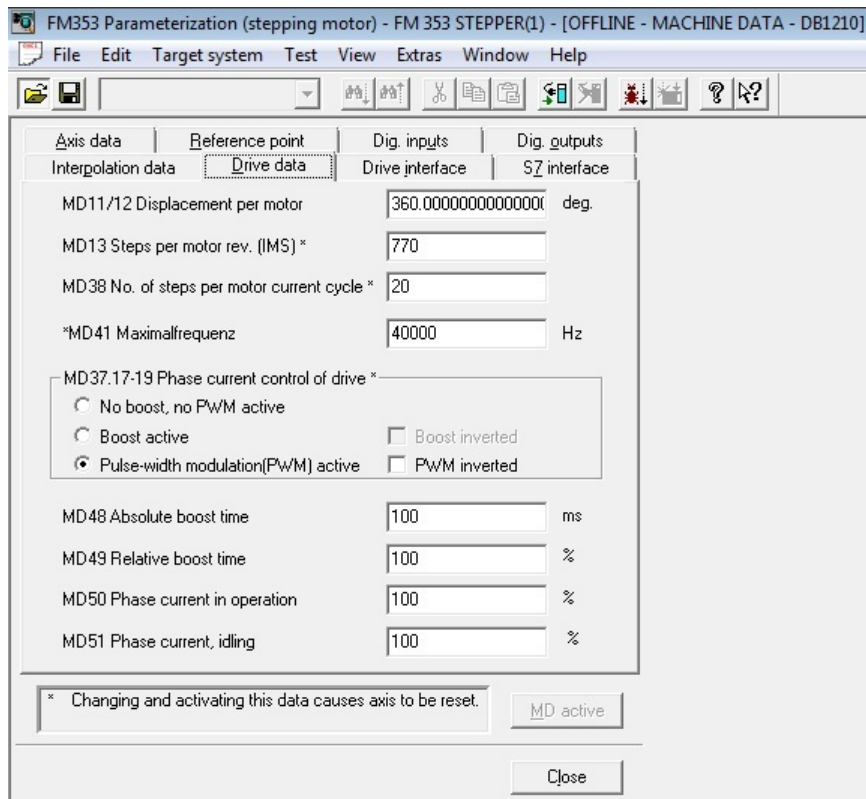


Fig. 28. FM 353 Motor 2: “Drive data”

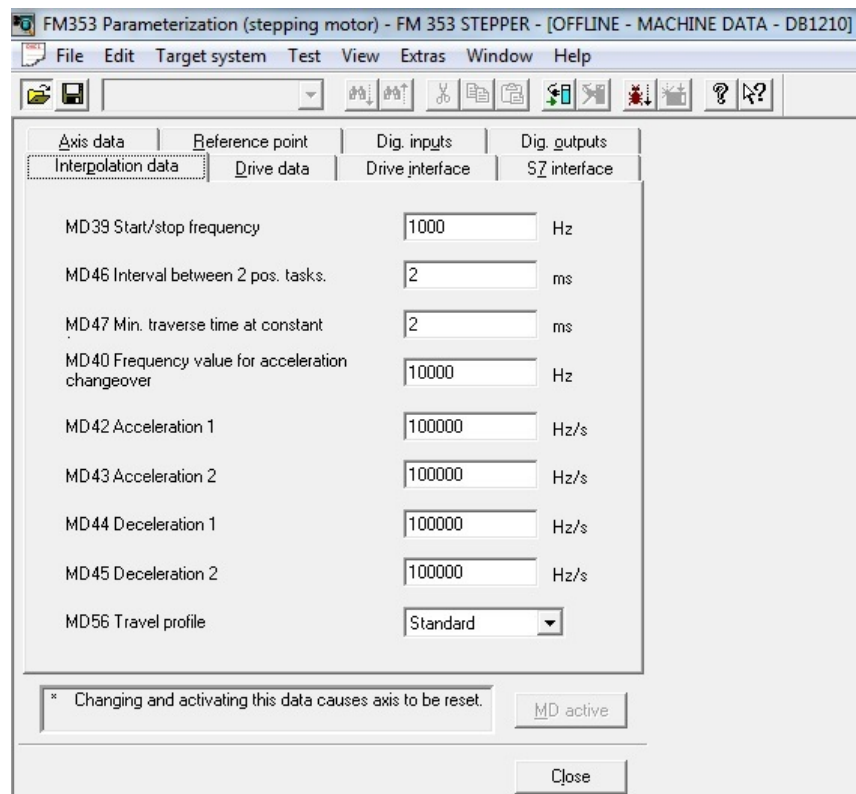


Fig. 29. FM 353 Motor 2: “Interpolation data”

FM353 Parameterization (stepping motor)

=====

Properties

MPI address.....5
Address / input.....368

Measure.....10**⁻² deg

Axis data

MD7 System of meas. * 10**⁻² deg
MD8 Axis type * Rotary axis
MD9 End of rotary axis * 360 deg.
MD6 Axis name Y
MD21 Software limit switch - start -10000000 deg.
MD22 Software limit switch - stop 10000000 deg.

Reference point

MD18 Type of ref. point approach * Direction +, ref. point switch edge
MD16 Reference point coordinate * 0.00 deg.
MD27 Reference point shift 0.00 deg.
MD 37.24-27 zero pulse generation * No zero reference mark defined
MD28 Referencing speed 400.00 deg./min
MD29 Reducing speed 400.00 deg./min

Dig. inp&uts

MD 34 digital inputs and MD 36 input adjustment *
Input0 Input not used
Signal inverted No
Input1 Ref. point switch for ref. point approach
Signal inverted No
Input2 Reversing switch for ref. point approach
Signal inverted No
Input3 Input not used
Signal inverted No

Dig. outputs

MD35 Digital outputs *
Output0 Output not used
Output1 Output not used
Output2 Output not used
Output3 Output not used

Interpolation data

MD39 Start/stop frequency 1000 Hz
MD46 Interval between 2 pos. tasks. 2 ms
MD47 Min. traverse time at constant frequency 2 ms
MD40 Frequency value for acceleration changeover 10000 Hz
MD42 Acceleration 1 100000 Hz/s
MD43 Acceleration 2 100000 Hz/s
MD44 Deceleration 1 100000 Hz/s
MD45 Deceleration 2 100000 Hz/s
MD56 Travel profile Standard

Drive data

MD11/12 Displacement per motor revolutions * 360.0000000000000000 deg.
MD13 Steps per motor rev. (IMS) * 770
MD38 No. of steps per motor current cycle * 20
MD41 Maximum frequency 40000 Hz
MD37.17-19 Phase current control of drive * Pulse-width modulation (PWM) active
PWM inverted No
MD48 Absolute boost time 100 ms
MD49 Relative boost time 100 %

Fig. 30. FM 353 Motor 2: Resumo da parametrização efetuada (1 de 2)

MD50 Phase current in operation	100 %
MD51 Phase current, idling	100 %
<u>Drive interface</u>	
MD37 Special control signals *	
Controller enable active	No
Controller ready active	No
Pulse output inverted	No
Direction output inverted	No
MD30 Backlash compensation	0.00 deg.
MD31 Directional ref., backlash	Like ref. point approach
MD52 Speed for backlash compensation	0 %
MD53 Mode for backlash compensation	befor positioning
<u>S7 interface</u>	
MD32 M function output type *	During pos.: time-controlled
MD33 M function output time *	10 ms
MD5 Process interrupt generation	
Position reached	No
Length measurement end	No
Block change on-the-fly	No
Meas. on-the-fly	No
MD37.15 Continue after emergency stop *	No
MD37.7 Time override active *	No

* Changing and activating this data causes axis to be reset.

Fig. 31. FM 353 Motor 2: Resumo da parametrização efetuada (2 de 2)

Anexo F: Diagramas dos varrimentos programados

Sequência de movimentos do Varrimento 1					
N.º	Eixo 1 (graus)		Eixo 2 (graus)		N.º
1	-50		↓	1	-50
				2	0
2	-40	←			
				3	-50
3	-30	←			
				4	0
4	0	←			

Sequência de movimentos do Varrimento 2					
N.º	Eixo 1 (graus)		Eixo 2 (graus)		N.º
1	-30		↓	1	-50
				2	0
2	-20	←			
				3	-50
3	-10	←			
				4	0
4	0	←			

Sequência de movimentos do Varrimento 3					
N.º	Eixo 1 (graus)		Eixo 2 (graus)		N.º
1	30		↓	1	50
				2	0
2	20	←			
				3	50
3	10	←			
				4	0
4	0	←			

Sequência de movimentos do Varrimento 4					
N.º	Eixo 1 (graus)		Eixo 2 (graus)		N.º
1	50		↓	1	-50
				2	0
2	40	←			
				3	-50
3	30	←			
				4	0
4	0	←			

Fig. 32. Diagramas de sequência de movimentos dos varrimentos programados (varrimento 1 até 4)

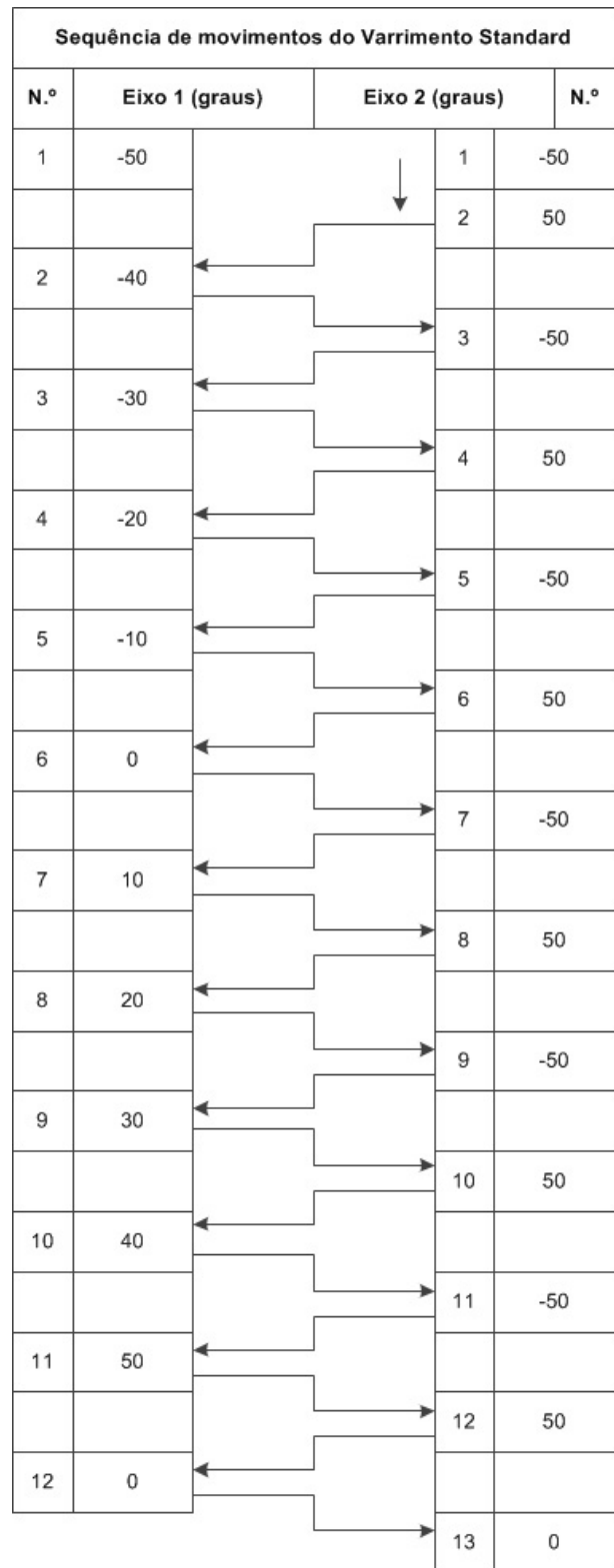
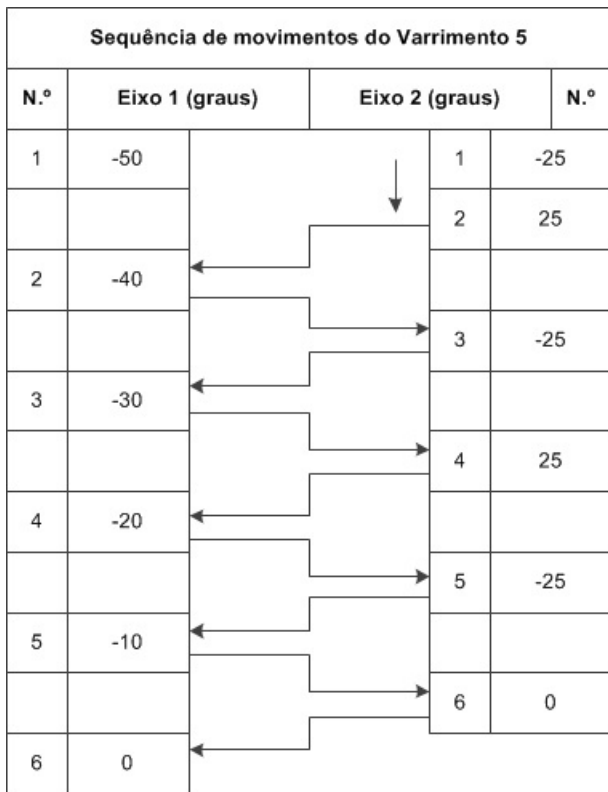


Fig. 33. Diagramas de seqüência de movimentos dos varrimentos programados (varramento 5 e varramento geral)

Anexo G: Programação dos varrimentos (G-Code)

FM353 Parameterization (stepping motor)

=====

Properties

MPI address.....4
 Address / input.....352

 Measure.....10**-3 mm

OFFLINE - TRAVERSE PROGRAM - DB1001

%1
 N1 G90 X310.00 F400.00
 N2 G04 X6000
 N3 G90 X320.00 F400.00
 N4 G04 X5750
 N5 G90 X330.00 F400.00
 N6 G04 X6000
 N7 G90 X0.00 F400.00

OFFLINE - DB1002

%2
 N1 G90 X330.00 F400.00
 N2 G04 X7500
 N3 G90 X340.00 F400.00
 N4 G04 X5000
 N5 G90 X350.00 F400.00
 N6 G04 X6000
 N7 G90 X0.00 F400.00

OFFLINE - DB1003

%3
 N1 G90 X330.00 F400.00
 N2 G04 X7500
 N3 G90 X340.00 F400.00
 N4 G04 X6000
 N5 G90 X350.00 F400.00
 N6 G04 X7000
 N7 G90 X0.00 F400.00

OFFLINE - DB1004

%4
 N1 G90 X310.00 F400.00
 N2 G04 X6500
 N3 G90 X320.00 F400.00
 N4 G04 X6000
 N5 G90 X330.00 F400.00
 N6 G04 X6000
 N7 G90 X0.00 F400.00

OFFLINE - DB1005

%5
 N1 G90 X310.00 F400.00
 N2 G04 X6000
 N3 G90 X320.00 F400.00
 N4 G04 X6000
 N5 G90 X330.00 F400.00
 N6 G04 X6000
 N7 G90 X340.00 F400.00
 N8 G04 X6000
 N9 G90 X350.00 F400.00
 N10 G04 X6000
 N11 G90 X0.00 F400.00

OFFLINE - DB1007

%7
 N1 G90 X310.00 F400.00
 N2 G04 X9600
 N3 G90 X320.00 F400.00
 N4 G04 X10600
 N5 G90 X330.00 F400.00
 N6 G04 X10600
 N7 G90 X340.00 F400.00
 N8 G04 X10600
 N9 G90 X350.00 F400.00
 N10 G04 X10600
 N11 G90 X0.00 F400.00
 N12 G04 X9600
 N13 G90 X10.00 F400.00
 N14 G04 X10600
 N15 G90 X20.00 F400.00
 N16 G04 X9600
 N17 G90 X30.00 F400.00
 N18 G04 X9600
 N19 G90 X40.00 F400.00
 N20 G04 X10600
 N21 G90 X50.00 F400.00
 N22 G04 X11600
 N23 G90 X0.00 F400.00

Fig. 34. FM 353: Programação da sequência de movimentos do motor 1

FM353 Parameterization (stepping motor)

Properties

MPI address.....5
 Address / input.....368

 Measure.....10**-3 mm

OFFLINE - TRAVERSE PROGRAM - DB1001

%1
 N1 G90 X310.00 F700.00
 N2 G04 X4000
 N3 G90 X0.00 F700.00
 N4 G04 X2500
 N5 G90 X310.00 F700.00
 N6 G04 X3000
 N7 G90 X0.00 F700.00

OFFLINE - DB1002

%2
 N1 G90 X310.00 F700.00
 N2 G04 X2500
 N3 G90 X0.00 F700.00
 N4 G04 X2400
 N5 G90 X310.00 F700.00
 N6 G04 X2400
 N7 G90 X0.00 F700.00
 N10 G04 X3000
 N11 G90 X0.00 F700.00

OFFLINE - DB1003

%3
 N1 G90 X50.00 F700.00
 N2 G04 X3000
 N3 G90 X0.00 F700.00
 N4 G04 X2400
 N5 G90 X50.00 F700.00
 N6 G04 X2400
 N7 G90 X0.00 F700.00

OFFLINE - DB1004

%4
 N1 G90 X50.00 F700.00
 N2 G04 X4000
 N3 G90 X0.00 F700.00
 N4 G04 X3250
 N5 G90 X50.00 F700.00
 N6 G04 X3250
 N7 G90 X0.00 F700.00

OFFLINE - DB1005

%5
 N1 G90 X335.00 F700.00
 N2 G04 X6000
 N3 G90 X25.00 F700.00
 N4 G04 X2750
 N5 G90 X335.00 F700.00
 N6 G04 X3500
 N7 G90 X25.00 F700.00
 N8 G04 X3500
 N9 G90 X335.00 F700.00
 N10 G04 X3500
 N11 G90 X0.00 F700.00

OFFLINE - DB1007

%7
 N1 G90 X310.00 F700.00
 N2 G04 X4000
 N3 G90 X50.00 F700.00
 N4 G04 X3000
 N5 G90 X310.00 F700.00
 N6 G04 X3000
 N7 G90 X50.00 F700.00
 N8 G04 X3000
 N9 G90 X310.00 F700.00
 N10 G04 X3000
 N11 G90 X50.00 F700.00
 N12 G04 X3000
 N13 G90 X310.00 F700.00
 N14 G04 X3000
 N15 G90 X50.00 F700.00
 N16 G04 X3000
 N17 G90 X310.00 F700.00
 N18 G04 X3000
 N19 G90 X50.00 F700.00
 N20 G04 X3000
 N21 G90 X310.00 F700.00
 N22 G04 X3000
 N23 G90 X50.00 F700.00
 N24 G04 X3000
 N25 G90 X0.00 F700.00

Fig. 35. FM 353: Programação da sequência de movimentos do motor 2

Anexo H: Configuração da rede para sistema de supervisão

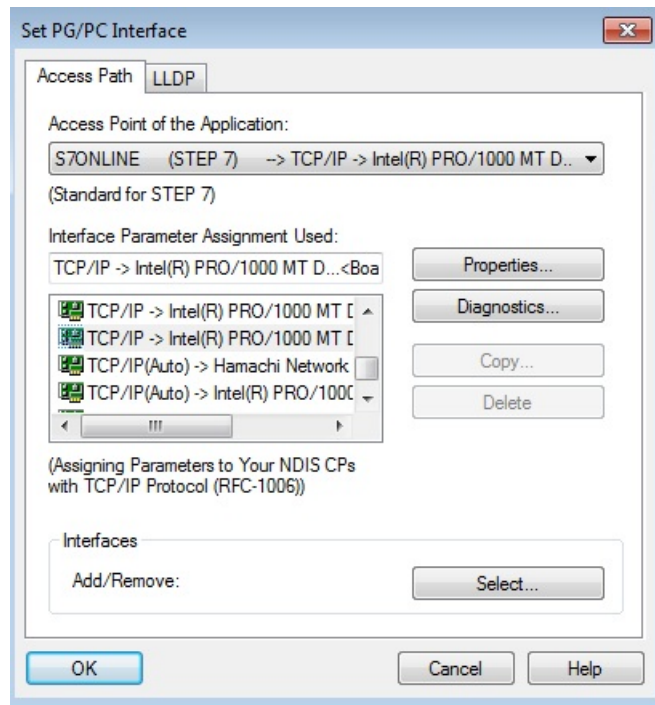


Fig. 36. Seleção da interface de comunicação entre PC e PLC: TPC/IP, de acordo com a ligação de Área Local existente. Notar que o STEP 7 foi instalado em Windows 7, numa máquina virtual (Oracle VirtualBox)

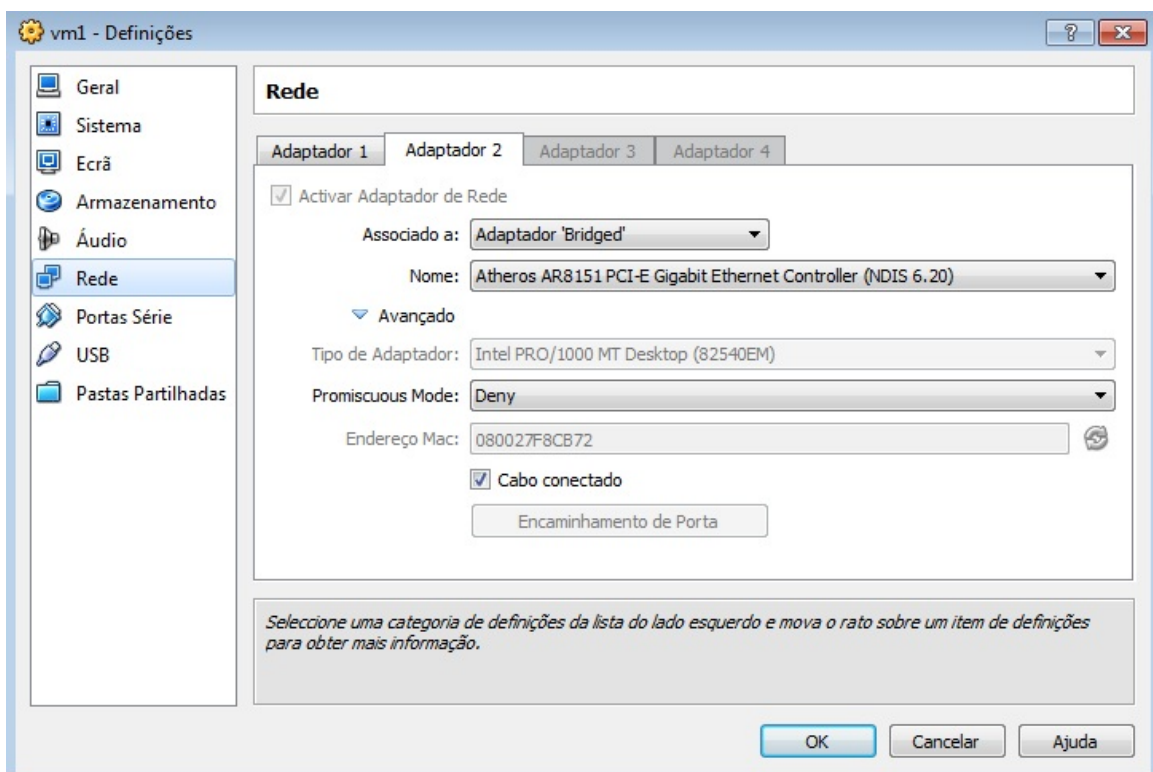


Fig. 37. No menu de configurações da Máquina Virtual atribuir um segundo adaptador, para estabelecer a comunicação entre PC, PLC e WinCC

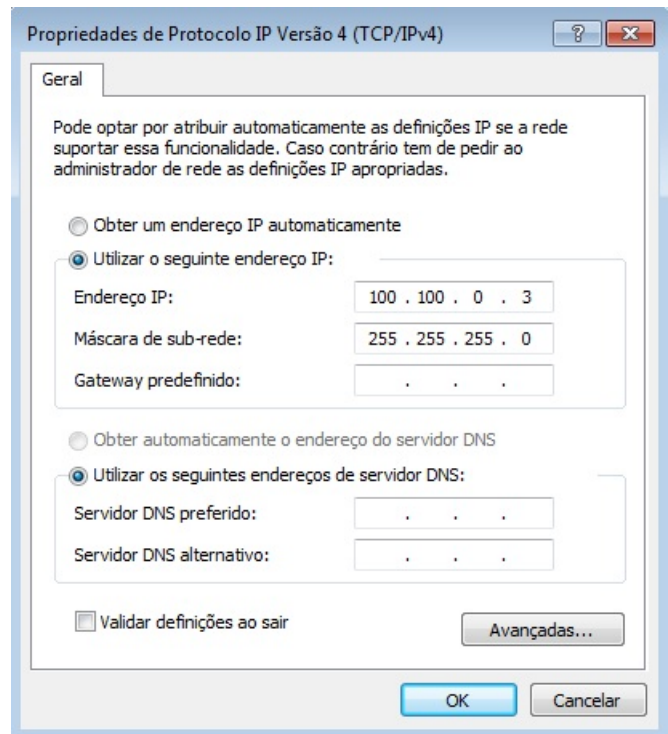
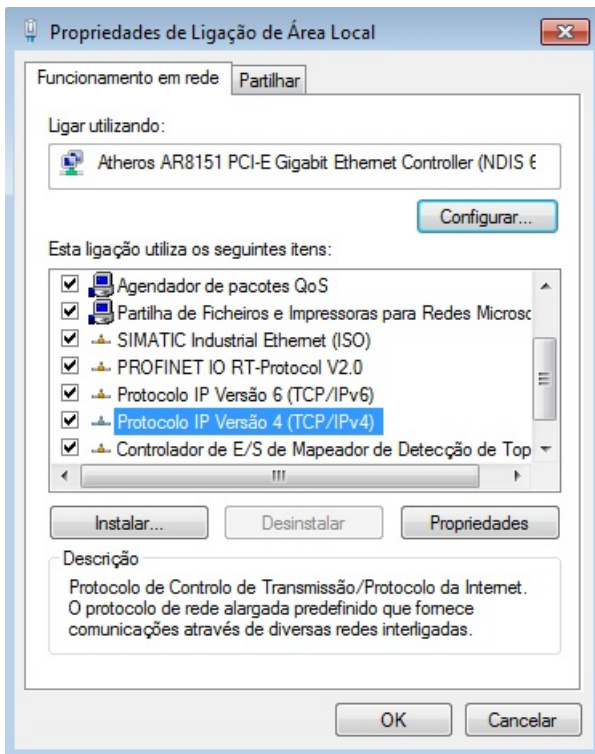


Fig. 38. No sistema operativo nativo, nas propriedades de ligação de área local correspondente, atribuir um IP fixo

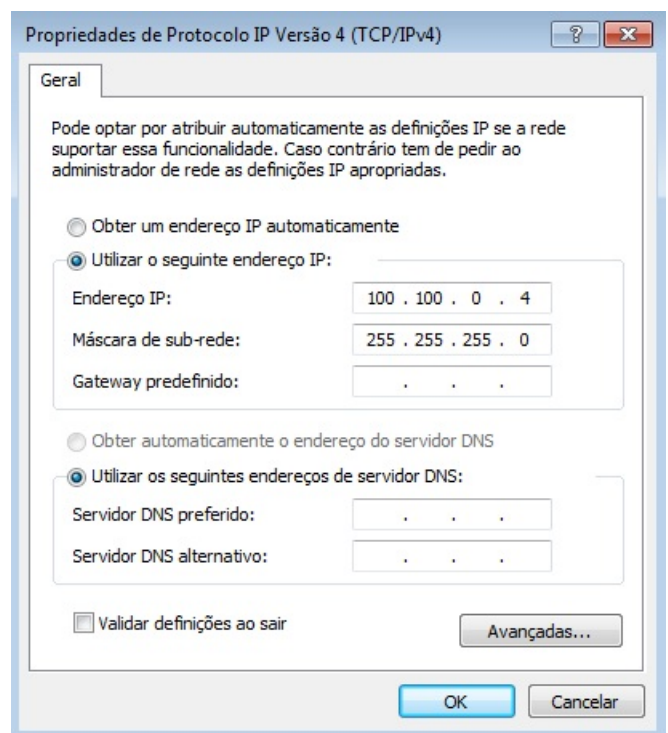
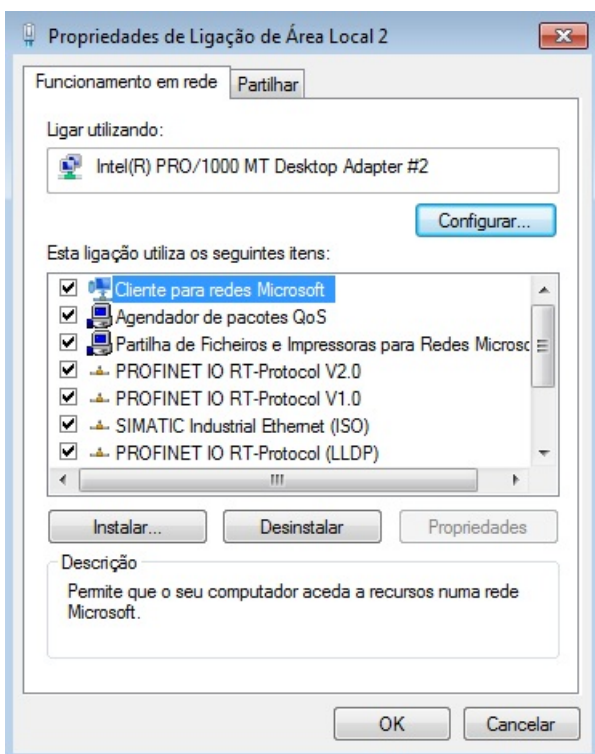


Fig. 39. No sistema operativo da máquina virtual, nas propriedades de ligação de área local correspondente, atribuir um IP fixo. Manter gama de IPs (CP, Sistemas Operativos e WinCC)

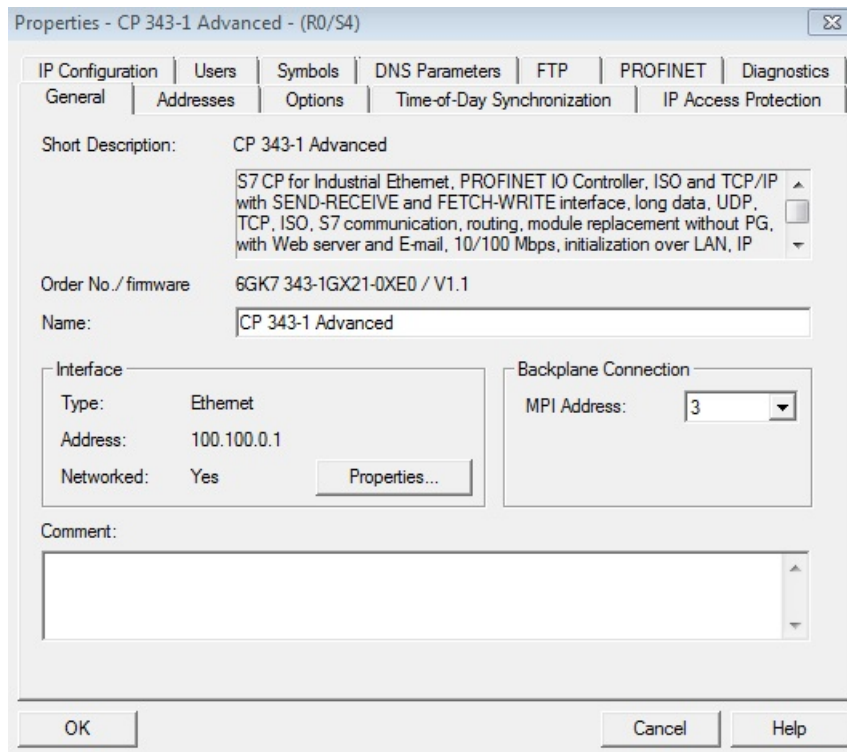


Fig. 40. No STEP 7, em “HW Config”, nas propriedades do módulo de comunicação CP 343-1 Advanced, no separador “General”, clicar no botão “Properties”

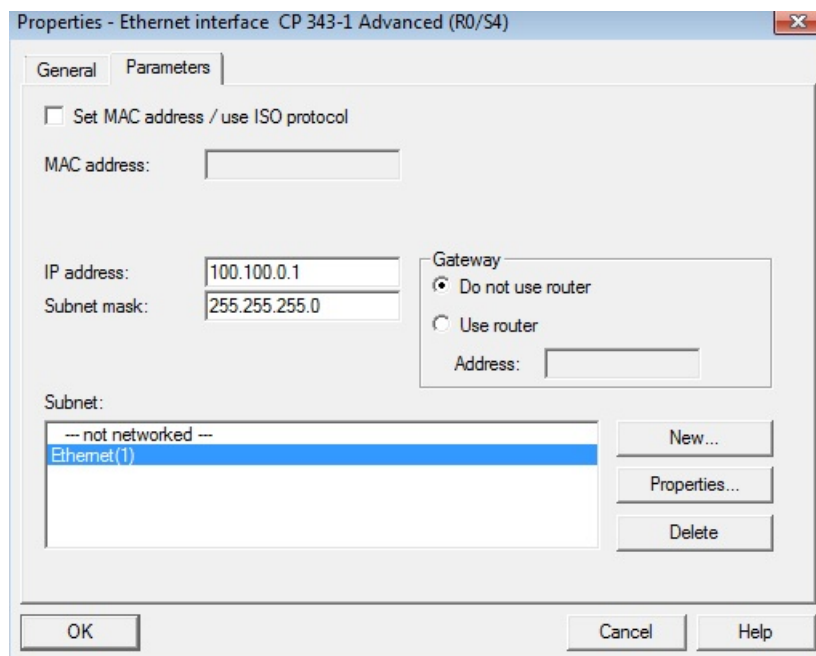


Fig. 41. Criar uma rede Ethernet e atribuir um IP fixo, na mesma gama daqueles que já foram criados

Anexo I: Aplicação para Painel HMI (WinCC Flexible 2008)

Configuração do modo de ligação entre OP270 e S7-300

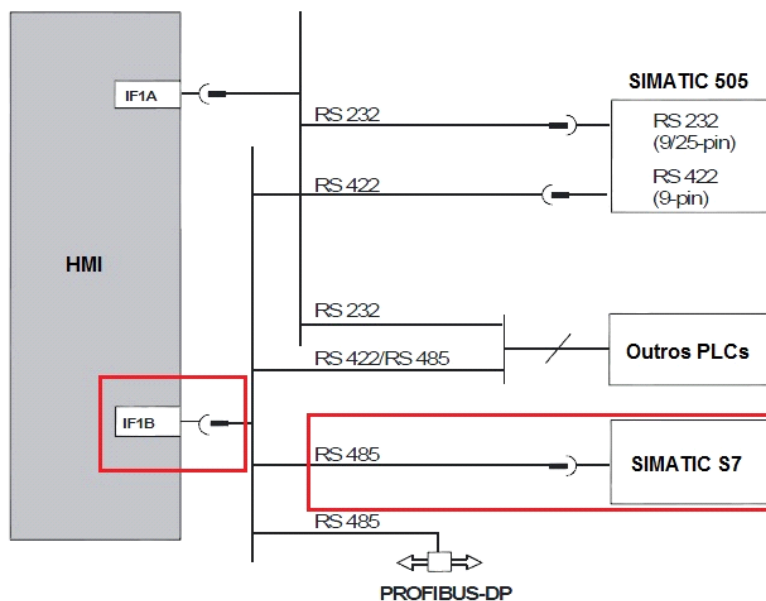


Fig. 42. Configurações possíveis de efetuar entre estes dispositivos e um controlador [25]

A interface IF1B (RS 422/485) configura-se por meio de um switch, existente nesta gama de OPs HMI. Por defeito está configurado para a comunicação com controladores SIMATIC S7.

Configurações de Comunicação	Configuração do Switch
	<p>Configuração por defeito</p>

Fig. 43. Configurações de comunicação entre o OP e PLC, associadas à configuração do switch da interface IF1B [25]

Por forma a configurar a transferência da aplicação desenvolvida no *WinCC Flexible* para o painel, há que seleccionar o canal 2 e atribuir-lhe o tipo de comunicação MPI, no painel de controlo do OP 270 (MPI/DP).

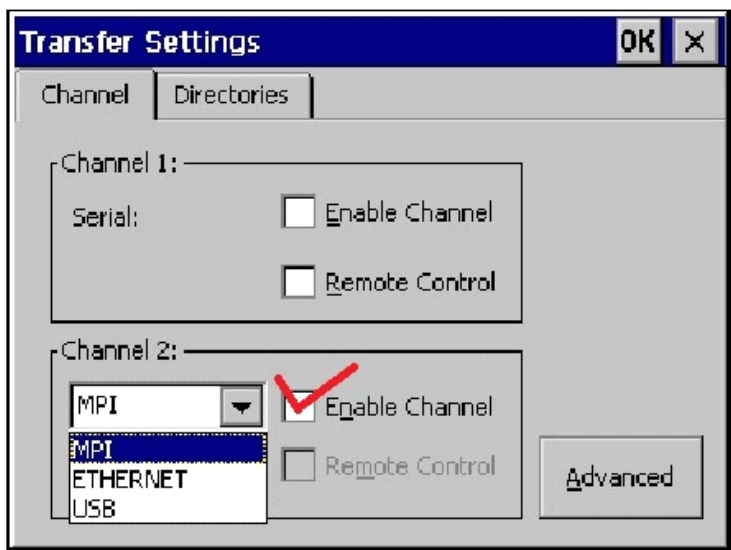


Fig. 44. Configuração de comunicação entre OP 270 e PC [25]

Tutorial para criação da aplicação para painel HMI

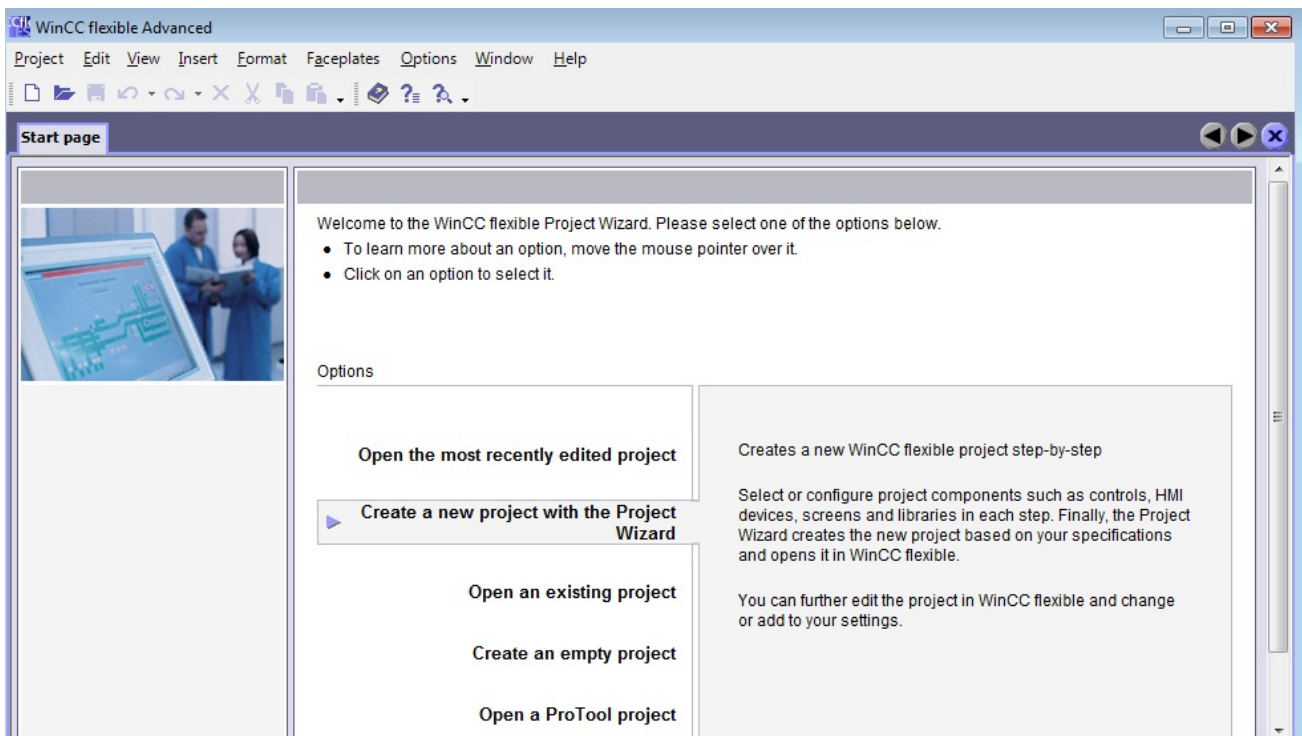


Fig. 45. Correr o “WinCC Flexible 200”8 e criar um novo projeto com ajudante

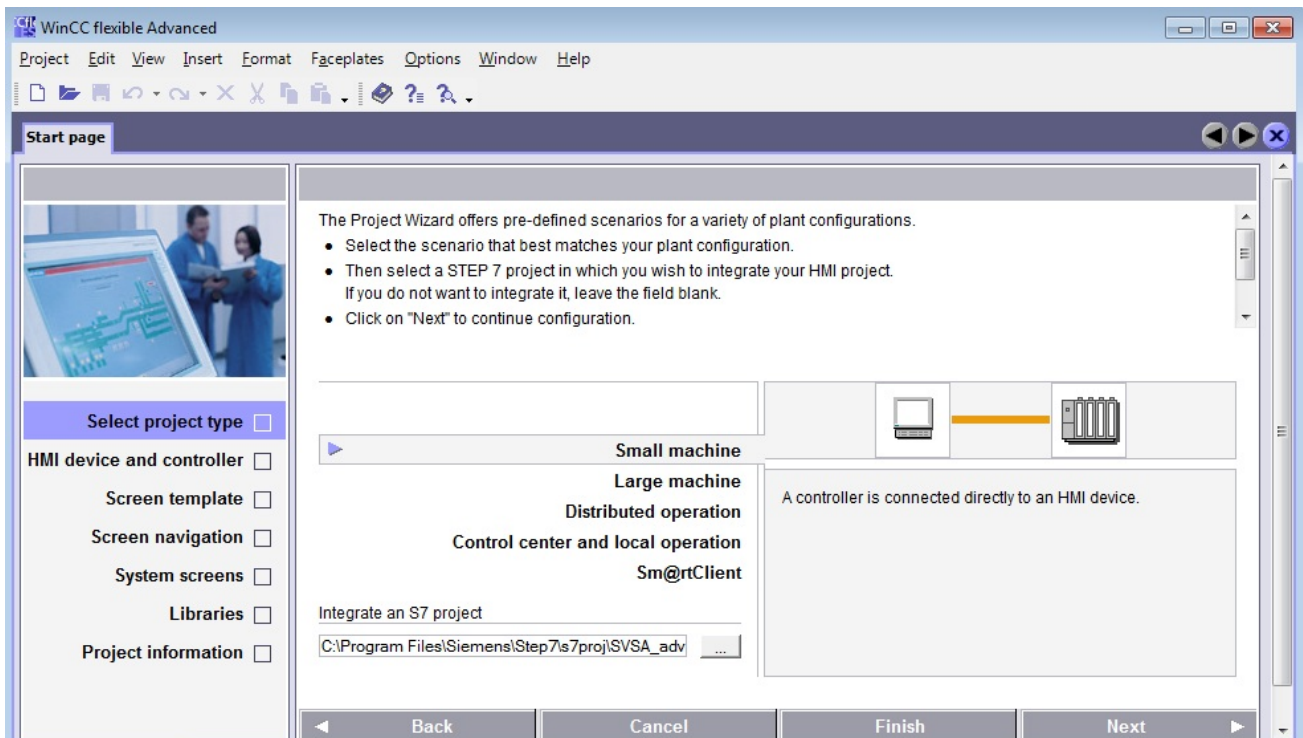


Fig. 46. Selecionar o tipo de projeto (“*Small machine*”) e fazer a integração com o projeto STEP 7

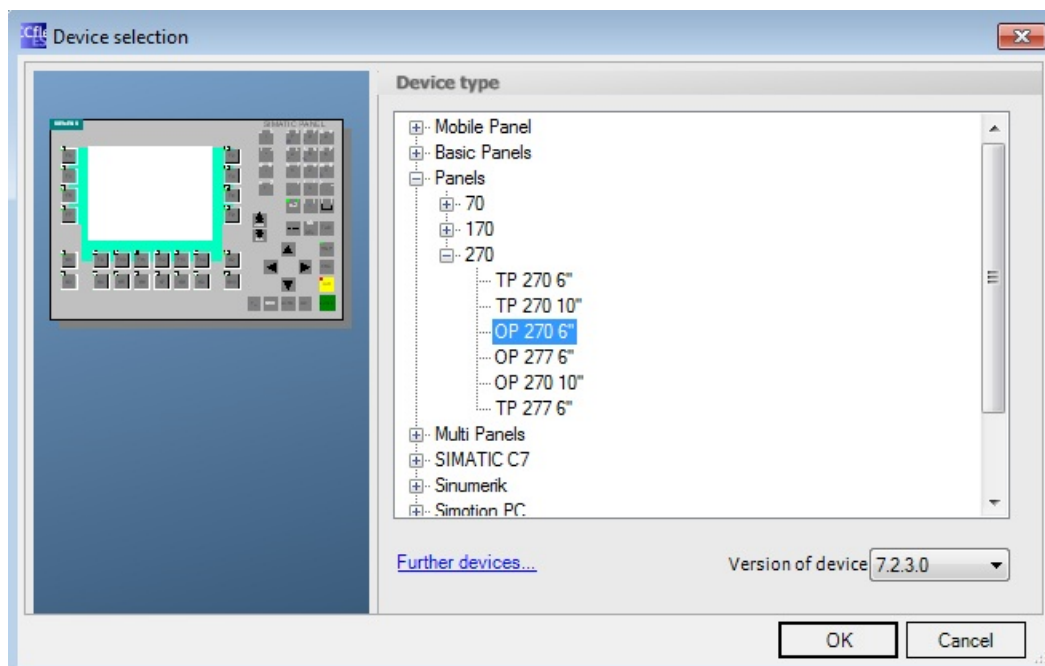


Fig. 47. Selecionar o dispositivo HMI (OP 270’6)

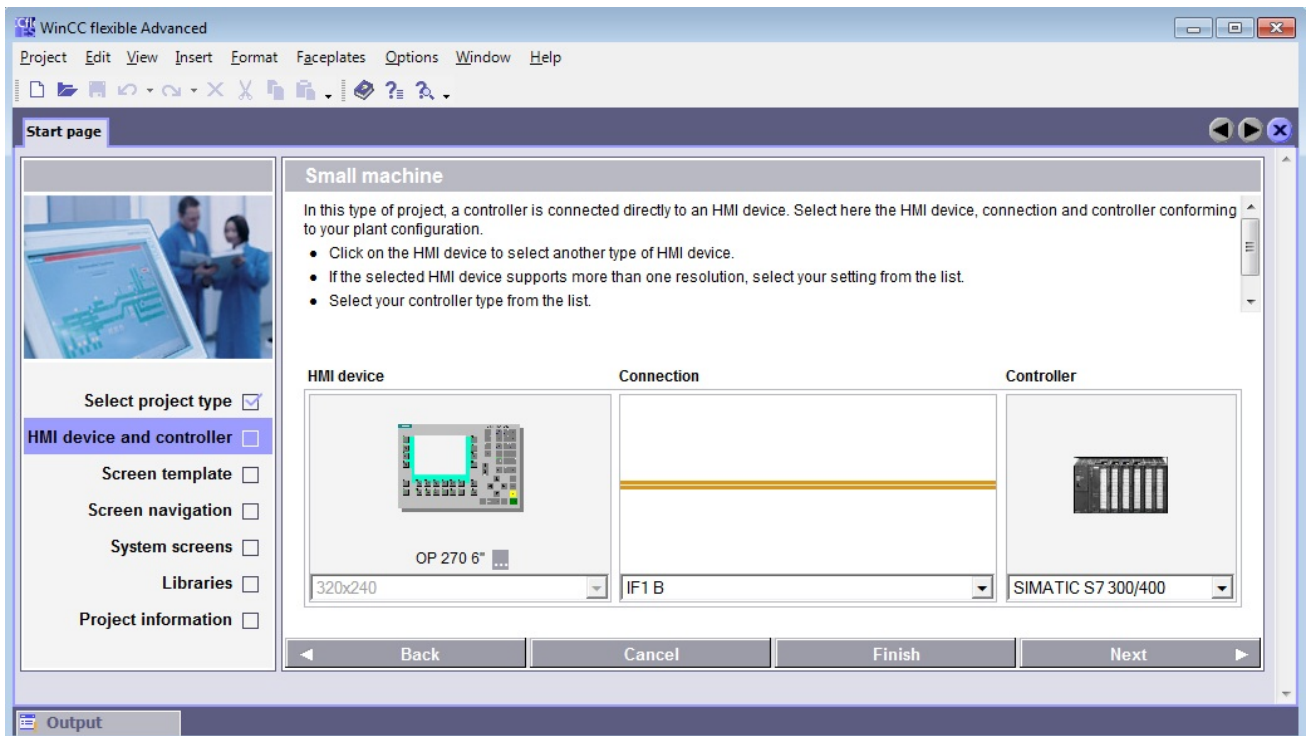


Fig. 48. Observar o tipo de conexão entre o dispositivo HMI e o controlador (comunicação por rede PROFIBUS-DP)

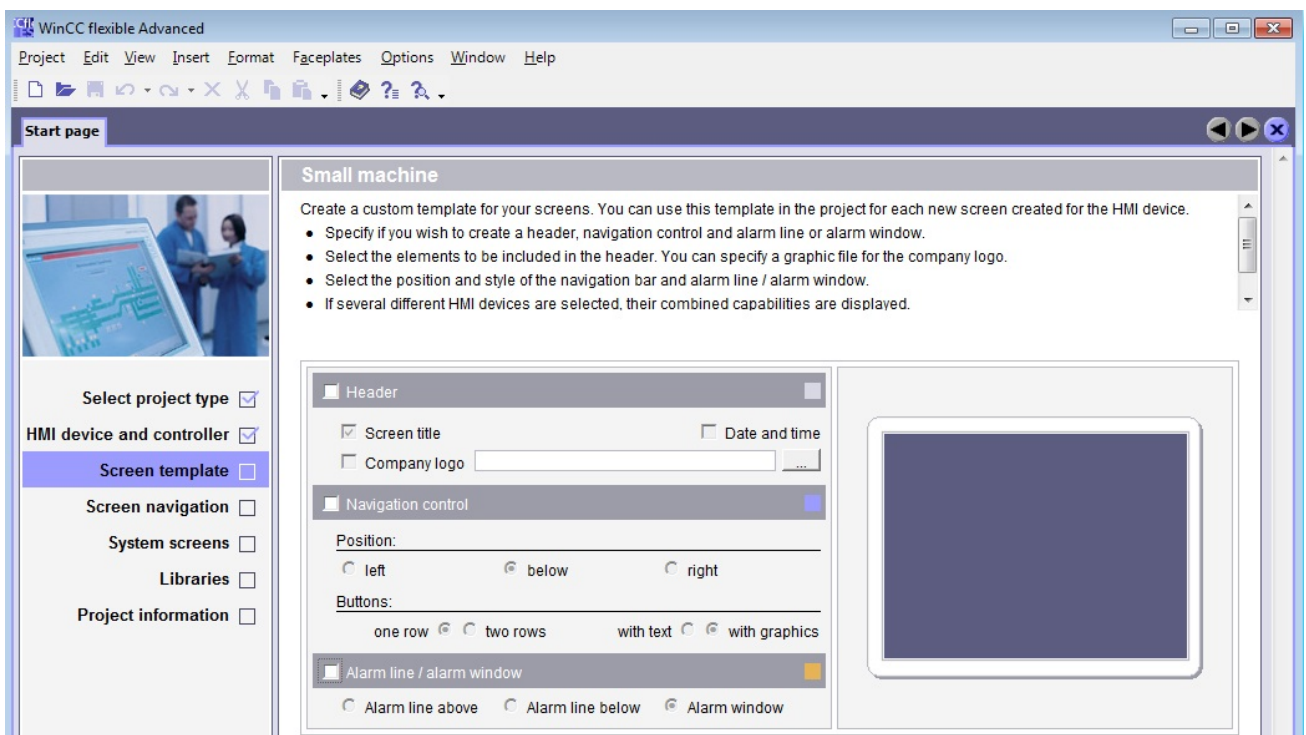


Fig. 49. Selecionar as opções para o ecrã modelo (“*template*”). O que for colocado neste ecrã surge automaticamente nos outros ecrãs adicionados ao projeto

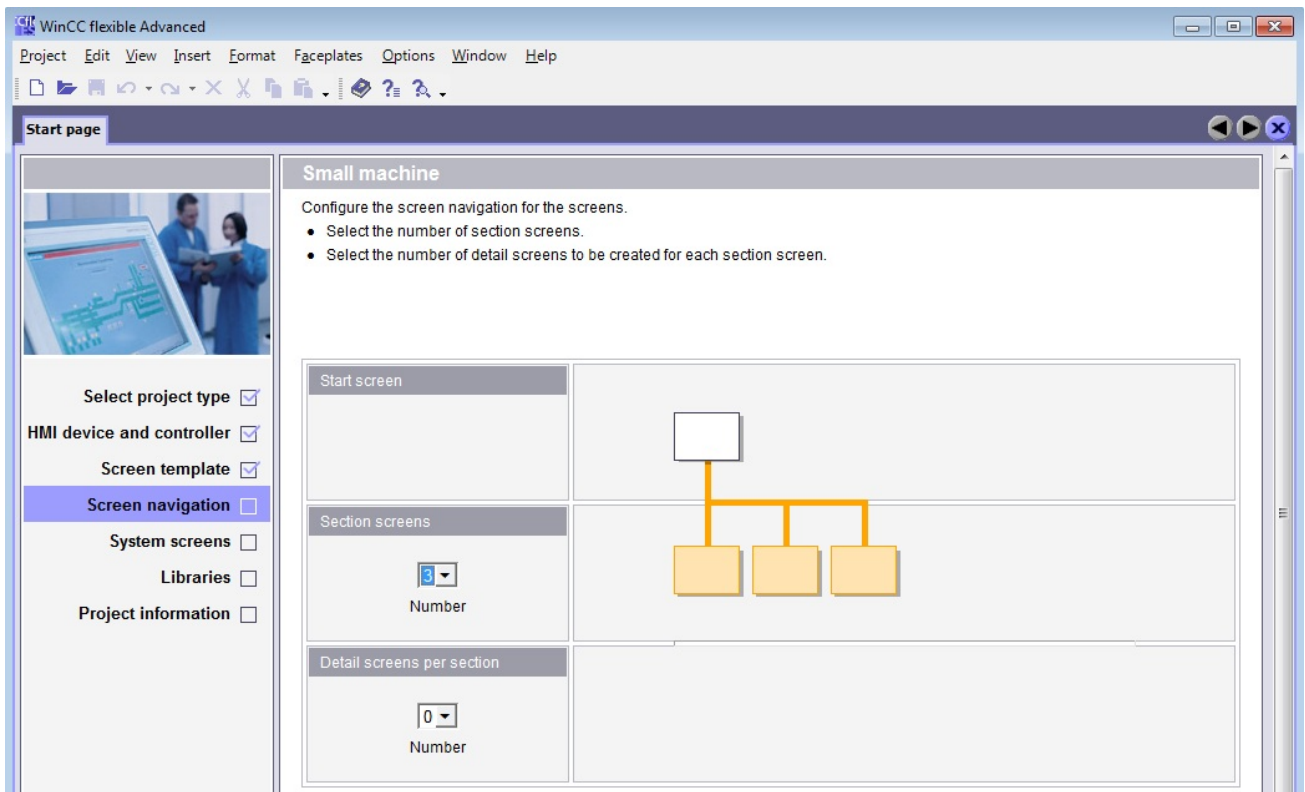


Fig. 50. Selecionar o número de ecrãs utilizados. Notar que podem ser feitas modificações posteriormente

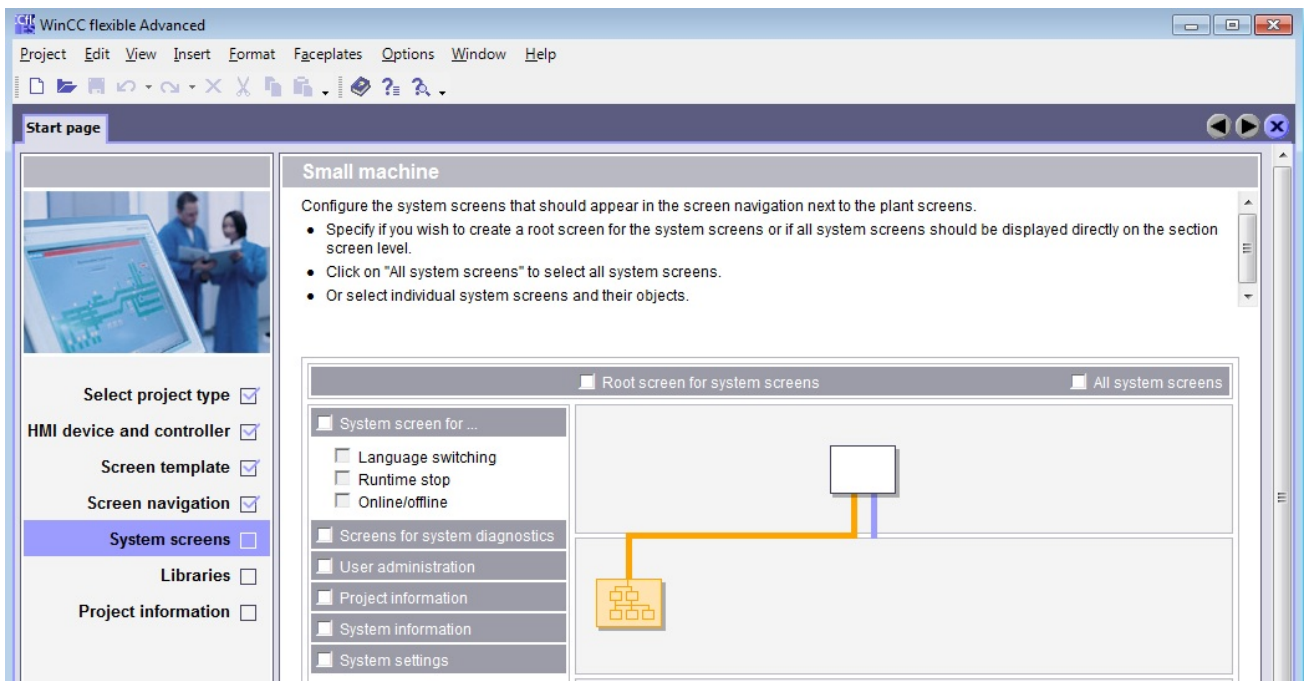


Fig. 51. Indicar, no ecrã de sistema, as funcionalidades pretendidas

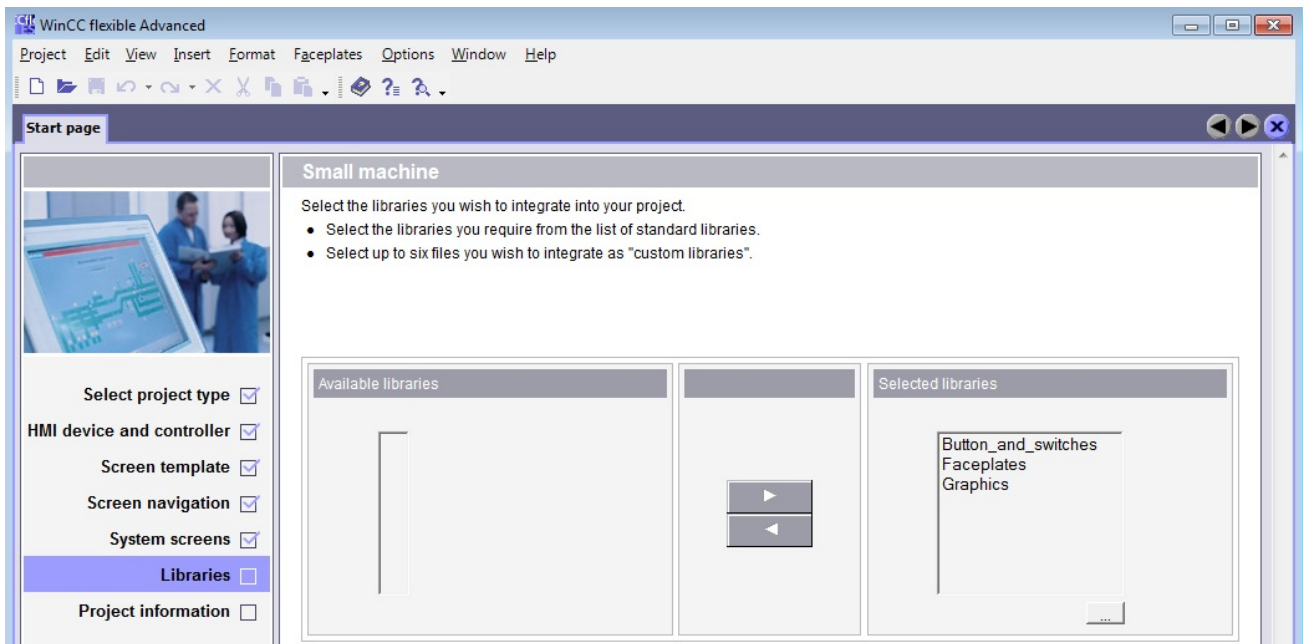


Fig. 52. Selecionar as livrarias às quais se pretende ter acesso aquando da elaboração do projeto

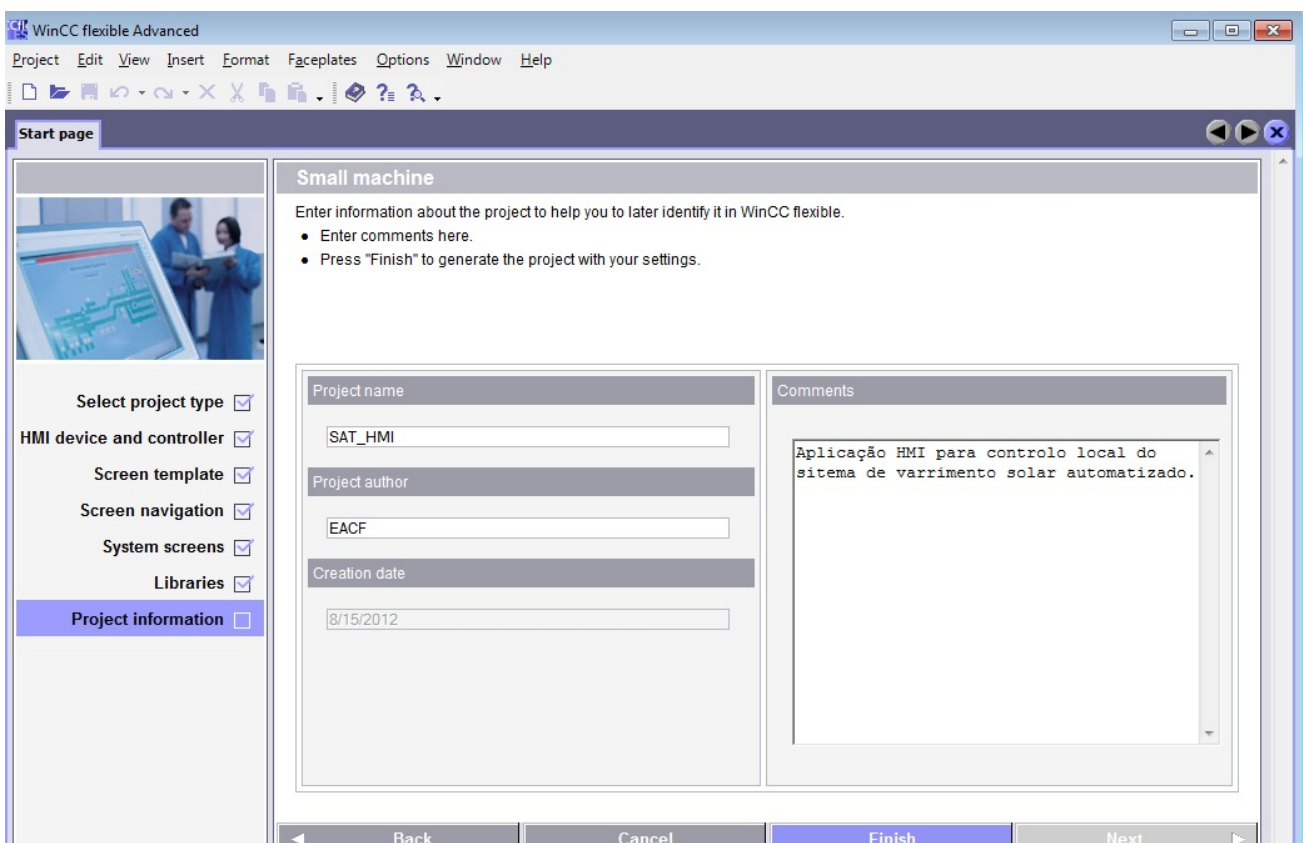


Fig. 53. Por último, atribuir o nome, o autor e os comentários do projeto

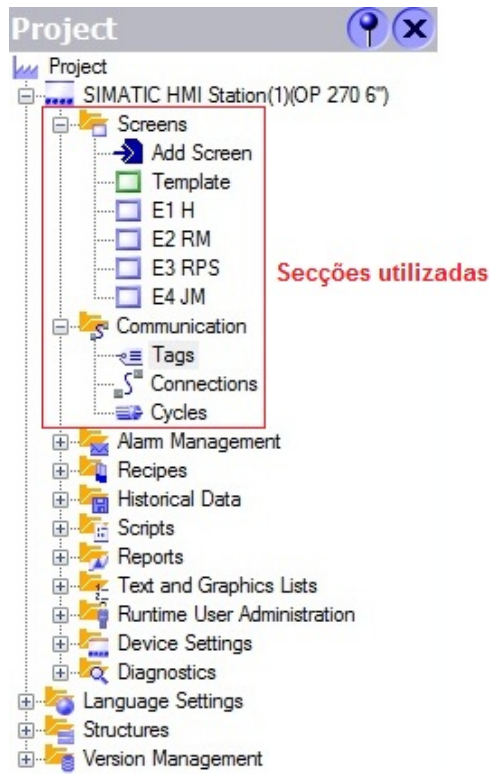


Fig. 54. Surge um projeto com a estrutura em árvores representada

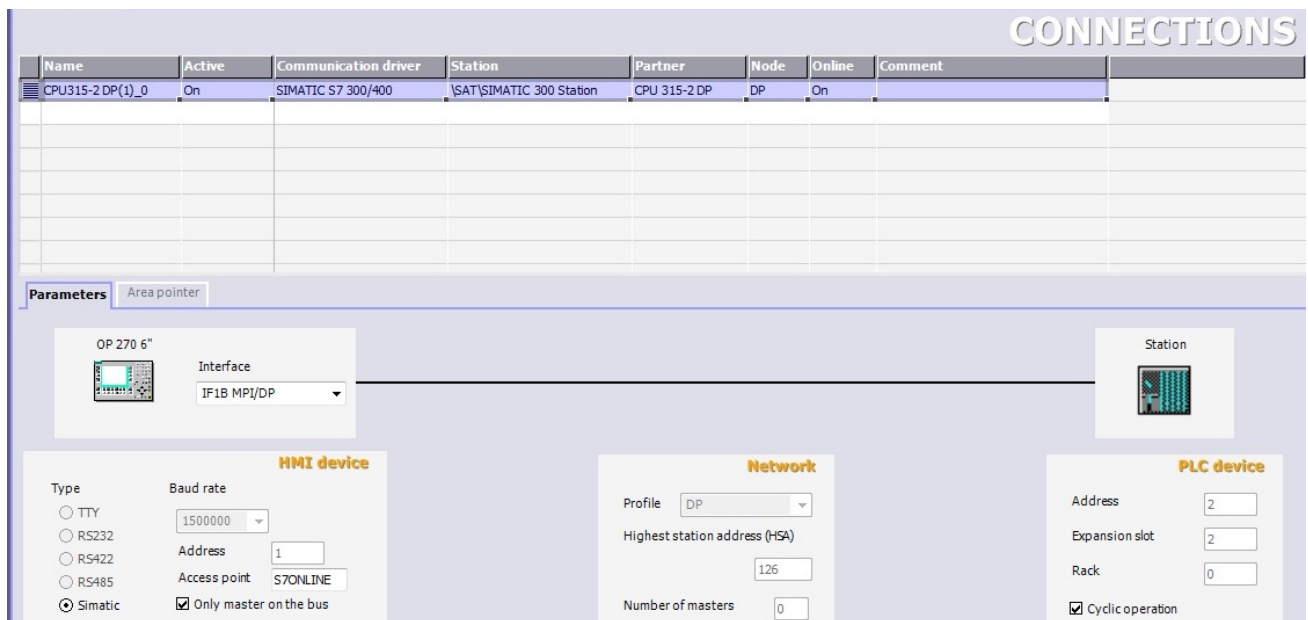


Fig. 55. Opções de comunicação entre o painel e o controlador

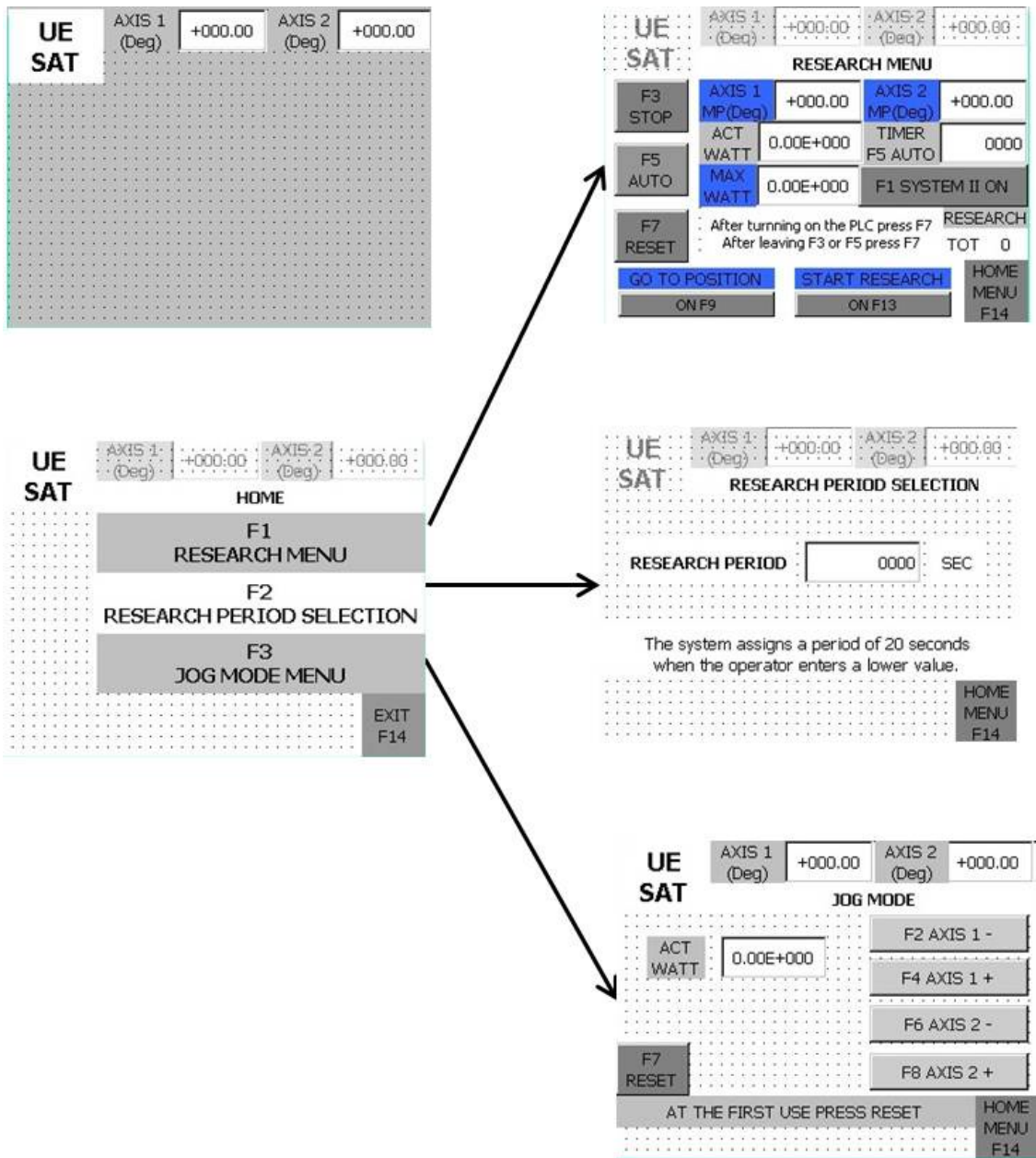


Fig. 56. Organização dos ecrãs da aplicação HMI

	Name	Connection	Data type	Symbol	Address	Array elements	Acquisition cycle
	time_busca	CPU315-2 DP(1)_0	Timer	time_busca	T 14	1	100 ms
	reset_m	CPU315-2 DP(1)_0	Bool	reset_m	M 7.1	1	100 ms
	auto go	CPU315-2 DP(1)_0	Bool	auto go	M 250.0	1	500 ms
	time_auto	CPU315-2 DP(1)_0	Bool	time_auto	M 6.4	1	100 ms
	stop_m	CPU315-2 DP(1)_0	Bool	stop_m	M 7.0	1	100 ms
	mem_research	CPU315-2 DP(1)_0	Bool	mem_research	M 248.0	1	500 ms
	research_num	CPU315-2 DP(1)_0	Int	research_num	MW 80	1	500 ms
	pot	CPU315-2 DP(1)_0	Real	pot	MD 35	1	100 ms
	RESEARCH_MODE	CPU315-2 DP(1)_0	Bool	RESEARCH_MODE	M 26.0	1	500 ms
	Counter_value_M2	CPU315-2 DP(1)_0	DInt	Counter_value_M2	MD 48	1	100 ms
	t_hmi	CPU315-2 DP(1)_0	Int	t_hmi	MW 128	1	500 ms
	b1_mem	CPU315-2 DP(1)_0	Int	b1_mem	MW 78	1	500 ms
	TRAVEL_M_M2_m	CPU315-2 DP(1)_0	Bool	TRAVEL_M_M2_m	M 7.4	1	500 ms
	t_in	CPU315-2 DP(1)_0	Int	t_in	MW 120	1	500 ms
	Counter_value	CPU315-2 DP(1)_0	DInt	Counter_value	MD 44	1	100 ms
	potact3	CPU315-2 DP(1)_0	Real	potact3	MD 65	1	100 ms
	Counter_v11	CPU315-2 DP(1)_0	DInt	Counter_v11	MD 158	1	100 ms
	b7_ok	CPU315-2 DP(1)_0	Bool	b7_ok	M 69.7	1	500 ms
	TRAVEL_P_M2_m	CPU315-2 DP(1)_0	Bool	TRAVEL_P_M2_m	M 7.5	1	500 ms
	auto pulse	CPU315-2 DP(1)_0	Bool	auto pulse	M 245.0	1	500 ms
	mem_auto	CPU315-2 DP(1)_0	Bool	mem_auto	M 248.1	1	500 ms
	zero pulse	CPU315-2 DP(1)_0	Bool	zero pulse	M 245.1	1	100 ms
	sis2_mem	CPU315-2 DP(1)_0	Bool	sis2_mem	M 69.0	1	500 ms
	time_pulse	CPU315-2 DP(1)_0	Bool	time_pulse	M 6.7	1	100 ms
	go zero busca	CPU315-2 DP(1)_0	Bool	go zero busca	M 250.1	1	100 ms
	Counter_v22	CPU315-2 DP(1)_0	DInt	Counter_v22	MD 166	1	100 ms
	mw_volt	CPU315-2 DP(1)_0	Int	mw_volt	MW 100	1	100 ms
	TRAVEL_M_M1_m	CPU315-2 DP(1)_0	Bool	TRAVEL_M_M1_m	M 7.2	1	500 ms
	TRAVEL_P_M1_m	CPU315-2 DP(1)_0	Bool	TRAVEL_P_M1_m	M 7.3	1	100 ms

Fig. 57. Lista das variáveis controladas a partir do painel HMI (“Tags”)

Correspondência entre variáveis, funções atribuídas aos objetos do painel HMI e blocos de programação no PLC

Cross-References

+/- Name	Property name	Path	Infotext	Comment
auto pulse		SIMATIC HMI Station(1)/Communication/Tags	M 245.0	
Softkey_F9	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F9	Press:SetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
mem_auto		SIMATIC HMI Station(1)/Communication/Tags	M 248.1	
Switch_2	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
zero pulse		SIMATIC HMI Station(1)/Communication/Tags	M 245.1	

Softkey_F10	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F10	Press:SetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
TRAVEL_P_M2_m		SIMATIC HMI Station(1)/Communication/Tags	M 7.5	
Softkey_F8	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Softkey_F8	Press:SetBit	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Softkey_F8	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F8	Press:SetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Switch_9	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263

potact3		SIMATIC HMI Station(1)/Communication/Tags	MD 65	
IO Field_6	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
IO Field_6	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
Counter_v11		SIMATIC HMI Station(1)/Communication/Tags	MD 158	
IO Field_5	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
b7_ok		SIMATIC HMI Station(1)/Communication/Tags	M 69.7	
IO Field 9	Visibility	SIMATIC HMI Station(1)/Screens/E2	E2 RM	
Text_Field_28		RM		
Text_Field_28	Visibility	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
mw_volt		SIMATIC HMI Station(1)/Communication/Tags	MW 100	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
TRAVEL_M_M1_m		SIMATIC HMI Station(1)/Communication/Tags	M 7.2	
Softkey_F2	Press:SetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F2	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F2	Press:SetBit	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Softkey_F2	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Switch_6	Property/tag association:Process	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	

	value			
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
TRAVEL_P_M1_m		SIMATIC HMI Station(1)/Communication/Tags	M 7.3	
Softkey_F4	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F4	Press:SetBit	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Softkey_F4	Press:SetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F4	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Switch_7	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
Counter_v22		SIMATIC HMI Station(1)/Communication/Tags	MD 166	
IO Field_4	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
sis2_mem		SIMATIC HMI Station(1)/Communication/Tags	M 69.0	
Switch_5	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F1	Press:InvertBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
time_pulse		SIMATIC HMI Station(1)/Communication/Tags	M 6.7	
Softkey_F5	Press:SetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F5	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	

FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263

go zero busca SIMATIC HMI Station(1)/Communication/Tags M 250.1

Softkey_F13	Press:SetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F13	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263

Counter_value SIMATIC HMI Station(1)/Communication/Tags MD 44

IO Field_1	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Template_IO Field_1	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens	TemplateTemplate	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O



FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263

stop_m SIMATIC HMI Station(1)/Communication/Tags M 7.0

Softkey_F3	Press:InvertBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Switch_3	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263

mem_research SIMATIC HMI Station(1)/Communication/Tags M 248.0

Switch_1	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260



FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
research_num		SIMATIC HMI Station(1)/Communication/Tags	MW 80	
IO Field_9	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
time_auto		SIMATIC HMI Station(1)/Communication/Tags	M 6.4	
Softkey_F5	Press:InvertBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Switch_10	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257

FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
time_busca		SIMATIC HMI Station(1)/Communication/Tags	T 14	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
reset_m		SIMATIC HMI Station(1)/Communication/Tags	M 7.1	
Softkey_F7	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F7	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Switch_4	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Switch_4	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F7	Press:SetBit	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Softkey_F7	Press:SetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264

FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
auto go		SIMATIC HMI Station(1)/Communication/Tags	M 250.0	
Softkey_F9	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F13	Press:ResetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F9	Press:SetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
b1_mem		SIMATIC HMI Station(1)/Communication/Tags	MW 78	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
TRAVEL_M_M2_m		SIMATIC HMI Station(1)/Communication/Tags	M 7.4	
Softkey_F6	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Softkey_F6	Release:ResetBit	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Softkey_F6	Press:SetBit	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
Switch_8	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Softkey_F6	Press:SetBit	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
t_in		SIMATIC HMI Station(1)/Communication/Tags	MW 120	
IO Field_2	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E3 RPS	E3 RPS	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
t_hmi		SIMATIC HMI Station(1)/Communication/Tags	MW 128	

IO Field_3	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
pot		SIMATIC HMI Station(1)/Communication/Tags	MD 35	
IO Field_7	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E2 RM	E2 RM	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
RESEARCH_MODE		SIMATIC HMI Station(1)/Communication/Tags	M 26.0	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263
Counter_value_M2		SIMATIC HMI Station(1)/Communication/Tags	MD 48	
IO Field_3	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens/E4 JM	E4 JM	
Template_IO Field_3	Property/tag association:Process value	SIMATIC HMI Station(1)/Screens	TemplateTemplate	
FC102	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC202	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: STL Type: R	NW 2 STA 2 /O
FC8	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 260
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW4 0 264
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: R	NW3 0 257
FC15	M 245.0	/SAT/SIMATIC 300 Station/CPU 315-2 DP	Language: LAD Type: W	NW2 0 263

Fig. 59. Cruzamento de dados entre as variáveis utilizadas (HMI e PLC). Definição das funcionalidades dos objetos colocados nos ecrãs

Anexo J: Sistema de Supervisão em WinCC V7

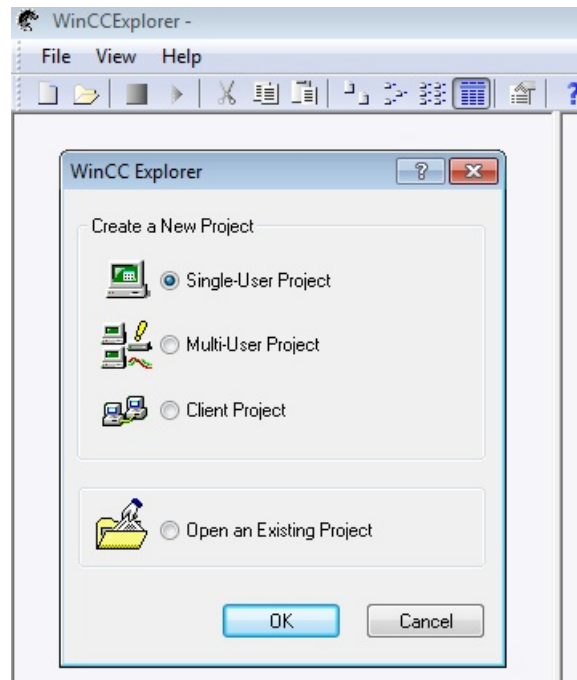


Fig. 60. Criação do projeto WinCC (tipo: “Single-User Project”)

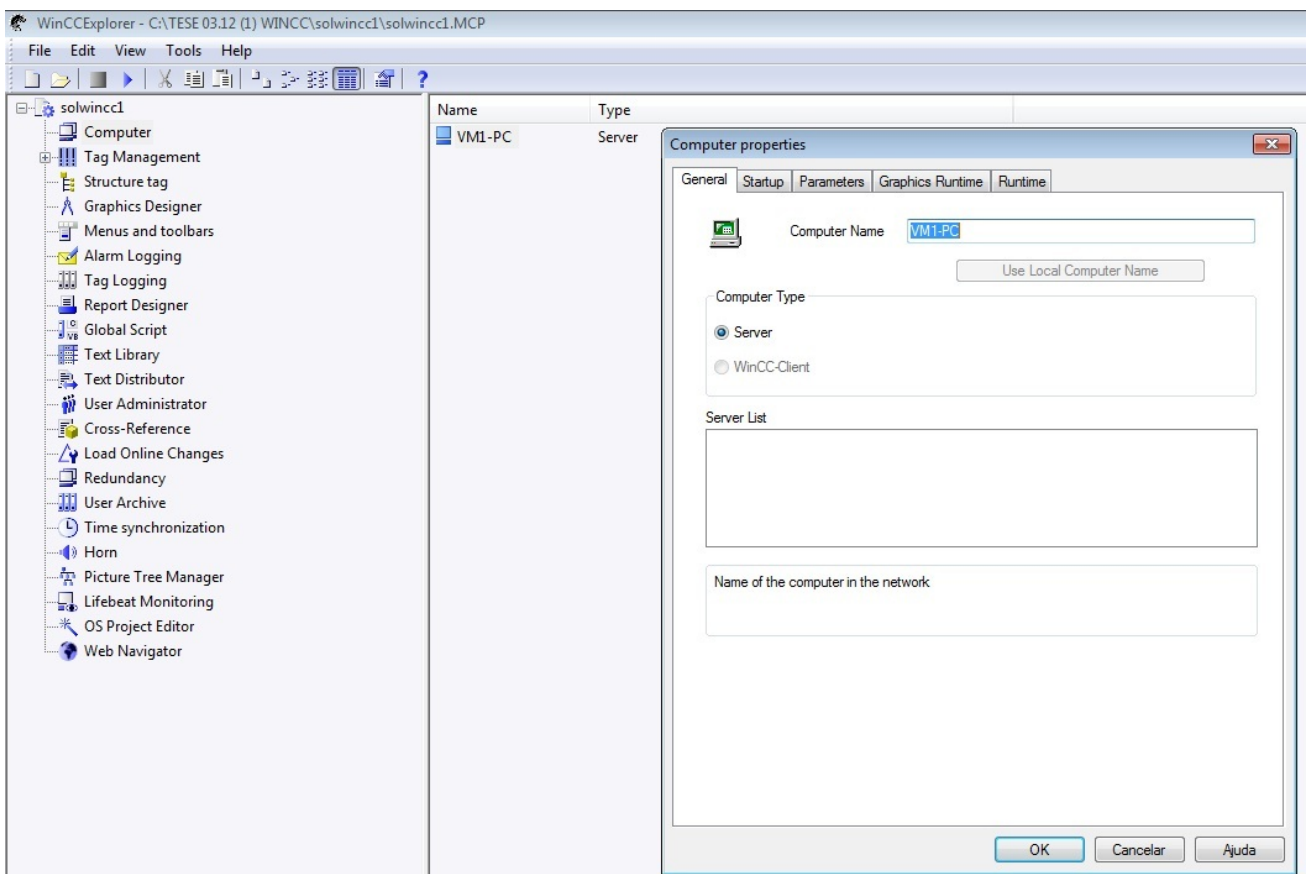


Fig. 61. Na árvore do projeto, selecionar o separador Computador e introduzir o nome do computador (VM1-PC)

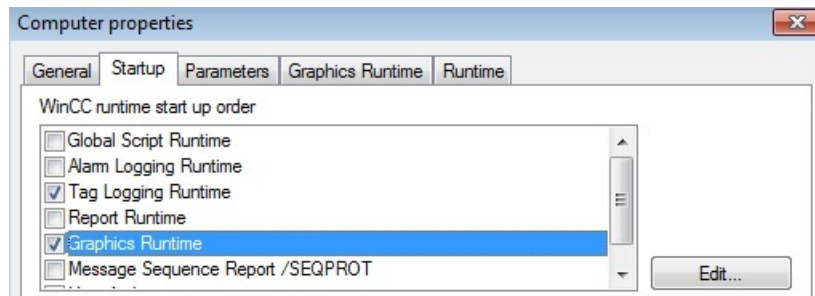


Fig. 62. Nas propriedades do separador Computador, em “Startup”, disponibilizar: “Graphics Runtime” (telas), “Tag Logging runtime” (para visualizar gráficos com atualização permanente das variáveis monitorizadas)

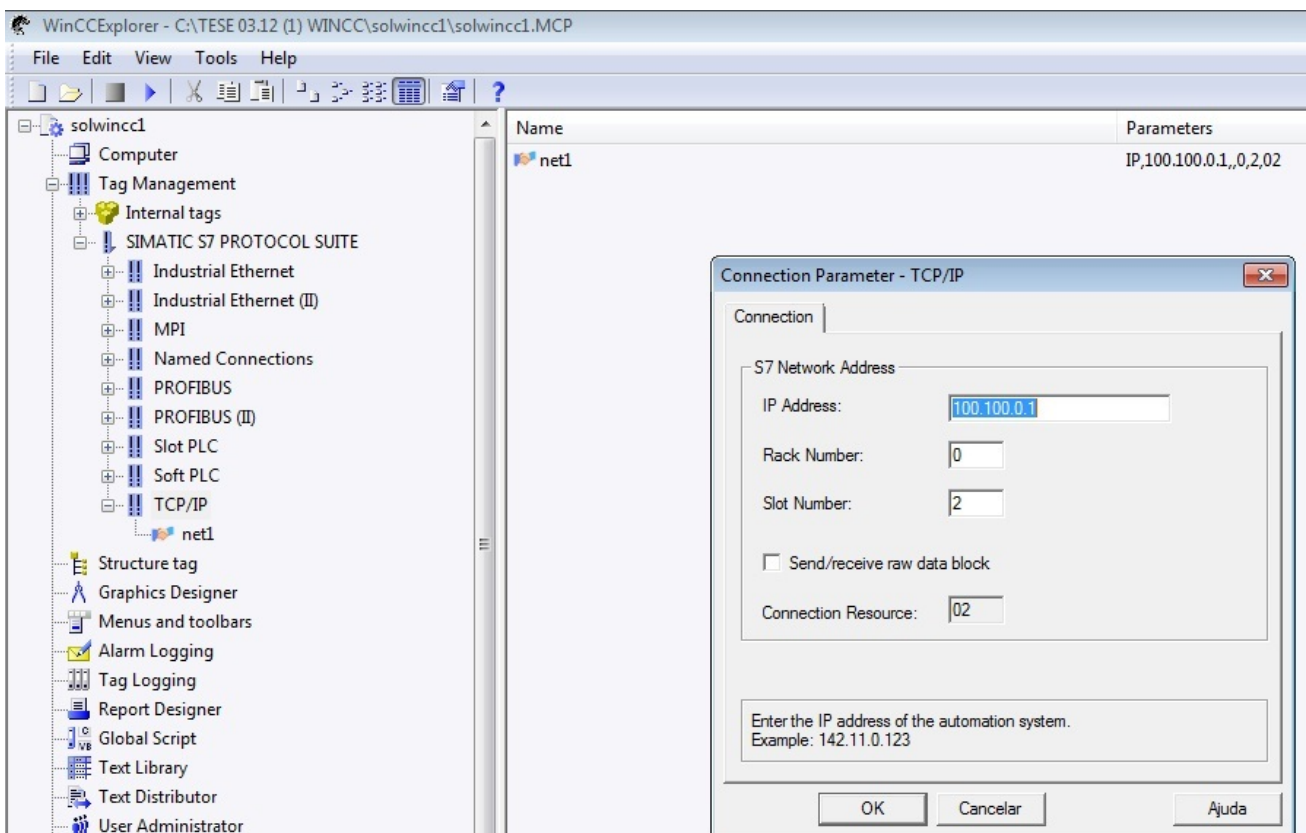


Fig. 63. Na árvore do projeto WinCC, no separador “Tag Management”, em “SIMATIC S7 PROTOCOL SUITE”, selecionar o separador TCP/IP e criar uma nova conexão para o driver escolhido (“New Driver Connection”)

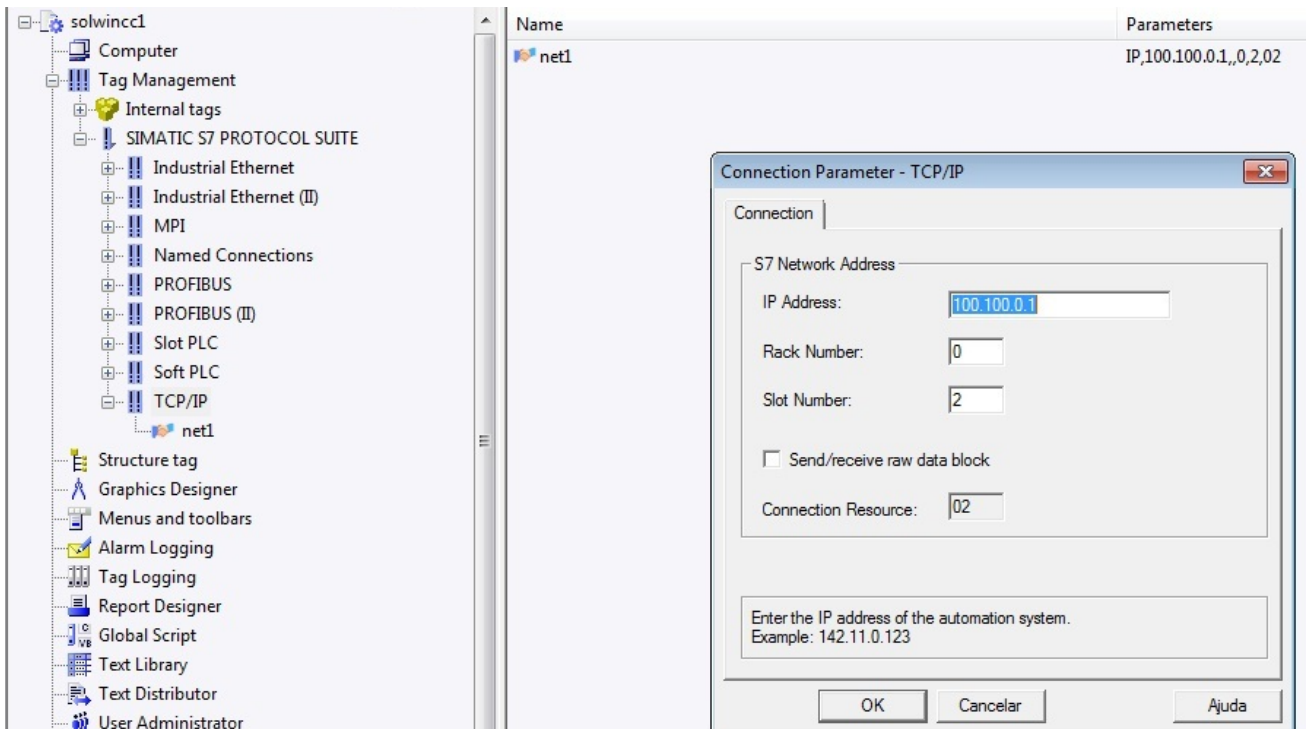


Fig. 64. Introduzir o IP atribuído à rede Ethernet TCP/IP (os IPs devem ser todos da mesma gama)

Name	Type	Parameters	Last Change
auto_go_wcc	Binary Tag	M250.0	7/25/2012 6:04:18 PM
auto_pulse_wcc	Binary Tag	M245.0	7/25/2012 6:04:40 PM
b1_ok_wcc	Binary Tag	M69.1	7/27/2012 3:37:08 PM
b2_ok_wcc	Binary Tag	M69.2	7/27/2012 3:37:23 PM
b3_ok_wcc	Binary Tag	M69.3	7/27/2012 3:37:46 PM
b4_ok_wcc	Binary Tag	M69.4	7/27/2012 3:38:06 PM
b5_ok_wcc	Binary Tag	M69.5	7/27/2012 3:38:22 PM
b7_ok_wcc	Binary Tag	M69.7	7/25/2012 6:09:55 PM
bi_mem_wcc	Unsigned 16-bit value	MW78	7/25/2012 6:05:38 PM
counter_v11_wcc	Signed 32-bit value	MD158	7/31/2012 6:32:45 PM
counter_v22_wcc	Signed 32-bit value	MD166	7/31/2012 6:33:25 PM
counter_value_m2_wcc	Signed 32-bit value	MD48	7/31/2012 6:35:02 PM
counter_value_wcc	Signed 32-bit value	MD44	7/31/2012 6:34:28 PM
go_zero_busca_wcc	Binary Tag	M250.1	7/25/2012 6:17:38 PM
mem_auto_wcc	Binary Tag	M248.1	7/25/2012 6:18:46 PM
mem_research_wcc	Binary Tag	M248.0	7/25/2012 6:19:18 PM
mf9_wcc	Binary Tag	M8.7	7/31/2012 5:08:48 PM
mw_volt_wcc	Unsigned 16-bit value	MW100	7/25/2012 6:20:28 PM
pot_wcc	Floating-point number 32-bit IEEE 754	MD35	7/25/2012 6:21:08 PM
potact3_wcc	Floating-point number 32-bit IEEE 754	MD65	7/25/2012 6:24:54 PM
research_mode_wcc	Binary Tag	M26.0	7/25/2012 6:22:10 PM
research_num_wcc	Unsigned 16-bit value	MW80	7/25/2012 6:23:22 PM
reset_m_wcc	Binary Tag	M7.1	7/25/2012 6:24:44 PM
sis2_mem_wcc	Binary Tag	M69.0	7/25/2012 6:25:07 PM
stop_m_wcc	Binary Tag	M7.0	7/25/2012 6:25:40 PM
t_hmi_wcc	Unsigned 16-bit value	MW128	8/1/2012 4:59:53 PM
t_in_wcc	Unsigned 16-bit value	MW120	7/31/2012 7:25:21 PM
time_auto_wcc	Binary Tag	M6.4	7/25/2012 6:26:25 PM
time_pulse_wcc	Binary Tag	M6.7	7/25/2012 6:30:02 PM
travel_m_m1_wcc	Binary Tag	M7.2	7/25/2012 6:30:58 PM
travel_m_m2_wcc	Binary Tag	M7.4	7/25/2012 6:31:25 PM
travel_p_m1_wcc	Binary Tag	M7.3	7/25/2012 6:31:51 PM
travel_p_m2_wcc	Binary Tag	M7.5	7/25/2012 6:32:08 PM
zero_pulse_wcc	Binary Tag	M245.1	7/25/2012 6:34:36 PM

Fig. 65. Lista completa das “tags” criadas para interação com o controlador

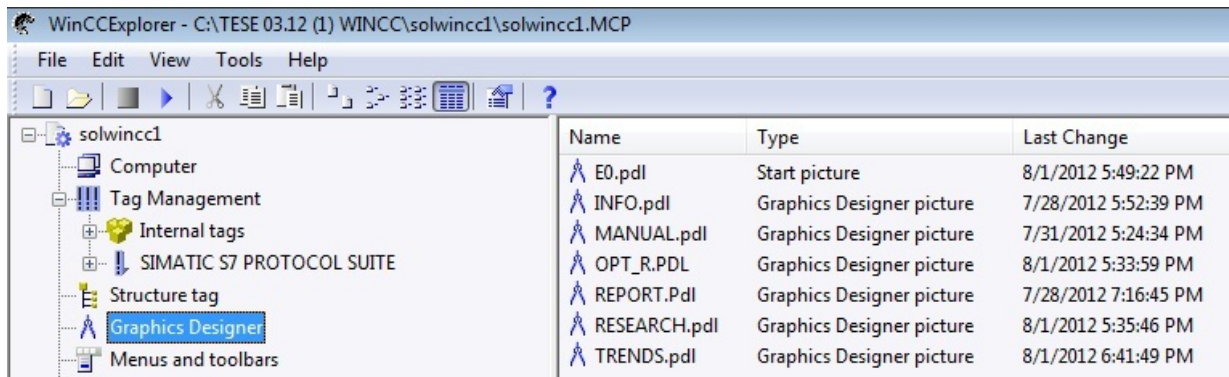


Fig. 66. Na árvore do projeto WinCC, no separador “Graphics Designer”, criar as telas da aplicação SCADA

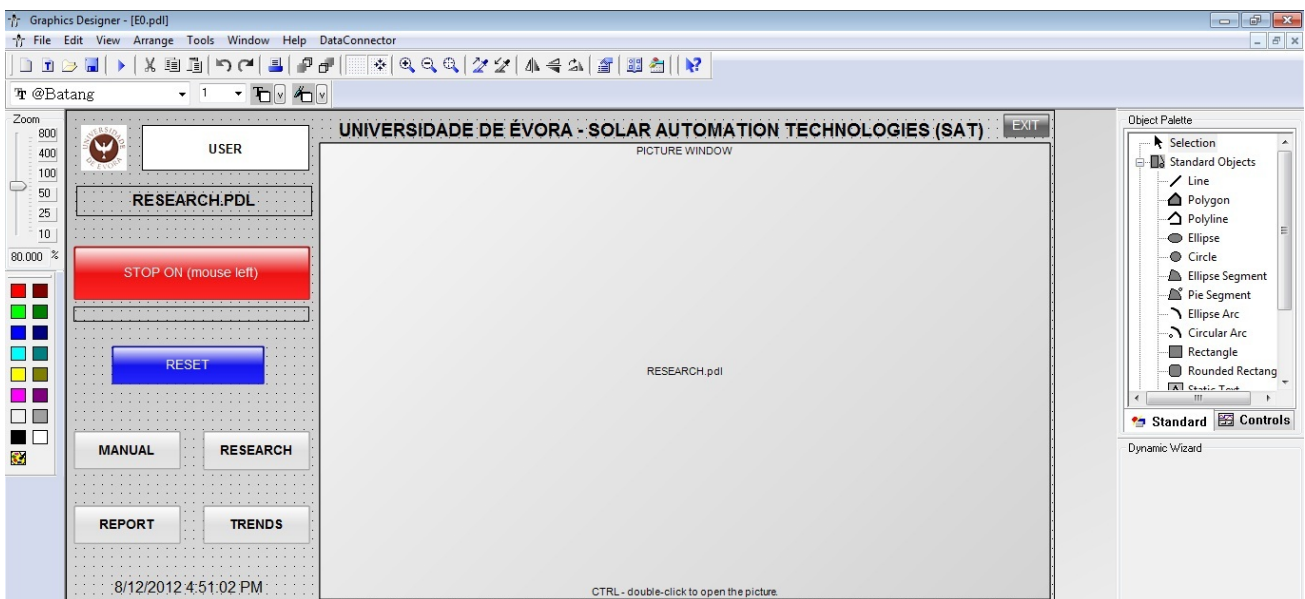


Fig. 67. Desenvolvimento da tela principal na aplicação “Graphics Designer”.

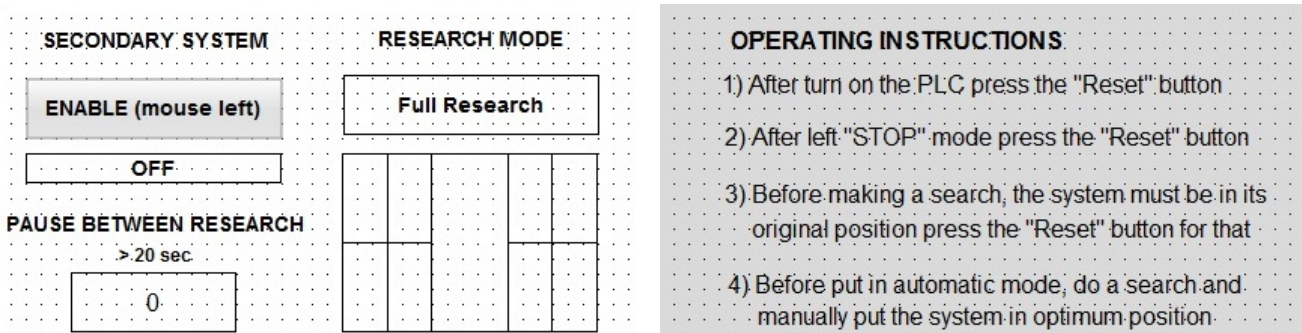


Fig. 68. Tela do sistema secundário (esquerda) e tela com instruções de operação (direita). Estas surgem num espaço reservado, na tela de varrimento (ver a fig. 70)

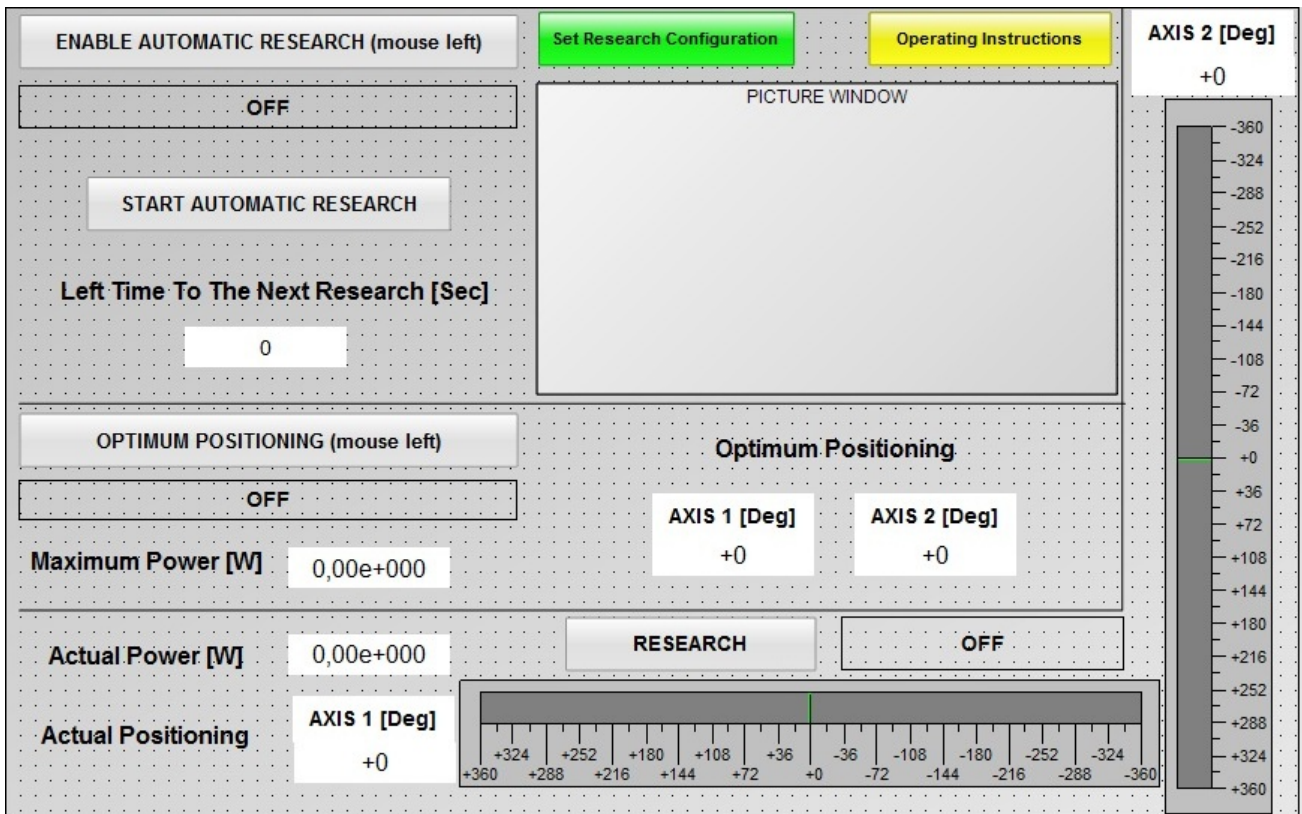


Fig. 69. Tela de operação de varrimentos

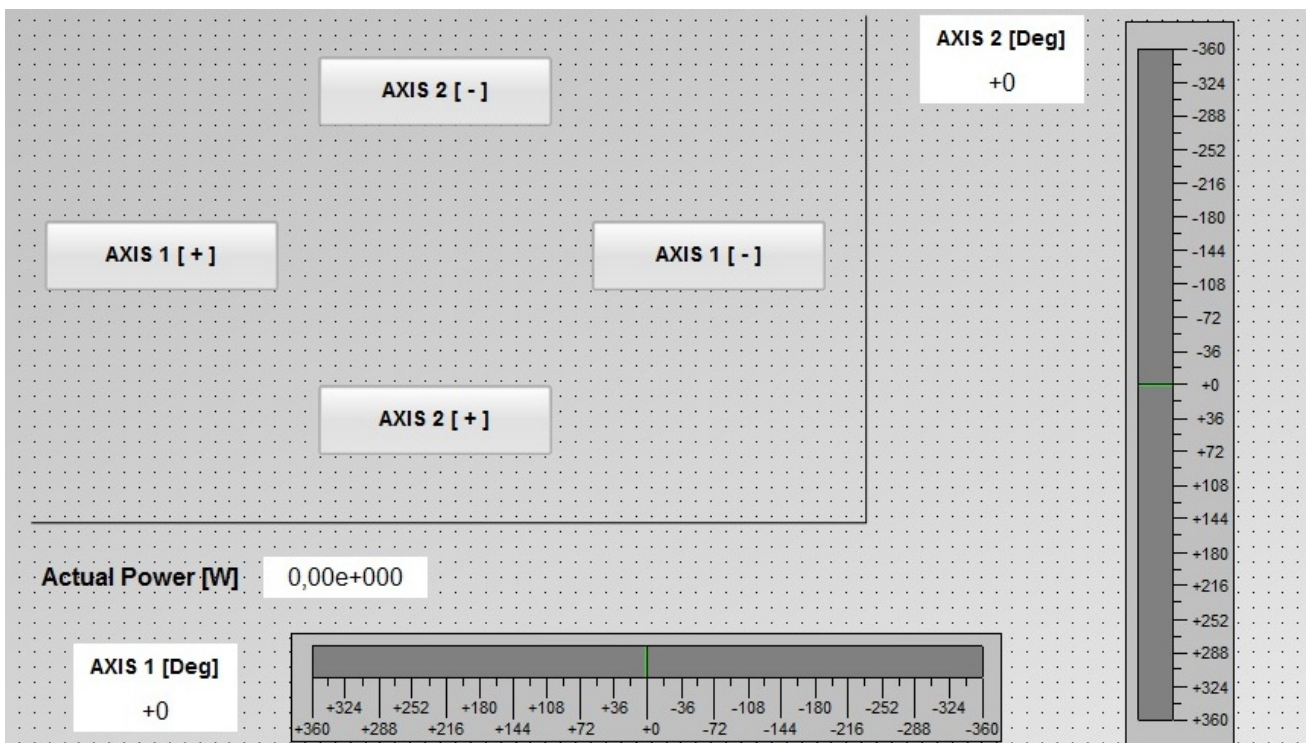


Fig. 70. Tela de posicionamento manual dos eixos

Configuração do objeto “WinCC Online Trend Control”:

- Esta configuração tem o objectivo de implementar um gráfico online para visualização do valor da potência da célula FV (atual e máximo registado);
- Para tal aceder à aplicação “Graphics Designer”, a partir do menu principal do WinCC;
- Em “Data Source” seleccionar a segunda opção: “Online tags” (1);
- Seleccionar as “tags” que se pretendem associar ao gráfico (2);
- Atribuir a cor para cada variável (3).

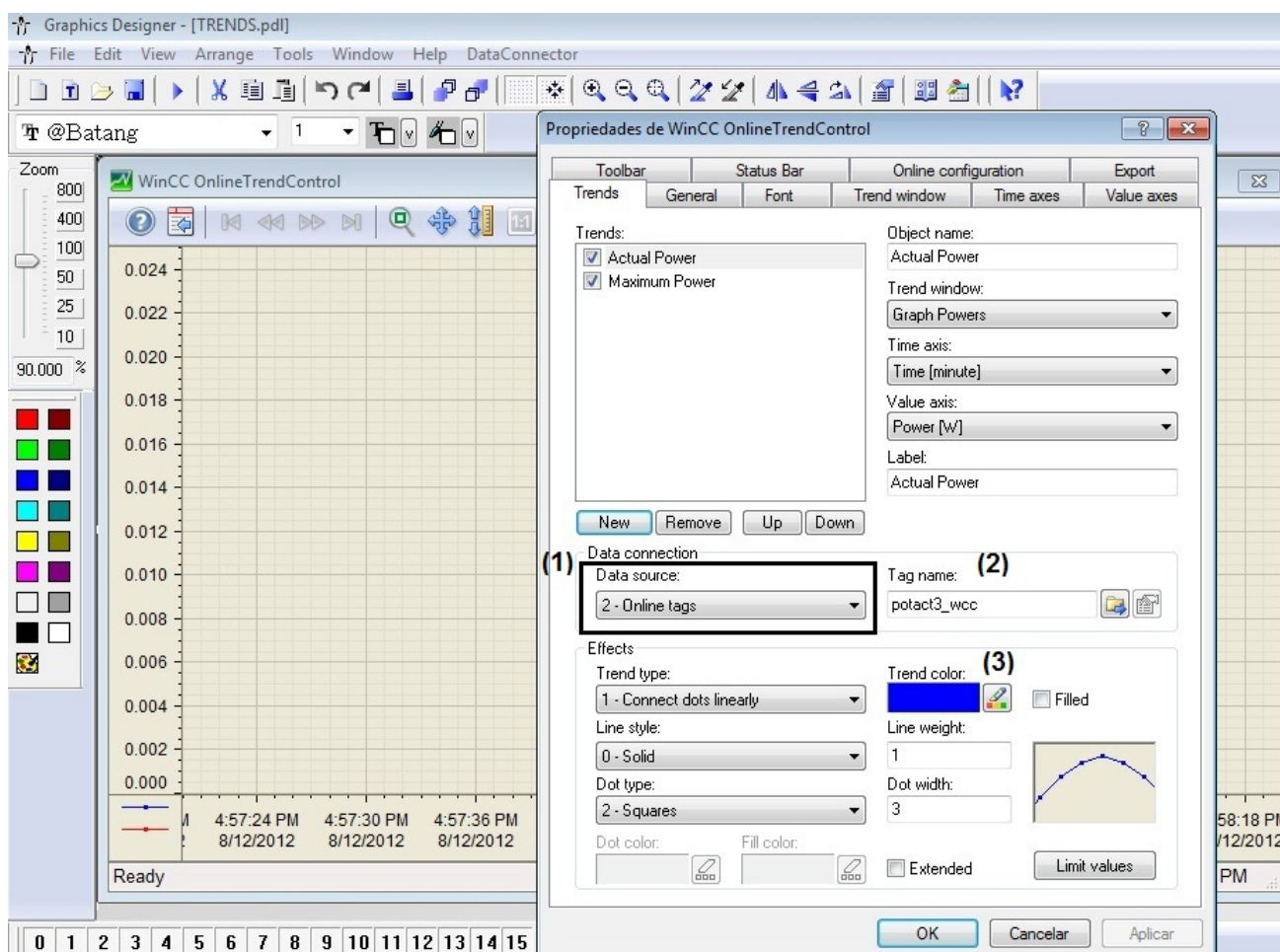


Fig. 71. Configuração do objeto “WinCC Online Trend Control”

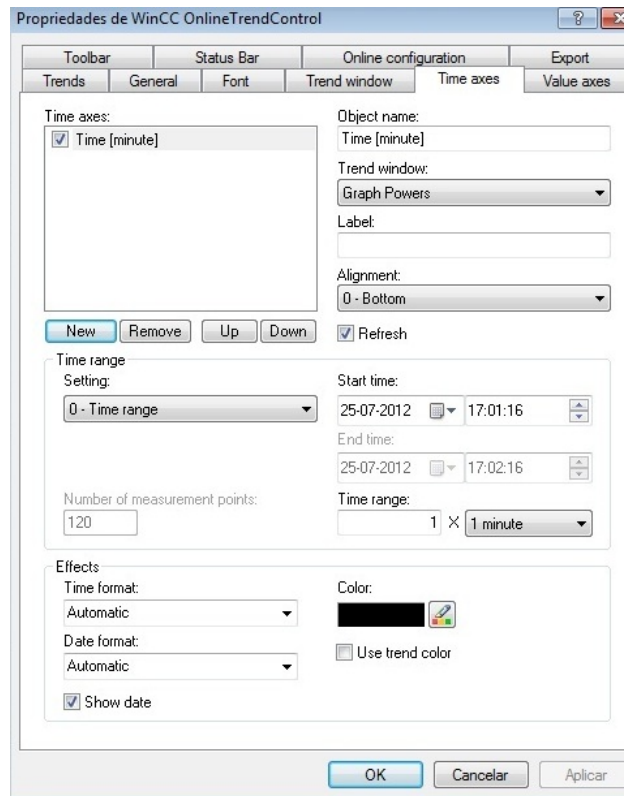
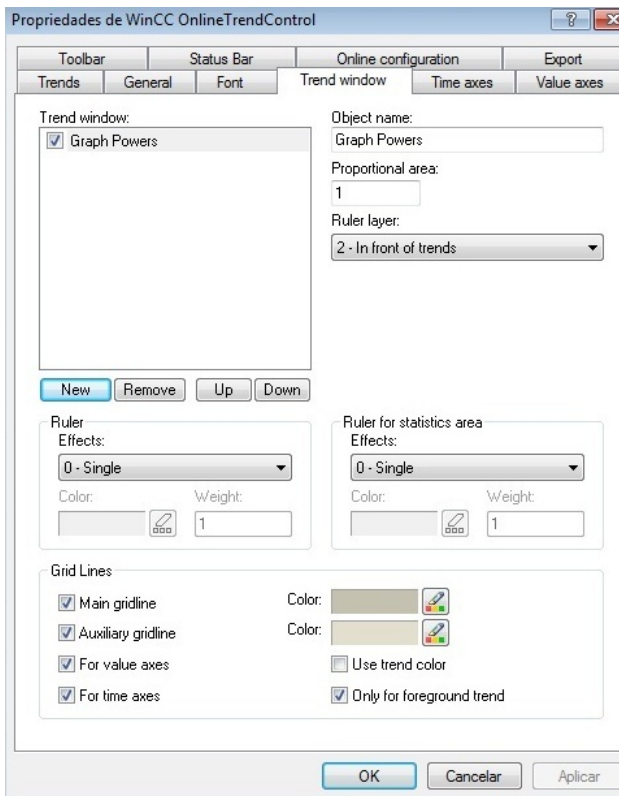


Fig. 72. Configuração da janela de gráficos (esquerda) e configuração do eixo temporal (direita)

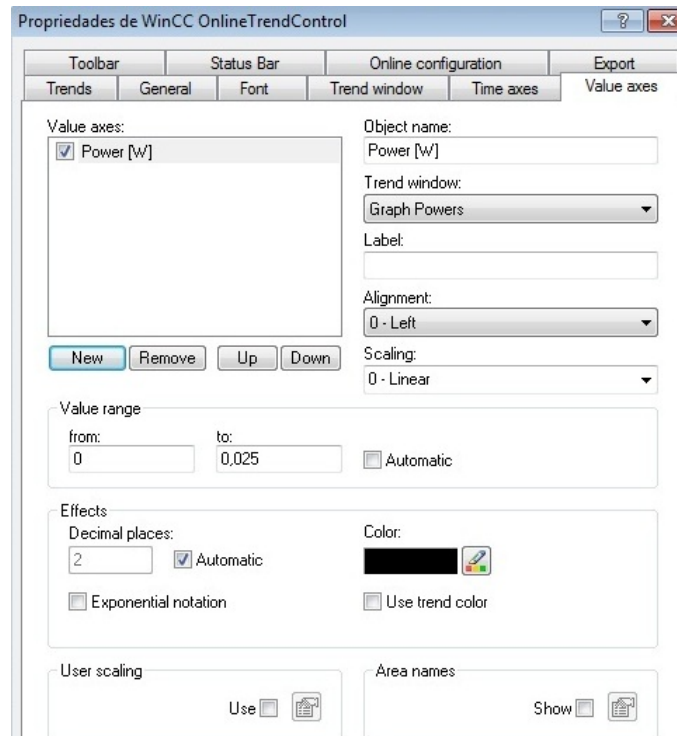


Fig. 73. Configuração do eixo potência (seleção do intervalo de leitura)

Configuração da aplicação de criação de relatórios

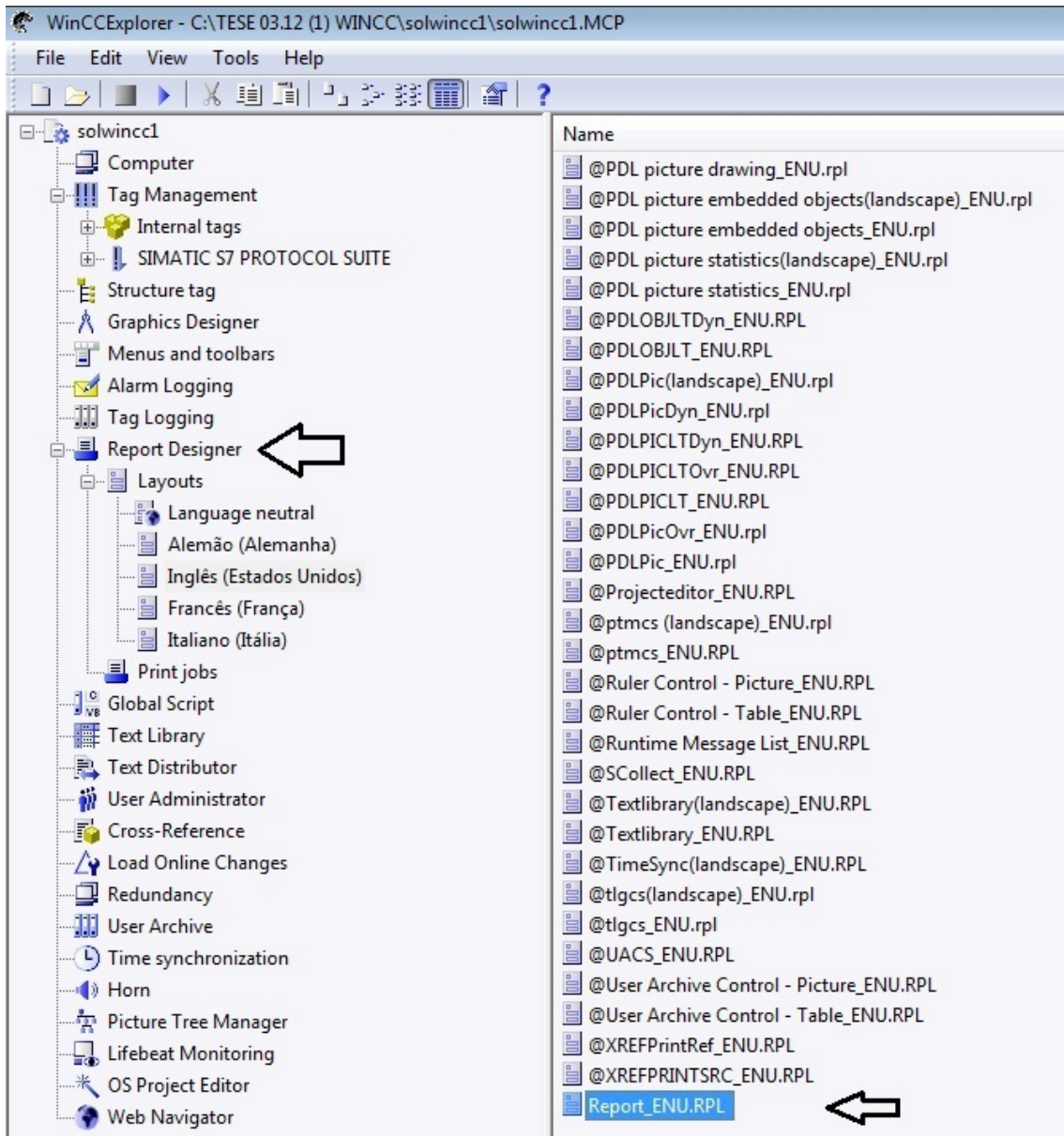


Fig. 74. Na árvore do projeto WinCC, no separador “*Report Designer*”, selecionar a opção de criação em língua inglesa e criar um novo relatório (“*Report_UNU*”), em “*Layouts*”

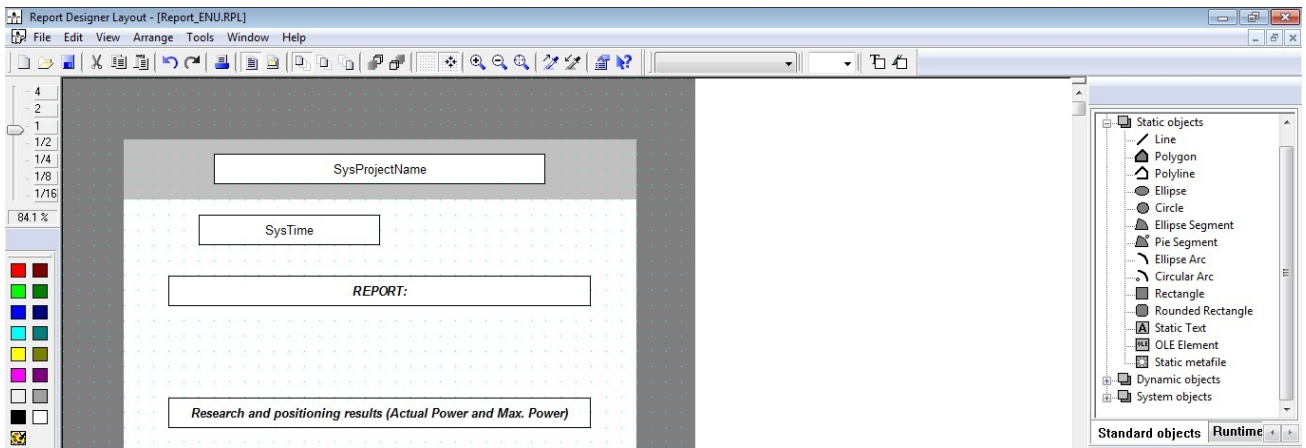


Fig. 75. Criação do relatório na aplicação “Report Designer Layout”

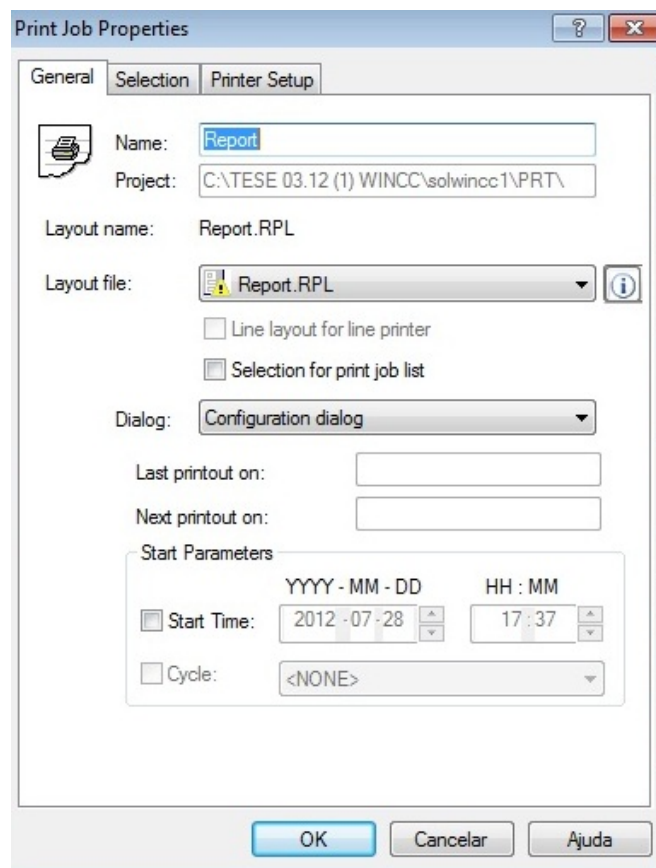


Fig. 76. Na árvore do projeto WinCC, no separador “Report Designer”, selecionar as propriedades da aplicação “Print jobs”, para escolher o layout de relatório criado anteriormente

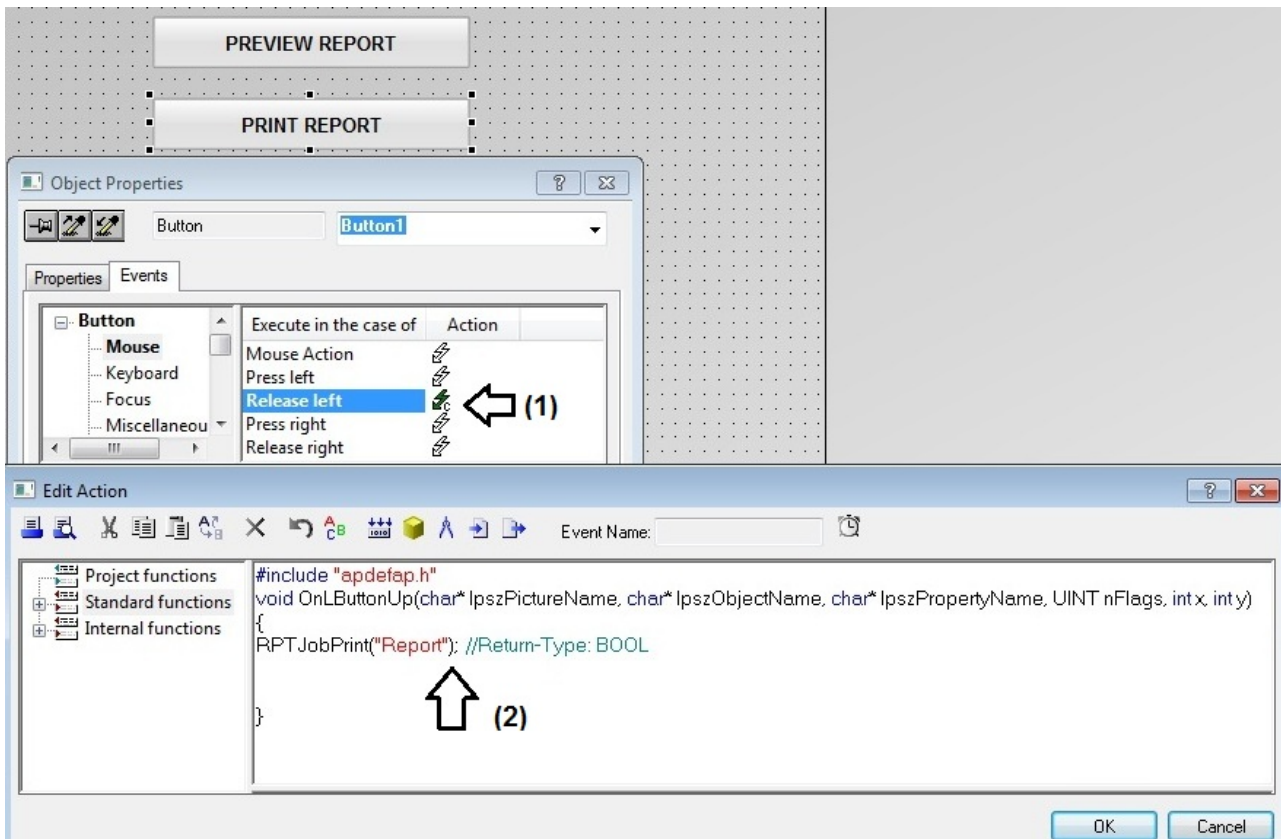


Fig. 78. Atribuição da funcionalidade de impressão, num botão da tela de relatório

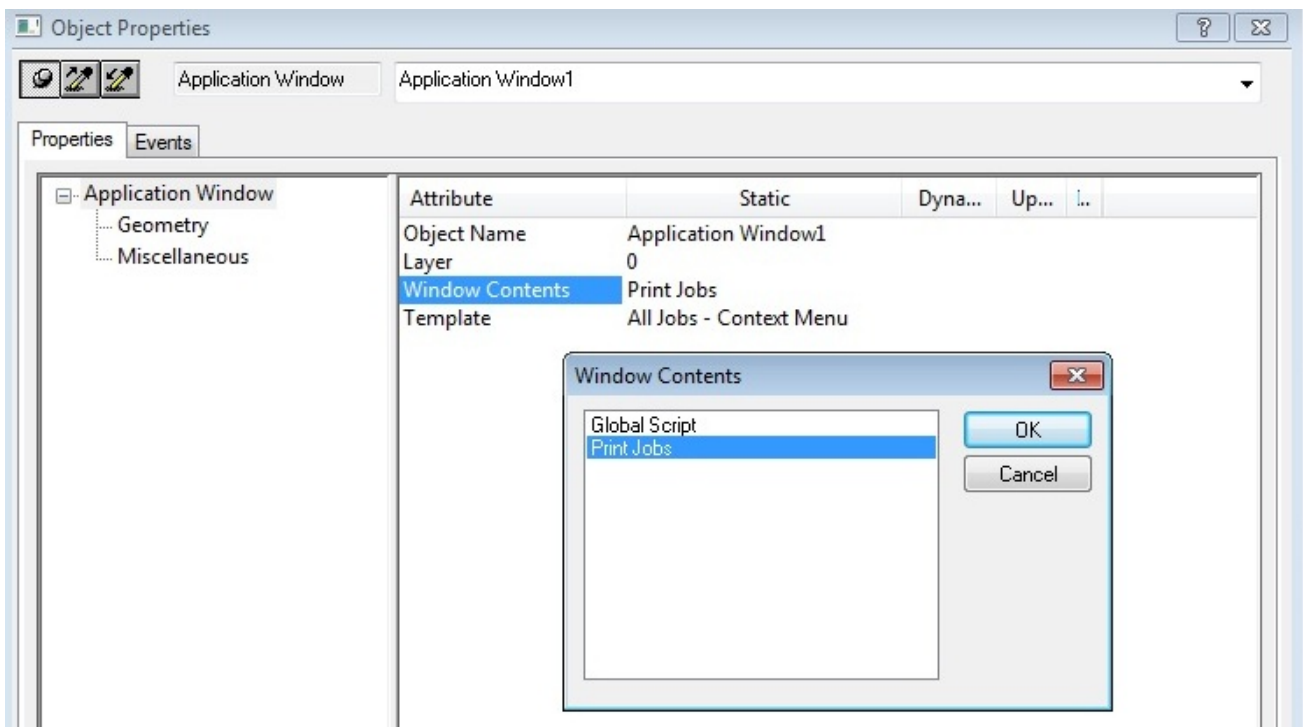


Fig. 79. Definição da funcionalidade do objeto "Application Window" colocado na tela de relatório

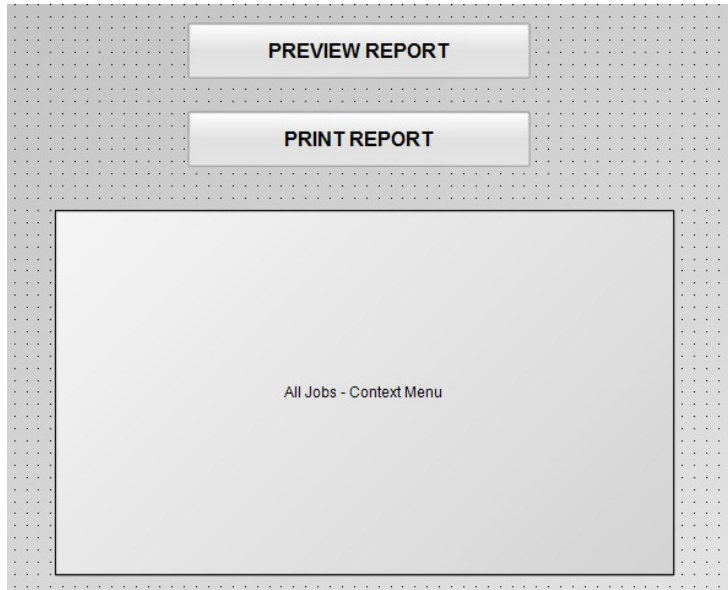


Fig. 80. Aspeto final da tela de relatório

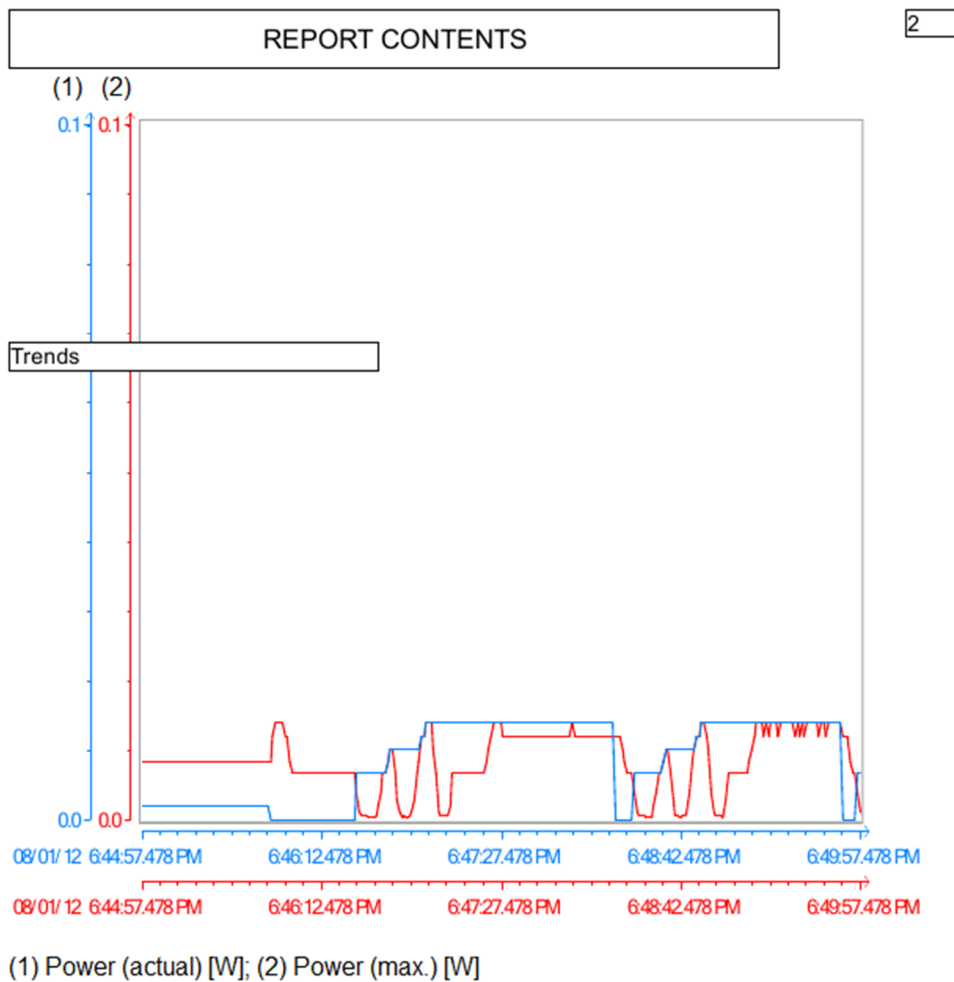


Fig.81. Conteúdo do relatório emitido através da aplicação SCADA (exemplo)

Criação do portal web através da aplicação “WinCC Navigator”

Com o auxílio da aplicação “Web Navigator” foi possível criar um portal Web para monitorizar a sistema local. Apenas é possível navegar entre telas, os valores das variáveis estão disponíveis mas a sua alteração está desabilitada. Em caso de necessidade é possível criar utilizadores e personalizar as condições de acesso de cada um.

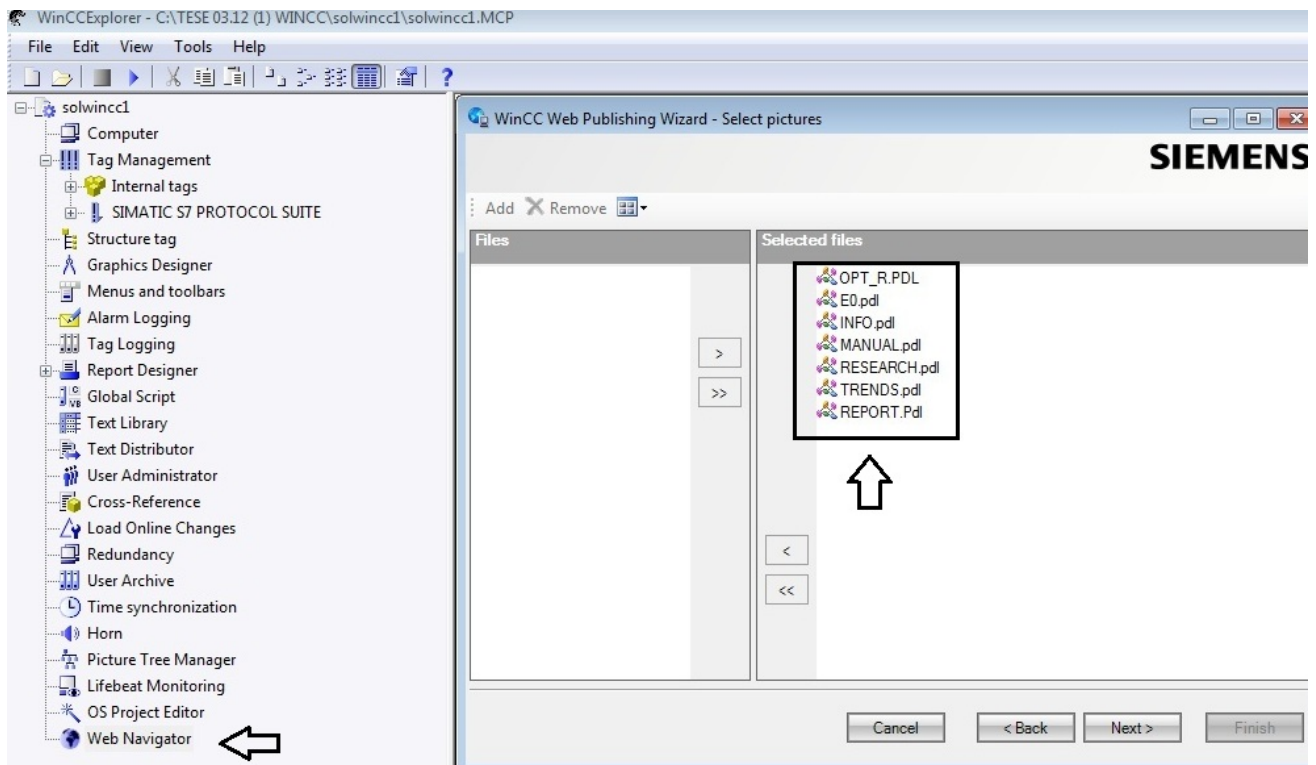


Fig. 82. Na árvore do projeto WinCC, no separador “Web Navigator”, correr a aplicação “WinCC Web Publishing Wizard” e seleccionar as telas a disponibilizar no portal Web

Name	Type	Last Change
E0.pd_		8/1/2012 7:26:58 PM
INFO.pd_		8/1/2012 7:26:58 PM
MANUAL.pd_		8/1/2012 7:26:58 PM
OPT_R.PD_		8/1/2012 7:26:58 PM
REPORT.Pd_		8/1/2012 7:27:01 PM
RESEARCH.pd_		8/1/2012 7:26:59 PM
TRENDS.pd_		8/1/2012 7:27:01 PM

Fig. 83. Telas do portal Web criadas a partir da aplicação “WinCC Web Publishing Wizard”

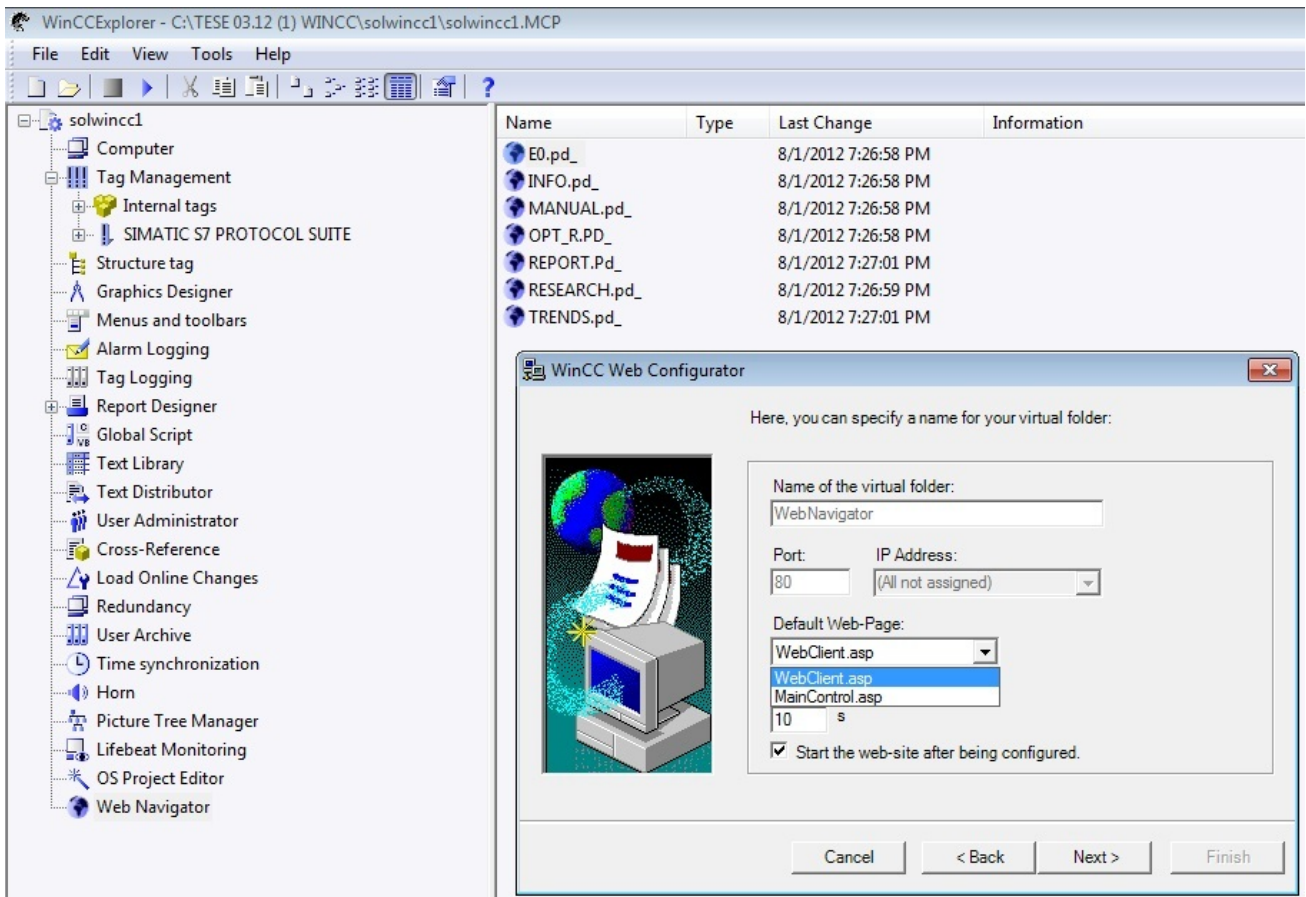


Fig. 84. Utilização da aplicação “WinCC Web Configurator” para criar o portal Web, para monitorização da aplicação SCADA

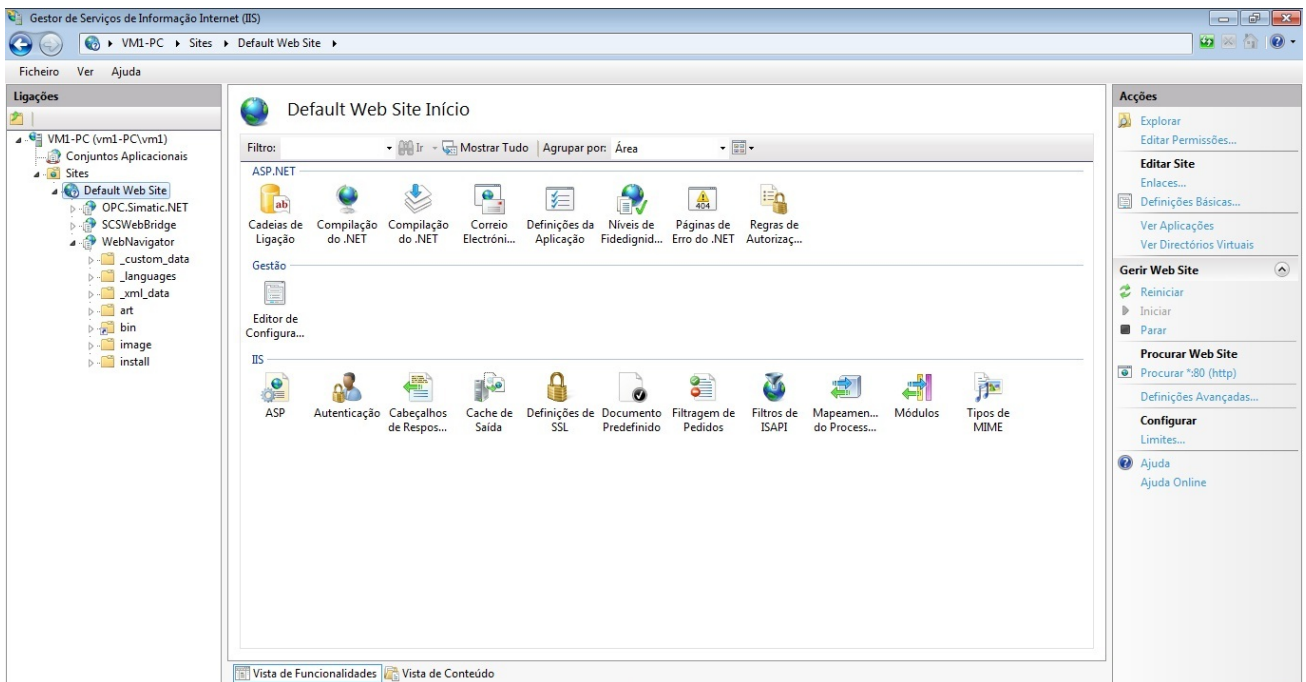


Fig. 85. Utilização do “Gestor de Serviços de Informação e Internet” (IIS) para gestão do “Default Web Site”

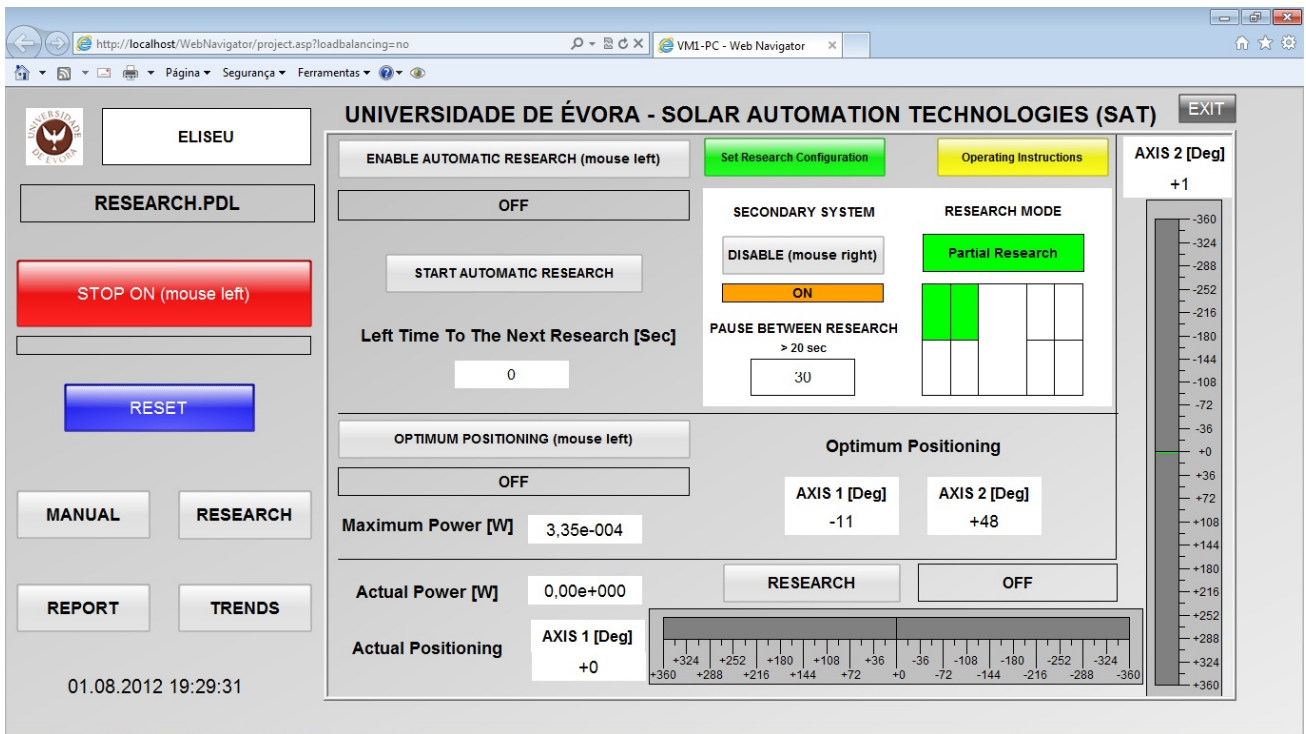
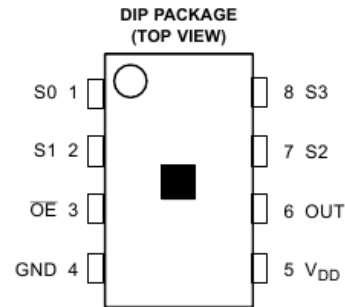


Fig. 86. Portal Web para monitorização do sistema, acessível por rede local (compatível com Internet Explorer)

Anexo K: Ficha técnica do circuito integrado TSL230R-LF

- High-Resolution Conversion of Light Intensity to Frequency With No External Components
- Programmable Sensitivity and Full-Scale Output Frequency
- Communicates Directly With a Microcontroller
- Single-Supply Operation Down to 2.7 V, With Power-Down Feature
- Absolute Output Frequency Tolerance of $\pm 5\%$ (TSL230BR-LF)
- Nonlinearity Error Typically 0.2% at 100 kHz
- Stable 150 ppm/°C Temperature Coefficient
- Replacements for TSL230, TSL230R, TSL230A, TSL230AR, TSL230B and TSL230BR
- RoHS Compliant

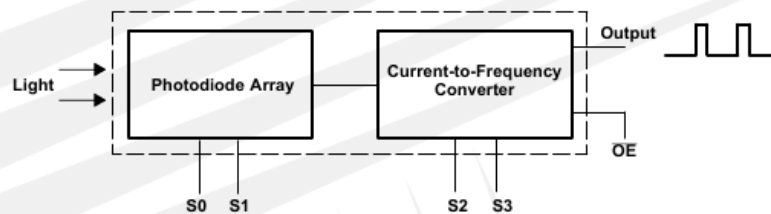


Description

The TSL230R-LF, TSL230AR-LF, and TSL230BR-LF programmable light-to-frequency converters combine a configurable silicon photodiode and a current-to-frequency converter on single monolithic CMOS integrated circuits. The output can be either a pulse train or a square wave (50% duty cycle) with frequency directly proportional to light intensity. Device sensitivity is selectable in three ranges, providing two decades of adjustment. The full-scale output frequency can be scaled by one of four preset values. All inputs and the output are TTL compatible, allowing direct two-way communication with a microcontroller for programming and output interface. An output enable (\overline{OE}) is provided that places the output in the high-impedance state for multiple-unit sharing of a microcontroller input line.

The devices are available with absolute-output-frequency tolerances of $\pm 5\%$ (TSL230BR-LF), $\pm 10\%$ (TSL230AR-LF), or $\pm 20\%$ (TSL230R-LF). They have been temperature compensated for the ultraviolet-to-visible light range of 320 nm to 700 nm and respond over the light range of 320 nm to 1050 nm. The devices are characterized over the temperature range of -25°C to 70°C .

Functional Block Diagram



TSL230R-LF, TSL230AR-LF, TSL230BR-LF PROGRAMMABLE LIGHT-TO-FREQUENCY CONVERTERS

TAOS079 – JANUARY 2006

Terminal Functions

TERMINAL NAME	NO.	TYPE	DESCRIPTION
GND	4		Ground
OE	3	I	Enable for f_O (active low)
OUT	6	O	Scaled-frequency (f_O) output
S0, S1	1, 2	I	Sensitivity-select inputs
S2, S3	7, 8	I	f_O scaling-select inputs
V _{DD}	5		Supply voltage

Selectable Options

S1	S0	SENSITIVITY
L	L	Power down
L	H	1×
H	L	10×
H	H	100×

S3	S2	f_O SCALING (divide-by)
L	L	1
L	H	2
H	L	10
H	H	100

Available Options

DEVICE	T _A	PACKAGE – LEADS	PACKAGE DESIGNATOR	ORDERING NUMBER
TSL230R-LF	-25°C to 70°C	PDIP-8	P	TSL230R-LFP
TSL230AR-LF	-25°C to 70°C	PDIP-8	P	TSL230AR-LFP
TSL230BR-LF	-25°C to 70°C	PDIP-8	P	TSL230BR-LFP

Absolute Maximum Ratings over operating free-air temperature range (unless otherwise noted)†

Supply voltage, V _{DD} (see Note 1)	6 V
Input voltage range, all inputs, V _I	-0.3 V to V _{DD} + 0.3 V
Operating free-air temperature range, T _A	-25°C to 70°C
Storage temperature range	-25°C to 85°C
Lead temperature 1,6 mm (1/16 inch) from case for 10 seconds‡	260°C

† Stresses beyond those listed under "absolute maximum ratings" may cause permanent damage to the device. These are stress ratings only, and functional operation of the device at these or any other conditions beyond those indicated under "recommended operating conditions" is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

‡ Not recommended for solder reflow.

NOTE 1: All voltage values are with respect to GND.

Recommended Operating Conditions

	MIN	NOM	MAX	UNIT
Supply voltage, V _{DD}	2.7	5	5.5	V
High-level input voltage, V _{IH}		V _{DD} = 4.5 V to 5.5 V	2	V
Low-level input voltage, V _{IL}		V _{DD} = 4.5 V to 5.5 V	0	V
Operating free-air temperature range, T _A	-25		70	°C

Copyright © 2006, TAOS Inc.



The LUMENOLOGY® Company

TSL230R-LF, TSL230AR-LF, TSL230BR-LF PROGRAMMABLE LIGHT-TO-FREQUENCY CONVERTERS

TAOS079 – JANUARY 2006

Electrical Characteristics at $T_A = 25^\circ\text{C}$, $V_{DD} = 5\text{ V}$ (unless otherwise noted)

PARAMETER		TEST CONDITIONS	MIN	TYP	MAX	UNIT
V_{OH}	High-level output voltage	$I_{OH} = -4\text{ mA}$	4	4.5		V
V_{OL}	Low-level output voltage	$I_{OL} = 4\text{ mA}$		0.25	0.4	V
I_{IH}	High-level input current				5	μA
I_{IL}	Low-level input current				5	μA
I_{DD}	Supply current	Power-on mode		2	3	mA
		Power-down mode		5	12	μA
F.S.	Full-scale frequency [†]		1.1			MHz
	Temperature coefficient of output frequency	$\lambda = 700\text{ nm}$		± 150		ppm/ $^\circ\text{C}$
k_{SVS}	Supply voltage sensitivity	$V_{DD} = 5\text{ V} \pm 10\%$		± 0.5		%/V

[†] Full-scale frequency is the maximum operating frequency of the device without saturation.

Operating Characteristics at $V_{DD} = 5\text{ V}$, $T_A = 25^\circ\text{C}$, $E_e = 130\text{ }\mu\text{W}/\text{cm}^2$, $\lambda_p = 640\text{ nm}$ (unless otherwise noted)

PARAMETER	TEST CONDITIONS	TSL230R-LF			TSL230AR-LF			TSL230BR-LF			UNIT	
		MIN	TYP	MAX	MIN	TYP	MAX	MIN	TYP	MAX		
f_O	Output frequency	$S0 = S1 = H,$ $S2 = S3 = L$	80	100	120	90	100	110	95	100	105	kHz
		$S1 = H,$ $S0 = S2 = S3 = L$	8	10	12	9	10	11	9.5	10	10.5	kHz
		$S0 = H,$ $S1 = S2 = S3 = L$	0.8	1	1.2	0.9	1	1.1	0.95	1	1.05	kHz
		$S0 = S1 = S2 = H,$ $S3 = L$	40	50	60	45	50	55	47.5	50	52.5	kHz
		$S0 = S1 = S3 = H,$ $S2 = L$	8	10	12	9	10	11	9.5	10	10.5	kHz
		$S0 = S1 = S2 = S3 = H$	0.8	1	1.2	0.9	1	1.1	0.95	1	1.05	kHz
		$E_e = 0,$ $S0 = S1 = H,$ $S2 = S3 = L$		0.4	10		0.4	10		0.4	10	Hz
R_e	Responsivity	$S0 = S1 = H,$ $S2 = S3 = L$		0.77			0.77		0.77		kHz/ ($\mu\text{W}/\text{cm}^2$)	
t_w	Output pulse duration	$S2 = S3 = L$	125		600	125		600	125		600	ns
		$S2$ or $S3 = H$		$1/2f_O$			$1/2f_O$			$1/2f_O$		s
Nonlinearity ^{‡#}		$f_O = 0\text{ MHz to }10\text{ kHz}$		$\pm 0.1\%$			$\pm 0.1\%$		$\pm 0.1\%$		%F.S.	
		$f_O = 0\text{ MHz to }100\text{ kHz}$		$\pm 0.2\%$			$\pm 0.2\%$		$\pm 0.2\%$		%F.S.	
		$f_O = 0\text{ MHz to }1\text{ MHz}$		$\pm 0.5\%$			$\pm 0.5\%$		$\pm 0.5\%$		%F.S.	
Recovery from power down				100			100			100	μs	
Step response to full-scale step input		1 pulse of new frequency plus 1 μs										
Response time to programming change		2 periods of new principal frequency plus 1 μs [§]										
Response time to output enable (OE)			50	150		50	150		50	150	ns	

[‡] Nonlinearity is defined as the deviation of f_O from a straight line between zero and full scale, expressed as a percent of full scale.

[#] Nonlinearity test condition: $S0 = S1 = H$, $S2 = S3 = L$.

[§] Principal frequency is the internal oscillator frequency, equivalent to divide-by-1 output selection.

The LUMENOLOGY[®] Company



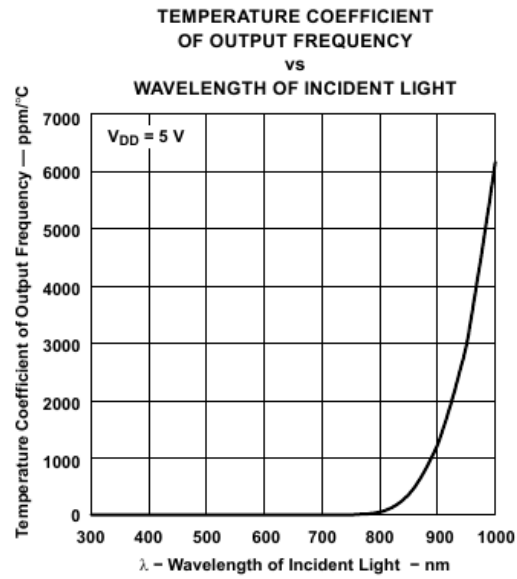
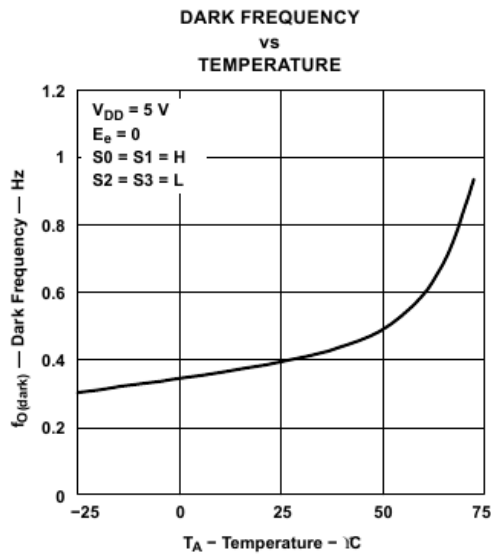
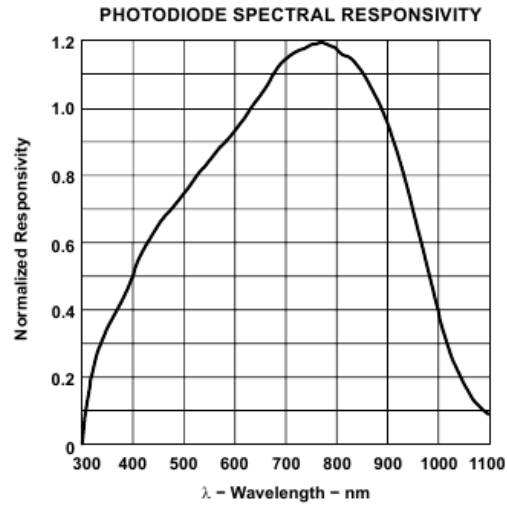
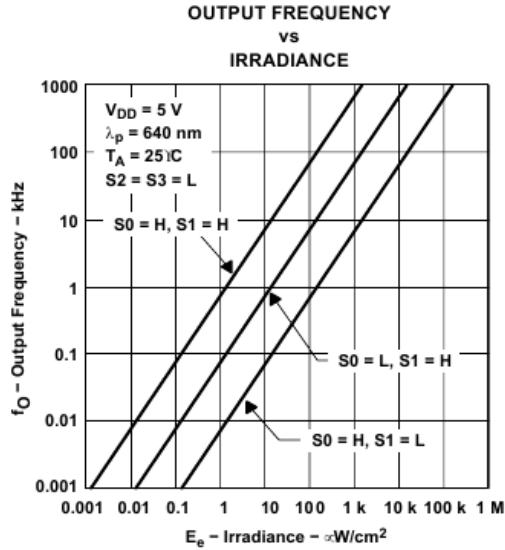
Copyright © 2006, TAOS Inc.

3

TSL230R-LF, TSL230AR-LF, TSL230BR-LF PROGRAMMABLE LIGHT-TO-FREQUENCY CONVERTERS

TAOS079 - JANUARY 2006

TYPICAL CHARACTERISTICS



Copyright © 2006, TAOS Inc.



The LUMENOLOGY® Company

TSL230R-LF, TSL230AR-LF, TSL230BR-LF PROGRAMMABLE LIGHT-TO-FREQUENCY CONVERTERS

TAOS079 – JANUARY 2006

TYPICAL CHARACTERISTICS

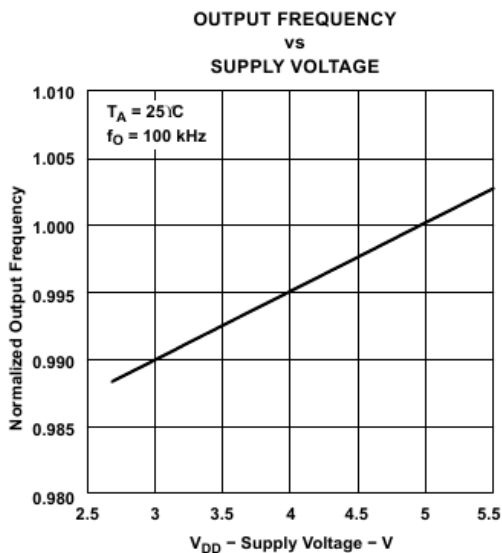


Figure 5

APPLICATION INFORMATION

Power-supply considerations

Power-supply lines must be decoupled by a 0.01- μ F to 0.1- μ F capacitor with short leads placed close to the device package. A low-noise power supply is required to minimize jitter on output pulses.

Input interface

A low-impedance electrical connection between the device \overline{OE} pin and the device GND pin is required for improved noise immunity.

Output interface

The output of the device is designed to drive a standard TTL or CMOS logic input over short distances. If lines greater than 12 inches are used on the output, a buffer or line driver is recommended.

Sensitivity adjustment

Sensitivity is controlled by two logic inputs, S0 and S1. Sensitivity is adjusted using an electronic iris technique — effectively an aperture control — to change the response of the device to a given amount of light. The sensitivity can be set to one of three levels: 1 \times , 10 \times or 100 \times , providing two decades of adjustment. This allows the responsivity of the device to be optimized to a given light level while preserving the full-scale output-frequency range. Changing of sensitivity also changes the effective photodiode area by the same factor.

The LUMENOLOGY[®] Company



Copyright © 2006, TAOS Inc.

5

TSL230R-LF, TSL230AR-LF, TSL230BR-LF PROGRAMMABLE LIGHT-TO-FREQUENCY CONVERTERS

TAOS079 – JANUARY 2006

APPLICATION INFORMATION

Output-frequency scaling

Output-frequency scaling is controlled by two logic inputs, S2 and S3. Scaling is accomplished on chip by internally connecting the pulse-train output of the converter to a series of frequency dividers. Divided outputs available are divide-by 2, 10, 100, and 1 (no division). Divided outputs are 50 percent-duty-cycle square waves while the direct output (divide-by 1) is a fixed-pulse-width pulse train. Because division of the output frequency is accomplished by counting pulses of the principal (divide-by 1) frequency, the final-output period represents an average of n (where n is 2, 10, or 100) periods of the principal frequency. The output-scaling-counter registers are cleared upon the next pulse of the principal frequency after any transition of the S0, S1, S2, S3, or OE lines. The output goes high upon the next subsequent pulse of the principal frequency, beginning a new valid period. This minimizes the time delay between a change on the input lines and the resulting new output period in the divided output modes. In contrast with the sensitivity adjust, use of the divided outputs lowers both the full-scale frequency and the dark frequency by the selected scale factor.

The frequency-scaling function allows the output range to be optimized for a variety of measurement techniques. The divide-by-1 or straight-through output can be used with a frequency counter, pulse accumulator, or high-speed timer (period measurement). The divided-down outputs may be used where only a slower frequency counter is available, such as a low-cost microcontroller, or where period measurement techniques are used. The divide-by-10 and divide-by-100 outputs provide lower frequency ranges for high resolution-period measurement.

Measuring the frequency

The choice of interface and measurement technique depends on the desired resolution and data acquisition rate. For maximum data-acquisition rate, period-measurement techniques are used.

Using the divide-by-2 output, data can be collected at a rate of twice the output frequency or one data point every microsecond for full-scale output. Period measurement requires the use of a fast reference clock with available resolution directly related to reference-clock rate. Output scaling can be used to increase the resolution for a given clock rate or to maximize resolution as the light input changes. Period measurement is used to measure rapidly varying light levels or to make a very fast measurement of a constant light source.

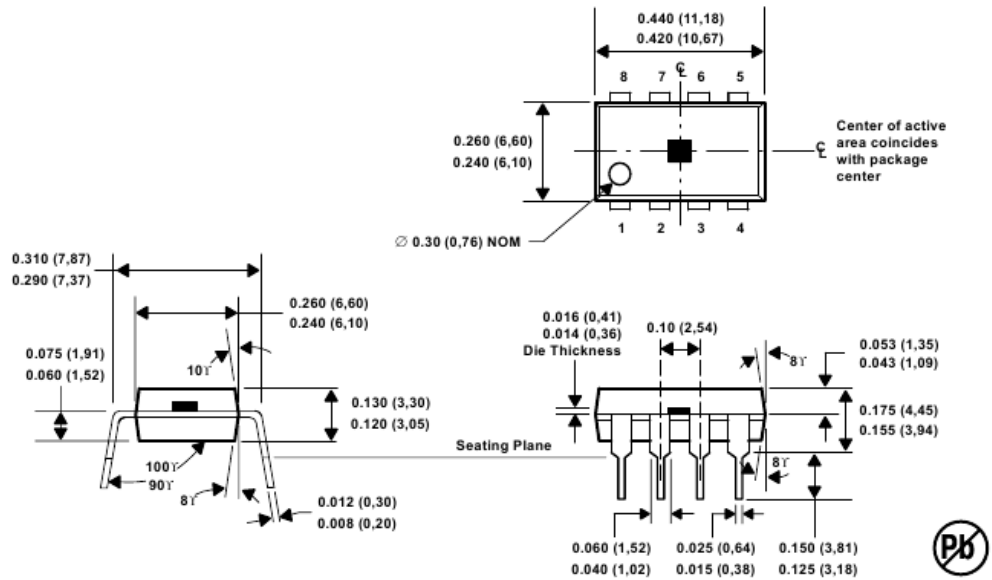
Maximum resolution and accuracy may be obtained using frequency-measurement, pulse-accumulation, or integration techniques. Frequency measurements provide the added benefit of averaging out random or high-frequency variations (jitter) resulting from noise in the light signal or from noise in the power supply. Resolution is limited mainly by available counter registers and allowable measurement time. Frequency measurement is well suited for slowly varying or constant light levels and for reading average light levels over short periods of time. Integration (the accumulation of pulses over a very long period of time) can be used to measure exposure, the amount of light present in an area over a given time period.

TSL230R-LF, TSL230AR-LF, TSL230BR-LF PROGRAMMABLE LIGHT-TO-FREQUENCY CONVERTERS

TAOS079 – JANUARY 2006

MECHANICAL INFORMATION

This dual-in-line package consists of an integrated circuit mounted on a lead frame and encapsulated with an electrically nonconductive clear plastic compound. The photodiode area is typically 0.92 mm^2 (0.0014 in^2) ($S0 = S1 = H$).



- NOTES: A. All linear dimensions are in inches and (millimeters).
 B. Index of refraction of clear plastic is 1.55.
 C. Lead finish is NiPd.
 D. This drawing is subject to change without notice.

Figure 6. Plastic Dual-In-Line Packaging Configuration

TSL230R-LF, TSL230AR-LF, TSL230BR-LF PROGRAMMABLE LIGHT-TO-FREQUENCY CONVERTERS

TAOS079 – JANUARY 2006

PRODUCTION DATA — information in this document is current at publication date. Products conform to specifications in accordance with the terms of Texas Advanced Optoelectronic Solutions, Inc. standard warranty. Production processing does not necessarily include testing of all parameters.

LEAD-FREE (Pb-FREE) and GREEN STATEMENT

Pb-Free (RoHS) TAOS' terms *Lead-Free* or *Pb-Free* mean semiconductor products that are compatible with the current RoHS requirements for all 6 substances, including the requirement that lead not exceed 0.1% by weight in homogeneous materials. Where designed to be soldered at high temperatures, TAOS Pb-Free products are suitable for use in specified lead-free processes.

Green (RoHS & no Sb/Br) TAOS defines *Green* to mean Pb-Free (RoHS compatible), and free of Bromine (Br) and Antimony (Sb) based flame retardants (Br or Sb do not exceed 0.1% by weight in homogeneous material).

Important Information and Disclaimer The information provided in this statement represents TAOS' knowledge and belief as of the date that it is provided. TAOS bases its knowledge and belief on information provided by third parties, and makes no representation or warranty as to the accuracy of such information. Efforts are underway to better integrate information from third parties. TAOS has taken and continues to take reasonable steps to provide representative and accurate information but may not have conducted destructive testing or chemical analysis on incoming materials and chemicals. TAOS and TAOS suppliers consider certain information to be proprietary, and thus CAS numbers and other limited information may not be available for release.

NOTICE

Texas Advanced Optoelectronic Solutions, Inc. (TAOS) reserves the right to make changes to the products contained in this document to improve performance or for any other purpose, or to discontinue them without notice. Customers are advised to contact TAOS to obtain the latest product information before placing orders or designing TAOS products into systems.

TAOS assumes no responsibility for the use of any products or circuits described in this document or customer product design, conveys no license, either expressed or implied, under any patent or other right, and makes no representation that the circuits are free of patent infringement. TAOS further makes no claim as to the suitability of its products for any particular purpose, nor does TAOS assume any liability arising out of the use of any product or circuit, and specifically disclaims any and all liability, including without limitation consequential or incidental damages.

TEXAS ADVANCED OPTOELECTRONIC SOLUTIONS, INC. PRODUCTS ARE NOT DESIGNED OR INTENDED FOR USE IN CRITICAL APPLICATIONS IN WHICH THE FAILURE OR MALFUNCTION OF THE TAOS PRODUCT MAY RESULT IN PERSONAL INJURY OR DEATH. USE OF TAOS PRODUCTS IN LIFE SUPPORT SYSTEMS IS EXPRESSLY UNAUTHORIZED AND ANY SUCH USE BY A CUSTOMER IS COMPLETELY AT THE CUSTOMER'S RISK.

LUMENOLOGY, TAOS, the TAOS logo, and Texas Advanced Optoelectronic Solutions are registered trademarks of Texas Advanced Optoelectronic Solutions Incorporated.

Anexo L: Tabela de sinais de controlo e de verificação de FC1

German	English	AW-DB	Description																
Control signals																			
TFB	TEST_EN	DBX14.1	Switch P bus interface to "Start-up"																
BFQ/FSQ	OT_ERR_A	DBX14.3	Acknowledge operator errors and traversing errors																
ST	START	DBX15.0	Start																
STP	STOP	DBX15.1	Stop																
R-	DIR_M	DBX15.2	Negative direction																
R+	DIR_P	DBX15.3	Positive direction																
QMF	ACK_MF	DBX15.4	Acknowledge M function																
EFG	READ_EN	DBX15.5	Read Enable																
SA	SKIP_BLK	DBX15.6	Skip block																
AF	DRV_EN	DBX15.7	Drive enable																
BA	MODE_IN	DBB16	<table border="0"> <tr> <td>Operating mode</td> <td>Code</td> </tr> <tr> <td>Jog</td> <td>01</td> </tr> <tr> <td>Control</td> <td>02</td> </tr> <tr> <td>Approach to reference point</td> <td>03</td> </tr> <tr> <td>Incremental mode, relative</td> <td>04</td> </tr> <tr> <td>MDI</td> <td>06</td> </tr> <tr> <td>Automatic</td> <td>08</td> </tr> <tr> <td>Automatic single block</td> <td>09</td> </tr> </table>	Operating mode	Code	Jog	01	Control	02	Approach to reference point	03	Incremental mode, relative	04	MDI	06	Automatic	08	Automatic single block	09
Operating mode	Code																		
Jog	01																		
Control	02																		
Approach to reference point	03																		
Incremental mode, relative	04																		
MDI	06																		
Automatic	08																		
Automatic single block	09																		
BP	MODE_TYPE	DBB17	<table border="0"> <tr> <td>Operating mode parameters</td> <td>Code</td> </tr> <tr> <td>Speed levels</td> <td>1 and 2</td> </tr> <tr> <td>Frequency levels</td> <td>1 and 2</td> </tr> <tr> <td>Incremental dimension selection</td> <td>1...100, 254</td> </tr> </table>	Operating mode parameters	Code	Speed levels	1 and 2	Frequency levels	1 and 2	Incremental dimension selection	1...100, 254								
Operating mode parameters	Code																		
Speed levels	1 and 2																		
Frequency levels	1 and 2																		
Incremental dimension selection	1...100, 254																		
OVERR	OVERRIDE	DBB18	Override																
Checkback signals																			
TFGS	TST_STAT	DBX22.1	Switching of P BUS interface completed																
BF/FS	OT_ERR	DBX22.3	Operator-/traversing error																
DF	DATA_ERR	DBX22.4	Data error																
PARA	PARA	DBX22.7	Channel initialized																
SFG	ST_ENBLD	DBX23.0	Start Enable																
BL	WORKING	DBX23.1	Process in progress																
WFG	WAIT_EI	DBX23.2	Wait for external Enable																
T-L	DT_RUN	DBX23.5	Dwell time running																
PBR	PR_BACK	DBX23.6	Reverse program scanning																
BAR	MODE_OUT	DBB24	Active operating mode																
SYN	SYNC	DBX25.0	Channel synchronized																
ME	MSR_DONE	DBX25.1	End of measurement																
FR-	GO_M	DBX25.2	Travel in negative direction																
FR+	GO_P	DBX25.3	Travel in positive direction																
SRFG	ST_SERVO	DBX25.4	Servo enable status																
FIWS	FVAL_DONE	DBX25.5	Setting of on-the-fly actual value successfully completed																
PEH	POS_RCD	DBX25.7	Position reached. Stop.																
MNR	NUM_MF	DBB26	M function number																
AMF	STR_MF	DBX27.4	M function modification																

Anexo M: Programação

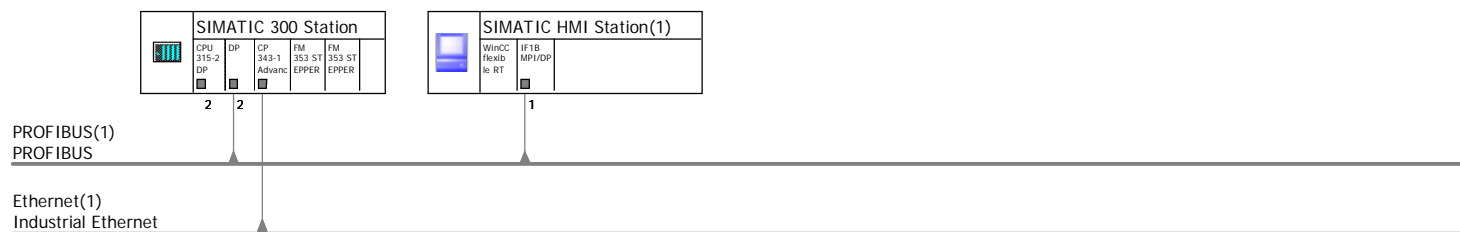
- **Tópico 1: Configuração de redes – NetPro**
Pág. 1
- **Tópico 2: Lista de símbolos**
Pág. 4
- **Tópico 3: Lista de blocos**
Pág. 8
- **Tópico 4: Programação – STEP 7**
Pág. 11
- **Tópico 5: Programação – Arduino**
Pág. 156

Tópico 4 – Índice de blocos (Programação STEP 7)

Bloco	Pág. (Anexo M)
DB1	103
DB2	113
DB3	124
DB4	126
DB21	129
DB100	129
DB200	130
DB300	131
DB400	132
FB21	93
FB300	94
FB400	98
FC0	18
FC1	20
FC2	32
FC3	33
FC4	35
FC5	37
FC6	39
FC7	46
FC8	48
FC9	50
FC10	57
FC11	57
FC12	58
FC13	59
FC14	60
FC15	51
FC16	62
FC100	63
FC101	65
FC102	67
FC103	69
FC200	71

Bloco	Pág. (Anexo M)
FC201	73
FC202	75
FC203	77
FC301	79
FC302	81
FC401	85
FC402	87
FC500	91
FC600	92
FC700	92
OB1	12
OB40	16
OB82	17
OB100	14
SFC6	153
SFC51	154
SFC58	155
SFC59	155
UDT1	133
UDT2	141
UDT3	149
UDT4	151

Tópico 1 – Configuração de redes – NetPro



1-1

Page 1 of 3

Industrial Ethernet: Ethernet(1) (S7 subnet ID: 0048 - 001F) contains the following network connections:

Station address:	Station:	Module:	R/S:
100.100.0.1	SIMATIC 300 Station	CP 343-1 Advanced	0/4

PROFIBUS: PROFIBUS(1) (S7 subnet ID: 001A - 0019) contains the following network connections:

Station address:	Station:	Module:	R/S:
1	SIMATIC HMI Station(1)	IF1B MPI/DP	-
2	SIMATIC 300 Station	CPU 315-2 DP	0/2

SAT (C:\Program Files\Siemens\Step7\s7proj\SVSA_adv)

The project contains no cross-project subnets

Tópico 2: Lista de símbolos

Properties of symbol table

Name:	Symbols
Author:	EACF
Comment:	
Created on:	07/18/2012 07:32:51 PM
Last modified on:	08/13/2012 03:56:55 PM
Last filter criterion:	All Symbols
Number of symbols:	166/166
Last Sorting:	Symbol Ascending

Status	Symbol	Address	Data type	Comment
	arduino	PIW 292	WORD	Entrada analógica: sinal do sistema secundário
	AUTO	FC 6	FC 6	Transição entre o modo de varrimento e o o modo de posicionamento (com F5)
	auto go	M 250.0	BOOL	
	auto pulse	M 245.0	BOOL	
	b1_mem	MW 78	INT	
	b1_ok	M 69.1	BOOL	
	b2_ok	M 69.2	BOOL	
	b3_ok	M 69.3	BOOL	
	b4_ok	M 69.4	BOOL	
	b5_ok	M 69.5	BOOL	
	b7_ok	M 69.7	BOOL	
	BEGIN_DATA	FC 500	FC 500	Após fazer reset introduz valor standard de velocidade M1 e M2
	Call_FB21	FC 16	FC 16	Ajuda a chamar mais facilmente, no OB1, a FB21
	CLEAR	FC 5	FC 5	Limpa a potência máxima e respectivos ângulos. Reset aos DBX's do modo MDI
	CNT_CTL1_ENC1	FC 302	FC 302	Control counter module FM 350-1 Encoder 1
	CNT_CTL1_ENC2	FC 402	FC 402	Control counter module FM 350-1 Encoder 2
	cont_aux_fc103	C 30	COUNTER	
	cont_aux_fc12	C 100	COUNTER	
	cont_aux_fc203	C 25	COUNTER	
	cont1auto1	C 15	COUNTER	
	cont1auto2	C 16	COUNTER	
	CONV_T	FB 21	FB 21	Converte INT para S5TIME
	CONV_T_DB	DB 21	FB 21	
	COUNTER_REV_ENC1	FC 600	FC 600	New counter value Encoder 1
	COUNTER_REV_ENC2	FC 700	FC 700	New counter value Encoder 2
	Counter_v11	MD 158	DINT	
	Counter_v111	MD 162	DINT	
	Counter_v22	MD 166	DINT	

Page 1 of 5

Status	Symbol	Address	Data type	Comment
	Counter_v222	MD 170	DINT	
	Counter_value	MD 44	DINT	
	Counter_value_M2	MD 48	DINT	
	Data_Enter	M 23.0	BOOL	
	DB_PROG_ENC1	DB 300	FB 300	DB300 (FB300 DB instance)
	DB_PROG_ENC2	DB 400	FB 400	DB400 (FB400 DB instance)
	DB_UDT_ENC1	DB 3	UDT 3	
	DB_UDT_ENC2	DB 4	UDT 4	
	DB_UDT_M1	DB 1	UDT 1	
	DB_UDT_M2	DB 2	UDT 2	
	DBEX_100_M1	DB 100	DB 100	Interface (FC100, DB1, UDT1)
	DBEX_200_M2	DB 200	DB 200	Interface (FC200, DB2, UDT2)
	DIAG_INF	FC 301	FC 301	Read diagnostic data Encoder 1
	DIAG_INF_M2	FC 401	FC 401	Read diagnostic data Encoder 2
	DIAGNOSTICS	OB 82	OB 82	Diagnostics FM353 e FM350-1
	go zero busca	M 250.1	BOOL	
	HMI_WCC	FC 15	FC 15	Auxiliar na criação de botões (HMI - WinCC)
	m byte db	MB 90	BYTE	
	m_auto_mp_m1	M 27.0	BOOL	
	m_auto_mp_m2	M 27.2	BOOL	
	m_auto_pi_m1	M 27.1	BOOL	
	m_aux0_fc6	M 30.0	BOOL	
	m_aux1_fc6	M 30.1	BOOL	
	m_aux2_fc6	M 30.2	BOOL	
	m_aux3_fc6_time_pulse	M 30.3	BOOL	
	m_aux4_fc6	M 30.4	BOOL	
	m_aux5_fc6	M 30.5	BOOL	
	m_byte_db_m2	MB 91	BYTE	
	m_uato_pi_m2	M 27.3	BOOL	
	MCU_ENC1	FC 10	FC 10	FM350-1 Encoder 1 (module adress, channel adress, user data length)
	MCU_ENC2	FC 11	FC 11	FM350-1 Encoder 2 (module adress, channel adress, user data length)
	mem_auto	M 248.1	BOOL	
	mem_auto_off	M 245.2	BOOL	
	mem_aux_fc102	M 6.2	BOOL	
	mem_aux_fc202	M 6.3	BOOL	
	mem_aux1_fc103	M 252.0	BOOL	
	mem_aux1_fc12	M 249.0	BOOL	

(Anexo M) 5

Page 2 of 5

Status	Symbol	Address	Data type	Comment
	mem_aux1_fc203	M 251.0	BOOL	
	mem_aux2_fc103	M 252.1	BOOL	
	mem_aux2_fc12	M 249.1	BOOL	
	mem_aux2_fc203	M 251.1	BOOL	
	mem_research	M 248.0	BOOL	
	mem_research_off	M 255.1	BOOL	
	mem1auto1	M 247.2	BOOL	
	mem1auto2	M 246.2	BOOL	
	mem2auto1	M 253.0	BOOL	
	mem2auto2	M 254.0	BOOL	
	mem3auto1	M 253.1	BOOL	
	mem3auto2	M 254.1	BOOL	
	mf9	M 8.7	BOOL	
	MICRO	FC 9	FC 9	Varrimentos solares, com ou sem sinal do sistema secundário
	MOD_68	FC 8	FC 8	Seleção do modo de funcionamento
	mw_volt	MW 100	INT	
	P_INTERRUPT	OB 40	OB 40	HW Interrupt FM350-1
	POS_CTRL	FC 1	FC 1	Mode, Commands and Datarecords
	POS_DIAG	FC 2	FC 2	Diagnostic Interrupt Information
	POS_INIT	FC 0	FC 0	Initialize channel datablock
	POS_MSRRM	FC 3	FC 3	Measurement Values
	pot	MD 35	REAL	
	pot2	MD 39	REAL	
	pot3	MD 52	REAL	
	potact1	MD 56	REAL	
	potact2	MD 61	REAL	
	potact3	MD 65	REAL	
	PRE_MOD	FC 12	FC 12	Código de interligação entre funções tecnológicas
	PROG_ENC_1	FB 300	FB 300	Programa Encoder 1 (FM350-1)
	PROG_ENC_2	FB 400	FB 400	Programa Encoder 2 (FM350-1)
	PROG_M1	FC 100	FC 100	Função principal FM353 (motor1)
	PROG_M2	FC 200	FC 200	Função principal FM353 (motor2)
	RD_REC	SFC 59	SFC 59	Read a Data Record
	RD_SINFO	SFC 6	SFC 6	Read OB Start Information
	RDSYSST	SFC 51	SFC 51	Read a System Status List or Partial List
	RESEARCH_MODE	M 26.0	BOOL	
	research_num	MW 80	INT	

Status	Symbol	Address	Data type	Comment
	RESET_FC	FC 13	FC 13	Função do Reset (F7 HMI)
	reset_m	M 7.1	BOOL	
	sis2_mem	M 69.0	BOOL	
	START_UP	OB 100	OB 100	Restart (start-up)
	stop_m	M 7.0	BOOL	
	SUB_PROG_M1_1	FC 101	FC 101	Ref. Pt. Approach (motor 1)
	SUB_PROG_M1_2	FC 102	FC 102	Deslocamento para posição de máx. pot. (motor 1)
	SUB_PROG_M1_3	FC 103	FC 103	Varrimento (motor 1)
	SUB_PROG_M2_1	FC 201	FC 201	Ref. Pt. Approach (motor 1)
	SUB_PROG_M2_2	FC 202	FC 202	Deslocamento para posição de máx. pot. (motor 2)
	SUB_PROG_M2_3	FC 203	FC 203	Varrimento (motor 2)
	SVSA_OB1	OB 1	OB 1	Chama as funções que constituem o programa a utilizador
	SYNC_FC	FC 14	FC 14	Confirma sincronização (eixo 1 e eixo 2)
	SYNC_OK	M 25.0	BOOL	
	t_20s	MW 70	S5TIME	
	t_hmi	MW 128	INT	
	t_in	MW 120	INT	
	t_out	MW 122	S5TIME	
	Tem_aux_fc8	T 40	TIMER	
	tem2auto1	T 11	TIMER	
	temp_aux1_fc103	T 30	TIMER	
	temp_aux1_fc12	T 100	TIMER	
	temp_aux1_fc203	T 25	TIMER	
	temp_aux15_fc6	T 15	TIMER	
	temp_aux16_fc6	T 16	TIMER	
	temp_aux17_fc6	T 17	TIMER	
	temp_aux18_fc6	T 18	TIMER	
	temp_aux19_fc6	T 19	TIMER	
	temp_aux2_fc103	T 31	TIMER	
	temp_aux2_fc12	T 101	TIMER	
	temp_aux2_fc203	T 26	TIMER	
	temp1auto1	T 10	TIMER	
	temp1auto2	T 12	TIMER	
	temp2auto2	T 13	TIMER	
	time_auto	M 6.4	BOOL	
	time_busca	T 14	TIMER	
	time_pulse	M 6.7	BOOL	

Status	Symbol	Address	Data type	Comment
	TIMER	FC 7	FC 7	Atribuição do intervalo entre varrimentos
	TRAVEL_M_M1_m	M 7.2	BOOL	
	TRAVEL_M_M2_m	M 7.4	BOOL	
	TRAVEL_P_M1_m	M 7.3	BOOL	
	TRAVEL_P_M2_m	M 7.5	BOOL	
	ttt2	MW 72	INT	
	ttt3	MW 74	S5TIME	
	ttt4	MW 76	INT	
	UDT_ENC1	UDT 3	UDT 3	Data type for counter channel (Encoder 1)
	UDT_ENC2	UDT 4	UDT 4	Data type for counter channel (Encoder 2)
	UDT_M1	UDT 1	UDT 1	Data type for stepper channel (Motor 1)
	UDT_M2	UDT 2	UDT 2	Data type for stepper channel (Motor 2)
	VAT_12	VAT 12		arduino
	VAT_8	VAT 8		
	VAT_9	VAT 9		
	VAT_CNT_EXAM_M1	VAT 6		
	VAT_CNT_EXAM_M2	VAT 7		
	VAT1_M1	VAT 1		
	VAT1_M2	VAT 4		
	VAT2	VAT 2		
	VAT3	VAT 3		
	VLEVEL1_M1_OP	MD 15	DWORD	
	VLEVEL1_M2_OP	MD 19	DWORD	
	VOLT	FC 4	FC 4	Determinação do valor de potencia maxima e ângulos dos eixos (modo research)
	vol_sol	PIW 288	WORD	Entrada analógica: sinal da célula FV
	WR_REC	SFC 58	SFC 58	Write Data Record
	zero pulse	M 245.1	BOOL	

Tópico 3: Lista de blocos

SVSA\SIMATIC 300 Station\CPU 315-2 DP\S7 Program(1)\Blocks

Object name	Symbolic name	Created in language	Size in the work memc	Type	Version (Header)
System data	---	---	---	SDB	---
OB1		STL	846	Organization Block	2.0
OB40	HW_INT0	STL	58	Organization Block	3.0
OB82		STL	340	Organization Block	2.0
OB100		STL	328	Organization Block	2.0
FB300	CNT_EXAM	STL	746	Function Block	3.0
FB400	CNT_EXAM_M2	STL	746	Function Block	3.0
FC0	POS_INIT	STL	178	Function	2.2
FC1	POS_CTRL	STL	3000	Function	2.3
FC2	POS_DIAG	STL	222	Function	2.0
FC3	POS_MSRRM	STL	212	Function	2.0
FC4	volt	STL	462	Function	0.1
FC5	clear	STL	170	Function	0.1
FC6	time	STL	550	Function	0.1
FC7	period	STL	112	Function	0.1
FC8	mem modos	STL	220	Function	0.1
FC100	MODE_EX	STL	230	Function	2.0
FC101	EXAMPLE1	STL	244	Function	2.1
FC102	EXAMPLE2	STL	368	Function	2.0
FC103	EXAMPLE3	STL	260	Function	2.0
FC104	POSITION	STL	112	Function	0.1
FC105	POSITION_AUTO	STL	112	Function	0.1
FC200	MODE_EX_M2	STL	230	Function	2.0
FC201	EXAMPLE1_M2	STL	244	Function	2.1
FC202	EXAMPLE2_M2	STL	368	Function	2.0
FC203	EXAMPLE3_M2	STL	246	Function	2.0
FC204	POSITION_M2	STL	112	Function	0.1
FC205	POSITION_M2_AUTO	STL	112	Function	0.1
FC301	DIAG_INF	STL	246	Function	3.0

Page 1 of 3

SVSA\SIMATIC 300 Station\CPU 315-2 DP\S7 Program(1)\Blocks

Object name	Symbolic name	Created in language	Size in the work memc	Type	Version (Header)
FC302	CNT_CTL1	STL	796	Function	3.0
FC401	DIAG_INF_M2	STL	246	Function	3.0
FC402	CNT_CTL1_M2	STL	796	Function	3.0
FC500	BEGIN_DATA	STL	60	Function	0.1
FC600	Counter_revolution_M1	STL	92	Function	0.1
FC700	Counter_revolution_M2	STL	92	Function	0.1
DB1	db1	DB	552	Data block derived f	0.1
DB2	db2	DB	552	Data block derived f	0.1
DB3	CNT_CHAN1	DB	106	Data block derived f	0.1
DB4	CNT_CHAN1_M2	DB	106	Data block derived f	0.1
DB100	DBEX	DB	54	Data Block	2.0
DB200	DBEX_200	DB	54	Data Block	2.0
DB300	CNT_EXAM_DB	DB	78	Instance data block	0.1
DB400	CNT_EXAM_DB_M2	DB	78	Instance data block	0.1
UDT1		STL	---	Data Type	0.0
UDT2		STL	---	Data Type	0.0
UDT3	CNT_CHANATYPE1	STL	---	Data Type	3.0
UDT4	CNT_CHANATYPE1_M2	STL	---	Data Type	3.0
Teste	Teste		---	Variable Table	0.1
teste auto	teste auto		---	Variable Table	0.1
VAT1	VAT1		---	Variable Table	0.1
VAT1_M2	VAT1_M2		---	Variable Table	0.1
VAT2	VAT2		---	Variable Table	0.1
VAT3	VAT3		---	Variable Table	0.1
VAT_11	VAT_11		---	Variable Table	0.1
VAT_8	VAT_8		---	Variable Table	0.1
VAT_9	VAT_9		---	Variable Table	0.1
VAT_CNT_EXAM	VAT_CNT_EXAM		---	Variable Table	3.0
VAT_CNT_EXAM_M2	VAT_CNT_EXAM_M2		---	Variable Table	3.0

(Anexo M) 9

Page 2 of 3

SVSA\SIMATIC 300 Station\CPU 315-2 DP\S7 Program(1)\Blocks

Object name	Symbolic name	Created in language	Size in the work memc	Type	Version (Header)
SFC6	RD_SINFO	STL	---	System function	1.0
SFC51	RDSYSST	STL	---	System function	1.0
SFC58	WR_REC	STL	---	System function	1.0
SFC59	RD_REC	STL	---	System function	1.0
			14736		

Tópico 4: Programação – STEP 7

OB1 - <offline>

"SVSA_OB1" Chama as funções que constituem o programa a utilizador

Name: SVSA_OB1 **Family:**
Author: EACF **Version:** 2.0
Block version: 2
Time stamp Code: 08/01/2012 04:55:04 PM
Interface: 02/24/2000 11:23:07 AM
Lengths (block/logic/data): 00600 00474 00026

Name	Data Type	Address	Comment
TEMP		0.0	
OB1_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (coming event), bits 4-7 = 1 (Event class 1)
OB1_SCAN_1	Byte	1.0	1 (cold restart scan 1 of OB 1), 3 (scan 2-n of OB 1)
OB1_PRIORITY	Byte	2.0	1 (priority of 1 is lowest)
OB1_OB_NUMBR	Byte	3.0	1 (organization block 1, OB1)
OB1_RESERVED_1	Byte	4.0	Reserved for system
OB1_RESERVED_2	Byte	5.0	Reserved for system
OB1_PREV_CYCLE	Int	6.0	Cycle time of previous OB1 scan (milliseconds)
OB1_MIN_CYCLE	Int	8.0	Minimum cycle time of OB1 (milliseconds)
OB1_MAX_CYCLE	Int	10.0	Maximum cycle time of OB1 (milliseconds)
OB1_DATE_TIME	Date_And_Time	12.0	Date and time OB1 started

Block: OB1 SVSA OB1

Chama todas as funções com o código do programa de utilizador

Network: 1 Atenção se alterar da ordem das FCs / FBs

```
CALL "Call_FB21"           FC16           -- Ajuda a chamar mais facilmente, no OB1, a FB21
CALL "HMI_WCC"            FC15           -- Auxiliar na criação de botões (HMI - WinCC)
CALL "TIMER"              FC7            -- Atribuição do intervalo entre varrimentos
CALL "AUTO"                FC6            -- Transição entre o modo de varrimento e o o modo de posicionamento (
com F5)
CALL "VOLT"                FC4            -- Determinação do valor de potencia maxima e ângulos dos eixos (modo
research)
CALL "MICRO"               FC9            -- Varrimentos solares, com ou sem sinal do sistema secundário
CALL "POS_CTRL"           FC1            -- Mode, Commands and Datarecords
DB_NO :=1
RET_VAL:="DBEX_100_M1".ERR_CODE_CTRL DB100.DBW2 -- Error code FC POS_CTRL
CALL "POS_CTRL"           FC1            -- Mode, Commands and Datarecords
DB_NO :=2
RET_VAL:="DBEX_200_M2".ERR_CODE_CTRL_M2 DB200.DBW2 -- Error code FC POS_CTRL
CALL "MCU_ENC1"           FC10           -- FM350-1 Encoder 1 (module adress, channel adress, user data length)
```

```
// Call FB Example with Instance DB.
// The block parameters are written and read via the VAT supplied.
// The parameters must not be interconnected in OB1.
```

```
CALL "PROG_ENC_1" , "DB_PROG_ENC1"   FB300 / DB300 -- Programa Encoder 1 (FM350-1) / DB300 (FB300 DB instance)
SW_GATE :=
GATE_STP :=
OT_ERR_A :=
DIAG_ERR_A :=
HW_INT_ERR_A :=
SET_D00 :=
SET_D01 :=
SET_L_DIRECT :=
SET_L_PREPAR :=
SET_T_CMP_V1 :=
SET_T_CMP_V2 :=
SET_C_DOPARA :=
SET_RES_SYNC :=
SET_RES_ZERO :=
```

```

ENSET_UP      :=
ENSET_DN      :=
CTRL_DO0      :=
CTRL_DO1      :=
L_DIRECT_VAL  :=
L_PREPAR_VAL  :=
T_CMP_V1_VAL  :=
T_CMP_V2_VAL  :=
DO0_MODE      :=
DO1_MODE      :=
HYSTERESIS    :=
PULSE_DURATION:=
OT_ERR        :=
DIAG_ERR      :=
HW_INT_ERR    :=
STS_RUN       :=
STS_DIR       :=
STS_ZERO      :=
STS_OFLW      :=
STS_UFLW      :=
STS_SW_G      :=
STS_GATE      :=
OT_ERR_B      :=
LATCH_LOAD    :=
ACT_CNTV      :=
L_DIRECT      :=
L_PREPAR      :=
T_CMP_V1      :=
T_CMP_V2      :=
C_DOPARA      :=
RES_SYNC      :=
RES_ZERO      :=
DIAG_INF      :=

```

```
CALL "MCU_ENC2"          FC11          -- FM350-1 Encoder 2 (module adress, channel adress, user data length)
```

```
// Call FB Example with Instance DB.
// The block parameters are written and read via the VAT supplied.
// The parameters must not be interconnected in OB1.
```

```
CALL "PROG_ENC_2" , "DB_PROG_ENC2"    FB400 / DB400    -- Programa Encoder 2 (FM350-1) / DB400 (FB400 DB instance)
SW_GATE        :=
GATE_STP       :=
OT_ERR_A       :=
DIAG_ERR_A     :=
HW_INT_ERR_A   :=
SET_DO0        :=
SET_DO1        :=

```

Page 3 of 5

```

SET_L_DIRECT  :=
SET_L_PREPAR  :=
SET_T_CMP_V1  :=
SET_T_CMP_V2  :=
SET_C_DOPARA  :=
SET_RES_SYNC  :=
SET_RES_ZERO  :=
ENSET_UP      :=
ENSET_DN      :=
CTRL_DO0      :=
CTRL_DO1      :=
L_DIRECT_VAL  :=
L_PREPAR_VAL  :=
T_CMP_V1_VAL  :=
T_CMP_V2_VAL  :=
DO0_MODE      :=
DO1_MODE      :=
HYSTERESIS    :=
PULSE_DURATION:=
OT_ERR        :=
DIAG_ERR      :=
HW_INT_ERR    :=
STS_RUN       :=
STS_DIR       :=
STS_ZERO      :=
STS_OFLW      :=
STS_UFLW      :=
STS_SW_G      :=
STS_GATE      :=
OT_ERR_B      :=
LATCH_LOAD    :=
ACT_CNTV      :=
L_DIRECT      :=
L_PREPAR      :=
T_CMP_V1      :=
T_CMP_V2      :=
C_DOPARA      :=
RES_SYNC      :=
RES_ZERO      :=
DIAG_INF      :=

```

```

A "reset_m"          M7.1
JC RSET
CALL "SYNC_FC"       FC14          -- Confirma sincronização (eixo 1 e eixo 2)
CALL "PRE_MOD"       FC12          -- Código de interligação entre funções tecnológicas

```

```
//*****
```

(Anexo M) 13

Page 4 of 5


```
// Funções Motor 1
//*****

CALL "SUB_PROG_M1_1"          FC101          -- Ref. Pt. Aproach (motor 1)
CALL "SUB_PROG_M1_3"          FC103          -- Varrimento (motor 1)
CALL "SUB_PROG_M1_2"          FC102          -- Deslocamento para posição de máx. pot. (motor 1)

// Copy checkback signals DB_FM DBEX motor 1
CALL "PROG_M1"                FC100          -- Função principal FM353 (motor1)

//*****
// Funções Motor 2
//*****

CALL "SUB_PROG_M2_1"          FC201          -- Ref. Pt. Aproach (motor 1)
CALL "SUB_PROG_M2_3"          FC203          -- Varrimento (motor 2)
CALL "SUB_PROG_M2_2"          FC202          -- Deslocamento para posição de máx. pot. (motor 2)

// Copy checkback signals DB_FM DBEX motor 2
CALL "PROG_M2"                FC200          -- Função principal FM353 (motor2)

CALL "COUNTER_REV_ENC1"       FC600          -- New counter value Encoder 1
CALL "COUNTER_REV_ENC2"       FC700          -- New counter value Encoder 2

RSET: A "reset_m"              M7.1
CALL "RESET_FC"                FC13          -- Função do Reset (F7 HMI)

CALL "BEGIN_DATA"             FC500          -- Após fazer reset introduz valor standard de velocidade M1 e M2

BE
```

Page 5 of 5

OB100 - <offline>

```
"START_UP" Restart (start-up)
Name: START_UP      Family:
Author: SIEMENS     Version: 2.0
                  Block version: 2
Time stamp Code:   07/18/2012 12:17:54 PM
                  Interface: 02/24/2000 11:23:08 AM
Lengths (block/logic/data): 00416 00292 00030
```

Name	Data Type	Address	Comment
TEMP		0.0	
OB100_EV_CLASS	Byte	0.0	16#13, Event class 1, Entering event state, Event logged in diagnostic buffer
OB100_STARTUP	Byte	1.0	16#81/82/83/84 Method of startup
OB100_PRIORITY	Byte	2.0	27 (priority of 1 is lowest)
OB100_ORG_NUMBR	Byte	3.0	100 (organization block 100, OB100)
OB100_RESERVED_1	Byte	4.0	Reserved for system
OB100_RESERVED_2	Byte	5.0	Reserved for system
OB100_STOP	Word	6.0	Event that caused CPU to stop (16#4xxx)
OB100_START_INFO	DWord	8.0	Information on how system started
OB100_DATE_TIME	Date_And_Time	12.0	Date and time OB100 started

Block: OB100 Cold restart

Network: 1 INITIALIZATION OF THE USER DATA BLOCK

```
CALL "POS_INIT" // Initialization of the user DB FC0 -- Initialize channel dat
ablock
DB_NO :=1 // DB number
CH_NO :=1 // Channel number
LADDR :=352 // Module address
RET_VAL:="DBEX_100_M1".ERR_CODE_INIT // Error code DB100.DBW0 -- Error code FC POS_INI
T
```

(Anexo M) 14

Page 1 of 3

```

L   "DBEX_100_M1".ERR_CODE_INIT    // Error code evaluation          DB100.DBW0    -- Error code FC POS_INI
L   B#16#0
==I
R   "DBEX_100_M1".INIT_ERR         // Reset error for INIT function    DB100.DBX10.6 -- Error FC POS_INIT
JC  NWE
S   "DBEX_100_M1".INIT_ERR         // Flag error for INIT function     DB100.DBX10.6 -- Error FC POS_INIT
NWE: NOP 0

```

Network: 2 INITIALIZATION OF THE EXAMPLE DB

```

OPN  "DBEX_100_M1"                  DB100          -- Interface (FC100, DB1, UDT1)
L   B#16#0                          // Clear DBEX
T   DBD 0                            // Begin with DBEX.DBD0
T   DBD 4                            //
T   DBD 8                            //
T   DBD 12                           //
T   DBW 16

```

Network: 3 SETTING DEFAULTS IN THE EXAMPLE DB

```

L   B#16#64
T   "DBEX_100_M1".OVERRIDE         // Set override to 100%          DB100.DBB6    -- Override
SET
S   "DBEX_100_M1".SERVO_EN         // Set servo enable              DB100.DBX9.1  -- Servo enable
S   "DBEX_100_M1".DRV_EN           // Set drive enable              DB100.DBX9.0  -- Drive enable
S   "DBEX_100_M1".EX3.READ_EN     // Set read enable (EX3)        DB100.DBX16.2 -- Read enable

```

Network: 4 INITIALIZATION OF THE USER DATA BLOCK

```

CALL "POS_INIT"                    // Initialization of the user DB          FC0            -- Initialize channel da
                                     tablock
DB_NO :=2                          // DB number
CH_NO :=1                          // Channel number
LADDR :=368                        // Module address
RET_VAL:="DBEX_200_M2".ERR_CODE_INIT_M2 // Error code          DB200.DBW0    -- Error code FC POS_IN
                                     IT

```

Page 2 of 3

```

L   "DBEX_200_M2".ERR_CODE_INIT_M2 // Error code evaluation          DB200.DBW0    -- Error code FC POS_IN
L   B#16#0
==I
R   "DBEX_200_M2".INIT_ERR_M2     // Reset error for INIT function    DB200.DBX10.6 -- Error FC POS_INIT
JC  NWE
S   "DBEX_200_M2".INIT_ERR_M2     // Flag error for INIT function     DB200.DBX10.6 -- Error FC POS_INIT
NWE: NOP 0

```

Network: 5 INITIALIZATION OF THE EXAMPLE DB

```

OPN  "DBEX_200_M2"                  DB200          -- Interface (FC200, DB2, UDT2)
L   B#16#0                          // Clear DBEX
T   DBD 0                            // Begin with DBEX.DBD0
T   DBD 4                            //
T   DBD 8                            //
T   DBD 12                           //
T   DBW 16

```

Network: 6 SETTING DEFAULTS IN THE EXAMPLE DB

```

L   B#16#64
T   "DBEX_200_M2".OVERRIDE_M2     // Set override to 100%          DB200.DBB6    -- Override
SET
S   "DBEX_200_M2".SERVO_EN_M2     // Set servo enable              DB200.DBX9.1  -- Servo enable
S   "DBEX_200_M2".DRV_EN_M2       // Set drive enable              DB200.DBX9.0  -- Drive enable
S   "DBEX_200_M2".EX3_M2.READ_EN_M2 // Set read enable (EX3)        DB200.DBX16.2 -- Read enable

```

BE

OB40 - <offline>

"P_INTERRUPT" HW Interrupt FM350-1
Name: HW_INT0 **Family:** FM_CNT_1
Author: FM **Version:** 3.0
Block version: 2
Time stamp Code: 01/21/2011 11:54:41 PM
Interface: 12/06/1996 02:51:35 PM
Lengths (block/logic/data): 00134 00022 00020

Name	Data Type	Address	Comment
TEMP		0.0	
OB40_EV_CLASS	Byte	0.0	Bits 0-3 = 1 (Coming event), Bits 4-7 = 1 (Event class 1)
OB40_STRT_INF	Byte	1.0	16#41 (OB 40 has started)
OB40_PRIORITY	Byte	2.0	Priority of OB Execution
OB40_OB_NUMBR	Byte	3.0	40 (Organization block 40, OB40)
OB40_RESERVED_1	Byte	4.0	Reserved for system
OB40_IO_FLAG	Byte	5.0	16#54 (input module), 16#55 (output module)
OB40_MDL_ADDR	Word	6.0	Base address of module initiating interrupt
OB40_POINT_ADDR	DWord	8.0	Interrupt status of the module
OB40_DATE_TIME	Date_And_Time	12.0	Date and time OB40 started

Block: OB40 "Hardware Interrupt"

Network: 1 1=Hardware interrupt received

```

SET
S   "DB_PROG_ENC1".HW_INT_ERR // Display Hardware interrupt          DB300.DBX24.2    -- 1=Hardware interrupt rece
                                     ived

```

Page 1 of 2

Network: 2 1=Hardware interrupt received

```

SET
S   "DB_PROG_ENC2".HW_INT_ERR // Display Hardware interrupt          DB400.DBX24.2    -- 1=Hardware interrupt rece
                                     ived

```

OB82 - <offline>

"DIAGNOSTICS" Diagnostics FM353 e FM350-1
Name: DIAGNOST **Family:**
Author: SIEMENS **Version:** 2.0
 Block version: 2
Time stamp Code: 07/18/2012 12:17:30 PM
 Interface: 02/24/2000 11:23:08 AM
Lengths (block/logic/data): 00470 00304 00026

Name	Data Type	Address	Comment
TEMP		0.0	
OB82_EV_CLASS	Byte	0.0	16#39, Event class 3, Entering event state, Internal fault event
OB82_FLT_ID	Byte	1.0	16#XX, fault identification code
OB82_PRIORITY	Byte	2.0	26/28 (priority of 1 is lowest)
OB82_OB_NUMBR	Byte	3.0	82 (organization block 82, OB82)
OB82_RESERVED_1	Byte	4.0	Reserved for system
OB82_IO_FLAG	Byte	5.0	Input (01010100), Output (01010101)
OB82_MDL_ADDR	Int	6.0	Base address of module with fault
OB82_MDL_DEFECT	Bool	8.0	Module defective
OB82_INT_FAULT	Bool	8.1	Internal fault
OB82_EXT_FAULT	Bool	8.2	External fault
OB82_PNT_INFO	Bool	8.3	Point information
OB82_EXT_VOLTAGE	Bool	8.4	External voltage low
OB82_FLD_CONNCTR	Bool	8.5	Field wiring connector missing
OB82_NO_CONFIG	Bool	8.6	Module has no configuration data
OB82_CONFIG_ERR	Bool	8.7	Module has configuration error
OB82_MDL_TYPE	Byte	9.0	Type of module
OB82_SUB_NDL_ERR	Bool	10.0	Submodule missing or faulty
OB82_COMM_FAULT	Bool	10.1	Communication fault
OB82_MDL_STOP	Bool	10.2	Module is stopped
OB82_WTCH_DOG_FLT	Bool	10.3	Watch dog timer stopped module
OB82_INT_PS_FLT	Bool	10.4	Internal power supply fault

Page 1 of 3

Name	Data Type	Address	Comment
OB82_PRIM_BATT_FLT	Bool	10.5	Primary battery is in fault
OB82_BCKUP_BATT_FLT	Bool	10.6	Backup battery is in fault
OB82_RESERVED_2	Bool	10.7	Reserved for system
OB82_RACK_FLT	Bool	11.0	Rack fault, only for bus interface module
OB82_PROC_FLT	Bool	11.1	Processor fault
OB82_EPROM_FLT	Bool	11.2	EPROM fault
OB82_RAM_FLT	Bool	11.3	RAM fault
OB82_ADU_FLT	Bool	11.4	ADU fault
OB82_FUSE_FLT	Bool	11.5	Fuse fault
OB82_HW_INTR_FLT	Bool	11.6	Hardware interrupt input in fault
OB82_RESERVED_3	Bool	11.7	Reserved for system
OB82_DATE_TIME	Date_And_Time	12.0	Date and time OB82 started

Block: OB82 Diagnostic interrupt

Network: 1 1=Diagnostic interrupt received

```

SET
S   "DB_PROG_ENC1".DIAG_ERR      // Display diagnostic interrupt      DB300.DBX24.1      -- 1=Diagnostic interrupt r
   eceived
S   "DB_PROG_ENC1".DIAG_INF      // Activate DIAG_INF function      DB300.DBX36.7      -- 1=Read diagnostics (only
   for OB82)

CALL "DIAG_INF"      FC301      -- Read diagnostic data Enc
   oder 1
   DB_NO :=3      // DB for counter module
   RET_VAL:="DB_PROG_ENC1".SFC_ERR      // Error status      DB300.DBW40      -- SFC51 error status
   IN_DIAG:="DB_PROG_ENC1".DIAG_INF      DB300.DBX36.7      -- 1=Read diagnostics (only
   for OB82)

SET
S   "DB_PROG_ENC2".DIAG_ERR      // Display diagnostic interrupt      DB400.DBX24.1      -- 1=Diagnostic interrupt r
   eceived
S   "DB_PROG_ENC2".DIAG_INF      // Activate DIAG_INF function      DB400.DBX36.7      -- 1=Read diagnostics (only
   for OB82)

```

(Anexo M) 17

Page 2 of 3

```

CALL "DIAG_INF_M2"                                FC401          -- Read diagnostic data Enc
                                                    oder 2
DB_NO :=4                                         // DB for counter module
RET_VAL:="DB_PROG_ENC2".SFC_ERR // Error status      DB400.DBW40    -- SFC51 error status
IN_DIAG:="DB_PROG_ENC2".DIAG_INF                 DB400.DBX36.7 -- 1=Read diagnostics (only
                                                    for OB82)

```

Network: 2 QUERY WHETHER THE FM CAUSED THE INTERRUPT

```

L #OB82_MDL_ADDR                                #OB82_MDL_ADDR -- Base address of module with
                                                    fault
L "DB_UDT_M1".MOD_ADDR                          DB1.DBW0       -- Module address
==I // Is int. address=module add. of the FM ?
JCN ENDE

```

Network: 3 READ DIAGNOSTIC INTERRUPT DATA

```

SET
S "DBEX_100_M1".DIAG_RD // Initiate read function      DB100.DBX9.4  -- Job start FC POS_DIAG
R "DBEX_100_M1".DIAG_ERR // Reset error          DB100.DBX10.7 -- Error FC POS_DIAG

CALL "POS_DIAG" // Call FC diagnostic information      FC2          -- Diagnostic Interrupt I
                                                    nformation
DB_NO :=1 // DB number
RET_VAL:="DBEX_100_M1".ERR_CODE_DIAG // Return value  DB100.DBW4G   -- Error code FC POS_DIA
IN_DIAG:="DBEX_100_M1".DIAG_RD // Start bit for reading the diagnostic inform DB100.DBX9.4  -- Job start FC POS_DIAG
.

L "DBEX_100_M1".ERR_CODE_DIAG // Error code evaluation      DB100.DBW4G   -- Error code FC POS_DIA
L B#16#0
==I
R "DBEX_100_M1".DIAG_ERR // Reset error for DIAG function      DB100.DBX10.7 -- Error FC POS_DIAG
JC ENDE
S "DBEX_100_M1".DIAG_ERR // Flag error for DIAG function      DB100.DBX10.7 -- Error FC POS_DIAG

```

ENDE: BE

Page 3 of 3

FC0 - <offline>

```

"POS_INIT" Initialize channel datablock
Name: POS_INIT Family: FM_ST_SV
Author: AuD Version: 2.2
Block version: 2
Time stamp Code: 04/10/2002 03:50:22 PM
Interface: 08/31/2000 10:03:10 AM
Lengths (block/logic/data): 00256 00142 00004

```

Name	Data Type	Address	Comment
IN		0.0	
DB_NO	Int	0.0	channel datablock number
CH_NO	Int	2.0	channel number
LADDR	Int	4.0	module address
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
dbnr	Int	0.0	DB-Nr. Kanal-DB
chno	Int	2.0	CH_NO 0,1,2 als Merker
RETURN		0.0	
RET_VAL	Int	6.0	

Block: FC0 initialize channel datablock

Network: 1 CH_NO prüfen und Kanal-DB initialisieren

```

L #CH_NO // CH_NO <= 3 ? #CH_NO -- channel number
L B#16#3 // ja, ok BIE = 1 und weiter
<=I // nein, Fehler BIE = 0 und Ende
SAVE // BIE = VKE
NOT // NOT VKE
L B#16#FF // Wert RET_VAL des FC0 -1

```

(Anexo M) 18

Page 1 of 3

```

JC ENDE // Ende bei Fehler

L B#16#0
L #DB_NO // Kanal-DB indirekt öffnen #DB_NO -- channel datablock number
==I
L B#16#FE // Wert RET_VAL des FC0 -2
JC ENDE // Ende bei Fehler

L #DB_NO // Kanal-DB indirekt öffnen #DB_NO -- channel datablock number
T #dbnr #dbnr -- DB-Nr. Kanal-DB
OPN DB [#dbnr] #dbnr -- DB-Nr. Kanal-DB

// Löschen des AWDB-Bereiches DB0 bis DBB139

L B#16#0 // Beginn bei DBW 0
LAR1
L B#16#23 // Anzahl 35DW, bis DBB139
LOES: L B#16#0
T DBD [AR1,P#0.0] // löschen
+AR1 P#4.0 // erhöhen des AR1 um Doppelwordbreite
TAK // Anzahl Durchgänge =0 ?
LOOP LOES

L B#16#0
L #LADDR // log. Basisadresse in Kanal-DB MOD_ADR #LADDR -- module address
==I
JC ENDE
T DBW 0

L B#16#0 // CH_NO == 0 ?
L #CH_NO #CH_NO -- channel number
<>I
JC CH0 // im Akku CH_NO == 0,dann
L B#16#1 // 1 in DBW 2 eintragen
CH0: T DBW 2 // Awnst
DEC 1 // im Akku CH_NO - 1
T #chno // in chno merken #chno -- CH_NO 0,1,2 als Merker
SLD 3 // chno * 8, pro Kanal 8 Byte
L DBW 0 // CH_ADR = (MOD_ADR + chno * 8) * 8
+I
SLD 3
T DBD 4 // CH_ADR eintragen

L #chno #chno -- CH_NO 0,1,2 als Merker
L B#16#28
*I
T DBW 8 // DS_OFFS = chno * 40

```

Page 2 of 3

```

L B#16#0
ENDE: T #RET_VAL // RET_VAL des Bausteins (0, -1, -2) #RET_VAL
BE

```

FC1 - <offline>

"POS_CTRL" Mode, Commands and Datarecords
Name: POS_CTRL **Family:** FM_ST_SV
Author: AuD **Version:** 2.3
Block version: 2
Time stamp Code: 04/10/2002 03:50:46 PM
Interface: 10/06/2000 03:04:47 PM
Lengths (block/logic/data): 03400 02964 00022

Name	Data Type	Address	Comment
IN		0.0	
DB_NO	Int	0.0	channel datablock number
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
rettar1	DWord	0.0	Zwischenspeicher für AR1
dbnr	Int	4.0	DB-Nr. Kanal-DB
DS_any	Any	6.0	record Parameter des SFC
DS_nr_wr	Byte	16.0	DSNr. schreibend
DS_nr_rd	Byte	17.0	DSNr. lesend
sfc_busy_wr	Bool	18.0	BUSY-Ausgang des SFC58
sfc_busy_rd	Bool	18.1	BUSY-Ausgang des SFC59
start	Bool	18.2	Speicher für gesetzten Start
RETURN		0.0	
RET_VAL	Int	2.0	

Block: FC1 Mode, Commands and Read Datarecords

Page 1 of 24

Network: 1 Rückmeldebytes lesen

BIE setzen, AR1 retten, Kanal-DB öffnen,
 Rückmeldesignale lesen

```

TAR1 #rettar1 // AR1 retten #rettar1 -- Zwischenspeicher für AR1
L #DB_NO // Kanal-DB indirekt öffnen #DB_NO -- channel datablock number
T #dbnr #dbnr -- DB-Nr. Kanal-DB
OPN DB [#dbnr] #dbnr -- DB-Nr. Kanal-DB

LAR1 DBD 4 // CH_ADR (Kanaladresse in AR1 laden)
L PIW [AR1,P#0.0] // CHECKBACK_SIGNALS Byte 0
CAW // Bytes tauschen im gelesenen Word
T DBB 22 // AWWST
R DBX 68.1 // Auftragsbearbeitung DS schreiben nicht möglich
R DBX 68.3 // Auftragsbearbeitung DS lesen nicht möglich

```

Network: 2 Anlaufbearbeitung

Anlaufbits kontrollieren

```

A DBX 22.5 // FMNEUSTQ_Rückmeldung
A DBX 22.6 // FMNEUST_Rückmeldung
O
AN DBX 22.5 // FMNEUSTQ_Rückmeldung
AN DBX 22.6 // FMNEUST_Rückmeldung
JCN ANL
L B#16#0 // bei FM-Anlauf undefiniert
T #RET_VAL // RET_VAL = 0 #RET_VAL
T DBW 108 // interner Bereich
S DBX 68.1 // Auftragsbearbeitung DS schreiben nicht möglich
S DBX 68.3 // Auftragsbearbeitung DS lesen nicht möglich

AN DBX 22.6 // Anlaufkoord. bei DBX22.5+22.6 = 0
R DBX 14.6 // DBX14.6 rücksetzen und zur FM
JU STEN // Steuersignale uebertragen, Bausteinende

ANL: A DBX 22.6 // FMNEUST_Rückmeldung
= DBX 14.6 // FBNEUST_Steuersignal zur Anlaufbearbeitung setzen
JCN SFBI // Auftrag Status- und Fehlerbit ?
L B#16#0 // bei FM-Anlauf
T #RET_VAL // RET_VAL = 0 #RET_VAL
T DBW 108 // interner Bereich

```

(Anexo M) 20

Page 2 of 24

```

S   DBX  68.1 // Auftragsbearbeitung DS schreiben nicht möglich
S   DBX  68.3 // Auftragsbearbeitung DS lesen nicht möglich
JU  STEN           // Steuersignale uebertragen, Bausteinende

```

Network: 3	Bereiche Löschen
------------	------------------

Status- und Fehlermeldungen der Auftragsverwaltung, sowie interne Bereiche löschen

```

SFBI: AN  DBX  69.1 // Auftrag Status-, Fehlerbits löschen ?
JC      AUF           // kein Löschen der Status-, und Fehlerbits der Schreibaufträge
R      DBX  69.1 // Auftrag Status-, Fehlerbits löschen zurücksetzen
L      B#16#0        // Fehlerbits löschen
T      DBD  44        // Statusbereich Einstellungen, Kommandos
T      DBD  48        // Statusbereich Schreibaufträge
T      DBW  52        // Statusbereich Leseaufträge
T      DBD  54        // Fehlerbereich Schreibaufträge
T      DBD  58        // Fehlerbereich Einstellungen, Kommandos
T      DBW  62        // Fehlerbereich Leseaufträge
T      DBW  66        // Fehlercode
T      DBW  106       // interne Schreibauftragsmerker
T      DBD  108       // interner Fehlercode Schreibauftrag
T      DBD  120       // interner Fehlercode Leseauftrag
T      DBW  126       // interner Bereich DS12
T      DBD  128       // interner Bereich Schreibaufträge
T      DBW  132       // interner Bereich Leseaufträge

```

Network: 4	Auftragsbearbeitung
------------	---------------------

Rückmeldesignale der FM lesen, Betriebsart auf Übereinstimmung Steuer- und Rückmeldesignale überprüfen

```

AUF: L  PID [AR1,P#0.0] // erst jetzt alle lesen
      T  DBD  22        // CHECKBACK_SIGNALS (Byte 0-3)
      L  PID [AR1,P#4.0]
      T  DBD  26        // CHECKBACK_SIGNALS (Byte 4-7)

// Abfrage nach PARA entfällt, da bei PARA=0 immer FM-Anlauf aktiv

A   DBX  22.1        // [TFGS]? (Testbetrieb?)
S   DBX  68.1        // Auftragsbearbeitung nicht möglich
JC  OURV             // RET_VAL = 0 des SFCs ausgeben

L   DBB  24          // BAR

```

Page 3 of 24

```

L      B#16#0
==I
S   DBX  68.1        // Auftragsbearbeitung DS schreiben nicht möglich
S   DBX  68.3        // Auftragsbearbeitung DS lesen nicht möglich
JC  OURV             // RET_VAL = 0 des SFCs ausgeben
L   DBB  24          // BAR
L   DBB  16          // BA
<>I
S   DBX  68.1        // Auftragsbearbeitung DS schreiben nicht möglich
L   B#16#0
JC  OURV             // RET_VAL = 0 des SFCs ausgeben
R   DBX  68.1        // Auftragsbearbeitung DS schreiben möglich
R   DBX  68.3        // Auftragsbearbeitung DS lesen möglich

```

Network: 5	Schreibaufträge
------------	-----------------

Schreibaufträge erkennen, Anstoß setzen

```

A   DBX  68.0        // Auftrag läuft bereits
JC  DIQ
L   DBW  34          // Einstellungen
AW  W#16#E3FC       // Maske für Auftragsbits
T   DBW  34          // zurückschreiben
T   DBW  100
L   DBW  124        // Status der Einstellungen
<>I
S   DBX  68.0        // Anstoß Auftrag
=   DBX  106.5       // DS11

L   DBW  36          // Kommandos
AW  W#16#6F         // Maske für Auftragsbits
T   DBW  36          // zurückschreiben
L   B#16#0
T   DBW  126        // alten Auftrag löschen
<>I
S   DBX  68.0        // Anstoß Auftrag
=   DBX  106.6       // DS12
JCN ntr
L   DBW  36
T   DBW  126

ntr: L   DBD  38          // Schreibaufträge
      AD  DW#16#DFFB0300 // Maske für Auftragsbits
      T   DBD  38
      L   B#16#0
      <>D

```

(Anexo M) 21

Page 4 of 24


```
S DBX 68.0 // Anstoß Auftrag
= DBX 106.7 // DS anderer Schreibaufträge
```

Network: 6 DIQ erzeugen

kein Anstoß: Statussignale rücksetzen und Daten lesen
Anstoß: DIQ setzen

```
DIQ: AN DBX 68.0 // Anstoß Auftrag läuft gelöscht
      R DBX 14.0 // DIQ löschen

      JC BA //***Anstoß schreiben nicht gesetzt: FM-Steuersignale rücksetzen und Daten lesen ****

      A DBX 68.0 // Anstoß Auftrag läuft gesetzt
      AN DBX 106.0 // Auftrag läuft nicht gesetzt
      S DBX 14.0 // DIQ setzen
      S DBX 106.0 // Auftrag läuft setzen
      R DBX 106.1 // Auftrag beendet löschen
      JCN AUFT

      L B#16#0 // Auftragsbearbeitung beginnt
      T DBW 108 // Zwischenspeicher RET_VAL löschen
```

Network: 7 Kommandos, Einstellungen, Schreibaufträge erfassen

Aufträge nach Prioritäten selektieren

```
AUFT: L B#16#1
      T DBW 110 // RET_VAL aktiv setzen
      SET // VKE setzen
      A DBX 106.6 // DS12 aktiv ?
      JC DS12 // DS12

      A DBX 106.5 // DS11 aktiv ?
      JC DS11 // DS11
```

// Auftragsablauf

```
L B#16#1
L DBW 134 // RET_VAL
==I
JC Adz // Auftrag dezentral
L DBW 104 // Auftragsspeicher
L B#16#0
```

Page 5 of 24

```
<>I
JC Aufz // Auftrag suchen
L DBW 38
T DBW 138 // interner Status
Adz: L DBW 138
      T DBW 104
```

Network: 8 Datensatz - Auftrag suchen

Datensätze erfassen

```
Aufs: L B#16#0
      T DBW 126 // alte Schreibaufträge löschen
      T DBD 128
      A DBX 104.0
      R DBX 104.0
      JC DS2s // DS2 Geschwindigkeitsstufen
      A DBX 104.1
      R DBX 104.1
      JC DS3s // DS3 Spannungs-/Frequenzstufen
      A DBX 104.2
      R DBX 104.2
      JC DS4s // DS4 Schrittmaß
      A DBX 104.3
      R DBX 104.3
      JC DS7s // DS7 MDI-Satz
      A DBX 104.4
      R DBX 104.4
      JC D17s // DS17 MDI-Satz fliegend
      A DBX 104.6
      R DBX 104.6
      JC D22s // DS22 Bezugspunktsetzen
      A DBX 104.7
      R DBX 104.7
      JC D14s // DS14 Istwert setzen
      A DBX 105.0
      R DBX 105.0
      JC D15s // DS15 Istwert setzen fliegend
      A DBX 105.1
      R DBX 105.1
      JC D13s // DS13 Nullpunktverschiebung
      A DBX 105.3
      R DBX 105.3
      JC DS9s // DS9 Parameter/Daten
      A DBX 105.4
      R DBX 105.4
```

(Anexo M) 22

Page 6 of 24

```

JC D16s // DS16 dig. E/A
A DBX 105.5
R DBX 105.5
JC D18s // DS18 Programmanwahl
A DBX 105.6
R DBX 105.6
JC D19s // DS19 Applikationsdaten
A DBX 105.7
R DBX 105.7
JC D20s // DS20 Teach In
A DBX 40.0
JC D24s // DS24 Achskopplung 1 aktivieren/definieren

JU KEDS // kein Auftrag bzw. DS-Nr.

```

Network: 9	Parameter für Datensätze
------------	--------------------------

DS-NR., DS-Länge und Zeiger auf Datensatz erfassen
--

```

DS2s: S DBX 128.0
L B#16#2 // DS-Nr. =02
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#8 // DS-Länge = 8 Byte
L P#DBX 160.0 // Zeiger auf Anfang des Datensatzes
JU GEMP

DS3s: S DBX 128.1
L B#16#3 // DS-Nr. =03
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#8 // DS-Länge = 8 Byte
L P#DBX 168.0 // Zeiger auf Anfang des Datensatzes
JU GEMP

DS4s: S DBX 128.2
L B#16#4 // DS-Nr. =04
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#4 // DS-Länge = 4 Byte
L P#DBX 156.0 // Zeiger auf Anfang des Datensatzes
JU GEMP

DS7s: S DBX 128.3
L B#16#7 // DS-Nr. =07
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#14 // DS-Länge = 20 Byte
L P#DBX 176.0 // Zeiger auf Anfang des Datensatzes
JU GEMP

```

Page 7 of 24

```

DS9s: S DBX 129.3
L B#16#9 // DS-Nr. =09
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#18 // DS-Länge = 24 Byte
L P#DBX 196.0 // Zeiger auf Anfang des Datensatzes
JU GEMP

DS11: L B#16#B // DS-Nr. =11
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#2 // DS-Länge = 2 Byte
L P#DBX 100.0 // Zeiger auf Anfang des Datensatzes (alt:34.0)
JU GEMP

DS12: L B#16#C // DS-Nr. =12
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#2 // DS-Länge = 2 Byte
L P#DBX 126.0 // Zeiger auf Anfang des Datensatzes (alt:36.0)
JU GEMP

D13s: S DBX 129.1
L B#16#D // DS-Nr. =13
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#4 // DS-Länge = 4 Byte
L P#DBX 140.0 // Zeiger auf Anfang des Datensatzes
JU GEMP

D14s: S DBX 128.7
L B#16#E // DS-Nr. =14
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#4 // DS-Länge = 4 Byte
L P#DBX 144.0 // Zeiger auf Anfang des Datensatzes
JU GEMP

D15s: S DBX 129.0
L B#16#F // DS-Nr. =15
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#4 // DS-Länge = 4 Byte
L P#DBX 148.0 // Zeiger auf Anfang des Datensatzes
JU GEMP

D16s: S DBX 129.4
L B#16#10 // DS-Nr. =16
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#2 // DS-Länge = 2 Byte
L P#DBX 220.0 // Zeiger auf Anfang des Datensatzes
JU GEMP

```

(Anexo M) 23

Page 8 of 24

```

D17s: S   DBX 128.4
      L   B#16#11 // DS-Nr. =17
      T   #DS_nr_wr // DS-Nr. =17 #DS_nr_wr -- DSNr. schreibend
      L   W#16#14 // DS-Länge = 20 Byte
      L   P#DBX 222.0 // Zeiger auf Anfang des Datensatzes
      JU   GEMP

D18s: S   DBX 129.5
      L   B#16#12 // DS-Nr. =18
      T   #DS_nr_wr // DS-Nr. =18 #DS_nr_wr -- DSNr. schreibend
      L   W#16#4 // DS-Länge = 4 Byte
      L   P#DBX 242.0 // Zeiger auf Anfang des Datensatzes
      JU   GEMP

D19s: S   DBX 129.6
      L   B#16#13 // DS-Nr. = 19
      T   #DS_nr_wr // DS-Nr. =19 #DS_nr_wr -- DSNr. schreibend
      L   W#16#4 // DS-Länge = 4 Bytes
      L   P#DBX 246.0 // Zeiger auf Anfang des Datensatzes
      JU   GEMP

D20s: S   DBX 129.7
      L   B#16#14 // DS-Nr. = 20
      T   #DS_nr_wr // DS-Nr. =20 #DS_nr_wr -- DSNr. schreibend
      L   W#16#2 // DS-Länge = 2 Bytes
      L   P#DBX 250.0 // Zeiger auf Anfang des Datensatzes
      JU   GEMP

D22s: S   DBX 128.6
      L   B#16#16 // DS-Nr. = 22
      T   #DS_nr_wr // DS-Nr. =22 #DS_nr_wr -- DSNr. schreibend
      L   W#16#4 // DS-Länge = 4 Bytes
      L   P#DBX 152.0 // Zeiger auf Anfang des Datensatzes
      JU   GEMP

D24s: S   DBX 130.0
      L   B#16#18 // DS-Nr. = 24
      T   #DS_nr_wr // DS-Nr. =24 #DS_nr_wr -- DSNr. schreibend
      L   W#16#2 // DS-Länge = 2 Bytes
      L   P#DBX 252.0 // Zeiger auf Anfang des Datensatzes
      JU   GEMP

KEDS: R   DBX 14.0 // kein Auftrag DIQ löschen
      JU   DUEA

GEMP: ON  DBX 14.0 // DIQ-Steuerbit
      ON  DBX 22.0 // DI-Rückmeldebit
      JC  DUEA // kein SFC Aufruf

```

Page 9 of 24

```

Network: 10 Datensatz für Auftrag schreiben

Aktual-Parameter bilden und SFC WR_REC aufrufen, wenn DIQ = 1 und DI = 1
record Parameter vom Typ any in Lokaldaten DS_any aufbauen
dazu indirekt über AR1 adressieren

LAR1 P##DS_any // Anfangsadresse des ANY-Pointers
T LD [AR1,P#6.0] // Zeiger auf Datensatz
POP
T LW [AR1,P#2.0] // Datensatzlänge aus Lowword

L #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
T DBB 103 // interner Merker : letzter DS
L DBW 8 // + kanalspezifischen DS-Offset
+I
T #DS_nr_wr // DS-Nr. #DS_nr_wr -- DSNr. schreibend
L W#16#1002 // Syntax-ID und Typ
T LW [AR1,P#0.0]
L #dbnr
T LW [AR1,P#4.0] // DB-Nummer

CALL "WR_REC" SFC58 -- Write Data Record
REQ :=TRUE
IOID :=B#16#54
LADDR :=DBW0
RECNUM :=#DS_nr_wr #DS_nr_wr -- DSNr. schreibend
RECORD :=#DS_any // Anfangsadresse des ANY-Pointers #DS_any -- record Parameter des SFC
RET_VAL:=DBW108
BUSY :=#sfc_busy_wr #sfc_busy_wr -- BUSY-Ausgang des SFC58

AN #sfc_busy_wr // DS-Übertragung abgeschlossen? #sfc_busy_wr -- BUSY-Ausgang des SFC58
R DBX 14.0 // DIQ-Steuerbit löschen

// Auftrag beendet bilden
DUEA: A DBX 68.0 // Anstoß = 1, Auftrag läuft
AN DBX 14.0 // DIQ-Steuerbit = 0
AN DBX 22.0 // DI-Rückmeldebit = 0, FM fertig
R DBX 68.0 // Anstoßmerker löschen
R DBX 106.0 // Auftrag läuft löschen
S DBX 106.1 // Auftrag beendet setzen

JCN DALE // DS lesen

```

Network: 11 Auftragsverwaltung

Auftragsbereiche aktualisieren

Einstellungen bleiben erhalten

```

L   B#16#B    // DS11-Einstellungen
L   #DS_nr_wr // aufgerufener DS                #DS_nr_wr    -- DSNr. schreibend
==I
JC  SFCR      // SFC_RET_VAL-Test
L   B#16#C    // DS12-Kommandos
<>I
JC  AvEs      // Auftragsverwaltung Einzelschreibaufträge

// Kommandos
L   DBW 126   // interner Auftragsspeicher
INVI // Maske
L   DBW 36    // Einstellungen und Kommandos
AW
T   DBW 36    // letzten erledigten Auftrag löschen
JU  SFCR     // SFC-RET_VAL-Test

// Einzelschreibaufträge
AvEs: L   DBD 128 // interner Auftragsspeicher
      INVD // Maske
      L   DBD 38  // Schreibaufträge
      AD
      T   DBD 38  // letzten erledigten Auftrag löschen
      L   DBW 38
      T   DBW 138

// SFC_RET < 0 : Fehler (z.B. 80c3); > 0 : dezentral (7001)?
SFCR: L   B#16#0
      L   DBW 108 // RET_VAL des sfc
      <>I
      JC  OURV

// Statusbereiche aktualisieren
// Auftragsverwaltung bei Fehler aktualisieren

L   B#16#B    // DS11-Einstellungen
L   #DS_nr_wr // aufgerufener DS                #DS_nr_wr    -- DSNr. schreibend
==I
JC  StEi      // Status Einstellungen
L   B#16#C    // DS12-Kommandos
==I

```

Page 11 of 24

```

JC  StKo      // Status Kommandos
JU  StEs      // Status Einzelschreibaufträge

// Einstellungen
StEi: R   DBX 106.5 // DS11 beendet
      L   DBW 100 // aktiver Auftrag
      L   DBW 124 // letzter Auftrag
      XOW // Unterschiede erfassen
      L   DBW 44  // letzter Status
      OW
      T   DBW 44  // letzten Auftrag im Status hinzufügen
      L   DBW 100 // Auftrag
      T   DBW 124 // internen Bereich aktualisieren

      AN  DBX 22.4 // Datenfehler
      JC  BA
      L   DBW 124 // letzter Auftrag
      L   DBW 54  // Fehlermeldungen
      OW
      T   DBW 54  // als Fehlermeldungen eintragen
      JU  FeRe // Fehler RET_VAL

// Kommandos
StKo: R   DBX 106.6 // DS12 beendet
      L   DBW 126 // interner Auftragsspeicher
      L   DBW 46  // Status der Auftragsverwaltung
      OW
      T   DBW 46  // letzten erledigten Auftrag im Status hinzufügen

      AN  DBX 22.4 // Datenfehler
      JC  BA
      L   DBW 126 // interner Auftragsspeicher
      L   DBW 56  // Fehlermeldung
      OW
      T   DBW 56  // als Fehlermeldungen eintragen
      JU  FeRe // Fehler RET_VAL

// Einzelschreibaufträge
StEs: R   DBX 106.7 // Einzelauftrag beendet
      L   DBD 128 // interner Auftragsspeicher
      L   DBD 48  // Status der Auftragsverwaltung
      OD
      T   DBD 48  // letzten erledigten Auftrag im Status hinzufügen

      AN  DBX 22.4 // Datenfehler
      JC  BA
      L   DBD 128 // aktiver Auftrag
      L   DBD 58

```

(Anexo M) 25

Page 12 of 24

```

OD
T   DBD  58 // als Fehlermeldungen eintragen
JU   FeRe // Fehler RET_VAL

OURV: L   DBW 108 // RET_VAL des SFC schreiben
      L   B#16#0
      >=I // VKE = 1 im fehlerfreien Fall
      JC   FL // Fehler löschen

      L   B#16#0
      T   DBB 106 // interner Auftrag löschen
      L   DBW 100 // aktiver Auftrag
      ==I
      JC   kAll
      L   DBW 124 // vorletzter Auftrag
      INVI // invertieren
      L   DBW 100 // aktiver Auftrag
      AW // filtern
      T   DBW 54 // als Fehlermeldungen eintragen

kAll: L   DBW 126 // aktiver Auftrag
      L   DBW 56 // Fehlermeldung
      OW
      T   DBW 56 // als Fehlermeldungen eintragen

      L   DBD 128 // aktiver Auftrag
      L   DBD 38 // alle Aufträge
      OD
      L   DBD 58 // vorhandene Fehler
      OD
      T   DBD 58 // als Fehlermeldungen eintragen

      L   B#16#0
      T   DBD 38 // alle Aufträge löschen

FeRe: L   -1
      T   DBW 110 // RET_VAL Funktion Auftrag schreiben
      JU   BA

FL:   L   DBD 128 // aktiver Auftrag
      INVD
      L   DBD 58
      AD
      T   DBD 58 // Fehler löschen

```

Page 13 of 24

Network: 12	FM-Steuersignale bearbeiten
-------------	-----------------------------

```

löschen der Steuersignale START, R-, R+ und alle zur FM schreiben

BA:  A   DBX 23.1 // Bearbeitung läuft und
      AN  DBX 23.0 // Startfreigabe fehlt oder
      O   DBX 22.3 // Bedienfehler steht an
// VKE gesetzt -> Startsignale löschen

      L   DBB 24 // Betriebsartenrückmeldung auswerten
      JL  LIST
      JU  DALE // 0: keine BA
      JU  RFL // 1: Tippen
      JU  RFL // 2: Steuern
      JU  AFL // 3: Referenzpunktfahren
      JU  AFL // 4: Schrittmaß relativ
      JU  DALE // 5: Schrittmaß absolut
      JU  SFL // 6: MDI
      JU  DALE // 7: nicht verwendet
      JU  SFL // 8: Automatik Folgesatz
      JU  SFL // 9: Automatik Einzelsatz
LIST: JU  DALE // 10-255: keine bekannte BA

SFL:  R   DBX 15.0 // START löschen
      O   DBX 15.0 // bei START BL und PEH beeinflussen
      =   #start // Speicher für gesetzten Start
      JU  RUWR

AFL:  R   DBX 15.2 // R- löschen
      R   DBX 15.3 // R+ löschen
      R   DBX 15.0 // START löschen

      O   DBX 15.0 // bei START oder
RFL:  O   DBX 15.2 // R- oder
      O   DBX 15.3 // R+ BL und PEH beeinflussen
      =   #start // Speicher für gesetzten Start

RUWR: O   #start // VKE nach = neu zuweisen // Speicher für gesetzten Start
      O   DBX 23.1 // Bearbeitung läuft = b_start or BL
      =   DBX 13.6 // MODE_BUSY (Bearbeiten)
      AN  #start // Position erreicht = !start and PEH // Speicher für gesetzten Start
      A   DBX 25.7
      =   DBX 13.7 // POS_REACHED (Position)

//***** D A T E N L E S E N *****

```

(Anexo M) 26

Page 14 of 24

Network: 13 Auftragsbearbeitung Leseaufträge

Leseaufträge erfassen, Auftragsbit setzen
Bedien- und Fahrfehler, sowie Datenfehler erfassen

```
DALE: SET // VKE = 1
A DBX 68.3 // Auftragsbearbeitung DS lesen nicht möglich
JC STEN // Steuersignale an FM
A DBX 68.2 // Leseauftrag gesetzt ?
AN DBX 118.3 // Merker für Bedien- und Fahrfehler
AN DBX 118.4 // Merker für Datenfehler
JC LEA
A DBX 118.3 // Merker für Bedien- und Fahrfehler
JC D162 // DS 162 Bedien- und Fahrfehler
A DBX 118.4 // Merker für Datenfehler
JC D163 // DS 163 Datenfehler
A DBX 22.3 // Bedien- und Fahrfehler
FP DBX 118.6 // Flankenmerker
JC D162 // DS 162 Bedien- und Fahrfehler
A DBX 22.4 // Datenfehler
FP DBX 118.7 // Flankenmerker
JC D163 // DS 163 Datenfehler
L B#16#0
L DBW 112 // interner Lesespeicher
<>I
S DBX 68.2 // Leseauftrag aktiv setzen
JC LEA
L DBW 42 // Leseauftragsbereich abfragen
AW W#16#3FF9 // Maske für Auftragsbits
T DBW 42 // löschen
T DBW 112 // interner Lesespeicher
L B#16#0
<>I
S DBX 68.2 // Leseauftrag aktiv setzen
A DBX 68.2 // Leseauftrag gesetzt ?
JC LEA
L B#16#0
T DBW 116 // Schieberegister
T DBW 120 // Fehlercode des letzten Leseauftrages
T DBW 132 // letzter Leseauftrag
JU RETL

// Auftragsablauf
LEA: L B#16#1
L DBW 134 // RET_VAL
==I
```

Page 15 of 24

```
JC Adez // Auftrag läuft dezentral
L B#16#0
L DBW 116
<>I
JC klol
Adez: L DBW 112 // Leseauftrag
T DBW 116 // Schieberegister
```

Network: 14 Leseaufträge selektieren

Datensätze für Leseaufträge selektieren

```
klol: L B#16#0
T DBW 120 // alten SFC-RET_VAL löschen
T DBW 132 // alten Leseauftrag löschen
A DBX 116.0
R DBX 116.0
JC DS26 // DS26 Grundbetriebsdaten
A DBX 116.1
R DBX 116.1
JC DS27 // DS27 aktiver NC-Satz
A DBX 116.2
R DBX 116.2
JC DS28 // DS28 nächster NC-Satz
A DBX 116.3
R DBX 116.3
JC DS31 // DS31 Istwert bei G50 ...
A DBX 116.4
R DBX 116.4
JC DS32 // DS32 Servicedaten
A DBX 116.5
R DBX 116.5
JC D164 // DS164 Betriebsfehler
A DBX 117.0
R DBX 117.0
JC DS37 // DS37 Status Achskopplung
A DBX 117.3
R DBX 117.3
JC DS38 // DS38 Parameteränderung
A DBX 117.4
R DBX 117.4
JC D16L // DS16 Digital I/O
A DBX 117.5
R DBX 117.5
JC DS34 // DS34 Betriebsdaten
A DBX 117.6
```

(Anexo M) 27

Page 16 of 24

```

R   DBX 117.6
JC  DS29      // DS29 Applikationsdaten
A   DBX 117.7
R   DBX 117.7
JC  DS30      // DS30 Meßwerte lesen
JU  D163

```

Network: 15 Parameter für Datensätze zuordnen

DS-Nr., DS-Länge, Zeiger auf Datensatzanfang erfassen

```

D16L: S   DBX 133.4
      L   B#16#10      // DS-Nr. =16
      T   #DS_nr_rd          #DS_nr_rd      -- DSNr. lesend
      L   W#16#2        // DS-Länge = 2 Byte
      L   P#DBX 220.0    // Zeiger auf Anfang des Datensatzes
      JU  GEML

DS26: S   DBX 132.0
      L   B#16#1A      // DS-Nr. =26
      T   #DS_nr_rd          #DS_nr_rd      -- DSNr. lesend
      L   W#16#20      // DS-Länge = 32 Byte
      L   P#DBX 310.0    // Zeiger auf Anfang des Datensatzes
      JU  GEML

DS27: S   DBX 132.1
      L   B#16#1B      // DS-Nr. =27
      T   #DS_nr_rd          #DS_nr_rd      -- DSNr. lesend
      L   W#16#14      // DS-Länge = 20 Byte
      L   P#DBX 342.0    // Zeiger auf Anfang des Datensatzes
      JU  GEML

DS28: S   DBX 132.2
      L   B#16#1C      // DS-Nr. =28
      T   #DS_nr_rd          #DS_nr_rd      -- DSNr. lesend
      L   W#16#14      // DS-Länge = 20 Byte
      L   P#DBX 362.0    // Zeiger auf Anfang des Datensatzes
      JU  GEML

DS29: S   DBX 133.6
      L   B#16#1D      // DS-Nr. =29
      T   #DS_nr_rd          #DS_nr_rd      -- DSNr. lesend
      L   W#16#10      // DS-Länge = 16 Byte
      L   P#DBX 382.0    // Zeiger auf Anfang des Datensatzes
      JU  GEML

```

Page 17 of 24

```

DS30: S   DBX 133.7
      L   B#16#1E      // DS-Nr. =30
      T   #DS_nr_rd          #DS_nr_rd      -- DSNr. lesend
      L   W#16#C        // DS-Länge = 12 Byte
      L   P#DBX 486.0    // Zeiger auf Anfang des Datensatzes
      JU  GEML

DS31: S   DBX 132.3
      L   B#16#1F      // DS-Nr. =31
      T   #DS_nr_rd          #DS_nr_rd      -- DSNr. lesend
      L   W#16#4        // DS-Länge = 4 Byte
      L   P#DBX 398.0    // Zeiger auf Anfang des Datensatzes
      JU  GEML

DS32: S   DBX 132.4
      L   B#16#20      // DS-Nr. =32
      T   #DS_nr_rd          #DS_nr_rd      -- DSNr. lesend
      L   W#16#20      // DS-Länge = 32 Byte
      L   P#DBX 402.0    // Zeiger auf Anfang des Datensatzes
      JU  GEML

DS34: S   DBX 133.5
      L   B#16#22      // DS-Nr. =34
      T   #DS_nr_rd          #DS_nr_rd      -- DSNr. lesend
      L   W#16#C        // DS-Länge = 12 Byte
      L   P#DBX 434.0    // Zeiger auf Anfang des Datensatzes
      JU  GEML

DS37: S   DBX 133.0
      L   B#16#25      // DS-Nr. =37
      T   #DS_nr_rd          #DS_nr_rd      -- DSNr. lesend
      L   W#16#2        // DS-Länge = 2 Byte
      L   P#DBX 470.0    // Zeiger auf Anfang des Datensatzes
      JU  GEML

DS38: S   DBX 133.3
      L   B#16#26      // DS-Nr. =38
      T   #DS_nr_rd          #DS_nr_rd      -- DSNr. lesend
      L   W#16#18      // DS-Länge = 24 Byte
      L   P#DBX 446.0    // Zeiger auf Anfang des Datensatzes
      JU  GEML

D162: L   B#16#0
      T   DBW 66        // Fehlercode loeschen
      T   DBW 132       // alten Leseauftrag
      S   DBX 68.2      // Auftrag aktiv
      S   DBX 118.3     // Merker für Bedien- und Fahrfehler
      S   DBX 132.6     // neuer Leseauftrag

```

(Anexo M) 28

Page 18 of 24

```

L   B#16#1
T   DBW 122 // RET_VAL aktiv setzen
L   B#16#A2 // DS-Nr. =162
T   #DS_nr_rd // DS-Nr. // #DS_nr_rd -- DSNr. lesend
L   W#16#2 // DS-Länge = 2 Byte
L   P#DBX 90.0 // Zeiger auf Anfang des Datensatzes
JU  GEML

D163: L   B#16#0
T   DBW 66 // Fehlercode loeschen
T   DBW 132 // alten Leseauftrag
S   DBX 68.2 // Auftrag aktiv
S   DBX 118.4 // Merker für Datenfehler
S   DBX 132.7 // neuer Leseauftrag
L   B#16#1
T   DBW 122 // RET_VAL aktiv setzen
L   B#16#A3 // DS-Nr. =163
T   #DS_nr_rd // DS-Nr. // #DS_nr_rd -- DSNr. lesend
L   W#16#2 // DS-Länge = 2 Byte
L   P#DBX 94.0 // Zeiger auf Anfang des Datensatzes
JU  GEML

D164: S   DBX 132.5
L   B#16#A4 // DS-Nr. =164
T   #DS_nr_rd // DS-Nr. // #DS_nr_rd -- DSNr. lesend
L   W#16#2 // DS-Länge = 2 Byte
L   P#DBX 86.0 // Zeiger auf Anfang des Datensatzes

```

Network: 16 Datensatz Auftrag lesen ausführen

record Parameter vom Typ any in Lokaldaten DS_any aufbauen
dazu indirekt über AR1 adressieren

```

GEML: LAR1 P##DS_any // Anfangsadresse des ANY-Pointers
T   LD [AR1,P#6.0] // Zeiger auf Datensatz
POP
T   LW [AR1,P#2.0] // Datensatzlänge aus Lowword
L   #DS_nr_rd // DS-Nr. // #DS_nr_rd -- DSNr. lesend
T   DBB 115 // interner DS-Merker
O   DBX 132.6 // DS162
O   DBX 132.7 // DS163
O   DBX 132.5 // DS164
JCN ksDS // entspricht nicht diesen DS

```

Page 19 of 24

```

L   DBW 2 // Kanalnummer
DEC 1 // CH_NO - 1
L   B#16#23 // DS_OFFS = chno * 35
*I
L   #DS_nr_rd // DS-Nr. // #DS_nr_rd -- DSNr. lesend
JU sDS // entspricht den DS162,163,164
ksDS: L   DBW 8 // + kanalspezifischen DS-Offset
sDS: +I
T   #DS_nr_rd // DS-Nr. // #DS_nr_rd -- DSNr. lesend
L   W#16#1002 // Syntax-ID und Typ
T   LW [AR1,P#0.0]
L   #dbnr // DB-Nr. // #dbnr -- DB-Nr. Kanal-DB
T   LW [AR1,P#4.0] // DB-Nummer

// DS von FM lesen
CALL "RD_REC" // SFC59 -- Read a Data Record
REQ :=TRUE
I OID :=B#16#54
LADDR :=DBW0
RECNUM :=#DS_nr_rd // #DS_nr_rd -- DSNr. lesend
RET_VAL:=DBW120
BUSY :=#sfc_busy_rd // #sfc_busy_rd -- BUSY-Ausgang des SFC59
RECORD :=#DS_any // Anfangsadresse des ANY-Pointers // #DS_any -- record Parameter des SFC
AN #sfc_busy_rd // DS-Übertragung abgeschlossen? // #sfc_busy_rd -- BUSY-Ausgang des SFC59
JCN STEN // nein

```

Network: 17 Auftragsverwaltung aktualisieren

Auftrag aktualisieren

```

L   DBW 132 // interner Auftragsmerker
INVI // Maske
L   DBW 112 // Auftrag
AW
T   DBW 112 // letzten Auftrag löschen
L   DBW 132 // interner Auftragsmerker
INVI // Maske
L   DBW 42 // Auftrag
AW
T   DBW 42 // letzten Auftrag löschen

// Status aktualisieren
L   DBW 120 // RET_VAL Auftrag lesen
L   B#16#0
<>I

```



```
JC  LABC      // Leseauftrag beenden
L   DBW 132  // interner Auftragspeicher
L   DBW 52   // Status der Auftragsverwaltung
OD
T   DBW 52   // letzten erledigten Auftrag hinzufügen
```

```
// Auftrag beendet bilden
```

```
LABC: A   DBX 68.2 // Anstoß = 1, Auftrag läuft
R       DBX 68.2 // Anstoßmerker löschen
R       DBX 118.0 // Auftrag läuft löschen
S       DBX 118.1 // Auftrag beendet setzen
R       DBX 118.3 // DS162-Auftrag löschen
R       DBX 118.4 // DS163-Auftrag löschen
```

```
Network: 18 Fehlerbehandlung Leseaufträge
```

```
Auftragsverwaltung aktualisieren
```

```
RETL: L   DBW 120 // RET_VAL Auftrag lesen
L       B#16#0
>=I
JC      STEN

R       DBX 68.2 // Auftragsbearbeitung aktiv löschen
L       B#16#0
T       DBB 118 // interner Auftrag löschen

L       DBW 132 // Leseauftrag
L       DBW 62  // Fehlermeldungen
OW
T       DBW 62  // letzte Fehlermeldung dazufügen

L       -2
T       DBW 122 // RET_VAL des FC1
```

Page 21 of 24

```
Network: 19 Steuersignale an FM
```

```
Peripherieausgabe der Steuersignale, FC-RET_VAL
```

```
STEN: LAR1 DBD 4 // Kanaladresse in AR1 laden
L       DBD 14 // aus Kanal-DB
T       PQD [AR1,P#0.0] // Steuersignale Byte 0-3 zur FM schreiben
L       DBD 18
T       PQD [AR1,P#4.0] // Steuersignale Byte 4-7 zur FM schreiben
```

```
// Auftragsstatus, RET-VAL, Fehlercode und BI ausgeben
```

```
//=====
```

```
A   DBX 68.0 // Auftrag schreiben aktiv
JC  Sakt
A   DBX 68.2 // Auftrag lesen aktiv
JC  Lakt
JU  SLna // "Schreiben" und "Lesen" nicht aktiv, beides auswerten
```

```
// "Schreiben" und "Lesen" aktiv ?
```

```
Sakt: A   DBX 68.2 // Auftrag lesen aktiv
JC      SLak // "Schreiben" und "Lesen" aktiv
```

```
// "Schreiben" aktiv; "Lesen" nicht aktiv
```

```
//-----
```

```
L   B#16#0
L   DBW 120 // Fehlercode Leseauftrag
>I
JC  ninL // nicht 0
L   DBW 122 // RET_VAL: Auftrag lesen
<>I
JC  ErrL
L   B#16#0
L   DBW 108 // Fehlercode Schreibauftrag
<=I
JC  kfe // nicht 0
T   DBW 66 // letzter Fehlercode
kfe: L   B#16#1 // "Schreiben" aktiv; "Lesen" nicht aktiv
JU  RESL
ninL: T   DBW 66 // letzter Fehlercode
ErrL: L   B#16#0
L   DBW 120
L   DBW 122 // RET_VAL: Auftrag lesen
JU  RESL
```

```
// "Lesen" aktiv, "Schreiben" nicht aktiv
```

(Anexo M) 30

Page 22 of 24

```

//-----
Lakt: L   B#16#0
      L   DBW 108      // Fehlercode Schreibauftrag
      >I
      JC   ninS        // nicht 0
      L   DBW 110      // RET_VAL: Auftrag schreiben
      <>I
      JC   ErrS
      L   B#16#1        // "Lesen" aktiv; "Schreiben" nicht aktiv;
      JU   RESL
ninS: T   DBW 66        // letzter Fehlercode
ErrS: L   B#16#0
      T   DBW 108
      L   DBW 110      // RET_VAL: Auftrag schreiben
      JU   RESL

// "Lesen" aktiv; "Schreiben" aktiv
//-----
SLak: L   B#16#1        // "Schreiben" und "Lesen" aktiv
      JU   RESL

// "Schreiben" und "Lesen" nicht aktiv
// "Schreiben" und "Lesen" auswerten
//-----
SLna: L   B#16#0
      L   DBW 108      // Fehlercode Schreibauftrag
      >I
      JC   ninu
      L   B#16#0
      L   DBW 120      // Fehlercode Leseauftrag
      >I
      JC   ninL
      L   B#16#0
      JU   RESL

// "Schreiben" und "Lesen" nicht 0
ninu: L   B#16#0
      L   DBW 120      // Fehlercode Leseauftrag
      >I
      JC   niSL
      L   DBW 108      // Fehlercode Schreibauftrag
      JU   ninS
niSL: L   DBW 110      // RET_VAL: Auftrag schreiben
      L   DBW 122      // RET_VAL: Auftrag lesen
      +I
RESL: T   DBW 134      // in internen Belegung schreiben
      T   #RET_VAL      // RET_VAL: "Baustein FC 1"          #RET_VAL
      SET

```

Page 23 of 24

```

SAVE          // BIE = 1 setzen
L   B#16#0
>I
JC   RV1
==I
JC   RV1o      // Fehlermerker nicht löschen
CLR
SAVE          // BIE = 0 setzen
RV1o: L   B#16#0
      T   DBW 110      // RET_VAL: Auftrag schreiben
      T   DBW 122      // RET_VAL: Auftrag lesen

RV1: LAR1 #rettar1      // AR1 restaurieren          #rettar1          -- Zwischenspeicher für AR1

BE

```

FC2 - <offline>

"POS_DIAG" Diagnostic Interrupt Information
Name: POS_DIAG **Family:** FM_ST_SV
Author: AUT **Version:** 2.0
Block version: 2
Time stamp Code: 02/24/2000 10:50:29 AM
Interface: 02/24/2000 10:50:29 AM
Lengths (block/logic/data): 00316 00186 00046

Name	Data Type	Address	Comment
IN		0.0	
DB_NO	Int	0.0	datablock number
OUT		0.0	
IN_OUT		0.0	
IN_DIAG	Bool	4.0	initiate diagnostic interrupt
TEMP		0.0	
rettar1	DWord	0.0	Zwischenspeicher für AR1
dbnr	Int	4.0	DB-Nr. Kanal-DB
DS_any	Any	6.0	ANY-Pointer
szl_header	Struct	16.0	SZL-Header für SFC51
len	Word	16.0	
n_dr	Word	18.0	
diag	Array [1..128] Of Bool	20.0	
sfc_busy	Bool	36.0	BUSY-Ausgang des SFCs
RETURN		0.0	
RET_VAL	Int	2.0	

Block: FC2 DIAGNOSTIC INTERRUPT INFORMATION

Page 1 of 3

Network: 1 DB aufschlagen und Diagnosedaten lesen

```

SET
SAVE          // BIE = 1 setzen

TAR1 #rettar1 // AR1 retten                #rettar1      -- Zwischenspeicher für AR1

L   #DB_NO    // Kanal-DB indirekt öffnen #DB_NO       -- datablock number
T   #dbnr     //                               #dbnr        -- DB-Nr. Kanal-DB
OPN DB [#dbnr] //                               #dbnr        -- DB-Nr. Kanal-DB

```

Network: 2 Auftragsbearbeitung

```

AN   #IN_DIAG                #IN_DIAG      -- initiate diagnostic interrupt
JC   RET0

//
// SFC RDSYSST Parameter bilden und aufrufen, wenn IN_DIAG = 1
//
LAR1 P##DS_any              // Anfangsadresse des ANY-Pointers
// 16 Byte in Lokaldaten lesen
L   W#16#1002              // Syntax-ID und Typ
T   LW [AR1,P#0.0]
L   B#16#10                // Länge = 16 Byte
T   LW [AR1,P#2.0]
L   #dbnr                  //                               #dbnr        -- DB-Nr. Kanal-DB
T   LW [AR1,P#4.0]        // Teilbereich 0
L   P#DBX 70.0
T   LD [AR1,P#6.0]        // Zeiger auf Anfang der Lokaldaten diag

//
CALL "RDSYSST"              // Diagnosedaten von FM lesen                SFC51
Partial List                -- Read a System Status List or

REQ      :=TRUE
SZL_ID   :=W#16#B3
INDEX   :=DBW0
RET_VAL  :=DBW96
BUSY     :=#sfc_busy        #sfc_busy     -- BUSY-Ausgang des SFCs
SZL_HEADER:=#szl_header    #szl_header   -- SZL-Header für SFC51
DR       :=#DS_any         #DS_any       -- ANY-Pointer

```

(Anexo M) 32

Page 2 of 3

```

AN  #sfc_busy      // DS-Übertragung abgeschlossen      #sfc_busy      -- BUSY-Ausgang des SFCs
R  #IN_DIAG       // --> Anstoßparameter rücksetzen      #IN_DIAG       -- initiate diagnostic interrupt
//
L  DBW 96         // Der Returnwert des SFCs
L  B#16#0        // BIE bilden
>=I // BIE = 1 im fehlerfreien Fall
SAVE // BIE = 0 im Fehlerfall
JCN ERR
RET0: L B#16#0    // RET_VAL = 0, kein Fehler
JU  RETV
ERR:  L -1       // RET_VAL = -1, Fehler
RETV: T #RET_VAL // Ausgangspar. RET_VAL durchgereicht.      #RET_VAL
//
LAR1 #rettar1    // AR1 restaurieren      #rettar1       -- Zwischenspeicher für AR1
//
BE

```

FC3 - <offline>

```

"POS_MSRM"      Measurement Values
Name: POS_MSRM   Family: FM_ST_SV
Author: AUT      Version: 2.0
                Block version: 2
Time stamp Code: 02/24/2000 10:49:05 AM
Interface:       02/24/2000 10:49:05 AM
Lengths (block/logic/data): 00292 00176 00020

```

Name	Data Type	Address	Comment
IN		0.0	
DB_NO	Int	0.0	datablock number
OUT		0.0	
IN_OUT		0.0	
IN_MSR	Bool	4.0	initiate measurement
TEMP		0.0	
rettar1	DWord	0.0	Zwischenspeicher für AR1
dbnr	Int	4.0	DB-Nr. Kanal-DB
DS_any	Any	6.0	ANY-Pointer
DS_nr	Byte	16.0	Datensatz-Nummer
busy	Bool	17.0	BUSY-Ausgang des SFCs
RETURN		0.0	
RET_VAL	Int	2.0	

Block: FC3 MEASUREMENT VALUES

Network: 1	DB aufschlagen
------------	----------------

```

SET
SAVE          // BIE = 1 setzen

TAR1 #rettar1 // AR1 retten                                #rettar1      -- Zwischenspeicher für AR1

L   #DB_NO    // Kanal-DB indirekt öffnen                #DB_NO        -- datablock number
T   #dbnr     //                                         #dbnr         -- DB-Nr. Kanal-DB
OPN DB [#dbnr] //                                         #dbnr         -- DB-Nr. Kanal-DB

```

Network: 2	Auftragsbearbeitung
------------	---------------------

```

AN   #IN_MSR          #IN_MSR      -- initiate measurement
JC   RET0

//
// SFC RD_REC aufrufen, wenn IN_MSR = 1
//
L   B#16#1E          // DS-Nr. = 30
L   DBW 8            // + kanalspezifischen DS-Offset
+I
T   #DS_nr           #DS_nr        -- Datensatz-Nummer

//
// LAR1 P##DS_any    // Anfangsadresse des ANY-Pointers
//
L   W#16#1002        // Syntax-ID und Typ
T   LW [AR1,P#0.0]  //
L   B#16#C           // DS-Länge = 12 Byte
T   LW [AR1,P#2.0]  //
L   DBNO
T   LW [AR1,P#4.0]  // DB-Nummer
L   P#DBX 486.0
T   LD [AR1,P#6.0]  // Zeiger auf Anfang des Datensatzes

//
CALL "RD_REC"        // DS Meßwerte von FM lesen          SFC59          -- Read a Data Record
REQ   :=TRUE
IOID  :=B#16#54
LADDR :=DBW0
RECNM :=#DS_nr      #DS_nr        -- Datensatz-Nummer
RET_VAL:=DBW98
BUSY  :=#busy       #busy         -- BUSY-Ausgang des SFCs
RECORD :=#DS_any    // Anfangsadresse des ANY-Pointers    #DS_any       -- ANY-Pointer

```

Page 2 of 3

```

AN   #busy           #busy         -- BUSY-Ausgang des SFCs
//
R   #IN_MSR          // DS-Übertragung abgeschlossen    #IN_MSR        -- initiate measurement
L   DBW 98           // --> Anstoßparameter rücksetzen

//
L   B#16#0           // Ausgangspar. RET_VAL durchgereicht.
>=I // BIE bilden
SAVE // BIE = 1 im fehlerfreien Fall
JCN  ERR             // BIE = 0 im Fehlerfall
RET0: L B#16#0
JU   RETV            // RET_VAL = 0, kein Fehler
ERR: L -1
RETV: T #RET_VAL     // RET_VAL = -1, Fehler          #RET_VAL
LAR1 #rettar1       // Ausgangspar. RET_VAL durchgereicht. #rettar1      -- Zwischenspeicher für AR1

//
BE   // AR1 restaurieren
//

```

FC4 - <offline>

"VOLT" Determinação do valor de potencia maxima e ângulos dos eixos (modo research)

Name: VOLT **Family:**
Author: EACF **Version:** 0.1
Block version: 2
Time stamp Code: 07/18/2012 05:15:26 PM
Interface: 01/26/2011 11:09:56 PM
Lengths (block/logic/data): 00536 00406 00002

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC4 VOLT

Determina o valor de potencia maxima, no modo research;
 Indica o valor do angulo de cada eixo.

Network: 1 Pos máx. pot.

Guarda o valor da posição angular para o maior valor de tensão eléctrica recebido

```

A      "mem_research"          M248.0
A(
O      "DBEX_100_M1".WORKING   DB100.DBX10.1    -- Machining in progress
O      "DBEX_200_M2".WORKING_M2 DB200.DBX10.1    -- Machining in progress
)
=      L      0.0
A(
A      L      0.0
A(
L      "volt_sol"              PIW288            -- Entrada analógica: sinal da célula FV

```

Page 1 of 5

```

L      "mw_volt"              MW100
>=I
)
L      S5T#10MS
SD      T      20
NOP      0
NOP      0
NOP      0
A      T      20
)
AN      T      21
JNB      _001
L      "volt_sol"              PIW288            -- Entrada analógica: sinal da célula FV
T      "mw_volt"              MW100
SET
SAVE
CLR
_001: A      BR
=      L      0.1
A      L      0.1
A(
L      "Counter_value"        MD44
L      L#0
<D
)
JNB      _002
L      "Counter_value"        MD44
L      L#36000
+D
T      "Counter_vl11"         MD162
L      "Counter_value"        MD44
T      "Counter_vl1"         MD158
_002: NOP      0
A      L      0.1
A(
L      "Counter_value"        MD44
L      L#0
>=D
)
JNB      _003
L      "Counter_value"        MD44
T      "Counter_vl11"         MD162
L      "Counter_value"        MD44
T      "Counter_vl1"         MD158
_003: NOP      0
A      L      0.1
A(
L      "Counter_value_M2"     MD48

```

(Anexo M) 35

Page 2 of 5

```

L      L#0
<D
)
JNB   _004
L      "Counter_value_M2"      MD48
L      L#36000
+D
T      "Counter_v222"          MD170
L      "Counter_value_M2"      MD48
T      "Counter_v222"          MD166
_004: NOP 0
A      L      0.1
A(
L      "Counter_value_M2"      MD48
L      L#0
>=D
)
JNB   _005
L      "Counter_value_M2"      MD48
T      "Counter_v222"          MD170
L      "Counter_value_M2"      MD48
T      "Counter_v222"          MD166
_005: NOP 0
A      L      0.0
A(
L      "volt_sol"              PIW288      -- Entrada analógica: sinal da célula FV
L      "mw_volt"               MW100
==I
)
L      S5T#2S500MS
SD     T      21
NOP    0
NOP    0
NOP    0
NOP    0
A      L      0.0
A(
L      "volt_sol"              PIW288      -- Entrada analógica: sinal da célula FV
L      "mw_volt"               MW100
<>I
)
R      T      21

```

Page 3 of 5

Network: 2	Máx Potência produzida pela célula FV
------------	---------------------------------------

```

L      "mw_volt"      MW100
// Integer (16 Bit) to Double Integer (32-Bit)
ITD
// Double Integer (32-Bit) to Floating-Point (32-Bit IEEE 754)
DTR
L      2.764800e+004
/R
L      1.000000e+001
*R
T      "pot2"      MD39

```

Network: 3

```

L      "mw_volt"      MW100
ITD
DTR
L      2.764800e+004
/R
L      1.000000e+001
*R
T      "pot3"      MD52

```

Network: 4

```

L      "pot2"      MD39
L      "pot3"      MD52
*R
L      1.000000e-001
/R
T      "pot"      MD35

```

Network: 5

```

L   "volt_sol"   PIW288      -- Entrada analógica: sinal da célula FV
ITD
DTR
L   2.764800e+004
/R
L   1.000000e+001
*R
T   "potact1"    MD56

```

Network: 6

```

L   "volt_sol"   PIW288      -- Entrada analógica: sinal da célula FV
ITD
DTR
L   2.764800e+004
/R
L   1.000000e+001
*R
T   "potact2"    MD61

```

Network: 7

```

L   "potact1"    MD56
L   "potact2"    MD61
*R
L   1.000000e-001
/R
T   "potact3"    MD65

```

Page 5 of 5

FC5 - <offline>

"CLEAR" Limpa a potência máxima e respectivos ângulos. Reset aos DBX's do modo MDI

```

Name: CLEAR      Family:
Author: EACF     Version: 0.1
                Block version: 2
Time stamp Code: 07/24/2012 07:02:05 PM
                Interface: 02/24/2011 11:56:53 PM
Lengths (block/logic/data): 00270 00166 00002

```

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

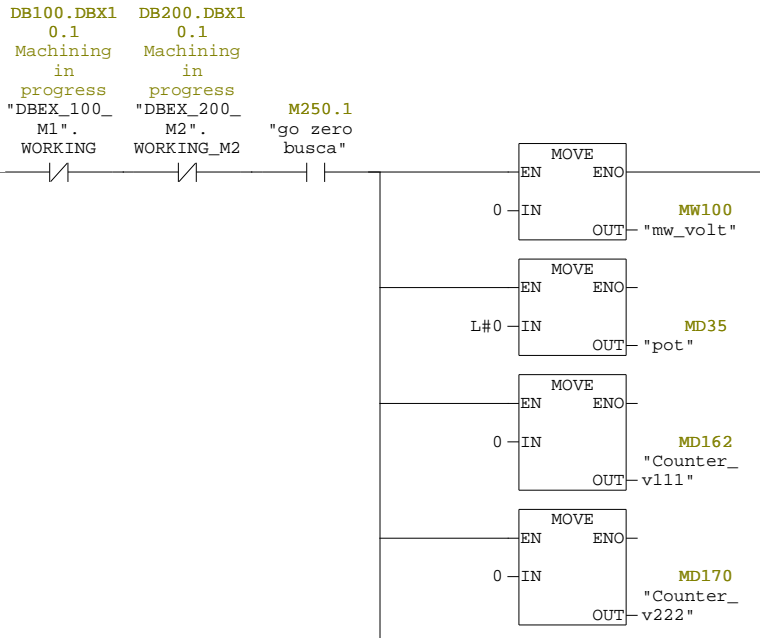
Block: FC5 CLEAR

Limpa o valor de potência eléctrica e da posição dos eixos motorizados, quando se incia um novo varrimento solar.

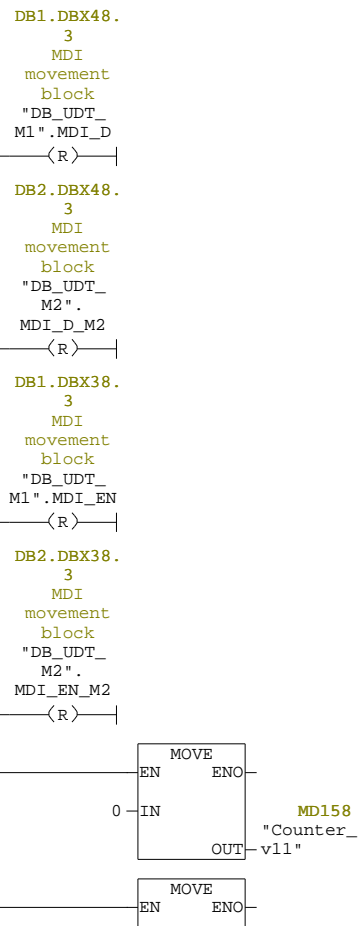
Network: 1 Limpa o bloco de movimento MDI anterior

Se os eixos motorizados não estiverem em funcionamento (DB100.DBX10.1="0" e DB200.DBX10.1="0") e o operador iniciar um novo processo de varrimento solar (M250.1="1"):

- Limpa o maior valor de tensão eléctrica registado no varrimento anterior;
- Limpa o maior valor de potência eléctrica registado no varrimento anterior;
- Reset ("0") aos DBX's responsáveis pela transferência de dados entre os DB's de interface e os módulos FM, no caso do modo de funcionamento associado à deslocação para a posição de máxima potência registada. MDI, FC102 (motor 1) e FC202 (motor 2).

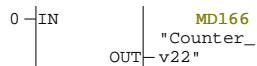


Page 2 of 4



(Anexo M) 38

Page 3 of 4

**FC6 - <offline>**

"AUTO" Transição entre o modo de varrimento e o o modo de posicionamento (com F5)

Name: AUTO **Family:**
Author: EACF **Version:** 0.1
Block version: 2
Time stamp Code: 08/12/2012 03:55:57 PM
Interface: 02/28/2011 06:04:38 PM
Lengths (block/logic/data): 00636 00514 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC6 Modo Auto

Antes de colocar o sistema em modo de varrimento automático há que executar um varrimento manual F13 (HMI) em FC103 (motor 1) e FC203 (motor 2) e deslocar o sistema para a posição de máxima potência registada (F9 - HMI), atribuir um valor para o intervalo de tempo entre varrimentos e colocar em modo automático (F5 = "1").

FC102 (motor 1) e FC 202 (motor): modo de funcionamento que desloca o sistema para uma determinada posição (eixo 1 e eixo 2) com velocidades pré-definidas.

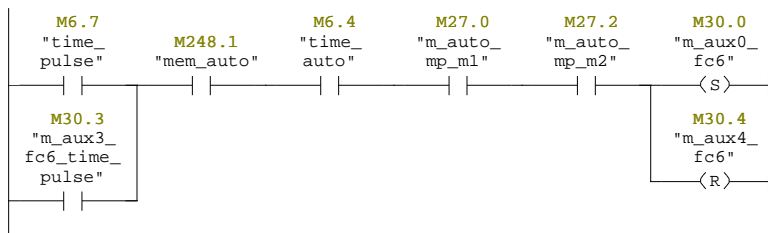
A mesma FC está preparada para funcionamento em modo manual (1) ou varrimento automático (2).

(1) Após realizar um varrimento (F13 - HMI), deslocar o sistema para o posição de máxima potência registada (F9 - HMI, M245.0). Para voltar à posição inicial pressionar F10 - HMI, M245.1;

(2) Após efectuar o processo (1), indicar um tempo de pausa entre varrimentos no menu F2 e pressionar F5 (HMI) para deixar o sistema no modo de varrimento solar automático

Network: 1

Cada varrimento inicia-se da posição inicial (eixos a 0°);
 M27.0="1" (motor 1) M27.2="1" (motor 2) para o sistema se deslocar para a posição de máx. potência;
 M27.1="1" (motor 1) M27.3="1" (motor 2) para o sistema voltar à posição inicial após passar o intervalo de tempo entre varrimentos, limpar os valores de posição e tensão eléctrica registados.
 M30.0="1" para iniciar automaticamente, mesmo sem impulso F5 (time_pulse).



Page 2 of 13

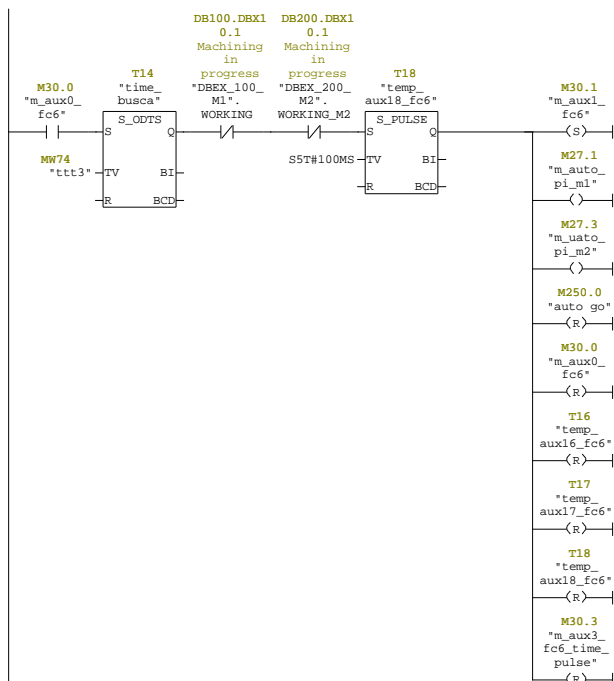
SIMATIC

SAT\SIMATIC 300 Station\CPU 315-2 DP\...\FC6 - <offline>

08/12/2012 03:56:28 PM

Network: 2

Carrega o temporizador com o tempo de intervalo entre varrimentos;
 Se o sistema estiver parado, faz o reset dos temporizadores e memórias auxiliares; faz "=" das memórias responsáveis por colocar o sistema na posição inicial, antes de iniciar um novo varrimento. Não esquecer que o sistema está na posição de máx. pot. quando se pressiona F5 (Modo de varrimento automático). Como tal, faz-se reset de "auto_go" (o set é feito com F9).

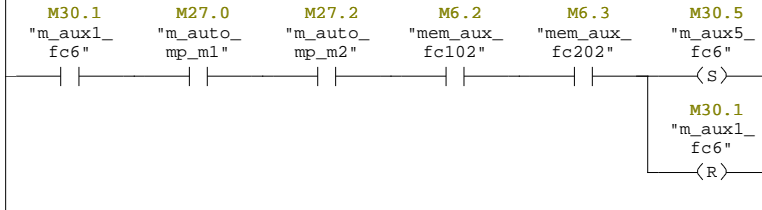


(Anexo M) 40

Page 3 of 13

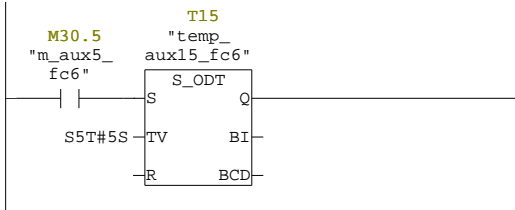
Network: 3

M6.2="1" (motor 1) e M6.3="1" (motor 2), implica que em FC102 (motor 1) e FC202 (motor 2) o sistema recebe ordem para ir para a posição inicial. Tendo pos reached, quer dizer que chegou a zero.
Se vai para "research" tem de fazer reset de T15, para não estar sempre a ir para esse modo.



Network: 4

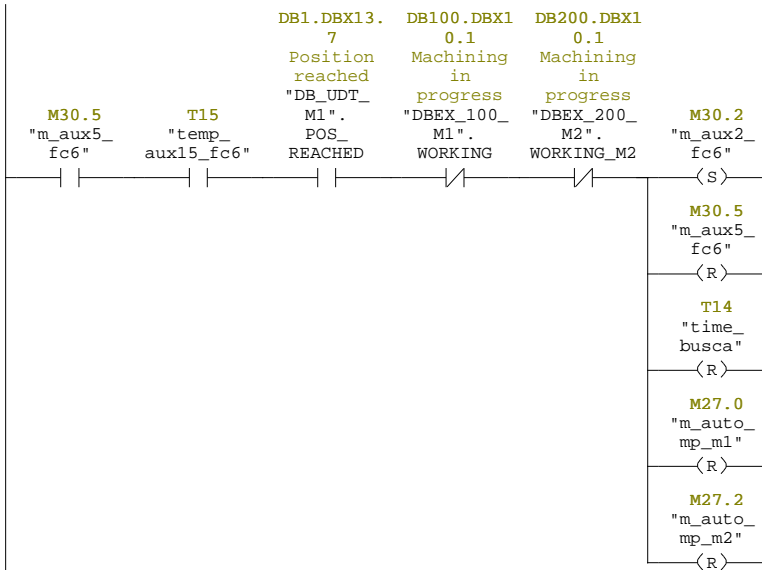
O sistema está a deslocar-se para a posição inicial. Arranca-se um temporizador S_ODT como uma pausa de 5s, de forma a garantir que as próximas indicações sobre o sistema só são atribuídas depois sair da posição onde se encontrava.



Page 4 of 13

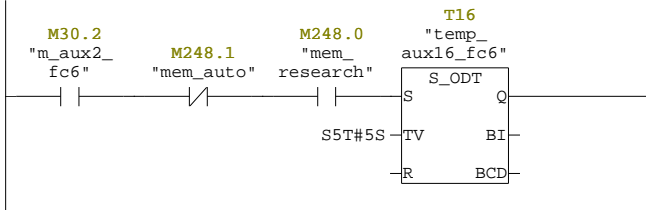
Network: 5

O sistema sai do modo automático FC102 (motor 1) e FC202 (motor 2) e vai para o modo de varrimento FC103 (motor 1) FC203 (motor 2)



Network: 6

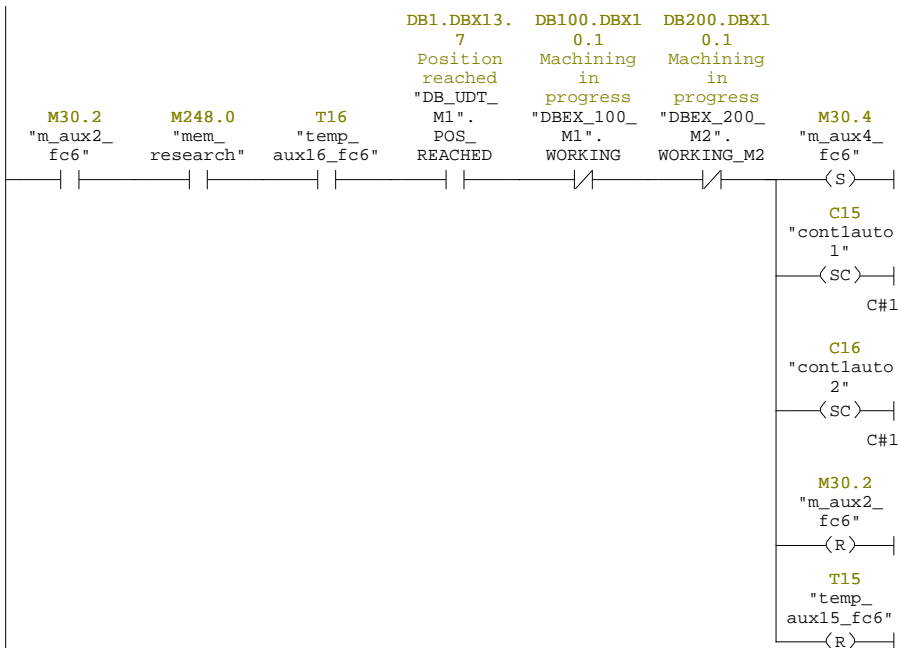
Estado o sistema no modo de varrimento, arranca um temporizador S_ODT com 5s, para que as próximas indicações só se tornem efectivas depois deste sair da posição em que se encontrava. Trabalhos futuros: criar condição com memory bits



Page 6 of 13

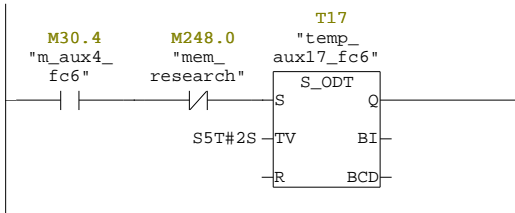
Network: 7

Depois de finalizado o varrimento faz-se o set dos contadores auxiliares, utilizados nas FCs 102 e 202.



Network: 8

Pausa criada para evitar a vibração/escorregamento dos motores, na transição entre modos de funcionamento



Network: 9 Retirado de FC102

Código para o motor 1 se deslocar para a posição de máxima potência registada, depois de ter executado um varrimento em modo automático

A	"m_aux4_fc6"	M30.4	
AN	"mem_research"	M248.0	
A	"temp_aux17_fc6"	T17	
R	"DB_UDT_M1".MDIB.G_2_EN	DB1.DBX178.1	-- enable bit for G function
		group 2	
R	"DB_UDT_M1".MDIB.M_1_EN	DB1.DBX179.1	-- enable bit for M function
		group 1	
R	"DB_UDT_M1".MDIB.M_2_EN	DB1.DBX179.2	-- enable bit for M function
		group 2	
R	"DB_UDT_M1".MDIB.M_3_EN	DB1.DBX179.3	-- enable bit for M function
		group 3	
L	B#16#0		
T	"DB_UDT_M1".MDIB.G_2_VAL	DB1.DBB181	-- value G function group 2
T	"DB_UDT_M1".MDIB.M_1_VAL	DB1.DBB192	-- value M function group 1
T	"DB_UDT_M1".MDIB.M_2_VAL	DB1.DBB193	-- value M function group 2
T	"DB_UDT_M1".MDIB.M_3_VAL	DB1.DBB194	-- value M function group 3
R	"memlautol"	M247.2	
S	"DB_UDT_M1".MDIB.G_1_EN	DB1.DBX178.0	-- enable bit for G function
		group 1	
L	90		
T	"DB_UDT_M1".MDIB.G_1_VAL	DB1.DBB180	-- value G function group 1

Page 8 of 13

S	"DB_UDT_M1".MDIB.X_T_EN	DB1.DBX178.4	-- enable bit for position /
		dwelt time	
L	"Counter_v111"	MD162	
T	"DB_UDT_M1".MDIB.X_T_VAL	DB1.DBD184	-- value position / dwell ti
		me	
S	"DB_UDT_M1".MDIB.V_EN	DB1.DBX179.0	-- enable bit for velocity
L	L#25000		//EXEMPLO 30000 CORRESPONDE A 300 DE GRAUS P MINUTO
T	"DB_UDT_M1".MDIB.V_VAL	DB1.DBD188	-- value velocity
S	"DB_UDT_M1".MDI_EN	DB1.DBX38.3	-- MDI movement block

Network: 10 Retirado de FC202

Código para o motor 2 se deslocar para a posição de máxima potência registada, depois de ter executado um varrimento em modo automático

A	"m_aux4_fc6"	M30.4	
AN	"mem_research"	M248.0	
A	"temp_aux17_fc6"	T17	
R	"DB_UDT_M2".MDIB.M2.G_2_EN_M2	DB2.DBX178.1	-- enable bit for G funct
		ion group 2	
R	"DB_UDT_M2".MDIB.M2.M_1_EN_M2	DB2.DBX179.1	-- enable bit for M funct
		ion group 1	
R	"DB_UDT_M2".MDIB.M2.M_2_EN_M2	DB2.DBX179.2	-- enable bit for M funct
		ion group 2	
R	"DB_UDT_M2".MDIB.M2.M_3_EN_M2	DB2.DBX179.3	-- enable bit for M funct
		ion group 3	
L	B#16#0		
T	"DB_UDT_M2".MDIB.M2.G_2_VAL_M2	DB2.DBB181	-- value G function group
		2	
T	"DB_UDT_M2".MDIB.M2.M_1_VAL_M2	DB2.DBB192	-- value M function group
		1	
T	"DB_UDT_M2".MDIB.M2.M_2_VAL_M2	DB2.DBB193	-- value M function group
		2	
T	"DB_UDT_M2".MDIB.M2.M_3_VAL_M2	DB2.DBB194	-- value M function group
		3	
R	"memlautao2"	M246.2	
S	"DB_UDT_M2".MDIB.M2.G_1_EN_M2	DB2.DBX178.0	-- enable bit for G funct
		ion group 1	
L	90		
T	"DB_UDT_M2".MDIB.M2.G_1_VAL_M2	DB2.DBB180	-- value G function group
		1	
S	"DB_UDT_M2".MDIB.M2.X_T_EN_M2	DB2.DBX178.4	-- enable bit for positio
		n / dwell time	
L	"Counter_v222"	MD170	
T	"DB_UDT_M2".MDIB.M2.X_T_VAL_M2	DB2.DBD184	-- value position / dwell
		time	

(Anexo M) 43

Page 9 of 13

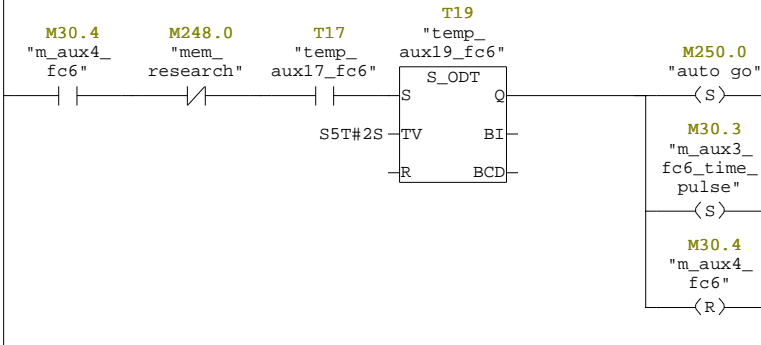
S	"DB_UDT_M2".MDIB_M2.V_EN_M2	DB2.DBX179.0	-- enable bit for velocity
L	L#30000		
T	"DB_UDT_M2".MDIB_M2.V_VAL_M2	DB2.DBD188	-- value velocity
S	"DB_UDT_M2".MDI_EN_M2 // Set request bit for MDI block	DB2.DBX38.3	-- MDI movement block

Network: 11

Pausa criada dada a necessidade de enviar os dados para as respectivas funções e só depois deslocar o sistema para posição de máxima potência registada.

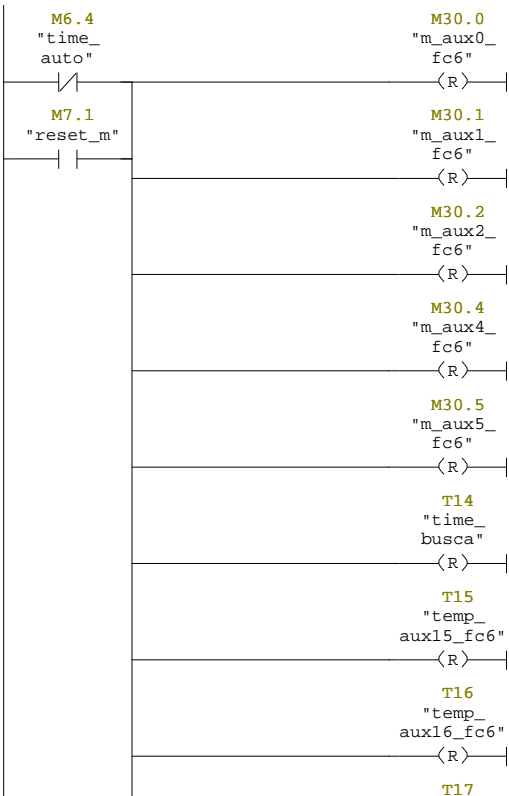
Indicação para melhoramentos do sistema:

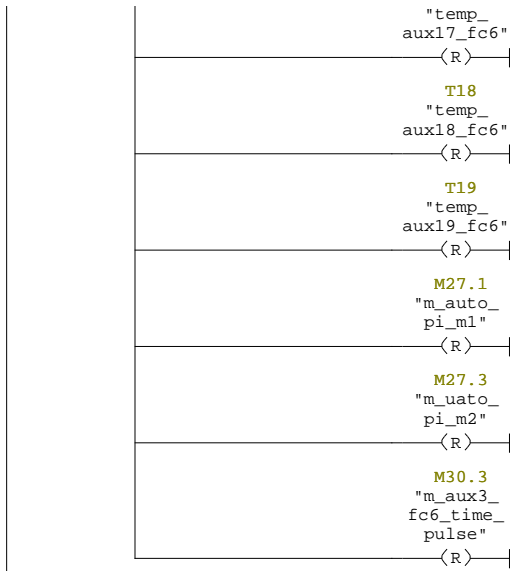
- Alterar a programação evitando a utilização de temporizadores.



Network: 12

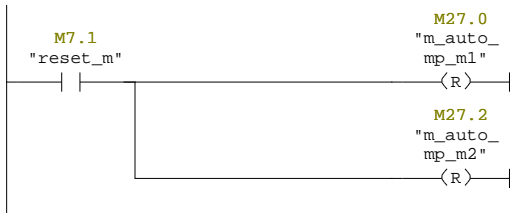
Reset das memórias e temporizadores auxiliares





Network: 13

Reset de memórias auxiliares



FC7 - <offline>

"TIMER" Atribuição do intervalo entre varrimentos
Name: TIMER **Family:**
Author: EACF **Version:** 0.1
Block version: 2
Time stamp Code: 08/12/2012 03:57:32 PM
Interface: 03/05/2011 04:48:17 AM
Lengths (block/logic/data): 00284 00172 00006

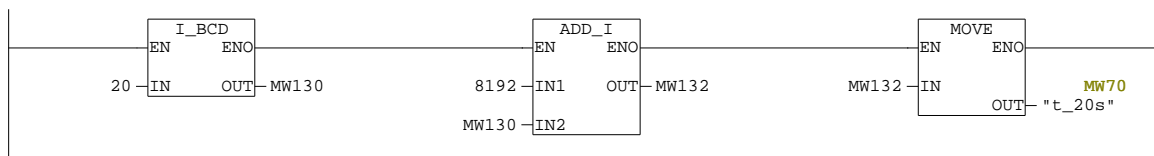
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC7 Intervalo_Varrimentos

Por defeito, o tempo mínimo entre varrimentos é 20s

Network: 1

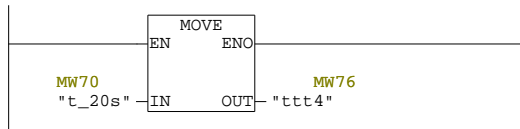
Igual à efectuada na FB de conversão (tempo de pausa)



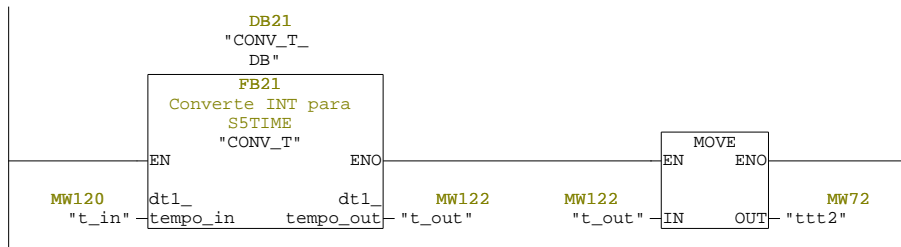
Page 1 of 4

Network: 2

Para possibilitar a comparação em número inteiros

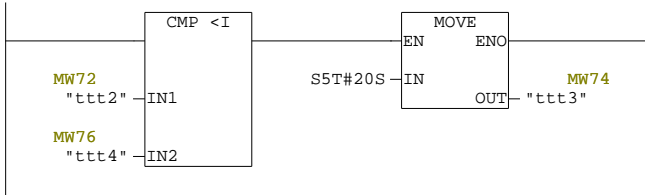
**Network: 3**

Para possibilitar a comparação em número inteiros



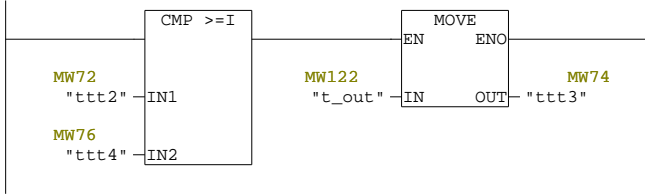
Network: 4

Se o tempo introduzido pelo operador no painel HMI for inferior a 20s, o programa de utilizador ignora esse valor e carrega o temporizador com 20s. Se o valor introduzido for igual ou maior a 20s, é esse o valor carregado no temporizador.



Network: 5

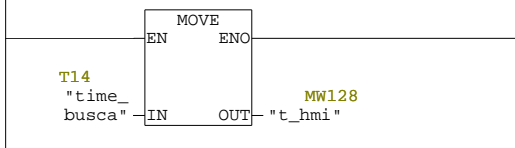
Igual ...



Page 3 of 4

Network: 6

Para o utilizador ler o tempo em falta em segundos (HMI e WinCC)



FC8 - <offline>

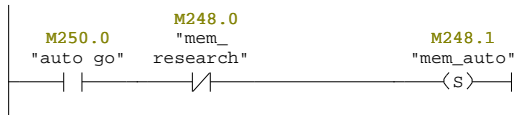
"MOD_68" Seleção do modo de funcionamento
Name: MOD_86 **Family:**
Author: EACF **Version:** 0.1
Block version: 2
Time stamp Code: 08/12/2012 03:58:47 PM
Interface: 02/18/2011 04:07:17 PM
Lengths (block/logic/data): 00294 00184 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC8 MOD_86

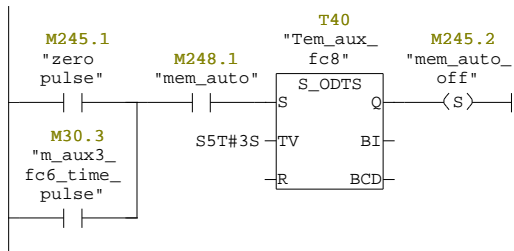
Onde é activado o modo de varrimento (F13 HMI) ou o modo de para colocação na posição de máx. pot. (F9 HMI). Notar que para o sistema voltar à posição inicial e sair do modo F9 (HMI), basta pressionar F10 (HMI).
 Notar que os estado das variáveis na FC8 interferem no funcionamento da FC12.

Network: 1 Activação do modo de colocação manual na pos. de máx. pot.



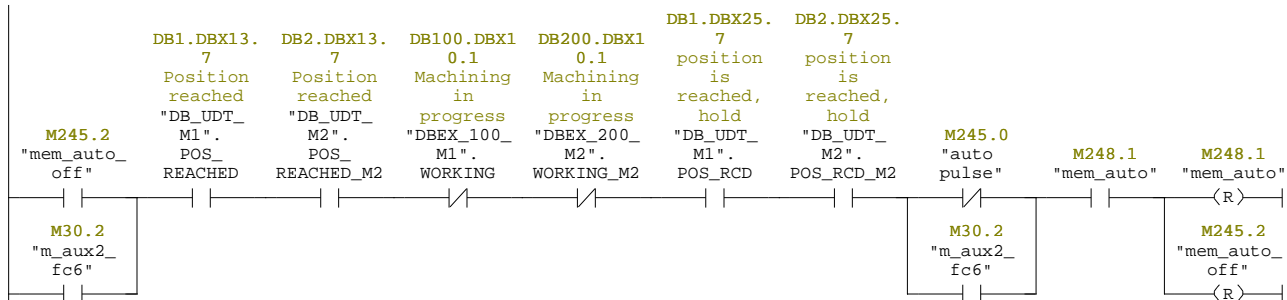
Page 1 of 5

Network: 2 Activação do modo de colocação automática na posição de máx pot



Network: 3

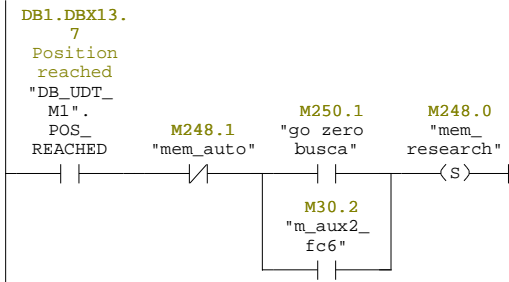
Verificação das condições do sistema para activação



Network: 4 Res do temp auxiliar

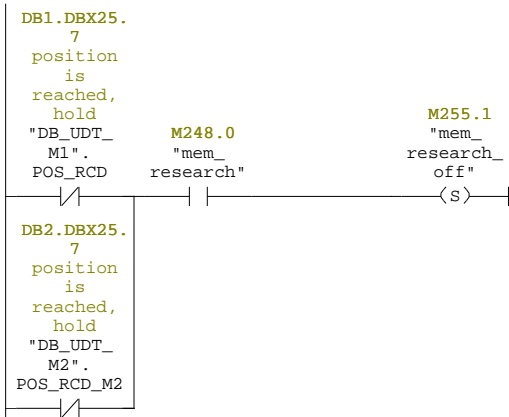


Network: 5 Activação do modo de varrimento



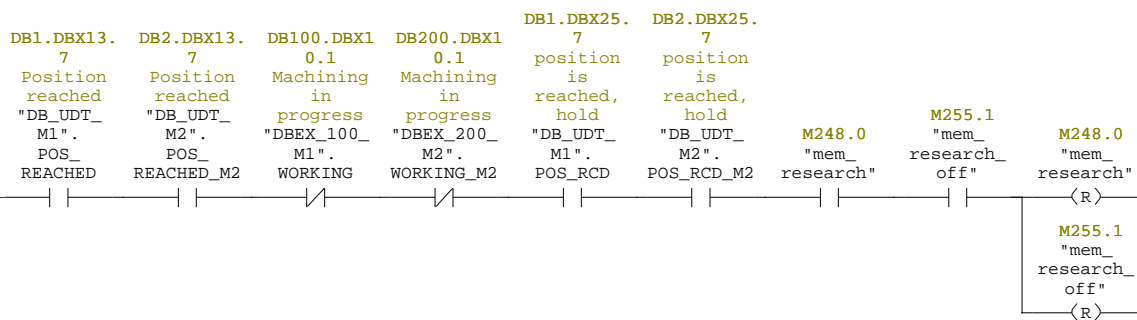
Page 3 of 5

Network: 6 Confirmação de sistema a executar varrimento

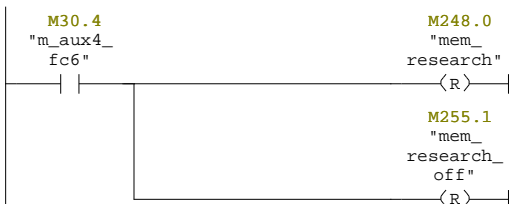


Network: 7 Desactivação do modo research (1 de 2)

Verificação das condições do sistema



Network: 8 Desactivação do modo research (2 de 2)



Page 5 of 5

FC9 - <offline>

"MICRO" Varrimentos solares, com ou sem sinal do sistema secundário

Name: MICRO Family:
 Author: EACF Version: 0.1
 Block version: 2
 Time stamp Code: 07/18/2012 05:48:47 PM
 Interface: 07/07/2012 07:33:21 PM
 Lengths (block/logic/data): 00738 00576 00002

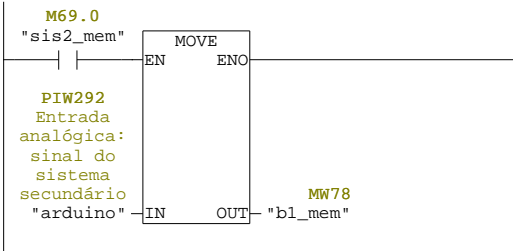
Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC9 Comparação dos sinais enviados pelo Sistema Secundário

Comparação entre os sinais [0 a 5 V] enviados pelo microcontrolador
 Notar que na programação do Arduino existe um delay e por esse motivo não é
 necesserário o uso de temporizadores para regular a leitura do sinal de tensão
 eléctrica.

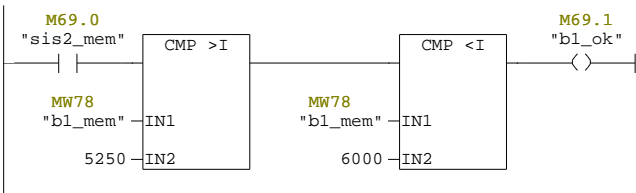
Network: 1 Recebe PIW

Se o Sistema Secundário (Sis2) estiver activo (HMI ou WinnCC):
Recebe PIW e guarda o valor em MW (INT)



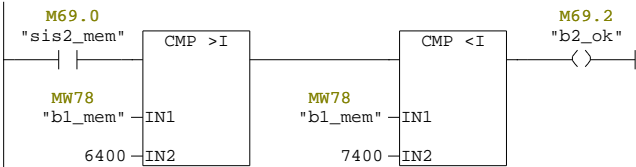
Network: 2 Enquadramento de valores para validar o varrimento 1

[]
[] [x]
[x]



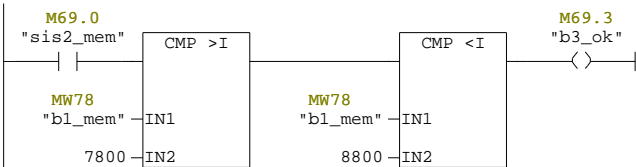
Network: 3 Enquadramento de valores para validar o varrimento 2

[x]
[] [x]

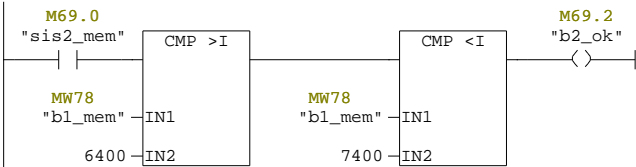


Network: 4 Enquadramento de valores para validar o varrimento 3

[x]
[x] []
[]

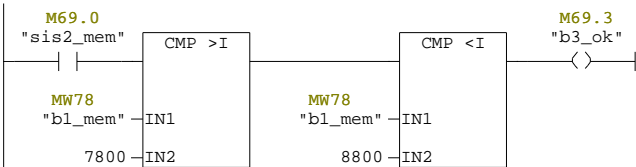


[]



Network: 4 Enquadramento de valores para validar o varrimento 3

[x]
[x] []
[]

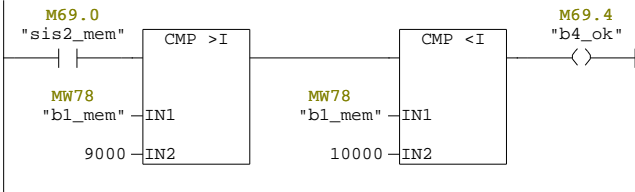


Network: 5 Enquadramento de valores para validar o varrimento 4

```

[ ]
[x] [ ]
[x]

```

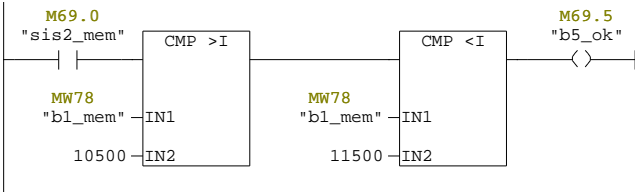


Network: 6 Enquadramento de valores para validar o varrimento 5

```

[x]
[ ] [ ]
[x]

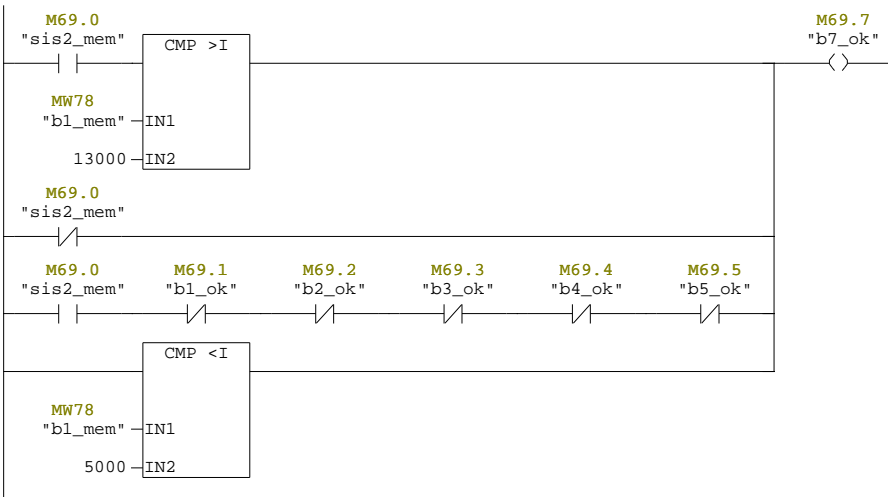
```



Page 4 of 13

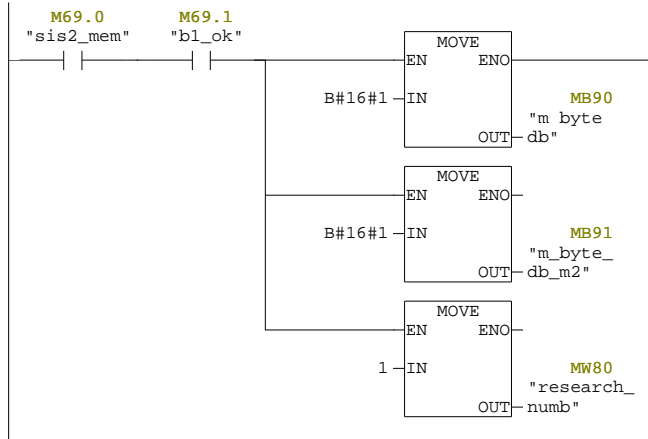
Network: 7 Enquadramento de valores para validar o varrimento total

No caso de nenhum dos varrimentos referidos se adequar, o Sistema Secundário apresentar algum problema de funcionamento ou o operador o desactivar no painel HMI, o Sistema Principal utiliza o varrimento total (b7)



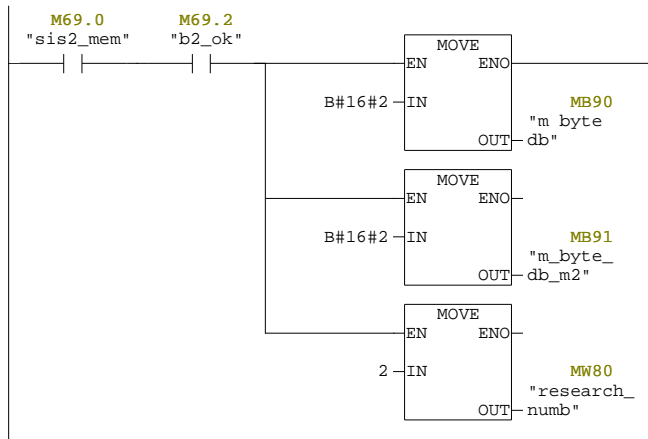
Network: 8 Número do DB que corresponde ao varrimento pré-programado 1

Nas FCs 103 e 203 (Sub_Prog_M1_3 e Sub_Prog_M2_3), que correspondem ao modo de varrimento, é aguardado o valor armazenado nas variáveis "m byte db" e "m_byte_db_m2" por forma a aceder a um dos DBs arquivados nos FM353, que contêm a sequência dos movimentos a executar pelos motores, para cada varrimento.

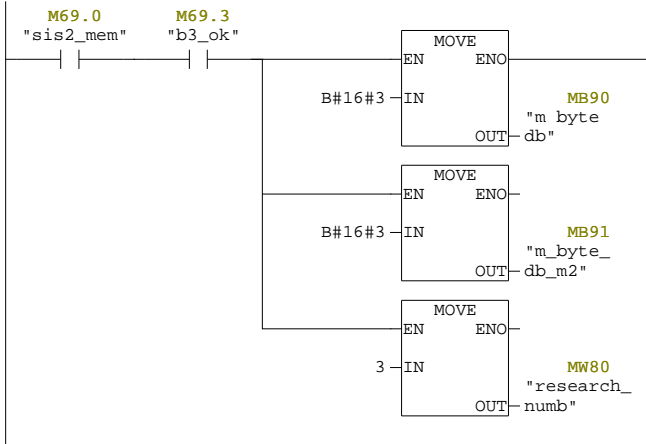


Page 6 of 13

Network: 9 Número do DB que corresponde ao varrimento pré-programado 2

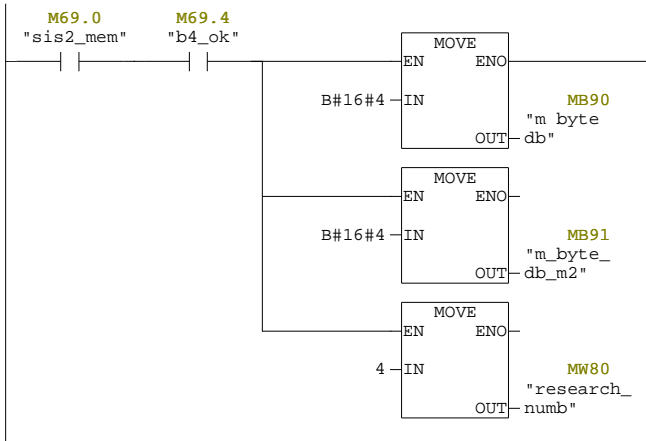


Network: 10 Número do DB que corresponde ao varrimento pré-programado 3

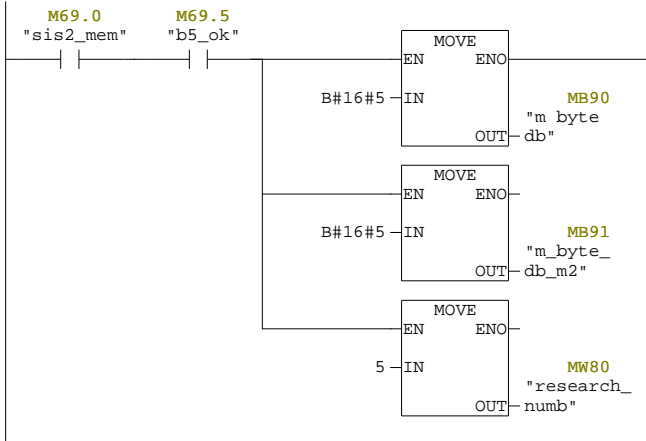


Page 8 of 13

Network: 11 Número do DB que corresponde ao varrimento pré-programado 4

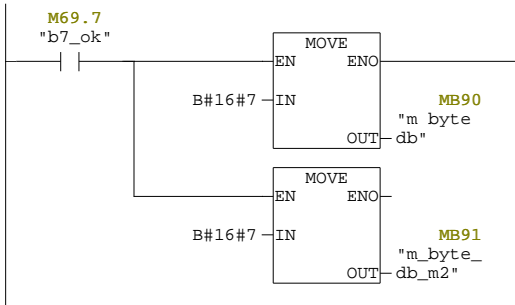


Network: 12 Número do DB que corresponde ao varrimento pré-programado 5



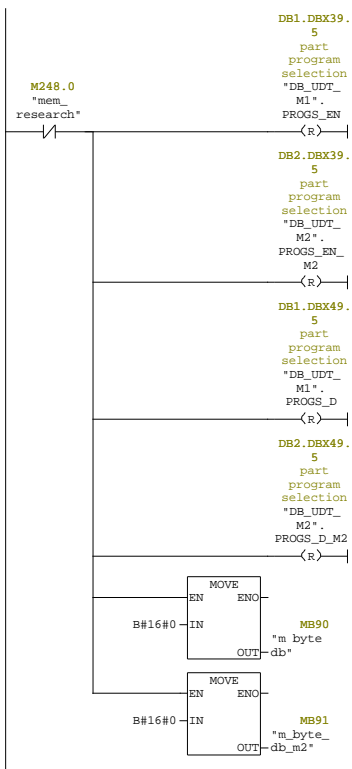
Page 10 of 13

Network: 13 Número do DB que corresponde ao varrimento pré-programado total



Network: 14 Reset das variáveis

Quando sistema sai do modo de varrimento o valor do DB que corresponde ao varrimento executado é colocado a "0"



Page 12 of 13

FC10 - <offline>

"MCU_ENC1" FM350-1 Encoder 1 (module address, channel address, user data length)
Name: MCU_ENC1 **Family:**
Author: EACF **Version:** 0.1
Block version: 2
Time stamp Code: 07/18/2012 04:49:39 PM
Interface: 07/14/2012 08:37:23 PM
Lengths (block/logic/data): 00126 00034 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC10 ENC MCU

Função relativa ao FM350-1 (Encoder 1)

Network: 1

```

L 320
T "DB_UDT_ENC1".MOD_ADR //FM 350-1/450-1 address DB3.DBW6 -- Module address (write user)

L P#320.0
T "DB_UDT_ENC1".CH_ADR //FM channel address (pointer format) DB3.DBD8 -- Channel address (write user)

L 16
T "DB_UDT_ENC1".U_D_LGTH //Length of user data interface DB3.DBB12 -- User data length (write user)

```

Page 1 of 1

FC11 - <offline>

"MCU_ENC2" FM350-1 Encoder 2 (module address, channel address, user data length)
Name: MCU_ENC2 **Family:**
Author: EACF **Version:** 0.1
Block version: 2
Time stamp Code: 07/18/2012 04:49:53 PM
Interface: 07/14/2012 08:39:46 PM
Lengths (block/logic/data): 00126 00034 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC11**Network: 1**

```

L 336
T "DB_UDT_ENC2".MOD_ADR //FM 350-1/450-1 address DB4.DBW6 -- Module address (write user)

L P#336.0
T "DB_UDT_ENC2".CH_ADR //FM channel address (pointer format) DB4.DBD8 -- Channel address (write user)

L 16
T "DB_UDT_ENC2".U_D_LGTH //Length of user data interface DB4.DBB12 -- User data length (write user)

```

FC12 - <offline>

"PRE_MOD" Código de interligação entre funções tecnológicas
Name: PRE_MOD **Family:**
Author: EACF **Version:** 0.1
Block version: 2
Time stamp Code: 07/18/2012 04:50:11 PM
Interface: 07/14/2012 08:41:31 PM
Lengths (block/logic/data): 00336 00234 00002

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC12 MOD_PRE_CODE

Leitura da posição angular dos eixos 1 e 2

Chama a função onde se programou a transição entre o modo de varrimento e o de colocação na pos. de máx. pot.

Ref. Point Approach (Pos. Inicial)
 Jog (Comando manual a partir do painel HMI)
 Automatic (varrimento)
 MDI (colocação na pos. de máx. pot.)

Network: 1 Selecção do modo de funcionamento

```
//ENCODER *****
A   "SYNC_OK"                M25.0

L   W#16#0
T   "DB_PROG_ENC1".L_DIRECT_VAL  DB300.DBD4    -- New counter value
T   "DB_PROG_ENC2".L_DIRECT_VAL  DB400.DBD4    -- New counter value

A   "SYNC_OK"                M25.0
S   "DB_PROG_ENC1".SET_L_DIRECT  DB300.DBX0.7  -- l=New counter value
S   "DB_PROG_ENC2".SET_L_DIRECT  DB400.DBX0.7  -- l=New counter value

// Obtenção dos valores da posição angular dos eixos 1 e 2
// Atencao: OP (tag; decimal; 2 casas => 9000=90.00)
L   "DB_PROG_ENC1".ACT_CNTV     DB300.DBD32   -- Current count value
L   L#72
*I
T   "Counter_value"            MD44

L   "DB_PROG_ENC2".ACT_CNTV     DB400.DBD32   -- Current count value
L   L#18
*I
T   "Counter_value_M2"         MD48

//*****

JC   M001
// Ref. Pt. Approach (pos. inicial)
L   B#16#3
T   "DBEX_200_M2".MODE_IN_M2    DB200.DBB7    -- Mode setting (coded)
T   "DBEX_100_M1".MODE_IN      DB100.DBB7    -- Mode setting (coded)
A   "cont_aux_fc12"            C100
AN  "temp_aux1_fc12"           T100
SD  "temp_aux2_fc12"           T101
A   "temp_aux2_fc12"           T101
SD  "temp_aux1_fc12"           T100
FN  "mem_aux1_fc12"            M249.0
=   "mem_aux2_fc12"            M249.1
A   "mem_aux2_fc12"            M249.1
S   "DBEX_200_M2".EX1_M2.START_M2  DB200.DBX12.2 -- Start
S   "DBEX_100_M1".EX1.START    DB100.DBX12.2 -- Start
R   "mem_aux1_fc12"            M249.0
CD  "cont_aux_fc12"            C100
```

M001: NOP 0

```
//Confirma se o motor 1 está sincronizado
A "DB_UDT_M1".SYNC DB1.DBX25.0 -- synchronized
//Confirma se o motor 2 está sincronizado
A "DB_UDT_M2".SYNC_M2 DB2.DBX25.0 -- synchronized
JCN M002

CALL "MOD_68" FC8 -- Selecção do modo de funcionamento

// Após ref pt aproach, chama o modo desejado
AN "mem_auto" M248.1
AN "mem_research" M248.0

// Jog
L B#16#1
JC m100
A "mem_auto" M248.1
AN "mem_research" M248.0

// Pos máx pot
L B#16#6
JC m100
A "mem_research" M248.0
AN "mem_auto" M248.1

// Varrimento
L B#16#8
JC m100

m100: NOP 0
T "DBEX_100_M1".MODE_IN DB100.DBB7 -- Mode setting (coded)
T "DBEX_200_M2".MODE_IN_M2 DB200.DBB7 -- Mode setting (coded)
T "DB_UDT_M1".MODE_IN DB1.DBB16 -- mode
T "DB_UDT_M2".MODE_IN_M2 DB2.DBB16 -- mode

M002: NOP 0
A "Data_Enter" M23.0
S "DB_UDT_M1".JOBRESET DB1.DBX69.1 -- Reset_ERR and _D
S "DB_UDT_M2".JOBRESET_M2 DB2.DBX69.1 -- Reset_ERR and _D
```

Page 3 of 3

FC13 - <offline>

"RESET_FC" Função do Reset (F7 HMI)
Name: RESET_FC **Family:**
Author: EACF **Version:** 0.1
Block version: 2
Time stamp Code: 07/18/2012 06:36:32 PM
Interface: 07/14/2012 08:44:52 PM
Lengths (block/logic/data): 00140 00048 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC13 Reset_FC

Quando o operador pressiona o botão de Reset (F7 HMI), o sistema sai do modo de STOP, posicionamento na máx. pot ou varrimento.
 Algumas variáveis tomam valor lógico zero e são criadas as condições para o sistema ir para o modo de Ref. Pt. Aproach, deslocando-se para a posição inicial.

Network: 1

```
A "reset_m" M7.1
L C#1
S "cont_aux_fc12" C100
R "mem_auto" M248.1
R "mem_research" M248.0
R "DBEX_100_M1".EX1.STOP DB100.DBX12.3 -- Stop
S "DB_UDT_M1".RESET_AX DB1.DBX37.5 -- reset axis
S "DB_UDT_M2".RESET_AX_M2 DB2.DBX37.5 -- reset axis
R "DB_PROG_ENC1".SET_L_DIRECT DB300.DBX0.7 -- 1=New counter value
```

(Anexo M) 59

Page 1 of 2

R "DB_PROG_ENC2".SET_L_DIRECT DB400.DBX0.7 -- l=New counter value

Page 2 of 2

FC14 - <offline>

"SYNC_FC" Confirma sincronização (eixo 1 e eixo 2)

Name: SYNC_FC **Family:**
Author: EACF **Version:** 0.1
Block version: 2
Time stamp Code: 07/18/2012 04:50:51 PM
Interface: 07/14/2012 08:46:05 PM
Lengths (block/logic/data): 00112 00016 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC14 Confirma_SYNC

Se os sistemas de eixos motorizados estão sincronizados: M25.0="1"

Network: 1

A "DB_UDT_M1".SYNC DB1.DBX25.0 -- synchronized
A "DB_UDT_M2".SYNC_M2 DB2.DBX25.0 -- synchronized
= "SYNC_OK" M25.0

FC15 - <offline>

"HMI_WCC" Auxiliar na criação de botões (HMI - WinCC)

Name: HMI_WCC

Family:

Author: EACF

Version: 0.1

Block version: 2

Time stamp Code: 07/31/2012 05:07:37 PM

Interface: 07/26/2012 06:40:00 PM

Lengths (block/logic/data): 00222 00122 00002

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC15 HMI_WCC

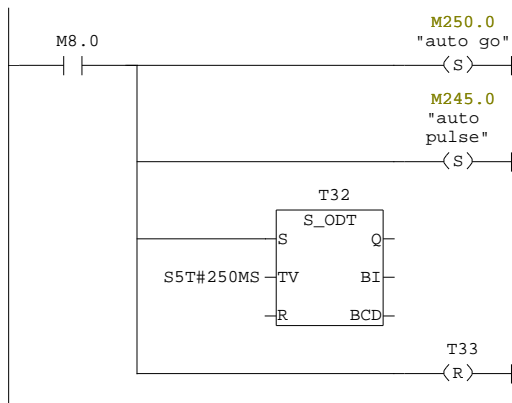
Função auxiliar à activação das variáveis "auto go" e "auto pulse", no WinCC Flexible e no WinCC

Network: 1



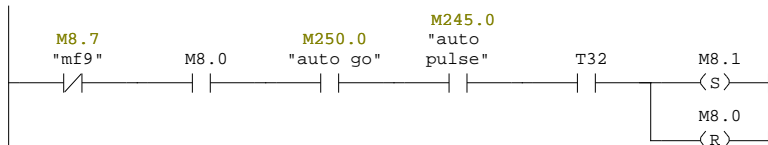
Page 1 of 3

Network: 2

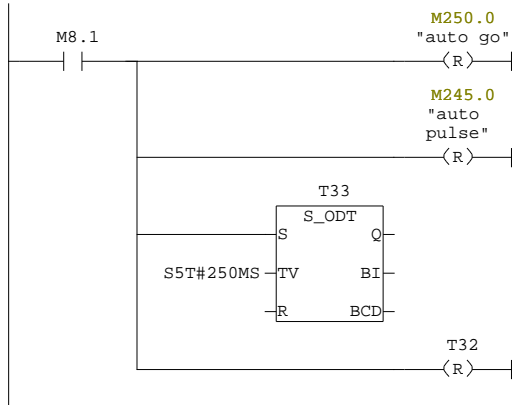


Network: 3

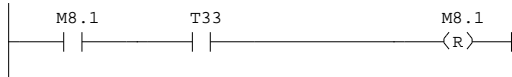
só quero que funcione se n tiver lá o dedo: mf9 negado na condição



Network: 4



Network: 5



FC16 - <offline>

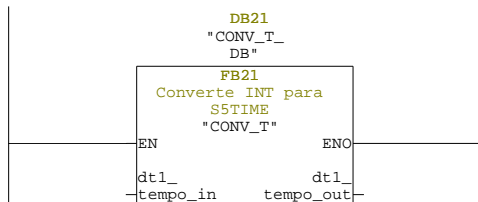
"Call_FB21" Ajuda a chamar mais facilmente, no OB1, a FB21
Name: Call_FB21 **Family:**
Author: EACF **Version:** 0.1
Block version: 2
Time stamp Code: 08/01/2012 04:55:57 PM
Interface: 07/31/2012 08:19:48 PM
Lengths (block/logic/data): 00128 00036 00006

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC16 FC_Call_FB21

Ajuda a chamar mais facilmente, no OB1, a FB21

Network: 1



FC100 - <offline>

"PROG_M1" Função principal FM353 (motor1)
Name: MODE_EX **Family:** FM_ST_SV
Author: AuDMCE8 **Version:** 2.0
Block version: 2
Time stamp Code: 10/23/2000 10:14:22 AM
Interface: 02/24/2000 11:22:02 AM
Lengths (block/logic/data): 00296 00194 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC100 Mode select

This example is always required for Examples 1 to 3.
The block for this example is FC 100.
The signals are in "DBEX".
This example must always be called. It sets the desired modes, evaluates the mode checkback, and displays the current mode.
The checkback signals required for the examples are copied to "DBEX".
In order to be able to work with Example 1 in "Jog" or "Reference point approach" mode, the user must enter the relevant code in the MODE_IN byte in "DBEX" (01 for "Jog" mode, 03 for "Reference point approach" mode).
If "Jog" mode is selected, mode parameter 01 (MODE_TYPE) is also set to activate velocity level 1 for "Jogging".
Mode Code
Jog 01
Ref.point approach 03
MDI 06
Automatic 08
In Example 2 you must set "MDI" mode (code in byte MODE_IN = 06).

In Example 3 you must set "Automatic" mode (code in byte MODE_IN = 08). The code in byte MODE_OUT shows the mode that is currently active. In order to restart the module (e.g. following a diagnostic interrupt), the RESET_AX bit has to be set in "DBEX". The example then sets the RESET_AX bit in the "user DB", the module is restarted, and the RESET_AX bit in "DBEX" is once again reset. In order to be able to work with any example, you must set the mode required for the example you want to use.

Network: 1 SETTING THE MODE

```

A      "DB_UDT_M1".WORKING      // When work in progress, -> no mode s.      DB1.DBX23.1      -- working
JC     MDOU

L      B#16#0
L      "DBEX_100_M1".MODE_IN    // Mode selection      DB100.DBB7      -- Mode setting (coded)
<>I
JC     MDIN
L      B#16#1
MDIN: T      "DB_UDT_M1".MODE_IN // If no mode entered, select Jog      DB1.DBB16      -- mode
L      "DB_UDT_M1".MODE_IN      // Is Jog mode activated ?      DB1.DBB16      -- mode
L      B#16#1
<>I
JC     MDOU
L      B#16#1
T      "DB_UDT_M1".MODE_TYPE    // Enter mode parameter 1      DB1.DBB17      -- mode parameter
// in user DB

MDOU: L      "DB_UDT_M1".MODE_OUT // Mode checkback      DB1.DBB24      -- mode
T      "DBEX_100_M1".MODE_OUT  // Entry in DBEX      DB100.DBB8      -- Mode status (codiert)

```

Network: 2 FM-RESTART

```

A      "DBEX_100_M1".RESET_AX  // Restart ?      DB100.DBX9.3      -- Restart
S      "DB_UDT_M1".RESET_AX    // Enter Restart in DB_FM and      DB1.DBX37.5      -- reset axis
R      "DBEX_100_M1".RESET_AX  // reset it in DBEX      DB100.DBX9.3      -- Restart

```

Page 2 of 3

Network: 3 COPY CHECKBACK SIGNALS TO 'DBEX'

```

A      "DB_UDT_M1".GO_M        // Travel -      DB1.DBX25.2      -- go_minus
=      "DBEX_100_M1".GO_M
A      "DB_UDT_M1".GO_P        // Travel +      DB100.DBX10.2    -- Travel -
=      "DBEX_100_M1".GO_P      DB1.DBX25.3      -- go_plus
DB100.DBX10.3    -- Travel +

A      "DB_UDT_M1".OT_ERR      // Operator/traversing error      DB1.DBX22.3      -- ot-error
=      "DBEX_100_M1".OT_ERR    DB100.DBX10.4    -- Operator/traversing error

A      "DB_UDT_M1".DATA_ERR    // Data error      DB1.DBX22.4      -- data-error
=      "DBEX_100_M1".DATA_ERR  DB100.DBX10.5    -- Data error

A      "DB_UDT_M1".PARA        // Channel initialized      DB1.DBX22.7      -- parameterized
=      "DBEX_100_M1".PARA      DB100.DBX9.5      -- Parameterized

A      "DB_UDT_M1".SYNC        // synchronized      DB1.DBX25.0      -- synchronized
=      "DBEX_100_M1".SYNC      DB100.DBX9.6      -- Synchronized

A      "DB_UDT_M1".ST_ENBLD    // Start enable      DB1.DBX23.0      -- start enable
=      "DBEX_100_M1".START_EN  DB100.DBX9.7      -- Start enable

A      "DB_UDT_M1".POS_RCD     // Position reached, stop      DB1.DBX25.7      -- position is reached, hold
=      "DBEX_100_M1".POS_RCD   DB100.DBX10.0    -- Position reached

A      "DB_UDT_M1".WORKING     // Machining in progress      DB1.DBX23.1      -- working
=      "DBEX_100_M1".WORKING   DB100.DBX10.1    -- Machining in progress

```

FC101 - <offline>

"SUB_PROG_M1_1" Ref. Pt. Approach (motor 1)
Name: TIP_REF **Family:** FM_ST_SV
Author: AuDMCE8 **Version:** 2.1
Block version: 2
Time stamp Code: 07/18/2012 06:51:00 PM
Interface: 02/24/2000 11:22:46 AM
Lengths (block/logic/data): 00326 00208 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC101 Jog and Reference point approach

The block for this example is FC 101.
The signals are in "DBEX", the signals for Example 1 only are in structure "EX1".
The drive enable and the servo enable for the axis are set in "DBEX" (in OB100: DRV_EN = TRUE, SERVO_EN = TRUE) and are transferred to the interface in Example 1 (user DB).
In order for the example to work, either "Jog" mode (mode code 01) or "Reference point approach" mode (mode code 03) must have been entered in byte MODE_IN in "DBEX". The relevant mode checkback is shown in byte MODE_OUT.
The traversing movements are flagged in bits "GO_M" = TRUE (axis motion -) or "GO_P" = TRUE (axis 1 motion +).
"Jog" mode active:
When a mode is activated, the Write command "VLEV_EN" (user DB, transfer velocity levels 1, 2) is executed automatically. If retransfer is required, you must either reset the "VLEV_D" bit (job status/checkback) or set the "JOBRESET" bit (reset status/error) in the user DB.
If you set the "DIR_M" (motion -) or "DIR_P" (motion +) bit to = TRUE

Page 1 of 4

in "DBEX", the axis will traverse in either a negative or positive direction.
"Reference point approach" mode active:
When the "START" bit is set to TRUE, the axis is moved in a negative or positive direction (depending on machine data parameterization) until the reference point is located. Following successful reference point approach, the axis is synchronized (SYNC=TRUE).
If an operator error or traversing error occurred, it is flagged in the "OT_ERR" bit (bit = TRUE).
The error can be acknowledged by setting the "OT_ERR_A" bit to TRUE.
Note:
The "Blocks" directory contains variable table 1 (VAT1), which in turn contains all signals needed to monitor and control Example 1 ("Monitor and control variables" tool).

Network: 1 QUERY MODE

```

L   B#16#1           // Code for Jog mode
L   "DBEX_100_M1".MODE_OUT // Mode checkback                DB100.DBB8      -- Mode status (codiert)
==I
JC  DRV             // Jump if Jog mode active
L   B#16#3           // Code for ref. point approach mode
==I
JC  DRV             // Jump if ref. point approach mode active
JU  END             // If these modes are not active -> End

```

Network: 2 SETTING DEFAULTS IN THE USER DATA BLOCK

```

DRV: A   "DBEX_100_M1".DRV_EN // Drive enable ?                DB100.DBX9.0   -- Drive enable
      =   "DB_UDT_M1".DRV_EN // Entry in user DB                DB1.DBX15.7   -- drive enable

A   "DBEX_100_M1".SERVO_EN // Servo enable ?                DB100.DBX9.1   -- Servo enable
      =   "DB_UDT_M1".SERVO_EN // Entry in user DB                DB1.DBX34.0   -- servo enable

L   "DBEX_100_M1".OVERRIDE // Default = 100%                DB100.DBB6     -- Override
T   "DB_UDT_M1".OVERRIDE // Entry in user DB                DB1.DBB18     -- override

```

Network: 3		JOG / REFERENCE POINT APPROACH			
L	B#16#1	// Code for Jog mode			
L	"DBEX_100_M1".MODE_OUT	// Mode checkback	DB100.DBB8	--	Mode status (codiert)
==I	JC	JOG			
L	B#16#3	// Code for reference point approach mode			
==I	JC	REFP			
JU	END	// If modes not active -> End			
JOG: A	"DB_UDT_M1".VLEV_D	// Transfer velocity levels and	DB1.DBX48.0	--	velocity levels 1 and 2
JC	DIR				
L	"VLEVEL1_M1_OP"		MD15		
T	"DB_UDT_M1".VLEVEL_1	// Value for velocity level 1	DB1.DBD160	--	velocity level 1
L	L#80000				
T	"DB_UDT_M1".VLEVEL_2	// Value for velocity level 2	DB1.DBD164	--	velocity level 2
S	"DB_UDT_M1".VLEV_EN	// Transfer velocity levels	DB1.DBX38.0	--	velocity levels 1 and 2
JU	MA01				
DIR: A	"TRAVEL_M_M1_m"	//A "DBEX".EX1.DIR_M	Travel - comm	M7.2	
=	"DB_UDT_M1".DIR_M	and			
A	"TRAVEL_P_M1_m"	//A "DBEX".EX1.DIR_P	Travel + comm	M7.3	-- direction minus
=	"DB_UDT_M1".DIR_P	and			
JU	MA01				-- direction plus
REFP: A	"DBEX_100_M1".EX1.START		DB100.DBX12.2	--	Start
S	"DB_UDT_M1".START	// Entry in user DB	DB1.DBX15.0	--	start
R	"DBEX_100_M1".EX1.START		DB100.DBX12.2	--	Start
A	"stop_m"		M7.0		
S	"DBEX_100_M1".EX1.STOP		DB100.DBX12.3	--	Stop
=	"DB_UDT_M1".STOP	// Entry in user DB	DB1.DBX15.1	--	stop

Page 3 of 4

Network: 4		OPERATOR/TRAVERSING ERROR; ACKNOWLEDGEMENT			
MA01: AN	"DB_UDT_M1".OT_ERR	// If no operator/traversing error present, ..	DB1.DBX22.3	--	ot-error
R	"DBEX_100_M1".OT_ERR_A	// .. reset error acknowledgement	DB100.DBX9.2	--	Operator/traversing error
A	"DBEX_100_M1".OT_ERR_A	// Error acknowledgement	DB100.DBX9.2	--	Operator/traversing error
=	"DB_UDT_M1".OT_ERR_A	// Enter error acknowledgement in user DB	DB1.DBX14.3	--	quit error
END: NOP	0				

FC102 - <offline>

"SUB_PROG_M1_2" Deslocamento para posição de máx. pot. (motor 1)
Name: MDI **Family:** FM_ST_SV
Author: AuDMCE8 **Version:** 2.0
Block version: 2
Time stamp Code: 07/18/2012 06:22:31 PM
Interface: 02/24/2000 11:22:32 AM
Lengths (block/logic/data): 00460 00332 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC102 MDI

The block for this example is FC 102.
The signals are in "DBEX", the signals for Example 2 only are in structure "EX2".
The drive enable and servo enable signals for the axis are set in "DBEX" (in QB100: DRV_EN = TRUE, SERVO_EN = TRUE), and are transferred to the interface in Example 2 (user DB).
In order for the example to work, you have to set the "MDI" mode. Enter "MDI" mode (code 06) in the MODE_IN byte in "DBEX". The relevant mode status is signalled in the MODE_OUT byte.
Following successful mode selection, a default MDI block is automatically transferred to the module (network MDI) by setting Write request "MDI_EN" in the user DB (transfer MDI block).
This block is modifiable in dependence on the system or the request.
To retransfer the block, you must either reset the "MDI_D" bit (job status/checkback)
or set the "JOBRESET" bit (reset status/error) in the user DB.
Set the "START" bit in the "DBEX" in structure "EX2" to TRUE. The activated MDI block is started if the axis is synchronized and has a start enable.
The "START" bit is then reset.
The MDI block cannot be restarted until the start enable signal is once

Page 1 of 5

again present.
The block can be stopped via the "STOP" bit. It cannot be started again until the "STOP" bit has been reset to FALSE (and "START" is = TRUE).
An operator error or traversing error, if any, is flagged in the "OT_ERR" bit (bit = TRUE).
The error can be acknowledged by setting the "OT_ERR_A" bit to TRUE.
Note:
The "Blocks" directory contains variable table 2 (VAT2), which in turn contains all signals needed to monitor and control Example 2 (Tool "Monitor and control variables" tool).

Network: 1

```

L   B#16#6           // Code for MDI mode ---- AUTOMATIC OR MANUAL F9
L   "DBEX_100_M1".MODE_OUT // Mode checkback           DB100.DBB8      -- Mode status (codiert)
==I
JCN  END

```

Network: 2

F9 ou F10, para se movimentar

```

A(
O   "auto pulse"     M245.0
O   "zero pulse"     M245.1
O   "m_auto_pi_m1"   M27.1
)
L   C#1
S   "contlautol"     C15

```

Network: 3 SETTING DEFAULTS IN THE USER DATA BLOCK

```

A   "DBEX_100_M1".DRV_EN // Drive enable ?           DB100.DBX9.0  -- Drive enable
=   "DB_UDT_M1".DRV_EN   // Entry in user DB           DB1.DBX15.7   -- drive enable
A   "DBEX_100_M1".SERVO_EN // Servo enable ?         DB100.DBX9.1  -- Servo enable
=   "DB_UDT_M1".SERVO_EN // Entry in user DB           DB1.DBX34.0   -- servo enable
L   "DBEX_100_M1".OVERRIDE // Default = 100%         DB100.DBB6    -- Override
T   "DB_UDT_M1".OVERRIDE // Entry in user DB           DB1.DBB18     -- override

```

(Anexo M) 67

Page 2 of 5

Network: 4 MDI				
A	"DB_UDT_M1".MDI_D	// Transfer MDI block and	DB1.DBX48.3	-- MDI movement block
JC	MDST			
JCN	AAA			
AAA: R	"DB_UDT_M1".MDIB.G_2_EN	// Delete MDI block areas	DB1.DBX178.1 group 2	-- enable bit for G function
R	"DB_UDT_M1".MDIB.M_1_EN		DB1.DBX179.1 group 1	-- enable bit for M function
R	"DB_UDT_M1".MDIB.M_2_EN		DB1.DBX179.2 group 2	-- enable bit for M function
R	"DB_UDT_M1".MDIB.M_3_EN		DB1.DBX179.3 group 3	-- enable bit for M function
L	B#16#0			
T	"DB_UDT_M1".MDIB.G_2_VAL		DB1.DBB181	-- value G function group 2
T	"DB_UDT_M1".MDIB.M_1_VAL		DB1.DBB192	-- value M function group 1
T	"DB_UDT_M1".MDIB.M_2_VAL		DB1.DBB193	-- value M function group 2
T	"DB_UDT_M1".MDIB.M_3_VAL		DB1.DBB194	-- value M function group 3
R	"memlautol"		M247.2	
S	"DB_UDT_M1".MDIB.G_1_EN	// Enter MDI block (example)	DB1.DBX178.0 group 1	-- enable bit for G function
L	90			
T	"DB_UDT_M1".MDIB.G_1_VAL		DB1.DBB180	-- value G function group 1
S	"DB_UDT_M1".MDIB.X_T_EN		DB1.DBX178.4	-- enable bit for position / dwell time
L	"Counter_vl11"		MD162	
T	"DB_UDT_M1".MDIB.X_T_VAL		DB1.DBD184	-- value position / dwell ti me
S	"DB_UDT_M1".MDIB.V_EN		DB1.DBX179.0	-- enable bit for velocity
L	L#25000	//EXEMPLO 30000 CORRESPONDE A 300 DE GRAUS P MINUTO		
T	"DB_UDT_M1".MDIB.V_VAL		DB1.DBD188	-- value velocity
S	"DB_UDT_M1".MDI_EN	// Set request bit for MDI block	DB1.DBX38.3	-- MDI movement block
JU	MA01			
MDST: A	"DB_UDT_M1".DATA_ERR		DB1.DBX22.4	-- data-error
JC	MA01			
A	"contlautol"		C15	
L	S5T#250MS			
AN	"templautol"		T10	

Page 3 of 5

SD	"tem2autol"		T11	
A	"tem2autol"		T11	
SD	"templautol"		T10	
FN	"mem2autol"		M253.0	
S	"m_auto_mp_m1"		M27.0	
=	"mem3autol"		M253.1	
A	"mem3autol"	// para arrancar temp FC6	M253.1	
S	"DBEX_100_M1".EX2.START	// Start	DB100.DBX14.0	-- Start
S	"DB_UDT_M1".START	// Start signal in user DB	DB1.DBX15.0	-- start
R	"DBEX_100_M1".EX2.START		DB100.DBX14.0	-- Start
CD	"contlautol"	// mete o contador lá de cima a zero	C15	
A	"stop_m"		M7.0	
S	"DBEX_100_M1".EX2.STOP	// Stop	DB100.DBX14.1	-- Stop
=	"DB_UDT_M1".STOP	// Stop signal in user DB	DB1.DBX15.1	-- stop

Network: 5

A	"mem_auto"		M248.1	
A(// para ir para zero só qd passar o tempo da busca		
O	"zero pulse"		M245.1	
O	"m_auto_pi_m1"		M27.1	
)				
JNB	_001			
L	L#0			
T	"Counter_vl11"		MD162	
L	0			
T	"mw_volt"		MW100	
SET				
SAVE				
CLR				
_001: A	BR			
=	"mem_aux_fc102"	// mudar S para =	M6.2	

Network: 6

```

A   "mem_aux_fc102"  M6.2
A   "mem_auto"      M248.1
JC  AAA

```

Network: 7 OPERATOR/TRAVERSING ERROR; ACKNOWLEDGEMENT

```

MA01: AN  "DB_UDT_M1".OT_ERR      // If no operator/traversing error present, .. DB1.DBX22.3      -- ot-error
R        "DBEX_100_M1".OT_ERR_A  // .. reset error acknowledgement      DB100.DBX9.2    -- Operator/traversing error
                                                acknow.
A        "DBEX_100_M1".OT_ERR_A  // Error acknowledgement              DB100.DBX9.2    -- Operator/traversing error
                                                acknow.
=        "DB_UDT_M1".OT_ERR_A    // Entry in userDB                    DB1.DBX14.3     -- quit error

END: NOP  0

```

Page 5 of 5

FC103 - <offline>

```

"SUB_PROG_M1_3"      Varrimento (motor 1)
Name:  AUTO          Family: FM_ST_SV
Author: AuDMCE8      Version: 2.0
                          Block version: 2
Time stamp Code:    07/18/2012 06:24:59 PM
Interface:          02/24/2000 11:22:18 AM
Lengths (block/logic/data): 00376 00238 00002

```

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC103 Automatic

The block for this example is FC 103.
The signals are in "DBEX", the signals for Example 3 only are in structure "EX3".
The program to be selected in the example has the program number "10". This program number is entered in Example 3.
The read enable, the drive enable and the servo enable for the axis are set in "DBEX" (in OB 100: READ_EN=TRUE, DRV_EN=TRUE, SERVO_EN=TRUE), and are transferred to the interface in Example 3 (user DB).
Prerequisite for successful program selection is that the FM contain that program.
In order for the example to work, you have to set the "Automatic" mode.
Enter "Automatic" mode (code 08) in byte MODE_IN in the "DBEX". The relevant mode is indicated in the MODE_OUT byte.
Following successful mode selection, the program with the number "10" is automatically selected by setting Write request "PROGS_EN" in the user DB.
Set the "START" bit to TRUE in the "DBEX" in structure "EX3". The selected program is started if the axis is synchronized and has a start enable. The "START" bit is then reset.
The program can be stopped via the "STOP" bit. It cannot be started again until the "STOP" bit has been reset to FALSE (and the "START" bit is TRUE).

(Anexo M) 69

Page 1 of 4

If an operator error or traversing error occurs, it is flagged in the "OT_ERR" bit (bit = TRUE).
The error can be acknowledged by setting the "OT_ERR_A" bit to TRUE.
Note:
The Blocks directory contains variable table 3 (VAT3), which in turn contains all signals needed to monitor and control Example 3 (Tool "Monitor and control variables" tool).

Network: 1 QUERY MODE

```
CALL "VOLT" //COLOCADO PARA EFECTUAR MECIÇÃO DE TENSÃO FC4 -- Determinação do valor de po
tencia maxima e ângulos dos eixos (modo research)
CALL "MICRO" FC9 -- Varrimentos solares, com ou
sem sinal do sistema secundário

L B#16#8 // Code for Automatic mode
L "DBEX_100_M1".MODE_OUT // Mode checkback DB100.DBB8 -- Mode status (codiert)
==I
JCN END // If Automatic mode not active, END
```

Network: 2

```
A "mem_research" M248.0
L C#1
S "cont_aux_fc103" C30
R "mem_aux2_fc103" M252.1
R "m_auto_mp_m1" M27.0
```

Network: 3 SET DEFAULTS IN THE USER DATA BLOCK

```
A "DBEX_100_M1".DRV_EN // Drive enable ? DB100.DBX9.0 -- Drive enable
= "DB_UDT_M1".DRV_EN // Entry in user DB DB1.DBX15.7 -- drive enable

A "DBEX_100_M1".EX3.READ_EN // Read enable DB100.DBX16.2 -- Read enable
= "DB_UDT_M1".READ_EN // Entry in user DB DB1.DBX15.5 -- read enable

A "DBEX_100_M1".SERVO_EN // Servo enable ? DB100.DBX9.1 -- Servo enable
= "DB_UDT_M1".SERVO_EN // Entry in user DB DB1.DBX34.0 -- servo enable
```

Page 2 of 4

```
L "DBEX_100_M1".OVERRIDE // Default = 100% DB100.DBB6 -- Override
T "DB_UDT_M1".OVERRIDE DB1.DBB18 -- override
```

Network: 4 AUTOMATIC

Start de uma busca
A partir das saída do Sistema Auxiliar seleccionou-se uma busca
Cada busca corresponde a um (armazenado na FM 353)
DB100x (x = nº do "traversing program")

```
A "DB_UDT_M1".PROGS_D // Transfer program and DB1.DBX49.5 -- part program selection
JC AUST

L "m byte db" MB90
T "DB_UDT_M1".PROG_NO DB1.DBB242 -- part program number
S "DB_UDT_M1".PROGS_EN // Set request bit for program selection DB1.DBX39.5 -- part program selection
JU MA01

AUST: A "DB_UDT_M1".DATA_ERR // Data error ? DB1.DBX22.4 -- data-error
JC MA01 // If data error, no start

A "cont_aux_fc103" // Início de código auxiliar C30
L S5T#250MS
AN "temp_aux1_fc103" T30
SD "temp_aux2_fc103" T31
A "temp_aux2_fc103" T31
SD "temp_aux1_fc103" T30
FN "mem_aux1_fc103" M252.0
= "mem_aux2_fc103" M252.1
A "mem_aux2_fc103" M252.1
S "DBEX_100_M1".EX3.START // Fim de código auxiliar // Start DB100.DBX16.0 -- Start
S "DB_UDT_M1".START // Start signal in user DB DB1.DBX15.0 -- start
R "DBEX_100_M1".EX3.START // Start DB100.DBX16.0 -- Start
CD "cont_aux_fc103" C30

CALL "CLEAR" FC5 -- Limpa a potência máxima e
respectivos ângulos. Reset aos DBX's do modo MDI

A "stop_m" M7.0
S "DBEX_100_M1".EX3.STOP // Stop DB100.DBX16.1 -- Stop
= "DB_UDT_M1".STOP // Stop signal in user DB DB1.DBX15.1 -- stop
```

Network: 5 OPERATOR/TRAVERSING ERROR; ACKNOWLEDGEMENT			
MA01: AN	"DB_UDT_M1".OT_ERR	// Operator/traversing error ?	DB1.DBX22.3 -- ot-error
R	"DBEX_100_M1".OT_ERR_A	// If no error, reset acknowledgement	DB100.DBX9.2 -- Operator/traversing error acknow.
A	"DBEX_100_M1".OT_ERR_A	// Error acknowledgement	DB100.DBX9.2 -- Operator/traversing error acknow.
=	"DB_UDT_M1".OT_ERR_A	// Entry in user DB	DB1.DBX14.3 -- quit error
END: NOP	0		

FC200 - <offline>

"PROG_M2" Função principal FM353 (motor2)
Name: MODE_EX **Family:** FM_ST_SV
Author: AuDMCE8 **Version:** 2.0
Block version: 2
Time stamp Code: 05/01/2009 12:29:48 PM
Interface: 02/24/2000 11:22:02 AM
Lengths (block/logic/data): 00296 00194 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC200 Mode select

This example is always required for Examples 1 to 3.
The block for this example is FC 100.
The signals are in "DBEX".
This example must always be called. It sets the desired modes, evaluates the mode checkback, and displays the current mode.
The checkback signals required for the examples are copied to "DBEX".
In order to be able to work with Example 1 in "Jog" or "Reference point approach" mode, the user must enter the relevant code in the MODE_IN byte in "DBEX" (01 for "Jog" mode, 03 for "Reference point approach" mode).
If "Jog" mode is selected, mode parameter 01 (MODE_TYPE) is also set to activate velocity level 1 for "Jogging".
Mode Code
Jog 01
Ref.point approach 03
MDI 06
Automatic 08
In Example 2 you must set "MDI" mode (code in byte MODE_IN = 06).

In Example 3 you must set "Automatic" mode (code in byte MODE_IN = 08).
 The code in byte MODE_OUT shows the mode that is currently active.
 In order to restart the module (e.g. following a diagnostic interrupt),
 the RESET_AX bit has to be set in "DBEX". The example then sets the
 RESET_AX bit in the "user DB", the module is restarted, and the RESET_AX
 bit in "DBEX" is once again reset.
 In order to be able to work with any example, you must set the mode required
 for the example you want to use.

Network: 1 SETTING THE MODE

```

A      "DB_UDT_M2".WORKING_M2    // When work in progress, -> no mode s.      DB2.DBX23.1    -- working
JC     MDOU

L      B#16#0
L      "DBEX_200_M2".MODE_IN_M2 // Mode selection                      DB200.DBB7     -- Mode setting (coded)
<>I
JC     MDIN
L      B#16#1                    // If no mode entered, select Jog
MDIN: T  "DB_UDT_M2".MODE_IN_M2 // Entry in user DB                      DB2.DBB16     -- mode

L      "DB_UDT_M2".MODE_IN_M2    // Is Jog mode activated ?              DB2.DBB16     -- mode
L      B#16#1
<>I
JC     MDOU
L      B#16#1                    // Enter mode parameter 1
T      "DB_UDT_M2".MODE_TYPE_M2 // in user DB                          DB2.DBB17     -- mode parameter

MDOU: L  "DB_UDT_M2".MODE_OUT_M2 // Mode checkback                      DB2.DBB24     -- mode
T      "DBEX_200_M2".MODE_OUT_M2 // Entry in DBEX                       DB200.DBB8    -- Mode status (codiert)

```

Network: 2 FM-RESTART

```

A      "DBEX_200_M2".RESET_AX_M2 // Restart ?                            DB200.DBX9.3   -- Restart
S      "DB_UDT_M2".RESET_AX_M2 // Enter Restart in DB_FM and          DB2.DBX37.5   -- reset axis
R      "DBEX_200_M2".RESET_AX_M2 // reset it in DBEX                    DB200.DBX9.3   -- Restart

```

Page 2 of 3

Network: 3 COPY CHECKBACK SIGNALS TO 'DBEX'

```

A      "DB_UDT_M2".GO_M_M2        // Travel -                            DB2.DBX25.2    -- go_minus
=      "DBEX_200_M2".GO_M_M2      DB200.DBX10.2  -- Travel -
A      "DB_UDT_M2".GO_P_M2        // Travel +                            DB2.DBX25.3    -- go_plus
=      "DBEX_200_M2".GO_P_M2      DB200.DBX10.3  -- Travel +

A      "DB_UDT_M2".OT_ERR_M2      // Operator/traversing error           DB2.DBX22.3    -- ot-error
=      "DBEX_200_M2".OT_ERR_M2    DB200.DBX10.4  -- Operator/traversing erro
r

A      "DB_UDT_M2".DATA_ERR_M2    // Data error                          DB2.DBX22.4    -- data-error
=      "DBEX_200_M2".DATA_ERR_M2  DB200.DBX10.5  -- Data error

A      "DB_UDT_M2".PARA_M2        // Channel initialized                 DB2.DBX22.7    -- parameterized
=      "DBEX_200_M2".PARA_M2      DB200.DBX9.5   -- Parameterized

A      "DB_UDT_M2".SYNC_M2        // synchronized                        DB2.DBX25.0    -- synchronized
=      "DBEX_200_M2".SYNC_M2      DB200.DBX9.6   -- Synchronized
A      "DB_UDT_M2".ST_ENBLD_M2    // Start enable                        DB2.DBX23.0    -- start enable
=      "DBEX_200_M2".START_EN_M2  DB200.DBX9.7   -- Start enable
A      "DB_UDT_M2".POS_RCD_M2     // Position reached, stop              DB2.DBX25.7    -- position is reached, hol
d
=      "DBEX_200_M2".POS_RCD_M2    DB200.DBX10.0  -- Position reached, stop
A      "DB_UDT_M2".WORKING_M2     // Machining in progress               DB2.DBX23.1    -- working
=      "DBEX_200_M2".WORKING_M2   DB200.DBX10.1  -- Machining in progress

```

FC201 - <offline>

"SUB_PROG_M2_1" Ref. Pt. Approach (motor 1)
Name: TIP_REF **Family:** FM_ST_SV
Author: AuDMCE8 **Version:** 2.1
Block version: 2
Time stamp Code: 07/18/2012 06:52:19 PM
Interface: 02/24/2000 11:22:46 AM
Lengths (block/logic/data): 00326 00208 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC201 Jog and Reference point approach

The block for this example is FC 101.
The signals are in "DBEX", the signals for Example 1 only are in structure "EX1".
The drive enable and the servo enable for the axis are set in "DBEX" (in OB100: DRV_EN = TRUE, SERVO_EN = TRUE) and are transferred to the interface in Example 1 (user DB).
In order for the example to work, either "Jog" mode (mode code 01) or "Reference point approach" mode (mode code 03) must have been entered in byte MODE_IN in "DBEX". The relevant mode checkback is shown in byte MODE_OUT.
The traversing movements are flagged in bits "GO_M" = TRUE (axis motion -) or "GO_P" = TRUE (axis 1 motion +).
"Jog" mode active:
When a mode is activated, the Write command "VLEV_EN" (user DB, transfer velocity levels 1, 2) is executed automatically. If retransfer is required, you must either reset the "VLEV_D" bit (job status/checkback) or set the "JOBRESET" bit (reset status/error) in the user DB.
If you set the "DIR_M" (motion -) or "DIR_P" (motion +) bit to = TRUE

Page 1 of 4

in "DBEX", the axis will traverse in either a negative or positive direction.
"Reference point approach" mode active:
When the "START" bit is set to TRUE, the axis is moved in a negative or positive direction (depending on machine data parameterization) until the reference point is located. Following successful reference point approach, the axis is synchronized (SYNC=TRUE).
If an operator error or traversing error occurred, it is flagged in the "OT_ERR" bit (bit = TRUE).
The error can be acknowledged by setting the "OT_ERR_A" bit to TRUE.
Note:
The "Blocks" directory contains variable table 1 (VAT1), which in turn contains all signals needed to monitor and control Example 1 ("Monitor and control variables" tool).

Network: 1 QUERY MODE

```

L   B#16#1           // Code for Jog mode
L   "DBEX_200_M2".MODE_OUT_M2 // Mode checkback                DB200.DBB8      -- Mode status (codiert)
==I
JC  DRV             // Jump if Jog mode active
L   B#16#3           // Code for ref. point approach mode
==I
JC  DRV             // Jump if ref. point approach mode active
JU  END             // If these modes are not active -> End

```

Network: 2 SETTING DEFAULTS IN THE USER DATA BLOCK

```

DRV: A   "DBEX_200_M2".DRV_EN_M2 // Drive enable ?                DB200.DBX9.0   -- Drive enable
      =   "DB_UDT_M2".DRV_EN_M2 // Entry in user DB                DB2.DBX15.7   -- drive enable

A   "DBEX_200_M2".SERVO_EN_M2 // Servo enable ?                DB200.DBX9.1   -- Servo enable
      =   "DB_UDT_M2".SERVO_EN_M2 // Entry in user DB                DB2.DBX34.0   -- servo enable

L   "DBEX_200_M2".OVERRIDE_M2 // Default = 100%                DB200.DBB6     -- Override
T   "DB_UDT_M2".OVERRIDE_M2 // Entry in user DB                DB2.DBB18     -- override

```

Network: 3		JOG / REFERENCE POINT APPROACH		
L	B#16#1	// Code for Jog mode		
L	"DBEX_200_M2".MODE_OUT_M2	// Mode checkback	DB200.DBB8	-- Mode status (codiert)
==I				
JC	JOG	// Jump if Jog mode active		
L	B#16#3	// Code for reference point approach mode		
==I				
JC	REFP	// Jump if ref. point approach mode active		
JU	END	// If modes not active -> End		
JOG: A	"DB_UDT_M2".VLEV_D_M2	// Transfer velocity levels and	DB2.DBX48.0	-- velocity levels 1 and 2
JC	DIR			
L	"VLEVEL1_M2_OP"		MD19	
T	"DB_UDT_M2".VLEVEL_1_M2	// Value for velocity level 1	DB2.DBD160	-- velocity level 1
L	L#80000			
T	"DB_UDT_M2".VLEVEL_2_M2	// Value for velocity level 2	DB2.DBD164	-- velocity level 2
S	"DB_UDT_M2".VLEV_EN_M2	// Transfer velocity levels	DB2.DBX38.0	-- velocity levels 1 and 2
JU	MA01			
DIR: A	"TRAVEL_M_M2_m"	//A "DBEX_200".EX1_M2.DIR_M_M2 Travel - c	M7.4	
=	"DB_UDT_M2".DIR_M_M2	ommand		
A	"TRAVEL_P_M2_m"	//A "DBEX_200".EX1_M2.DIR_P_M2 // Travel +	DB2.DBX15.2	-- direction minus
=	"DB_UDT_M2".DIR_P_M2	command	M7.5	
JU	MA01		DB2.DBX15.3	-- direction plus
REFP: A	"DBEX_200_M2".EX1_M2.START_M2	// Start ?	DB200.DBX12.2	-- Start
S	"DB_UDT_M2".START_M2	// Entry in user DB	DB2.DBX15.0	-- start
R	"DBEX_200_M2".EX1_M2.START_M2	// Start of reset	DB200.DBX12.2	-- Start
A	"stop_m"		M7.0	
S	"DBEX_200_M2".EX1_M2.STOP_M2	// Stop ?	DB200.DBX12.3	-- Stop
=	"DB_UDT_M2".STOP_M2	// Entry in user DB	DB2.DBX15.1	-- stop

Page 3 of 4

Network: 4		OPERATOR/TRAVERSING ERROR; ACKNOWLEDGEMENT		
MA01: AN	"DB_UDT_M2".OT_ERR_M2	// If no operator/traversing error present, ..	DB2.DBX22.3	-- ot-error
R	"DBEX_200_M2".OT_ERR_A_M2	// .. reset error acknowledgement	DB200.DBX9.2	-- Operator/traversing error acknow.
A	"DBEX_200_M2".OT_ERR_A_M2	// Error acknowledgement	DB200.DBX9.2	-- Operator/traversing error acknow.
=	"DB_UDT_M2".OT_ERR_A_M2	// Enter error acknowledgement in user DB	DB2.DBX14.3	-- quit error
END: NOP	0			

FC202 - <offline>

"SUB_PROG_M2_2" Deslocamento para posição de máx. pot. (motor 2)
Name: MDI **Family:** FM_ST_SV
Author: AuDMCE8 **Version:** 2.0
Block version: 2
Time stamp Code: 07/18/2012 06:28:42 PM
Interface: 02/24/2000 11:22:32 AM
Lengths (block/logic/data): 00458 00332 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC202 MDI

The block for this example is FC 102.
The signals are in "DBEX", the signals for Example 2 only are in structure "EX2".
The drive enable and servo enable signals for the axis are set in "DBEX" (in QB100: DRV_EN = TRUE, SERVO_EN = TRUE), and are transferred to the interface in Example 2 (user DB).
In order for the example to work, you have to set the "MDI" mode. Enter "MDI" mode (code 06) in the MODE_IN byte in "DBEX". The relevant mode status is signalled in the MODE_OUT byte.
Following successful mode selection, a default MDI block is automatically transferred to the module (network MDI) by setting Write request "MDI_EN" in the user DB (transfer MDI block).
This block is modifiable in dependence on the system or the request.
To retransfer the block, you must either reset the "MDI_D" bit (job status/checkback)
or set the "JOBRESET" bit (reset status/error) in the user DB.
Set the "START" bit in the "DBEX" in structure "EX2" to TRUE. The activated MDI block is started if the axis is synchronized and has a start enable.
The "START" bit is then reset.
The MDI block cannot be restarted until the start enable signal is once

Page 1 of 5

again present.
The block can be stopped via the "STOP" bit. It cannot be started again until the "STOP" bit has been reset to FALSE (and "START" is = TRUE).
An operator error or traversing error, if any, is flagged in the "OT_ERR" bit (bit = TRUE).
The error can be acknowledged by setting the "OT_ERR_A" bit to TRUE.
Note:
The "Blocks" directory contains variable table 2 (VAT2), which in turn contains all signals needed to monitor and control Example 2 (Tool "Monitor and control variables" tool).

Network: 1

```

L   B#16#6           // Code for MDI mode
L   "DBEX_200_M2".MODE_OUT_M2 // Mode checkback           DB200.DBB8      -- Mode status (codiert)
==I
JCN  END             // If MDI mode not active, END

```

Network: 2

```

A(
O   "auto pulse"    M245.0
O   "zero pulse"    M245.1
O   "m_uato_pi_m2"  M27.3
)
L   C#1
S   "contlauto2"    C16

```

Network: 3 SETTING DEFAULTS IN THE USER DATA BLOCK

```

A   "DBEX_200_M2".DRV_EN_M2 // Drive enable ?           DB200.DBX9.0    -- Drive enable
=   "DB_UDT_M2".DRV_EN_M2   // Entry in user DB           DB2.DBX15.7     -- drive enable
A   "DBEX_200_M2".SERVO_EN_M2 // Servo enable ?           DB200.DBX9.1    -- Servo enable
=   "DB_UDT_M2".SERVO_EN_M2 // Entry in user DB           DB2.DBX34.0     -- servo enable
L   "DBEX_200_M2".OVERRIDE_M2 // Default = 100%           DB200.DBB6      -- Override
T   "DB_UDT_M2".OVERRIDE_M2 // Entry in user DB           DB2.DBB18       -- override

```

(Anexo M) 75

Page 2 of 5

Network: 4		MDI		
A	"DB_UDT_M2".MDI_D_M2	// Transfer MDI block and	DB2.DBX48.3	-- MDI movement block
JC	MDST			
JCN	AAA			
AAA: R	"DB_UDT_M2".MDIB_M2.G_2_EN_M2	// Delete MDI block areas	DB2.DBX178.1	-- enable bit for G funct
R	"DB_UDT_M2".MDIB_M2.M_1_EN_M2		ion group 2	-- enable bit for M funct
R	"DB_UDT_M2".MDIB_M2.M_2_EN_M2		DB2.DBX179.1	-- enable bit for M funct
R	"DB_UDT_M2".MDIB_M2.M_3_EN_M2		ion group 1	-- enable bit for M funct
L	B#16#0		DB2.DBX179.2	-- enable bit for M funct
T	"DB_UDT_M2".MDIB_M2.G_2_VAL_M2		ion group 2	-- enable bit for M funct
T	"DB_UDT_M2".MDIB_M2.M_1_VAL_M2		DB2.DBX179.3	-- value G function group
T	"DB_UDT_M2".MDIB_M2.M_2_VAL_M2		ion group 3	-- value M function group
T	"DB_UDT_M2".MDIB_M2.M_3_VAL_M2		DB2.DBB181	-- value M function group
R	"mem1auto2"		2	-- value M function group
S	"DB_UDT_M2".MDIB_M2.G_1_EN_M2	// Enter MDI block (example)	DB2.DBB192	-- value M function group
L	90		1	-- value M function group
T	"DB_UDT_M2".MDIB_M2.G_1_VAL_M2		DB2.DBB193	-- value M function group
S	"DB_UDT_M2".MDIB_M2.X_T_EN_M2		2	-- value M function group
L	"Counter_v222"		DB2.DBB194	-- value M function group
T	"DB_UDT_M2".MDIB_M2.X_T_VAL_M2		3	-- value M function group
S	"DB_UDT_M2".MDIB_M2.V_EN_M2		M246.2	-- enable bit for G funct
L	L#30000		DB2.DBX178.0	-- enable bit for G funct
T	"DB_UDT_M2".MDIB_M2.V_VAL_M2		ion group 1	-- enable bit for G funct
S	"DB_UDT_M2".MDI_EN_M2	// Set request bit for MDI block	DB2.DBB180	-- value G function group
JU	MA01		1	-- value G function group
MDST: A	"DB_UDT_M2".DATA_ERR_M2	// Data error ?	DB2.DBX178.4	-- enable bit for positio
			n / dwell time	-- value position / dwell
			MD170	-- value position / dwell
			DB2.DBD184	-- value position / dwell
			time	-- value position / dwell
			DB2.DBX179.0	-- enable bit for velocit
			y	-- enable bit for velocit
			DB2.DBD188	-- value velocity
			DB2.DBX38.3	-- MDI movement block

Page 3 of 5

JC	MA01	// No start in case of data error		
A	"cont1auto2"		C16	
L	S5T#250MS			
AN	"temp1auto2"		T12	
SD	"temp2auto2"		T13	
A	"temp2auto2"		T13	
SD	"temp1auto2"		T12	
FN	"mem2auto2"		M254.0	
S	"m_auto_mp_m2"		M27.2	
=	"mem3auto2"		M254.1	
A	"mem3auto2"		M254.1	
S	"DBEX_200_M2".EX2_M2.START_M2	// Start	DB200.DBX14.0	-- Start
S	"DB_UDT_M2".START_M2	// Start signal in user DB	DB2.DBX15.0	-- start
R	"DBEX_200_M2".EX2_M2.START_M2		DB200.DBX14.0	-- Start
CD	"cont1auto2"		C16	
A	"stop_m"		M7.0	
S	"DBEX_200_M2".EX2_M2.STOP_M2	// Stop	DB200.DBX14.1	-- Stop
=	"DB_UDT_M2".STOP_M2	// Stop signal in user DB	DB2.DBX15.1	-- stop

Network: 5

A	"mem_auto"	M248.1
A("zero pulse"	M245.1
O	"m_uato_pi_m2"	M27.3
O	"m_uato_pi_m2"	M27.3
)		
JNB	_001	
L	L#0	
T	"Counter_v222"	MD170
L	0	
T	"mw_volt"	MW100
SET		
SAVE		
CLR		
_001: A	BR	
=	"mem_aux_fc202"	M6.3

Network: 6

```

A   "mem_aux_fc202"  M6.3
A   "mem_auto"      M248.1
JC  AAA

```

Network: 7 OPERATOR/TRAVERSING ERROR; ACKNOWLEDGEMENT

```

MA01: AN  "DB_UDT_M2".OT_ERR_M2 // If no operator/traversing error present, .. DB2.DBX22.3 -- ot-error
R        "DBEX_200_M2".OT_ERR_A_M2 // .. reset error acknowledgement DB200.DBX9.2 -- Operator/traversing erro
                                                r acknow.
A        "DBEX_200_M2".OT_ERR_A_M2 // Error acknowledgement DB200.DBX9.2 -- Operator/traversing erro
=        "DB_UDT_M2".OT_ERR_A_M2 // Entry in userDB DB2.DBX14.3 -- quit error

END: NOP 0

```

Page 5 of 5

FC203 - <offline>

```

"SUB_PROG_M2_3"      Varrimento (motor 2)
Name: AUTO           Family: FM_ST_SV
Author: AuDMCE8      Version: 2.0
                    Block version: 2
Time stamp Code:    07/18/2012 06:31:09 PM
Interface:          02/24/2000 11:22:18 AM
Lengths (block/logic/data): 00362 00224 00002

```

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC203 Automatic

The block for this example is FC 103.
The signals are in "DBEX", the signals for Example 3 only are in structure "EX3".
The program to be selected in the example has the program number "10". This program number is entered in Example 3.
The read enable, the drive enable and the servo enable for the axis are set in "DBEX" (in OB 100: READ_EN=TRUE, DRV_EN=TRUE, SERVO_EN=TRUE), and are transferred to the interface in Example 3 (user DB).
Prerequisite for successful program selection is that the FM contain that program.
In order for the example to work, you have to set the "Automatic" mode.
Enter "Automatic" mode (code 08) in byte MODE_IN in the "DBEX". The relevant mode is indicated in the MODE_OUT byte.
Following successful mode selection, the program with the number "10" is automatically selected by setting Write request "PROGS_EN" in the user DB.
Set the "START" bit to TRUE in the "DBEX" in structure "EX3". The selected program is started if the axis is synchronized and has a start enable. The "START" bit is then reset.
The program can be stopped via the "STOP" bit. It cannot be started again until the "STOP" bit has been reset to FALSE (and the "START" bit is TRUE).

(Anexo M) 77

Page 1 of 4

If an operator error or traversing error occurs, it is flagged in the "OT_ERR" bit (bit = TRUE).
The error can be acknowledged by setting the "OT_ERR_A" bit to TRUE.
Note:
The Blocks directory contains variable table 3 (VAT3), which in turn contains all signals needed to monitor and control Example 3 (Tool "Monitor and control variables" tool).

Network: 1 QUERY MODE

```
CALL "MICRO"                                FC9                -- Varrimentos solares, com
                                                ou sem sinal do sistema secundário
L      B#16#8                                // Code for Automatic mode
L      "DBEX_200_M2".MODE_OUT_M2            // Mode checkback                DB200.DBB8        -- Mode status (codiert)
==I
JCN   END                                    // If Automatic mode not active, END
```

Network: 2

```
A      "mem_research"    M248.0
L      C#1
S      "cont_aux_fc203"  C25
R      "mem_aux2_fc203"  M251.1
R      "m_auto_mp_m2"    M27.2
```

Network: 3 SET DEFAULTS IN THE USER DATA BLOCK

```
A      "DBEX_200_M2".DRV_EN_M2              // Drive enable ?                DB200.DBX9.0     -- Drive enable
=      "DB_UDT_M2".DRV_EN_M2                // Entry in user DB              DB2.DBX15.7     -- drive enable

A      "DBEX_200_M2".EX3_M2.READ_EN_M2     // Read enable                    DB200.DBX16.2   -- Read enable
=      "DB_UDT_M2".READ_EN_M2              // Entry in user DB              DB2.DBX15.5     -- read enable

A      "DBEX_200_M2".SERVO_EN_M2           // Servo enable ?                 DB200.DBX9.1    -- Servo enable
=      "DB_UDT_M2".SERVO_EN_M2             // Entry in user DB              DB2.DBX34.0     -- servo enable

L      "DBEX_200_M2".OVERRIDE_M2           // Default = 100%                 DB200.DBB6      -- Override
T      "DB_UDT_M2".OVERRIDE_M2             //                                DB2.DBB18      -- override
```

Page 2 of 4

Network: 4 AUTOMATIC

Start de uma busca
A partir das saída do Sistema Auxiliar seleccionou-se uma busca
Cada busca corresponde a um (armazenado na FM 353)
DB100x (x = nº do "traversing program")

```
A      "DB_UDT_M2".PROGS_D_M2              // Transfer program and          DB2.DBX49.5     -- part program selection
JC     AUST
L      "m_byte_db_m2"                      MB91
T      "DB_UDT_M2".PROG_NO_M2              DB2.DBB242      -- part program number
S      "DB_UDT_M2".PROGS_EN_M2            // Set request bit for program selection DB2.DBX39.5     -- part program selection
JU     MA01

AUST: A      "DB_UDT_M2".DATA_ERR_M2        // Data error ?                  DB2.DBX22.4     -- data-error
JC     MA01                                // If data error, no start

A      "cont_aux_fc203"                    C25
L      S5T#250MS
AN     "temp_aux1_fc203"                   T25
SD     "temp_aux2_fc203"                   T26
A      "temp_aux2_fc203"                   T26
SD     "temp_aux1_fc203"                   T25
FN     "mem_aux1_fc203"                    M251.0
=      "mem_aux2_fc203"                    M251.1
A      "mem_aux2_fc203"                    M251.1

S      "DBEX_200_M2".EX3_M2.START_M2      DB200.DBX16.0   -- Start
S      "DB_UDT_M2".START_M2               DB2.DBX15.0     -- start
R      "DBEX_200_M2".EX3_M2.START_M2      DB200.DBX16.0   -- Start
CD     "cont_aux_fc203"                    C25

CALL  "CLEAR"                                FC5                -- Limpa a potência máxima
                                                e respectivos ângulos. Reset aos DBX's do mo
                                                do MDI

A      "stop_m"                             M7.0
S      "DBEX_200_M2".EX3_M2.STOP_M2      DB200.DBX16.1   -- Stop
=      "DB_UDT_M2".STOP_M2                DB2.DBX15.1     -- stop
```

Network: 5		OPERATOR/TRAVERSING ERROR; ACKNOWLEDGEMENT	
MA01: AN	"DB_UDT_M2".OT_ERR_M2	// Operator/traversing error ?	DB2.DBX22.3 -- ot-error
R	"DBEX_200_M2".OT_ERR_A_M2	// If no error, reset acknowledgement	DB200.DBX9.2 -- Operator/traversing error acknow.
A	"DBEX_200_M2".OT_ERR_A_M2	// Error acknowledgement	DB200.DBX9.2 -- Operator/traversing error acknow.
=	"DB_UDT_M2".OT_ERR_A_M2	// Entry in user DB	DB2.DBX14.3 -- quit error
END: NOP	0		

Page 4 of 4

FC301 - <offline>

"DIAG_INF" Read diagnostic data Encoder 1
Name: DIAG_INF **Family:** FM_CNT_1
Author: FM **Version:** 3.0
Block version: 2
Time stamp Code: 10/15/2002 08:03:22 AM
Interface: 08/26/2002 02:41:29 PM
Lengths (block/logic/data): 00332 00210 00038

Name	Data Type	Address	Comment
IN		0.0	
DB_NO	Int	0.0	datablock
OUT		0.0	
IN_OUT		0.0	
IN_DIAG	Bool	4.0	initiate diagnostic interrupt
TEMP		0.0	
x_record	Any	0.0	"Zeiger" auf Zielbereich im DB
!!! immer LB0 bis LB9 !!!			
w_bgadr	Word	10.0	Baugruppemadresse
w_ret_val	Int	12.0	RET_VAL des SFCs
b_busy	Bool	14.0	BUSY-Ausgang des SFCs
dbnr	Word	16.0	DB-Nr. als Lokaldatum
rett_ar1	DWord	18.0	Zwischenspeicher für AR1
rett_dbnr	Word	22.0	Nummer des DB, der beim Aufruf des FC gerade offen war (von AWL benutzt)
s_szl_header	Struct	24.0	SZL-Header für SFC51
len	Word	24.0	
n_dr	Word	26.0	
RETURN		0.0	
RET_VAL	Int	2.0	

Block: FC301 DIAGNOSTIC INTERRUPT INFORMATION

Network: 1 Anlaufkennung prüfen

```

SET
SAVE          // BIE = 1 setzen

TAR1 #rett_ar1 // AR1 retten                                #rett_ar1    -- Zwischenspeicher für AR1

L   #DB_NO     // Übergebene Nummer in dbnr zwischenspeichern #DB_NO       -- datablock
T   #dbnr     //                                     #dbnr       -- DB-Nr. als Lokaldatum
L   DBNO      // Lade Nummer Global-DB in AKKU 1
T   #rett_dbnr // und rette sie in rett_dbnr                #rett_dbnr   -- Nummer des DB, der beim Aufruf d
es FC gerade offen war (von AWL benutzt)

OPN DB [#dbnr] // Instanz-DB öffnen                          #dbnr       -- DB-Nr. als Lokaldatum
AN  DBX 41.5  // FMNEUSTQ_Rückmeldung
A   DBX 41.6  // FMNEUST_Rückmeldung
BEC // *** Bausteinende ***

```

Network: 2 Auftragsbearbeitung

```

AN  #IN_DIAG                                #IN_DIAG     -- initiate diagnostic interrupt
L   B#16#0
JC  RETV

// SFC RD_REC aufrufen, wenn IN_DIAG = 1
L   DBW 6 // Baugruppenadresse
T   #w_bgadr                                #w_bgadr     -- Baugruppemaschine

LAR1 P##x_record // Anfangsadresse des ANY-Pointers

L   W#16#1002 // Syntax-ID und Typ
T   LW [AR1,P#0.0]
L   B#16#10 // DS-Länge = 12 Byte
T   LW [AR1,P#2.0]
L   #DB_NO // DB-Nummer                                #DB_NO       -- datablock

```

Page 2 of 3

```

T   LW [AR1,P#4.0]
L   432 // Zeiger auf Anfang des Datensatzes
T   LD [AR1,P#6.0]
L   B#16#84 // Datenbaustein
T   LB [AR1,P#6.0]

CALL "RDSYSST" // Diagnosedaten von FM lesen                SFC51       -- Read a System Status List or
Partial List

REQ      :=TRUE
SZL_ID   :=W#16#B3
INDEX    :=#w_bgadr                                #w_bgadr     -- Baugruppemaschine
RET_VAL  :=#w_ret_val                               #w_ret_val   -- RET_VAL des SFCs
BUSY     :=#b_busy                                  #b_busy     -- BUSY-Ausgang des SFCs
SZL_HEADER:=#s_szl_header                          #s_szl_header -- SZL-Header für SFC51
DR       :=#x_record // *** ANY-Pointers auf Zielbereich #x_record    -- "Zeiger" auf Zielbereich im
DB !!! immer LB0 bis LB9 !!!

OPN DB [#dbnr] // DB-Nr. als Lokaldatum                    #dbnr       -- DB-Nr. als Lokaldatum

AN  #b_busy // DS-Übertragung abgeschlossen                #b_busy     -- BUSY-Ausgang des SFCs
R   #IN_DIAG // --> Anstoßparameter rücksetzen            #IN_DIAG     -- initiate diagnostic interrupt

L   #w_ret_val // Der Returnwert des SFCs wird an den #w_ret_val   -- RET_VAL des SFCs
RETV: T #RET_VAL // Ausgangsparameter RET_VAL durchgereicht. #RET_VAL
L   B#16#0 // BIE bilden
>=I // BIE = 1 im fehlerfreien Fall
SAVE // BIE = 0 im Fehlerfall

OPN DB [#rett_dbnr] // alten vor Aufruf geöffneten DB wieder öffnen #rett_dbnr   -- Nummer des DB, der beim Aufruf
des FC gerade offen war (von AWL benutzt)
LAR1 #rett_ar1 // AR1 restaurieren                          #rett_ar1    -- Zwischenspeicher für AR1
BE

```

FC302 - <offline>

"CNT_CTL1_ENC1" Control counter module FM 350-1 Encoder 1
Name: CNT_CTL1 **Family:** FM_CNT_1
Author: FM **Version:** 3.0
Block version: 2
Time stamp Code: 10/16/2002 09:14:43 AM
Interface: 09/12/2002 09:01:30 AM
Lengths (block/logic/data): 00956 00760 00046

Name	Data Type	Address	Comment
IN		0.0	
DB_NO	Int	0.0	module datablock number
SW_GATE	Bool	2.0	1=start software gate
GATE_STP	Bool	2.1	1=gate stop
OT_ERR_A	Bool	2.2	1=operator error acknowledge
SET_DO0	Bool	2.3	1=set DO0
SET_DO1	Bool	2.4	1=set DO1
OUT		0.0	
OT_ERR	Bool	4.0	1=operator error
IN_OUT		0.0	
L_DIRECT	Bool	6.0	1=load direct / transfer lower limit
L_PREPAR	Bool	6.1	1=prepare load
T_CMP_V1	Bool	6.2	1=transfer comparator value 1 / transfer upper limit
T_CMP_V2	Bool	6.3	1=transfer comparator value 2 / transfer Aktualisierungszeit ????
C_DOPARA	Bool	6.4	1=change parameters
RES_SYNC	Bool	6.5	1=reset synchronisation bit
RES_ZERO	Bool	6.6	1=reset zero crossing bit
TEMP		0.0	
dbnr	Word	0.0	DB-Nr. als Lokaldatum
rett_dbnr	Word	2.0	Nummer des DB, der beim Aufruf des FC gerade offen war (von AWL benutzt)
SFCErr	Int	4.0	Rückgabewert des SFC6 (RD_SINFO) lokal, da keine relevante Info
Taktsynchron	Bool	6.0	1=Aufruf im taktsynchronen OB
TOP_SI	Struct	8.0	

Page 1 of 9

Name	Data Type	Address	Comment
EV_CLASS	Byte	8.0	
EV_NUM	Byte	9.0	
PRIORITY	Byte	10.0	
NUM	Byte	11.0	
TYP2_3	Byte	12.0	
TYP1	Byte	13.0	
ZI1	Word	14.0	
ZI2_3	DWord	16.0	
START_UP_SI	Struct	20.0	
EV_CLASS	Byte	20.0	
EV_NUM	Byte	21.0	
PRIORITY	Byte	22.0	
NUM	Byte	23.0	
TYP2_3	Byte	24.0	
TYP1	Byte	25.0	
ZI1	Word	26.0	
ZI2_3	DWord	28.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC302 COUNTER MODULE CONTROL

Network: 1

Zuordnung Parameter <==> DB Bit

TFB = A_BIT0_1 Testfreigabe
OT_ERR_A = A_BIT0_3 Bedienfehler quittieren
NEUSTQ = A_BIT0_6 Neustart quittieren
GATE_STP = A_BIT1_2 Torstopp
SW_GATE = A_BIT1_3 SW-Tor Start/Stopp
SET_DO0 = A_BIT2_2 Steuern Ausgang DO0
SET_DO1 = A_BIT2_3 Steuern Ausgang DO1

(Anexo M) 81

Page 2 of 9

```

L_DIRECT = A_BIT3_0      Zähler mit Ladewert laden / Untergrenze
übernehmen
L_PREPAR = A_BIT3_1      Ladewert übernehmen
T_CMP_V1 = A_BIT3_2      Vergleichswert 1 übernehmen / Obergrenze
übernehmen
T_CMP_V2 = A_BIT3_3      Vergleichswert 2 übernehmen /
Aktualisierungszeit übernehmen
RES_SYNC = A_BIT3_4      Synchronisationsbit rücksetzen
RES_ZERO = A_BIT3_5      alle Statusbits rücksetzen
C_DOPARA = A_BIT3_6      Parameteränderung übernehmen

OT_ERR = E_BIT0_3        Bedienfehler
FM-NEUSTQ = E_BIT0_5     FM-Neustart quittiert
FM-NEUST = E_BIT0_6     FM-Neustart
E_BIT3_0                Ladewert in Zähler übernommen
E_BIT3_1                Ladewert ins Laderegister übernommen
E_BIT3_2                Vergleichswert 1 übernommen
E_BIT3_3                Vergleichswert 2 übernommen
E_BIT3_4                Synchronisationsbit gelöscht
E_BIT3_5                Nulldurchgangsbit gelöscht

```

```

*****
Rückmeldungen durch Lesen von der FM aktualisieren
*****

```

```

SET
SAVE // BIE fest auf 1 !!!

L #DB_NO // Übergabene Nummer in dbnr zwischenspeichern #DB_NO -- module datablock number
T #dbnr // #dbnr -- DB-Nr. als Lokaldatum
L DBNO // Lade Nummer Global-DB in AKKU 1
T #rett_dbnr // und rette sie in rett_dbnr #rett_dbnr -- Nummer des DB, der beim Aufruf
des FC gerade offen war (von AWL benutzt)

OPN DB [#dbnr] // Öffne Kanal-DB mit Übergabener Nummer #dbnr -- DB-Nr. als Lokaldatum

TAR1 DBD 0 // Adressregister 1 retten

L DBD 8 // (CH_ADR) Kanalanzfangsadresse
LAR1 // in das Adressregister 1

// *****
// taktischer Betrieb (d.h. Aufruf in OB61..64)?
// *****

CALL "RD_SINFO" SFC6 -- Read OB Start Information

```

Page 3 of 9

```

RET_VAL :=#SFCErr #SFCErr -- Rückgabewert des SFC6 (RD_SI
NFO) lokal, da keine relevante Info
TOP_SI :=#TOP_SI #TOP_SI
START_UP_SI:=#START_UP_SI #START_UP_SI

L #TOP_SI.NUM #TOP_SI.NUM
L 61
>=I
TAK
L 64
A(
<=I
)
= #Taktisynchron #Taktisynchron -- 1=Aufruf im taktisynchronen O
B

// *****
// Neustart-Quittung
// *****

A #Taktisynchron #Taktisynchron -- 1=Aufruf im taktisynchronen O
B

JC TS1t
L PID [AR1,P#8.0] // Datenfehler, Bedienfehler und Koordinierungsbyte
JU TS1e
TS1t: L ID [AR1,P#8.0] // Datenfehler, Bedienfehler und Koordinierungsbyte
TS1e: T DBD 38

A DBX 41.6 // FM-NEUST (E_BIT0_6)
AN DBX 41.5 // FM-NEUSTQ (E_BIT0_5)
= DBX 26.6 // NEUSTQ (A_BIT0_6)
JCN OK // Kein Neustart

L DW#16#40000000 // Koordinierungsbyte schreiben
A #Taktisynchron #Taktisynchron -- 1=Aufruf im taktisynchronen O
B

JC TS2t
T PQD [AR1,P#12.0] // und Steuerbits löschen
JU TS2e
TS2t: T QD [AR1,P#12.0] // und Steuerbits löschen
TS2e: L B#16#0
T DBB 4 // (FP) Flankenmerker löschen
JU END // und Baustein verlassen

OK: A #Taktisynchron #Taktisynchron -- 1=Aufruf im taktisynchronen O
B

```

(Anexo M) 82

Page 4 of 9

```

JC    TS3t
L    PID [AR1,P#0.0] // (ACT_LOAD) Loadwert in den Kanal-DB
T    DBD 30
L    PID [AR1,P#4.0] // (ACT_CNTV) Zählstand in den Kanal-DB
T    DBD 34
L    PID [AR1,P#12.0] // Reserve und Statusbytes
JU    TS3e
TS3t: L    ID [AR1,P#0.0] // (ACT_LOAD) Loadwert in den Kanal-DB
      T    DBD 30
      L    ID [AR1,P#4.0] // (ACT_CNTV) Zählstand in den Kanal-DB
      T    DBD 34
      L    ID [AR1,P#12.0] // Reserve und Statusbytes
TS3e: T    DBD 42

L    DBB 12 // (U_D_LGTH) 16 : S7-300, 32 : S7-400
L    B#16#20
<I //
JC    L1

A    #Taktssynchron #Taktssynchron -- 1=Aufruf im taktsynchronen O
      B

JC    TS4t
L    PID [AR1,P#16.0] // Aktueller Vergleichswert 1
T    DBD 46 // (ACT_CMP1)
L    PID [AR1,P#20.0] // Aktueller Vergleichswert 2
JU    TS4e
TS4t: L    ID [AR1,P#16.0] // Aktueller Vergleichswert 1
      T    DBD 46 // (ACT_CMP1)
      L    ID [AR1,P#20.0] // Aktueller Vergleichswert 2
TS4e: T    DBD 50 // (ACT_CMP2)

// *****
// FB-Steuer-Parameter bitweise in den Datenbaustein übertragen
// *****

// Zähler mit Ladewert laden

L1: A    #L_DIRECT // FC-Parameter einlesen #L_DIRECT -- 1=load direct / transfer low
      er limit
      FP DBX 4.0 // 1. Anstoß
      S DBX 29.0 // (A_BIT3_0) Steuersignal setzen

      A DBX 45.0 // (E_BIT3_0) Rückmeldung erhalten
      R DBX 29.0 // (A_BIT3_0) Steuersignal löschen

      AN DBX 45.0 // (E_BIT3_0) Bearbeitung fertig
      AN DBX 29.0 // (A_BIT3_0)

```

Page 5 of 9

```

R    #L_DIRECT // FC-Parameter löschen #L_DIRECT -- 1=load direct / transfer low
      er limit
R    DBX 4.0 // Merker 1. Anstoß löschen

// Ladewert ins Laderegister

A    #L_PREPAR // FC-Parameter einlesen #L_PREPAR -- 1=prepare load
FP DBX 4.1 // 1. Anstoß
S DBX 29.1 // (A_BIT3_1) Steuersignal setzen

A DBX 45.1 // (E_BIT3_1) Rückmeldung erhalten
R DBX 29.1 // (A_BIT3_1) Steuersignal löschen

AN DBX 45.1 // (E_BIT3_1) Bearbeitung fertig
AN DBX 29.1 // (A_BIT3_1)
R #L_PREPAR // FC-Parameter löschen #L_PREPAR -- 1=prepare load
R DBX 4.1 // Merker 1. Anstoß löschen

// Vergleichswert 1 übernehmen

A    #T_CMP_V1 // FC-Parameter einlesen #T_CMP_V1 -- 1=transfer comparator value
      1 / transfer upper limit
FP DBX 4.2 // 1. Anstoß
S DBX 29.2 // (A_BIT3_2) Steuersignal setzen

A DBX 45.2 // (E_BIT3_2) Rückmeldung erhalten
R DBX 29.2 // (A_BIT3_2) Steuersignal löschen

AN DBX 45.2 // (E_BIT3_2) Bearbeitung fertig
AN DBX 29.2 // (A_BIT3_2)
R #T_CMP_V1 // FC-Parameter löschen #T_CMP_V1 -- 1=transfer comparator value
      1 / transfer upper limit
R DBX 4.2 // Merker 1. Anstoß löschen

// Vergleichswert 2 übernehmen

A    #T_CMP_V2 // FC-Parameter einlesen #T_CMP_V2 -- 1=transfer comparator value
      2 / transfer Aktualisierungszeit ???
FP DBX 4.3 // 1. Anstoß
S DBX 29.3 // (A_BIT3_3) Steuersignal setzen

A DBX 45.3 // (E_BIT3_3) Rückmeldung erhalten
R DBX 29.3 // (A_BIT3_3) Steuersignal löschen

AN DBX 45.3 // (E_BIT3_3) Bearbeitung fertig
AN DBX 29.3 // (A_BIT3_3)
R #T_CMP_V2 // FC-Parameter löschen #T_CMP_V2 -- 1=transfer comparator value
      2 / transfer Aktualisierungszeit ???

```

```

R   DBX   4.3           // Merker 1. Anstoß löschen

// Synchronisationsbit rücksetzen

A   #RES_SYNC          // FC-Parameter einlesen           #RES_SYNC          -- 1=reset synchronisation bit
FP  DBX   4.4           // 1. Anstoß
S   DBX   29.4          // (A_BIT3_4) Steuersignal setzen

A   DBX   45.4          // (E_BIT3_4) Rückmeldung erhalten
R   DBX   29.4          // (A_BIT3_4) Steuersignal löschen

AN  DBX   45.4          // (E_BIT3_4) Bearbeitung fertig
AN  DBX   29.4          // (A_BIT3_4)
R   #RES_SYNC          // FC-Parameter löschen           #RES_SYNC          -- 1=reset synchronisation bit
R   DBX   4.4           // Merker 1. Anstoß löschen

// Nulldurchgangsbit usw. rücksetzen

A   #RES_ZERO          // FC-Parameter einlesen           #RES_ZERO          -- 1=reset zero crossing bit
FP  DBX   4.5           // 1. Anstoß
S   DBX   29.5          // (A_BIT3_5) Steuersignal setzen

A   DBX   45.5          // (E_BIT3_5) Rückmeldung erhalten
R   DBX   29.5          // (A_BIT3_5) Steuersignal löschen

AN  DBX   45.5          // (E_BIT3_5) Bearbeitung fertig
AN  DBX   29.5          // (A_BIT3_5)
R   #RES_ZERO          // FC-Parameter löschen           #RES_ZERO          -- 1=reset zero crossing bit
R   DBX   4.5           // Merker 1. Anstoß löschen

// Parameteränderungen übernehmen

A   #C_DOPARA          // FC-Parameter einlesen           #C_DOPARA          -- 1=change parameters
FP  DBX   4.6           // 1. Anstoß
S   DBX   29.6          // (A_BIT3_6) Steuersignal setzen

A   DBX   45.6          // (E_BIT3_6) Rückmeldung erhalten
R   DBX   29.6          // (A_BIT3_6) Steuersignal löschen

AN  DBX   45.6          // (E_BIT3_6) Bearbeitung fertig
AN  DBX   29.6          // (A_BIT3_6)
R   #C_DOPARA          // FC-Parameter löschen           #C_DOPARA          -- 1=change parameters
R   DBX   4.6           // Merker 1. Anstoß löschen

// Tor-Steuerung

```

Page 7 of 9

```

A   #GATE_STP          // Torstopp                       #GATE_STP          -- 1=gate stop
=   DBX   27.2          // (A_BIT1_2) Steuern Zähler

A   #SW_GATE           // SW-Tor Start/Stopp             #SW_GATE           -- 1=start software gate
=   DBX   27.3          // (A_BIT1_3) Steuern Zähler

// Bedienfehler-Bearbeitung

A   DBX   41.3          // (E_BIT0_3) Status Bedienfehler
=   #OT_ERR            // am FC-Parameter mitteilen      #OT_ERR            -- 1=operator error
JCN OTQ
L   DBB   40            //Art des Bedienfehlers
L   5                  //Parametrieren zusammen mit anderen Aufträgen gesetzt?
==I
JCN OTQ
CLR
=   #L_DIRECT          //Anstoßbits löschen für inkompatible Aufträge #L_DIRECT          -- 1=load direct / transfer low
=   #L_PREPAR          //                                er limit
=   #T_CMP_V1          //                                #L_PREPAR          -- 1=prepare load
=   #T_CMP_V2          //                                #T_CMP_V1          -- 1=transfer comparator value
=   #C_DOPARA          //                                1 / transfer upper limit
L   2#10110000        //zugehörige Flankenmerker löschen #T_CMP_V2          -- 1=transfer comparator value
L   DBB   4            //                                2 / transfer Aktualisierungszeit ????
AW                                #C_DOPARA          -- 1=change parameters
T   DBB   4
L   2#10110000        //zugehörige Steuersignale löschen
L   DBB   29
AW
T   DBB   29

OTQ: A   #OT_ERR_A      // Fehler quittieren             #OT_ERR_A          -- 1=operator error acknowledge
=   DBX   26.3          // (A_BIT0_3)

// BG-Ausgänge in Kanal-DB setzen

A   #SET_D00           //                                #SET_D00          -- 1=set D00
=   DBX   28.2

A   #SET_D01           //                                #SET_D01          -- 1=set D01
=   DBX   28.3

// *****
// Load- und die Vergleichswerte zur FM transferieren
// Koordinierungs- und Steuerbytes zur FM transferieren

```

```
// *****
L   DBD   14           // (LOAD_VAL) Loadwert in Steuerschnittstelle
A   #Takttsynchron           #Takttsynchron   -- 1=Aufruf im takttsynchronen O
                                   B
JC   TS5t
T   PQD [AR1,P#0.0]
L   DBD   18           // (CMP_V1) Vergleichswert 1 in Steuerschnittstelle
T   PQD [AR1,P#4.0]
L   DBD   22           // (CMP_V2) Vergleichswert 2 in Steuerschnittstelle
T   PQD [AR1,P#8.0]
L   DBD   26           // Doppelwort mit den Koord.- und Steuerbytes
T   PQD [AR1,P#12.0]
JU   TS5e
TS5t: T   QD [AR1,P#0.0]
      L   DBD   18           // (CMP_V1) Vergleichswert 1 in Steuerschnittstelle
      T   QD [AR1,P#4.0]
      L   DBD   22           // (CMP_V2) Vergleichswert 2 in Steuerschnittstelle
      T   QD [AR1,P#8.0]
      L   DBD   26           // Doppelwort mit den Koord.- und Steuerbytes
      T   QD [AR1,P#12.0]
TS5e: NOP   1
// *****

END: LAR1 DBD   0           // Adressregister 1 laden
      OPN  DB [#rett_dbnr]   // alten vor Aufruf geöffneten DB wieder öffnen   #rett_dbnr   -- Nummer des DB, der beim Aufruf
                                   uf des FC gerade offen war (von AWL benutzt)

      BE
```

FC401 - <offline>

```
"DIAG_INF_M2"   Read diagnostic data Encoder 2
Name:  DIAG_INF   Family: FM_CNT_1
Author: FM        Version: 3.0
                                   Block version: 2
Time stamp Code: 10/15/2002 08:03:22 AM
Interface:      08/26/2002 02:41:29 PM
Lengths (block/logic/data): 00332 00210 00038
```

Name	Data Type	Address	Comment
IN		0.0	
DB_NO	Int	0.0	datablock
OUT		0.0	
IN_OUT		0.0	
IN_DIAG	Bool	4.0	initiate diagnostic interrupt
TEMP		0.0	
x_record	Any	0.0	"Zeiger" auf Zielbereich im DB
!!! immer LB0 bis LB9 !!!			
w_bgadr	Word	10.0	Baugruppemadresse
w_ret_val	Int	12.0	RET_VAL des SFCs
b_busy	Bool	14.0	BUSY-Ausgang des SFCs
dbnr	Word	16.0	DB-Nr. als Lokaldatum
rett_ar1	DWord	18.0	Zwischenspeicher für AR1
rett_dbnr	Word	22.0	Nummer des DB, der beim Aufruf des FC gerade offen war (von AWL benutzt)
s_szl_header	Struct	24.0	SZL-Header für SFC51
len	Word	24.0	
n_dr	Word	26.0	
RETURN		0.0	
RET_VAL	Int	2.0	

Block: FC401 DIAGNOSTIC INTERRUPT INFORMATION

Network: 1 Anlaufkennung prüfen

```

SET
SAVE          // BIE = 1 setzen

TAR1 #rett_ar1 // AR1 retten                                #rett_ar1    -- Zwischenspeicher für AR1

L   #DB_NO     // Übergebene Nummer in dbnr zwischenspeichern #DB_NO       -- datablock
T   #dbnr     //                                     #dbnr       -- DB-Nr. als Lokaldatum
L   DBNO      // Lade Nummer Global-DB in AKKU 1
T   #rett_dbnr // und rette sie in rett_dbnr                #rett_dbnr   -- Nummer des DB, der beim Aufruf d
es FC gerade offen war (von AWL benutzt)

OPN DB [#dbnr] // Instanz-DB öffnen                          #dbnr       -- DB-Nr. als Lokaldatum
AN  DBX 41.5  // FMNEUSTQ_Rückmeldung
A   DBX 41.6  // FMNEUST_Rückmeldung
BEC                                     // *** Bausteinende ***

```

Network: 2 Auftragsbearbeitung

```

AN  #IN_DIAG                                #IN_DIAG     -- initiate diagnostic interrupt
L   B#16#0
JC  RETV

// SFC RD_REC aufrufen, wenn IN_DIAG = 1
L   DBW 6 // Baugruppenadresse
T   #w_bgadr                                #w_bgadr     -- Baugruppemaschine

LAR1 P##x_record // Anfangsadresse des ANY-Pointers

L   W#16#1002 // Syntax-ID und Typ
T   LW [AR1,P#0.0]
L   B#16#10 // DS-Länge = 12 Byte
T   LW [AR1,P#2.0]
L   #DB_NO // DB-Nummer                                #DB_NO       -- datablock

```

Page 2 of 3

```

T   LW [AR1,P#4.0]
L   432 // Zeiger auf Anfang des Datensatzes
T   LD [AR1,P#6.0]
L   B#16#84 // Datenbaustein
T   LB [AR1,P#6.0]

CALL "RDSYSST" // Diagnosedaten von FM lesen                SFC51        -- Read a System Status List or
Partial List

REQ      :=TRUE
SZL_ID   :=W#16#B3
INDEX    :=#w_bgadr                                #w_bgadr     -- Baugruppemaschine
RET_VAL  :=#w_ret_val                               #w_ret_val   -- RET_VAL des SFCs
BUSY     :=#b_busy                                  #b_busy      -- BUSY-Ausgang des SFCs
SZL_HEADER:=#s_szl_header                          #s_szl_header -- SZL-Header für SFC51
DR       :=#x_record // *** ANY-Pointers auf Zielbereich #x_record    -- "Zeiger" auf Zielbereich im
DB !!! immer LB0 bis LB9 !!!

OPN DB [#dbnr] // DB-Nr. als Lokaldatum                    #dbnr       -- DB-Nr. als Lokaldatum

AN  #b_busy // DS-Übertragung abgeschlossen                #b_busy      -- BUSY-Ausgang des SFCs
R   #IN_DIAG // --> Anstoßparameter rücksetzen            #IN_DIAG     -- initiate diagnostic interrupt

L   #w_ret_val // Der Returnwert des SFCs wird an den #w_ret_val   -- RET_VAL des SFCs
RETV: T #RET_VAL // Ausgangsparameter RET_VAL durchgereicht. #RET_VAL
L   B#16#0 // BIE bilden
>=I // BIE = 1 im fehlerfreien Fall
SAVE // BIE = 0 im Fehlerfall

OPN DB [#rett_dbnr] // alten vor Aufruf geöffneten DB wieder öffnen #rett_dbnr   -- Nummer des DB, der beim Aufruf
des FC gerade offen war (von AWL benutzt)
LAR1 #rett_ar1 // AR1 restaurieren                          #rett_ar1    -- Zwischenspeicher für AR1
BE

```

FC402 - <offline>

"CNT_CTL1_ENC2" Control counter module FM 350-1 Encoder 2
Name: CNT_CTL1 **Family:** FM_CNT_1
Author: FM **Version:** 3.0
Block version: 2
Time stamp Code: 10/16/2002 09:14:43 AM
Interface: 09/12/2002 09:01:30 AM
Lengths (block/logic/data): 00956 00760 00046

Name	Data Type	Address	Comment
IN		0.0	
DB_NO	Int	0.0	module datablock number
SW_GATE	Bool	2.0	1=start software gate
GATE_STP	Bool	2.1	1=gate stop
OT_ERR_A	Bool	2.2	1=operator error acknowledge
SET_DO0	Bool	2.3	1=set DO0
SET_DO1	Bool	2.4	1=set DO1
OUT		0.0	
OT_ERR	Bool	4.0	1=operator error
IN_OUT		0.0	
L_DIRECT	Bool	6.0	1=load direct / transfer lower limit
L_PREPAR	Bool	6.1	1=prepare load
T_CMP_V1	Bool	6.2	1=transfer comparator value 1 / transfer upper limit
T_CMP_V2	Bool	6.3	1=transfer comparator value 2 / transfer Aktualisierungszeit ????
C_DOPARA	Bool	6.4	1=change parameters
RES_SYNC	Bool	6.5	1=reset synchronisation bit
RES_ZERO	Bool	6.6	1=reset zero crossing bit
TEMP		0.0	
dbnr	Word	0.0	DB-Nr. als Lokaldatum
rett_dbnr	Word	2.0	Nummer des DB, der beim Aufruf des FC gerade offen war (von AWL benutzt)
SFCErr	Int	4.0	Rückgabewert des SFC6 (RD_SINFO) lokal, da keine relevante Info
Taktsynchron	Bool	6.0	1=Aufruf im taktsynchronen OB
TOP_SI	Struct	8.0	

Page 1 of 9

Name	Data Type	Address	Comment
EV_CLASS	Byte	8.0	
EV_NUM	Byte	9.0	
PRIORITY	Byte	10.0	
NUM	Byte	11.0	
TYP2_3	Byte	12.0	
TYP1	Byte	13.0	
ZI1	Word	14.0	
ZI2_3	DWord	16.0	
START_UP_SI	Struct	20.0	
EV_CLASS	Byte	20.0	
EV_NUM	Byte	21.0	
PRIORITY	Byte	22.0	
NUM	Byte	23.0	
TYP2_3	Byte	24.0	
TYP1	Byte	25.0	
ZI1	Word	26.0	
ZI2_3	DWord	28.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC402 COUNTER MODULE CONTROL

Network: 1

Zuordnung Parameter <==> DB Bit

TFB = A_BIT0_1 Testfreigabe
OT_ERR_A = A_BIT0_3 Bedienfehler quittieren
NEUSTQ = A_BIT0_6 Neustart quittieren
GATE_STP = A_BIT1_2 Torstopp
SW_GATE = A_BIT1_3 SW-Tor Start/Stopp
SET_DO0 = A_BIT2_2 Steuern Ausgang DO0
SET_DO1 = A_BIT2_3 Steuern Ausgang DO1

(Anexo M) 87

Page 2 of 9

```

L_DIRECT = A_BIT3_0      Zähler mit Ladewert laden / Untergrenze
übernehmen
L_PREPAR = A_BIT3_1      Ladewert übernehmen
T_CMP_V1 = A_BIT3_2      Vergleichswert 1 übernehmen / Obergrenze
übernehmen
T_CMP_V2 = A_BIT3_3      Vergleichswert 2 übernehmen /
Aktualisierungszeit übernehmen
RES_SYNC = A_BIT3_4      Synchronisationsbit rücksetzen
RES_ZERO = A_BIT3_5      alle Statusbits rücksetzen
C_DOPARA = A_BIT3_6      Parameteränderung übernehmen

OT_ERR = E_BIT0_3        Bedienfehler
FM-NEUSTQ = E_BIT0_5     FM-Neustart quittiert
FM-NEUST = E_BIT0_6      FM-Neustart
E_BIT3_0                Ladewert in Zähler übernommen
E_BIT3_1                Ladewert ins Laderegister übernommen
E_BIT3_2                Vergleichswert 1 übernommen
E_BIT3_3                Vergleichswert 2 übernommen
E_BIT3_4                Synchronisationsbit gelöscht
E_BIT3_5                Nulldurchgangsbit gelöscht

```

```

*****
Rückmeldungen durch Lesen von der FM aktualisieren
*****

```

```

SET
SAVE // BIE fest auf 1 !!!

L #DB_NO // Übergabene Nummer in dbnr zwischenspeichern #DB_NO -- module datablock number
T #dbnr // #dbnr -- DB-Nr. als Lokaldatum
L DBNO // Lade Nummer Global-DB in AKKU 1
T #rett_dbnr // und rette sie in rett_dbnr #rett_dbnr -- Nummer des DB, der beim Aufruf
// des FC gerade offen war (von AWL benutzt)

OPN DB [#dbnr] // Öffne Kanal-DB mit Übergabener Nummer #dbnr -- DB-Nr. als Lokaldatum

TAR1 DBD 0 // Adressregister 1 retten

L DBD 8 // (CH_ADR) Kanalanzfangsadresse
LAR1 // in das Adressregister 1

// *****
// taktischer Betrieb (d.h. Aufruf in OB61..64)?
// *****

CALL "RD_SINFO" SFC6 -- Read OB Start Information

```

Page 3 of 9

```

RET_VAL :=#SFCErr #SFCErr -- Rückgabewert des SFC6 (RD_SI
NFO) lokal, da keine relevante Info
TOP_SI :=#TOP_SI #TOP_SI
START_UP_SI:=#START_UP_SI #START_UP_SI

L #TOP_SI.NUM #TOP_SI.NUM
L 61
>=I
TAK
L 64
A(
<=I
)
= #Taktisynchron #Taktisynchron -- 1=Aufruf im taktisynchronen O
B

// *****
// Neustart-Quittung
// *****

A #Taktisynchron #Taktisynchron -- 1=Aufruf im taktisynchronen O
B

JC TS1t
L PID [AR1,P#8.0] // Datenfehler, Bedienfehler und Koordinierungsbyte
JU TS1e
TS1t: L ID [AR1,P#8.0] // Datenfehler, Bedienfehler und Koordinierungsbyte
TS1e: T DBD 38

A DBX 41.6 // FM-NEUST (E_BIT0_6)
AN DBX 41.5 // FM-NEUSTQ (E_BIT0_5)
= DBX 26.6 // NEUSTQ (A_BIT0_6)
JCN OK // Kein Neustart

L DW#16#40000000 // Koordinierungsbyte schreiben
A #Taktisynchron #Taktisynchron -- 1=Aufruf im taktisynchronen O
B

JC TS2t
T PQD [AR1,P#12.0] // und Steuerbits löschen
JU TS2e
TS2t: T QD [AR1,P#12.0] // und Steuerbits löschen
TS2e: L B#16#0
T DBB 4 // (FP) Flankenmerker löschen
JU END // und Baustein verlassen

OK: A #Taktisynchron #Taktisynchron -- 1=Aufruf im taktisynchronen O
B

```

(Anexo M) 88

Page 4 of 9

```

JC      TS3t
L      PID [AR1,P#0.0] // (ACT_LOAD) Loadwert in den Kanal-DB
T      DBD 30
L      PID [AR1,P#4.0] // (ACT_CNTV) Zählstand in den Kanal-DB
T      DBD 34
L      PID [AR1,P#12.0] // Reserve und Statusbytes
JU     TS3e
TS3t: L ID [AR1,P#0.0] // (ACT_LOAD) Loadwert in den Kanal-DB
      T DBD 30
      L ID [AR1,P#4.0] // (ACT_CNTV) Zählstand in den Kanal-DB
      T DBD 34
      L ID [AR1,P#12.0] // Reserve und Statusbytes
TS3e: T DBD 42

L      DBB 12 // (U_D_LGTH) 16 : S7-300, 32 : S7-400
L      B#16#20
<I
JC     L1 // Wenn S7-300 : Keine Vergleichswerte lesen

A      #Taktsynchron #Taktsynchron -- 1=Aufruf im taktsynchronen O
      B

JC     TS4t
L      PID [AR1,P#16.0] // Aktueller Vergleichswert 1
T      DBD 46 // (ACT_CMP1)
L      PID [AR1,P#20.0] // Aktueller Vergleichswert 2
JU     TS4e
TS4t: L ID [AR1,P#16.0] // Aktueller Vergleichswert 1
      T DBD 46 // (ACT_CMP1)
      L ID [AR1,P#20.0] // Aktueller Vergleichswert 2
TS4e: T DBD 50 // (ACT_CMP2)

// *****
// FB-Steuer-Parameter bitweise in den Datenbaustein übertragen
// *****

// Zähler mit Ladewert laden

L1: A #L_DIRECT // FC-Parameter einlesen #L_DIRECT -- 1=load direct / transfer low
      er limit
FP   DBX 4.0 // 1. Anstoß
S    DBX 29.0 // (A_BIT3_0) Steuersignal setzen

A    DBX 45.0 // (E_BIT3_0) Rückmeldung erhalten
R    DBX 29.0 // (A_BIT3_0) Steuersignal löschen

AN   DBX 45.0 // (E_BIT3_0) Bearbeitung fertig
AN   DBX 29.0 // (A_BIT3_0)

```

Page 5 of 9

```

R      #L_DIRECT // FC-Parameter löschen #L_DIRECT -- 1=load direct / transfer low
      er limit
R      DBX 4.0 // Merker 1. Anstoß löschen

// Ladewert ins Laderegister

A      #L_PREPAR // FC-Parameter einlesen #L_PREPAR -- 1=prepare load
FP     DBX 4.1 // 1. Anstoß
S      DBX 29.1 // (A_BIT3_1) Steuersignal setzen

A      DBX 45.1 // (E_BIT3_1) Rückmeldung erhalten
R      DBX 29.1 // (A_BIT3_1) Steuersignal löschen

AN     DBX 45.1 // (E_BIT3_1) Bearbeitung fertig
AN     DBX 29.1 // (A_BIT3_1)
R      #L_PREPAR // FC-Parameter löschen #L_PREPAR -- 1=prepare load
R      DBX 4.1 // Merker 1. Anstoß löschen

// Vergleichswert 1 übernehmen

A      #T_CMP_V1 // FC-Parameter einlesen #T_CMP_V1 -- 1=transfer comparator value
      1 / transfer upper limit
FP     DBX 4.2 // 1. Anstoß
S      DBX 29.2 // (A_BIT3_2) Steuersignal setzen

A      DBX 45.2 // (E_BIT3_2) Rückmeldung erhalten
R      DBX 29.2 // (A_BIT3_2) Steuersignal löschen

AN     DBX 45.2 // (E_BIT3_2) Bearbeitung fertig
AN     DBX 29.2 // (A_BIT3_2)
R      #T_CMP_V1 // FC-Parameter löschen #T_CMP_V1 -- 1=transfer comparator value
      1 / transfer upper limit
R      DBX 4.2 // Merker 1. Anstoß löschen

// Vergleichswert 2 übernehmen

A      #T_CMP_V2 // FC-Parameter einlesen #T_CMP_V2 -- 1=transfer comparator value
      2 / transfer Aktualisierungszeit ???
FP     DBX 4.3 // 1. Anstoß
S      DBX 29.3 // (A_BIT3_3) Steuersignal setzen

A      DBX 45.3 // (E_BIT3_3) Rückmeldung erhalten
R      DBX 29.3 // (A_BIT3_3) Steuersignal löschen

AN     DBX 45.3 // (E_BIT3_3) Bearbeitung fertig
AN     DBX 29.3 // (A_BIT3_3)
R      #T_CMP_V2 // FC-Parameter löschen #T_CMP_V2 -- 1=transfer comparator value
      2 / transfer Aktualisierungszeit ???

```

```

R   DBX   4.3           // Merker 1. Anstoß löschen

// Synchronisationsbit rücksetzen

A   #RES_SYNC          // FC-Parameter einlesen           #RES_SYNC          -- 1=reset synchronisation bit
FP  DBX   4.4           // 1. Anstoß
S   DBX   29.4          // (A_BIT3_4) Steuersignal setzen

A   DBX   45.4          // (E_BIT3_4) Rückmeldung erhalten
R   DBX   29.4          // (A_BIT3_4) Steuersignal löschen

AN  DBX   45.4          // (E_BIT3_4) Bearbeitung fertig
AN  DBX   29.4          // (A_BIT3_4)
R   #RES_SYNC          // FC-Parameter löschen           #RES_SYNC          -- 1=reset synchronisation bit
R   DBX   4.4           // Merker 1. Anstoß löschen

// Nulldurchgangsbit usw. rücksetzen

A   #RES_ZERO          // FC-Parameter einlesen           #RES_ZERO          -- 1=reset zero crossing bit
FP  DBX   4.5           // 1. Anstoß
S   DBX   29.5          // (A_BIT3_5) Steuersignal setzen

A   DBX   45.5          // (E_BIT3_5) Rückmeldung erhalten
R   DBX   29.5          // (A_BIT3_5) Steuersignal löschen

AN  DBX   45.5          // (E_BIT3_5) Bearbeitung fertig
AN  DBX   29.5          // (A_BIT3_5)
R   #RES_ZERO          // FC-Parameter löschen           #RES_ZERO          -- 1=reset zero crossing bit
R   DBX   4.5           // Merker 1. Anstoß löschen

// Parameteränderungen übernehmen

A   #C_DOPARA          // FC-Parameter einlesen           #C_DOPARA          -- 1=change parameters
FP  DBX   4.6           // 1. Anstoß
S   DBX   29.6          // (A_BIT3_6) Steuersignal setzen

A   DBX   45.6          // (E_BIT3_6) Rückmeldung erhalten
R   DBX   29.6          // (A_BIT3_6) Steuersignal löschen

AN  DBX   45.6          // (E_BIT3_6) Bearbeitung fertig
AN  DBX   29.6          // (A_BIT3_6)
R   #C_DOPARA          // FC-Parameter löschen           #C_DOPARA          -- 1=change parameters
R   DBX   4.6           // Merker 1. Anstoß löschen

// Tor-Steuerung

```

Page 7 of 9

```

A   #GATE_STP          // Torstopp                         #GATE_STP          -- 1=gate stop
=   DBX   27.2          // (A_BIT1_2) Steuern Zähler

A   #SW_GATE           // SW-Tor Start/Stopp               #SW_GATE           -- 1=start software gate
=   DBX   27.3          // (A_BIT1_3) Steuern Zähler

// Bedienfehler-Bearbeitung

A   DBX   41.3          // (E_BIT0_3) Status Bedienfehler
=   #OT_ERR            // am FC-Parameter mitteilen        #OT_ERR            -- 1=operator error
JCN OTQ
L   DBB   40            //Art des Bedienfehlers
L   5                  //Parametrieren zusammen mit anderen Aufträgen gesetzt?
==I
JCN OTQ
CLR
=   #L_DIRECT          //Anstoßbits löschen für inkompatible Aufträge #L_DIRECT          -- 1=load direct / transfer low
=   #L_PREPAR          //                                er limit
=   #T_CMP_V1          //                                #L_PREPAR          -- 1=prepare load
=   #T_CMP_V2          //                                #T_CMP_V1          -- 1=transfer comparator value
=   #C_DOPARA          //                                1 / transfer upper limit
L   2#10110000         //zugehörige Flankenmerker löschen #T_CMP_V2          -- 1=transfer comparator value
L   DBB   4            //                                2 / transfer Aktualisierungszeit ????
AW                                #C_DOPARA          -- 1=change parameters
T   DBB   4
L   2#10110000         //zugehörige Steuersignale löschen
L   DBB   29
AW
T   DBB   29

OTQ: A   #OT_ERR_A      // Fehler quittieren                 #OT_ERR_A          -- 1=operator error acknowledge
=   DBX   26.3          // (A_BIT0_3)

// BG-Ausgänge in Kanal-DB setzen

A   #SET_D00           //                                #SET_D00          -- 1=set D00
=   DBX   28.2

A   #SET_D01           //                                #SET_D01          -- 1=set D01
=   DBX   28.3

// *****
// Load- und die Vergleichswerte zur FM transferieren
// Koordinierungs- und Steuerbytes zur FM transferieren

```

```
// *****
L   DBD   14           // (LOAD_VAL) Loadwert in Steuerschnittstelle
A   #Takttsynchron           #Takttsynchron   -- 1=Aufruf im takttsynchronen O
                                   B
JC   TS5t
T   PQD [AR1,P#0.0]
L   DBD   18           // (CMP_V1) Vergleichswert 1 in Steuerschnittstelle
T   PQD [AR1,P#4.0]
L   DBD   22           // (CMP_V2) Vergleichswert 2 in Steuerschnittstelle
T   PQD [AR1,P#8.0]
L   DBD   26           // Doppelwort mit den Koordin.- und Steuerbytes
T   PQD [AR1,P#12.0]
JU   TS5e
TS5t: T   QD [AR1,P#0.0]
      L   DBD   18           // (CMP_V1) Vergleichswert 1 in Steuerschnittstelle
      T   QD [AR1,P#4.0]
      L   DBD   22           // (CMP_V2) Vergleichswert 2 in Steuerschnittstelle
      T   QD [AR1,P#8.0]
      L   DBD   26           // Doppelwort mit den Koordin.- und Steuerbytes
      T   QD [AR1,P#12.0]
TS5e: NOP   1
// *****

END: LAR1 DBD   0           // Adressregister 1 laden

      OPN DB [#rett_dbnr]   // alten vor Aufruf geöffneten DB wieder öffnen   #rett_dbnr   -- Nummer des DB, der beim Aufruf
                                   uf des FC gerade offen war (von AWL benutzt)

      BE
```

Page 9 of 9

FC500 - <offline>

"BEGIN_DATA" Após fazer reset introduz valor standard de velocidade M1 e M2
Name: BEGIN_DA **Family:**
Author: BRobalo **Version:** 0.1
Block version: 2
Time stamp Code: 07/18/2012 06:37:24 PM
Interface: 08/01/2009 11:52:47 AM
Lengths (block/logic/data): 00118 00024 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC500 BEGIN_DATA

Quando o operador pressiona o botão de Reset (F7 HMI) o sistema carrega os valores de velocidade por defeito, para os motores 1 e 2.

Network: 1

```
A   "reset_m"      M7.1
JCN END
L   L#25000
T   "VLEVEL1_M1_OP" MD15
T   "VLEVEL1_M2_OP" MD19
S   "Data_Enter"   M23.0
R   "Data_Enter"   M23.0
```

END: NOP 0

FC600 - <offline>

"COUNTER_REV_ENC1" New counter value Encoder 1
Name: COUNTER1 **Family:**
Author: BRobalo **Version:** 0.1
Block version: 2
Time stamp Code: 07/18/2012 04:52:39 PM
Interface: 08/08/2009 03:05:24 PM
Lengths (block/logic/data): 00144 00050 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC600

Network: 1

```

A      "DB_UDT_M1".D_IN1          DB1.DBX220.1    -- digital input 1
AN     "DBEX_100_M1".WORKING     DB100.DBX10.1  -- Machining in progress
A      "RESEARCH_MODE"          M26.0
JCN    END
L      W#16#0
T      "DB_PROG_ENC1".L_DIRECT_VAL DB300.DBD4      -- New counter value
S      "DB_PROG_ENC1".SET_L_DIRECT DB300.DBX0.7   -- 1=New counter value
R      "DB_PROG_ENC1".SET_L_DIRECT DB300.DBX0.7   -- 1=New counter value

```

END: NOP 0

Page 1 of 1

FC700 - <offline>

"COUNTER_REV_ENC2" New counter value Encoder 2
Name: COUNTER2 **Family:**
Author: BRobalo **Version:** 0.1
Block version: 2
Time stamp Code: 07/18/2012 04:52:53 PM
Interface: 08/08/2009 03:05:24 PM
Lengths (block/logic/data): 00144 00050 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
IN_OUT		0.0	
TEMP		0.0	
RETURN		0.0	
RET_VAL		0.0	

Block: FC700

Network: 1 1=New counter value

```

A      "DB_UDT_M2".D_IN1_M2       DB2.DBX220.1    -- digital input 1
AN     "DBEX_100_M1".WORKING     DB100.DBX10.1  -- Machining in progress
A      "RESEARCH_MODE"          M26.0
JCN    END
L      W#16#0
T      "DB_PROG_ENC2".L_DIRECT_VAL DB400.DBD4      -- New counter value
S      "DB_PROG_ENC2".SET_L_DIRECT DB400.DBX0.7   -- 1=New counter value
R      "DB_PROG_ENC2".SET_L_DIRECT DB400.DBX0.7   -- 1=New counter value

```

END: NOP 0

FB21 - <offline>

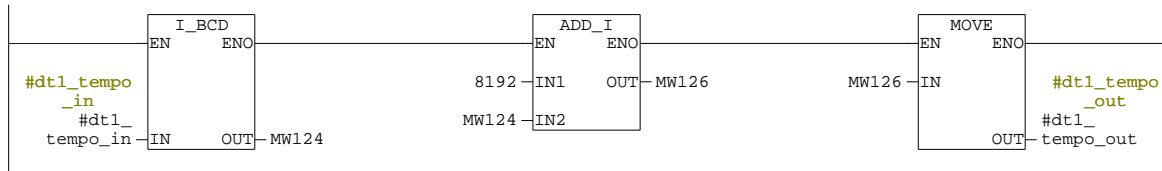
"CONV_T" Converte INT para S5TIME
Name: CONV_T **Family:**
Author: EACF **Version:** 0.1
Block version: 2
Time stamp Code: 08/01/2012 04:55:37 PM
Interface: 07/31/2012 08:04:58 PM
Lengths (block/logic/data): 00160 00060 00000

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
dt1_tempo_in	Int	0.0	0	
OUT		0.0		
dt1_tempo_out	S5Time	2.0	S5T#0MS	
IN_OUT		0.0		
STAT		0.0		
TEMP		0.0		

Block: FB21

Faz a conversão do tempo introduzido como um número inteiro para dado do tipo S5TIME

Network: 1



FB300 - <offline>

"PROG_ENC_1" Programa Encoder 1 (FM350-1)
Name: PROG_E1 **Family:** FM_CNT_1
Author: FM **Version:** 3.0
Block version: 2
Time stamp Code: 07/18/2012 05:08:11 PM
Interface: 11/27/2002 12:00:24 PM
Lengths (block/logic/data): 00926 00710 00008

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
SW_GATE	Bool	0.0	FALSE	1=Control software gate
GATE_STP	Bool	0.1	FALSE	1=Stop gate
OT_ERR_A	Bool	0.2	FALSE	1=Confirm operator error
DIAG_ERR_A	Bool	0.3	FALSE	1=Confirm diagnostic interrupt
HW_INT_ERR_A	Bool	0.4	FALSE	1=Confirm hardware interrupt
SET_DO0	Bool	0.5	FALSE	1=Set output DO0
SET_DO1	Bool	0.6	FALSE	1=Set output DO1
SET_L_DIRECT	Bool	0.7	FALSE	1=New counter value
SET_L_PREPAR	Bool	1.0	FALSE	1=Prepare new counter value
SET_T_CMP_V1	Bool	1.1	FALSE	1=New compare value 1
SET_T_CMP_V2	Bool	1.2	FALSE	1=New compare value 2
SET_C_DOPARA	Bool	1.3	FALSE	1=Triggering parameter change
SET_RES_SYNC	Bool	1.4	FALSE	1=Reset status bit "synchronization"
SET_RES_ZERO	Bool	1.5	FALSE	1=Reset status bit "zero-crossing"
ENSET_UP	Bool	1.6	FALSE	1=Enable set in direction up (forward)
ENSET_DN	Bool	1.7	FALSE	1=Enable set in direction down (backward)
CTRL_DO0	Bool	2.0	FALSE	1=Control digital output DO0
CTRL_DO1	Bool	2.1	FALSE	1=Control digital output DO1
L_DIRECT_VAL	DInt	4.0	L#0	New counter value
L_PREPAR_VAL	DInt	8.0	L#0	New preparation value
T_CMP_V1_VAL	DInt	12.0	L#0	New compare value 1
T_CMP_V2_VAL	DInt	16.0	L#0	New compare value 2

Page 1 of 9

Name	Data Type	Address	Initial Value	Comment
DO0_MODE	Byte	20.0	B#16#0	Set DO0 behavior (0..5)
DO1_MODE	Byte	21.0	B#16#0	Set DO1 behavior (0..6)
HYSTERESIS	Byte	22.0	B#16#0	Hysteresis value (0..255)
PULSE_DURATION	Byte	23.0	B#16#0	Pulse duration (0..250)
OUT		0.0		
OT_ERR	Bool	24.0	FALSE	Error in CNT_CTL1 function
DIAG_ERR	Bool	24.1	FALSE	1=Diagnostic interrupt received
HW_INT_ERR	Bool	24.2	FALSE	1=Hardware interrupt received
STS_RUN	Bool	24.3	FALSE	"Counter is running" status
STS_DIR	Bool	24.4	FALSE	"Direction bit" status
STS_ZERO	Bool	24.5	FALSE	"Zero-Crossing" status
STS_OFLW	Bool	24.6	FALSE	"Counter Overflow" status
STS_UFLW	Bool	24.7	FALSE	"Counter Underflow" status
STS_SW_G	Bool	25.0	FALSE	"Software Gate" status
STS_GATE	Bool	25.1	FALSE	"Internal Gate" status
OT_ERR_B	Byte	26.0	B#16#0	Operator error
LATCH_LOAD	DWord	28.0	DW#16#0	Current load value or latch value
ACT_CNTV	DWord	32.0	DW#16#0	Current count value
IN_OUT		0.0		
L_DIRECT	Bool	36.0	FALSE	1=Load new counter value
L_PREPAR	Bool	36.1	FALSE	1=Prepare new counter value
T_CMP_V1	Bool	36.2	FALSE	1=Load new compare value 1
T_CMP_V2	Bool	36.3	FALSE	1=Load new compare value 2
C_DOPARA	Bool	36.4	FALSE	1=Triggering parameter change
RES_SYNC	Bool	36.5	FALSE	1=Reset status bit "synchronization"
RES_ZERO	Bool	36.6	FALSE	1=Reset status bit "zero-crossing"
DIAG_INF	Bool	36.7	FALSE	1=Read diagnostics (only for OB82)
STAT		0.0		
FP_L_DIRECT	Bool	38.0	FALSE	1=Load new counter value
FP_L_PREPAR	Bool	38.1	FALSE	1=Prepare new counter value

(Anexo M) 94

Page 2 of 9

Name	Data Type	Address	Initial Value	Comment
FP_T_CMP_V1	Bool	38.2	FALSE	1=Load new compare value 1
FP_T_CMP_V2	Bool	38.3	FALSE	1=Load new compare value 2
FP_C_DOPARA	Bool	38.4	FALSE	1=Triggering parameter change
FP_RES_SYNC	Bool	38.5	FALSE	1=Reset status bit "synchronization"
FP_RES_ZERO	Bool	38.6	FALSE	1=Reset status bit "zero-crossing"
SFC_ERR	Int	40.0	0	SFC51 error status
TEMP		0.0		

Block: FB300 EXAMPLE PROGRAM
Network: 1 Triggering functions

```

*****
In the event of a positive edge in the initiation bit:
- set corresponding bit in the FC302 call
- transfer corresponding value from the instance DB to the channel DB
*****

NOP 0
// *****
// Triggering the function "Direct load"
// *****

A #SET_L_DIRECT //Direct load #SET_L_DIRECT -- 1=New counter value
FP #FP_L_DIRECT //positive edge #FP_L_DIRECT -- 1=Load new counter value
S #L_DIRECT #L_DIRECT -- 1=Load new counter value
JCN NEX1
L #L_DIRECT_VAL //Download parameters in channel DB #L_DIRECT_VAL -- New counter value
T "DB_UDT_ENC1".LOAD_VAL DB3.DBD14 -- New load value (write user)

// *****
// Triggering the function "Prepare to load"
// *****

NEX1: A #SET_L_PREPAR //Prepare to load #SET_L_PREPAR -- 1=Prepare new counter value
FP #FP_L_PREPAR //positive edge #FP_L_PREPAR -- 1=Prepare new counter value
S #L_PREPAR #L_PREPAR -- 1=Prepare new counter value
JCN NEX2
L #L_PREPAR_VAL //Download parameters in channel DB #L_PREPAR_VAL -- New preparation value

T "DB_UDT_ENC1".LOAD_VAL DB3.DBD14 -- New load value (write user)

// *****
// Triggering the function "Load compare value 1"
// *****

NEX2: A #SET_T_CMP_V1 //Transfer compare value 1 #SET_T_CMP_V1 -- 1=New compare value 1
FP #FP_T_CMP_V1 //positive edge #FP_T_CMP_V1 -- 1=Load new compare value 1
S #T_CMP_V1 #T_CMP_V1 -- 1=Load new compare value 1
JCN NEX3
L #T_CMP_V1_VAL //Download parameters in channel DB #T_CMP_V1_VAL -- New compare value 1
T "DB_UDT_ENC1".CMP_V1 DB3.DBD18 -- New comparator value 1 (write user)

// *****
// Triggering the function "Load compare value 2"
// *****

NEX3: A #SET_T_CMP_V2 //Transfer compare value 2 #SET_T_CMP_V2 -- 1=New compare value 2
FP #FP_T_CMP_V2 //positive edge #FP_T_CMP_V2 -- 1=Load new compare value 2
S #T_CMP_V2 #T_CMP_V2 -- 1=Load new compare value 2
JCN NEX4
L #T_CMP_V2_VAL //Download parameters in channel DB #T_CMP_V2_VAL -- New compare value 2
T "DB_UDT_ENC1".CMP_V2 DB3.DBD22 -- New comparator value 2 (write user)

// *****
// Triggering the function "Triggering parameter change"
//
// Note: do not use together with the jobs "L_DIRECT" or
// "L_PREPAR" or "T_CMP_V1" or "T_CMP_V2"
// (otherwise operator error 5 will occur)
// Note: Must be DO0_MODE = 0 for DOL_MODE = 6
// (otherwise operator error 21 will occur)
//
// The multiple usage of DBD14 is hidden by FB100.
// The use of half bytes in DBB14 is hidden by FB100.
// *****

NEX4: A #SET_C_DOPARA #SET_C_DOPARA -- 1=Triggering parameter change
FP #FP_C_DOPARA //positive edge #FP_C_DOPARA -- 1=Triggering parameter change
S #C_DOPARA #C_DOPARA -- 1=Triggering parameter change
JCN NEX5
//Download changed parameters in channel DB
L #DO0_MODE //DO0 response #DO0_MODE -- Set DO0 behavior (0..5)

```

Page 3 of 9

```

T "DB_UDT_ENC1".LOAD_VAL DB3.DBD14 -- New load value (write user)

// *****
// Triggering the function "Load compare value 1"
// *****

NEX2: A #SET_T_CMP_V1 //Transfer compare value 1 #SET_T_CMP_V1 -- 1=New compare value 1
FP #FP_T_CMP_V1 //positive edge #FP_T_CMP_V1 -- 1=Load new compare value 1
S #T_CMP_V1 #T_CMP_V1 -- 1=Load new compare value 1
JCN NEX3
L #T_CMP_V1_VAL //Download parameters in channel DB #T_CMP_V1_VAL -- New compare value 1
T "DB_UDT_ENC1".CMP_V1 DB3.DBD18 -- New comparator value 1 (write user)

// *****
// Triggering the function "Load compare value 2"
// *****

NEX3: A #SET_T_CMP_V2 //Transfer compare value 2 #SET_T_CMP_V2 -- 1=New compare value 2
FP #FP_T_CMP_V2 //positive edge #FP_T_CMP_V2 -- 1=Load new compare value 2
S #T_CMP_V2 #T_CMP_V2 -- 1=Load new compare value 2
JCN NEX4
L #T_CMP_V2_VAL //Download parameters in channel DB #T_CMP_V2_VAL -- New compare value 2
T "DB_UDT_ENC1".CMP_V2 DB3.DBD22 -- New comparator value 2 (write user)

// *****
// Triggering the function "Triggering parameter change"
//
// Note: do not use together with the jobs "L_DIRECT" or
// "L_PREPAR" or "T_CMP_V1" or "T_CMP_V2"
// (otherwise operator error 5 will occur)
// Note: Must be DO0_MODE = 0 for DOL_MODE = 6
// (otherwise operator error 21 will occur)
//
// The multiple usage of DBD14 is hidden by FB100.
// The use of half bytes in DBB14 is hidden by FB100.
// *****

NEX4: A #SET_C_DOPARA #SET_C_DOPARA -- 1=Triggering parameter change
FP #FP_C_DOPARA //positive edge #FP_C_DOPARA -- 1=Triggering parameter change
S #C_DOPARA #C_DOPARA -- 1=Triggering parameter change
JCN NEX5
//Download changed parameters in channel DB
L #DO0_MODE //DO0 response #DO0_MODE -- Set DO0 behavior (0..5)

```

(Anexo M) 95

Page 4 of 9

```

L   W#16#7           //Mask relevant bits
AW
T   DB3.DBB 14
L   #DO1_MODE       //DO1 response           #DO1_MODE       -- Set DO1 behavior (0..6)
L   W#16#7           //Mask relevant bits
AW
SLW 4               //shift to bit 7..4
L   DB3.DBB 14       //OR with DO0_MODE
OW
T   DB3.DBB 14

L   #HYSTERESIS     #HYSTERESIS     -- Hysteresis value (0..255)
T   DB3.DBB 15

L   #PULSE_DURATION #PULSE_DURATION  -- Pulse duration (0..250)
T   DB3.DBB 16

L   0
T   DB3.DBB 17

// *****
// Triggering the function "Reset status bit synchronization"
// *****
NEX5: A   #SET_RES_SYNC           #SET_RES_SYNC   -- 1=Reset status bit "synchron
FP   #FP_RES_SYNC           //positive edge           #FP_RES_SYNC   -- 1=Reset status bit "synchron
S   #RES_SYNC               #RES_SYNC       -- 1=Reset status bit "synchron

// *****
// Triggering the function "Reset status bit zero-crossing"
// *****
A   #SET_RES_ZERO           #SET_RES_ZERO   -- 1=Reset status bit "zero-cr
FP   #FP_RES_ZERO           //positive edge           #FP_RES_ZERO   -- 1=Reset status bit "zero-cr
S   #RES_ZERO               #RES_ZERO       -- 1=Reset status bit "zero-cr

// *****
// Download enable bits in channel DB
// *****
A   #ENSET_UP               #ENSET_UP       -- 1=Enable set in direction u
p (forward)

```

Page 5 of 9

```

=   "DB_UDT_ENC1".ENSET_UP     DB3.DBX27.0     -- Enable set in direction up
                               (=forward) (write user)
A   #ENSET_DN                 #ENSET_DN       -- 1=Enable set in direction d
                               own (backward)
=   "DB_UDT_ENC1".ENSET_DN     DB3.DBX27.1     -- Enable set in direction dow
                               n (=backward) (write user)
A   #CTRL_DO0                 #CTRL_DO0       -- 1=Control digital output DO
                               0
=   "DB_UDT_ENC1".CTRL_DO0     DB3.DBX28.0     -- Control digital output DO0
                               (write user)
A   #CTRL_DO1                 #CTRL_DO1       -- 1=Control digital output DO
                               1
=   "DB_UDT_ENC1".CTRL_DO1     DB3.DBX28.1     -- Control digital output DO1
                               (write user)

```

Network: 2 Call FC

```

*****
Call standard function "Control FM 350-1/450-1"
*****

```

```

CALL "CNT_CTL1_ENC1" //Control FM 350-1/450-1           FC302           -- Control counter module FM 350-
1 Encoder 1
DB_NO :=3           //Data block for channel 1
SW_GATE :=#SW_GATE //Control software gate           #SW_GATE       -- 1=Control software gate
GATE_STP:=#GATE_STP //Stop internal gate           #GATE_STP     -- 1=Stop gate
OT_ERR_A:=#OT_ERR_A //Confirm operator error           #OT_ERR_A     -- 1=Confirm operator error
SET_DO0 :=#SET_DO0 //Set output DO0           #SET_DO0      -- 1=Set output DO0
SET_DO1 :=#SET_DO1 //Set output DO1           #SET_DO1      -- 1=Set output DO1
OT_ERR :=#OT_ERR //Operator error occurred           #OT_ERR       -- Error in CNT_CTL1 function
L_DIRECT:=#L_DIRECT //Load new counter value           #L_DIRECT     -- 1=Load new counter value
L_PREPAR:=#L_PREPAR //Prepare new counter value           #L_PREPAR     -- 1=Prepare new counter value
T_CMP_V1:=#T_CMP_V1 //Load new compare value 1           #T_CMP_V1     -- 1=Load new compare value 1
T_CMP_V2:=#T_CMP_V2 //Load new compare value 2           #T_CMP_V2     -- 1=Load new compare value 2
C_DOPARA:=#C_DOPARA //Triggering parameter change           #C_DOPARA     -- 1=Triggering parameter change
RES_SYNC:=#RES_SYNC //Reset synchronization bit           #RES_SYNC     -- 1=Reset status bit "synchron
RES_ZERO:=#RES_ZERO //Reset zero-crossing bit           #RES_ZERO     -- 1=Reset status bit "zero-cross
ing"

```

Network: 3 Post-editing

NOP 0

// *****
// Reset edge memory bit
// *****

AN	#SET_L_DIRECT	#SET_L_DIRECT	-- 1=New counter value
AN	#L_DIRECT	#L_DIRECT	-- 1=Load new counter value
R	#FP_L_DIRECT	#FP_L_DIRECT	-- 1=Load new counter value
AN	#SET_L_PREPAR	#SET_L_PREPAR	-- 1=Prepare new counter value
AN	#L_PREPAR	#L_PREPAR	-- 1=Prepare new counter value
R	#FP_L_PREPAR	#FP_L_PREPAR	-- 1=Prepare new counter value
AN	#SET_T_CMP_V1	#SET_T_CMP_V1	-- 1=New compare value 1
AN	#T_CMP_V1	#T_CMP_V1	-- 1=Load new compare value 1
R	#FP_T_CMP_V1	#FP_T_CMP_V1	-- 1=Load new compare value 1
AN	#SET_T_CMP_V2	#SET_T_CMP_V2	-- 1=New compare value 2
AN	#T_CMP_V2	#T_CMP_V2	-- 1=Load new compare value 2
R	#FP_T_CMP_V2	#FP_T_CMP_V2	-- 1=Load new compare value 2
AN	#SET_C_DOPARA	#SET_C_DOPARA	-- 1=Triggering parameter change
AN	#C_DOPARA	#C_DOPARA	-- 1=Triggering parameter change
R	#FP_C_DOPARA	#FP_C_DOPARA	-- 1=Triggering parameter change
AN	#SET_RES_SYNC	#SET_RES_SYNC	-- 1=Reset status bit "synchronization"
AN	#RES_SYNC	#RES_SYNC	-- 1=Reset status bit "synchronization"
R	#FP_RES_SYNC	#FP_RES_SYNC	-- 1=Reset status bit "synchronization"
AN	#SET_RES_ZERO	#SET_RES_ZERO	-- 1=Reset status bit "zero-crossing"

Page 7 of 9

AN	#RES_ZERO	#RES_ZERO	-- 1=Reset status bit "zero-crossing"
R	#FP_RES_ZERO	#FP_RES_ZERO	-- 1=Reset status bit "zero-crossing"

// *****
// Display status messages
// *****

A	"DB_UDT_ENC1".STS_RUN	DB3.DBX43.0	-- Status counter is running
=	#STS_RUN	(read user) #STS_RUN	-- "Counter is running" status
A	"DB_UDT_ENC1".STS_DIR	DB3.DBX43.1	-- Status of the counter direction bit (read user)
=	#STS_DIR	#STS_DIR	-- "Direction bit" status
A	"DB_UDT_ENC1".STS_ZERO	DB3.DBX43.2	-- Status of the counter zero-crossing bit (read user)
=	#STS_ZERO	#STS_ZERO	-- "Zero-Crossing" status
A	"DB_UDT_ENC1".STS_OFLW	DB3.DBX43.3	-- Status counter passed the overflow value (read user)
=	#STS_OFLW	#STS_OFLW	-- "Counter Overflow" status
A	"DB_UDT_ENC1".STS_UFLW	DB3.DBX43.4	-- Status counter passed the underflow value (read user)
=	#STS_UFLW	#STS_UFLW	-- "Counter Underflow" status
A	"DB_UDT_ENC1".STS_SW_G	DB3.DBX43.7	-- Status of the software gate (read user)
=	#STS_SW_G	#STS_SW_G	-- "Software Gate" status
A	"DB_UDT_ENC1".STS_GATE	DB3.DBX43.6	-- Status of the internal gate (read user)
=	#STS_GATE	#STS_GATE	-- "Internal Gate" status
L	"DB_UDT_ENC1".OT_ERR_B	DB3.DBB40	-- Operator error byte (read user)
T	#OT_ERR_B	#OT_ERR_B	-- Operator error
L	"DB_UDT_ENC1".LATCH_LOAD	DB3.DBD30	-- Actual latch or load value (read user)
T	#LATCH_LOAD	#LATCH_LOAD	-- Current load value or latch value
L	"DB_UDT_ENC1".ACT_CNTV	DB3.DBD34	-- Actual counter value (read user)
T	#ACT_CNTV	#ACT_CNTV	-- Current count value

// *****
// Confirm interrupts
// *****

(Anexo M) 97

Page 8 of 9

```

A  #DIAG_ERR_A          //Confirm diagnostic interrupt          #DIAG_ERR_A          -- l=Confirm diagnostic inter
R  #DIAG_ERR            #DIAG_ERR            #DIAG_ERR            -- l=Diagnostic interrupt rec
                                eived

A  #HW_INT_ERR_A       //Confirm hardware interrupt          #HW_INT_ERR_A       -- l=Confirm hardware interr
R  #HW_INT_ERR         #HW_INT_ERR         #HW_INT_ERR         -- l=Hardware interrupt recei
                                ved

```

FB400 - <offline>

```

"PROG_ENC_2"      Programa Encoder 2 (FM350-1)
Name:  PROG_E2      Family: FM_CNT_1
Author:  FM          Version: 3.0
                                Block version: 2
Time stamp Code:  07/18/2012 05:08:29 PM
Interface:        11/27/2002 12:00:24 PM
Lengths (block/logic/data): 00926 00710 00008

```

Name	Data Type	Address	Initial Value	Comment
IN		0.0		
SW_GATE	Bool	0.0	FALSE	l=Control software gate
GATE_STP	Bool	0.1	FALSE	l=Stop gate
OT_ERR_A	Bool	0.2	FALSE	l=Confirm operator error
DIAG_ERR_A	Bool	0.3	FALSE	l=Confirm diagnostic interrupt
HW_INT_ERR_A	Bool	0.4	FALSE	l=Confirm hardware interrupt
SET_DO0	Bool	0.5	FALSE	l=Set output DO0
SET_DO1	Bool	0.6	FALSE	l=Set output DO1
SET_L_DIRECT	Bool	0.7	FALSE	l=New counter value
SET_L_PREPAR	Bool	1.0	FALSE	l=Prepare new counter value
SET_T_CMP_V1	Bool	1.1	FALSE	l=New compare value 1
SET_T_CMP_V2	Bool	1.2	FALSE	l=New compare value 2
SET_C_DOPARA	Bool	1.3	FALSE	l=Triggering parameter change
SET_RES_SYNC	Bool	1.4	FALSE	l=Reset status bit "synchronization"
SET_RES_ZERO	Bool	1.5	FALSE	l=Reset status bit "zero-crossing"
ENSET_UP	Bool	1.6	FALSE	l=Enable set in direction up (forward)
ENSET_DN	Bool	1.7	FALSE	l=Enable set in direction down (backward)
CTRL_DO0	Bool	2.0	FALSE	l=Control digital output DO0
CTRL_DO1	Bool	2.1	FALSE	l=Control digital output DO1
L_DIRECT_VAL	DInt	4.0	L#0	New counter value
L_PREPAR_VAL	DInt	8.0	L#0	New preparation value
T_CMP_V1_VAL	DInt	12.0	L#0	New compare value 1
T_CMP_V2_VAL	DInt	16.0	L#0	New compare value 2

Name	Data Type	Address	Initial Value	Comment
DO0_MODE	Byte	20.0	B#16#0	Set DO0 behavior (0..5)
DO1_MODE	Byte	21.0	B#16#0	Set DO1 behavior (0..6)
HYSTERESIS	Byte	22.0	B#16#0	Hysteresis value (0..255)
PULSE_DURATION	Byte	23.0	B#16#0	Pulse duration (0..250)
OUT		0.0		
OT_ERR	Bool	24.0	FALSE	Error in CNT_CTL1 function
DIAG_ERR	Bool	24.1	FALSE	1=Diagnostic interrupt received
HW_INT_ERR	Bool	24.2	FALSE	1=Hardware interrupt received
STS_RUN	Bool	24.3	FALSE	"Counter is running" status
STS_DIR	Bool	24.4	FALSE	"Direction bit" status
STS_ZERO	Bool	24.5	FALSE	"Zero-Crossing" status
STS_OFLW	Bool	24.6	FALSE	"Counter Overflow" status
STS_UFLW	Bool	24.7	FALSE	"Counter Underflow" status
STS_SW_G	Bool	25.0	FALSE	"Software Gate" status
STS_GATE	Bool	25.1	FALSE	"Internal Gate" status
OT_ERR_B	Byte	26.0	B#16#0	Operator error
LATCH_LOAD	DWord	28.0	DW#16#0	Current load value or latch value
ACT_CNTV	DWord	32.0	DW#16#0	Current count value
IN_OUT		0.0		
L_DIRECT	Bool	36.0	FALSE	1=Load new counter value
L_PREPAR	Bool	36.1	FALSE	1=Prepare new counter value
T_CMP_V1	Bool	36.2	FALSE	1=Load new compare value 1
T_CMP_V2	Bool	36.3	FALSE	1=Load new compare value 2
C_DOPARA	Bool	36.4	FALSE	1=Triggering parameter change
RES_SYNC	Bool	36.5	FALSE	1=Reset status bit "synchronization"
RES_ZERO	Bool	36.6	FALSE	1=Reset status bit "zero-crossing"
DIAG_INF	Bool	36.7	FALSE	1=Read diagnostics (only for OB82)
STAT		0.0		
FP_L_DIRECT	Bool	38.0	FALSE	1=Load new counter value
FP_L_PREPAR	Bool	38.1	FALSE	1=Prepare new counter value

Page 2 of 9

Name	Data Type	Address	Initial Value	Comment
FP_T_CMP_V1	Bool	38.2	FALSE	1=Load new compare value 1
FP_T_CMP_V2	Bool	38.3	FALSE	1=Load new compare value 2
FP_C_DOPARA	Bool	38.4	FALSE	1=Triggering parameter change
FP_RES_SYNC	Bool	38.5	FALSE	1=Reset status bit "synchronization"
FP_RES_ZERO	Bool	38.6	FALSE	1=Reset status bit "zero-crossing"
SFC_ERR	Int	40.0	0	SFC51 error status
TEMP		0.0		

Block: FB400 EXAMPLE PROGRAM

Network: 1 Triggering functions

```

*****
In the event of a positive edge in the initiation bit:
- set corresponding bit in the FC402 call
- transfer corresponding value from the instance DB to the channel DB
*****

NOP    0
// *****
// Triggering the function "Direct load"
// *****

A    #SET_L_DIRECT            //Direct load                    #SET_L_DIRECT            -- 1=New counter value
FP   #FP_L_DIRECT            //positive edge                #FP_L_DIRECT            -- 1=Load new counter value
S    #L_DIRECT                #L_DIRECT                    -- 1=Load new counter value
JCN   NEX1
L    #L_DIRECT_VAL            //Download parameters in channel DB    #L_DIRECT_VAL            -- New counter value
T    "DB_UDT_ENC2".LOAD_VAL                                            DB4.DBD14                -- New load value (write user)

// *****
// Triggering the function "Prepare to load"
// *****

NEX1: A    #SET_L_PREPAR            //Prepare to load                #SET_L_PREPAR            -- 1=Prepare new counter value
FP   #FP_L_PREPAR            //positive edge                #FP_L_PREPAR            -- 1=Prepare new counter value
S    #L_PREPAR                #L_PREPAR                    -- 1=Prepare new counter value
JCN   NEX2
L    #L_PREPAR_VAL            //Download parameters in channel DB    #L_PREPAR_VAL            -- New preparation value

```

(Anexo M) 99

Page 3 of 9

```

T      "DB_UDT_ENC2".LOAD_VAL                                DB4.DBD14      -- New load value (write user)
// *****
// Triggering the function "Load compare value 1"
// *****
NEX2: A      #SET_T_CMP_V1      //Transfer compare value 1      #SET_T_CMP_V1      -- 1=New compare value 1
FP      #FP_T_CMP_V1      //positive edge      #FP_T_CMP_V1      -- 1=Load new compare value 1
S      #T_CMP_V1      #T_CMP_V1      -- 1=Load new compare value 1
JCN      NEX3
L      #T_CMP_V1_VAL      //Download parameters in channel DB      #T_CMP_V1_VAL      -- New compare value 1
T      "DB_UDT_ENC2".CMP_V1      DB4.DBD18      -- New comparator value 1 (write user)
// *****
// Triggering the function "Load compare value 2"
// *****
NEX3: A      #SET_T_CMP_V2      //Transfer compare value 2      #SET_T_CMP_V2      -- 1=New compare value 2
FP      #FP_T_CMP_V2      //positive edge      #FP_T_CMP_V2      -- 1=Load new compare value 2
S      #T_CMP_V2      #T_CMP_V2      -- 1=Load new compare value 2
JCN      NEX4
L      #T_CMP_V2_VAL      //Download parameters in channel DB      #T_CMP_V2_VAL      -- New compare value 2
T      "DB_UDT_ENC2".CMP_V2      DB4.DBD22      -- New comparator value 2 (write user)
// *****
// Triggering the function "Triggering parameter change"
//
// Note: do not use together with the jobs "L_DIRECT" or
//       "L_PREPAR" or "T_CMP_V1" or "T_CMP_V2"
//       (otherwise operator error 5 will occur)
// Note: Must be DO0_MODE = 0 for DO1_MODE = 6
//       (otherwise operator error 21 will occur)
//
// The multiple usage of DBD14 is hidden by FB100.
// The use of half bytes in DBB14 is hidden by FB100.
// *****
NEX4: A      #SET_C_DOPARA      #SET_C_DOPARA      -- 1=Triggering parameter change
FP      #FP_C_DOPARA      //positive edge      #FP_C_DOPARA      -- 1=Triggering parameter change
S      #C_DOPARA      #C_DOPARA      -- 1=Triggering parameter change
JCN      NEX5
//Download changed parameters in channel DB
L      #DO0_MODE      //DO0 response      #DO0_MODE      -- Set DO0 behavior (0..5)

```

```

L      W#16#7      //Mask relevant bits
AW
T      DB4.DBB 14
L      #DO1_MODE      //DO1 response      #DO1_MODE      -- Set DO1 behavior (0..6)
L      W#16#7      //Mask relevant bits
AW
SLW      4      //shift to bit 7..4
L      DB4.DBB 14      //OR with DO0_MODE
OW
T      DB4.DBB 14
L      #HYSTERESIS      #HYSTERESIS      -- Hysteresis value (0..255)
T      DB4.DBB 15
L      #PULSE_DURATION      #PULSE_DURATION      -- Pulse duration (0..250)
T      DB4.DBB 16
L      0
T      DB4.DBB 17
// *****
// Triggering the function "Reset status bit synchronization"
// *****
NEX5: A      #SET_RES_SYNC      #SET_RES_SYNC      -- 1=Reset status bit "synchronization"
FP      #FP_RES_SYNC      //positive edge      #FP_RES_SYNC      -- 1=Reset status bit "synchronization"
S      #RES_SYNC      #RES_SYNC      -- 1=Reset status bit "synchronization"
// *****
// Triggering the function "Reset status bit zero-crossing"
// *****
A      #SET_RES_ZERO      #SET_RES_ZERO      -- 1=Reset status bit "zero-crossing"
FP      #FP_RES_ZERO      //positive edge      #FP_RES_ZERO      -- 1=Reset status bit "zero-crossing"
S      #RES_ZERO      #RES_ZERO      -- 1=Reset status bit "zero-crossing"
// *****
// Download enable bits in channel DB
// *****
A      #ENSET_UP      #ENSET_UP      -- 1=Enable set in direction up (forward)

```

```

= "DB_UDT_ENC2".ENSET_UP          DB4.DBX27.0    -- Enable set in direction up
                                   (=forward) (write user)
A #ENSET_DN                       #ENSET_DN      -- 1=Enable set in direction d
                                   own (backward)
= "DB_UDT_ENC2".ENSET_DN          DB4.DBX27.1    -- Enable set in direction dow
                                   n (=backward) (write user)
A #CTRL_DO0                       #CTRL_DO0      -- 1=Control digital output DO
                                   0
= "DB_UDT_ENC2".CTRL_DO0          DB4.DBX28.0    -- Control digital output DO0
                                   (write user)
A #CTRL_D01                       #CTRL_D01      -- 1=Control digital output DO
                                   1
= "DB_UDT_ENC2".CTRL_D01          DB4.DBX28.1    -- Control digital output DO1
                                   (write user)

```

Network: 2 Call FC

```

*****
Call standard function "Control FM 350-1/450-1"
*****

```

```

CALL "CNT_CTL1_ENC2" //Control FM 350-1/450-1          FC402          -- Control counter module FM 350-
1 Encoder 2
DB_NO :=4 //Data block for channel 1
SW_GATE :=#SW_GATE //Control software gate            #SW_GATE       -- 1=Control software gate
GATE_STP:=#GATE_STP //Stop internal gate              #GATE_STP      -- 1=Stop gate
OT_ERR_A:=#OT_ERR_A //Confirm operator error          #OT_ERR_A      -- 1=Confirm operator error
SET_DO0 :=#SET_DO0 //Set output DO0                  #SET_DO0       -- 1=Set output DO0
SET_D01 :=#SET_D01 //Set output DO1                  #SET_D01       -- 1=Set output DO1
OT_ERR :=#OT_ERR //Operator error occurred            #OT_ERR        -- Error in CNT_CTL1 function
L_DIRECT:=#L_DIRECT //Load new counter value          #L_DIRECT      -- 1=Load new counter value
L_PREPAR:=#L_PREPAR //Prepare new counter value       #L_PREPAR      -- 1=Prepare new counter value
T_CMP_V1:=#T_CMP_V1 //Load new compare value 1       #T_CMP_V1      -- 1=Load new compare value 1
T_CMP_V2:=#T_CMP_V2 //Load new compare value 2       #T_CMP_V2      -- 1=Load new compare value 2
C_DOPARA:=#C_DOPARA //Triggering parameter change    #C_DOPARA      -- 1=Triggering parameter change
RES_SYNC:=#RES_SYNC //Reset synchronization bit      #RES_SYNC      -- 1=Reset status bit "synchroniza
tion"
RES_ZERO:=#RES_ZERO //Reset zero-crossing bit        #RES_ZERO      -- 1=Reset status bit "zero-cross
ing"

```

Page 6 of 9

Network: 3 Post-editing

```

*****

```

NOP 0

```

// *****
// Reset edge memory bit
// *****

```

```

AN #SET_L_DIRECT          #SET_L_DIRECT    -- 1=New counter value
AN #L_DIRECT              #L_DIRECT        -- 1=Load new counter value
R #FP_L_DIRECT            #FP_L_DIRECT    -- 1=Load new counter value

AN #SET_L_PREPAR          #SET_L_PREPAR    -- 1=Prepare new counter valu
e
AN #L_PREPAR              #L_PREPAR        -- 1=Prepare new counter valu
e
R #FP_L_PREPAR            #FP_L_PREPAR    -- 1=Prepare new counter valu
e

AN #SET_T_CMP_V1          #SET_T_CMP_V1    -- 1=New compare value 1
AN #T_CMP_V1              #T_CMP_V1      -- 1=Load new compare value 1
R #FP_T_CMP_V1            #FP_T_CMP_V1    -- 1=Load new compare value 1

AN #SET_T_CMP_V2          #SET_T_CMP_V2    -- 1=New compare value 2
AN #T_CMP_V2              #T_CMP_V2      -- 1=Load new compare value 2
R #FP_T_CMP_V2            #FP_T_CMP_V2    -- 1=Load new compare value 2

AN #SET_C_DOPARA          #SET_C_DOPARA    -- 1=Triggering parameter cha
nge
AN #C_DOPARA              #C_DOPARA      -- 1=Triggering parameter cha
nge
R #FP_C_DOPARA            #FP_C_DOPARA    -- 1=Triggering parameter cha
nge

AN #SET_RES_SYNC          #SET_RES_SYNC    -- 1=Reset status bit "sychro
nization"
AN #RES_SYNC              #RES_SYNC      -- 1=Reset status bit "sychro
nization"
R #FP_RES_SYNC            #FP_RES_SYNC    -- 1=Reset status bit "sychro
nization"

AN #SET_RES_ZERO          #SET_RES_ZERO    -- 1=Reset status bit "zero-c
rossing"

```

(Anexo M) 101

Page 7 of 9


```

AN  #RES_ZERO                                #RES_ZERO      -- l=Reset status bit "zero-c
R   #FP_RES_ZERO                             #FP_RES_ZERO   -- l=Reset status bit "zero-c
                                        #RES_ZERO      -- l=Reset status bit "zero-c
                                        #FP_RES_ZERO   -- l=Reset status bit "zero-c
                                        crossing"
                                        crossing"

// *****
// Display status messages
// *****

A   "DB_UDT_ENC2".STS_RUN                    DB4.DBX43.0    -- Status counter is running
=   #STS_RUN                                (read user)
                                        #STS_RUN      -- "Counter is running" statu
                                        s
A   "DB_UDT_ENC2".STS_DIR                    DB4.DBX43.1    -- Status of the counter dire
=   #STS_DIR                                ction bit (read user)
                                        #STS_DIR      -- "Direction bit" status
A   "DB_UDT_ENC2".STS_ZERO                  DB4.DBX43.2    -- Status of the counter zero
=   #STS_ZERO                               crossing bit (read user)
                                        #STS_ZERO     -- "Zero-Crossing" status
A   "DB_UDT_ENC2".STS_OFLW                  DB4.DBX43.3    -- Status counter passed the
=   #STS_OFLW                               overflow value (read user)
                                        #STS_OFLW    -- "Counter Overflow" status
A   "DB_UDT_ENC2".STS_UFLW                  DB4.DBX43.4    -- Status counter passed the
=   #STS_UFLW                               underflow value (read user)
                                        #STS_UFLW    -- "Counter Underflow" statu
                                        s
A   "DB_UDT_ENC2".STS_SW_G                  DB4.DBX43.7    -- Status of the software gat
=   #STS_SW_G                               e (read user)
                                        #STS_SW_G    -- "Software Gate" status
A   "DB_UDT_ENC2".STS_GATE                  DB4.DBX43.6    -- Status of the internal gat
=   #STS_GATE                               e (read user)
                                        #STS_GATE    -- "Internal Gate" status

L   "DB_UDT_ENC2".OT_ERR_B                  DB4.DBB40      -- Operator error byte (read
T   #OT_ERR_B                               user)
                                        #OT_ERR_B    -- Operator error

L   "DB_UDT_ENC2".LATCH_LOAD                DB4.DBD30      -- Actual latch or load value
T   #LATCH_LOAD                             (read user)
                                        #LATCH_LOAD  -- Current load value or latc
                                        h value
L   "DB_UDT_ENC2".ACT_CNTV                  DB4.DBD34      -- Actual counter value (read
T   #ACT_CNTV                               user)
                                        #ACT_CNTV    -- Current count value

// *****
// Confirm interrupts
// *****

```

```

A   #DIAG_ERR_A                            //Confirm diagnostic interrupt #DIAG_ERR_A   -- l=Confirm diagnostic inter
R   #DIAG_ERR                               #DIAG_ERR     -- l=Diagnostic interrupt rec
                                        eived

A   #HW_INT_ERR_A                           //Confirm hardware interrupt  #HW_INT_ERR_A   -- l=Confirm hardware interru
R   #HW_INT_ERR                             #HW_INT_ERR   -- l=Hardware interrupt recei
                                        ved

```

DB1 - <offline>
"DB_UDT_M1"

Data view

Data block type:

UDT1 data block

Name: DB_UDTM1
 Author: SIEMENS
 Family:
 Version: 0.1
 Block version: 2
 Lengths (Block / Data): 01890 / 00516
 Time stamp
 Code: 07/18/2012 04:53:46 PM
 Interface: 07/04/2000 03:52:02 PM

Comment:

Address	Name	Type	Initial value	Actual value	Comment
0.0	MOD_ADR	INT	0	352	Module address
2.0	CH_NO	INT	0	1	Channel number
4.0	CH_ADR	DWORD	DW#16#0	DW#16#B00	Channel address
8.0	DS_OFFS	INT	0	0	data block offset
10.0	b_10	WORD	W#16#0	W#16#0	
12.0	b_12	BYTE	B#16#0	B#16#0	
13.0	b_13_0	BOOL	FALSE	FALSE	
13.1	b_13_1	BOOL	FALSE	FALSE	
13.2	b_13_2	BOOL	FALSE	FALSE	
13.3	b_13_3	BOOL	FALSE	FALSE	
13.4	b_13_4	BOOL	FALSE	FALSE	
13.5	b_13_5	BOOL	FALSE	FALSE	
13.6	MODE_BUSY	BOOL	FALSE	FALSE	Mode busy
13.7	POS_REACHED	BOOL	FALSE	FALSE	Position reached
14.0	b_14_0	BOOL	FALSE	FALSE	
14.1	TEST_EN	BOOL	FALSE	FALSE	test enable
14.2	b_14_2	BOOL	FALSE	FALSE	
14.3	OT_ERR_A	BOOL	FALSE	FALSE	quit error
14.4	b_14_4	BOOL	FALSE	FALSE	

Page 1 of 21

Address	Name	Type	Initial value	Actual value	Comment
14.5	b_14_5	BOOL	FALSE	FALSE	
14.6	b_14_6	BOOL	FALSE	FALSE	
14.7	b_14_7	BOOL	FALSE	FALSE	
15.0	START	BOOL	FALSE	FALSE	start
15.1	STOP	BOOL	FALSE	FALSE	stop
15.2	DIR_M	BOOL	FALSE	FALSE	direction minus
15.3	DIR_P	BOOL	FALSE	FALSE	direction plus
15.4	ACK_MF	BOOL	FALSE	FALSE	acknowledge M function
15.5	READ_EN	BOOL	FALSE	FALSE	read enable
15.6	SKIP_BLK	BOOL	FALSE	FALSE	skip blank
15.7	DRV_EN	BOOL	FALSE	FALSE	drive enable
16.0	MODE_IN	BYTE	B#16#0	B#16#0	mode
17.0	MODE_TYPE	BYTE	B#16#0	B#16#0	mode parameter
18.0	OVERRIDE	BYTE	B#16#0	B#16#0	override
19.0	b_19	BYTE	B#16#0	B#16#0	
20.0	b_20	WORD	W#16#0	W#16#0	
22.0	b_22_0	BOOL	FALSE	FALSE	
22.1	TST_STAT	BOOL	FALSE	FALSE	test state
22.2	b_22_2	BOOL	FALSE	FALSE	
22.3	OT_ERR	BOOL	FALSE	FALSE	ot-error
22.4	DATA_ERR	BOOL	FALSE	FALSE	data-error
22.5	b_22_5	BOOL	FALSE	FALSE	
22.6	b_22_6	BOOL	FALSE	FALSE	
22.7	PARA	BOOL	FALSE	FALSE	parameterized
23.0	ST_ENBLD	BOOL	FALSE	FALSE	start enable
23.1	WORKING	BOOL	FALSE	FALSE	working
23.2	WAIT_EI	BOOL	FALSE	FALSE	wait for external enable
23.3	b_23_3	BOOL	FALSE	FALSE	
23.4	b_23_4	BOOL	FALSE	FALSE	
23.5	DT_RUN	BOOL	FALSE	FALSE	dwelt time running
23.6	PR_BACK	BOOL	FALSE	FALSE	program run backward
23.7	b_23_7	BOOL	FALSE	FALSE	
24.0	MODE_OUT	BYTE	B#16#0	B#16#0	mode
25.0	SYNC	BOOL	FALSE	FALSE	synchronized
25.1	MSR_DONE	BOOL	FALSE	FALSE	measurement done
25.2	GO_M	BOOL	FALSE	FALSE	go_minus
25.3	GO_P	BOOL	FALSE	FALSE	go_plus

Address	Name	Type	Initial value	Actual value	Comment
25.4	ST_SERVO	BOOL	FALSE	FALSE	status servo
25.5	FVAL_DONE	BOOL	FALSE	FALSE	flying actual value done
25.6	b_25_6	BOOL	FALSE	FALSE	
25.7	POS_RCD	BOOL	FALSE	FALSE	position is reached, hold
26.0	NUM_MF	BYTE	B#16#0	B#16#0	number M function
27.0	b_27_0	BOOL	FALSE	FALSE	
27.1	b_27_1	BOOL	FALSE	FALSE	
27.2	b_27_2	BOOL	FALSE	FALSE	
27.3	b_27_3	BOOL	FALSE	FALSE	
27.4	STR_MF	BOOL	FALSE	FALSE	strobe signal for M functions
27.5	b_27_5	BOOL	FALSE	FALSE	
27.6	b_27_6	BOOL	FALSE	FALSE	
27.7	b_27_7	BOOL	FALSE	FALSE	
28.0	b_28	DWORD	DW#16#0	DW#16#0	
32.0	b_32	WORD	W#16#0	W#16#0	
34.0	SERVO_EN	BOOL	FALSE	FALSE	servo enable
34.1	GAUG_FLY	BOOL	FALSE	FALSE	gauging on the fly
34.2	b_34_2	BOOL	FALSE	FALSE	
34.3	b_34_3	BOOL	FALSE	FALSE	
34.4	b_34_4	BOOL	FALSE	FALSE	
34.5	TRAV_MON	BOOL	FALSE	FALSE	traverse monitoring
34.6	PARK_AX	BOOL	FALSE	FALSE	parking axis
34.7	SIM_ON	BOOL	FALSE	FALSE	simulation on
35.0	b_35_0	BOOL	FALSE	FALSE	
35.1	b_35_1	BOOL	FALSE	FALSE	
35.2	MSR_EN	BOOL	FALSE	FALSE	measurement
35.3	REF_TRIG	BOOL	FALSE	FALSE	trigger reference point
35.4	DI_OFF	BOOL	FALSE	FALSE	disable enable input
35.5	FOLLOWUP	BOOL	FALSE	FALSE	follow-up
35.6	SSW_DIS	BOOL	FALSE	FALSE	software switch disable
35.7	DRIFT_OFF	BOOL	FALSE	FALSE	automatic drift compensation off
36.0	b_36	BYTE	B#16#0	B#16#0	
37.0	MD_EN	BOOL	FALSE	FALSE	machine data enable
37.1	DELDIST_EN	BOOL	FALSE	FALSE	delete distance to go
37.2	SEARCH_F	BOOL	FALSE	FALSE	automatic block search forward
37.3	SEARCH_B	BOOL	FALSE	FALSE	automatic block search backward
37.4	b_37_4	BOOL	FALSE	FALSE	

Address	Name	Type	Initial value	Actual value	Comment
37.5	RESET_AX	BOOL	FALSE	FALSE	reset axis
37.6	AVALREM_EN	BOOL	FALSE	FALSE	remove setting actual value
37.7	b_37_7	BOOL	FALSE	FALSE	
38.0	VLEV_EN	BOOL	FALSE	FALSE	velocity levels 1 and 2
38.1	CLEV_EN	BOOL	FALSE	FALSE	control levels 1 and 2
38.2	TRG254_EN	BOOL	FALSE	FALSE	target 254
38.3	MDI_EN	BOOL	FALSE	FALSE	MDI movement block
38.4	MDIFLY_EN	BOOL	FALSE	FALSE	MDI block on the fly
38.5	b_38_5	BOOL	FALSE	FALSE	
38.6	REFPT_EN	BOOL	FALSE	FALSE	reference coordinate
38.7	AVAL_EN	BOOL	FALSE	FALSE	actual value coordinate
39.0	FVAL_EN	BOOL	FALSE	FALSE	actual value coordinate (flying)
39.1	ZOFF_EN	BOOL	FALSE	FALSE	zero offset value
39.2	b_39_2	BOOL	FALSE	FALSE	
39.3	PARCH_EN	BOOL	FALSE	FALSE	parameter change
39.4	DIGO_EN	BOOL	FALSE	FALSE	digital 0
39.5	PROGS_EN	BOOL	FALSE	FALSE	part program selection
39.6	REQAPP_EN	BOOL	FALSE	FALSE	request application data
39.7	TEACHIN_EN	BOOL	FALSE	FALSE	Teach In
40.0	AXCOU_EN	BOOL	FALSE	FALSE	write job axis coupling FM453
40.1	b_40_1	BOOL	FALSE	FALSE	
40.2	b_40_2	BOOL	FALSE	FALSE	
40.3	b_40_3	BOOL	FALSE	FALSE	
40.4	b_40_4	BOOL	FALSE	FALSE	
40.5	b_40_5	BOOL	FALSE	FALSE	
40.6	b_40_6	BOOL	FALSE	FALSE	
40.7	b_40_7	BOOL	FALSE	FALSE	
41.0	b_41	BYTE	B#16#0	B#16#0	
42.0	OPDAT_EN	BOOL	FALSE	FALSE	operating_data
42.1	ACT_BL_EN	BOOL	FALSE	FALSE	actual part program block
42.2	NXT_BL_EN	BOOL	FALSE	FALSE	next part program block
42.3	BLEXT_EN	BOOL	FALSE	FALSE	actual value at external block change
42.4	SERVDAT_EN	BOOL	FALSE	FALSE	service data
42.5	OC_ERR_EN	BOOL	FALSE	FALSE	oc-error
42.6	b_42_6	BOOL	FALSE	FALSE	
42.7	b_42_7	BOOL	FALSE	FALSE	
43.0	AXCOURD_EN	BOOL	FALSE	FALSE	read job axis coupling FM453

Address	Name	Type	Initial value	Actual value	Comment
43.1	b_43_1	BOOL	FALSE	FALSE	
43.2	b_43_2	BOOL	FALSE	FALSE	
43.3	PARRD_EN	BOOL	FALSE	FALSE	parameter data
43.4	DIGIO_EN	BOOL	FALSE	FALSE	digital in_out
43.5	OPDAT1_EN	BOOL	FALSE	FALSE	operating data 1
43.6	APPDAT_EN	BOOL	FALSE	FALSE	applying data
43.7	MSRRD_EN	BOOL	FALSE	FALSE	read measurement values
44.0	SERVO_D	BOOL	FALSE	FALSE	servo enable
44.1	GAUG_FLY_D	BOOL	FALSE	FALSE	gauging on the fly
44.2	b_44_2	BOOL	FALSE	FALSE	
44.3	b_44_3	BOOL	FALSE	FALSE	
44.4	b_44_4	BOOL	FALSE	FALSE	
44.5	TRAV_MON_D	BOOL	FALSE	FALSE	traverse monitoring
44.6	PARK_AX_D	BOOL	FALSE	FALSE	parking axis
44.7	SIM_ON_D	BOOL	FALSE	FALSE	simulation on
45.0	b_45_0	BOOL	FALSE	FALSE	
45.1	b_45_1	BOOL	FALSE	FALSE	
45.2	MSR_D	BOOL	FALSE	FALSE	measurement
45.3	REF_TRIG_D	BOOL	FALSE	FALSE	trigger reference point
45.4	DI_OFF_D	BOOL	FALSE	FALSE	disable enable input
45.5	FOLLOWUP_D	BOOL	FALSE	FALSE	follow-up
45.6	SSW_DIS_D	BOOL	FALSE	FALSE	software switch disable
45.7	DRIFT_OFF_D	BOOL	FALSE	FALSE	automatic drift compensation off
46.0	b_46	BYTE	B#16#0	B#16#0	
47.0	MD_D	BOOL	FALSE	FALSE	machine data enable
47.1	DELDIST_D	BOOL	FALSE	FALSE	delete distance to go
47.2	SEARCH_F_D	BOOL	FALSE	FALSE	automatic block search forward
47.3	SEARCH_B_D	BOOL	FALSE	FALSE	automatic block search backward
47.4	b_47_4	BOOL	FALSE	FALSE	
47.5	RESET_AX_D	BOOL	FALSE	FALSE	reset axis
47.6	AVALREM_D	BOOL	FALSE	FALSE	remove setting actual value
47.7	b_47_7	BOOL	FALSE	FALSE	
48.0	VLEV_D	BOOL	FALSE	FALSE	velocity levels 1 and 2
48.1	CLEV_D	BOOL	FALSE	FALSE	control levels 1 and 2
48.2	TRG254_D	BOOL	FALSE	FALSE	target 254
48.3	MDI_D	BOOL	FALSE	FALSE	MDI movement block
48.4	MDIFLY_D	BOOL	FALSE	FALSE	MDI block on the fly

Page 5 of 21

Address	Name	Type	Initial value	Actual value	Comment
48.5	b_48_5	BOOL	FALSE	FALSE	
48.6	REFPT_D	BOOL	FALSE	FALSE	reference coordinate
48.7	AVAL_D	BOOL	FALSE	FALSE	actual value coordinate
49.0	FVAL_D	BOOL	FALSE	FALSE	actual value coordinate (flying)
49.1	ZOFF_D	BOOL	FALSE	FALSE	zero offset value
49.2	b_49_2	BOOL	FALSE	FALSE	
49.3	PARCH_D	BOOL	FALSE	FALSE	parameter change
49.4	DIGO_D	BOOL	FALSE	FALSE	digital 0
49.5	PROGS_D	BOOL	FALSE	FALSE	part program selection
49.6	REQAPP_D	BOOL	FALSE	FALSE	request application data
49.7	TEACHIN_D	BOOL	FALSE	FALSE	Teach In
50.0	AXCOU_D	BOOL	FALSE	FALSE	axis coupling FM453
50.1	b_50_1	BOOL	FALSE	FALSE	
50.2	b_50_2	BOOL	FALSE	FALSE	
50.3	b_50_3	BOOL	FALSE	FALSE	
50.4	b_50_4	BOOL	FALSE	FALSE	
50.5	b_50_5	BOOL	FALSE	FALSE	
50.6	b_50_6	BOOL	FALSE	FALSE	
50.7	b_50_7	BOOL	FALSE	FALSE	
51.0	b_51	BYTE	B#16#0	B#16#0	
52.0	OPDAT_D	BOOL	FALSE	FALSE	operating_data
52.1	ACT_BL_D	BOOL	FALSE	FALSE	actual part program block
52.2	NXT_BL_D	BOOL	FALSE	FALSE	next part program block
52.3	BLEXT_D	BOOL	FALSE	FALSE	actual value at external block change
52.4	SERVDAT_D	BOOL	FALSE	FALSE	service data
52.5	OC_ERR_D	BOOL	FALSE	FALSE	oc_error
52.6	OT_ERR_D	BOOL	FALSE	FALSE	ot_error
52.7	DA_ERR_D	BOOL	FALSE	FALSE	data error
53.0	AXCOURD_D	BOOL	FALSE	FALSE	axis coupling FM453
53.1	b_53_1	BOOL	FALSE	FALSE	
53.2	b_53_2	BOOL	FALSE	FALSE	
53.3	PARRD_D	BOOL	FALSE	FALSE	parameter data
53.4	DIGIO_D	BOOL	FALSE	FALSE	digital in_out
53.5	OPDAT1_D	BOOL	FALSE	FALSE	operating data 1
53.6	APPDAT_D	BOOL	FALSE	FALSE	applying data
53.7	MSRRD_D	BOOL	FALSE	FALSE	read measurement values
54.0	SERVO_ERR	BOOL	FALSE	FALSE	servo enable

(Anexo M) 105

Page 6 of 21

Address	Name	Type	Initial value	Actual value	Comment
54.1	GAUG_FLY_ERR	BOOL	FALSE	FALSE	gauging on the fly
54.2	b_54_2	BOOL	FALSE	FALSE	
54.3	b_54_3	BOOL	FALSE	FALSE	
54.4	b_54_4	BOOL	FALSE	FALSE	
54.5	TRAV_MON_ERR	BOOL	FALSE	FALSE	traverse monitoring
54.6	PARK_AX_ERR	BOOL	FALSE	FALSE	parking axis
54.7	SIM_ON_ERR	BOOL	FALSE	FALSE	simulation on
55.0	b_55_0	BOOL	FALSE	FALSE	
55.1	b_55_1	BOOL	FALSE	FALSE	
55.2	MSR_ERR	BOOL	FALSE	FALSE	measurement
55.3	REF_TRIG_ERR	BOOL	FALSE	FALSE	trigger reference point
55.4	DI_OFF_ERR	BOOL	FALSE	FALSE	disable enable input
55.5	FOLLOWUP_ERR	BOOL	FALSE	FALSE	follow-up
55.6	SSW_DIS_ERR	BOOL	FALSE	FALSE	software switch disable
55.7	DRIFT_OFF_ERR	BOOL	FALSE	FALSE	automatic drift compensation off
56.0	b_56	BYTE	B#16#0	B#16#0	
57.0	MD_ERR	BOOL	FALSE	FALSE	machine data enable
57.1	DELDIST_ERR	BOOL	FALSE	FALSE	delete distance to go
57.2	SEARCH_F_ERR	BOOL	FALSE	FALSE	automatic block search forward
57.3	SEARCH_B_ERR	BOOL	FALSE	FALSE	automatic block search backward
57.4	b_57_4	BOOL	FALSE	FALSE	
57.5	RESET_AX_ERR	BOOL	FALSE	FALSE	reset axis
57.6	AVALREM_ERR	BOOL	FALSE	FALSE	remove setting actual value
57.7	b_57_7	BOOL	FALSE	FALSE	
58.0	VLEV_ERR	BOOL	FALSE	FALSE	velocity levels 1 and 2
58.1	CLEV_ERR	BOOL	FALSE	FALSE	control levels 1 and 2
58.2	TRG254_ERR	BOOL	FALSE	FALSE	target 254
58.3	MDI_ERR	BOOL	FALSE	FALSE	MDI movement block
58.4	MDIFLY_ERR	BOOL	FALSE	FALSE	MDI block on the fly
58.5	b_58_5	BOOL	FALSE	FALSE	
58.6	REFPT_ERR	BOOL	FALSE	FALSE	reference coordinate
58.7	AVAL_ERR	BOOL	FALSE	FALSE	actual value coordinate
59.0	FVAL_ERR	BOOL	FALSE	FALSE	actual value coordinate (flying)
59.1	ZOFF_ERR	BOOL	FALSE	FALSE	zero offset value
59.2	b_59_2	BOOL	FALSE	FALSE	
59.3	PARCH_ERR	BOOL	FALSE	FALSE	parameter change
59.4	DIGO_ERR	BOOL	FALSE	FALSE	digital 0

Address	Name	Type	Initial value	Actual value	Comment
59.5	PROGS_ERR	BOOL	FALSE	FALSE	part program selection
59.6	REQAPP_ERR	BOOL	FALSE	FALSE	request application data
59.7	TEACHIN_ERR	BOOL	FALSE	FALSE	Teach In
60.0	AXCOU_ERR	BOOL	FALSE	FALSE	axis coupling write error FM453
60.1	b_60_1	BOOL	FALSE	FALSE	
60.2	b_60_2	BOOL	FALSE	FALSE	
60.3	b_60_3	BOOL	FALSE	FALSE	
60.4	b_60_4	BOOL	FALSE	FALSE	
60.5	b_60_5	BOOL	FALSE	FALSE	
60.6	b_60_6	BOOL	FALSE	FALSE	
60.7	b_60_7	BOOL	FALSE	FALSE	
61.0	b_61	BYTE	B#16#0	B#16#0	
62.0	OPDAT_ERR	BOOL	FALSE	FALSE	operating_data
62.1	ACT_BL_ERR	BOOL	FALSE	FALSE	actual part program block
62.2	NXT_BL_ERR	BOOL	FALSE	FALSE	next part program block
62.3	BLEXT_ERR	BOOL	FALSE	FALSE	actual value at external block change
62.4	SERVDAT_ERR	BOOL	FALSE	FALSE	service data
62.5	OC_ERR_ERR	BOOL	FALSE	FALSE	oc_error
62.6	OT_ERR_ERR	BOOL	FALSE	FALSE	ot_error
62.7	DA_ERR_ERR	BOOL	FALSE	FALSE	data error
63.0	AXCOURD_ERR	BOOL	FALSE	FALSE	axis coupling read error FM453
63.1	b_63_1	BOOL	FALSE	FALSE	
63.2	b_63_2	BOOL	FALSE	FALSE	
63.3	PARRD_ERR	BOOL	FALSE	FALSE	parameter data
63.4	DIGIO_ERR	BOOL	FALSE	FALSE	digital in_out
63.5	OPDAT1_ERR	BOOL	FALSE	FALSE	operating data 1
63.6	APPDAT_ERR	BOOL	FALSE	FALSE	applying data
63.7	MSRRD_ERR	BOOL	FALSE	FALSE	read measurement values
64.0	b_64	WORD	W#16#0	W#16#0	
66.0	JOB_ERR	INT	0	0	error-code SFC58/59 (FC POS_CTRL)
68.0	JOBBUSY_WR	BOOL	FALSE	FALSE	job aktiv write
68.1	IMPO_WR	BOOL	FALSE	FALSE	imposs write
68.2	JOBBUSY_RD	BOOL	FALSE	FALSE	job aktiv read
68.3	IMPO_RD	BOOL	FALSE	FALSE	imposs read
68.4	b_68_4	BOOL	FALSE	FALSE	
68.5	b_68_5	BOOL	FALSE	FALSE	
68.6	b_68_6	BOOL	FALSE	FALSE	

Address	Name	Type	Initial value	Actual value	Comment
68.7	b_68_7	BOOL	FALSE	FALSE	
69.0	b_69_0	BOOL	FALSE	FALSE	
69.1	JOBRESET	BOOL	FALSE	FALSE	Reset_ERR and _D
69.2	b_69_2	BOOL	FALSE	FALSE	
69.3	b_69_3	BOOL	FALSE	FALSE	
69.4	b_69_4	BOOL	FALSE	FALSE	
69.5	b_69_5	BOOL	FALSE	FALSE	
69.6	b_69_6	BOOL	FALSE	FALSE	
69.7	b_69_7	BOOL	FALSE	FALSE	
70.0	MDL_DEFFECT	BOOL	FALSE	FALSE	Module defective
70.1	INT_FAULT	BOOL	FALSE	FALSE	Internal fault
70.2	EXT_FAULT	BOOL	FALSE	FALSE	External fault
70.3	PNT_INFO	BOOL	FALSE	FALSE	Point information
70.4	b_70_4	BOOL	FALSE	FALSE	
70.5	FLD_CONNCTR	BOOL	FALSE	FALSE	Field wiring connector missing
70.6	NO_CONFIG	BOOL	FALSE	FALSE	Module has no configuration data
70.7	b_70_7	BOOL	FALSE	FALSE	
71.0	MDL_TYPE	BYTE	B#16#0	B#16#0	Type of module
72.0	b_72_0	BOOL	FALSE	FALSE	
72.1	COMM_FAULT	BOOL	FALSE	FALSE	Communication fault
72.2	b_72_2	BOOL	FALSE	FALSE	
72.3	WTCH_DOG_FLT	BOOL	FALSE	FALSE	Watch dog timer stopped module
72.4	INT_PS_FLT	BOOL	FALSE	FALSE	Internal power supply fault
72.5	b_72_5	BOOL	FALSE	FALSE	
72.6	b_72_6	BOOL	FALSE	FALSE	
72.7	b_72_7	BOOL	FALSE	FALSE	
73.0	b_73_0	BOOL	FALSE	FALSE	
73.1	b_73_1	BOOL	FALSE	FALSE	
73.2	EPROM_FLT	BOOL	FALSE	FALSE	EPROM fault
73.3	RAM_FLT	BOOL	FALSE	FALSE	RAM fault
73.4	b_73_4	BOOL	FALSE	FALSE	
73.5	b_73_5	BOOL	FALSE	FALSE	
73.6	HW_INTR_FLT	BOOL	FALSE	FALSE	Hardware interrupt input in fault
73.7	b_73_7	BOOL	FALSE	FALSE	
74.0	POS_ID	BYTE	B#16#0	B#16#0	channel type info
75.0	LEN_INFO	BYTE	B#16#0	B#16#0	channel info length
76.0	CHEN_NO	BYTE	B#16#0	B#16#0	number of channels

Address	Name	Type	Initial value	Actual value	Comment
77.0	CH_ERR_VE1	BOOL	FALSE	FALSE	channel error vector 1
77.1	CH_ERR_VE2	BOOL	FALSE	FALSE	channel error vector 2
77.2	CH_ERR_VE3	BOOL	FALSE	FALSE	channel error vector 3
77.3	b_77_3	BOOL	FALSE	FALSE	
77.4	b_77_4	BOOL	FALSE	FALSE	
77.5	b_77_5	BOOL	FALSE	FALSE	
77.6	b_77_6	BOOL	FALSE	FALSE	
77.7	b_77_7	BOOL	FALSE	FALSE	
78.0	CAB_BR1	BOOL	FALSE	FALSE	channel 1 cablebraek
78.1	ERR_ABE1	BOOL	FALSE	FALSE	channel 1 error sensor_absolut
78.2	ERR_PU1	BOOL	FALSE	FALSE	channel 1 error impulse
78.3	VO_ENC1	BOOL	FALSE	FALSE	channel 1 voltage control sensor
78.4	VO_15_1	BOOL	FALSE	FALSE	channel 1 voltage control 15 V
78.5	VO_DIO1	BOOL	FALSE	FALSE	channel 1 voltage control digital output
78.6	b_78_6	BOOL	FALSE	FALSE	
78.7	OC_ERR_EN1	BOOL	FALSE	FALSE	channel 1 operation error en1
79.0	b_79	BYTE	B#16#0	B#16#0	
80.0	CAB_BR2	BOOL	FALSE	FALSE	channel 2 cablebraek
80.1	ERR_ABE2	BOOL	FALSE	FALSE	channel 2 error sensor_absolut
80.2	ERR_PU2	BOOL	FALSE	FALSE	channel 2 error impulse
80.3	VO_ENC2	BOOL	FALSE	FALSE	channel 2 voltage control sensor
80.4	VO_15_2	BOOL	FALSE	FALSE	channel 2 voltage control 15 V
80.5	VO_DIO2	BOOL	FALSE	FALSE	channel 2 voltage control digital output
80.6	b_80_6	BOOL	FALSE	FALSE	
80.7	OC_ERR_EN2	BOOL	FALSE	FALSE	channel 2 operation error en2
81.0	b_81	BYTE	B#16#0	B#16#0	
82.0	CAB_BR3	BOOL	FALSE	FALSE	channel 3 cablebraek
82.1	ERR_ABE3	BOOL	FALSE	FALSE	channel 3 error sensor_absolut
82.2	ERR_PU3	BOOL	FALSE	FALSE	channel 3 error impulse
82.3	VO_ENC3	BOOL	FALSE	FALSE	channel 3 voltage control sensor
82.4	VO_15_3	BOOL	FALSE	FALSE	channel 3 voltage control 15 V
82.5	VO_DIO3	BOOL	FALSE	FALSE	channel 3 voltage control digital output
82.6	b_82_6	BOOL	FALSE	FALSE	
82.7	OC_ERR_EN3	BOOL	FALSE	FALSE	channel 3 operation error en3
83.0	b_83	BYTE	B#16#0	B#16#0	
84.0	b_84	WORD	W#16#0	W#16#0	
86.0	OC_ERR_NO	BYTE	B#16#0	B#16#0	operation error, detail event number

Address	Name	Type	Initial value	Actual value	Comment
87.0	OC_REE_CL	BYTE	B#16#0	B#16#0	detail event class
88.0	b_88	WORD	W#16#0	W#16#0	
90.0	OT_ERR_NO	BYTE	B#16#0	B#16#0	control/process error, detail event number
91.0	OT_ERR_CL	BYTE	B#16#0	B#16#0	detail event class
92.0	b_92	WORD	W#16#0	W#16#0	
94.0	DA_ERR_NO	BYTE	B#16#0	B#16#0	data error, detail event number
95.0	DA_ERR_CL	BYTE	B#16#0	B#16#0	detail event class
96.0	DIAG_ERR	INT	0	0	error-code SFC51 (FC POS_DIAG)
98.0	MSRM_ERR	INT	0	0	error-code SFC59 (FC POS_MSRM)
100.0	feld1_intern[100]	BYTE	B#16#0	B#16#0	
101.0	feld1_intern[101]	BYTE	B#16#0	B#16#0	
102.0	feld1_intern[102]	BYTE	B#16#0	B#16#0	
103.0	feld1_intern[103]	BYTE	B#16#0	B#16#0	
104.0	feld1_intern[104]	BYTE	B#16#0	B#16#0	
105.0	feld1_intern[105]	BYTE	B#16#0	B#16#0	
106.0	feld1_intern[106]	BYTE	B#16#0	B#16#0	
107.0	feld1_intern[107]	BYTE	B#16#0	B#16#0	
108.0	feld1_intern[108]	BYTE	B#16#0	B#16#0	
109.0	feld1_intern[109]	BYTE	B#16#0	B#16#0	
110.0	feld1_intern[110]	BYTE	B#16#0	B#16#0	
111.0	feld1_intern[111]	BYTE	B#16#0	B#16#0	
112.0	feld1_intern[112]	BYTE	B#16#0	B#16#0	
113.0	feld1_intern[113]	BYTE	B#16#0	B#16#0	
114.0	feld1_intern[114]	BYTE	B#16#0	B#16#0	
115.0	feld1_intern[115]	BYTE	B#16#0	B#16#0	
116.0	feld1_intern[116]	BYTE	B#16#0	B#16#0	
117.0	feld1_intern[117]	BYTE	B#16#0	B#16#0	
118.0	feld1_intern[118]	BYTE	B#16#0	B#16#0	
119.0	feld1_intern[119]	BYTE	B#16#0	B#16#0	
120.0	feld1_intern[120]	BYTE	B#16#0	B#16#0	
121.0	feld1_intern[121]	BYTE	B#16#0	B#16#0	
122.0	feld1_intern[122]	BYTE	B#16#0	B#16#0	
123.0	feld1_intern[123]	BYTE	B#16#0	B#16#0	
124.0	feld1_intern[124]	BYTE	B#16#0	B#16#0	
125.0	feld1_intern[125]	BYTE	B#16#0	B#16#0	
126.0	feld1_intern[126]	BYTE	B#16#0	B#16#0	
127.0	feld1_intern[127]	BYTE	B#16#0	B#16#0	

Page 11 of 21

Address	Name	Type	Initial value	Actual value	Comment
128.0	feld1_intern[128]	BYTE	B#16#0	B#16#0	
129.0	feld1_intern[129]	BYTE	B#16#0	B#16#0	
130.0	feld1_intern[130]	BYTE	B#16#0	B#16#0	
131.0	feld1_intern[131]	BYTE	B#16#0	B#16#0	
132.0	feld1_intern[132]	BYTE	B#16#0	B#16#0	
133.0	feld1_intern[133]	BYTE	B#16#0	B#16#0	
134.0	feld1_intern[134]	BYTE	B#16#0	B#16#0	
135.0	feld1_intern[135]	BYTE	B#16#0	B#16#0	
136.0	feld1_intern[136]	BYTE	B#16#0	B#16#0	
137.0	feld1_intern[137]	BYTE	B#16#0	B#16#0	
138.0	feld1_intern[138]	BYTE	B#16#0	B#16#0	
139.0	feld1_intern[139]	BYTE	B#16#0	B#16#0	
140.0	ZOFF	DINT	L#0	L#0	zero offset value
144.0	AVAL	DINT	L#0	L#0	actual value coordinate
148.0	FVAL	DINT	L#0	L#0	actual value coordinate (flying)
152.0	REFPT	DINT	L#0	L#0	reference coordinate
156.0	TRG254	DWORD	DW#16#0	DW#16#0	target 254
160.0	VLEVEL_1	DWORD	DW#16#0	DW#16#0	velocity level 1
164.0	VLEVEL_2	DWORD	DW#16#0	DW#16#0	velocity level 2
168.0	CLEVEL_1	DWORD	DW#16#0	DW#16#0	control level 1
172.0	CLEVEL_2	DWORD	DW#16#0	DW#16#0	control level 2
176.0	MDIB.b_176	BYTE	B#16#0	B#16#0	
177.0	MDIB.b_177	BYTE	B#16#0	B#16#0	
178.0	MDIB.G_1_EN	BOOL	FALSE	FALSE	enable bit for G function group 1
178.1	MDIB.G_2_EN	BOOL	FALSE	FALSE	enable bit for G function group 2
178.2	MDIB.b_178_2	BOOL	FALSE	FALSE	
178.3	MDIB.b_178_3	BOOL	FALSE	FALSE	
178.4	MDIB.X_T_EN	BOOL	FALSE	FALSE	enable bit for position / dwell time
178.5	MDIB.b_178_5	BOOL	FALSE	FALSE	
178.6	MDIB.b_178_6	BOOL	FALSE	FALSE	
178.7	MDIB.b_178_7	BOOL	FALSE	FALSE	
179.0	MDIB.V_EN	BOOL	FALSE	FALSE	enable bit for velocity
179.1	MDIB.M_1_EN	BOOL	FALSE	FALSE	enable bit for M function group 1
179.2	MDIB.M_2_EN	BOOL	FALSE	FALSE	enable bit for M function group 2
179.3	MDIB.M_3_EN	BOOL	FALSE	FALSE	enable bit for M function group 3
179.4	MDIB.b_179_4	BOOL	FALSE	FALSE	
179.5	MDIB.b_179_5	BOOL	FALSE	FALSE	

(Anexo M) 108

Page 12 of 21

Address	Name	Type	Initial value	Actual value	Comment
179.6	MDIB.b_179_6	BOOL	FALSE	FALSE	
179.7	MDIB.b_179_7	BOOL	FALSE	FALSE	
180.0	MDIB.G_1_VAL	BYTE	B#16#0	B#16#0	value G function group 1
181.0	MDIB.G_2_VAL	BYTE	B#16#0	B#16#0	value G function group 2
182.0	MDIB.b_182	BYTE	B#16#0	B#16#0	
183.0	MDIB.b_183	BYTE	B#16#0	B#16#0	
184.0	MDIB.X_T_VAL	DINT	L#0	L#0	value position / dwell time
188.0	MDIB.V_VAL	DINT	L#0	L#0	value velocity
192.0	MDIB.M_1_VAL	BYTE	B#16#0	B#16#0	value M function group 1
193.0	MDIB.M_2_VAL	BYTE	B#16#0	B#16#0	value M function group 2
194.0	MDIB.M_3_VAL	BYTE	B#16#0	B#16#0	value M function group 3
195.0	MDIB.b_195	BYTE	B#16#0	B#16#0	
196.0	PAR_CHAN.TYP	BYTE	B#16#0	B#16#0	DB type
197.0	PAR_CHAN.NUMB	BYTE	B#16#0	B#16#0	data number
198.0	PAR_CHAN.COUN	BYTE	B#16#0	B#16#0	number of data
199.0	PAR_CHAN.JOB	BYTE	B#16#0	B#16#0	job type
200.0	PAR_CHAN.DATA[200]	BYTE	B#16#0	B#16#0	
201.0	PAR_CHAN.DATA[201]	BYTE	B#16#0	B#16#0	
202.0	PAR_CHAN.DATA[202]	BYTE	B#16#0	B#16#0	
203.0	PAR_CHAN.DATA[203]	BYTE	B#16#0	B#16#0	
204.0	PAR_CHAN.DATA[204]	BYTE	B#16#0	B#16#0	
205.0	PAR_CHAN.DATA[205]	BYTE	B#16#0	B#16#0	
206.0	PAR_CHAN.DATA[206]	BYTE	B#16#0	B#16#0	
207.0	PAR_CHAN.DATA[207]	BYTE	B#16#0	B#16#0	
208.0	PAR_CHAN.DATA[208]	BYTE	B#16#0	B#16#0	
209.0	PAR_CHAN.DATA[209]	BYTE	B#16#0	B#16#0	
210.0	PAR_CHAN.DATA[210]	BYTE	B#16#0	B#16#0	
211.0	PAR_CHAN.DATA[211]	BYTE	B#16#0	B#16#0	
212.0	PAR_CHAN.DATA[212]	BYTE	B#16#0	B#16#0	
213.0	PAR_CHAN.DATA[213]	BYTE	B#16#0	B#16#0	
214.0	PAR_CHAN.DATA[214]	BYTE	B#16#0	B#16#0	
215.0	PAR_CHAN.DATA[215]	BYTE	B#16#0	B#16#0	
216.0	PAR_CHAN.DATA[216]	BYTE	B#16#0	B#16#0	
217.0	PAR_CHAN.DATA[217]	BYTE	B#16#0	B#16#0	
218.0	PAR_CHAN.DATA[218]	BYTE	B#16#0	B#16#0	
219.0	PAR_CHAN.DATA[219]	BYTE	B#16#0	B#16#0	
220.0	D_IN0	BOOL	FALSE	FALSE	digital input 0

Address	Name	Type	Initial value	Actual value	Comment
220.1	D_IN1	BOOL	FALSE	FALSE	digital input 1
220.2	D_IN2	BOOL	FALSE	FALSE	digital input 2
220.3	D_IN3	BOOL	FALSE	FALSE	digital input 3
220.4	b_220_4	BOOL	FALSE	FALSE	
220.5	b_220_5	BOOL	FALSE	FALSE	
220.6	b_220_6	BOOL	FALSE	FALSE	
220.7	b_220_7	BOOL	FALSE	FALSE	
221.0	D_OUT0	BOOL	FALSE	FALSE	digital output 0
221.1	D_OUT1	BOOL	FALSE	FALSE	digital output 1
221.2	D_OUT2	BOOL	FALSE	FALSE	digital output 2
221.3	D_OUT3	BOOL	FALSE	FALSE	digital output 3
221.4	b_221_4	BOOL	FALSE	FALSE	
221.5	b_221_5	BOOL	FALSE	FALSE	
221.6	b_221_6	BOOL	FALSE	FALSE	
221.7	b_221_7	BOOL	FALSE	FALSE	
222.0	MDI.F.b_222	BYTE	B#16#0	B#16#0	
223.0	MDI.F.b_223	BYTE	B#16#0	B#16#0	
224.0	MDI.F.G_1_EN	BOOL	FALSE	FALSE	enable bit for G function group 1
224.1	MDI.F.G_2_EN	BOOL	FALSE	FALSE	enable bit for G function group 2
224.2	MDI.F.b_224_2	BOOL	FALSE	FALSE	
224.3	MDI.F.b_224_3	BOOL	FALSE	FALSE	
224.4	MDI.F.X_T_EN	BOOL	FALSE	FALSE	enable bit for position / dwell time
224.5	MDI.F.b_224_5	BOOL	FALSE	FALSE	
224.6	MDI.F.b_224_6	BOOL	FALSE	FALSE	
224.7	MDI.F.b_224_7	BOOL	FALSE	FALSE	
225.0	MDI.F.V_EN	BOOL	FALSE	FALSE	enable bit for velocity
225.1	MDI.F.M_1_EN	BOOL	FALSE	FALSE	enable bit for M function group 1
225.2	MDI.F.M_2_EN	BOOL	FALSE	FALSE	enable bit for M function group 2
225.3	MDI.F.M_3_EN	BOOL	FALSE	FALSE	enable bit for M function group 3
225.4	MDI.F.b_225_4	BOOL	FALSE	FALSE	
225.5	MDI.F.b_225_5	BOOL	FALSE	FALSE	
225.6	MDI.F.b_225_6	BOOL	FALSE	FALSE	
225.7	MDI.F.b_225_7	BOOL	FALSE	FALSE	
226.0	MDI.F.G_1_VAL	BYTE	B#16#0	B#16#0	value G function group 1
227.0	MDI.F.G_2_VAL	BYTE	B#16#0	B#16#0	value G function group 2
228.0	MDI.F.b_228	BYTE	B#16#0	B#16#0	
229.0	MDI.F.b_229	BYTE	B#16#0	B#16#0	

Address	Name	Type	Initial value	Actual value	Comment
230.0	MDI_F_X_T_VAL	DINT	L#0	L#0	value position / dwell time
234.0	MDI_F_V_VAL	DINT	L#0	L#0	value velocity
238.0	MDI_F_M_1_VAL	BYTE	B#16#0	B#16#0	value M function group 1
239.0	MDI_F_M_2_VAL	BYTE	B#16#0	B#16#0	value M function group 2
240.0	MDI_F_M_3_VAL	BYTE	B#16#0	B#16#0	value M function group 3
241.0	MDI_F_b_241	BYTE	B#16#0	B#16#0	
242.0	PROG_NO	BYTE	B#16#0	B#16#0	part program number
243.0	BLCK_NO	BYTE	B#16#0	B#16#0	part program block number
244.0	PROG_DIR	BYTE	B#16#0	B#16#0	program run direction
245.0	b_245	BYTE	B#16#0	B#16#0	
246.0	CODE_AP1	BYTE	B#16#0	B#16#0	code application 1
247.0	CODE_AP2	BYTE	B#16#0	B#16#0	code application 2
248.0	CODE_AP3	BYTE	B#16#0	B#16#0	code application 3
249.0	CODE_AP4	BYTE	B#16#0	B#16#0	code application 4
250.0	TEA_PRO_NO	BYTE	B#16#0	B#16#0	part program number
251.0	TEA_BLKCK_NO	BYTE	B#16#0	B#16#0	part program block number
252.0	AXCOU_DEF	BYTE	B#16#0	B#16#0	axis coupling locate DS24 FM453
253.0	b_253	BYTE	B#16#0	B#16#0	
254.0	feld2_intern[254]	BYTE	B#16#0	B#16#0	
255.0	feld2_intern[255]	BYTE	B#16#0	B#16#0	
256.0	feld2_intern[256]	BYTE	B#16#0	B#16#0	
257.0	feld2_intern[257]	BYTE	B#16#0	B#16#0	
258.0	feld2_intern[258]	BYTE	B#16#0	B#16#0	
259.0	feld2_intern[259]	BYTE	B#16#0	B#16#0	
260.0	feld2_intern[260]	BYTE	B#16#0	B#16#0	
261.0	feld2_intern[261]	BYTE	B#16#0	B#16#0	
262.0	feld2_intern[262]	BYTE	B#16#0	B#16#0	
263.0	feld2_intern[263]	BYTE	B#16#0	B#16#0	
264.0	feld2_intern[264]	BYTE	B#16#0	B#16#0	
265.0	feld2_intern[265]	BYTE	B#16#0	B#16#0	
266.0	feld2_intern[266]	BYTE	B#16#0	B#16#0	
267.0	feld2_intern[267]	BYTE	B#16#0	B#16#0	
268.0	feld2_intern[268]	BYTE	B#16#0	B#16#0	
269.0	feld2_intern[269]	BYTE	B#16#0	B#16#0	
270.0	feld2_intern[270]	BYTE	B#16#0	B#16#0	
271.0	feld2_intern[271]	BYTE	B#16#0	B#16#0	
272.0	feld2_intern[272]	BYTE	B#16#0	B#16#0	

Address	Name	Type	Initial value	Actual value	Comment
273.0	feld2_intern[273]	BYTE	B#16#0	B#16#0	
274.0	feld2_intern[274]	BYTE	B#16#0	B#16#0	
275.0	feld2_intern[275]	BYTE	B#16#0	B#16#0	
276.0	feld2_intern[276]	BYTE	B#16#0	B#16#0	
277.0	feld2_intern[277]	BYTE	B#16#0	B#16#0	
278.0	feld2_intern[278]	BYTE	B#16#0	B#16#0	
279.0	feld2_intern[279]	BYTE	B#16#0	B#16#0	
280.0	feld2_intern[280]	BYTE	B#16#0	B#16#0	
281.0	feld2_intern[281]	BYTE	B#16#0	B#16#0	
282.0	feld2_intern[282]	BYTE	B#16#0	B#16#0	
283.0	feld2_intern[283]	BYTE	B#16#0	B#16#0	
284.0	feld2_intern[284]	BYTE	B#16#0	B#16#0	
285.0	feld2_intern[285]	BYTE	B#16#0	B#16#0	
286.0	feld2_intern[286]	BYTE	B#16#0	B#16#0	
287.0	feld2_intern[287]	BYTE	B#16#0	B#16#0	
288.0	feld2_intern[288]	BYTE	B#16#0	B#16#0	
289.0	feld2_intern[289]	BYTE	B#16#0	B#16#0	
290.0	feld2_intern[290]	BYTE	B#16#0	B#16#0	
291.0	feld2_intern[291]	BYTE	B#16#0	B#16#0	
292.0	feld2_intern[292]	BYTE	B#16#0	B#16#0	
293.0	feld2_intern[293]	BYTE	B#16#0	B#16#0	
294.0	feld2_intern[294]	BYTE	B#16#0	B#16#0	
295.0	feld2_intern[295]	BYTE	B#16#0	B#16#0	
296.0	feld2_intern[296]	BYTE	B#16#0	B#16#0	
297.0	feld2_intern[297]	BYTE	B#16#0	B#16#0	
298.0	feld2_intern[298]	BYTE	B#16#0	B#16#0	
299.0	feld2_intern[299]	BYTE	B#16#0	B#16#0	
300.0	feld2_intern[300]	BYTE	B#16#0	B#16#0	
301.0	feld2_intern[301]	BYTE	B#16#0	B#16#0	
302.0	feld2_intern[302]	BYTE	B#16#0	B#16#0	
303.0	feld2_intern[303]	BYTE	B#16#0	B#16#0	
304.0	feld2_intern[304]	BYTE	B#16#0	B#16#0	
305.0	feld2_intern[305]	BYTE	B#16#0	B#16#0	
306.0	feld2_intern[306]	BYTE	B#16#0	B#16#0	
307.0	feld2_intern[307]	BYTE	B#16#0	B#16#0	
308.0	feld2_intern[308]	BYTE	B#16#0	B#16#0	
309.0	feld2_intern[309]	BYTE	B#16#0	B#16#0	

Address	Name	Type	Initial value	Actual value	Comment
310.0	ACT_VAL	DINT	L#0	L#0	actual value
314.0	SPEED	DWORD	DW#16#0	DW#16#0	speed
318.0	REM_DIST	DINT	L#0	L#0	remaining distance
322.0	SET_POS	DINT	L#0	L#0	set position to be approached
326.0	SUM_OFST	DINT	L#0	L#0	sum of tool offset and zero offset
330.0	TRAV_SPE	DWORD	DW#16#0	DW#16#0	traversing speed
334.0	di_334	DINT	L#0	L#0	
338.0	di_338	DINT	L#0	L#0	
342.0	ACT_BL_PROG_NO	BYTE	B#16#0	B#16#0	part program number
343.0	ACT_BL_BLK_NO	BYTE	B#16#0	B#16#0	part program block number
344.0	ACT_BL_G_1_EN	BOOL	FALSE	FALSE	enable bit for G function group 1
344.1	ACT_BL_G_2_EN	BOOL	FALSE	FALSE	enable bit for G function group 2
344.2	ACT_BL_G_3_EN	BOOL	FALSE	FALSE	enable bit for G function group 3
344.3	ACT_BL_b_344_3	BOOL	FALSE	FALSE	
344.4	ACT_BL_X_T_EN	BOOL	FALSE	FALSE	enable bit for position / dwell time
344.5	ACT_BL_SR_L_EN	BOOL	FALSE	FALSE	enable bit for number of subroutine loops
344.6	ACT_BL_SR_N_EN	BOOL	FALSE	FALSE	enable bit for subroutine number
344.7	ACT_BL_SKIP_EN	BOOL	FALSE	FALSE	enable bit for block skip
345.0	ACT_BL_V_EN	BOOL	FALSE	FALSE	enable bit for velocity
345.1	ACT_BL_M_1_EN	BOOL	FALSE	FALSE	enable bit for M function group 1
345.2	ACT_BL_M_2_EN	BOOL	FALSE	FALSE	enable bit for M function group 2
345.3	ACT_BL_M_3_EN	BOOL	FALSE	FALSE	enable bit for M function group 3
345.4	ACT_BL_TO_EN	BOOL	FALSE	FALSE	enable bit for tool offset number
345.5	ACT_BL_b_345_5	BOOL	FALSE	FALSE	
345.6	ACT_BL_b_345_6	BOOL	FALSE	FALSE	
345.7	ACT_BL_b_345_7	BOOL	FALSE	FALSE	
346.0	ACT_BL_G_1_VAL	BYTE	B#16#0	B#16#0	G function group 1
347.0	ACT_BL_G_2_VAL	BYTE	B#16#0	B#16#0	G function group 2
348.0	ACT_BL_G_3_VAL	BYTE	B#16#0	B#16#0	G function group 3
349.0	ACT_BL_b_349	BYTE	B#16#0	B#16#0	
350.0	ACT_BL_X_T_VAL	DINT	L#0	L#0	position / dwell time / subroutine number
354.0	ACT_BL_V_VAL	DINT	L#0	L#0	velocity / subroutine loops
358.0	ACT_BL_M_1_VAL	BYTE	B#16#0	B#16#0	M function group 1
359.0	ACT_BL_M_2_VAL	BYTE	B#16#0	B#16#0	M function group 2
360.0	ACT_BL_M_3_VAL	BYTE	B#16#0	B#16#0	M function group 3
361.0	ACT_BL_TO_VAL	BYTE	B#16#0	B#16#0	tool offset number
362.0	NXT_BLK.PROG_NO	BYTE	B#16#0	B#16#0	part program number

Address	Name	Type	Initial value	Actual value	Comment
363.0	NXT_BLK.BLCK_NO	BYTE	B#16#0	B#16#0	part program block number
364.0	NXT_BLK_G_1_EN	BOOL	FALSE	FALSE	enable bit for G function group 1
364.1	NXT_BLK_G_2_EN	BOOL	FALSE	FALSE	enable bit for G function group 2
364.2	NXT_BLK_G_3_EN	BOOL	FALSE	FALSE	enable bit for G function group 3
364.3	NXT_BLK_b_364_3	BOOL	FALSE	FALSE	
364.4	NXT_BLK_X_T_EN	BOOL	FALSE	FALSE	enable bit for position / dwell time
364.5	NXT_BLK_SR_L_EN	BOOL	FALSE	FALSE	enable bit for number of subroutine loops
364.6	NXT_BLK_SR_N_EN	BOOL	FALSE	FALSE	enable bit for subroutine number
364.7	NXT_BLK_SKIP_EN	BOOL	FALSE	FALSE	enable bit for block skip
365.0	NXT_BLK_V_EN	BOOL	FALSE	FALSE	enable bit for velocity
365.1	NXT_BLK_M_1_EN	BOOL	FALSE	FALSE	enable bit for M function group 1
365.2	NXT_BLK_M_2_EN	BOOL	FALSE	FALSE	enable bit for M function group 2
365.3	NXT_BLK_M_3_EN	BOOL	FALSE	FALSE	enable bit for M function group 3
365.4	NXT_BLK.TO_EN	BOOL	FALSE	FALSE	enable bit for tool offset number
365.5	NXT_BLK_b_365_5	BOOL	FALSE	FALSE	
365.6	NXT_BLK_b_365_6	BOOL	FALSE	FALSE	
365.7	NXT_BLK_b_365_7	BOOL	FALSE	FALSE	
366.0	NXT_BLK_G_1_VAL	BYTE	B#16#0	B#16#0	G function group 1
367.0	NXT_BLK_G_2_VAL	BYTE	B#16#0	B#16#0	G function group 2
368.0	NXT_BLK_G_3_VAL	BYTE	B#16#0	B#16#0	G function group 3
369.0	NXT_BLK_b_369	BYTE	B#16#0	B#16#0	
370.0	NXT_BLK_X_T_VAL	DINT	L#0	L#0	position / dwell time / subroutine number
374.0	NXT_BLK.V_VAL	DINT	L#0	L#0	velocity / subroutine loops
378.0	NXT_BLK.M_1_VAL	BYTE	B#16#0	B#16#0	M function group 1
379.0	NXT_BLK.M_2_VAL	BYTE	B#16#0	B#16#0	M function group 2
380.0	NXT_BLK.M_3_VAL	BYTE	B#16#0	B#16#0	M function group 3
381.0	NXT_BLK.TO_VAL	BYTE	B#16#0	B#16#0	tool offset number
382.0	APP1	DINT	L#0	L#0	application data 1
386.0	APP2	DINT	L#0	L#0	application data 2
390.0	APP3	DINT	L#0	L#0	application data 3
394.0	APP4	DINT	L#0	L#0	application data 4
398.0	BLCK_EXT	DINT	L#0	L#0	actual value at external block change
402.0	OUT_VAL	DINT	L#0	L#0	DAC value / frequency
406.0	ENC_VAL	DINT	L#0	L#0	encoder actual value
410.0	PULS_ERR	DINT	L#0	L#0	error pulses
414.0	KV_FA	DINT	L#0	L#0	Kv factor
418.0	FOLL_ERR	DINT	L#0	L#0	following error

Address	Name	Type	Initial value	Actual value	Comment
422.0	FERR_LIM	DINT	L#0	L#0	following error limit
426.0	OSC_ERR	DINT	L#0	L#0	oscillation error
430.0	DR_TIME	DINT	L#0	L#0	drive time
434.0	OVERRIDE1	BYTE	B#16#0	B#16#0	override
435.0	PROG_NO1	BYTE	B#16#0	B#16#0	part program number
436.0	BLCK_NO1	BYTE	B#16#0	B#16#0	part program block number
437.0	LOOP_NO1	BYTE	B#16#0	B#16#0	number of subroutine loops
438.0	G90_91	BYTE	B#16#0	B#16#0	G90/91 active
439.0	G60_64	BYTE	B#16#0	B#16#0	G60/64 active
440.0	G43_44	BYTE	B#16#0	B#16#0	G43/44 active
441.0	TO_NO	BYTE	B#16#0	B#16#0	tool offset number
442.0	b_442_0	BOOL	FALSE	FALSE	
442.1	LIM_SP	BOOL	FALSE	FALSE	limitation speed
442.2	LIM_10	BOOL	FALSE	FALSE	limitation +/- 10V
442.3	LIM_SU	BOOL	FALSE	FALSE	limitation speed_up
442.4	b_442_4	BOOL	FALSE	FALSE	
442.5	b_442_5	BOOL	FALSE	FALSE	
442.6	b_442_6	BOOL	FALSE	FALSE	
442.7	b_442_7	BOOL	FALSE	FALSE	
443.0	LIM_FR	BOOL	FALSE	FALSE	limitation frequency
443.1	LIM_FV	BOOL	FALSE	FALSE	limitation frequency value
443.2	b_443_2	BOOL	FALSE	FALSE	
443.3	LIM_FS	BOOL	FALSE	FALSE	limitation frequency speed-up
443.4	b_443_4	BOOL	FALSE	FALSE	
443.5	b_443_5	BOOL	FALSE	FALSE	
443.6	b_443_6	BOOL	FALSE	FALSE	
443.7	b_443_7	BOOL	FALSE	FALSE	
444.0	b_444	BYTE	B#16#0	B#16#0	
445.0	b_445	BYTE	B#16#0	B#16#0	
446.0	PAR_RD.TYP1	BYTE	B#16#0	B#16#0	DB type
447.0	PAR_RD.NUMB	BYTE	B#16#0	B#16#0	data number
448.0	PAR_RD.COUN	BYTE	B#16#0	B#16#0	number of data
449.0	PAR_RD.JOB	BYTE	B#16#0	B#16#0	job type
450.0	PAR_RD.DATA1[450]	BYTE	B#16#0	B#16#0	
451.0	PAR_RD.DATA1[451]	BYTE	B#16#0	B#16#0	
452.0	PAR_RD.DATA1[452]	BYTE	B#16#0	B#16#0	
453.0	PAR_RD.DATA1[453]	BYTE	B#16#0	B#16#0	

Page 19 of 21

Address	Name	Type	Initial value	Actual value	Comment
454.0	PAR_RD.DATA1[454]	BYTE	B#16#0	B#16#0	
455.0	PAR_RD.DATA1[455]	BYTE	B#16#0	B#16#0	
456.0	PAR_RD.DATA1[456]	BYTE	B#16#0	B#16#0	
457.0	PAR_RD.DATA1[457]	BYTE	B#16#0	B#16#0	
458.0	PAR_RD.DATA1[458]	BYTE	B#16#0	B#16#0	
459.0	PAR_RD.DATA1[459]	BYTE	B#16#0	B#16#0	
460.0	PAR_RD.DATA1[460]	BYTE	B#16#0	B#16#0	
461.0	PAR_RD.DATA1[461]	BYTE	B#16#0	B#16#0	
462.0	PAR_RD.DATA1[462]	BYTE	B#16#0	B#16#0	
463.0	PAR_RD.DATA1[463]	BYTE	B#16#0	B#16#0	
464.0	PAR_RD.DATA1[464]	BYTE	B#16#0	B#16#0	
465.0	PAR_RD.DATA1[465]	BYTE	B#16#0	B#16#0	
466.0	PAR_RD.DATA1[466]	BYTE	B#16#0	B#16#0	
467.0	PAR_RD.DATA1[467]	BYTE	B#16#0	B#16#0	
468.0	PAR_RD.DATA1[468]	BYTE	B#16#0	B#16#0	
469.0	PAR_RD.DATA1[469]	BYTE	B#16#0	B#16#0	
470.0	AXCOU_STAT	BYTE	B#16#0	B#16#0	status axis coupling DS37 FM453
471.0	b_471	BYTE	B#16#0	B#16#0	
472.0	feld3_intern[472]	BYTE	B#16#0	B#16#0	
473.0	feld3_intern[473]	BYTE	B#16#0	B#16#0	
474.0	feld3_intern[474]	BYTE	B#16#0	B#16#0	
475.0	feld3_intern[475]	BYTE	B#16#0	B#16#0	
476.0	feld3_intern[476]	BYTE	B#16#0	B#16#0	
477.0	feld3_intern[477]	BYTE	B#16#0	B#16#0	
478.0	feld3_intern[478]	BYTE	B#16#0	B#16#0	
479.0	feld3_intern[479]	BYTE	B#16#0	B#16#0	
480.0	feld3_intern[480]	BYTE	B#16#0	B#16#0	
481.0	feld3_intern[481]	BYTE	B#16#0	B#16#0	
482.0	feld3_intern[482]	BYTE	B#16#0	B#16#0	
483.0	feld3_intern[483]	BYTE	B#16#0	B#16#0	
484.0	feld3_intern[484]	BYTE	B#16#0	B#16#0	
485.0	feld3_intern[485]	BYTE	B#16#0	B#16#0	
486.0	BEGIN_VAL	DINT	L#0	L#0	begin value
490.0	END_VAL	DINT	L#0	L#0	end value
494.0	LENGTH_VAL	DWORD	DW#16#0	DW#16#0	length value
498.0	USR.BITC_0	BOOL	FALSE	FALSE	MD write request
498.1	USR.BITC_1	BOOL	FALSE	FALSE	MD read request

(Anexo M) 112

Page 20 of 21

Address	Name	Type	Initial value	Actual value	Comment
498.2	USR.BITC_2	BOOL	FALSE	FALSE	MDI block request
498.3	USR.BITC_3	BOOL	FALSE	FALSE	part program selection request
498.4	USR.BITC_4	BOOL	FALSE	FALSE	Teach In request
498.5	USR.BITC_5	BOOL	FALSE	FALSE	incremental value request
498.6	USR.BITC_6	BOOL	FALSE	FALSE	velocity levels request
498.7	USR.BITC_7	BOOL	FALSE	FALSE	control levels request
499.0	USR.BITC_8	BOOL	FALSE	FALSE	MDI block on the fly request
499.1	USR.BITC_9	BOOL	FALSE	FALSE	actual value on the fly request
499.2	USR.BITC_10	BOOL	FALSE	FALSE	zero point offset request
499.3	USR.BITC_11	BOOL	FALSE	FALSE	reserved
499.4	USR.BITC_12	BOOL	FALSE	FALSE	reserved
499.5	USR.BITC_13	BOOL	FALSE	FALSE	Alarm
499.6	USR.BITC_14	BOOL	FALSE	FALSE	Data-Error
499.7	USR.BITC_15	BOOL	FALSE	FALSE	OT-Error
500.0	USR.MD_NO	WORD	W#16#0	W#16#0	MD number
502.0	USR.MD_VALUE	DINT	L#0	L#0	MD value
506.0	USR.INC_NO	BYTE	B#16#0	B#16#0	incremental value number
507.0	USR.b_507	BYTE	B#16#0	B#16#0	
508.0	USR.PICT_NO	WORD	W#16#0	W#16#0	picture number
510.0	USR.KEY_CODE	WORD	W#16#0	W#16#0	keyboard code
512.0	USR.b_512	WORD	W#16#0	W#16#0	
514.0	USR.BITA_0	BOOL	FALSE	FALSE	Control
514.1	USR.BITA_1	BOOL	FALSE	FALSE	Reference point approach
514.2	USR.BITA_2	BOOL	FALSE	FALSE	Increment
514.3	USR.BITA_3	BOOL	FALSE	FALSE	MDI
514.4	USR.BITA_4	BOOL	FALSE	FALSE	Automatic / single move
514.5	USR.BITA_5	BOOL	FALSE	FALSE	Automatic / follow move
514.6	USR.BITA_6	BOOL	FALSE	FALSE	Jog-Mode
514.7	USR.b_514_7	BOOL	FALSE	FALSE	
515.0	USR.b_515_0	BOOL	FALSE	FALSE	
515.1	USR.b_515_1	BOOL	FALSE	FALSE	
515.2	USR.b_515_2	BOOL	FALSE	FALSE	
515.3	USR.b_515_3	BOOL	FALSE	FALSE	
515.4	USR.b_515_4	BOOL	FALSE	FALSE	
515.5	USR.b_515_5	BOOL	FALSE	FALSE	
515.6	USR.BITA_14	BOOL	FALSE	FALSE	Quit Error
515.7	USR.BITA_15	BOOL	FALSE	FALSE	Quit Alarm

DB2 - <offline>

"DB_UDTM2"

Data view

Data block type:

UDT2 data block

Name: DB_UDTM2
Author: SIEMENS
Family:
Version: 0.1
Block version: 2
Lengths (Block / Data): 01890 / 00516
Time stamp
Code: 07/18/2012 04:54:03 PM
Interface: 07/04/2000 03:52:02 PM

Comment:

Address	Name	Type	Initial value	Actual value	Comment
0.0	MOD_ADR_M2	INT	0	368	Module address
2.0	CH_NO_M2	INT	0	1	Channel number
4.0	CH_ADR_M2	DWORD	DW#16#0	DW#16#B80	Channel address
8.0	DS_OFFS_M2	INT	0	0	data block offset
10.0	b_10_M2	WORD	W#16#0	W#16#0	
12.0	b_12_M2	BYTE	B#16#0	B#16#0	
13.0	b_13_0_M2	BOOL	FALSE	FALSE	
13.1	b_13_1_M2	BOOL	FALSE	FALSE	
13.2	b_13_2_M2	BOOL	FALSE	FALSE	
13.3	b_13_3_M2	BOOL	FALSE	FALSE	
13.4	b_13_4_M2	BOOL	FALSE	FALSE	
13.5	b_13_5_M2	BOOL	FALSE	FALSE	
13.6	MODE_BUSY_M2	BOOL	FALSE	FALSE	Mode busy
13.7	POS_REACHED_M2	BOOL	FALSE	FALSE	Position reached
14.0	b_14_0_M2	BOOL	FALSE	FALSE	
14.1	TEST_EN_M2	BOOL	FALSE	FALSE	test enable
14.2	b_14_2_M2	BOOL	FALSE	FALSE	
14.3	OT_ERR_A_M2	BOOL	FALSE	FALSE	quit error
14.4	b_14_4_M2	BOOL	FALSE	FALSE	

Address	Name	Type	Initial value	Actual value	Comment
14.5	b_14_5_M2	BOOL	FALSE	FALSE	
14.6	b_14_6_M2	BOOL	FALSE	FALSE	
14.7	b_14_7_M2	BOOL	FALSE	FALSE	
15.0	START_M2	BOOL	FALSE	FALSE	start
15.1	STOP_M2	BOOL	FALSE	FALSE	stop
15.2	DIR_M_M2	BOOL	FALSE	FALSE	direction minus
15.3	DIR_P_M2	BOOL	FALSE	FALSE	direction plus
15.4	ACK_MF_M2	BOOL	FALSE	FALSE	acknowledge M function
15.5	READ_EN_M2	BOOL	FALSE	FALSE	read enable
15.6	SKIP_BLK_M2	BOOL	FALSE	FALSE	skip blank
15.7	DRV_EN_M2	BOOL	FALSE	FALSE	drive enable
16.0	MODE_IN_M2	BYTE	B#16#0	B#16#0	mode
17.0	MODE_TYPE_M2	BYTE	B#16#0	B#16#0	mode parameter
18.0	OVERRIDE_M2	BYTE	B#16#0	B#16#0	override
19.0	b_19_M2	BYTE	B#16#0	B#16#0	
20.0	b_20_M2	WORD	W#16#0	W#16#0	
22.0	b_22_0_M2	BOOL	FALSE	FALSE	
22.1	TST_STAT_M2	BOOL	FALSE	FALSE	test state
22.2	b_22_2_M2	BOOL	FALSE	FALSE	
22.3	OT_ERR_M2	BOOL	FALSE	FALSE	ot-error
22.4	DATA_ERR_M2	BOOL	FALSE	FALSE	data-error
22.5	b_22_5_M2	BOOL	FALSE	FALSE	
22.6	b_22_6_M2	BOOL	FALSE	FALSE	
22.7	PARA_M2	BOOL	FALSE	FALSE	parameterized
23.0	ST_ENBLD_M2	BOOL	FALSE	FALSE	start enable
23.1	WORKING_M2	BOOL	FALSE	FALSE	working
23.2	WAIT_EI_M2	BOOL	FALSE	FALSE	wait for external enable
23.3	b_23_3_M2	BOOL	FALSE	FALSE	
23.4	b_23_4_M2	BOOL	FALSE	FALSE	
23.5	DT_RUN_M2	BOOL	FALSE	FALSE	dwelt time running
23.6	PR_BACK_M2	BOOL	FALSE	FALSE	program run backward
23.7	b_23_7_M2	BOOL	FALSE	FALSE	
24.0	MODE_OUT_M2	BYTE	B#16#0	B#16#0	mode
25.0	SYNC_M2	BOOL	FALSE	FALSE	synchronized
25.1	MSR_DONE_M2	BOOL	FALSE	FALSE	measurement done
25.2	GO_M_M2	BOOL	FALSE	FALSE	go_minus
25.3	GO_P_M2	BOOL	FALSE	FALSE	go_plus

Page 2 of 21

Address	Name	Type	Initial value	Actual value	Comment
25.4	ST_SERVO_M2	BOOL	FALSE	FALSE	status servo
25.5	FVAL_DONE_M2	BOOL	FALSE	FALSE	flying actual value done
25.6	b_25_6_M2	BOOL	FALSE	FALSE	
25.7	POS_RCD_M2	BOOL	FALSE	FALSE	position is reached, hold
26.0	NUM_MF_M2	BYTE	B#16#0	B#16#0	number M function
27.0	b_27_0_M2	BOOL	FALSE	FALSE	
27.1	b_27_1_M2	BOOL	FALSE	FALSE	
27.2	b_27_2_M2	BOOL	FALSE	FALSE	
27.3	b_27_3_M2	BOOL	FALSE	FALSE	
27.4	STR_MF_M2	BOOL	FALSE	FALSE	strobe signal for M functions
27.5	b_27_5_M2	BOOL	FALSE	FALSE	
27.6	b_27_6_M2	BOOL	FALSE	FALSE	
27.7	b_27_7_M2	BOOL	FALSE	FALSE	
28.0	b_28_M2	DWORD	DW#16#0	DW#16#0	
32.0	b_32_M2	WORD	W#16#0	W#16#0	
34.0	SERVO_EN_M2	BOOL	FALSE	FALSE	servo enable
34.1	GAUG_FLY_M2	BOOL	FALSE	FALSE	gauging on the fly
34.2	b_34_2_M2	BOOL	FALSE	FALSE	
34.3	b_34_3_M2	BOOL	FALSE	FALSE	
34.4	b_34_4_M2	BOOL	FALSE	FALSE	
34.5	TRAV_MON_M2	BOOL	FALSE	FALSE	traverse monitoring
34.6	PARK_AX_M2	BOOL	FALSE	FALSE	parking axis
34.7	SIM_ON_M2	BOOL	FALSE	FALSE	simulation on
35.0	b_35_0_M2	BOOL	FALSE	FALSE	
35.1	b_35_1_M2	BOOL	FALSE	FALSE	
35.2	MSR_EN_M2	BOOL	FALSE	FALSE	measurement
35.3	REF_TRIG_M2	BOOL	FALSE	FALSE	trigger reference point
35.4	DI_OFF_M2	BOOL	FALSE	FALSE	disable enable input
35.5	FOLLOWUP_M2	BOOL	FALSE	FALSE	follow-up
35.6	SSW_DIS_M2	BOOL	FALSE	FALSE	software switch disable
35.7	DRIFT_OFF_M2	BOOL	FALSE	FALSE	automatic drift compensation off
36.0	b_36_M2	BYTE	B#16#0	B#16#0	
37.0	MD_EN_M2	BOOL	FALSE	FALSE	machine data enable
37.1	DELDIST_EN_M2	BOOL	FALSE	FALSE	delete distance to go
37.2	SEARCH_F_M2	BOOL	FALSE	FALSE	automatic block search forward
37.3	SEARCH_B_M2	BOOL	FALSE	FALSE	automatic block search backward
37.4	b_37_4_M2	BOOL	FALSE	FALSE	

(Anexo M) 114

Page 3 of 21

Address	Name	Type	Initial value	Actual value	Comment
37.5	RESET_AX_M2	BOOL	FALSE	FALSE	reset axis
37.6	AVALREM_EN_M2	BOOL	FALSE	FALSE	remove setting actual value
37.7	b_37_7_M2	BOOL	FALSE	FALSE	
38.0	VLEV_EN_M2	BOOL	FALSE	FALSE	velocity levels 1 and 2
38.1	CLEV_EN_M2	BOOL	FALSE	FALSE	control levels 1 and 2
38.2	TRG254_EN_M2	BOOL	FALSE	FALSE	target 254
38.3	MDI_EN_M2	BOOL	FALSE	FALSE	MDI movement block
38.4	MDIFLY_EN_M2	BOOL	FALSE	FALSE	MDI block on the fly
38.5	b_38_5_M2	BOOL	FALSE	FALSE	
38.6	REFPT_EN_M2	BOOL	FALSE	FALSE	reference coordinate
38.7	AVAL_EN_M2	BOOL	FALSE	FALSE	actual value coordinate
39.0	FVAL_EN_M2	BOOL	FALSE	FALSE	actual value coordinate (flying)
39.1	ZOFF_EN_M2	BOOL	FALSE	FALSE	zero offset value
39.2	b_39_2_M2	BOOL	FALSE	FALSE	
39.3	PARCH_EN_M2	BOOL	FALSE	FALSE	parameter change
39.4	DIGO_EN_M2	BOOL	FALSE	FALSE	digital 0
39.5	PROGS_EN_M2	BOOL	FALSE	FALSE	part program selection
39.6	REQAPP_EN_M2	BOOL	FALSE	FALSE	request application data
39.7	TEACHIN_EN_M2	BOOL	FALSE	FALSE	Teach In
40.0	AXCOU_EN_M2	BOOL	FALSE	FALSE	write job axis coupling FM453
40.1	b_40_1_M2	BOOL	FALSE	FALSE	
40.2	b_40_2_M2	BOOL	FALSE	FALSE	
40.3	b_40_3_M2	BOOL	FALSE	FALSE	
40.4	b_40_4_M2	BOOL	FALSE	FALSE	
40.5	b_40_5_M2	BOOL	FALSE	FALSE	
40.6	b_40_6_M2	BOOL	FALSE	FALSE	
40.7	b_40_7_M2	BOOL	FALSE	FALSE	
41.0	b_41_M2	BYTE	B#16#0	B#16#0	
42.0	OPDAT_EN_M2	BOOL	FALSE	FALSE	operating_data
42.1	ACT_BL_EN_M2	BOOL	FALSE	FALSE	actual part program block
42.2	NXT_BL_EN_M2	BOOL	FALSE	FALSE	next part program block
42.3	BLEXT_EN_M2	BOOL	FALSE	FALSE	actual value at external block change
42.4	SERVDAT_EN_M2	BOOL	FALSE	FALSE	service data
42.5	OC_ERR_EN_M2	BOOL	FALSE	FALSE	oc-error
42.6	b_42_6_M2	BOOL	FALSE	FALSE	
42.7	b_42_7_M2	BOOL	FALSE	FALSE	
43.0	AXCOURD_EN_M2	BOOL	FALSE	FALSE	read job axis coupling FM453

Address	Name	Type	Initial value	Actual value	Comment
43.1	b_43_1_M2	BOOL	FALSE	FALSE	
43.2	b_43_2_M2	BOOL	FALSE	FALSE	
43.3	PARRD_EN_M2	BOOL	FALSE	FALSE	parameter data
43.4	DIGIO_EN_M2	BOOL	FALSE	FALSE	digital in_out
43.5	OPDAT1_EN_M2	BOOL	FALSE	FALSE	operating data 1
43.6	APPDAT_EN_M2	BOOL	FALSE	FALSE	applying data
43.7	MSRRD_EN_M2	BOOL	FALSE	FALSE	read measurement values
44.0	SERVO_D_M2	BOOL	FALSE	FALSE	servo enable
44.1	GAUG_FLY_D_M2	BOOL	FALSE	FALSE	gauging on the fly
44.2	b_44_2_M2	BOOL	FALSE	FALSE	
44.3	b_44_3_M2	BOOL	FALSE	FALSE	
44.4	b_44_4_M2	BOOL	FALSE	FALSE	
44.5	TRAV_MON_D_M2	BOOL	FALSE	FALSE	traverse monitoring
44.6	PARK_AX_D_M2	BOOL	FALSE	FALSE	parking axis
44.7	SIM_ON_D_M2	BOOL	FALSE	FALSE	simulation on
45.0	b_45_0_M2	BOOL	FALSE	FALSE	
45.1	b_45_1_M2	BOOL	FALSE	FALSE	
45.2	MSR_D_M2	BOOL	FALSE	FALSE	measurement
45.3	REF_TRIG_D_M2	BOOL	FALSE	FALSE	trigger reference point
45.4	DI_OFF_D_M2	BOOL	FALSE	FALSE	disable enable input
45.5	FOLLOWUP_D_M2	BOOL	FALSE	FALSE	follow-up
45.6	SSW_DIS_D_M2	BOOL	FALSE	FALSE	software switch disable
45.7	DRIFT_OFF_D_M2	BOOL	FALSE	FALSE	automatic drift compensation off
46.0	b_46_M2	BYTE	B#16#0	B#16#0	
47.0	MD_D_M2	BOOL	FALSE	FALSE	machine data enable
47.1	DELDIST_D_M2	BOOL	FALSE	FALSE	delete distance to go
47.2	SEARCH_F_D_M2	BOOL	FALSE	FALSE	automatic block search forward
47.3	SEARCH_B_D_M2	BOOL	FALSE	FALSE	automatic block search backward
47.4	b_47_4_M2	BOOL	FALSE	FALSE	
47.5	RESET_AX_D_M2	BOOL	FALSE	FALSE	reset axis
47.6	AVALREM_D_M2	BOOL	FALSE	FALSE	remove setting actual value
47.7	b_47_7_M2	BOOL	FALSE	FALSE	
48.0	VLEV_D_M2	BOOL	FALSE	FALSE	velocity levels 1 and 2
48.1	CLEV_D_M2	BOOL	FALSE	FALSE	control levels 1 and 2
48.2	TRG254_D_M2	BOOL	FALSE	FALSE	target 254
48.3	MDI_D_M2	BOOL	FALSE	FALSE	MDI movement block
48.4	MDIFLY_D_M2	BOOL	FALSE	FALSE	MDI block on the fly

Address	Name	Type	Initial value	Actual value	Comment
48.5	b_48_5_M2	BOOL	FALSE	FALSE	
48.6	REFPT_D_M2	BOOL	FALSE	FALSE	reference coordinate
48.7	AVAL_D_M2	BOOL	FALSE	FALSE	actual value coordinate
49.0	FVAL_D_M2	BOOL	FALSE	FALSE	actual value coordinate (flying)
49.1	ZOFF_D_M2	BOOL	FALSE	FALSE	zero offset value
49.2	b_49_2_M2	BOOL	FALSE	FALSE	
49.3	PARCH_D_M2	BOOL	FALSE	FALSE	parameter change
49.4	DIGO_D_M2	BOOL	FALSE	FALSE	digital 0
49.5	PROGS_D_M2	BOOL	FALSE	FALSE	part program selection
49.6	REQAPP_D_M2	BOOL	FALSE	FALSE	request application data
49.7	TEACHIN_D_M2	BOOL	FALSE	FALSE	Teach In
50.0	AXCOU_D_M2	BOOL	FALSE	FALSE	axis coupling FM453
50.1	b_50_1_M2	BOOL	FALSE	FALSE	
50.2	b_50_2_M2	BOOL	FALSE	FALSE	
50.3	b_50_3_M2	BOOL	FALSE	FALSE	
50.4	b_50_4_M2	BOOL	FALSE	FALSE	
50.5	b_50_5_M2	BOOL	FALSE	FALSE	
50.6	b_50_6_M2	BOOL	FALSE	FALSE	
50.7	b_50_7_M2	BOOL	FALSE	FALSE	
51.0	b_51_M2	BYTE	B#16#0	B#16#0	
52.0	OPDAT_D_M2	BOOL	FALSE	FALSE	operating_data
52.1	ACT_BL_D_M2	BOOL	FALSE	FALSE	actual part program block
52.2	NXT_BL_D_M2	BOOL	FALSE	FALSE	next part program block
52.3	BLEXT_D_M2	BOOL	FALSE	FALSE	actual value at external block change
52.4	SERVDAT_D_M2	BOOL	FALSE	FALSE	service data
52.5	OC_ERR_D_M2	BOOL	FALSE	FALSE	oc_error
52.6	OT_ERR_D_M2	BOOL	FALSE	FALSE	ot_error
52.7	DA_ERR_D_M2	BOOL	FALSE	FALSE	data error
53.0	AXCOURD_D_M2	BOOL	FALSE	FALSE	axis coupling FM453
53.1	b_53_1_M2	BOOL	FALSE	FALSE	
53.2	b_53_2_M2	BOOL	FALSE	FALSE	
53.3	PARRD_D_M2	BOOL	FALSE	FALSE	parameter data
53.4	DIGIO_D_M2	BOOL	FALSE	FALSE	digital in_out
53.5	OPDAT1_D_M2	BOOL	FALSE	FALSE	operating data 1
53.6	APPDAT_D_M2	BOOL	FALSE	FALSE	applying data
53.7	MSRRD_D_M2	BOOL	FALSE	FALSE	read measurement values
54.0	SERVO_ERR_M2	BOOL	FALSE	FALSE	servo enable

Page 6 of 21

Address	Name	Type	Initial value	Actual value	Comment
54.1	GAUG_FLY_ERR_M2	BOOL	FALSE	FALSE	gauging on the fly
54.2	b_54_2_M2	BOOL	FALSE	FALSE	
54.3	b_54_3_M2	BOOL	FALSE	FALSE	
54.4	b_54_4_M2	BOOL	FALSE	FALSE	
54.5	TRAV_MON_ERR_M2	BOOL	FALSE	FALSE	traverse monitoring
54.6	PARK_AX_ERR_M2	BOOL	FALSE	FALSE	parking axis
54.7	SIM_ON_ERR_M2	BOOL	FALSE	FALSE	simulation on
55.0	b_55_0_M2	BOOL	FALSE	FALSE	
55.1	b_55_1_M2	BOOL	FALSE	FALSE	
55.2	MSR_ERR_M2	BOOL	FALSE	FALSE	measurement
55.3	REF_TRIG_ERR_M2	BOOL	FALSE	FALSE	trigger reference point
55.4	DI_OFF_ERR_M2	BOOL	FALSE	FALSE	disable enable input
55.5	FOLLOWUP_ERR_M2	BOOL	FALSE	FALSE	follow-up
55.6	SSW_DIS_ERR_M2	BOOL	FALSE	FALSE	software switch disable
55.7	DRIFT_OFF_ERR_M2	BOOL	FALSE	FALSE	automatic drift compensation off
56.0	b_56_M2	BYTE	B#16#0	B#16#0	
57.0	MD_ERR_M2	BOOL	FALSE	FALSE	machine data enable
57.1	DELDIST_ERR_M2	BOOL	FALSE	FALSE	delete distance to go
57.2	SEARCH_F_ERR_M2	BOOL	FALSE	FALSE	automatic block search forward
57.3	SEARCH_B_ERR_M2	BOOL	FALSE	FALSE	automatic block search backward
57.4	b_57_4_M2	BOOL	FALSE	FALSE	
57.5	RESET_AX_ERR_M2	BOOL	FALSE	FALSE	reset axis
57.6	AVALREM_ERR_M2	BOOL	FALSE	FALSE	remove setting actual value
57.7	b_57_7_M2	BOOL	FALSE	FALSE	
58.0	VLEV_ERR_M2	BOOL	FALSE	FALSE	velocity levels 1 and 2
58.1	CLEV_ERR_M2	BOOL	FALSE	FALSE	control levels 1 and 2
58.2	TRG254_ERR_M2	BOOL	FALSE	FALSE	target 254
58.3	MDI_ERR_M2	BOOL	FALSE	FALSE	MDI movement block
58.4	MDIFLY_ERR_M2	BOOL	FALSE	FALSE	MDI block on the fly
58.5	b_58_5_M2	BOOL	FALSE	FALSE	
58.6	REFPT_ERR_M2	BOOL	FALSE	FALSE	reference coordinate
58.7	AVAL_ERR_M2	BOOL	FALSE	FALSE	actual value coordinate
59.0	FVAL_ERR_M2	BOOL	FALSE	FALSE	actual value coordinate (flying)
59.1	ZOFF_ERR_M2	BOOL	FALSE	FALSE	zero offset value
59.2	b_59_2_M2	BOOL	FALSE	FALSE	
59.3	PARCH_ERR_M2	BOOL	FALSE	FALSE	parameter change
59.4	DIGO_ERR_M2	BOOL	FALSE	FALSE	digital 0

(Anexo M) 116

Page 7 of 21

Address	Name	Type	Initial value	Actual value	Comment
59.5	PROGS_ERR_M2	BOOL	FALSE	FALSE	part program selection
59.6	REQAPP_ERR_M2	BOOL	FALSE	FALSE	request application data
59.7	TEACHIN_ERR_M2	BOOL	FALSE	FALSE	Teach In
60.0	AXCOU_ERR_M2_M2	BOOL	FALSE	FALSE	axis coupling write error FM453
60.1	b_60_1_M2	BOOL	FALSE	FALSE	
60.2	b_60_2_M2	BOOL	FALSE	FALSE	
60.3	b_60_3_M2	BOOL	FALSE	FALSE	
60.4	b_60_4_M2	BOOL	FALSE	FALSE	
60.5	b_60_5_M2	BOOL	FALSE	FALSE	
60.6	b_60_6_M2	BOOL	FALSE	FALSE	
60.7	b_60_7_M2	BOOL	FALSE	FALSE	
61.0	b_61_M2	BYTE	B#16#0	B#16#0	
62.0	OPDAT_ERR_M2	BOOL	FALSE	FALSE	operating_data
62.1	ACT_BL_ERR_M2	BOOL	FALSE	FALSE	actual part program block
62.2	NXT_BL_ERR_M2	BOOL	FALSE	FALSE	next part program block
62.3	BLEXT_ERR_M2	BOOL	FALSE	FALSE	actual value at external block change
62.4	SERVDAT_ERR_M2	BOOL	FALSE	FALSE	service data
62.5	OC_ERR_ERR_M2	BOOL	FALSE	FALSE	oc_error
62.6	OT_ERR_ERR_M2	BOOL	FALSE	FALSE	ot_error
62.7	DA_ERR_ERR_M2	BOOL	FALSE	FALSE	data error
63.0	AXCOURD_ERR_M2	BOOL	FALSE	FALSE	axis coupling read error FM453
63.1	b_63_1_M2	BOOL	FALSE	FALSE	
63.2	b_63_2_M2	BOOL	FALSE	FALSE	
63.3	PARRD_ERR_M2	BOOL	FALSE	FALSE	parameter data
63.4	DIGIO_ERR_M2	BOOL	FALSE	FALSE	digital in_out
63.5	OPDAT1_ERR_M2	BOOL	FALSE	FALSE	operating data 1
63.6	APPDAT_ERR_M2	BOOL	FALSE	FALSE	applying data
63.7	MSRRD_ERR_M2	BOOL	FALSE	FALSE	read measurement values
64.0	b_64_M2	WORD	W#16#0	W#16#0	
66.0	JOB_ERR_M2	INT	0	0	error-code SFC58/59 (FC POS_CTRL)
68.0	JOBBUSY_WR_M2	BOOL	FALSE	FALSE	job aktiv write
68.1	IMPO_WR_M2	BOOL	FALSE	FALSE	imposs write
68.2	JOBBUSY_RD_M2	BOOL	FALSE	FALSE	job aktiv read
68.3	IMPO_RD_M2	BOOL	FALSE	FALSE	imposs read
68.4	b_68_4_M2	BOOL	FALSE	FALSE	
68.5	b_68_5_M2	BOOL	FALSE	FALSE	
68.6	b_68_6_M2	BOOL	FALSE	FALSE	

Page 8 of 21

Address	Name	Type	Initial value	Actual value	Comment
68.7	b_68_7_M2	BOOL	FALSE	FALSE	
69.0	b_69_0_M2	BOOL	FALSE	FALSE	
69.1	JOBRESET_M2	BOOL	FALSE	FALSE	Reset_ERR and _D
69.2	b_69_2_M2	BOOL	FALSE	FALSE	
69.3	b_69_3_M2	BOOL	FALSE	FALSE	
69.4	b_69_4_M2	BOOL	FALSE	FALSE	
69.5	b_69_5_M2	BOOL	FALSE	FALSE	
69.6	b_69_6_M2	BOOL	FALSE	FALSE	
69.7	b_69_7_M2	BOOL	FALSE	FALSE	
70.0	MDL_DEFECT_M2	BOOL	FALSE	FALSE	Module defective
70.1	INT_FAULT_M2	BOOL	FALSE	FALSE	Internal fault
70.2	EXT_FAULT_M2	BOOL	FALSE	FALSE	External fault
70.3	PNT_INFO_M2	BOOL	FALSE	FALSE	Point information
70.4	b_70_4_M2	BOOL	FALSE	FALSE	
70.5	FLD_CONNCTR_M2	BOOL	FALSE	FALSE	Field wiring connector missing
70.6	NO_CONFIG_M2	BOOL	FALSE	FALSE	Module has no configuration data
70.7	b_70_7_M2	BOOL	FALSE	FALSE	
71.0	MDL_TYPE_M2	BYTE	B#16#0	B#16#0	Type of module
72.0	b_72_0_M2	BOOL	FALSE	FALSE	
72.1	COMM_FAULT_M2	BOOL	FALSE	FALSE	Communication fault
72.2	b_72_2_M2	BOOL	FALSE	FALSE	
72.3	WTCH_DOG_FLT_M2	BOOL	FALSE	FALSE	Watch dog timer stopped module
72.4	INT_PS_FLT_M2	BOOL	FALSE	FALSE	Internal power supply fault
72.5	b_72_5_M2	BOOL	FALSE	FALSE	
72.6	b_72_6_M2	BOOL	FALSE	FALSE	
72.7	b_72_7_M2	BOOL	FALSE	FALSE	
73.0	b_73_0_M2	BOOL	FALSE	FALSE	
73.1	b_73_1_M2	BOOL	FALSE	FALSE	
73.2	EPROM_FLT_M2	BOOL	FALSE	FALSE	EPROM fault
73.3	RAM_FLT_M2	BOOL	FALSE	FALSE	RAM fault
73.4	b_73_4_M2	BOOL	FALSE	FALSE	
73.5	b_73_5_M2	BOOL	FALSE	FALSE	
73.6	HW_INTR_FLT_M2	BOOL	FALSE	FALSE	Hardware interrupt input in fault
73.7	b_73_7_M2	BOOL	FALSE	FALSE	
74.0	POS_ID_M2	BYTE	B#16#0	B#16#0	channel type info
75.0	LEN_INFO_M2	BYTE	B#16#0	B#16#0	channel info length
76.0	CHEN_NO_M2	BYTE	B#16#0	B#16#0	number of channels

(Anexo M) 117

Page 9 of 21

Address	Name	Type	Initial value	Actual value	Comment
77.0	CH_ERR_VE1_M2	BOOL	FALSE	FALSE	channel error vector 1
77.1	CH_ERR_VE2_M2	BOOL	FALSE	FALSE	channel error vector 2
77.2	CH_ERR_VE3_M2	BOOL	FALSE	FALSE	channel error vector 3
77.3	b_77_3_M2	BOOL	FALSE	FALSE	
77.4	b_77_4_M2	BOOL	FALSE	FALSE	
77.5	b_77_5_M2	BOOL	FALSE	FALSE	
77.6	b_77_6_M2	BOOL	FALSE	FALSE	
77.7	b_77_7_M2	BOOL	FALSE	FALSE	
78.0	CAB_BR1_M2	BOOL	FALSE	FALSE	channel 1 cablebraek
78.1	ERR_ABE1_M2	BOOL	FALSE	FALSE	channel 1 error sensor_absolut
78.2	ERR_PU1_M2	BOOL	FALSE	FALSE	channel 1 error impulse
78.3	VO_ENC1_M2	BOOL	FALSE	FALSE	channel 1 voltage control sensor
78.4	VO_15_1_M2	BOOL	FALSE	FALSE	channel 1 voltage control 15 V
78.5	VO_DIO1_M2	BOOL	FALSE	FALSE	channel 1 voltage control digital output
78.6	b_78_6_M2	BOOL	FALSE	FALSE	
78.7	OC_ERR_EN1_M2	BOOL	FALSE	FALSE	channel 1 operation error en1
79.0	b_79_M2	BYTE	B#16#0	B#16#0	
80.0	CAB_BR2_M2	BOOL	FALSE	FALSE	channel 2 cablebraek
80.1	ERR_ABE2_M2	BOOL	FALSE	FALSE	channel 2 error sensor_absolut
80.2	ERR_PU2_M2	BOOL	FALSE	FALSE	channel 2 error impulse
80.3	VO_ENC2_M2	BOOL	FALSE	FALSE	channel 2 voltage control sensor
80.4	VO_15_2_M2	BOOL	FALSE	FALSE	channel 2 voltage control 15 V
80.5	VO_DIO2_M2	BOOL	FALSE	FALSE	channel 2 voltage control digital output
80.6	b_80_6_M2	BOOL	FALSE	FALSE	
80.7	OC_ERR_EN2_M2	BOOL	FALSE	FALSE	channel 2 operation error en2
81.0	b_81_M2	BYTE	B#16#0	B#16#0	
82.0	CAB_BR3_M2	BOOL	FALSE	FALSE	channel 3 cablebraek
82.1	ERR_ABE3_M2	BOOL	FALSE	FALSE	channel 3 error sensor_absolut
82.2	ERR_PU3_M2	BOOL	FALSE	FALSE	channel 3 error impulse
82.3	VO_ENC3_M2	BOOL	FALSE	FALSE	channel 3 voltage control sensor
82.4	VO_15_3_M2	BOOL	FALSE	FALSE	channel 3 voltage control 15 V
82.5	VO_DIO3_M2	BOOL	FALSE	FALSE	channel 3 voltage control digital output
82.6	b_82_6_M2	BOOL	FALSE	FALSE	
82.7	OC_ERR_EN3_M2	BOOL	FALSE	FALSE	channel 3 operation error en3
83.0	b_83_M2	BYTE	B#16#0	B#16#0	
84.0	b_84_M2	WORD	W#16#0	W#16#0	
86.0	OC_ERR_NO_M2	BYTE	B#16#0	B#16#0	operation error, detail event number

Address	Name	Type	Initial value	Actual value	Comment
87.0	OC_REE_CL_M2	BYTE	B#16#0	B#16#0	detail event class
88.0	b_88_M2	WORD	W#16#0	W#16#0	
90.0	OT_ERR_NO_M2	BYTE	B#16#0	B#16#0	control/process error, detail event number
91.0	OT_ERR_CL_M2	BYTE	B#16#0	B#16#0	detail event class
92.0	b_92_M2	WORD	W#16#0	W#16#0	
94.0	DA_ERR_NO_M2	BYTE	B#16#0	B#16#0	data error, detail event number
95.0	DA_ERR_CL_M2	BYTE	B#16#0	B#16#0	detail event class
96.0	DIAG_ERR_M2	INT	0	0	error-code SFC51 (FC POS_DIAG)
98.0	MSRM_ERR_M2	INT	0	0	error-code SFC59 (FC POS_MSRM)
100.0	feld1_intern_M2[100]	BYTE	B#16#0	B#16#0	
101.0	feld1_intern_M2[101]	BYTE	B#16#0	B#16#0	
102.0	feld1_intern_M2[102]	BYTE	B#16#0	B#16#0	
103.0	feld1_intern_M2[103]	BYTE	B#16#0	B#16#0	
104.0	feld1_intern_M2[104]	BYTE	B#16#0	B#16#0	
105.0	feld1_intern_M2[105]	BYTE	B#16#0	B#16#0	
106.0	feld1_intern_M2[106]	BYTE	B#16#0	B#16#0	
107.0	feld1_intern_M2[107]	BYTE	B#16#0	B#16#0	
108.0	feld1_intern_M2[108]	BYTE	B#16#0	B#16#0	
109.0	feld1_intern_M2[109]	BYTE	B#16#0	B#16#0	
110.0	feld1_intern_M2[110]	BYTE	B#16#0	B#16#0	
111.0	feld1_intern_M2[111]	BYTE	B#16#0	B#16#0	
112.0	feld1_intern_M2[112]	BYTE	B#16#0	B#16#0	
113.0	feld1_intern_M2[113]	BYTE	B#16#0	B#16#0	
114.0	feld1_intern_M2[114]	BYTE	B#16#0	B#16#0	
115.0	feld1_intern_M2[115]	BYTE	B#16#0	B#16#0	
116.0	feld1_intern_M2[116]	BYTE	B#16#0	B#16#0	
117.0	feld1_intern_M2[117]	BYTE	B#16#0	B#16#0	
118.0	feld1_intern_M2[118]	BYTE	B#16#0	B#16#0	
119.0	feld1_intern_M2[119]	BYTE	B#16#0	B#16#0	
120.0	feld1_intern_M2[120]	BYTE	B#16#0	B#16#0	
121.0	feld1_intern_M2[121]	BYTE	B#16#0	B#16#0	
122.0	feld1_intern_M2[122]	BYTE	B#16#0	B#16#0	
123.0	feld1_intern_M2[123]	BYTE	B#16#0	B#16#0	
124.0	feld1_intern_M2[124]	BYTE	B#16#0	B#16#0	
125.0	feld1_intern_M2[125]	BYTE	B#16#0	B#16#0	
126.0	feld1_intern_M2[126]	BYTE	B#16#0	B#16#0	
127.0	feld1_intern_M2[127]	BYTE	B#16#0	B#16#0	

Address	Name	Type	Initial value	Actual value	Comment
128.0	feld1_intern_M2[128]	BYTE	B#16#0	B#16#0	
129.0	feld1_intern_M2[129]	BYTE	B#16#0	B#16#0	
130.0	feld1_intern_M2[130]	BYTE	B#16#0	B#16#0	
131.0	feld1_intern_M2[131]	BYTE	B#16#0	B#16#0	
132.0	feld1_intern_M2[132]	BYTE	B#16#0	B#16#0	
133.0	feld1_intern_M2[133]	BYTE	B#16#0	B#16#0	
134.0	feld1_intern_M2[134]	BYTE	B#16#0	B#16#0	
135.0	feld1_intern_M2[135]	BYTE	B#16#0	B#16#0	
136.0	feld1_intern_M2[136]	BYTE	B#16#0	B#16#0	
137.0	feld1_intern_M2[137]	BYTE	B#16#0	B#16#0	
138.0	feld1_intern_M2[138]	BYTE	B#16#0	B#16#0	
139.0	feld1_intern_M2[139]	BYTE	B#16#0	B#16#0	
140.0	ZOFF_M2	DINT	L#0	L#0	zero offset value
144.0	AVAL_M2	DINT	L#0	L#0	actual value coordinate
148.0	FVAL_M2	DINT	L#0	L#0	actual value coordinate (flying)
152.0	REFPT_M2	DINT	L#0	L#0	reference coordinate
156.0	TRG254_M2	DWORD	DW#16#0	DW#16#0	target 254
160.0	VLEVEL_1_M2	DWORD	DW#16#0	DW#16#0	velocity level 1
164.0	VLEVEL_2_M2	DWORD	DW#16#0	DW#16#0	velocity level 2
168.0	CLEVEL_1_M2	DWORD	DW#16#0	DW#16#0	control level 1
172.0	CLEVEL_2_M2	DWORD	DW#16#0	DW#16#0	control level 2
176.0	MDIB_M2.b_176_M2	BYTE	B#16#0	B#16#0	
177.0	MDIB_M2.b_177_M2	BYTE	B#16#0	B#16#0	
178.0	MDIB_M2.G_1_EN_M2	BOOL	FALSE	FALSE	enable bit for G function group 1
178.1	MDIB_M2.G_2_EN_M2	BOOL	FALSE	FALSE	enable bit for G function group 2
178.2	MDIB_M2.b_178_2_M2	BOOL	FALSE	FALSE	
178.3	MDIB_M2.b_178_3_M2	BOOL	FALSE	FALSE	
178.4	MDIB_M2.X_T_EN_M2	BOOL	FALSE	FALSE	enable bit for position / dwell time
178.5	MDIB_M2.b_178_5_M2	BOOL	FALSE	FALSE	
178.6	MDIB_M2.b_178_6_M2	BOOL	FALSE	FALSE	
178.7	MDIB_M2.b_178_7_M2	BOOL	FALSE	FALSE	
179.0	MDIB_M2.V_EN_M2	BOOL	FALSE	FALSE	enable bit for velocity
179.1	MDIB_M2.M_1_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 1
179.2	MDIB_M2.M_2_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 2
179.3	MDIB_M2.M_3_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 3
179.4	MDIB_M2.b_179_4_M2	BOOL	FALSE	FALSE	
179.5	MDIB_M2.b_179_5_M2	BOOL	FALSE	FALSE	

Page 12 of 21

Address	Name	Type	Initial value	Actual value	Comment
179.6	MDIB_M2.b_179_6_M2	BOOL	FALSE	FALSE	
179.7	MDIB_M2.b_179_7_M2	BOOL	FALSE	FALSE	
180.0	MDIB_M2.G_1_VAL_M2	BYTE	B#16#0	B#16#0	value G function group 1
181.0	MDIB_M2.G_2_VAL_M2	BYTE	B#16#0	B#16#0	value G function group 2
182.0	MDIB_M2.b_182_M2	BYTE	B#16#0	B#16#0	
183.0	MDIB_M2.b_183_M2	BYTE	B#16#0	B#16#0	
184.0	MDIB_M2.X_T_VAL_M2	DINT	L#0	L#0	value position / dwell time
188.0	MDIB_M2.V_VAL_M2	DINT	L#0	L#0	value velocity
192.0	MDIB_M2.M_1_VAL_M2	BYTE	B#16#0	B#16#0	value M function group 1
193.0	MDIB_M2.M_2_VAL_M2	BYTE	B#16#0	B#16#0	value M function group 2
194.0	MDIB_M2.M_3_VAL_M2	BYTE	B#16#0	B#16#0	value M function group 3
195.0	MDIB_M2.b_195_M2	BYTE	B#16#0	B#16#0	
196.0	PAR_CHAN_M2.TYP_M2	BYTE	B#16#0	B#16#0	DB type
197.0	PAR_CHAN_M2.NUMB_M2	BYTE	B#16#0	B#16#0	data number
198.0	PAR_CHAN_M2.COUN_M2	BYTE	B#16#0	B#16#0	number of data
199.0	PAR_CHAN_M2.JOB_M2	BYTE	B#16#0	B#16#0	job type
200.0	PAR_CHAN_M2.DATA_M2[200]	BYTE	B#16#0	B#16#0	
201.0	PAR_CHAN_M2.DATA_M2[201]	BYTE	B#16#0	B#16#0	
202.0	PAR_CHAN_M2.DATA_M2[202]	BYTE	B#16#0	B#16#0	
203.0	PAR_CHAN_M2.DATA_M2[203]	BYTE	B#16#0	B#16#0	
204.0	PAR_CHAN_M2.DATA_M2[204]	BYTE	B#16#0	B#16#0	
205.0	PAR_CHAN_M2.DATA_M2[205]	BYTE	B#16#0	B#16#0	
206.0	PAR_CHAN_M2.DATA_M2[206]	BYTE	B#16#0	B#16#0	
207.0	PAR_CHAN_M2.DATA_M2[207]	BYTE	B#16#0	B#16#0	
208.0	PAR_CHAN_M2.DATA_M2[208]	BYTE	B#16#0	B#16#0	
209.0	PAR_CHAN_M2.DATA_M2[209]	BYTE	B#16#0	B#16#0	
210.0	PAR_CHAN_M2.DATA_M2[210]	BYTE	B#16#0	B#16#0	
211.0	PAR_CHAN_M2.DATA_M2[211]	BYTE	B#16#0	B#16#0	
212.0	PAR_CHAN_M2.DATA_M2[212]	BYTE	B#16#0	B#16#0	
213.0	PAR_CHAN_M2.DATA_M2[213]	BYTE	B#16#0	B#16#0	
214.0	PAR_CHAN_M2.DATA_M2[214]	BYTE	B#16#0	B#16#0	
215.0	PAR_CHAN_M2.DATA_M2[215]	BYTE	B#16#0	B#16#0	
216.0	PAR_CHAN_M2.DATA_M2[216]	BYTE	B#16#0	B#16#0	
217.0	PAR_CHAN_M2.DATA_M2[217]	BYTE	B#16#0	B#16#0	
218.0	PAR_CHAN_M2.DATA_M2[218]	BYTE	B#16#0	B#16#0	
219.0	PAR_CHAN_M2.DATA_M2[219]	BYTE	B#16#0	B#16#0	
220.0	D_IN0_M2	BOOL	FALSE	FALSE	digital input 0

(Anexo M) 119

Page 13 of 21

Address	Name	Type	Initial value	Actual value	Comment
220.1	D_IN1_M2	BOOL	FALSE	FALSE	digital input 1
220.2	D_IN2_M2	BOOL	FALSE	FALSE	digital input 2
220.3	D_IN3_M2	BOOL	FALSE	FALSE	digital input 3
220.4	b_220_4_M2	BOOL	FALSE	FALSE	
220.5	b_220_5_M2	BOOL	FALSE	FALSE	
220.6	b_220_6_M2	BOOL	FALSE	FALSE	
220.7	b_220_7_M2	BOOL	FALSE	FALSE	
221.0	D_OUT0_M2	BOOL	FALSE	FALSE	digital output 0
221.1	D_OUT1_M2	BOOL	FALSE	FALSE	digital output 1
221.2	D_OUT2_M2	BOOL	FALSE	FALSE	digital output 2
221.3	D_OUT3_M2	BOOL	FALSE	FALSE	digital output 3
221.4	b_221_4_M2	BOOL	FALSE	FALSE	
221.5	b_221_5_M2	BOOL	FALSE	FALSE	
221.6	b_221_6_M2	BOOL	FALSE	FALSE	
221.7	b_221_7_M2	BOOL	FALSE	FALSE	
222.0	MDI_F_M2.b_222_M2	BYTE	B#16#0	B#16#0	
223.0	MDI_F_M2.b_223_M2	BYTE	B#16#0	B#16#0	
224.0	MDI_F_M2.G_1_EN_M2	BOOL	FALSE	FALSE	enable bit for G function group 1
224.1	MDI_F_M2.G_2_EN_M2	BOOL	FALSE	FALSE	enable bit for G function group 2
224.2	MDI_F_M2.b_224_2_M2	BOOL	FALSE	FALSE	
224.3	MDI_F_M2.b_224_3_M2	BOOL	FALSE	FALSE	
224.4	MDI_F_M2.X_T_EN_M2	BOOL	FALSE	FALSE	enable bit for position / dwell time
224.5	MDI_F_M2.b_224_5_M2	BOOL	FALSE	FALSE	
224.6	MDI_F_M2.b_224_6_M2	BOOL	FALSE	FALSE	
224.7	MDI_F_M2.b_224_7_M2	BOOL	FALSE	FALSE	
225.0	MDI_F_M2.V_EN_M2	BOOL	FALSE	FALSE	enable bit for velocity
225.1	MDI_F_M2.M_1_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 1
225.2	MDI_F_M2.M_2_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 2
225.3	MDI_F_M2.M_3_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 3
225.4	MDI_F_M2.b_225_4_M2	BOOL	FALSE	FALSE	
225.5	MDI_F_M2.b_225_5_M2	BOOL	FALSE	FALSE	
225.6	MDI_F_M2.b_225_6_M2	BOOL	FALSE	FALSE	
225.7	MDI_F_M2.b_225_7_M2	BOOL	FALSE	FALSE	
226.0	MDI_F_M2.G_1_VAL_M2	BYTE	B#16#0	B#16#0	value G function group 1
227.0	MDI_F_M2.G_2_VAL_M2	BYTE	B#16#0	B#16#0	value G function group 2
228.0	MDI_F_M2.b_228_M2	BYTE	B#16#0	B#16#0	
229.0	MDI_F_M2.b_229_M2	BYTE	B#16#0	B#16#0	

Address	Name	Type	Initial value	Actual value	Comment
230.0	MDI_F_M2.X_T_VAL_M2	DINT	L#0	L#0	value position / dwell time
234.0	MDI_F_M2.V_VAL_M2	DINT	L#0	L#0	value velocity
238.0	MDI_F_M2.M_1_VAL_M2	BYTE	B#16#0	B#16#0	value M function group 1
239.0	MDI_F_M2.M_2_VAL_M2	BYTE	B#16#0	B#16#0	value M function group 2
240.0	MDI_F_M2.M_3_VAL_M2	BYTE	B#16#0	B#16#0	value M function group 3
241.0	MDI_F_M2.b_241_M2	BYTE	B#16#0	B#16#0	
242.0	PROG_NO_M2	BYTE	B#16#0	B#16#0	part program number
243.0	BLCK_NO_M2	BYTE	B#16#0	B#16#0	part program block number
244.0	PROG_DIR_M2	BYTE	B#16#0	B#16#0	program run direction
245.0	b_245_M2	BYTE	B#16#0	B#16#0	
246.0	CODE_AP1_M2	BYTE	B#16#0	B#16#0	code application 1
247.0	CODE_AP2_M2	BYTE	B#16#0	B#16#0	code application 2
248.0	CODE_AP3_M2	BYTE	B#16#0	B#16#0	code application 3
249.0	CODE_AP4_M2	BYTE	B#16#0	B#16#0	code application 4
250.0	TEA_PRO_NO_M2	BYTE	B#16#0	B#16#0	part program number
251.0	TEA_BLK_NO_M2	BYTE	B#16#0	B#16#0	part program block number
252.0	AXCOU_DEF_M2	BYTE	B#16#0	B#16#0	axis coupling locate DS24 FM453
253.0	b_253_M2	BYTE	B#16#0	B#16#0	
254.0	feld2_intern_M2[254]	BYTE	B#16#0	B#16#0	
255.0	feld2_intern_M2[255]	BYTE	B#16#0	B#16#0	
256.0	feld2_intern_M2[256]	BYTE	B#16#0	B#16#0	
257.0	feld2_intern_M2[257]	BYTE	B#16#0	B#16#0	
258.0	feld2_intern_M2[258]	BYTE	B#16#0	B#16#0	
259.0	feld2_intern_M2[259]	BYTE	B#16#0	B#16#0	
260.0	feld2_intern_M2[260]	BYTE	B#16#0	B#16#0	
261.0	feld2_intern_M2[261]	BYTE	B#16#0	B#16#0	
262.0	feld2_intern_M2[262]	BYTE	B#16#0	B#16#0	
263.0	feld2_intern_M2[263]	BYTE	B#16#0	B#16#0	
264.0	feld2_intern_M2[264]	BYTE	B#16#0	B#16#0	
265.0	feld2_intern_M2[265]	BYTE	B#16#0	B#16#0	
266.0	feld2_intern_M2[266]	BYTE	B#16#0	B#16#0	
267.0	feld2_intern_M2[267]	BYTE	B#16#0	B#16#0	
268.0	feld2_intern_M2[268]	BYTE	B#16#0	B#16#0	
269.0	feld2_intern_M2[269]	BYTE	B#16#0	B#16#0	
270.0	feld2_intern_M2[270]	BYTE	B#16#0	B#16#0	
271.0	feld2_intern_M2[271]	BYTE	B#16#0	B#16#0	
272.0	feld2_intern_M2[272]	BYTE	B#16#0	B#16#0	

Address	Name	Type	Initial value	Actual value	Comment
273.0	feld2_intern_M2[273]	BYTE	B#16#0	B#16#0	
274.0	feld2_intern_M2[274]	BYTE	B#16#0	B#16#0	
275.0	feld2_intern_M2[275]	BYTE	B#16#0	B#16#0	
276.0	feld2_intern_M2[276]	BYTE	B#16#0	B#16#0	
277.0	feld2_intern_M2[277]	BYTE	B#16#0	B#16#0	
278.0	feld2_intern_M2[278]	BYTE	B#16#0	B#16#0	
279.0	feld2_intern_M2[279]	BYTE	B#16#0	B#16#0	
280.0	feld2_intern_M2[280]	BYTE	B#16#0	B#16#0	
281.0	feld2_intern_M2[281]	BYTE	B#16#0	B#16#0	
282.0	feld2_intern_M2[282]	BYTE	B#16#0	B#16#0	
283.0	feld2_intern_M2[283]	BYTE	B#16#0	B#16#0	
284.0	feld2_intern_M2[284]	BYTE	B#16#0	B#16#0	
285.0	feld2_intern_M2[285]	BYTE	B#16#0	B#16#0	
286.0	feld2_intern_M2[286]	BYTE	B#16#0	B#16#0	
287.0	feld2_intern_M2[287]	BYTE	B#16#0	B#16#0	
288.0	feld2_intern_M2[288]	BYTE	B#16#0	B#16#0	
289.0	feld2_intern_M2[289]	BYTE	B#16#0	B#16#0	
290.0	feld2_intern_M2[290]	BYTE	B#16#0	B#16#0	
291.0	feld2_intern_M2[291]	BYTE	B#16#0	B#16#0	
292.0	feld2_intern_M2[292]	BYTE	B#16#0	B#16#0	
293.0	feld2_intern_M2[293]	BYTE	B#16#0	B#16#0	
294.0	feld2_intern_M2[294]	BYTE	B#16#0	B#16#0	
295.0	feld2_intern_M2[295]	BYTE	B#16#0	B#16#0	
296.0	feld2_intern_M2[296]	BYTE	B#16#0	B#16#0	
297.0	feld2_intern_M2[297]	BYTE	B#16#0	B#16#0	
298.0	feld2_intern_M2[298]	BYTE	B#16#0	B#16#0	
299.0	feld2_intern_M2[299]	BYTE	B#16#0	B#16#0	
300.0	feld2_intern_M2[300]	BYTE	B#16#0	B#16#0	
301.0	feld2_intern_M2[301]	BYTE	B#16#0	B#16#0	
302.0	feld2_intern_M2[302]	BYTE	B#16#0	B#16#0	
303.0	feld2_intern_M2[303]	BYTE	B#16#0	B#16#0	
304.0	feld2_intern_M2[304]	BYTE	B#16#0	B#16#0	
305.0	feld2_intern_M2[305]	BYTE	B#16#0	B#16#0	
306.0	feld2_intern_M2[306]	BYTE	B#16#0	B#16#0	
307.0	feld2_intern_M2[307]	BYTE	B#16#0	B#16#0	
308.0	feld2_intern_M2[308]	BYTE	B#16#0	B#16#0	
309.0	feld2_intern_M2[309]	BYTE	B#16#0	B#16#0	

Page 16 of 21

Address	Name	Type	Initial value	Actual value	Comment
310.0	ACT_VAL_M2	DINT	L#0	L#0	actual value
314.0	SPEED_M2	DWORD	DW#16#0	DW#16#0	speed
318.0	REM_DIST_M2	DINT	L#0	L#0	remaining distance
322.0	SET_POS_M2	DINT	L#0	L#0	set position to be approached
326.0	SUM_OFST_M2	DINT	L#0	L#0	sum of tool offset and zero offset
330.0	TRAV_SPE_M2	DWORD	DW#16#0	DW#16#0	traversing speed
334.0	di_334_M2	DINT	L#0	L#0	
338.0	di_338_M2	DINT	L#0	L#0	
342.0	ACT_BL_M2.PROG_NO_M2	BYTE	B#16#0	B#16#0	part program number
343.0	ACT_BL_M2.BLCK_NO_M2	BYTE	B#16#0	B#16#0	part program block number
344.0	ACT_BL_M2.G_1_EN_M2	BOOL	FALSE	FALSE	enable bit for G function group 1
344.1	ACT_BL_M2.G_2_EN_M2	BOOL	FALSE	FALSE	enable bit for G function group 2
344.2	ACT_BL_M2.G_3_EN_M2	BOOL	FALSE	FALSE	enable bit for G function group 3
344.3	ACT_BL_M2.b_344_3_M2	BOOL	FALSE	FALSE	
344.4	ACT_BL_M2.X_T_EN_M2	BOOL	FALSE	FALSE	enable bit for position / dwell time
344.5	ACT_BL_M2.SR_L_EN_M2	BOOL	FALSE	FALSE	enable bit for number of subroutine loops
344.6	ACT_BL_M2.SR_N_EN_M2	BOOL	FALSE	FALSE	enable bit for subroutine number
344.7	ACT_BL_M2.SKIP_EN_M2	BOOL	FALSE	FALSE	enable bit for block skip
345.0	ACT_BL_M2.V_EN_M2	BOOL	FALSE	FALSE	enable bit for velocity
345.1	ACT_BL_M2.M_1_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 1
345.2	ACT_BL_M2.M_2_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 2
345.3	ACT_BL_M2.M_3_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 3
345.4	ACT_BL_M2.TO_EN_M2	BOOL	FALSE	FALSE	enable bit for tool offset number
345.5	ACT_BL_M2.b_345_5_M2	BOOL	FALSE	FALSE	
345.6	ACT_BL_M2.b_345_6_M2	BOOL	FALSE	FALSE	
345.7	ACT_BL_M2.b_345_7_M2	BOOL	FALSE	FALSE	
346.0	ACT_BL_M2.G_1_VAL_M2	BYTE	B#16#0	B#16#0	G function group 1
347.0	ACT_BL_M2.G_2_VAL_M2	BYTE	B#16#0	B#16#0	G function group 2
348.0	ACT_BL_M2.G_3_VAL_M2	BYTE	B#16#0	B#16#0	G function group 3
349.0	ACT_BL_M2.b_349_M2	BYTE	B#16#0	B#16#0	
350.0	ACT_BL_M2.X_T_VAL_M2	DINT	L#0	L#0	position / dwell time / subroutine number
354.0	ACT_BL_M2.V_VAL_M2	DINT	L#0	L#0	velocity / subroutine loops
358.0	ACT_BL_M2.M_1_VAL_M2	BYTE	B#16#0	B#16#0	M function group 1
359.0	ACT_BL_M2.M_2_VAL_M2	BYTE	B#16#0	B#16#0	M function group 2
360.0	ACT_BL_M2.M_3_VAL_M2	BYTE	B#16#0	B#16#0	M function group 3
361.0	ACT_BL_M2.TO_VAL_M2	BYTE	B#16#0	B#16#0	tool offset number
362.0	NXT_BLK_M2.PROG_NO_M2	BYTE	B#16#0	B#16#0	part program number

(Anexo M) 121

Page 17 of 21

Address	Name	Type	Initial value	Actual value	Comment
363.0	NXT_BLK_M2.BLCK_NO_M2	BYTE	B#16#0	B#16#0	part program block number
364.0	NXT_BLK_M2.G_1_EN_M2	BOOL	FALSE	FALSE	enable bit for G function group 1
364.1	NXT_BLK_M2.G_2_EN_M2	BOOL	FALSE	FALSE	enable bit for G function group 2
364.2	NXT_BLK_M2.G_3_EN_M2	BOOL	FALSE	FALSE	enable bit for G function group 3
364.3	NXT_BLK_M2.b_364_3_M2	BOOL	FALSE	FALSE	
364.4	NXT_BLK_M2.X_T_EN_M2	BOOL	FALSE	FALSE	enable bit for position / dwell time
364.5	NXT_BLK_M2.SR_L_EN_M2	BOOL	FALSE	FALSE	enable bit for number of subroutine loops
364.6	NXT_BLK_M2.SR_N_EN_M2	BOOL	FALSE	FALSE	enable bit for subroutine number
364.7	NXT_BLK_M2.SKIP_EN_M2	BOOL	FALSE	FALSE	enable bit for block skip
365.0	NXT_BLK_M2.V_EN_M2	BOOL	FALSE	FALSE	enable bit for velocity
365.1	NXT_BLK_M2.M_1_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 1
365.2	NXT_BLK_M2.M_2_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 2
365.3	NXT_BLK_M2.M_3_EN_M2	BOOL	FALSE	FALSE	enable bit for M function group 3
365.4	NXT_BLK_M2.TO_EN_M2	BOOL	FALSE	FALSE	enable bit for tool offset number
365.5	NXT_BLK_M2.b_365_5_M2	BOOL	FALSE	FALSE	
365.6	NXT_BLK_M2.b_365_6_M2	BOOL	FALSE	FALSE	
365.7	NXT_BLK_M2.b_365_7_M2	BOOL	FALSE	FALSE	
366.0	NXT_BLK_M2.G_1_VAL_M2	BYTE	B#16#0	B#16#0	G function group 1
367.0	NXT_BLK_M2.G_2_VAL_M2	BYTE	B#16#0	B#16#0	G function group 2
368.0	NXT_BLK_M2.G_3_VAL_M2	BYTE	B#16#0	B#16#0	G function group 3
369.0	NXT_BLK_M2.b_369_M2	BYTE	B#16#0	B#16#0	
370.0	NXT_BLK_M2.X_T_VAL_M2	DINT	L#0	L#0	position / dwell time / subroutine number
374.0	NXT_BLK_M2.V_VAL_M2	DINT	L#0	L#0	velocity / subroutine loops
378.0	NXT_BLK_M2.M_1_VAL_M2	BYTE	B#16#0	B#16#0	M function group 1
379.0	NXT_BLK_M2.M_2_VAL_M2	BYTE	B#16#0	B#16#0	M function group 2
380.0	NXT_BLK_M2.M_3_VAL_M2	BYTE	B#16#0	B#16#0	M function group 3
381.0	NXT_BLK_M2.TO_VAL_M2	BYTE	B#16#0	B#16#0	tool offset number
382.0	APP1_M2	DINT	L#0	L#0	application data 1
386.0	APP2_M2	DINT	L#0	L#0	application data 2
390.0	APP3_M2	DINT	L#0	L#0	application data 3
394.0	APP4_M2	DINT	L#0	L#0	application data 4
398.0	BLCK_EXT_M2	DINT	L#0	L#0	actual value at external block change
402.0	OUT_VAL_M2	DINT	L#0	L#0	DAC value / frequency
406.0	ENC_VAL_M2	DINT	L#0	L#0	encoder actual value
410.0	PULS_ERR_M2	DINT	L#0	L#0	error pulses
414.0	KV_FA_M2	DINT	L#0	L#0	Kv factor
418.0	FOLL_ERR_M2	DINT	L#0	L#0	following error

Address	Name	Type	Initial value	Actual value	Comment
422.0	FERR_LIM_M2	DINT	L#0	L#0	following error limit
426.0	OSC_ERR_M2	DINT	L#0	L#0	oscillation error
430.0	DR_TIME_M2	DINT	L#0	L#0	drive time
434.0	OVERRIDE1_M2	BYTE	B#16#0	B#16#0	override
435.0	PROG_NO1_M2	BYTE	B#16#0	B#16#0	part program number
436.0	BLCK_NO1_M2	BYTE	B#16#0	B#16#0	part program block number
437.0	LOOP_NO1_M2	BYTE	B#16#0	B#16#0	number of subroutine loops
438.0	G90_91_M2	BYTE	B#16#0	B#16#0	G90/91 active
439.0	G60_64_M2	BYTE	B#16#0	B#16#0	G60/64 active
440.0	G43_44_M2	BYTE	B#16#0	B#16#0	G43/44 active
441.0	TO_NO_M2	BYTE	B#16#0	B#16#0	tool offset number
442.0	b_442_0_M2	BOOL	FALSE	FALSE	
442.1	LIM_SP_M2	BOOL	FALSE	FALSE	limitation speed
442.2	LIM_10_M2	BOOL	FALSE	FALSE	limitation +/- 10V
442.3	LIM_SU_M2	BOOL	FALSE	FALSE	limitation speed_up
442.4	b_442_4_M2	BOOL	FALSE	FALSE	
442.5	b_442_5_M2	BOOL	FALSE	FALSE	
442.6	b_442_6_M2	BOOL	FALSE	FALSE	
442.7	b_442_7_M2	BOOL	FALSE	FALSE	
443.0	LIM_FR_M2	BOOL	FALSE	FALSE	limitation frequency
443.1	LIM_FV_M2	BOOL	FALSE	FALSE	limitation frequency value
443.2	b_443_2_M2	BOOL	FALSE	FALSE	
443.3	LIM_FS_M2	BOOL	FALSE	FALSE	limitation frequency speed-up
443.4	b_443_4_M2	BOOL	FALSE	FALSE	
443.5	b_443_5_M2	BOOL	FALSE	FALSE	
443.6	b_443_6_M2	BOOL	FALSE	FALSE	
443.7	b_443_7_M2	BOOL	FALSE	FALSE	
444.0	b_444_M2	BYTE	B#16#0	B#16#0	
445.0	b_445_M2	BYTE	B#16#0	B#16#0	
446.0	PAR_RD_M2.TYP1_M2	BYTE	B#16#0	B#16#0	DB type
447.0	PAR_RD_M2.NUMB_M2	BYTE	B#16#0	B#16#0	data number
448.0	PAR_RD_M2.COUN_M2	BYTE	B#16#0	B#16#0	number of data
449.0	PAR_RD_M2.JOB_M2	BYTE	B#16#0	B#16#0	job type
450.0	PAR_RD_M2.DATA1_M2[450]	BYTE	B#16#0	B#16#0	
451.0	PAR_RD_M2.DATA1_M2[451]	BYTE	B#16#0	B#16#0	
452.0	PAR_RD_M2.DATA1_M2[452]	BYTE	B#16#0	B#16#0	
453.0	PAR_RD_M2.DATA1_M2[453]	BYTE	B#16#0	B#16#0	

Address	Name	Type	Initial value	Actual value	Comment
454.0	PAR_RD_M2.DATA1_M2[454]	BYTE	B#16#0	B#16#0	
455.0	PAR_RD_M2.DATA1_M2[455]	BYTE	B#16#0	B#16#0	
456.0	PAR_RD_M2.DATA1_M2[456]	BYTE	B#16#0	B#16#0	
457.0	PAR_RD_M2.DATA1_M2[457]	BYTE	B#16#0	B#16#0	
458.0	PAR_RD_M2.DATA1_M2[458]	BYTE	B#16#0	B#16#0	
459.0	PAR_RD_M2.DATA1_M2[459]	BYTE	B#16#0	B#16#0	
460.0	PAR_RD_M2.DATA1_M2[460]	BYTE	B#16#0	B#16#0	
461.0	PAR_RD_M2.DATA1_M2[461]	BYTE	B#16#0	B#16#0	
462.0	PAR_RD_M2.DATA1_M2[462]	BYTE	B#16#0	B#16#0	
463.0	PAR_RD_M2.DATA1_M2[463]	BYTE	B#16#0	B#16#0	
464.0	PAR_RD_M2.DATA1_M2[464]	BYTE	B#16#0	B#16#0	
465.0	PAR_RD_M2.DATA1_M2[465]	BYTE	B#16#0	B#16#0	
466.0	PAR_RD_M2.DATA1_M2[466]	BYTE	B#16#0	B#16#0	
467.0	PAR_RD_M2.DATA1_M2[467]	BYTE	B#16#0	B#16#0	
468.0	PAR_RD_M2.DATA1_M2[468]	BYTE	B#16#0	B#16#0	
469.0	PAR_RD_M2.DATA1_M2[469]	BYTE	B#16#0	B#16#0	
470.0	AXCOU_STAT_M2	BYTE	B#16#0	B#16#0	status axis coupling DS37 FM453
471.0	b_471_M2	BYTE	B#16#0	B#16#0	
472.0	feld3_intern_M2[472]	BYTE	B#16#0	B#16#0	
473.0	feld3_intern_M2[473]	BYTE	B#16#0	B#16#0	
474.0	feld3_intern_M2[474]	BYTE	B#16#0	B#16#0	
475.0	feld3_intern_M2[475]	BYTE	B#16#0	B#16#0	
476.0	feld3_intern_M2[476]	BYTE	B#16#0	B#16#0	
477.0	feld3_intern_M2[477]	BYTE	B#16#0	B#16#0	
478.0	feld3_intern_M2[478]	BYTE	B#16#0	B#16#0	
479.0	feld3_intern_M2[479]	BYTE	B#16#0	B#16#0	
480.0	feld3_intern_M2[480]	BYTE	B#16#0	B#16#0	
481.0	feld3_intern_M2[481]	BYTE	B#16#0	B#16#0	
482.0	feld3_intern_M2[482]	BYTE	B#16#0	B#16#0	
483.0	feld3_intern_M2[483]	BYTE	B#16#0	B#16#0	
484.0	feld3_intern_M2[484]	BYTE	B#16#0	B#16#0	
485.0	feld3_intern_M2[485]	BYTE	B#16#0	B#16#0	
486.0	BEGIN_VAL_M2	DINT	L#0	L#0	begin value
490.0	END_VAL_M2	DINT	L#0	L#0	end value
494.0	LENGTH_VAL_M2	DWORD	DW#16#0	DW#16#0	length value
498.0	USR_M2.BITC_0_M2	BOOL	FALSE	FALSE	MD write request
498.1	USR_M2.BITC_1_M2	BOOL	FALSE	FALSE	MD read request

Page 20 of 21

Address	Name	Type	Initial value	Actual value	Comment
498.2	USR_M2.BITC_2_M2	BOOL	FALSE	FALSE	MDI block request
498.3	USR_M2.BITC_3_M2	BOOL	FALSE	FALSE	part program selection request
498.4	USR_M2.BITC_4_M2	BOOL	FALSE	FALSE	Teach In request
498.5	USR_M2.BITC_5_M2	BOOL	FALSE	FALSE	incremental value request
498.6	USR_M2.BITC_6_M2	BOOL	FALSE	FALSE	velocity levels request
498.7	USR_M2.BITC_7_M2	BOOL	FALSE	FALSE	control levels request
499.0	USR_M2.BITC_8_M2	BOOL	FALSE	FALSE	MDI block on the fly request
499.1	USR_M2.BITC_9_M2	BOOL	FALSE	FALSE	actual value on the fly request
499.2	USR_M2.BITC_10_M2	BOOL	FALSE	FALSE	zero point offset request
499.3	USR_M2.BITC_11_M2	BOOL	FALSE	FALSE	reserved
499.4	USR_M2.BITC_12_M2	BOOL	FALSE	FALSE	reserved
499.5	USR_M2.BITC_13_M2	BOOL	FALSE	FALSE	Alarm
499.6	USR_M2.BITC_14_M2	BOOL	FALSE	FALSE	Data-Error
499.7	USR_M2.BITC_15_M2	BOOL	FALSE	FALSE	OT-Error
500.0	USR_M2.MD_NO_M2	WORD	W#16#0	W#16#0	MD number
502.0	USR_M2.MD_VALUE_M2	DINT	L#0	L#0	MD value
506.0	USR_M2.INC_NO_M2	BYTE	B#16#0	B#16#0	incremental value number
507.0	USR_M2.b_507_M2	BYTE	B#16#0	B#16#0	
508.0	USR_M2.PICT_NO_M2	WORD	W#16#0	W#16#0	picture number
510.0	USR_M2.KEY_CODE_M2	WORD	W#16#0	W#16#0	keyboard code
512.0	USR_M2.b_512_M2	WORD	W#16#0	W#16#0	
514.0	USR_M2.BITA_0_M2	BOOL	FALSE	FALSE	Control
514.1	USR_M2.BITA_1_M2	BOOL	FALSE	FALSE	Reference point approach
514.2	USR_M2.BITA_2_M2	BOOL	FALSE	FALSE	Increment
514.3	USR_M2.BITA_3_M2	BOOL	FALSE	FALSE	MDI
514.4	USR_M2.BITA_4_M2	BOOL	FALSE	FALSE	Automatic / single move
514.5	USR_M2.BITA_5_M2	BOOL	FALSE	FALSE	Automatic / follow move
514.6	USR_M2.BITA_6_M2	BOOL	FALSE	FALSE	Jog-Mode
514.7	USR_M2.b_514_7_M2	BOOL	FALSE	FALSE	
515.0	USR_M2.b_515_0_M2	BOOL	FALSE	FALSE	
515.1	USR_M2.b_515_1_M2	BOOL	FALSE	FALSE	
515.2	USR_M2.b_515_2_M2	BOOL	FALSE	FALSE	
515.3	USR_M2.b_515_3_M2	BOOL	FALSE	FALSE	
515.4	USR_M2.b_515_4_M2	BOOL	FALSE	FALSE	
515.5	USR_M2.b_515_5_M2	BOOL	FALSE	FALSE	
515.6	USR_M2.BITA_14_M2	BOOL	FALSE	FALSE	Quit Error
515.7	USR_M2.BITA_15_M2	BOOL	FALSE	FALSE	Quit Alarm

(Anexo M) 123

Page 21 of 21

DB3 - <offline>
"DB_UDT_ENC1"**Data view**

Data block type:

UDT3 data block

Name: DB_UDTE2 **Family:** FM_CNT_1
Author: FM **Version:** 0.1
Block version: 2
Lengths (Block / Data): 00436 / 00070
Time stamp
Code: 07/18/2012 04:54:43 PM
Interface: 11/21/2002 09:28:37 AM

Comment:

Address	Name	Type	Initial value	Actual value	Comment
0.0	AR1_BUFFER	DWORD	DW#16#0	DW#16#0	AR1 buffer (FC internal use)
4.0	FP	BYTE	B#16#0	B#16#0	Flag byte (FC internal use)
5.0	RESERVED	BYTE	B#16#0	B#16#0	reserved for FC use
6.0	MOD_ADR	WORD	W#16#0	W#16#140	Module address (write user)
8.0	CH_ADR	DWORD	DW#16#0	DW#16#A00	Channel address (write user)
12.0	U_D_LGTH	BYTE	B#16#0	B#16#10	User data length (write user)
13.0	A_BYTE_0	BYTE	B#16#0	B#16#0	reserved
14.0	LOAD_VAL	DINT	L#0	L#0	New load value (write user)
18.0	CMP_V1	DINT	L#0	L#0	New comparator value 1 (write user)
22.0	CMP_V2	DINT	L#0	L#0	New comparator value 2 (write user)
26.0	A_BIT0_0	BOOL	FALSE	FALSE	reserved
26.1	TFB	BOOL	FALSE	FALSE	Test free (internal use)
26.2	A_BIT0_2	BOOL	FALSE	FALSE	reserved
26.3	A_BIT0_3	BOOL	FALSE	FALSE	reserved
26.4	A_BIT0_4	BOOL	FALSE	FALSE	reserved
26.5	A_BIT0_5	BOOL	FALSE	FALSE	reserved
26.6	A_BIT0_6	BOOL	FALSE	FALSE	reserved
26.7	A_BIT0_7	BOOL	FALSE	FALSE	reserved
27.0	ENSET_UP	BOOL	FALSE	FALSE	Enable set in direction up (=forward) (write user)

Page 1 of 5

Address	Name	Type	Initial value	Actual value	Comment
27.1	ENSET_DN	BOOL	FALSE	FALSE	Enable set in direction down (=backward) (write user)
27.2	A_BIT1_2	BOOL	FALSE	FALSE	reserved
27.3	A_BIT1_3	BOOL	FALSE	FALSE	reserved
27.4	A_BIT1_4	BOOL	FALSE	FALSE	reserved
27.5	A_BIT1_5	BOOL	FALSE	FALSE	reserved
27.6	A_BIT1_6	BOOL	FALSE	FALSE	reserved
27.7	A_BIT1_7	BOOL	FALSE	FALSE	reserved
28.0	CTRL_D00	BOOL	FALSE	FALSE	Control digital output D00 (write user)
28.1	CTRL_D01	BOOL	FALSE	FALSE	Control digital output D01 (write user)
28.2	A_BIT2_2	BOOL	FALSE	FALSE	reserved
28.3	A_BIT2_3	BOOL	FALSE	FALSE	reserved
28.4	A_BIT2_4	BOOL	FALSE	FALSE	reserved
28.5	A_BIT2_5	BOOL	FALSE	FALSE	reserved
28.6	A_BIT2_6	BOOL	FALSE	FALSE	reserved
28.7	A_BIT2_7	BOOL	FALSE	FALSE	reserved
29.0	A_BIT3_0	BOOL	FALSE	FALSE	reserved
29.1	A_BIT3_1	BOOL	FALSE	FALSE	reserved
29.2	A_BIT3_2	BOOL	FALSE	FALSE	reserved
29.3	A_BIT3_3	BOOL	FALSE	FALSE	reserved
29.4	A_BIT3_4	BOOL	FALSE	FALSE	reserved
29.5	A_BIT3_5	BOOL	FALSE	FALSE	reserved
29.6	A_BIT3_6	BOOL	FALSE	FALSE	reserved
29.7	A_BIT3_7	BOOL	FALSE	FALSE	reserved
30.0	LATCH_LOAD	DINT	L#0	L#0	Actual latch or load value (read user)
34.0	ACT_CNTV	DINT	L#0	L#0	Actual counter value (read user)
38.0	DA_ERR_W	WORD	W#16#0	W#16#0	Data error word (read user)
40.0	OT_ERR_B	BYTE	B#16#0	B#16#0	Operator error byte (read user)
41.0	E_BIT0_0	BOOL	FALSE	FALSE	reserved
41.1	STS_TFB	BOOL	FALSE	FALSE	Status test free bit (internal use)
41.2	DIAG	BOOL	FALSE	FALSE	1=Diagnostic buffer changed (read user)
41.3	E_BIT0_3	BOOL	FALSE	FALSE	reserved
41.4	DATA_ERR	BOOL	FALSE	FALSE	Data error bit (read user)
41.5	E_BIT0_5	BOOL	FALSE	FALSE	reserved
41.6	E_BIT0_6	BOOL	FALSE	FALSE	reserved
41.7	PARA	BOOL	FALSE	FALSE	Module parametrized (read user)
42.0	E_BYTE_0	BYTE	B#16#0	B#16#0	reserved

Address	Name	Type	Initial value	Actual value	Comment
43.0	STS_RUN	BOOL	FALSE	FALSE	Status counter is running (read user)
43.1	STS_DIR	BOOL	FALSE	FALSE	Status of the counter direction bit (read user)
43.2	STS_ZERO	BOOL	FALSE	FALSE	Status of the counter zero crossing bit (read user)
43.3	STS_OFLW	BOOL	FALSE	FALSE	Status counter passed the overflow value (read user)
43.4	STS_UFLW	BOOL	FALSE	FALSE	Status counter passed the underflow value (read user)
43.5	STS_SYNC	BOOL	FALSE	FALSE	Status counter is synchronized (read user)
43.6	STS_GATE	BOOL	FALSE	FALSE	Status of the internal gate (read user)
43.7	STS_SW_G	BOOL	FALSE	FALSE	Status of the software gate (read user)
44.0	STS_SET	BOOL	FALSE	FALSE	Status digital input SET (read user)
44.1	STS_LATCH	BOOL	FALSE	FALSE	New latch value in isochronous mode (read user)
44.2	STS_STA	BOOL	FALSE	FALSE	Status of the digital input start (read user)
44.3	STS_STP	BOOL	FALSE	FALSE	Status of the digital input stop (read user)
44.4	STS_CMP1	BOOL	FALSE	FALSE	Status of the comparator output 1 (read user)
44.5	STS_CMP2	BOOL	FALSE	FALSE	Status of the comparator output 2 (read user)
44.6	STS_COMP1	BOOL	FALSE	FALSE	Remanent status of the comparator output 1 (read user)
44.7	STS_COMP2	BOOL	FALSE	FALSE	Remanent status of the comparator output 2 (read user)
45.0	E_BIT3_0	BOOL	FALSE	FALSE	reserved
45.1	E_BIT3_1	BOOL	FALSE	FALSE	reserved
45.2	E_BIT3_2	BOOL	FALSE	FALSE	reserved
45.3	E_BIT3_3	BOOL	FALSE	FALSE	reserved
45.4	E_BIT3_4	BOOL	FALSE	FALSE	reserved
45.5	E_BIT3_5	BOOL	FALSE	FALSE	reserved
45.6	E_BIT3_6	BOOL	FALSE	FALSE	reserved
45.7	E_BIT3_7	BOOL	FALSE	FALSE	reserved
46.0	ACT_CMP1	DINT	L#0	L#0	Actual comparator value 1 (FM 450 only; read user)
50.0	ACT_CMP2	DINT	L#0	L#0	Actual comparator value 2 (FM 450 only; read user)
54.0	MDL_DEFECT	BOOL	FALSE	FALSE	Module defective
54.1	INT_FAULT	BOOL	FALSE	FALSE	Internal fault

Address	Name	Type	Initial value	Actual value	Comment
54.2	EXT_FAULT	BOOL	FALSE	FALSE	External fault
54.3	PNT_INFO	BOOL	FALSE	FALSE	Point information
54.4	EXT_VOLTAGE	BOOL	FALSE	FALSE	External voltage low
54.5	FLD_CONNCTR	BOOL	FALSE	FALSE	Field wiring connector missing
54.6	NO_CONFIG	BOOL	FALSE	FALSE	Module has no configuration data
54.7	CONFIG_ERR	BOOL	FALSE	FALSE	Module has configuration error
55.0	MDL_TYPE	BYTE	B#16#0	B#16#0	Type of module
56.0	SUB_MDL_ERR	BOOL	FALSE	FALSE	Sub-Module is missing or has error
56.1	COMM_FAULT	BOOL	FALSE	FALSE	Communication fault
56.2	MDL_STOP	BOOL	FALSE	FALSE	Module is stopped
56.3	WTCH_DOG_FLT	BOOL	FALSE	FALSE	Watch dog timer stopped module
56.4	INT_PS_FLT	BOOL	FALSE	FALSE	Internal power supply fault
56.5	PRIM_BATT_FLT	BOOL	FALSE	FALSE	Primary battery is in fault
56.6	BCKUP_BATT_FLT	BOOL	FALSE	FALSE	Backup battery is in fault
56.7	RESERVED_2	BOOL	FALSE	FALSE	reserved for system
57.0	RACK_FLT	BOOL	FALSE	FALSE	Rack fault, only for bus interface module
57.1	PROC_FLT	BOOL	FALSE	FALSE	Processor fault
57.2	EPROM_FLT	BOOL	FALSE	FALSE	EPROM fault
57.3	RAM_FLT	BOOL	FALSE	FALSE	RAM fault
57.4	ADU_FLT	BOOL	FALSE	FALSE	ADU fault
57.5	FUSE_FLT	BOOL	FALSE	FALSE	Fuse fault
57.6	HW_INTR_FLT	BOOL	FALSE	FALSE	Hardware interrupt input in fault
57.7	RESERVED_3	BOOL	FALSE	FALSE	reserved for system
58.0	CH_TYPE	BYTE	B#16#0	B#16#0	Channel type
59.0	LGTH_DIA	BYTE	B#16#0	B#16#0	Length of diagnostics data per channel
60.0	CH_NO	BYTE	B#16#0	B#16#0	Channel number (numero)
61.0	GRP_ERR1	BOOL	FALSE	FALSE	Group error channel 1
61.1	GRP_ERR2	BOOL	FALSE	FALSE	Group error channel 2
61.2	D_BIT7_2	BOOL	FALSE	FALSE	DS1 byte 7 bit 2
61.3	D_BIT7_3	BOOL	FALSE	FALSE	DS1 byte 7 bit 3
61.4	D_BIT7_4	BOOL	FALSE	FALSE	DS1 byte 7 bit 4
61.5	D_BIT7_5	BOOL	FALSE	FALSE	DS1 byte 7 bit 5
61.6	D_BIT7_6	BOOL	FALSE	FALSE	DS1 byte 7 bit 6
61.7	D_BIT7_7	BOOL	FALSE	FALSE	DS1 byte 7 bit 7
62.0	CH1_SIGA	BOOL	FALSE	FALSE	Channel 1, signal A malfunction
62.1	CH1_SIGB	BOOL	FALSE	FALSE	Channel 1, signal B malfunction
62.2	CH1_SIGZ	BOOL	FALSE	FALSE	Channel 1, signal Zero mark malfunction

Address	Name	Type	Initial value	Actual value	Comment
62.3	CH1_BETW	BOOL	FALSE	FALSE	Channel 1, malfunction between channels
62.4	CH1_5V2	BOOL	FALSE	FALSE	Channel 1, malfunction of encoder supply voltage 5.2V
62.5	D_BIT8_5	BOOL	FALSE	FALSE	DS1 byte 8 bit 5
62.6	D_BIT8_6	BOOL	FALSE	FALSE	DS1 byte 8 bit 6
62.7	D_BIT8_7	BOOL	FALSE	FALSE	DS1 byte 8 bit 7
63.0	D_BYTE9	BYTE	B#16#0	B#16#0	DS1 byte 9
64.0	CH2_SIGA	BOOL	FALSE	FALSE	Channel 2, signal A malfunction
64.1	CH2_SIGB	BOOL	FALSE	FALSE	Channel 2, signal B malfunction
64.2	CH2_SIGZ	BOOL	FALSE	FALSE	Channel 2, signal Zero mark malfunction
64.3	CH2_BETW	BOOL	FALSE	FALSE	Channel 2, malfunction between channels
64.4	CH2_5V2	BOOL	FALSE	FALSE	Channel 2, malfunction of encoder supply voltage 5.2V
64.5	D_BIT10_5	BOOL	FALSE	FALSE	DS1 byte 10 bit 5
64.6	D_BIT10_6	BOOL	FALSE	FALSE	DS1 byte 10 bit 6
64.7	D_BIT10_7	BOOL	FALSE	FALSE	DS1 byte 10 bit 7
65.0	D_BYTE11	BYTE	B#16#0	B#16#0	DS1 byte 11
66.0	D_BYTE12	BYTE	B#16#0	B#16#0	DS1 byte 12
67.0	D_BYTE13	BYTE	B#16#0	B#16#0	DS1 byte 13
68.0	D_BYTE14	BYTE	B#16#0	B#16#0	DS1 byte 14
69.0	D_BYTE15	BYTE	B#16#0	B#16#0	DS1 byte 15

DB4 - <offline>

"DB_UDT_ENC2"

Data view

Data block type:

UDT4 data block

Name: DB_UDTE2 **Family:** FM_CNT_1
Author: FM **Version:** 0.1
Block version: 2
Lengths (Block / Data): 00436 / 00070

Time stamp

Code: 07/18/2012 04:54:56 PM

Interface: 11/21/2002 09:28:37 AM

Comment:

Address	Name	Type	Initial value	Actual value	Comment
0.0	AR1_BUFFER	DWORD	DW#16#0	DW#16#0	AR1 buffer (FC internal use)
4.0	FP	BYTE	B#16#0	B#16#0	Flag byte (FC internal use)
5.0	RESERVED	BYTE	B#16#0	B#16#0	reserved for FC use
6.0	MOD_ADR	WORD	W#16#0	W#16#150	Module address (write user)
8.0	CH_ADR	DWORD	DW#16#0	DW#16#A80	Channel address (write user)
12.0	U_D_LGTH	BYTE	B#16#0	B#16#10	User data length (write user)
13.0	A_BYTE_0	BYTE	B#16#0	B#16#0	reserved
14.0	LOAD_VAL	DINT	L#0	L#0	New load value (write user)
18.0	CMP_V1	DINT	L#0	L#0	New comparator value 1 (write user)
22.0	CMP_V2	DINT	L#0	L#0	New comparator value 2 (write user)
26.0	A_BIT0_0	BOOL	FALSE	FALSE	reserved
26.1	TFB	BOOL	FALSE	FALSE	Test free (internal use)
26.2	A_BIT0_2	BOOL	FALSE	FALSE	reserved
26.3	A_BIT0_3	BOOL	FALSE	FALSE	reserved
26.4	A_BIT0_4	BOOL	FALSE	FALSE	reserved
26.5	A_BIT0_5	BOOL	FALSE	FALSE	reserved
26.6	A_BIT0_6	BOOL	FALSE	FALSE	reserved
26.7	A_BIT0_7	BOOL	FALSE	FALSE	reserved
27.0	ENSET_UP	BOOL	FALSE	FALSE	Enable set in direction up (=forward) (write user)

Address	Name	Type	Initial value	Actual value	Comment
27.1	ENSET_DN	BOOL	FALSE	FALSE	Enable set in direction down (=backward) (write user)
27.2	A_BIT1_2	BOOL	FALSE	FALSE	reserved
27.3	A_BIT1_3	BOOL	FALSE	FALSE	reserved
27.4	A_BIT1_4	BOOL	FALSE	FALSE	reserved
27.5	A_BIT1_5	BOOL	FALSE	FALSE	reserved
27.6	A_BIT1_6	BOOL	FALSE	FALSE	reserved
27.7	A_BIT1_7	BOOL	FALSE	FALSE	reserved
28.0	CTRL_D00	BOOL	FALSE	FALSE	Control digital output D00 (write user)
28.1	CTRL_D01	BOOL	FALSE	FALSE	Control digital output D01 (write user)
28.2	A_BIT2_2	BOOL	FALSE	FALSE	reserved
28.3	A_BIT2_3	BOOL	FALSE	FALSE	reserved
28.4	A_BIT2_4	BOOL	FALSE	FALSE	reserved
28.5	A_BIT2_5	BOOL	FALSE	FALSE	reserved
28.6	A_BIT2_6	BOOL	FALSE	FALSE	reserved
28.7	A_BIT2_7	BOOL	FALSE	FALSE	reserved
29.0	A_BIT3_0	BOOL	FALSE	FALSE	reserved
29.1	A_BIT3_1	BOOL	FALSE	FALSE	reserved
29.2	A_BIT3_2	BOOL	FALSE	FALSE	reserved
29.3	A_BIT3_3	BOOL	FALSE	FALSE	reserved
29.4	A_BIT3_4	BOOL	FALSE	FALSE	reserved
29.5	A_BIT3_5	BOOL	FALSE	FALSE	reserved
29.6	A_BIT3_6	BOOL	FALSE	FALSE	reserved
29.7	A_BIT3_7	BOOL	FALSE	FALSE	reserved
30.0	LATCH_LOAD	DINT	L#0	L#0	Actual latch or load value (read user)
34.0	ACT_CNTV	DINT	L#0	L#0	Actual counter value (read user)
38.0	DA_ERR_W	WORD	W#16#0	W#16#0	Data error word (read user)
40.0	OT_ERR_B	BYTE	B#16#0	B#16#0	Operator error byte (read user)
41.0	E_BIT0_0	BOOL	FALSE	FALSE	reserved
41.1	STS_TFB	BOOL	FALSE	FALSE	Status test free bit (internal use)
41.2	DIAG	BOOL	FALSE	FALSE	1-Diagnostic buffer changed (read user)
41.3	E_BIT0_3	BOOL	FALSE	FALSE	reserved
41.4	DATA_ERR	BOOL	FALSE	FALSE	Data error bit (read user)
41.5	E_BIT0_5	BOOL	FALSE	FALSE	reserved
41.6	E_BIT0_6	BOOL	FALSE	FALSE	reserved
41.7	PARA	BOOL	FALSE	FALSE	Module parametrized (read user)
42.0	E_BYTE_0	BYTE	B#16#0	B#16#0	reserved

Address	Name	Type	Initial value	Actual value	Comment
43.0	STS_RUN	BOOL	FALSE	FALSE	Status counter is running (read user)
43.1	STS_DIR	BOOL	FALSE	FALSE	Status of the counter direction bit (read user)
43.2	STS_ZERO	BOOL	FALSE	FALSE	Status of the counter zero crossing bit (read user)
43.3	STS_OFLW	BOOL	FALSE	FALSE	Status counter passed the overflow value (read user)
43.4	STS_UFLW	BOOL	FALSE	FALSE	Status counter passed the underflow value (read user)
43.5	STS_SYNC	BOOL	FALSE	FALSE	Status counter is synchronized (read user)
43.6	STS_GATE	BOOL	FALSE	FALSE	Status of the internal gate (read user)
43.7	STS_SW_G	BOOL	FALSE	FALSE	Status of the software gate (read user)
44.0	STS_SET	BOOL	FALSE	FALSE	Status digital input SET (read user)
44.1	STS_LATCH	BOOL	FALSE	FALSE	New latch value in isochronous mode (read user)
44.2	STS_STA	BOOL	FALSE	FALSE	Status of the digital input start (read user)
44.3	STS_STP	BOOL	FALSE	FALSE	Status of the digital input stop (read user)
44.4	STS_CMP1	BOOL	FALSE	FALSE	Status of the comparator output 1 (read user)
44.5	STS_CMP2	BOOL	FALSE	FALSE	Status of the comparator output 2 (read user)
44.6	STS_COMP1	BOOL	FALSE	FALSE	Remanent status of the comparator output 1 (read user)
44.7	STS_COMP2	BOOL	FALSE	FALSE	Remanent status of the comparator output 2 (read user)
45.0	E_BIT3_0	BOOL	FALSE	FALSE	reserved
45.1	E_BIT3_1	BOOL	FALSE	FALSE	reserved
45.2	E_BIT3_2	BOOL	FALSE	FALSE	reserved
45.3	E_BIT3_3	BOOL	FALSE	FALSE	reserved
45.4	E_BIT3_4	BOOL	FALSE	FALSE	reserved
45.5	E_BIT3_5	BOOL	FALSE	FALSE	reserved
45.6	E_BIT3_6	BOOL	FALSE	FALSE	reserved
45.7	E_BIT3_7	BOOL	FALSE	FALSE	reserved
46.0	ACT_CMP1	DINT	L#0	L#0	Actual comparator value 1 (FM 450 only; read user)
50.0	ACT_CMP2	DINT	L#0	L#0	Actual comparator value 2 (FM 450 only; read user)
54.0	MDL_DEFECT	BOOL	FALSE	FALSE	Module defective
54.1	INT_FAULT	BOOL	FALSE	FALSE	Internal fault

Address	Name	Type	Initial value	Actual value	Comment
54.2	EXT_FAULT	BOOL	FALSE	FALSE	External fault
54.3	PNT_INFO	BOOL	FALSE	FALSE	Point information
54.4	EXT_VOLTAGE	BOOL	FALSE	FALSE	External voltage low
54.5	FLD_CONNCTR	BOOL	FALSE	FALSE	Field wiring connector missing
54.6	NO_CONFIG	BOOL	FALSE	FALSE	Module has no configuration data
54.7	CONFIG_ERR	BOOL	FALSE	FALSE	Module has configuration error
55.0	MDL_TYPE	BYTE	B#16#0	B#16#0	Type of module
56.0	SUB_MDL_ERR	BOOL	FALSE	FALSE	Sub-Module is missing or has error
56.1	COMM_FAULT	BOOL	FALSE	FALSE	Communication fault
56.2	MDL_STOP	BOOL	FALSE	FALSE	Module is stopped
56.3	WTCH_DOG_FLT	BOOL	FALSE	FALSE	Watch dog timer stopped module
56.4	INT_PS_FLT	BOOL	FALSE	FALSE	Internal power supply fault
56.5	PRIM_BATT_FLT	BOOL	FALSE	FALSE	Primary battery is in fault
56.6	BCKUP_BATT_FLT	BOOL	FALSE	FALSE	Backup battery is in fault
56.7	RESERVED_2	BOOL	FALSE	FALSE	reserved for system
57.0	RACK_FLT	BOOL	FALSE	FALSE	Rack fault, only for bus interface module
57.1	PROC_FLT	BOOL	FALSE	FALSE	Processor fault
57.2	EPROM_FLT	BOOL	FALSE	FALSE	EPROM fault
57.3	RAM_FLT	BOOL	FALSE	FALSE	RAM fault
57.4	ADU_FLT	BOOL	FALSE	FALSE	ADU fault
57.5	FUSE_FLT	BOOL	FALSE	FALSE	Fuse fault
57.6	HW_INTR_FLT	BOOL	FALSE	FALSE	Hardware interrupt input in fault
57.7	RESERVED_3	BOOL	FALSE	FALSE	reserved for system
58.0	CH_TYPE	BYTE	B#16#0	B#16#0	Channel type
59.0	LGTH_DIA	BYTE	B#16#0	B#16#0	Length of diagnostics data per channel
60.0	CH_NO	BYTE	B#16#0	B#16#0	Channel number (numero)
61.0	GRP_ERR1	BOOL	FALSE	FALSE	Group error channel 1
61.1	GRP_ERR2	BOOL	FALSE	FALSE	Group error channel 2
61.2	D_BIT7_2	BOOL	FALSE	FALSE	DS1 byte 7 bit 2
61.3	D_BIT7_3	BOOL	FALSE	FALSE	DS1 byte 7 bit 3
61.4	D_BIT7_4	BOOL	FALSE	FALSE	DS1 byte 7 bit 4
61.5	D_BIT7_5	BOOL	FALSE	FALSE	DS1 byte 7 bit 5
61.6	D_BIT7_6	BOOL	FALSE	FALSE	DS1 byte 7 bit 6
61.7	D_BIT7_7	BOOL	FALSE	FALSE	DS1 byte 7 bit 7
62.0	CH1_SIGA	BOOL	FALSE	FALSE	Channel 1, signal A malfunction
62.1	CH1_SIGB	BOOL	FALSE	FALSE	Channel 1, signal B malfunction
62.2	CH1_SIGZ	BOOL	FALSE	FALSE	Channel 1, signal Zero mark malfunction

Address	Name	Type	Initial value	Actual value	Comment
62.3	CH1_BETW	BOOL	FALSE	FALSE	Channel 1, malfunction between channels
62.4	CH1_5V2	BOOL	FALSE	FALSE	Channel 1, malfunction of encoder supply voltage 5.2V
62.5	D_BIT8_5	BOOL	FALSE	FALSE	DS1 byte 8 bit 5
62.6	D_BIT8_6	BOOL	FALSE	FALSE	DS1 byte 8 bit 6
62.7	D_BIT8_7	BOOL	FALSE	FALSE	DS1 byte 8 bit 7
63.0	D_BYTE9	BYTE	B#16#0	B#16#0	DS1 byte 9
64.0	CH2_SIGA	BOOL	FALSE	FALSE	Channel 2, signal A malfunction
64.1	CH2_SIGB	BOOL	FALSE	FALSE	Channel 2, signal B malfunction
64.2	CH2_SIGZ	BOOL	FALSE	FALSE	Channel 2, signal Zero mark malfunction
64.3	CH2_BETW	BOOL	FALSE	FALSE	Channel 2, malfunction between channels
64.4	CH2_5V2	BOOL	FALSE	FALSE	Channel 2, malfunction of encoder supply voltage 5.2V
64.5	D_BIT10_5	BOOL	FALSE	FALSE	DS1 byte 10 bit 5
64.6	D_BIT10_6	BOOL	FALSE	FALSE	DS1 byte 10 bit 6
64.7	D_BIT10_7	BOOL	FALSE	FALSE	DS1 byte 10 bit 7
65.0	D_BYTE11	BYTE	B#16#0	B#16#0	DS1 byte 11
66.0	D_BYTE12	BYTE	B#16#0	B#16#0	DS1 byte 12
67.0	D_BYTE13	BYTE	B#16#0	B#16#0	DS1 byte 13
68.0	D_BYTE14	BYTE	B#16#0	B#16#0	DS1 byte 14
69.0	D_BYTE15	BYTE	B#16#0	B#16#0	DS1 byte 15

DB21 - <offline>

"CONV_T_DB"

Data view

Data block type:

Instance data block for FB21

Name: CONV_T_D
Author: EACF
Family:
Version: 0.1
Block version: 2
 Lengths (Block / Data): 00094 / 00004
Time stamp
Code: 08/01/2012 04:56:14 PM
Interface: 07/31/2012 08:04:58 PM

Comment:

Address	Declaration	Name	Type	Initial value	Actual value	Comment
0.0	in	dt1_tempo_in	INT	0	0	
2.0	out	dt1_tempo_out	S5TIME	S5T#0MS	S5T#0MS	

Page 1 of 1

DB100 - <offline> - Declaration view

"DBEX_100_M1" Interface (FC100, DB1, UDT1)
 Global data block DB 100
Name: DB100_M1 **Family:** FM_ST_SV
Author: AuDMCE8 **Version:** 2.0
Block version: 2
Time stamp Code: 07/18/2012 06:15:48 PM
Interface: 04/26/2009 12:05:29 AM
Lengths (block/logic/data): 00184 00018 00000

Block: DB100 DBEX

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	ERR_CODE_INIT	INT	0	Error code FC POS_INIT
+2.0	ERR_CODE_CTRL	INT	0	Error code FC POS_CTRL
+4.0	ERR_CODE_DIAG	INT	0	Error code FC POS_DIAG
+6.0	OVERRIDE	BYTE	B#16#0	Override
+7.0	MODE_IN	BYTE	B#16#0	Mode setting (coded)
+8.0	MODE_OUT	BYTE	B#16#0	Mode status (codiert)
+9.0	DRV_EN	BOOL	FALSE	Drive enable
+9.1	SERVO_EN	BOOL	FALSE	Servo enable
+9.2	OT_ERR_A	BOOL	FALSE	Operator/traversing error acknow.
+9.3	RESET_AX	BOOL	FALSE	Restart
+9.4	DIAG_RD	BOOL	FALSE	Job start FC POS_DIAG
+9.5	PARAM	BOOL	FALSE	Parameterized
+9.6	SYNC	BOOL	FALSE	Synchronized
+9.7	START_EN	BOOL	FALSE	Start enable
+10.0	POS_RCD	BOOL	FALSE	Position reached
+10.1	WORKING	BOOL	FALSE	Machining in progress
+10.2	GO_M	BOOL	FALSE	Travel -
+10.3	GO_P	BOOL	FALSE	Travel +
+10.4	OT_ERR	BOOL	FALSE	Operator/traversing error
+10.5	DATA_ERR	BOOL	FALSE	Data error
+10.6	INIT_ERR	BOOL	FALSE	Error FC POS_INIT
+10.7	DIAG_ERR	BOOL	FALSE	Error FC POS_DIAG
+11.0	MINUS1	BOOL	FALSE	Error "MINUS1" in FC POS_CTRL

(Anexo M) 129

Page 1 of 2

Address	Name	Type	Initial value	Comment
+11.1	MINUS2	BOOL	FALSE	Error "MINUS2" in FC POS_CTRL
+11.2	MINUS3	BOOL	FALSE	Error "MINUS3" in FC POS_CTRL
+12.0	EX1	STRUCT		*** Signals for EXAMPLE 1 ***
+0.0	DIR_M	BOOL	FALSE	Travel - command
+0.1	DIR_P	BOOL	FALSE	Travel + command
+0.2	START	BOOL	FALSE	Start
+0.3	STOP	BOOL	FALSE	Stop
=2.0		END_STRUCT		
+14.0	EX2	STRUCT		*** Signals for EXAMPLE 2 ***
+0.0	START	BOOL	FALSE	Start
+0.1	STOP	BOOL	FALSE	Stop
=2.0		END_STRUCT		
+16.0	EX3	STRUCT		*** Signals for EXAMPLE 3 ***
+0.0	START	BOOL	FALSE	Start
+0.1	STOP	BOOL	FALSE	Stop
+0.2	READ_EN	BOOL	FALSE	Read enable
=2.0		END_STRUCT		
=18.0		END_STRUCT		

DB200 - <offline> - Declaration view

"DBEX_200_M2" Interface (FC200, DB2, UDT2)
 Global data block DB 200
Name: DB200_M2 **Family:** FM_ST_SV
Author: AuDMCE8 **Version:** 2.0
Block version: 2
Time stamp Code: 07/18/2012 05:58:24 PM
Interface: 04/26/2009 12:05:29 AM
Lengths (block/logic/data): 00184 00018 00000

Block: DB200	DBEX
---------------------	-------------

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	ERR_CODE_INIT_M2	INT	0	Error code FC POS_INIT
+2.0	ERR_CODE_CTRL_M2	INT	0	Error code FC POS_CTRL
+4.0	ERR_CODE_DIAG_M2	INT	0	Error code FC POS_DIAG
+6.0	OVERRIDE_M2	BYTE	B#16#0	Override
+7.0	MODE_IN_M2	BYTE	B#16#0	Mode setting (coded)
+8.0	MODE_OUT_M2	BYTE	B#16#0	Mode status (codiert)
+9.0	DRV_EN_M2	BOOL	FALSE	Drive enable
+9.1	SERVO_EN_M2	BOOL	FALSE	Servo enable
+9.2	OT_ERR_A_M2	BOOL	FALSE	Operator/traversing error acknow.
+9.3	RESET_AX_M2	BOOL	FALSE	Restart
+9.4	DIAG_RD_M2	BOOL	FALSE	Job start FC POS_DIAG
+9.5	PARAM_M2	BOOL	FALSE	Parameterized
+9.6	SYNC_M2	BOOL	FALSE	Synchronized
+9.7	START_EN_M2	BOOL	FALSE	Start enable
+10.0	POS_RCD_M2	BOOL	FALSE	Position reached, stop
+10.1	WORKING_M2	BOOL	FALSE	Machining in progress
+10.2	GO_M_M2	BOOL	FALSE	Travel -
+10.3	GO_P_M2	BOOL	FALSE	Travel +
+10.4	OT_ERR_M2	BOOL	FALSE	Operator/traversing error
+10.5	DATA_ERR_M2	BOOL	FALSE	Data error
+10.6	INIT_ERR_M2	BOOL	FALSE	Error FC POS_INIT
+10.7	DIAG_ERR_M2	BOOL	FALSE	Error FC POS_DIAG
+11.0	MINUS1_M2	BOOL	FALSE	Error "MINUS1" in FC POS_CTRL

Address	Name	Type	Initial value	Comment
+11.1	MINUS2_M2	BOOL	FALSE	Error "MINUS2" in FC POS_CTRL
+11.2	MINUS3_M2	BOOL	FALSE	Error "MINUS3" in FC POS_CTRL
+12.0	EX1_M2	STRUCT		*** Signals for EXAMPLE 1 ***
+0.0	DIR_M_M2	BOOL	FALSE	Travel - command
+0.1	DIR_P_M2	BOOL	FALSE	Travel + command
+0.2	START_M2	BOOL	FALSE	Start
+0.3	STOP_M2	BOOL	FALSE	Stop
=2.0		END_STRUCT		
+14.0	EX2_M2	STRUCT		*** Signals for EXAMPLE 2 ***
+0.0	START_M2	BOOL	FALSE	Start
+0.1	STOP_M2	BOOL	FALSE	Stop
=2.0		END_STRUCT		
+16.0	EX3_M2	STRUCT		*** Signals for EXAMPLE 3 ***
+0.0	START_M2	BOOL	FALSE	Start
+0.1	STOP_M2	BOOL	FALSE	Stop
+0.2	READ_EN_M2	BOOL	FALSE	Read enable
=2.0		END_STRUCT		
=18.0		END_STRUCT		

DB300 - <offline>

"DB_PROG_ENC1"

DB300 (FB300 DB instance)

Data view

Data block type:

Instance data block for FB300

Name: DB_FB300 Family: FM_CNT_1
 Author: FM Version: 0.1
 Block version: 2
 Lengths (Block / Data): 00238 / 00042

Time stamp

Code: 07/18/2012 05:58:56 PM

Interface: 11/27/2002 12:00:24 PM

Comment:

Address	Declaration	Name	Type	Initial value	Actual value	Comment
0.0	in	SW_GATE	BOOL	FALSE	FALSE	1=Control software gate
0.1	in	GATE_STP	BOOL	FALSE	FALSE	1=Stop gate
0.2	in	OT_ERR_A	BOOL	FALSE	FALSE	1=Confirm operator error
0.3	in	DIAG_ERR_A	BOOL	FALSE	FALSE	1=Confirm diagnostic interrupt
0.4	in	HW_INT_ERR_A	BOOL	FALSE	FALSE	1=Confirm hardware interrupt
0.5	in	SET_D00	BOOL	FALSE	FALSE	1=Set output D00
0.6	in	SET_D01	BOOL	FALSE	FALSE	1=Set output D01
0.7	in	SET_L_DIRECT	BOOL	FALSE	FALSE	1=New counter value
1.0	in	SET_L_PREPAR	BOOL	FALSE	FALSE	1=Prepare new counter value
1.1	in	SET_T_CMP_V1	BOOL	FALSE	FALSE	1=New compare value 1
1.2	in	SET_T_CMP_V2	BOOL	FALSE	FALSE	1=New compare value 2
1.3	in	SET_C_DOPARA	BOOL	FALSE	FALSE	1=Triggering parameter change
1.4	in	SET_RES_SYNC	BOOL	FALSE	FALSE	1=Reset status bit "synchronization"
1.5	in	SET_RES_ZERO	BOOL	FALSE	FALSE	1=Reset status bit "zero-crossing"
1.6	in	ENSET_UP	BOOL	FALSE	FALSE	1=Enable set in direction up (forward)
1.7	in	ENSET_DN	BOOL	FALSE	FALSE	1=Enable set in direction down (backward)
2.0	in	CTRL_D00	BOOL	FALSE	FALSE	1=Control digital output D00
2.1	in	CTRL_D01	BOOL	FALSE	FALSE	1=Control digital output D01
4.0	in	L_DIRECT_VAL	DINT	L#0	L#0	New counter value

Address	Declaration	Name	Type	Initial value	Actual value	Comment
8.0	in	L_PREPAR_VAL	DINT	L#0	L#0	New preparation value
12.0	in	T_CMP_V1_VAL	DINT	L#0	L#0	New compare value 1
16.0	in	T_CMP_V2_VAL	DINT	L#0	L#0	New compare value 2
20.0	in	DO0_MODE	BYTE	B#16#0	B#16#0	Set DO0 behavior (0..5)
21.0	in	DO1_MODE	BYTE	B#16#0	B#16#0	Set DO1 behavior (0..6)
22.0	in	HYSSTERESIS	BYTE	B#16#0	B#16#0	Hysteresis value (0..255)
23.0	in	PULSE_DURATION	BYTE	B#16#0	B#16#0	Pulse duration (0..250)
24.0	out	OT_ERR	BOOL	FALSE	FALSE	Error in CNT_CTL1 function
24.1	out	DIAG_ERR	BOOL	FALSE	FALSE	1=Diagnostic interrupt received
24.2	out	HW_INT_ERR	BOOL	FALSE	FALSE	1=Hardware interrupt received
24.3	out	STS_RUN	BOOL	FALSE	FALSE	"Counter is running" status
24.4	out	STS_DIR	BOOL	FALSE	FALSE	"Direction bit" status
24.5	out	STS_ZERO	BOOL	FALSE	FALSE	"Zero-Crossing" status
24.6	out	STS_OFLW	BOOL	FALSE	FALSE	"Counter Overflow" status
24.7	out	STS_UFLW	BOOL	FALSE	FALSE	"Counter Underflow" status
25.0	out	STS_SW_G	BOOL	FALSE	FALSE	"Software Gate" status
25.1	out	STS_GATE	BOOL	FALSE	FALSE	"Internal Gate" status
26.0	out	OT_ERR_B	BYTE	B#16#0	B#16#0	Operator error
28.0	out	LATCH_LOAD	DWORD	DW#16#0	DW#16#0	Current load value or latch value
32.0	out	ACT_CNTV	DWORD	DW#16#0	DW#16#0	Current count value
36.0	in_out	L_DIRECT	BOOL	FALSE	FALSE	1=Load new counter value
36.1	in_out	L_PREPAR	BOOL	FALSE	FALSE	1=Prepare new counter value
36.2	in_out	T_CMP_V1	BOOL	FALSE	FALSE	1=Load new compare value 1
36.3	in_out	T_CMP_V2	BOOL	FALSE	FALSE	1=Load new compare value 2
36.4	in_out	C_DOPARA	BOOL	FALSE	FALSE	1=Triggering parameter change
36.5	in_out	RES_SYNC	BOOL	FALSE	FALSE	1=Reset status bit "synchronization"
36.6	in_out	RES_ZERO	BOOL	FALSE	FALSE	1=Reset status bit "zero-crossing"
36.7	in_out	DIAG_INF	BOOL	FALSE	FALSE	1=Read diagnostics (only for OB82)
38.0	stat	FP_L_DIRECT	BOOL	FALSE	FALSE	1=Load new counter value
38.1	stat	FP_L_PREPAR	BOOL	FALSE	FALSE	1=Prepare new counter value
38.2	stat	FP_T_CMP_V1	BOOL	FALSE	FALSE	1=Load new compare value 1
38.3	stat	FP_T_CMP_V2	BOOL	FALSE	FALSE	1=Load new compare value 2
38.4	stat	FP_C_DOPARA	BOOL	FALSE	FALSE	1=Triggering parameter change
38.5	stat	FP_RES_SYNC	BOOL	FALSE	FALSE	1=Reset status bit "synchronization"
38.6	stat	FP_RES_ZERO	BOOL	FALSE	FALSE	1=Reset status bit "zero-crossing"
40.0	stat	SFC_ERR	INT	0	0	SFC51 error status

DB400 - <offline>

"DB_PROG_ENC2"

DB400 (FB400 DB instance)

Data view

Data block type:

Instance data block for FB400

Name: DB_FB400 **Family:** FM_CNT_1
Author: FM **Version:** 0.1
Block version: 2
00238 / 00042

Lengths (Block / Data):

Time stamp

Code: 07/18/2012 05:59:06 PM**Interface:** 11/27/2002 12:00:24 PM

Comment:

Address	Declaration	Name	Type	Initial value	Actual value	Comment
0.0	in	SW_GATE	BOOL	FALSE	FALSE	1=Control software gate
0.1	in	GATE_STP	BOOL	FALSE	FALSE	1=Stop gate
0.2	in	OT_ERR_A	BOOL	FALSE	FALSE	1=Confirm operator error
0.3	in	DIAG_ERR_A	BOOL	FALSE	FALSE	1=Confirm diagnostic interrupt
0.4	in	HW_INT_ERR_A	BOOL	FALSE	FALSE	1=Confirm hardware interrupt
0.5	in	SET_D00	BOOL	FALSE	FALSE	1=Set output D00
0.6	in	SET_D01	BOOL	FALSE	FALSE	1=Set output D01
0.7	in	SET_L_DIRECT	BOOL	FALSE	FALSE	1=New counter value
1.0	in	SET_L_PREPAR	BOOL	FALSE	FALSE	1=Prepare new counter value
1.1	in	SET_T_CMP_V1	BOOL	FALSE	FALSE	1=New compare value 1
1.2	in	SET_T_CMP_V2	BOOL	FALSE	FALSE	1=New compare value 2
1.3	in	SET_C_DOPARA	BOOL	FALSE	FALSE	1=Triggering parameter change
1.4	in	SET_RES_SYNC	BOOL	FALSE	FALSE	1=Reset status bit "synchronization"
1.5	in	SET_RES_ZERO	BOOL	FALSE	FALSE	1=Reset status bit "zero-crossing"
1.6	in	ENSET_UP	BOOL	FALSE	FALSE	1=Enable set in direction up (forward)
1.7	in	ENSET_DN	BOOL	FALSE	FALSE	1=Enable set in direction down (backward)
2.0	in	CTRL_D00	BOOL	FALSE	FALSE	1=Control digital output D00
2.1	in	CTRL_D01	BOOL	FALSE	FALSE	1=Control digital output D01
4.0	in	L_DIRECT_VAL	DINT	L#0	L#0	New counter value

Address	Declaration	Name	Type	Initial value	Actual value	Comment
8.0	in	L_PREPAR_VAL	DINT	L#0	L#0	New preparation value
12.0	in	T_CMP_V1_VAL	DINT	L#0	L#0	New compare value 1
16.0	in	T_CMP_V2_VAL	DINT	L#0	L#0	New compare value 2
20.0	in	DO0_MODE	BYTE	B#16#0	B#16#0	Set DO0 behavior (0..5)
21.0	in	DO1_MODE	BYTE	B#16#0	B#16#0	Set DO1 behavior (0..6)
22.0	in	HYSTERESIS	BYTE	B#16#0	B#16#0	Hysteresis value (0..255)
23.0	in	PULSE_DURATION	BYTE	B#16#0	B#16#0	Pulse duration (0..250)
24.0	out	OT_ERR	BOOL	FALSE	FALSE	Error in CNT_CTL1 function
24.1	out	DIAG_ERR	BOOL	FALSE	FALSE	1=Diagnostic interrupt received
24.2	out	HW_INT_ERR	BOOL	FALSE	FALSE	1=Hardware interrupt received
24.3	out	STS_RUN	BOOL	FALSE	FALSE	"Counter is running" status
24.4	out	STS_DIR	BOOL	FALSE	FALSE	"Direction bit" status
24.5	out	STS_ZERO	BOOL	FALSE	FALSE	"Zero-Crossing" status
24.6	out	STS_OFLW	BOOL	FALSE	FALSE	"Counter Overflow" status
24.7	out	STS_UFLW	BOOL	FALSE	FALSE	"Counter Underflow" status
25.0	out	STS_SW_G	BOOL	FALSE	FALSE	"Software Gate" status
25.1	out	STS_GATE	BOOL	FALSE	FALSE	"Internal Gate" status
26.0	out	OT_ERR_B	BYTE	B#16#0	B#16#0	Operator error
28.0	out	LATCH_LOAD	DWORD	DW#16#0	DW#16#0	Current load value or latch value
32.0	out	ACT_CNTV	DWORD	DW#16#0	DW#16#0	Current count value
36.0	in_out	L_DIRECT	BOOL	FALSE	FALSE	1=Load new counter value
36.1	in_out	L_PREPAR	BOOL	FALSE	FALSE	1=Prepare new counter value
36.2	in_out	T_CMP_V1	BOOL	FALSE	FALSE	1=Load new compare value 1
36.3	in_out	T_CMP_V2	BOOL	FALSE	FALSE	1=Load new compare value 2
36.4	in_out	C_DOPARA	BOOL	FALSE	FALSE	1=Triggering parameter change
36.5	in_out	RES_SYNC	BOOL	FALSE	FALSE	1=Reset status bit "synchronization"
36.6	in_out	RES_ZERO	BOOL	FALSE	FALSE	1=Reset status bit "zero-crossing"
36.7	in_out	DIAG_INF	BOOL	FALSE	FALSE	1=Read diagnostics (only for OB82)
38.0	stat	FP_L_DIRECT	BOOL	FALSE	FALSE	1=Load new counter value
38.1	stat	FP_L_PREPAR	BOOL	FALSE	FALSE	1=Prepare new counter value
38.2	stat	FP_T_CMP_V1	BOOL	FALSE	FALSE	1=Load new compare value 1
38.3	stat	FP_T_CMP_V2	BOOL	FALSE	FALSE	1=Load new compare value 2
38.4	stat	FP_C_DOPARA	BOOL	FALSE	FALSE	1=Triggering parameter change
38.5	stat	FP_RES_SYNC	BOOL	FALSE	FALSE	1=Reset status bit "synchronization"
38.6	stat	FP_RES_ZERO	BOOL	FALSE	FALSE	1=Reset status bit "zero-crossing"
40.0	stat	SFC_ERR	INT	0	0	SFC51 error status

UDT1 - <offline>

"UDT_M1" Data type for stepper channel (Motor 1)

Name: UDT_M1 **Family:**
Author: SIEMENS **Version:** 0.0
Time stamp Code: 07/18/2012 04:55:13 PM **Block version:** 2
Interface: 07/04/2000 03:52:02 PM
Lengths (block/logic/data): 00000 00000 00000
UNLINKED

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	MOD_ADR	INT	0	Module address
+2.0	CH_NO	INT	0	Channel number
+4.0	CH_ADR	DWORD	DW#16#0	Channel address
+8.0	DS_OFFS	INT	0	data block offset
+10.0	b_10	WORD	W#16#0	
+12.0	b_12	BYTE	B#16#0	
+13.0	b_13_0	BOOL	FALSE	
+13.1	b_13_1	BOOL	FALSE	
+13.2	b_13_2	BOOL	FALSE	
+13.3	b_13_3	BOOL	FALSE	
+13.4	b_13_4	BOOL	FALSE	
+13.5	b_13_5	BOOL	FALSE	
+13.6	MODE_BUSY	BOOL	FALSE	Mode busy
+13.7	POS_REACHED	BOOL	FALSE	Position reached
+14.0	b_14_0	BOOL	FALSE	
+14.1	TEST_EN	BOOL	FALSE	test enable
+14.2	b_14_2	BOOL	FALSE	
+14.3	OT_ERR_A	BOOL	FALSE	quit error
+14.4	b_14_4	BOOL	FALSE	
+14.5	b_14_5	BOOL	FALSE	
+14.6	b_14_6	BOOL	FALSE	
+14.7	b_14_7	BOOL	FALSE	
+15.0	START	BOOL	FALSE	start
+15.1	STOP	BOOL	FALSE	stop
+15.2	DIR_M	BOOL	FALSE	direction minus
+15.3	DIR_P	BOOL	FALSE	direction plus
+15.4	ACK_MF	BOOL	FALSE	acknowledge M function
+15.5	READ_EN	BOOL	FALSE	read enable

Address	Name	Type	Initial value	Comment
+15.6	SKIP_BLK	BOOL	FALSE	skip blank
+15.7	DRV_EN	BOOL	FALSE	drive enable
+16.0	MODE_IN	BYTE	B#16#0	mode
+17.0	MODE_TYPE	BYTE	B#16#0	mode parameter
+18.0	OVERRIDE	BYTE	B#16#0	override
+19.0	b_19	BYTE	B#16#0	
+20.0	b_20	WORD	W#16#0	
+22.0	b_22_0	BOOL	FALSE	
+22.1	TST_STAT	BOOL	FALSE	test state
+22.2	b_22_2	BOOL	FALSE	
+22.3	OT_ERR	BOOL	FALSE	ot-error
+22.4	DATA_ERR	BOOL	FALSE	data-error
+22.5	b_22_5	BOOL	FALSE	
+22.6	b_22_6	BOOL	FALSE	
+22.7	PARA	BOOL	FALSE	parameterized
+23.0	ST_ENBLD	BOOL	FALSE	start enable
+23.1	WORKING	BOOL	FALSE	working
+23.2	WAIT_EI	BOOL	FALSE	wait for external enable
+23.3	b_23_3	BOOL	FALSE	
+23.4	b_23_4	BOOL	FALSE	
+23.5	DT_RUN	BOOL	FALSE	dwelt time running
+23.6	PR_BACK	BOOL	FALSE	program run backward
+23.7	b_23_7	BOOL	FALSE	
+24.0	MODE_OUT	BYTE	B#16#0	mode
+25.0	SYNC	BOOL	FALSE	synchronized
+25.1	MSR_DONE	BOOL	FALSE	measurement done
+25.2	GO_M	BOOL	FALSE	go_minus
+25.3	GO_P	BOOL	FALSE	go_plus
+25.4	ST_SERVO	BOOL	FALSE	status servo
+25.5	FVAL_DONE	BOOL	FALSE	flying actual value done
+25.6	b_25_6	BOOL	FALSE	
+25.7	POS_RCD	BOOL	FALSE	position is reached, hold
+26.0	NUM_MF	BYTE	B#16#0	number M function
+27.0	b_27_0	BOOL	FALSE	
+27.1	b_27_1	BOOL	FALSE	
+27.2	b_27_2	BOOL	FALSE	
+27.3	b_27_3	BOOL	FALSE	
+27.4	STR_MF	BOOL	FALSE	strobe signal for M functions
+27.5	b_27_5	BOOL	FALSE	
+27.6	b_27_6	BOOL	FALSE	
+27.7	b_27_7	BOOL	FALSE	

Page 2 of 16

Address	Name	Type	Initial value	Comment
+28.0	b_28	DWORD	DW#16#0	
+32.0	b_32	WORD	W#16#0	
+34.0	SERVO_EN	BOOL	FALSE	servo enable
+34.1	GAUG_FLY	BOOL	FALSE	gauging on the fly
+34.2	b_34_2	BOOL	FALSE	
+34.3	b_34_3	BOOL	FALSE	
+34.4	b_34_4	BOOL	FALSE	
+34.5	TRAV_MON	BOOL	FALSE	traverse monitoring
+34.6	PARK_AX	BOOL	FALSE	parking axis
+34.7	SIM_ON	BOOL	FALSE	simulation on
+35.0	b_35_0	BOOL	FALSE	
+35.1	b_35_1	BOOL	FALSE	
+35.2	MSR_EN	BOOL	FALSE	measurement
+35.3	REF_TRIG	BOOL	FALSE	trigger reference point
+35.4	DI_OFF	BOOL	FALSE	disable enable input
+35.5	FOLLOWUP	BOOL	FALSE	follow-up
+35.6	SSW_DIS	BOOL	FALSE	software switch disable
+35.7	DRIFT_OFF	BOOL	FALSE	automatic drift compensation off
+36.0	b_36	BYTE	B#16#0	
+37.0	MD_EN	BOOL	FALSE	machine data enable
+37.1	DELDIST_EN	BOOL	FALSE	delete distance to go
+37.2	SEARCH_F	BOOL	FALSE	automatic block search forward
+37.3	SEARCH_B	BOOL	FALSE	automatic block search backward
+37.4	b_37_4	BOOL	FALSE	
+37.5	RESET_AX	BOOL	FALSE	reset axis
+37.6	AVALREM_EN	BOOL	FALSE	remove setting actual value
+37.7	b_37_7	BOOL	FALSE	
+38.0	VLEV_EN	BOOL	FALSE	velocity levels 1 and 2
+38.1	CLEV_EN	BOOL	FALSE	control levels 1 and 2
+38.2	TRG254_EN	BOOL	FALSE	target 254
+38.3	MDI_EN	BOOL	FALSE	MDI movement block
+38.4	MDIFLY_EN	BOOL	FALSE	MDI block on the fly
+38.5	b_38_5	BOOL	FALSE	
+38.6	REFPT_EN	BOOL	FALSE	reference coordinate
+38.7	AVAL_EN	BOOL	FALSE	actual value coordinate
+39.0	FVAL_EN	BOOL	FALSE	actual value coordinate (flying)
+39.1	ZOFF_EN	BOOL	FALSE	zero offset value
+39.2	b_39_2	BOOL	FALSE	
+39.3	PARCH_EN	BOOL	FALSE	parameter change
+39.4	DIGO_EN	BOOL	FALSE	digital 0
+39.5	PROGS_EN	BOOL	FALSE	part program selection

(Anexo M) 134

Page 3 of 16

Address	Name	Type	Initial value	Comment
+39.6	REQAPP_EN	BOOL	FALSE	request application data
+39.7	TEACHIN_EN	BOOL	FALSE	Teach In
+40.0	AXCOU_EN	BOOL	FALSE	write job axis coupling FM453
+40.1	b_40_1	BOOL	FALSE	
+40.2	b_40_2	BOOL	FALSE	
+40.3	b_40_3	BOOL	FALSE	
+40.4	b_40_4	BOOL	FALSE	
+40.5	b_40_5	BOOL	FALSE	
+40.6	b_40_6	BOOL	FALSE	
+40.7	b_40_7	BOOL	FALSE	
+41.0	b_41	BYTE	B#16#0	
+42.0	OPDAT_EN	BOOL	FALSE	operating_data
+42.1	ACT_BL_EN	BOOL	FALSE	actual part program block
+42.2	NXT_BL_EN	BOOL	FALSE	next part program block
+42.3	BLEXT_EN	BOOL	FALSE	actual value at external block change
+42.4	SERVDAT_EN	BOOL	FALSE	service data
+42.5	OC_ERR_EN	BOOL	FALSE	oc-error
+42.6	b_42_6	BOOL	FALSE	
+42.7	b_42_7	BOOL	FALSE	
+43.0	AXCOURD_EN	BOOL	FALSE	read job axis coupling FM453
+43.1	b_43_1	BOOL	FALSE	
+43.2	b_43_2	BOOL	FALSE	
+43.3	PARRD_EN	BOOL	FALSE	parameter data
+43.4	DIGIO_EN	BOOL	FALSE	digital in_out
+43.5	OPDAT1_EN	BOOL	FALSE	operating data 1
+43.6	APPDAT_EN	BOOL	FALSE	applicating data
+43.7	MSRRD_EN	BOOL	FALSE	read measurement values
+44.0	SERVO_D	BOOL	FALSE	servo enable
+44.1	GAUG_FLY_D	BOOL	FALSE	gauging on the fly
+44.2	b_44_2	BOOL	FALSE	
+44.3	b_44_3	BOOL	FALSE	
+44.4	b_44_4	BOOL	FALSE	
+44.5	TRAV_MON_D	BOOL	FALSE	traverse monitoring
+44.6	PARK_AX_D	BOOL	FALSE	parking axis
+44.7	SIM_ON_D	BOOL	FALSE	simulation on
+45.0	b_45_0	BOOL	FALSE	
+45.1	b_45_1	BOOL	FALSE	
+45.2	MSR_D	BOOL	FALSE	measurement
+45.3	REF_TRIG_D	BOOL	FALSE	trigger reference point
+45.4	DI_OFF_D	BOOL	FALSE	disable enable input
+45.5	FOLLOWUP_D	BOOL	FALSE	follow-up

Page 4 of 16

Address	Name	Type	Initial value	Comment
+45.6	SSW_DIS_D	BOOL	FALSE	software switch disable
+45.7	DRIFT_OFF_D	BOOL	FALSE	automatic drift compensation off
+46.0	b_46	BYTE	B#16#0	
+47.0	MD_D	BOOL	FALSE	machine data enable
+47.1	DELDIST_D	BOOL	FALSE	delete distance to go
+47.2	SEARCH_F_D	BOOL	FALSE	automatic block search forward
+47.3	SEARCH_B_D	BOOL	FALSE	automatic block search backward
+47.4	b_47_4	BOOL	FALSE	
+47.5	RESET_AX_D	BOOL	FALSE	reset axis
+47.6	AVALREM_D	BOOL	FALSE	remove setting actual value
+47.7	b_47_7	BOOL	FALSE	
+48.0	VLEV_D	BOOL	FALSE	velocity levels 1 and 2
+48.1	CLEV_D	BOOL	FALSE	control levels 1 and 2
+48.2	TRG254_D	BOOL	FALSE	target 254
+48.3	MDI_D	BOOL	FALSE	MDI movement block
+48.4	MDIFLY_D	BOOL	FALSE	MDI block on the fly
+48.5	b_48_5	BOOL	FALSE	
+48.6	REFPT_D	BOOL	FALSE	reference coordinate
+48.7	AVAL_D	BOOL	FALSE	actual value coordinate
+49.0	FVAL_D	BOOL	FALSE	actual value coordinate (flying)
+49.1	ZOFF_D	BOOL	FALSE	zero offset value
+49.2	b_49_2	BOOL	FALSE	
+49.3	PARCH_D	BOOL	FALSE	parameter change
+49.4	DIGO_D	BOOL	FALSE	digital 0
+49.5	PROGS_D	BOOL	FALSE	part program selection
+49.6	REQAPP_D	BOOL	FALSE	request application data
+49.7	TEACHIN_D	BOOL	FALSE	Teach In
+50.0	AXCOU_D	BOOL	FALSE	axis coupling FM453
+50.1	b_50_1	BOOL	FALSE	
+50.2	b_50_2	BOOL	FALSE	
+50.3	b_50_3	BOOL	FALSE	
+50.4	b_50_4	BOOL	FALSE	
+50.5	b_50_5	BOOL	FALSE	
+50.6	b_50_6	BOOL	FALSE	
+50.7	b_50_7	BOOL	FALSE	
+51.0	b_51	BYTE	B#16#0	
+52.0	OPDAT_D	BOOL	FALSE	operating_data
+52.1	ACT_BL_D	BOOL	FALSE	actual part program block
+52.2	NXT_BL_D	BOOL	FALSE	next part program block
+52.3	BLEXT_D	BOOL	FALSE	actual value at external block change
+52.4	SERVDAT_D	BOOL	FALSE	service data

(Anexo M) 135

Page 5 of 16

Address	Name	Type	Initial value	Comment
+52.5	OC_ERR_D	BOOL	FALSE	oc_error
+52.6	OT_ERR_D	BOOL	FALSE	ot_error
+52.7	DA_ERR_D	BOOL	FALSE	data error
+53.0	AXCOURD_D	BOOL	FALSE	axis coupling FM453
+53.1	b_53_1	BOOL	FALSE	
+53.2	b_53_2	BOOL	FALSE	
+53.3	PARRD_D	BOOL	FALSE	parameter data
+53.4	DIGIO_D	BOOL	FALSE	digital in_out
+53.5	OPDAT1_D	BOOL	FALSE	operating data 1
+53.6	APPDAT_D	BOOL	FALSE	applicating data
+53.7	MSRRD_D	BOOL	FALSE	read measurement values
+54.0	SERVO_ERR	BOOL	FALSE	servo enable
+54.1	GAUG_FLY_ERR	BOOL	FALSE	gauging on the fly
+54.2	b_54_2	BOOL	FALSE	
+54.3	b_54_3	BOOL	FALSE	
+54.4	b_54_4	BOOL	FALSE	
+54.5	TRAV_MON_ERR	BOOL	FALSE	traverse monitoring
+54.6	PARK_AX_ERR	BOOL	FALSE	parking axis
+54.7	SIM_ON_ERR	BOOL	FALSE	simulation on
+55.0	b_55_0	BOOL	FALSE	
+55.1	b_55_1	BOOL	FALSE	
+55.2	MSR_ERR	BOOL	FALSE	measurement
+55.3	REF_TRIG_ERR	BOOL	FALSE	trigger reference point
+55.4	DI_OFF_ERR	BOOL	FALSE	disable enable input
+55.5	FOLLOWUP_ERR	BOOL	FALSE	follow-up
+55.6	SSW_DIS_ERR	BOOL	FALSE	software switch disable
+55.7	DRIFT_OFF_ERR	BOOL	FALSE	automatic drift compensation off
+56.0	b_56	BYTE	B#16#0	
+57.0	MD_ERR	BOOL	FALSE	machine data enable
+57.1	DELDIST_ERR	BOOL	FALSE	delete distance to go
+57.2	SEARCH_F_ERR	BOOL	FALSE	automatic block search forward
+57.3	SEARCH_B_ERR	BOOL	FALSE	automatic block search backward
+57.4	b_57_4	BOOL	FALSE	
+57.5	RESET_AX_ERR	BOOL	FALSE	reset axis
+57.6	AVALREM_ERR	BOOL	FALSE	remove setting actual value
+57.7	b_57_7	BOOL	FALSE	
+58.0	VLEV_ERR	BOOL	FALSE	velocity levels 1 and 2
+58.1	CLEV_ERR	BOOL	FALSE	control levels 1 and 2
+58.2	TRG254_ERR	BOOL	FALSE	target 254
+58.3	MDI_ERR	BOOL	FALSE	MDI movement block
+58.4	MDIFLY_ERR	BOOL	FALSE	MDI block on the fly

Page 6 of 16

Address	Name	Type	Initial value	Comment
+58.5	b_58_5	BOOL	FALSE	
+58.6	REFPT_ERR	BOOL	FALSE	reference coordinate
+58.7	AVAL_ERR	BOOL	FALSE	actual value coordinate
+59.0	FVAL_ERR	BOOL	FALSE	actual value coordinate (flying)
+59.1	ZOFF_ERR	BOOL	FALSE	zero offset value
+59.2	b_59_2	BOOL	FALSE	
+59.3	PARCH_ERR	BOOL	FALSE	parameter change
+59.4	DIGO_ERR	BOOL	FALSE	digital 0
+59.5	PROGS_ERR	BOOL	FALSE	part program selection
+59.6	REQAPP_ERR	BOOL	FALSE	request application data
+59.7	TEACHIN_ERR	BOOL	FALSE	Teach In
+60.0	AXCOU_ERR	BOOL	FALSE	axis coupling write error FM453
+60.1	b_60_1	BOOL	FALSE	
+60.2	b_60_2	BOOL	FALSE	
+60.3	b_60_3	BOOL	FALSE	
+60.4	b_60_4	BOOL	FALSE	
+60.5	b_60_5	BOOL	FALSE	
+60.6	b_60_6	BOOL	FALSE	
+60.7	b_60_7	BOOL	FALSE	
+61.0	b_61	BYTE	B#16#0	
+62.0	OPDAT_ERR	BOOL	FALSE	operating_data
+62.1	ACT_BL_ERR	BOOL	FALSE	actual part program block
+62.2	NXT_BL_ERR	BOOL	FALSE	next part program block
+62.3	BLEXT_ERR	BOOL	FALSE	actual value at external block change
+62.4	SERVDAT_ERR	BOOL	FALSE	service data
+62.5	OC_ERR_ERR	BOOL	FALSE	oc_error
+62.6	OT_ERR_ERR	BOOL	FALSE	ot_error
+62.7	DA_ERR_ERR	BOOL	FALSE	data error
+63.0	AXCOURD_ERR	BOOL	FALSE	axis coupling read error FM453
+63.1	b_63_1	BOOL	FALSE	
+63.2	b_63_2	BOOL	FALSE	
+63.3	PARRD_ERR	BOOL	FALSE	parameter data
+63.4	DIGIO_ERR	BOOL	FALSE	digital in_out
+63.5	OPDAT1_ERR	BOOL	FALSE	operating data 1
+63.6	APPDAT_ERR	BOOL	FALSE	applicating data
+63.7	MSRRD_ERR	BOOL	FALSE	read measurement values
+64.0	b_64	WORD	W#16#0	
+66.0	JOB_ERR	INT	0	error-code SFC58/59 (FC POS_CTRL)
+68.0	JOBBUSY_WR	BOOL	FALSE	job aktiv write
+68.1	IMPO_WR	BOOL	FALSE	imposs write
+68.2	JOBBUSY_RD	BOOL	FALSE	job aktiv read

(Anexo M) 136

Page 7 of 16

Address	Name	Type	Initial value	Comment
+68.3	IMPO_RD	BOOL	FALSE	imposs read
+68.4	b_68_4	BOOL	FALSE	
+68.5	b_68_5	BOOL	FALSE	
+68.6	b_68_6	BOOL	FALSE	
+68.7	b_68_7	BOOL	FALSE	
+69.0	b_69_0	BOOL	FALSE	
+69.1	JOBRESET	BOOL	FALSE	Reset_ERR and _D
+69.2	b_69_2	BOOL	FALSE	
+69.3	b_69_3	BOOL	FALSE	
+69.4	b_69_4	BOOL	FALSE	
+69.5	b_69_5	BOOL	FALSE	
+69.6	b_69_6	BOOL	FALSE	
+69.7	b_69_7	BOOL	FALSE	
+70.0	MDL_DEFECT	BOOL	FALSE	Module defective
+70.1	INT_FAULT	BOOL	FALSE	Internal fault
+70.2	EXT_FAULT	BOOL	FALSE	External fault
+70.3	PNT_INFO	BOOL	FALSE	Point information
+70.4	b_70_4	BOOL	FALSE	
+70.5	FLD_CONNCTR	BOOL	FALSE	Field wiring connector missing
+70.6	NO_CONFIG	BOOL	FALSE	Module has no configuration data
+70.7	b_70_7	BOOL	FALSE	
+71.0	MDL_TYPE	BYTE	B#16#0	Type of module
+72.0	b_72_0	BOOL	FALSE	
+72.1	COMM_FAULT	BOOL	FALSE	Communication fault
+72.2	b_72_2	BOOL	FALSE	
+72.3	WTCH_DOG_FLT	BOOL	FALSE	Watch dog timer stopped module
+72.4	INT_PS_FLT	BOOL	FALSE	Internal power supply fault
+72.5	b_72_5	BOOL	FALSE	
+72.6	b_72_6	BOOL	FALSE	
+72.7	b_72_7	BOOL	FALSE	
+73.0	b_73_0	BOOL	FALSE	
+73.1	b_73_1	BOOL	FALSE	
+73.2	EPROM_FLT	BOOL	FALSE	EPROM fault
+73.3	RAM_FLT	BOOL	FALSE	RAM fault
+73.4	b_73_4	BOOL	FALSE	
+73.5	b_73_5	BOOL	FALSE	
+73.6	HW_INTR_FLT	BOOL	FALSE	Hardware interrupt input in fault
+73.7	b_73_7	BOOL	FALSE	
+74.0	POS_ID	BYTE	B#16#0	channel type info
+75.0	LEN_INFO	BYTE	B#16#0	channel info length
+76.0	CHEN_NO	BYTE	B#16#0	number of channels

Address	Name	Type	Initial value	Comment
+77.0	CH_ERR_VE1	BOOL	FALSE	channel error vector 1
+77.1	CH_ERR_VE2	BOOL	FALSE	channel error vector 2
+77.2	CH_ERR_VE3	BOOL	FALSE	channel error vector 3
+77.3	b_77_3	BOOL	FALSE	
+77.4	b_77_4	BOOL	FALSE	
+77.5	b_77_5	BOOL	FALSE	
+77.6	b_77_6	BOOL	FALSE	
+77.7	b_77_7	BOOL	FALSE	
+78.0	CAB_BR1	BOOL	FALSE	channel 1 cablebraek
+78.1	ERR_ABE1	BOOL	FALSE	channel 1 error sensor_absolut
+78.2	ERR_PU1	BOOL	FALSE	channel 1 error impulse
+78.3	VO_ENC1	BOOL	FALSE	channel 1 voltage control sensor
+78.4	VO_15_1	BOOL	FALSE	channel 1 voltage control 15 V
+78.5	VO_DIO1	BOOL	FALSE	channel 1 voltage control digital output
+78.6	b_78_6	BOOL	FALSE	
+78.7	OC_ERR_EN1	BOOL	FALSE	channel 1 operation error en1
+79.0	b_79	BYTE	B#16#0	
+80.0	CAB_BR2	BOOL	FALSE	channel 2 cablebraek
+80.1	ERR_ABE2	BOOL	FALSE	channel 2 error sensor_absolut
+80.2	ERR_PU2	BOOL	FALSE	channel 2 error impulse
+80.3	VO_ENC2	BOOL	FALSE	channel 2 voltage control sensor
+80.4	VO_15_2	BOOL	FALSE	channel 2 voltage control 15 V
+80.5	VO_DIO2	BOOL	FALSE	channel 2 voltage control digital output
+80.6	b_80_6	BOOL	FALSE	
+80.7	OC_ERR_EN2	BOOL	FALSE	channel 2 operation error en2
+81.0	b_81	BYTE	B#16#0	
+82.0	CAB_BR3	BOOL	FALSE	channel 3 cablebraek
+82.1	ERR_ABE3	BOOL	FALSE	channel 3 error sensor_absolut
+82.2	ERR_PU3	BOOL	FALSE	channel 3 error impulse
+82.3	VO_ENC3	BOOL	FALSE	channel 3 voltage control sensor
+82.4	VO_15_3	BOOL	FALSE	channel 3 voltage control 15 V
+82.5	VO_DIO3	BOOL	FALSE	channel 3 voltage control digital output
+82.6	b_82_6	BOOL	FALSE	
+82.7	OC_ERR_EN3	BOOL	FALSE	channel 3 operation error en3
+83.0	b_83	BYTE	B#16#0	
+84.0	b_84	WORD	W#16#0	
+86.0	OC_ERR_NO	BYTE	B#16#0	operation error, detail event number
+87.0	OC_REE_CL	BYTE	B#16#0	detail event class
+88.0	b_88	WORD	W#16#0	
+90.0	OT_ERR_NO	BYTE	B#16#0	control/process error, detail event number
+91.0	OT_ERR_CL	BYTE	B#16#0	detail event class

Address	Name	Type	Initial value	Comment
+92.0	b_92	WORD	W#16#0	
+94.0	DA_ERR_NO	BYTE	B#16#0	data error, detail event number
+95.0	DA_ERR_CL	BYTE	B#16#0	detail event class
+96.0	DIAG_ERR	INT	0	error-code SFC51 (FC POS_DIAG)
+98.0	MSRM_ERR	INT	0	error-code SFC59 (FC POS_MSRM)
+100.0	feld1_intern	ARRAY[100..139]		
*1.0		BYTE		
+140.0	ZOFF	DINT	L#0	zero offset value
+144.0	AVAL	DINT	L#0	actual value coordinate
+148.0	FVAL	DINT	L#0	actual value coordinate (flying)
+152.0	REFPT	DINT	L#0	reference coordinate
+156.0	TRG254	DWORD	DW#16#0	target 254
+160.0	VLEVEL_1	DWORD	DW#16#0	velocity level 1
+164.0	VLEVEL_2	DWORD	DW#16#0	velocity level 2
+168.0	CLEVEL_1	DWORD	DW#16#0	control level 1
+172.0	CLEVEL_2	DWORD	DW#16#0	control level 2
+176.0	MDIB	STRUCT		MDI movement block
+0.0	b_176	BYTE	B#16#0	
+1.0	b_177	BYTE	B#16#0	
+2.0	G_1_EN	BOOL	FALSE	enable bit for G function group 1
+2.1	G_2_EN	BOOL	FALSE	enable bit for G function group 2
+2.2	b_178_2	BOOL	FALSE	
+2.3	b_178_3	BOOL	FALSE	
+2.4	X_T_EN	BOOL	FALSE	enable bit for position / dwell time
+2.5	b_178_5	BOOL	FALSE	
+2.6	b_178_6	BOOL	FALSE	
+2.7	b_178_7	BOOL	FALSE	
+3.0	V_EN	BOOL	FALSE	enable bit for velocity
+3.1	M_1_EN	BOOL	FALSE	enable bit for M function group 1
+3.2	M_2_EN	BOOL	FALSE	enable bit for M function group 2
+3.3	M_3_EN	BOOL	FALSE	enable bit for M function group 3
+3.4	b_179_4	BOOL	FALSE	
+3.5	b_179_5	BOOL	FALSE	
+3.6	b_179_6	BOOL	FALSE	
+3.7	b_179_7	BOOL	FALSE	
+4.0	G_1_VAL	BYTE	B#16#0	value G function group 1
+5.0	G_2_VAL	BYTE	B#16#0	value G function group 2
+6.0	b_182	BYTE	B#16#0	
+7.0	b_183	BYTE	B#16#0	
+8.0	X_T_VAL	DINT	L#0	value position / dwell time
+12.0	V_VAL	DINT	L#0	value velocity

Page 10 of 16

Address	Name	Type	Initial value	Comment
+16.0	M_1_VAL	BYTE	B#16#0	value M function group 1
+17.0	M_2_VAL	BYTE	B#16#0	value M function group 2
+18.0	M_3_VAL	BYTE	B#16#0	value M function group 3
+19.0	b_195	BYTE	B#16#0	
=20.0		END_STRUCT		
+196.0	PAR_CHAN	STRUCT		parameter change
+0.0	TYP	BYTE	B#16#0	DB type
+1.0	NUMB	BYTE	B#16#0	data number
+2.0	COUN	BYTE	B#16#0	number of data
+3.0	JOB	BYTE	B#16#0	job type
+4.0	DATA	ARRAY[200..219]		
*1.0		BYTE		
=24.0		END_STRUCT		
+220.0	D_IN0	BOOL	FALSE	digital input 0
+220.1	D_IN1	BOOL	FALSE	digital input 1
+220.2	D_IN2	BOOL	FALSE	digital input 2
+220.3	D_IN3	BOOL	FALSE	digital input 3
+220.4	b_220_4	BOOL	FALSE	
+220.5	b_220_5	BOOL	FALSE	
+220.6	b_220_6	BOOL	FALSE	
+220.7	b_220_7	BOOL	FALSE	
+221.0	D_OUT0	BOOL	FALSE	digital output 0
+221.1	D_OUT1	BOOL	FALSE	digital output 1
+221.2	D_OUT2	BOOL	FALSE	digital output 2
+221.3	D_OUT3	BOOL	FALSE	digital output 3
+221.4	b_221_4	BOOL	FALSE	
+221.5	b_221_5	BOOL	FALSE	
+221.6	b_221_6	BOOL	FALSE	
+221.7	b_221_7	BOOL	FALSE	
+222.0	MDI_F	STRUCT		MDI block on the fly
+0.0	b_222	BYTE	B#16#0	
+1.0	b_223	BYTE	B#16#0	
+2.0	G_1_EN	BOOL	FALSE	enable bit for G function group 1
+2.1	G_2_EN	BOOL	FALSE	enable bit for G function group 2
+2.2	b_224_2	BOOL	FALSE	
+2.3	b_224_3	BOOL	FALSE	
+2.4	X_T_EN	BOOL	FALSE	enable bit for position / dwell time
+2.5	b_224_5	BOOL	FALSE	
+2.6	b_224_6	BOOL	FALSE	
+2.7	b_224_7	BOOL	FALSE	
+3.0	V_EN	BOOL	FALSE	enable bit for velocity

(Anexo M) 138

Page 11 of 16

Address	Name	Type	Initial value	Comment
+3.1	M_1_EN	BOOL	FALSE	enable bit for M function group 1
+3.2	M_2_EN	BOOL	FALSE	enable bit for M function group 2
+3.3	M_3_EN	BOOL	FALSE	enable bit for M function group 3
+3.4	b_225_4	BOOL	FALSE	
+3.5	b_225_5	BOOL	FALSE	
+3.6	b_225_6	BOOL	FALSE	
+3.7	b_225_7	BOOL	FALSE	
+4.0	G_1_VAL	BYTE	B#16#0	value G function group 1
+5.0	G_2_VAL	BYTE	B#16#0	value G function group 2
+6.0	b_228	BYTE	B#16#0	
+7.0	b_229	BYTE	B#16#0	
+8.0	X_T_VAL	DINT	L#0	value position / dwell time
+12.0	V_VAL	DINT	L#0	value velocity
+16.0	M_1_VAL	BYTE	B#16#0	value M function group 1
+17.0	M_2_VAL	BYTE	B#16#0	value M function group 2
+18.0	M_3_VAL	BYTE	B#16#0	value M function group 3
+19.0	b_241	BYTE	B#16#0	
=20.0		END_STRUCT		
+242.0	PROG_NO	BYTE	B#16#0	part program number
+243.0	BLCK_NO	BYTE	B#16#0	part program block number
+244.0	PROG_DIR	BYTE	B#16#0	program run direction
+245.0	b_245	BYTE	B#16#0	
+246.0	CODE_AP1	BYTE	B#16#0	code application 1
+247.0	CODE_AP2	BYTE	B#16#0	code application 2
+248.0	CODE_AP3	BYTE	B#16#0	code application 3
+249.0	CODE_AP4	BYTE	B#16#0	code application 4
+250.0	TEA_PRO_NO	BYTE	B#16#0	part program number
+251.0	TEA_BLK_NO	BYTE	B#16#0	part program block number
+252.0	AXCOU_DEF	BYTE	B#16#0	axis coupling locate DS24 FM453
+253.0	b_253	BYTE	B#16#0	
+254.0	feld2_intern	ARRAY[254..309]		
*1.0		BYTE		
+310.0	ACT_VAL	DINT	L#0	actual value
+314.0	SPEED	DWORD	DW#16#0	speed
+318.0	REM_DIST	DINT	L#0	remaining distance
+322.0	SET_POS	DINT	L#0	set position to be approached
+326.0	SUM_OFST	DINT	L#0	sum of tool offset and zero offset
+330.0	TRAV_SPE	DWORD	DW#16#0	traversing speed
+334.0	di_334	DINT	L#0	
+338.0	di_338	DINT	L#0	
+342.0	ACT_BL	STRUCT		actual part program block

Page 12 of 16

Address	Name	Type	Initial value	Comment
+0.0	PROG_NO	BYTE	B#16#0	part program number
+1.0	BLCK_NO	BYTE	B#16#0	part program block number
+2.0	G_1_EN	BOOL	FALSE	enable bit for G function group 1
+2.1	G_2_EN	BOOL	FALSE	enable bit for G function group 2
+2.2	G_3_EN	BOOL	FALSE	enable bit for G function group 3
+2.3	b_344_3	BOOL	FALSE	
+2.4	X_T_EN	BOOL	FALSE	enable bit for position / dwell time
+2.5	SR_L_EN	BOOL	FALSE	enable bit for number of subroutine loops
+2.6	SR_N_EN	BOOL	FALSE	enable bit for subroutine number
+2.7	SKIP_EN	BOOL	FALSE	enable bit for block skip
+3.0	V_EN	BOOL	FALSE	enable bit for velocity
+3.1	M_1_EN	BOOL	FALSE	enable bit for M function group 1
+3.2	M_2_EN	BOOL	FALSE	enable bit for M function group 2
+3.3	M_3_EN	BOOL	FALSE	enable bit for M function group 3
+3.4	TO_EN	BOOL	FALSE	enable bit for tool offset number
+3.5	b_345_5	BOOL	FALSE	
+3.6	b_345_6	BOOL	FALSE	
+3.7	b_345_7	BOOL	FALSE	
+4.0	G_1_VAL	BYTE	B#16#0	G function group 1
+5.0	G_2_VAL	BYTE	B#16#0	G function group 2
+6.0	G_3_VAL	BYTE	B#16#0	G function group 3
+7.0	b_349	BYTE	B#16#0	
+8.0	X_T_VAL	DINT	L#0	position / dwell time / subroutine number
+12.0	V_VAL	DINT	L#0	velocity / subroutine loops
+16.0	M_1_VAL	BYTE	B#16#0	M function group 1
+17.0	M_2_VAL	BYTE	B#16#0	M function group 2
+18.0	M_3_VAL	BYTE	B#16#0	M function group 3
+19.0	TO_VAL	BYTE	B#16#0	tool offset number
=20.0		END_STRUCT		
+362.0	NXT_BLK	STRUCT		next part program block
+0.0	PROG_NO	BYTE	B#16#0	part program number
+1.0	BLCK_NO	BYTE	B#16#0	part program block number
+2.0	G_1_EN	BOOL	FALSE	enable bit for G function group 1
+2.1	G_2_EN	BOOL	FALSE	enable bit for G function group 2
+2.2	G_3_EN	BOOL	FALSE	enable bit for G function group 3
+2.3	b_364_3	BOOL	FALSE	
+2.4	X_T_EN	BOOL	FALSE	enable bit for position / dwell time
+2.5	SR_L_EN	BOOL	FALSE	enable bit for number of subroutine loops
+2.6	SR_N_EN	BOOL	FALSE	enable bit for subroutine number
+2.7	SKIP_EN	BOOL	FALSE	enable bit for block skip
+3.0	V_EN	BOOL	FALSE	enable bit for velocity

(Anexo M) 139

Page 13 of 16

Address	Name	Type	Initial value	Comment
+3.1	M_1_EN	BOOL	FALSE	enable bit for M function group 1
+3.2	M_2_EN	BOOL	FALSE	enable bit for M function group 2
+3.3	M_3_EN	BOOL	FALSE	enable bit for M function group 3
+3.4	TO_EN	BOOL	FALSE	enable bit for tool offset number
+3.5	b_365_5	BOOL	FALSE	
+3.6	b_365_6	BOOL	FALSE	
+3.7	b_365_7	BOOL	FALSE	
+4.0	G_1_VAL	BYTE	B#16#0	G function group 1
+5.0	G_2_VAL	BYTE	B#16#0	G function group 2
+6.0	G_3_VAL	BYTE	B#16#0	G function group 3
+7.0	b_369	BYTE	B#16#0	
+8.0	X_T_VAL	DINT	L#0	position / dwell time / subroutine number
+12.0	V_VAL	DINT	L#0	velocity / subroutine loops
+16.0	M_1_VAL	BYTE	B#16#0	M function group 1
+17.0	M_2_VAL	BYTE	B#16#0	M function group 2
+18.0	M_3_VAL	BYTE	B#16#0	M function group 3
+19.0	TO_VAL	BYTE	B#16#0	tool offset number
=20.0		END_STRUCT		
+382.0	APP1	DINT	L#0	application data 1
+386.0	APP2	DINT	L#0	application data 2
+390.0	APP3	DINT	L#0	application data 3
+394.0	APP4	DINT	L#0	application data 4
+398.0	BLCK_EXT	DINT	L#0	actual value at external block change
+402.0	OUT_VAL	DINT	L#0	DAC value / frequency
+406.0	ENC_VAL	DINT	L#0	encoder actual value
+410.0	PULS_ERR	DINT	L#0	error pulses
+414.0	KV_FA	DINT	L#0	Kv factor
+418.0	FOLL_ERR	DINT	L#0	following error
+422.0	FERR_LIM	DINT	L#0	following error limit
+426.0	OSC_ERR	DINT	L#0	oscillation error
+430.0	DR_TIME	DINT	L#0	drive time
+434.0	OVERRIDE1	BYTE	B#16#0	override
+435.0	PROG_NO1	BYTE	B#16#0	part program number
+436.0	BLCK_NO1	BYTE	B#16#0	part program block number
+437.0	LOOP_NO1	BYTE	B#16#0	number of subroutine loops
+438.0	G90_91	BYTE	B#16#0	G90/91 active
+439.0	G60_64	BYTE	B#16#0	G60/64 active
+440.0	G43_44	BYTE	B#16#0	G43/44 active
+441.0	TO_NO	BYTE	B#16#0	tool offset number
+442.0	b_442_0	BOOL	FALSE	
+442.1	LIM_SP	BOOL	FALSE	limitation speed

Page 14 of 16

Address	Name	Type	Initial value	Comment
+442.2	LIM_10	BOOL	FALSE	limitation +/- 10V
+442.3	LIM_SU	BOOL	FALSE	limitation speed up
+442.4	b_442_4	BOOL	FALSE	
+442.5	b_442_5	BOOL	FALSE	
+442.6	b_442_6	BOOL	FALSE	
+442.7	b_442_7	BOOL	FALSE	
+443.0	LIM_FR	BOOL	FALSE	limitation frequency
+443.1	LIM_FV	BOOL	FALSE	limitation frequency value
+443.2	b_443_2	BOOL	FALSE	
+443.3	LIM_FS	BOOL	FALSE	limitation frequency speed-up
+443.4	b_443_4	BOOL	FALSE	
+443.5	b_443_5	BOOL	FALSE	
+443.6	b_443_6	BOOL	FALSE	
+443.7	b_443_7	BOOL	FALSE	
+444.0	b_444	BYTE	B#16#0	
+445.0	b_445	BYTE	B#16#0	
+446.0	PAR_RD	STRUCT		PAR_READ
+0.0	TYP1	BYTE	B#16#0	DB type
+1.0	NUMB	BYTE	B#16#0	data number
+2.0	COUN	BYTE	B#16#0	number of data
+3.0	JOB	BYTE	B#16#0	job type
+4.0	DATA1	ARRAY[450..469]		
*1.0		BYTE		
=24.0		END_STRUCT		
+470.0	AXCOU_STAT	BYTE	B#16#0	status axis coupling DS37 FM453
+471.0	b_471	BYTE	B#16#0	
+472.0	feld3_intern	ARRAY[472..485]		
*1.0		BYTE		
+486.0	BEGIN_VAL	DINT	L#0	begin value
+490.0	END_VAL	DINT	L#0	end value
+494.0	LENGTH_VAL	DWORD	DW#16#0	length value
+498.0	USR	STRUCT		user program function bits
+0.0	BITC_0	BOOL	FALSE	MD write request
+0.1	BITC_1	BOOL	FALSE	MD read request
+0.2	BITC_2	BOOL	FALSE	MDI block request
+0.3	BITC_3	BOOL	FALSE	part program selection request
+0.4	BITC_4	BOOL	FALSE	Teach In request
+0.5	BITC_5	BOOL	FALSE	incremental value request
+0.6	BITC_6	BOOL	FALSE	velocity levels request
+0.7	BITC_7	BOOL	FALSE	control levels request
+1.0	BITC_8	BOOL	FALSE	MDI block on the fly request

(Anexo M) 140

Page 15 of 16

Address	Name	Type	Initial value	Comment
+1.1	BITC_9	BOOL	FALSE	actual value on the fly request
+1.2	BITC_10	BOOL	FALSE	zero point offset request
+1.3	BITC_11	BOOL	FALSE	reserved
+1.4	BITC_12	BOOL	FALSE	reserved
+1.5	BITC_13	BOOL	FALSE	Alarm
+1.6	BITC_14	BOOL	FALSE	Data-Error
+1.7	BITC_15	BOOL	FALSE	OT-Error
+2.0	MD_NO	WORD	W#16#0	MD number
+4.0	MD_VALUE	DINT	L#0	MD value
+8.0	INC_NO	BYTE	B#16#0	incremental value number
+9.0	b_507	BYTE	B#16#0	
+10.0	PICT_NO	WORD	W#16#0	picture number
+12.0	KEY_CODE	WORD	W#16#0	keyboard code
+14.0	b_512	WORD	W#16#0	
+16.0	BITA_0	BOOL	FALSE	Control
+16.1	BITA_1	BOOL	FALSE	Reference point approach
+16.2	BITA_2	BOOL	FALSE	Increment
+16.3	BITA_3	BOOL	FALSE	MDI
+16.4	BITA_4	BOOL	FALSE	Automatic / single move
+16.5	BITA_5	BOOL	FALSE	Automatic / follow move
+16.6	BITA_6	BOOL	FALSE	Jog-Mode
+16.7	b_514_7	BOOL	FALSE	
+17.0	b_515_0	BOOL	FALSE	
+17.1	b_515_1	BOOL	FALSE	
+17.2	b_515_2	BOOL	FALSE	
+17.3	b_515_3	BOOL	FALSE	
+17.4	b_515_4	BOOL	FALSE	
+17.5	b_515_5	BOOL	FALSE	
+17.6	BITA_14	BOOL	FALSE	Quit Error
+17.7	BITA_15	BOOL	FALSE	Quit Alarm
=18.0		END_STRUCT		
=516.0		END_STRUCT		

UDT2 - <offline>

"UDT_M2" Data type for stepper channel (Motor 2)

Name: UDT_M2 **Family:**
Author: SIEMENS **Version:** 0.0
 Block version: 2
Time stamp Code: 07/18/2012 04:55:26 PM
 Interface: 07/04/2000 03:52:02 PM
Lengths (block/logic/data): 00000 00000 00000
UNLINKED

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	MOD_ADR_M2	INT	0	Module address
+2.0	CH_NO_M2	INT	0	Channel number
+4.0	CH_ADR_M2	DWORD	DW#16#0	Channel address
+8.0	DS_OFFS_M2	INT	0	data block offset
+10.0	b_10_M2	WORD	W#16#0	
+12.0	b_12_M2	BYTE	B#16#0	
+13.0	b_13_0_M2	BOOL	FALSE	
+13.1	b_13_1_M2	BOOL	FALSE	
+13.2	b_13_2_M2	BOOL	FALSE	
+13.3	b_13_3_M2	BOOL	FALSE	
+13.4	b_13_4_M2	BOOL	FALSE	
+13.5	b_13_5_M2	BOOL	FALSE	
+13.6	MODE_BUSY_M2	BOOL	FALSE	Mode busy
+13.7	POS_REACHED_M2	BOOL	FALSE	Position reached
+14.0	b_14_0_M2	BOOL	FALSE	
+14.1	TEST_EN_M2	BOOL	FALSE	test enable
+14.2	b_14_2_M2	BOOL	FALSE	
+14.3	OT_ERR_A_M2	BOOL	FALSE	quit error
+14.4	b_14_4_M2	BOOL	FALSE	
+14.5	b_14_5_M2	BOOL	FALSE	
+14.6	b_14_6_M2	BOOL	FALSE	
+14.7	b_14_7_M2	BOOL	FALSE	
+15.0	START_M2	BOOL	FALSE	start
+15.1	STOP_M2	BOOL	FALSE	stop
+15.2	DIR_M_M2	BOOL	FALSE	direction minus
+15.3	DIR_P_M2	BOOL	FALSE	direction plus
+15.4	ACK_MF_M2	BOOL	FALSE	acknowledge M function
+15.5	READ_EN_M2	BOOL	FALSE	read enable

Address	Name	Type	Initial value	Comment
+15.6	SKIP_BLK_M2	BOOL	FALSE	skip blank
+15.7	DRV_EN_M2	BOOL	FALSE	drive enable
+16.0	MODE_IN_M2	BYTE	B#16#0	mode
+17.0	MODE_TYPE_M2	BYTE	B#16#0	mode parameter
+18.0	OVERRIDE_M2	BYTE	B#16#0	override
+19.0	b_19_M2	BYTE	B#16#0	
+20.0	b_20_M2	WORD	W#16#0	
+22.0	b_22_0_M2	BOOL	FALSE	
+22.1	TST_STAT_M2	BOOL	FALSE	test state
+22.2	b_22_2_M2	BOOL	FALSE	
+22.3	OT_ERR_M2	BOOL	FALSE	ot-error
+22.4	DATA_ERR_M2	BOOL	FALSE	data-error
+22.5	b_22_5_M2	BOOL	FALSE	
+22.6	b_22_6_M2	BOOL	FALSE	
+22.7	PARA_M2	BOOL	FALSE	parameterized
+23.0	ST_ENBLD_M2	BOOL	FALSE	start enable
+23.1	WORKING_M2	BOOL	FALSE	working
+23.2	WAIT_EI_M2	BOOL	FALSE	wait for external enable
+23.3	b_23_3_M2	BOOL	FALSE	
+23.4	b_23_4_M2	BOOL	FALSE	
+23.5	DT_RUN_M2	BOOL	FALSE	dwelt time running
+23.6	PR_BACK_M2	BOOL	FALSE	program run backward
+23.7	b_23_7_M2	BOOL	FALSE	
+24.0	MODE_OUT_M2	BYTE	B#16#0	mode
+25.0	SYNC_M2	BOOL	FALSE	synchronized
+25.1	MSR_DONE_M2	BOOL	FALSE	measurement done
+25.2	GO_M_M2	BOOL	FALSE	go_minus
+25.3	GO_P_M2	BOOL	FALSE	go_plus
+25.4	ST_SERVO_M2	BOOL	FALSE	status servo
+25.5	FVAL_DONE_M2	BOOL	FALSE	flying actual value done
+25.6	b_25_6_M2	BOOL	FALSE	
+25.7	POS_RCD_M2	BOOL	FALSE	position is reached, hold
+26.0	NUM_MF_M2	BYTE	B#16#0	number M function
+27.0	b_27_0_M2	BOOL	FALSE	
+27.1	b_27_1_M2	BOOL	FALSE	
+27.2	b_27_2_M2	BOOL	FALSE	
+27.3	b_27_3_M2	BOOL	FALSE	
+27.4	STR_MF_M2	BOOL	FALSE	strobe signal for M functions
+27.5	b_27_5_M2	BOOL	FALSE	
+27.6	b_27_6_M2	BOOL	FALSE	
+27.7	b_27_7_M2	BOOL	FALSE	

Page 2 of 16

Address	Name	Type	Initial value	Comment
+28.0	b_28_M2	DWORD	DW#16#0	
+32.0	b_32_M2	WORD	W#16#0	
+34.0	SERVO_EN_M2	BOOL	FALSE	servo enable
+34.1	GAUG_FLY_M2	BOOL	FALSE	gauging on the fly
+34.2	b_34_2_M2	BOOL	FALSE	
+34.3	b_34_3_M2	BOOL	FALSE	
+34.4	b_34_4_M2	BOOL	FALSE	
+34.5	TRAV_MON_M2	BOOL	FALSE	traverse monitoring
+34.6	PARK_AX_M2	BOOL	FALSE	parking axis
+34.7	SIM_ON_M2	BOOL	FALSE	simulation on
+35.0	b_35_0_M2	BOOL	FALSE	
+35.1	b_35_1_M2	BOOL	FALSE	
+35.2	MSR_EN_M2	BOOL	FALSE	measurement
+35.3	REF_TRIG_M2	BOOL	FALSE	trigger reference point
+35.4	DI_OFF_M2	BOOL	FALSE	disable enable input
+35.5	FOLLOWUP_M2	BOOL	FALSE	follow-up
+35.6	SSW_DIS_M2	BOOL	FALSE	software switch disable
+35.7	DRIFT_OFF_M2	BOOL	FALSE	automatic drift compensation off
+36.0	b_36_M2	BYTE	B#16#0	
+37.0	MD_EN_M2	BOOL	FALSE	machine data enable
+37.1	DELDIST_EN_M2	BOOL	FALSE	delete distance to go
+37.2	SEARCH_F_M2	BOOL	FALSE	automatic block search forward
+37.3	SEARCH_B_M2	BOOL	FALSE	automatic block search backward
+37.4	b_37_4_M2	BOOL	FALSE	
+37.5	RESET_AX_M2	BOOL	FALSE	reset axis
+37.6	AVALREM_EN_M2	BOOL	FALSE	remove setting actual value
+37.7	b_37_7_M2	BOOL	FALSE	
+38.0	VLEV_EN_M2	BOOL	FALSE	velocity levels 1 and 2
+38.1	CLEV_EN_M2	BOOL	FALSE	control levels 1 and 2
+38.2	TRG254_EN_M2	BOOL	FALSE	target 254
+38.3	MDI_EN_M2	BOOL	FALSE	MDI movement block
+38.4	MDIFLY_EN_M2	BOOL	FALSE	MDI block on the fly
+38.5	b_38_5_M2	BOOL	FALSE	
+38.6	REFPT_EN_M2	BOOL	FALSE	reference coordinate
+38.7	AVAL_EN_M2	BOOL	FALSE	actual value coordinate
+39.0	FVAL_EN_M2	BOOL	FALSE	actual value coordinate (flying)
+39.1	ZOFF_EN_M2	BOOL	FALSE	zero offset value
+39.2	b_39_2_M2	BOOL	FALSE	
+39.3	PARCH_EN_M2	BOOL	FALSE	parameter change
+39.4	DIGO_EN_M2	BOOL	FALSE	digital 0
+39.5	PROGS_EN_M2	BOOL	FALSE	part program selection

(Anexo M) 142

Page 3 of 16

Address	Name	Type	Initial value	Comment
+39.6	REQAPP_EN_M2	BOOL	FALSE	request application data
+39.7	TEACHIN_EN_M2	BOOL	FALSE	Teach In
+40.0	AXCOU_EN_M2	BOOL	FALSE	write job axis coupling FM453
+40.1	b_40_1_M2	BOOL	FALSE	
+40.2	b_40_2_M2	BOOL	FALSE	
+40.3	b_40_3_M2	BOOL	FALSE	
+40.4	b_40_4_M2	BOOL	FALSE	
+40.5	b_40_5_M2	BOOL	FALSE	
+40.6	b_40_6_M2	BOOL	FALSE	
+40.7	b_40_7_M2	BOOL	FALSE	
+41.0	b_41_M2	BYTE	B#16#0	
+42.0	OPDAT_EN_M2	BOOL	FALSE	operating_data
+42.1	ACT_BL_EN_M2	BOOL	FALSE	actual part program block
+42.2	NXT_BL_EN_M2	BOOL	FALSE	next part program block
+42.3	BLEXT_EN_M2	BOOL	FALSE	actual value at external block change
+42.4	SERVDAT_EN_M2	BOOL	FALSE	service data
+42.5	OC_ERR_EN_M2	BOOL	FALSE	oc-error
+42.6	b_42_6_M2	BOOL	FALSE	
+42.7	b_42_7_M2	BOOL	FALSE	
+43.0	AXCOURD_EN_M2	BOOL	FALSE	read job axis coupling FM453
+43.1	b_43_1_M2	BOOL	FALSE	
+43.2	b_43_2_M2	BOOL	FALSE	
+43.3	PARRD_EN_M2	BOOL	FALSE	parameter data
+43.4	DIGIO_EN_M2	BOOL	FALSE	digital in_out
+43.5	OPDAT1_EN_M2	BOOL	FALSE	operating data 1
+43.6	APPDAT_EN_M2	BOOL	FALSE	applicating data
+43.7	MSRRD_EN_M2	BOOL	FALSE	read measurement values
+44.0	SERVO_D_M2	BOOL	FALSE	servo enable
+44.1	GAUG_FLY_D_M2	BOOL	FALSE	gauging on the fly
+44.2	b_44_2_M2	BOOL	FALSE	
+44.3	b_44_3_M2	BOOL	FALSE	
+44.4	b_44_4_M2	BOOL	FALSE	
+44.5	TRAV_MON_D_M2	BOOL	FALSE	traverse monitoring
+44.6	PARK_AX_D_M2	BOOL	FALSE	parking axis
+44.7	SIM_ON_D_M2	BOOL	FALSE	simulation on
+45.0	b_45_0_M2	BOOL	FALSE	
+45.1	b_45_1_M2	BOOL	FALSE	
+45.2	MSR_D_M2	BOOL	FALSE	measurement
+45.3	REF_TRIG_D_M2	BOOL	FALSE	trigger reference point
+45.4	DI_OFF_D_M2	BOOL	FALSE	disable enable input
+45.5	FOLLOWUP_D_M2	BOOL	FALSE	follow-up

Page 4 of 16

Address	Name	Type	Initial value	Comment
+45.6	SSW_DIS_D_M2	BOOL	FALSE	software switch disable
+45.7	DRIFT_OFF_D_M2	BOOL	FALSE	automatic drift compensation off
+46.0	b_46_M2	BYTE	B#16#0	
+47.0	MD_D_M2	BOOL	FALSE	machine data enable
+47.1	DELDIST_D_M2	BOOL	FALSE	delete distance to go
+47.2	SEARCH_F_D_M2	BOOL	FALSE	automatic block search forward
+47.3	SEARCH_B_D_M2	BOOL	FALSE	automatic block search backward
+47.4	b_47_4_M2	BOOL	FALSE	
+47.5	RESET_AX_D_M2	BOOL	FALSE	reset axis
+47.6	AVALREM_D_M2	BOOL	FALSE	remove setting actual value
+47.7	b_47_7_M2	BOOL	FALSE	
+48.0	VLEV_D_M2	BOOL	FALSE	velocity levels 1 and 2
+48.1	CLEV_D_M2	BOOL	FALSE	control levels 1 and 2
+48.2	TRG254_D_M2	BOOL	FALSE	target 254
+48.3	MDI_D_M2	BOOL	FALSE	MDI movement block
+48.4	MDIFLY_D_M2	BOOL	FALSE	MDI block on the fly
+48.5	b_48_5_M2	BOOL	FALSE	
+48.6	REFPT_D_M2	BOOL	FALSE	reference coordinate
+48.7	AVAL_D_M2	BOOL	FALSE	actual value coordinate
+49.0	FVAL_D_M2	BOOL	FALSE	actual value coordinate (flying)
+49.1	ZOFF_D_M2	BOOL	FALSE	zero offset value
+49.2	b_49_2_M2	BOOL	FALSE	
+49.3	PARCH_D_M2	BOOL	FALSE	parameter change
+49.4	DIGO_D_M2	BOOL	FALSE	digital 0
+49.5	PROGS_D_M2	BOOL	FALSE	part program selection
+49.6	REQAPP_D_M2	BOOL	FALSE	request application data
+49.7	TEACHIN_D_M2	BOOL	FALSE	Teach In
+50.0	AXCOU_D_M2	BOOL	FALSE	axis coupling FM453
+50.1	b_50_1_M2	BOOL	FALSE	
+50.2	b_50_2_M2	BOOL	FALSE	
+50.3	b_50_3_M2	BOOL	FALSE	
+50.4	b_50_4_M2	BOOL	FALSE	
+50.5	b_50_5_M2	BOOL	FALSE	
+50.6	b_50_6_M2	BOOL	FALSE	
+50.7	b_50_7_M2	BOOL	FALSE	
+51.0	b_51_M2	BYTE	B#16#0	
+52.0	OPDAT_D_M2	BOOL	FALSE	operating_data
+52.1	ACT_BL_D_M2	BOOL	FALSE	actual part program block
+52.2	NXT_BL_D_M2	BOOL	FALSE	next part program block
+52.3	BLEXT_D_M2	BOOL	FALSE	actual value at external block change
+52.4	SERVDAT_D_M2	BOOL	FALSE	service data

(Anexo M) 143

Page 5 of 16

Address	Name	Type	Initial value	Comment
+52.5	OC_ERR_D_M2	BOOL	FALSE	oc_error
+52.6	OT_ERR_D_M2	BOOL	FALSE	ot_error
+52.7	DA_ERR_D_M2	BOOL	FALSE	data error
+53.0	AXCOURD_D_M2	BOOL	FALSE	axis coupling FM453
+53.1	b_53_1_M2	BOOL	FALSE	
+53.2	b_53_2_M2	BOOL	FALSE	
+53.3	PARRD_D_M2	BOOL	FALSE	parameter data
+53.4	DIGIO_D_M2	BOOL	FALSE	digital in_out
+53.5	OPDAT1_D_M2	BOOL	FALSE	operating data 1
+53.6	APPDAT_D_M2	BOOL	FALSE	applicating data
+53.7	MSRRD_D_M2	BOOL	FALSE	read measurement values
+54.0	SERVO_ERR_M2	BOOL	FALSE	servo enable
+54.1	GAUG_FLY_ERR_M2	BOOL	FALSE	gauging on the fly
+54.2	b_54_2_M2	BOOL	FALSE	
+54.3	b_54_3_M2	BOOL	FALSE	
+54.4	b_54_4_M2	BOOL	FALSE	
+54.5	TRAV_MON_ERR_M2	BOOL	FALSE	traverse monitoring
+54.6	PARK_AX_ERR_M2	BOOL	FALSE	parking axis
+54.7	SIM_ON_ERR_M2	BOOL	FALSE	simulation on
+55.0	b_55_0_M2	BOOL	FALSE	
+55.1	b_55_1_M2	BOOL	FALSE	
+55.2	MSR_ERR_M2	BOOL	FALSE	measurement
+55.3	REF_TRIG_ERR_M2	BOOL	FALSE	trigger reference point
+55.4	DI_OFF_ERR_M2	BOOL	FALSE	disable enable input
+55.5	FOLLOWUP_ERR_M2	BOOL	FALSE	follow-up
+55.6	SSW_DIS_ERR_M2	BOOL	FALSE	software switch disable
+55.7	DRIFT_OFF_ERR_M2	BOOL	FALSE	automatic drift compensation off
+56.0	b_56_M2	BYTE	B#16#0	
+57.0	MD_ERR_M2	BOOL	FALSE	machine data enable
+57.1	DELDIST_ERR_M2	BOOL	FALSE	delete distance to go
+57.2	SEARCH_F_ERR_M2	BOOL	FALSE	automatic block search forward
+57.3	SEARCH_B_ERR_M2	BOOL	FALSE	automatic block search backward
+57.4	b_57_4_M2	BOOL	FALSE	
+57.5	RESET_AX_ERR_M2	BOOL	FALSE	reset axis
+57.6	AVALREM_ERR_M2	BOOL	FALSE	remove setting actual value
+57.7	b_57_7_M2	BOOL	FALSE	
+58.0	VLEV_ERR_M2	BOOL	FALSE	velocity levels 1 and 2
+58.1	CLEV_ERR_M2	BOOL	FALSE	control levels 1 and 2
+58.2	TRG254_ERR_M2	BOOL	FALSE	target 254
+58.3	MDI_ERR_M2	BOOL	FALSE	MDI movement block
+58.4	MDIFLY_ERR_M2	BOOL	FALSE	MDI block on the fly

Page 6 of 16

Address	Name	Type	Initial value	Comment
+58.5	b_58_5_M2	BOOL	FALSE	
+58.6	REFPT_ERR_M2	BOOL	FALSE	reference coordinate
+58.7	AVAL_ERR_M2	BOOL	FALSE	actual value coordinate
+59.0	FVAL_ERR_M2	BOOL	FALSE	actual value coordinate (flying)
+59.1	ZOFF_ERR_M2	BOOL	FALSE	zero offset value
+59.2	b_59_2_M2	BOOL	FALSE	
+59.3	PARCH_ERR_M2	BOOL	FALSE	parameter change
+59.4	DIGO_ERR_M2	BOOL	FALSE	digital 0
+59.5	PROGS_ERR_M2	BOOL	FALSE	part program selection
+59.6	REQAPP_ERR_M2	BOOL	FALSE	request application data
+59.7	TEACHIN_ERR_M2	BOOL	FALSE	Teach In
+60.0	AXCOU_ERR_M2_M2	BOOL	FALSE	axis coupling write error FM453
+60.1	b_60_1_M2	BOOL	FALSE	
+60.2	b_60_2_M2	BOOL	FALSE	
+60.3	b_60_3_M2	BOOL	FALSE	
+60.4	b_60_4_M2	BOOL	FALSE	
+60.5	b_60_5_M2	BOOL	FALSE	
+60.6	b_60_6_M2	BOOL	FALSE	
+60.7	b_60_7_M2	BOOL	FALSE	
+61.0	b_61_M2	BYTE	B#16#0	
+62.0	OPDAT_ERR_M2	BOOL	FALSE	operating_data
+62.1	ACT_BL_ERR_M2	BOOL	FALSE	actual part program block
+62.2	NXT_BL_ERR_M2	BOOL	FALSE	next part program block
+62.3	BLEXT_ERR_M2	BOOL	FALSE	actual value at external block change
+62.4	SERVDAT_ERR_M2	BOOL	FALSE	service data
+62.5	OC_ERR_ERR_M2	BOOL	FALSE	oc_error
+62.6	OT_ERR_ERR_M2	BOOL	FALSE	ot_error
+62.7	DA_ERR_ERR_M2	BOOL	FALSE	data error
+63.0	AXCOURD_ERR_M2	BOOL	FALSE	axis coupling read error FM453
+63.1	b_63_1_M2	BOOL	FALSE	
+63.2	b_63_2_M2	BOOL	FALSE	
+63.3	PARRD_ERR_M2	BOOL	FALSE	parameter data
+63.4	DIGIO_ERR_M2	BOOL	FALSE	digital in_out
+63.5	OPDAT1_ERR_M2	BOOL	FALSE	operating data 1
+63.6	APPDAT_ERR_M2	BOOL	FALSE	applicating data
+63.7	MSRRD_ERR_M2	BOOL	FALSE	read measurement values
+64.0	b_64_M2	WORD	W#16#0	
+66.0	JOB_ERR_M2	INT	0	error-code SFC58/59 (FC POS_CTRL)
+68.0	JOBBUSY_WR_M2	BOOL	FALSE	job aktiv write
+68.1	IMPO_WR_M2	BOOL	FALSE	imposs write
+68.2	JOBBUSY_RD_M2	BOOL	FALSE	job aktiv read

(Anexo M) 144

Page 7 of 16

Address	Name	Type	Initial value	Comment
+68.3	IMPO_RD_M2	BOOL	FALSE	imposs read
+68.4	b_68_4_M2	BOOL	FALSE	
+68.5	b_68_5_M2	BOOL	FALSE	
+68.6	b_68_6_M2	BOOL	FALSE	
+68.7	b_68_7_M2	BOOL	FALSE	
+69.0	b_69_0_M2	BOOL	FALSE	
+69.1	JOBRESET_M2	BOOL	FALSE	Reset_ERR and _D
+69.2	b_69_2_M2	BOOL	FALSE	
+69.3	b_69_3_M2	BOOL	FALSE	
+69.4	b_69_4_M2	BOOL	FALSE	
+69.5	b_69_5_M2	BOOL	FALSE	
+69.6	b_69_6_M2	BOOL	FALSE	
+69.7	b_69_7_M2	BOOL	FALSE	
+70.0	MDL_DEFECT_M2	BOOL	FALSE	Module defective
+70.1	INT_FAULT_M2	BOOL	FALSE	Internal fault
+70.2	EXT_FAULT_M2	BOOL	FALSE	External fault
+70.3	PNT_INFO_M2	BOOL	FALSE	Point information
+70.4	b_70_4_M2	BOOL	FALSE	
+70.5	FLD_CONNCTR_M2	BOOL	FALSE	Field wiring connector missing
+70.6	NO_CONFIG_M2	BOOL	FALSE	Module has no configuration data
+70.7	b_70_7_M2	BOOL	FALSE	
+71.0	MDL_TYPE_M2	BYTE	B#16#0	Type of module
+72.0	b_72_0_M2	BOOL	FALSE	
+72.1	COMM_FAULT_M2	BOOL	FALSE	Communication fault
+72.2	b_72_2_M2	BOOL	FALSE	
+72.3	WTCH_DOG_FLT_M2	BOOL	FALSE	Watch dog timer stopped module
+72.4	INT_PS_FLT_M2	BOOL	FALSE	Internal power supply fault
+72.5	b_72_5_M2	BOOL	FALSE	
+72.6	b_72_6_M2	BOOL	FALSE	
+72.7	b_72_7_M2	BOOL	FALSE	
+73.0	b_73_0_M2	BOOL	FALSE	
+73.1	b_73_1_M2	BOOL	FALSE	
+73.2	EPROM_FLT_M2	BOOL	FALSE	EPROM fault
+73.3	RAM_FLT_M2	BOOL	FALSE	RAM fault
+73.4	b_73_4_M2	BOOL	FALSE	
+73.5	b_73_5_M2	BOOL	FALSE	
+73.6	HW_INTR_FLT_M2	BOOL	FALSE	Hardware interrupt input in fault
+73.7	b_73_7_M2	BOOL	FALSE	
+74.0	POS_ID_M2	BYTE	B#16#0	channel type info
+75.0	LEN_INFO_M2	BYTE	B#16#0	channel info length
+76.0	CHEN_NO_M2	BYTE	B#16#0	number of channels

Page 8 of 16

Address	Name	Type	Initial value	Comment
+77.0	CH_ERR_VE1_M2	BOOL	FALSE	channel error vector 1
+77.1	CH_ERR_VE2_M2	BOOL	FALSE	channel error vector 2
+77.2	CH_ERR_VE3_M2	BOOL	FALSE	channel error vector 3
+77.3	b_77_3_M2	BOOL	FALSE	
+77.4	b_77_4_M2	BOOL	FALSE	
+77.5	b_77_5_M2	BOOL	FALSE	
+77.6	b_77_6_M2	BOOL	FALSE	
+77.7	b_77_7_M2	BOOL	FALSE	
+78.0	CAB_BR1_M2	BOOL	FALSE	channel 1 cablebraek
+78.1	ERR_ABE1_M2	BOOL	FALSE	channel 1 error sensor_absolut
+78.2	ERR_PU1_M2	BOOL	FALSE	channel 1 error impulse
+78.3	VO_ENC1_M2	BOOL	FALSE	channel 1 voltage control sensor
+78.4	VO_15_1_M2	BOOL	FALSE	channel 1 voltage control 15 V
+78.5	VO_DIO1_M2	BOOL	FALSE	channel 1 voltage control digital output
+78.6	b_78_6_M2	BOOL	FALSE	
+78.7	OC_ERR_EN1_M2	BOOL	FALSE	channel 1 operation error en1
+79.0	b_79_M2	BYTE	B#16#0	
+80.0	CAB_BR2_M2	BOOL	FALSE	channel 2 cablebraek
+80.1	ERR_ABE2_M2	BOOL	FALSE	channel 2 error sensor_absolut
+80.2	ERR_PU2_M2	BOOL	FALSE	channel 2 error impulse
+80.3	VO_ENC2_M2	BOOL	FALSE	channel 2 voltage control sensor
+80.4	VO_15_2_M2	BOOL	FALSE	channel 2 voltage control 15 V
+80.5	VO_DIO2_M2	BOOL	FALSE	channel 2 voltage control digital output
+80.6	b_80_6_M2	BOOL	FALSE	
+80.7	OC_ERR_EN2_M2	BOOL	FALSE	channel 2 operation error en2
+81.0	b_81_M2	BYTE	B#16#0	
+82.0	CAB_BR3_M2	BOOL	FALSE	channel 3 cablebraek
+82.1	ERR_ABE3_M2	BOOL	FALSE	channel 3 error sensor_absolut
+82.2	ERR_PU3_M2	BOOL	FALSE	channel 3 error impulse
+82.3	VO_ENC3_M2	BOOL	FALSE	channel 3 voltage control sensor
+82.4	VO_15_3_M2	BOOL	FALSE	channel 3 voltage control 15 V
+82.5	VO_DIO3_M2	BOOL	FALSE	channel 3 voltage control digital output
+82.6	b_82_6_M2	BOOL	FALSE	
+82.7	OC_ERR_EN3_M2	BOOL	FALSE	channel 3 operation error en3
+83.0	b_83_M2	BYTE	B#16#0	
+84.0	b_84_M2	WORD	W#16#0	
+86.0	OC_ERR_NO_M2	BYTE	B#16#0	operation error, detail event number
+87.0	OC_REE_CL_M2	BYTE	B#16#0	detail event class
+88.0	b_88_M2	WORD	W#16#0	
+90.0	OT_ERR_NO_M2	BYTE	B#16#0	control/process error, detail event number
+91.0	OT_ERR_CL_M2	BYTE	B#16#0	detail event class

(Anexo M) 145

Page 9 of 16

Address	Name	Type	Initial value	Comment
+92.0	b_92_M2	WORD	W#16#0	
+94.0	DA_ERR_NO_M2	BYTE	B#16#0	data error, detail event number
+95.0	DA_ERR_CL_M2	BYTE	B#16#0	detail event class
+96.0	DIAG_ERR_M2	INT	0	error-code SFC51 (FC POS_DIAG)
+98.0	MSRM_ERR_M2	INT	0	error-code SFC59 (FC POS_MSRM)
+100.0	feld1_intern_M2	ARRAY[100..139]		
*1.0		BYTE		
+140.0	ZOFF_M2	DINT	L#0	zero offset value
+144.0	AVAL_M2	DINT	L#0	actual value coordinate
+148.0	FVAL_M2	DINT	L#0	actual value coordinate (flying)
+152.0	REFPT_M2	DINT	L#0	reference coordinate
+156.0	TRG254_M2	DWORD	DW#16#0	target 254
+160.0	VLEVEL_1_M2	DWORD	DW#16#0	velocity level 1
+164.0	VLEVEL_2_M2	DWORD	DW#16#0	velocity level 2
+168.0	CLEVEL_1_M2	DWORD	DW#16#0	control level 1
+172.0	CLEVEL_2_M2	DWORD	DW#16#0	control level 2
+176.0	MDIB_M2	STRUCT		MDI movement block
+0.0	b_176_M2	BYTE	B#16#0	
+1.0	b_177_M2	BYTE	B#16#0	
+2.0	G_1_EN_M2	BOOL	FALSE	enable bit for G function group 1
+2.1	G_2_EN_M2	BOOL	FALSE	enable bit for G function group 2
+2.2	b_178_2_M2	BOOL	FALSE	
+2.3	b_178_3_M2	BOOL	FALSE	
+2.4	X_T_EN_M2	BOOL	FALSE	enable bit for position / dwell time
+2.5	b_178_5_M2	BOOL	FALSE	
+2.6	b_178_6_M2	BOOL	FALSE	
+2.7	b_178_7_M2	BOOL	FALSE	
+3.0	V_EN_M2	BOOL	FALSE	enable bit for velocity
+3.1	M_1_EN_M2	BOOL	FALSE	enable bit for M function group 1
+3.2	M_2_EN_M2	BOOL	FALSE	enable bit for M function group 2
+3.3	M_3_EN_M2	BOOL	FALSE	enable bit for M function group 3
+3.4	b_179_4_M2	BOOL	FALSE	
+3.5	b_179_5_M2	BOOL	FALSE	
+3.6	b_179_6_M2	BOOL	FALSE	
+3.7	b_179_7_M2	BOOL	FALSE	
+4.0	G_1_VAL_M2	BYTE	B#16#0	value G function group 1
+5.0	G_2_VAL_M2	BYTE	B#16#0	value G function group 2
+6.0	b_182_M2	BYTE	B#16#0	
+7.0	b_183_M2	BYTE	B#16#0	
+8.0	X_T_VAL_M2	DINT	L#0	value position / dwell time
+12.0	V_VAL_M2	DINT	L#0	value velocity

Page 10 of 16

Address	Name	Type	Initial value	Comment
+16.0	M_1_VAL_M2	BYTE	B#16#0	value M function group 1
+17.0	M_2_VAL_M2	BYTE	B#16#0	value M function group 2
+18.0	M_3_VAL_M2	BYTE	B#16#0	value M function group 3
+19.0	b_195_M2	BYTE	B#16#0	
=20.0		END_STRUCT		
+196.0	PAR_CHAN_M2	STRUCT		parameter change
+0.0	TYP_M2	BYTE	B#16#0	DB type
+1.0	NUMB_M2	BYTE	B#16#0	data number
+2.0	COUN_M2	BYTE	B#16#0	number of data
+3.0	JOB_M2	BYTE	B#16#0	job type
+4.0	DATA_M2	ARRAY[200..219]		
*1.0		BYTE		
=24.0		END_STRUCT		
+220.0	D_IN0_M2	BOOL	FALSE	digital input 0
+220.1	D_IN1_M2	BOOL	FALSE	digital input 1
+220.2	D_IN2_M2	BOOL	FALSE	digital input 2
+220.3	D_IN3_M2	BOOL	FALSE	digital input 3
+220.4	b_220_4_M2	BOOL	FALSE	
+220.5	b_220_5_M2	BOOL	FALSE	
+220.6	b_220_6_M2	BOOL	FALSE	
+220.7	b_220_7_M2	BOOL	FALSE	
+221.0	D_OUT0_M2	BOOL	FALSE	digital output 0
+221.1	D_OUT1_M2	BOOL	FALSE	digital output 1
+221.2	D_OUT2_M2	BOOL	FALSE	digital output 2
+221.3	D_OUT3_M2	BOOL	FALSE	digital output 3
+221.4	b_221_4_M2	BOOL	FALSE	
+221.5	b_221_5_M2	BOOL	FALSE	
+221.6	b_221_6_M2	BOOL	FALSE	
+221.7	b_221_7_M2	BOOL	FALSE	
+222.0	MDI_F_M2	STRUCT		MDI block on the fly
+0.0	b_222_M2	BYTE	B#16#0	
+1.0	b_223_M2	BYTE	B#16#0	
+2.0	G_1_EN_M2	BOOL	FALSE	enable bit for G function group 1
+2.1	G_2_EN_M2	BOOL	FALSE	enable bit for G function group 2
+2.2	b_224_2_M2	BOOL	FALSE	
+2.3	b_224_3_M2	BOOL	FALSE	
+2.4	X_T_EN_M2	BOOL	FALSE	enable bit for position / dwell time
+2.5	b_224_5_M2	BOOL	FALSE	
+2.6	b_224_6_M2	BOOL	FALSE	
+2.7	b_224_7_M2	BOOL	FALSE	
+3.0	V_EN_M2	BOOL	FALSE	enable bit for velocity

(Anexo M) 146

Page 11 of 16

Address	Name	Type	Initial value	Comment
+3.1	M_1_EN_M2	BOOL	FALSE	enable bit for M function group 1
+3.2	M_2_EN_M2	BOOL	FALSE	enable bit for M function group 2
+3.3	M_3_EN_M2	BOOL	FALSE	enable bit for M function group 3
+3.4	b_225_4_M2	BOOL	FALSE	
+3.5	b_225_5_M2	BOOL	FALSE	
+3.6	b_225_6_M2	BOOL	FALSE	
+3.7	b_225_7_M2	BOOL	FALSE	
+4.0	G_1_VAL_M2	BYTE	B#16#0	value G function group 1
+5.0	G_2_VAL_M2	BYTE	B#16#0	value G function group 2
+6.0	b_228_M2	BYTE	B#16#0	
+7.0	b_229_M2	BYTE	B#16#0	
+8.0	X_T_VAL_M2	DINT	L#0	value position / dwell time
+12.0	V_VAL_M2	DINT	L#0	value velocity
+16.0	M_1_VAL_M2	BYTE	B#16#0	value M function group 1
+17.0	M_2_VAL_M2	BYTE	B#16#0	value M function group 2
+18.0	M_3_VAL_M2	BYTE	B#16#0	value M function group 3
+19.0	b_241_M2	BYTE	B#16#0	
=20.0		END_STRUCT		
+242.0	PROG_NO_M2	BYTE	B#16#0	part program number
+243.0	BLCK_NO_M2	BYTE	B#16#0	part program block number
+244.0	PROG_DIR_M2	BYTE	B#16#0	program run direction
+245.0	b_245_M2	BYTE	B#16#0	
+246.0	CODE_AP1_M2	BYTE	B#16#0	code application 1
+247.0	CODE_AP2_M2	BYTE	B#16#0	code application 2
+248.0	CODE_AP3_M2	BYTE	B#16#0	code application 3
+249.0	CODE_AP4_M2	BYTE	B#16#0	code application 4
+250.0	TEA_PRO_NO_M2	BYTE	B#16#0	part program number
+251.0	TEA_BLK_NO_M2	BYTE	B#16#0	part program block number
+252.0	AXCOU_DEF_M2	BYTE	B#16#0	axis coupling locate DS24 FM453
+253.0	b_253_M2	BYTE	B#16#0	
+254.0	feld2_intern_M2	ARRAY[254..309]		
*1.0		BYTE		
+310.0	ACT_VAL_M2	DINT	L#0	actual value
+314.0	SPEED_M2	DWORD	DW#16#0	speed
+318.0	REM_DIST_M2	DINT	L#0	remaining distance
+322.0	SET_POS_M2	DINT	L#0	set position to be approached
+326.0	SUM_OFST_M2	DINT	L#0	sum of tool offset and zero offset
+330.0	TRAV_SPE_M2	DWORD	DW#16#0	traversing speed
+334.0	di_334_M2	DINT	L#0	
+338.0	di_338_M2	DINT	L#0	
+342.0	ACT_BL_M2	STRUCT		actual part program block

Page 12 of 16

Address	Name	Type	Initial value	Comment
+0.0	PROG_NO_M2	BYTE	B#16#0	part program number
+1.0	BLCK_NO_M2	BYTE	B#16#0	part program block number
+2.0	G_1_EN_M2	BOOL	FALSE	enable bit for G function group 1
+2.1	G_2_EN_M2	BOOL	FALSE	enable bit for G function group 2
+2.2	G_3_EN_M2	BOOL	FALSE	enable bit for G function group 3
+2.3	b_344_3_M2	BOOL	FALSE	
+2.4	X_T_EN_M2	BOOL	FALSE	enable bit for position / dwell time
+2.5	SR_L_EN_M2	BOOL	FALSE	enable bit for number of subroutine loops
+2.6	SR_N_EN_M2	BOOL	FALSE	enable bit for subroutine number
+2.7	SKIP_EN_M2	BOOL	FALSE	enable bit for block skip
+3.0	V_EN_M2	BOOL	FALSE	enable bit for velocity
+3.1	M_1_EN_M2	BOOL	FALSE	enable bit for M function group 1
+3.2	M_2_EN_M2	BOOL	FALSE	enable bit for M function group 2
+3.3	M_3_EN_M2	BOOL	FALSE	enable bit for M function group 3
+3.4	TO_EN_M2	BOOL	FALSE	enable bit for tool offset number
+3.5	b_345_5_M2	BOOL	FALSE	
+3.6	b_345_6_M2	BOOL	FALSE	
+3.7	b_345_7_M2	BOOL	FALSE	
+4.0	G_1_VAL_M2	BYTE	B#16#0	G function group 1
+5.0	G_2_VAL_M2	BYTE	B#16#0	G function group 2
+6.0	G_3_VAL_M2	BYTE	B#16#0	G function group 3
+7.0	b_349_M2	BYTE	B#16#0	
+8.0	X_T_VAL_M2	DINT	L#0	position / dwell time / subroutine number
+12.0	V_VAL_M2	DINT	L#0	velocity / subroutine loops
+16.0	M_1_VAL_M2	BYTE	B#16#0	M function group 1
+17.0	M_2_VAL_M2	BYTE	B#16#0	M function group 2
+18.0	M_3_VAL_M2	BYTE	B#16#0	M function group 3
+19.0	TO_VAL_M2	BYTE	B#16#0	tool offset number
=20.0		END_STRUCT		
+362.0	NXT_BLK_M2	STRUCT		next part program block
+0.0	PROG_NO_M2	BYTE	B#16#0	part program number
+1.0	BLCK_NO_M2	BYTE	B#16#0	part program block number
+2.0	G_1_EN_M2	BOOL	FALSE	enable bit for G function group 1
+2.1	G_2_EN_M2	BOOL	FALSE	enable bit for G function group 2
+2.2	G_3_EN_M2	BOOL	FALSE	enable bit for G function group 3
+2.3	b_364_3_M2	BOOL	FALSE	
+2.4	X_T_EN_M2	BOOL	FALSE	enable bit for position / dwell time
+2.5	SR_L_EN_M2	BOOL	FALSE	enable bit for number of subroutine loops
+2.6	SR_N_EN_M2	BOOL	FALSE	enable bit for subroutine number
+2.7	SKIP_EN_M2	BOOL	FALSE	enable bit for block skip
+3.0	V_EN_M2	BOOL	FALSE	enable bit for velocity

(Anexo M) 147

Page 13 of 16

Address	Name	Type	Initial value	Comment
+3.1	M_1_EN_M2	BOOL	FALSE	enable bit for M function group 1
+3.2	M_2_EN_M2	BOOL	FALSE	enable bit for M function group 2
+3.3	M_3_EN_M2	BOOL	FALSE	enable bit for M function group 3
+3.4	TO_EN_M2	BOOL	FALSE	enable bit for tool offset number
+3.5	b_365_5_M2	BOOL	FALSE	
+3.6	b_365_6_M2	BOOL	FALSE	
+3.7	b_365_7_M2	BOOL	FALSE	
+4.0	G_1_VAL_M2	BYTE	B#16#0	G function group 1
+5.0	G_2_VAL_M2	BYTE	B#16#0	G function group 2
+6.0	G_3_VAL_M2	BYTE	B#16#0	G function group 3
+7.0	b_369_M2	BYTE	B#16#0	
+8.0	X_T_VAL_M2	DINT	L#0	position / dwell time / subroutine number
+12.0	V_VAL_M2	DINT	L#0	velocity / subroutine loops
+16.0	M_1_VAL_M2	BYTE	B#16#0	M function group 1
+17.0	M_2_VAL_M2	BYTE	B#16#0	M function group 2
+18.0	M_3_VAL_M2	BYTE	B#16#0	M function group 3
+19.0	TO_VAL_M2	BYTE	B#16#0	tool offset number
=20.0		END_STRUCT		
+382.0	APP1_M2	DINT	L#0	application data 1
+386.0	APP2_M2	DINT	L#0	application data 2
+390.0	APP3_M2	DINT	L#0	application data 3
+394.0	APP4_M2	DINT	L#0	application data 4
+398.0	BLCK_EXT_M2	DINT	L#0	actual value at external block change
+402.0	OUT_VAL_M2	DINT	L#0	DAC value / frequency
+406.0	ENC_VAL_M2	DINT	L#0	encoder actual value
+410.0	PULS_ERR_M2	DINT	L#0	error pulses
+414.0	KV_FA_M2	DINT	L#0	Kv factor
+418.0	FOLL_ERR_M2	DINT	L#0	following error
+422.0	FERR_LIM_M2	DINT	L#0	following error limit
+426.0	OSC_ERR_M2	DINT	L#0	oscillation error
+430.0	DR_TIME_M2	DINT	L#0	drive time
+434.0	OVERRIDE1_M2	BYTE	B#16#0	override
+435.0	PROG_NO1_M2	BYTE	B#16#0	part program number
+436.0	BLCK_NO1_M2	BYTE	B#16#0	part program block number
+437.0	LOOP_NO1_M2	BYTE	B#16#0	number of subroutine loops
+438.0	G90_91_M2	BYTE	B#16#0	G90/91 active
+439.0	G60_64_M2	BYTE	B#16#0	G60/64 active
+440.0	G43_44_M2	BYTE	B#16#0	G43/44 active
+441.0	TO_NO_M2	BYTE	B#16#0	tool offset number
+442.0	b_442_0_M2	BOOL	FALSE	
+442.1	LIM_SP_M2	BOOL	FALSE	limitation speed

Page 14 of 16

Address	Name	Type	Initial value	Comment
+442.2	LIM_10_M2	BOOL	FALSE	limitation +/- 10V
+442.3	LIM_SU_M2	BOOL	FALSE	limitation speed up
+442.4	b_442_4_M2	BOOL	FALSE	
+442.5	b_442_5_M2	BOOL	FALSE	
+442.6	b_442_6_M2	BOOL	FALSE	
+442.7	b_442_7_M2	BOOL	FALSE	
+443.0	LIM_FR_M2	BOOL	FALSE	limitation frequency
+443.1	LIM_FV_M2	BOOL	FALSE	limitation frequency value
+443.2	b_443_2_M2	BOOL	FALSE	
+443.3	LIM_FS_M2	BOOL	FALSE	limitation frequency speed-up
+443.4	b_443_4_M2	BOOL	FALSE	
+443.5	b_443_5_M2	BOOL	FALSE	
+443.6	b_443_6_M2	BOOL	FALSE	
+443.7	b_443_7_M2	BOOL	FALSE	
+444.0	b_444_M2	BYTE	B#16#0	
+445.0	b_445_M2	BYTE	B#16#0	
+446.0	PAR_RD_M2	STRUCT		PAR_READ
+0.0	TYP1_M2	BYTE	B#16#0	DB type
+1.0	NUMB_M2	BYTE	B#16#0	data number
+2.0	COUN_M2	BYTE	B#16#0	number of data
+3.0	JOB_M2	BYTE	B#16#0	job type
+4.0	DATA1_M2	ARRAY[450..469]		
*1.0		BYTE		
=24.0		END_STRUCT		
+470.0	AXCOU_STAT_M2	BYTE	B#16#0	status axis coupling DS37 FM453
+471.0	b_471_M2	BYTE	B#16#0	
+472.0	feld3_intern_M2	ARRAY[472..485]		
*1.0		BYTE		
+486.0	BEGIN_VAL_M2	DINT	L#0	begin value
+490.0	END_VAL_M2	DINT	L#0	end value
+494.0	LENGTH_VAL_M2	DWORD	DW#16#0	length value
+498.0	USR_M2	STRUCT		user program function bits
+0.0	BITC_0_M2	BOOL	FALSE	MD write request
+0.1	BITC_1_M2	BOOL	FALSE	MD read request
+0.2	BITC_2_M2	BOOL	FALSE	MDI block request
+0.3	BITC_3_M2	BOOL	FALSE	part program selection request
+0.4	BITC_4_M2	BOOL	FALSE	Teach In request
+0.5	BITC_5_M2	BOOL	FALSE	incremental value request
+0.6	BITC_6_M2	BOOL	FALSE	velocity levels request
+0.7	BITC_7_M2	BOOL	FALSE	control levels request
+1.0	BITC_8_M2	BOOL	FALSE	MDI block on the fly request

(Anexo M) 148

Page 15 of 16

Address	Name	Type	Initial value	Comment
+1.1	BITC_9_M2	BOOL	FALSE	actual value on the fly request
+1.2	BITC_10_M2	BOOL	FALSE	zero point offset request
+1.3	BITC_11_M2	BOOL	FALSE	reserved
+1.4	BITC_12_M2	BOOL	FALSE	reserved
+1.5	BITC_13_M2	BOOL	FALSE	Alarm
+1.6	BITC_14_M2	BOOL	FALSE	Data-Error
+1.7	BITC_15_M2	BOOL	FALSE	OT-Error
+2.0	MD_NO_M2	WORD	W#16#0	MD number
+4.0	MD_VALUE_M2	DINT	L#0	MD value
+8.0	INC_NO_M2	BYTE	B#16#0	incremental value number
+9.0	b_507_M2	BYTE	B#16#0	
+10.0	PICT_NO_M2	WORD	W#16#0	picture number
+12.0	KEY_CODE_M2	WORD	W#16#0	keyboard code
+14.0	b_512_M2	WORD	W#16#0	
+16.0	BITA_0_M2	BOOL	FALSE	Control
+16.1	BITA_1_M2	BOOL	FALSE	Reference point approach
+16.2	BITA_2_M2	BOOL	FALSE	Increment
+16.3	BITA_3_M2	BOOL	FALSE	MDI
+16.4	BITA_4_M2	BOOL	FALSE	Automatic / single move
+16.5	BITA_5_M2	BOOL	FALSE	Automatic / follow move
+16.6	BITA_6_M2	BOOL	FALSE	Jog-Mode
+16.7	b_514_7_M2	BOOL	FALSE	
+17.0	b_515_0_M2	BOOL	FALSE	
+17.1	b_515_1_M2	BOOL	FALSE	
+17.2	b_515_2_M2	BOOL	FALSE	
+17.3	b_515_3_M2	BOOL	FALSE	
+17.4	b_515_4_M2	BOOL	FALSE	
+17.5	b_515_5_M2	BOOL	FALSE	
+17.6	BITA_14_M2	BOOL	FALSE	Quit Error
+17.7	BITA_15_M2	BOOL	FALSE	Quit Alarm
=18.0		END_STRUCT		
=516.0		END_STRUCT		

UDT3 - <offline>

"UDT_ENC1" Data type for counter channel (Encoder 1)

Name: CNT_CTY1 Family: FM_CNT_1
 Author: FM Version: 3.0
 Block version: 2
 Time stamp Code: 11/21/2002 09:28:37 AM
 Interface: 11/21/2002 09:28:37 AM
 Lengths (block/logic/data): 00000 00000 00000
 UNLINKED

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	AR1_BUFFER	DWORD	DW#16#0	AR1 buffer (FC internal use)
+4.0	FP	BYTE	B#16#0	Flag byte (FC internal use)
+5.0	RESERVED	BYTE	B#16#0	reserved for FC use
+6.0	MOD_ADR	WORD	W#16#0	Module address (write user)
+8.0	CH_ADR	DWORD	DW#16#0	Channel address (write user)
+12.0	U_D_LGTH	BYTE	B#16#0	User data length (write user)
+13.0	A_BYTE_0	BYTE	B#16#0	reserved
+14.0	LOAD_VAL	DINT	L#0	New load value (write user)
+18.0	CMP_V1	DINT	L#0	New comparator value 1 (write user)
+22.0	CMP_V2	DINT	L#0	New comparator value 2 (write user)
+26.0	A_BIT0_0	BOOL	FALSE	reserved
+26.1	TFB	BOOL	FALSE	Test free (internal use)
+26.2	A_BIT0_2	BOOL	FALSE	reserved
+26.3	A_BIT0_3	BOOL	FALSE	reserved
+26.4	A_BIT0_4	BOOL	FALSE	reserved
+26.5	A_BIT0_5	BOOL	FALSE	reserved
+26.6	A_BIT0_6	BOOL	FALSE	reserved
+26.7	A_BIT0_7	BOOL	FALSE	reserved
+27.0	ENSET_UP	BOOL	FALSE	Enable set in direction up (=forward) (write user)
+27.1	ENSET_DN	BOOL	FALSE	Enable set in direction down (=backward) (write user)
+27.2	A_BIT1_2	BOOL	FALSE	reserved
+27.3	A_BIT1_3	BOOL	FALSE	reserved
+27.4	A_BIT1_4	BOOL	FALSE	reserved
+27.5	A_BIT1_5	BOOL	FALSE	reserved
+27.6	A_BIT1_6	BOOL	FALSE	reserved
+27.7	A_BIT1_7	BOOL	FALSE	reserved
+28.0	CTRL_D00	BOOL	FALSE	Control digital output D00 (write user)
+28.1	CTRL_D01	BOOL	FALSE	Control digital output D01 (write user)

Address	Name	Type	Initial value	Comment
+28.2	A_BIT2_2	BOOL	FALSE	reserved
+28.3	A_BIT2_3	BOOL	FALSE	reserved
+28.4	A_BIT2_4	BOOL	FALSE	reserved
+28.5	A_BIT2_5	BOOL	FALSE	reserved
+28.6	A_BIT2_6	BOOL	FALSE	reserved
+28.7	A_BIT2_7	BOOL	FALSE	reserved
+29.0	A_BIT3_0	BOOL	FALSE	reserved
+29.1	A_BIT3_1	BOOL	FALSE	reserved
+29.2	A_BIT3_2	BOOL	FALSE	reserved
+29.3	A_BIT3_3	BOOL	FALSE	reserved
+29.4	A_BIT3_4	BOOL	FALSE	reserved
+29.5	A_BIT3_5	BOOL	FALSE	reserved
+29.6	A_BIT3_6	BOOL	FALSE	reserved
+29.7	A_BIT3_7	BOOL	FALSE	reserved
+30.0	LATCH_LOAD	DINT	L#0	Actual latch or load value (read user)
+34.0	ACT_CNTV	DINT	L#0	Actual counter value (read user)
+38.0	DA_ERR_W	WORD	W#16#0	Data error word (read user)
+40.0	OT_ERR_B	BYTE	B#16#0	Operator error byte (read user)
+41.0	E_BIT0_0	BOOL	FALSE	reserved
+41.1	STS_TFB	BOOL	FALSE	Status test free bit (internal use)
+41.2	DIAG	BOOL	FALSE	l=Diagnostic buffer changed (read user)
+41.3	E_BIT0_3	BOOL	FALSE	reserved
+41.4	DATA_ERR	BOOL	FALSE	Data error bit (read user)
+41.5	E_BIT0_5	BOOL	FALSE	reserved
+41.6	E_BIT0_6	BOOL	FALSE	reserved
+41.7	PARA	BOOL	FALSE	Module parametrized (read user)
+42.0	E_BYTE_0	BYTE	B#16#0	reserved
+43.0	STS_RUN	BOOL	FALSE	Status counter is running (read user)
+43.1	STS_DIR	BOOL	FALSE	Status of the counter direction bit (read user)
+43.2	STS_ZERO	BOOL	FALSE	Status of the counter zero crossing bit (read user)
+43.3	STS_OFLW	BOOL	FALSE	Status counter passed the overflow value (read user)
+43.4	STS_UFLW	BOOL	FALSE	Status counter passed the underflow value (read user)
+43.5	STS_SYNC	BOOL	FALSE	Status counter is synchronized (read user)
+43.6	STS_GATE	BOOL	FALSE	Status of the internal gate (read user)
+43.7	STS_SW_G	BOOL	FALSE	Status of the software gate (read user)
+44.0	STS_SET	BOOL	FALSE	Status digital input SET (read user)
+44.1	STS_LATCH	BOOL	FALSE	New latch value in isochronous mode (read user)
+44.2	STS_STA	BOOL	FALSE	Status of the digital input start (read user)
+44.3	STS_STP	BOOL	FALSE	Status of the digital input stop (read user)
+44.4	STS_CMP1	BOOL	FALSE	Status of the comparator output 1 (read user)
+44.5	STS_CMP2	BOOL	FALSE	Status of the comparator output 2 (read user)

Address	Name	Type	Initial value	Comment
+44.6	STS_COMP1	BOOL	FALSE	Remanent status of the comparator output 1 (read user)
+44.7	STS_COMP2	BOOL	FALSE	Remanent status of the comparator output 2 (read user)
+45.0	E_BIT3_0	BOOL	FALSE	reserved
+45.1	E_BIT3_1	BOOL	FALSE	reserved
+45.2	E_BIT3_2	BOOL	FALSE	reserved
+45.3	E_BIT3_3	BOOL	FALSE	reserved
+45.4	E_BIT3_4	BOOL	FALSE	reserved
+45.5	E_BIT3_5	BOOL	FALSE	reserved
+45.6	E_BIT3_6	BOOL	FALSE	reserved
+45.7	E_BIT3_7	BOOL	FALSE	reserved
+46.0	ACT_CMP1	DINT	L#0	Actual comparator value 1 (FM 450 only; read user)
+50.0	ACT_CMP2	DINT	L#0	Actual comparator value 2 (FM 450 only; read user)
+54.0	MDL_DEFECT	BOOL	FALSE	Module defective
+54.1	INT_FAULT	BOOL	FALSE	Internal fault
+54.2	EXT_FAULT	BOOL	FALSE	External fault
+54.3	PNT_INFO	BOOL	FALSE	Point information
+54.4	EXT_VOLTAGE	BOOL	FALSE	External voltage low
+54.5	FLD_CONNCTR	BOOL	FALSE	Field wiring connector missing
+54.6	NO_CONFIG	BOOL	FALSE	Module has no configuration data
+54.7	CONFIG_ERR	BOOL	FALSE	Module has configuration error
+55.0	MDL_TYPE	BYTE	B#16#0	Type of module
+56.0	SUB_MDL_ERR	BOOL	FALSE	Sub-Module is missing or has error
+56.1	COMM_FAULT	BOOL	FALSE	Communication fault
+56.2	MDL_STOP	BOOL	FALSE	Module is stopped
+56.3	WTCH_DOG_FLT	BOOL	FALSE	Watch dog timer stopped module
+56.4	INT_PS_FLT	BOOL	FALSE	Internal power supply fault
+56.5	PRIM_BATT_FLT	BOOL	FALSE	Primary battery is in fault
+56.6	BCKUP_BATT_FLT	BOOL	FALSE	Backup battery is in fault
+56.7	RESERVED_2	BOOL	FALSE	reserved for system
+57.0	RACK_FLT	BOOL	FALSE	Rack fault, only for bus interface module
+57.1	PROC_FLT	BOOL	FALSE	Processor fault
+57.2	EPROM_FLT	BOOL	FALSE	EPROM fault
+57.3	RAM_FLT	BOOL	FALSE	RAM fault
+57.4	ADU_FLT	BOOL	FALSE	ADU fault
+57.5	FUSE_FLT	BOOL	FALSE	Fuse fault
+57.6	HW_INTR_FLT	BOOL	FALSE	Hardware interrupt input in fault
+57.7	RESERVED_3	BOOL	FALSE	reserved for system
+58.0	CH_TYPE	BYTE	B#16#0	Channel type
+59.0	LGTH_DIA	BYTE	B#16#0	Length of diagnostics data per channel
+60.0	CH_NO	BYTE	B#16#0	Channel number (numero)
+61.0	GRP_ERR1	BOOL	FALSE	Group error channel 1

Address	Name	Type	Initial value	Comment
+61.1	GRP_ERR2	BOOL	FALSE	Group error channel 2
+61.2	D_BIT7_2	BOOL	FALSE	DS1 byte 7 bit 2
+61.3	D_BIT7_3	BOOL	FALSE	DS1 byte 7 bit 3
+61.4	D_BIT7_4	BOOL	FALSE	DS1 byte 7 bit 4
+61.5	D_BIT7_5	BOOL	FALSE	DS1 byte 7 bit 5
+61.6	D_BIT7_6	BOOL	FALSE	DS1 byte 7 bit 6
+61.7	D_BIT7_7	BOOL	FALSE	DS1 byte 7 bit 7
+62.0	CH1_SIGA	BOOL	FALSE	Channel 1, signal A malfunction
+62.1	CH1_SIGB	BOOL	FALSE	Channel 1, signal B malfunction
+62.2	CH1_SIGZ	BOOL	FALSE	Channel 1, signal Zero mark malfunction
+62.3	CH1_BETW	BOOL	FALSE	Channel 1, malfunction between channels
+62.4	CH1_5V2	BOOL	FALSE	Channel 1, malfunction of encoder supply voltage 5.2V
+62.5	D_BIT8_5	BOOL	FALSE	DS1 byte 8 bit 5
+62.6	D_BIT8_6	BOOL	FALSE	DS1 byte 8 bit 6
+62.7	D_BIT8_7	BOOL	FALSE	DS1 byte 8 bit 7
+63.0	D_BYTE9	BYTE	B#16#0	DS1 byte 9
+64.0	CH2_SIGA	BOOL	FALSE	Channel 2, signal A malfunction
+64.1	CH2_SIGB	BOOL	FALSE	Channel 2, signal B malfunction
+64.2	CH2_SIGZ	BOOL	FALSE	Channel 2, signal Zero mark malfunction
+64.3	CH2_BETW	BOOL	FALSE	Channel 2, malfunction between channels
+64.4	CH2_5V2	BOOL	FALSE	Channel 2, malfunction of encoder supply voltage 5.2V
+64.5	D_BIT10_5	BOOL	FALSE	DS1 byte 10 bit 5
+64.6	D_BIT10_6	BOOL	FALSE	DS1 byte 10 bit 6
+64.7	D_BIT10_7	BOOL	FALSE	DS1 byte 10 bit 7
+65.0	D_BYTE11	BYTE	B#16#0	DS1 byte 11
+66.0	D_BYTE12	BYTE	B#16#0	DS1 byte 12
+67.0	D_BYTE13	BYTE	B#16#0	DS1 byte 13
+68.0	D_BYTE14	BYTE	B#16#0	DS1 byte 14
+69.0	D_BYTE15	BYTE	B#16#0	DS1 byte 15
=70.0		END_STRUCT		

UDT4 - <offline>

"UDT_ENC2" Data type for counter channel (Encoder 2)

Name: CNT_CTY1 **Family:** FM_CNT_1
Author: FM **Version:** 3.0
 Block version: 2
Time stamp Code: 11/21/2002 09:28:37 AM
 Interface: 11/21/2002 09:28:37 AM
Lengths (block/logic/data): 00000 00000 00000
UNLINKED

Address	Name	Type	Initial value	Comment
0.0		STRUCT		
+0.0	AR1_BUFFER	DWORD	DW#16#0	AR1 buffer (FC internal use)
+4.0	FP	BYTE	B#16#0	Flag byte (FC internal use)
+5.0	RESERVED	BYTE	B#16#0	reserved for FC use
+6.0	MOD_ADR	WORD	W#16#0	Module address (write user)
+8.0	CH_ADR	DWORD	DW#16#0	Channel address (write user)
+12.0	U_D_LGTH	BYTE	B#16#0	User data length (write user)
+13.0	A_BYTE_0	BYTE	B#16#0	reserved
+14.0	LOAD_VAL	DINT	L#0	New load value (write user)
+18.0	CMP_V1	DINT	L#0	New comparator value 1 (write user)
+22.0	CMP_V2	DINT	L#0	New comparator value 2 (write user)
+26.0	A_BIT0_0	BOOL	FALSE	reserved
+26.1	TFB	BOOL	FALSE	Test free (internal use)
+26.2	A_BIT0_2	BOOL	FALSE	reserved
+26.3	A_BIT0_3	BOOL	FALSE	reserved
+26.4	A_BIT0_4	BOOL	FALSE	reserved
+26.5	A_BIT0_5	BOOL	FALSE	reserved
+26.6	A_BIT0_6	BOOL	FALSE	reserved
+26.7	A_BIT0_7	BOOL	FALSE	reserved
+27.0	ENSET_UP	BOOL	FALSE	Enable set in direction up (=forward) (write user)
+27.1	ENSET_DN	BOOL	FALSE	Enable set in direction down (=backward) (write user)
+27.2	A_BIT1_2	BOOL	FALSE	reserved
+27.3	A_BIT1_3	BOOL	FALSE	reserved
+27.4	A_BIT1_4	BOOL	FALSE	reserved
+27.5	A_BIT1_5	BOOL	FALSE	reserved
+27.6	A_BIT1_6	BOOL	FALSE	reserved
+27.7	A_BIT1_7	BOOL	FALSE	reserved
+28.0	CTRL_D00	BOOL	FALSE	Control digital output D00 (write user)
+28.1	CTRL_D01	BOOL	FALSE	Control digital output D01 (write user)

Address	Name	Type	Initial value	Comment
+28.2	A_BIT2_2	BOOL	FALSE	reserved
+28.3	A_BIT2_3	BOOL	FALSE	reserved
+28.4	A_BIT2_4	BOOL	FALSE	reserved
+28.5	A_BIT2_5	BOOL	FALSE	reserved
+28.6	A_BIT2_6	BOOL	FALSE	reserved
+28.7	A_BIT2_7	BOOL	FALSE	reserved
+29.0	A_BIT3_0	BOOL	FALSE	reserved
+29.1	A_BIT3_1	BOOL	FALSE	reserved
+29.2	A_BIT3_2	BOOL	FALSE	reserved
+29.3	A_BIT3_3	BOOL	FALSE	reserved
+29.4	A_BIT3_4	BOOL	FALSE	reserved
+29.5	A_BIT3_5	BOOL	FALSE	reserved
+29.6	A_BIT3_6	BOOL	FALSE	reserved
+29.7	A_BIT3_7	BOOL	FALSE	reserved
+30.0	LATCH_LOAD	DINT	L#0	Actual latch or load value (read user)
+34.0	ACT_CNTV	DINT	L#0	Actual counter value (read user)
+38.0	DA_ERR_W	WORD	W#16#0	Data error word (read user)
+40.0	OT_ERR_B	BYTE	B#16#0	Operator error byte (read user)
+41.0	E_BIT0_0	BOOL	FALSE	reserved
+41.1	STS_TFB	BOOL	FALSE	Status test free bit (internal use)
+41.2	DIAG	BOOL	FALSE	l=Diagnostic buffer changed (read user)
+41.3	E_BIT0_3	BOOL	FALSE	reserved
+41.4	DATA_ERR	BOOL	FALSE	Data error bit (read user)
+41.5	E_BIT0_5	BOOL	FALSE	reserved
+41.6	E_BIT0_6	BOOL	FALSE	reserved
+41.7	PARA	BOOL	FALSE	Module parametrized (read user)
+42.0	E_BYTE_0	BYTE	B#16#0	reserved
+43.0	STS_RUN	BOOL	FALSE	Status counter is running (read user)
+43.1	STS_DIR	BOOL	FALSE	Status of the counter direction bit (read user)
+43.2	STS_ZERO	BOOL	FALSE	Status of the counter zero crossing bit (read user)
+43.3	STS_OFLW	BOOL	FALSE	Status counter passed the overflow value (read user)
+43.4	STS_UFLW	BOOL	FALSE	Status counter passed the underflow value (read user)
+43.5	STS_SYNC	BOOL	FALSE	Status counter is synchronized (read user)
+43.6	STS_GATE	BOOL	FALSE	Status of the internal gate (read user)
+43.7	STS_SW_G	BOOL	FALSE	Status of the software gate (read user)
+44.0	STS_SET	BOOL	FALSE	Status digital input SET (read user)
+44.1	STS_LATCH	BOOL	FALSE	New latch value in isochronous mode (read user)
+44.2	STS_STA	BOOL	FALSE	Status of the digital input start (read user)
+44.3	STS_STP	BOOL	FALSE	Status of the digital input stop (read user)
+44.4	STS_CMP1	BOOL	FALSE	Status of the comparator output 1 (read user)
+44.5	STS_CMP2	BOOL	FALSE	Status of the comparator output 2 (read user)

Page 2 of 4

Address	Name	Type	Initial value	Comment
+44.6	STS_COMP1	BOOL	FALSE	Remanent status of the comparator output 1 (read user)
+44.7	STS_COMP2	BOOL	FALSE	Remanent status of the comparator output 2 (read user)
+45.0	E_BIT3_0	BOOL	FALSE	reserved
+45.1	E_BIT3_1	BOOL	FALSE	reserved
+45.2	E_BIT3_2	BOOL	FALSE	reserved
+45.3	E_BIT3_3	BOOL	FALSE	reserved
+45.4	E_BIT3_4	BOOL	FALSE	reserved
+45.5	E_BIT3_5	BOOL	FALSE	reserved
+45.6	E_BIT3_6	BOOL	FALSE	reserved
+45.7	E_BIT3_7	BOOL	FALSE	reserved
+46.0	ACT_CMP1	DINT	L#0	Actual comparator value 1 (FM 450 only; read user)
+50.0	ACT_CMP2	DINT	L#0	Actual comparator value 2 (FM 450 only; read user)
+54.0	MDL_DEFECT	BOOL	FALSE	Module defective
+54.1	INT_FAULT	BOOL	FALSE	Internal fault
+54.2	EXT_FAULT	BOOL	FALSE	External fault
+54.3	PNT_INFO	BOOL	FALSE	Point information
+54.4	EXT_VOLTAGE	BOOL	FALSE	External voltage low
+54.5	FLD_CONNCTR	BOOL	FALSE	Field wiring connector missing
+54.6	NO_CONFIG	BOOL	FALSE	Module has no configuration data
+54.7	CONFIG_ERR	BOOL	FALSE	Module has configuration error
+55.0	MDL_TYPE	BYTE	B#16#0	Type of module
+56.0	SUB_MDL_ERR	BOOL	FALSE	Sub-Module is missing or has error
+56.1	COMM_FAULT	BOOL	FALSE	Communication fault
+56.2	MDL_STOP	BOOL	FALSE	Module is stopped
+56.3	WTCH_DOG_FLT	BOOL	FALSE	Watch dog timer stopped module
+56.4	INT_PS_FLT	BOOL	FALSE	Internal power supply fault
+56.5	PRIM_BATT_FLT	BOOL	FALSE	Primary battery is in fault
+56.6	BCKUP_BATT_FLT	BOOL	FALSE	Backup battery is in fault
+56.7	RESERVED_2	BOOL	FALSE	reserved for system
+57.0	RACK_FLT	BOOL	FALSE	Rack fault, only for bus interface module
+57.1	PROC_FLT	BOOL	FALSE	Processor fault
+57.2	EPROM_FLT	BOOL	FALSE	EPROM fault
+57.3	RAM_FLT	BOOL	FALSE	RAM fault
+57.4	ADU_FLT	BOOL	FALSE	ADU fault
+57.5	FUSE_FLT	BOOL	FALSE	Fuse fault
+57.6	HW_INTR_FLT	BOOL	FALSE	Hardware interrupt input in fault
+57.7	RESERVED_3	BOOL	FALSE	reserved for system
+58.0	CH_TYPE	BYTE	B#16#0	Channel type
+59.0	LGTH_DIA	BYTE	B#16#0	Length of diagnostics data per channel
+60.0	CH_NO	BYTE	B#16#0	Channel number (numero)
+61.0	GRP_ERR1	BOOL	FALSE	Group error channel 1

(Anexo M) 152

Page 3 of 4

Address	Name	Type	Initial value	Comment
+61.1	GRP_ERR2	BOOL	FALSE	Group error channel 2
+61.2	D_BIT7_2	BOOL	FALSE	DS1 byte 7 bit 2
+61.3	D_BIT7_3	BOOL	FALSE	DS1 byte 7 bit 3
+61.4	D_BIT7_4	BOOL	FALSE	DS1 byte 7 bit 4
+61.5	D_BIT7_5	BOOL	FALSE	DS1 byte 7 bit 5
+61.6	D_BIT7_6	BOOL	FALSE	DS1 byte 7 bit 6
+61.7	D_BIT7_7	BOOL	FALSE	DS1 byte 7 bit 7
+62.0	CH1_SIGA	BOOL	FALSE	Channel 1, signal A malfunction
+62.1	CH1_SIGB	BOOL	FALSE	Channel 1, signal B malfunction
+62.2	CH1_SIGZ	BOOL	FALSE	Channel 1, signal Zero mark malfunction
+62.3	CH1_BETW	BOOL	FALSE	Channel 1, malfunction between channels
+62.4	CH1_5V2	BOOL	FALSE	Channel 1, malfunction of encoder supply voltage 5.2V
+62.5	D_BIT8_5	BOOL	FALSE	DS1 byte 8 bit 5
+62.6	D_BIT8_6	BOOL	FALSE	DS1 byte 8 bit 6
+62.7	D_BIT8_7	BOOL	FALSE	DS1 byte 8 bit 7
+63.0	D_BYTE9	BYTE	B#16#0	DS1 byte 9
+64.0	CH2_SIGA	BOOL	FALSE	Channel 2, signal A malfunction
+64.1	CH2_SIGB	BOOL	FALSE	Channel 2, signal B malfunction
+64.2	CH2_SIGZ	BOOL	FALSE	Channel 2, signal Zero mark malfunction
+64.3	CH2_BETW	BOOL	FALSE	Channel 2, malfunction between channels
+64.4	CH2_5V2	BOOL	FALSE	Channel 2, malfunction of encoder supply voltage 5.2V
+64.5	D_BIT10_5	BOOL	FALSE	DS1 byte 10 bit 5
+64.6	D_BIT10_6	BOOL	FALSE	DS1 byte 10 bit 6
+64.7	D_BIT10_7	BOOL	FALSE	DS1 byte 10 bit 7
+65.0	D_BYTE11	BYTE	B#16#0	DS1 byte 11
+66.0	D_BYTE12	BYTE	B#16#0	DS1 byte 12
+67.0	D_BYTE13	BYTE	B#16#0	DS1 byte 13
+68.0	D_BYTE14	BYTE	B#16#0	DS1 byte 14
+69.0	D_BYTE15	BYTE	B#16#0	DS1 byte 15
=70.0		END_STRUCT		

SFC6 - <offline>

"RD_SINFO" Read OB Start Information

Name: RD_SINFO Family: DB_FUNCT

Author: SIMATIC Version: 1.0

Block version: 2

Time stamp Code: 12/13/1995 05:11:46 PM

Interface: 12/13/1995 05:11:46 PM

Lengths (block/logic/data): 00132 00002 00000

Name	Data Type	Address	Comment
IN		0.0	
OUT		0.0	
TOP_SI	Struct	2.0	
EV_CLASS	Byte	2.0	
EV_NUM	Byte	3.0	
PRIORITY	Byte	4.0	
NUM	Byte	5.0	
TYP2_3	Byte	6.0	
TYP1	Byte	7.0	
ZI1	Word	8.0	
ZI2_3	DWord	10.0	
START_UP_SI	Struct	14.0	
EV_CLASS	Byte	14.0	
EV_NUM	Byte	15.0	
PRIORITY	Byte	16.0	
NUM	Byte	17.0	
TYP2_3	Byte	18.0	
TYP1	Byte	19.0	
ZI1	Word	20.0	
ZI2_3	DWord	22.0	
IN_OUT		0.0	

Block: SFC6

Page 2 of 2

SFC51 - <offline>

"RDSYSST" Read a System Status List or Partial List

Name: RDSYSST **Family:** DIAGNSTC
Author: SIMATIC **Version:** 1.0
 Block version: 2
Time stamp Code: 11/02/1994 11:19:56 AM
 Interface: 11/02/1994 11:19:56 AM
Lengths (block/logic/data): 00110 00002 00000

Name	Data Type	Address	Comment
IN		0.0	
REQ	Bool	0.0	
SZL_ID	Word	2.0	
INDEX	Word	4.0	
OUT		0.0	
BUSY	Bool	8.0	
SZL_HEADER	Struct	10.0	
LENTHDR	Word	10.0	
N_DR	Word	12.0	
DR	Any	14.0	
IN_OUT		0.0	

Block: SFC51

SFC58 - <offline>

"WR_REC" Write Data Record
Name: WR_REC **Family:** IO_FUNCT
Author: SIMATIC **Version:** 1.0
Block version: 2
Time stamp Code: 11/02/1994 11:20:44 AM
Interface: 11/02/1994 11:20:44 AM
Lengths (block/logic/data): 00106 00002 00000

Name	Data Type	Address	Comment
IN		0.0	
REQ	Bool	0.0	
IOID	Byte	1.0	
LADDR	Word	2.0	
RECNUM	Byte	4.0	
RECORD	Any	6.0	
OUT		0.0	
BUSY	Bool	18.0	
IN_OUT		0.0	

Block: SFC58

Page 1 of 1

SFC59 - <offline>

"RD_REC" Read a Data Record
Name: RD_REC **Family:** IO_FUNCT
Author: SIMATIC **Version:** 1.0
Block version: 2
Time stamp Code: 11/02/1994 11:20:54 AM
Interface: 11/02/1994 11:20:54 AM
Lengths (block/logic/data): 00106 00002 00000

Name	Data Type	Address	Comment
IN		0.0	
REQ	Bool	0.0	
IOID	Byte	1.0	
LADDR	Word	2.0	
RECNUM	Byte	4.0	
OUT		0.0	
BUSY	Bool	8.0	
RECORD	Any	10.0	
IN_OUT		0.0	

Block: SFC59

Tópico 5: Programação – Arduino

```

1 // %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
2 // Objectivo: otimizar a busca da posição óptima do sistema principal (PLC)
3 // Conexão: entre Arduino e PLC Siemens
4 // Utiliza: 4 conversores irradiância-frequência (TLS230R)
5 // Função: disponibilizar um sinal analógico (PWM) (cada sensor), indicador
6 // da irradiância solar
7 // Com base no projecto de Paul Christen (home.teampaulc.org)
8 // Eliseu Fernandes
9 // %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
10
11 // INÍCIO DE DECLARAÇÃO DE VARIÁVEIS %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
12 #define READ_TM 1000 // milisegundos (cálculo de frequência)
13 // Correspondência entre pinos do Arduino e pinos dos sensores (integrados)
14 // Sensor A
15 #define TSL_S0_A      22
16 #define TSL_S1_A      24
17 #define TSL_S2_A      26
18 #define TSL_S3_A      28
19 // Correspondência entre a saída do sensor e um pino do Arduino do tipo
20 // entrada digital, utilizado como interruptor
21 #define TSL_FREQ_PIN_A  2
22 // Pino do Arduino utilizado como saída analógica
23 const int analogOutPin = 4;
24 // Variável que recebe o valor da saída do Arduino (tipo PWM)
25 int outputValue = 0;
26 // Sensor B
27 #define TSL_S0_B      30
28 #define TSL_S1_B      31
29 #define TSL_S2_B      32
30 #define TSL_S3_B      33
31 #define TSL_FREQ_PIN_B  3
32 // Sensor C
33 #define TSL_S0_C      34
34 #define TSL_S1_C      35
35 #define TSL_S2_C      36
36 #define TSL_S3_C      37
37 #define TSL_FREQ_PIN_C 18
38 // Sensor D
39 #define TSL_S0_D      38
40 #define TSL_S1_D      39
41 #define TSL_S2_D      40
42 #define TSL_S3_D      41
43 #define TSL_FREQ_PIN_D 19
44 // Valor do modo de sensibilidade (10, 100 ou 1000)
45 int calcSensitivity_A;
46 int calcSensitivity_B;
47 int calcSensitivity_C;
48 int calcSensitivity_D;
49 // valores de irradiância - limites para modo de sensibilidade
50 unsigned long sensitivityHighThresh = 2000;
51 unsigned long sensitivityLowThresh = 100000;
52 // Sensor A
53 // Contador de pulsos
54 unsigned long pulseCount_A = 0;
55 // Returns the number of milliseconds since the Arduino board began running
56 // the current program.
57 unsigned long currentTime_A = millis();
58 unsigned long startTime_A = currentTime_A;
59 // Sensor B
60 unsigned long pulseCount_B = 0;

```



```

61 unsigned long currentTime_B = millis();
62 unsigned long startTime_B = currentTime_B;
63 // Sensor C
64 unsigned long pulseCount_C = 0;
65 unsigned long currentTime_C = millis();
66 unsigned long startTime_C = currentTime_C;
67 // Sensor D
68 unsigned long pulseCount_D = 0;
69 unsigned long currentTime_D = millis();
70 unsigned long startTime_D = currentTime_D;
71 // VARIÁVEIS: FREQUÊNCIA E IRRADIÂNCIA
72 // freq will be modified by the interrupt handler so needs to be volatile
73 // freq holds the latest frequency calculation
74 unsigned long frequency_A; // frequency = curPulseCount * scale
75 unsigned long frequency_B;
76 unsigned long frequency_C;
77 unsigned long frequency_D;
78 float uWattCm2_A; // irradiância = frequency / calcSensitivity (10, 100 ou 1000)
79 float uWattCm2_B;
80 float uWattCm2_C;
81 float uWattCm2_D;
82 // A variable should be declared VOLATILE whenever its value can be changed
83 // by something beyond the control of the code section in which it appears,
84 // such as a concurrently executing thread. In the Arduino, the only place that
85 // this is likely to occur is in sections of code associated with interrupts,
86 // called an interrupt service routine.
87 volatile unsigned long curPulseCount_A; // armazena o valor da contagem de pulsos
88 volatile unsigned long curPulseCount_B;
89 volatile unsigned long curPulseCount_C;
90 volatile unsigned long curPulseCount_D;
91 unsigned int scale; // armazena o valor da escala do integrado
92 // FIM DE DECLARAÇÃO DE VARIÁVEIS %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
93
94 void setup() {
95   Serial.begin(57600);
96   // Sensor A
97   sensitivity_A(1);
98   Serial.print("Sensitivity(A) ");
99   Serial.println(calcSensitivity_A, DEC);
100  Serial.println("Start(A)...");
101  // Specifies a function to call when an external interrupt occurs.
102  // Replaces any previous function that was attached to the interrupt.
103  // Attach interrupt to pin (analog output), send output pin of TSL230R to arduino 2
104  // call handler on each rising pulse
105  // Arduino Mega: 0 (on digital pin 2); 1 (on digital pin 3);
106  // 2 (pin 21), 3 (pin 20), 4 (pin 19), and 5 (pin 18)
107  // Syntax: attachInterrupt(interrupt, function, mode)
108  // Quando existe uma passagem do valor lógico do "pino 0" de "0" para "1"
109  // ocorrer um disparo da interrupção e conseqüentemente da função de contagem de
110  // pulsos
111  attachInterrupt(0, add_pulse_A, RISING);
112  // atribuição da função dos pinos do Arduino
113  pinMode(TSL_FREQ_PIN_A, INPUT);
114  pinMode(TSL_S0_A, OUTPUT);
115  pinMode(TSL_S1_A, OUTPUT);
116  pinMode(TSL_S2_A, OUTPUT);
117  pinMode(TSL_S3_A, OUTPUT);
118  // S2 and S3 HIGH = /100 output scaling
119  digitalWrite(TSL_S2_A, HIGH);
120  digitalWrite(TSL_S3_A, HIGH);

```

```

120 scale = 100; // set this to match TSL_S2 and TSL_S3
121 // Sensor B
122 sensitivity_B(1);
123 Serial.print("Sensitivity(B) ");
124 Serial.println(calcSensitivity_B, DEC);
125 Serial.println("Start(B)...");
126 attachInterrupt(1, add_pulse_B, RISING);
127 pinMode(TSL_FREQ_PIN_B, INPUT);
128 pinMode(TSL_S0_B, OUTPUT);
129 pinMode(TSL_S1_B, OUTPUT);
130 pinMode(TSL_S2_B, OUTPUT);
131 pinMode(TSL_S3_B, OUTPUT);
132 digitalWrite(TSL_S2_B, HIGH);
133 digitalWrite(TSL_S3_B, HIGH);
134 // Sensor C
135 sensitivity_C(1);
136 Serial.print("Sensitivity(C) ");
137 Serial.println(calcSensitivity_C, DEC);
138 Serial.println("Start(C)...");
139 attachInterrupt(5, add_pulse_C, RISING);
140 pinMode(TSL_FREQ_PIN_C, INPUT);
141 pinMode(TSL_S0_C, OUTPUT);
142 pinMode(TSL_S1_C, OUTPUT);
143 pinMode(TSL_S2_C, OUTPUT);
144 pinMode(TSL_S3_C, OUTPUT);
145 digitalWrite(TSL_S2_C, HIGH);
146 digitalWrite(TSL_S3_C, HIGH);
147 // Sensor D
148 sensitivity_D(1);
149 Serial.print("Sensitivity(D) ");
150 Serial.println(calcSensitivity_D, DEC);
151 Serial.println("Start(D)...");
152 attachInterrupt(4, add_pulse_D, RISING);
153 pinMode(TSL_FREQ_PIN_D, INPUT);
154 pinMode(TSL_S0_D, OUTPUT);
155 pinMode(TSL_S1_D, OUTPUT);
156 pinMode(TSL_S2_D, OUTPUT);
157 pinMode(TSL_S3_D, OUTPUT);
158 digitalWrite(TSL_S2_D, HIGH);
159 digitalWrite(TSL_S3_D, HIGH);
160 }
161 // FIM - void setup() %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
162 void loop() {
163 // %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
164 // this is just for debugging
165 // it shows that you can sample freq whenever you need it
166 // even though it is calculated once a second
167 // NOTE that the first time that freq is calculated, it is bogus
168 // %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
169 // map it to the range of the analog out:
170
171 if ( uWattCm2_A > uWattCm2_C && uWattCm2_A > uWattCm2_D && uWattCm2_B > uWattCm2_C &&
uWattCm2_B > uWattCm2_D) {
172     outputValue = 102; // busca 1
173 } else if ( uWattCm2_B > uWattCm2_A && uWattCm2_B > uWattCm2_D && uWattCm2_C >
uWattCm2_A && uWattCm2_C > uWattCm2_D) {
174     outputValue = 128; // busca 2
175 } else if ( uWattCm2_C > uWattCm2_A && uWattCm2_C > uWattCm2_B && uWattCm2_D >
uWattCm2_A && uWattCm2_D > uWattCm2_B) {
176     outputValue = 153; // busca 3

```

```

177 } else if ( uWattCm2_D > uWattCm2_B && uWattCm2_D > uWattCm2_C && uWattCm2_A >
uWattCm2_B && uWattCm2_A > uWattCm2_C) {
178     outputValue = 179; // busca 4
179 } else if ( uWattCm2_A > uWattCm2_B && uWattCm2_A > uWattCm2_D && uWattCm2_C >
uWattCm2_B && uWattCm2_C > uWattCm2_D) {
180     outputValue = 204; // busca 5
181 } else {
182     outputValue = 255; // busca 7
183 }
184
185 // change the analog out value:
186 analogWrite(analogOutPin, outputValue);
187 // analogWrite(analogOutPin_B, outputValue_B);
188 // analogWrite(analogOutPin_C, outputValue_C);
189 // analogWrite(analogOutPin_D, outputValue_D);
190 // print the results to the serial monitor:
191 Serial.print("\t\t\t\t\t");
192 Serial.println("\t\t\t\t\t");
193 Serial.print("\tPWM=");
194 Serial.print(outputValue);
195 Serial.print("\tuW/cm (A)=");
196 Serial.println(getUwattCm2_A(), DEC);
197 Serial.print("\tuW/cm (B)=");
198 Serial.println(getUwattCm2_B(), DEC);
199 Serial.print("\tuW/cm (C)=");
200 Serial.println(getUwattCm2_C(), DEC);
201 Serial.print("\tuW/cm (D)=");
202 Serial.println(getUwattCm2_D(), DEC);
203 // Chama as funções onde se verifica a necessidade de alterar o
204 // modo de sensibilidade de acordo como o valor de irradiância
205 setSensitivity_A();
206 setSensitivity_B();
207 setSensitivity_C();
208 setSensitivity_D();
209 // Pausa implementada para ser possível ler os valores que surgem no ecrã
210 delay(500);
211 }
212 // Fim - void loop () %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
213 // Início da função para contagem de pulsos: Sensor A
214 void add_pulse_A() {
215     pulseCount_A++; // increase pulse count
216     // DON'T calculate the frequency every READ_TM ms (1s)
217     // just store the pulse count to be used outside of the interrupt
218     currentTime_A = millis();
219     if( currentTime_A - startTime_A >= READ_TM )
220     {
221         curPulseCount_A = pulseCount_A; // use curPulseCount for calculating freq/uW
222         pulseCount_A = 0;
223         startTime_A = millis();
224     }
225 }
226 // Início da função para contagem de pulsos: Sensor B
227 void add_pulse_B() {
228     pulseCount_B++;
229     currentTime_B = millis();
230     if( currentTime_B - startTime_B >= READ_TM )
231     {
232         curPulseCount_B = pulseCount_B;
233         pulseCount_B = 0;
234         startTime_B = millis();

```

```

235     }
236 }
237 // Início da função para contagem de pulsos: Sensor C
238 void add_pulse_C() {
239     pulseCount_C++;
240     currentTime_C = millis();
241     if( currentTime_C - startTime_C >= READ_TM )
242     {
243         curPulseCount_C = pulseCount_C;
244         pulseCount_C = 0;
245         startTime_C = millis();
246     }
247 }
248 // Início da função para contagem de pulsos: Sensor D
249 void add_pulse_D() {
250     pulseCount_D++;
251     currentTime_D = millis();
252     if( currentTime_D - startTime_D >= READ_TM )
253     {
254         curPulseCount_D = pulseCount_D;
255         pulseCount_D = 0;
256         startTime_D = millis();
257     }
258 }
259 // Início da função para calcular a frequência e a irradiância: Sensor A
260 long getUwattCm2_A() {
261     // copy pulse counter and multiply.
262     // the multiplication is necessary for the current frequency scaling level.
263     frequency_A = curPulseCount_A * scale;
264     // get uW observed - assume 640nm wavelength
265     // calc_sensitivity is our divide-by to map to a given signal strength
266     // for a given sensitivity (each level of greater sensitivity reduces the signal
267     // (uW) by a factor of 10)
268     float uw_cm2_A = (float) frequency_A / (float) calcSensitivity_A;
269     // extrapolate into entire cm2 area
270     uWattCm2_A = uw_cm2_A * ( (float) 1 / (float) 0.0136 );
271     return(uWattCm2_A);
272 }
273 // Início da função para calcular a frequência e a irradiância: Sensor B
274 long getUwattCm2_B() {
275     frequency_B = curPulseCount_B * scale;
276     float uw_cm2_B = (float) frequency_B / (float) calcSensitivity_B;
277     uWattCm2_B = uw_cm2_B * ( (float) 1 / (float) 0.0136 );
278     return(uWattCm2_B);
279 }
280 // Início da função para calcular a frequência e a irradiância: Sensor C
281 long getUwattCm2_C() {
282     frequency_C = curPulseCount_C * scale;
283     float uw_cm2_C = (float) frequency_C / (float) calcSensitivity_C;
284     uWattCm2_C = uw_cm2_C * ( (float) 1 / (float) 0.0136 );
285     return(uWattCm2_C);
286 }
287 // Início da função para calcular a frequência e a irradiância: Sensor D
288 long getUwattCm2_D() {
289     frequency_D = curPulseCount_D * scale;
290     float uw_cm2_D = (float) frequency_D / (float) calcSensitivity_D;
291     uWattCm2_D = uw_cm2_D * ( (float) 1 / (float) 0.0136 );
292     return(uWattCm2_D);
293 }
294 // Início da função setSensivity Sensor A %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

```

295 // Verifica se o valor de irradiância se adequa ao modo de sensibilidade
296 void setSensitivity_A()
297 {
298     getUwattCm2_A();
299     if (uWattCm2_A < sensitivityHighThresh)
300     {
301         sensitivity_A(3);
302         return;
303     }
304     if (uWattCm2_A > sensitivityLowThresh )
305     {
306         sensitivity_A(1);
307         return;
308     }
309     sensitivity_A(2);
310 }
311 // Início da função setSensitivity Sensor B
312 void setSensitivity_B()
313 {
314     getUwattCm2_B();
315     if (uWattCm2_B < sensitivityHighThresh)
316     {
317         sensitivity_B(3);
318         return;
319     }
320     if (uWattCm2_B > sensitivityLowThresh )
321     {
322         sensitivity_B(1);
323         return;
324     }
325     sensitivity_B(2);
326 }
327 // Início da função setSensitivity Sensor C
328 void setSensitivity_C()
329 {
330     getUwattCm2_C();
331     if (uWattCm2_C < sensitivityHighThresh)
332     {
333         sensitivity_C(3);
334         return;
335     }
336     if (uWattCm2_C > sensitivityLowThresh )
337     {
338         sensitivity_C(1);
339         return;
340     }
341     sensitivity_C(2);
342 }
343 // Início da função setSensitivity Sensor D
344 void setSensitivity_D()
345 {
346     getUwattCm2_D();
347     if (uWattCm2_D < sensitivityHighThresh)
348     {
349         sensitivity_D(3);
350         return;
351     }
352     if (uWattCm2_D > sensitivityLowThresh )
353     {
354         sensitivity_D(1);

```

```

355     return;
356 }
357 sensitivity_D(2);
358 }
359 // Fim da função setSensitivity Sensor D
360 // Início da função sensitivity Sensor A%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
361 // Alteração do modo de sensibilidade (10=1x, 100=10x, 1000=100x)
362 void sensitivity_A(uint8_t level_A)
363 {
364     switch (level_A)
365     {
366     case 1:
367         if (calcSensitivity_A != 10)
368         {
369             Serial.println("Now at low sensitivity (A).");
370         }
371         digitalWrite(TSL_S0_A, HIGH); // S0 HIGH and S1 LOW = 1x sensitivity
372         digitalWrite(TSL_S1_A, LOW);
373         calcSensitivity_A = 10;
374         break;
375     case 2:
376         if (calcSensitivity_A != 100)
377         {
378             Serial.println("Now at medim sensitivity (A).");
379         }
380         digitalWrite(TSL_S0_A, LOW); // S0 LOW and S1 HIGH = 10x sensitivity
381         digitalWrite(TSL_S1_A, HIGH);
382         calcSensitivity_A = 100;
383         break;
384     case 3:
385         if (calcSensitivity_A != 1000)
386         {
387             Serial.println("Now at high sensitivity (A).");
388         }
389         digitalWrite(TSL_S0_A, HIGH); // S0 HIGH and S1 HIGH = 100x sensitivity
390         digitalWrite(TSL_S1_A, HIGH);
391         calcSensitivity_A = 1000;
392         break;
393     }
394     return;
395 }
396 // Início da função sensitivity Sensor B%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
397 void sensitivity_B(uint8_t level_B)
398 {
399     switch (level_B)
400     {
401     case 1:
402         if (calcSensitivity_B != 10)
403         {
404             Serial.println("Now at low sensitivity (B).");
405         }
406         digitalWrite(TSL_S0_B, HIGH);
407         digitalWrite(TSL_S1_B, LOW);
408         calcSensitivity_B = 10;
409         break;
410     case 2:
411         if (calcSensitivity_B != 100)
412         {
413             Serial.println("Now at medim sensitivity (B).");
414         }

```

```

415     digitalWrite(TSL_S0_B, LOW);
416     digitalWrite(TSL_S1_B, HIGH);
417     calcSensitivity_B = 100;
418     break;
419     case 3:
420         if (calcSensitivity_B != 1000)
421         {
422             Serial.println("Now at high sensitivity (B).");
423         }
424         digitalWrite(TSL_S0_B, HIGH);
425         digitalWrite(TSL_S1_B, HIGH);
426         calcSensitivity_B = 1000;
427         break;
428     }
429     return;
430 }
431 // Início da função sensivity Sensor C%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
432 void sensitivity_C(uint8_t level_C)
433 {
434     switch (level_C)
435     {
436         case 1:
437             if (calcSensitivity_C != 10)
438             {
439                 Serial.println("Now at low sensitivity (C).");
440             }
441             digitalWrite(TSL_S0_C, HIGH);
442             digitalWrite(TSL_S1_C, LOW);
443             calcSensitivity_C = 10;
444             break;
445         case 2:
446             if (calcSensitivity_C != 100)
447             {
448                 Serial.println("Now at medim sensitivity (C).");
449             }
450             digitalWrite(TSL_S0_C, LOW);
451             digitalWrite(TSL_S1_C, HIGH);
452             calcSensitivity_C = 100;
453             break;
454         case 3:
455             if (calcSensitivity_C != 1000)
456             {
457                 Serial.println("Now at high sensitivity (C).");
458             }
459             digitalWrite(TSL_S0_C, HIGH);
460             digitalWrite(TSL_S1_C, HIGH);
461             calcSensitivity_C = 1000;
462             break;
463     }
464     return;
465 }
466 // Início da função sensivity Sensor D %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
467 void sensitivity_D(uint8_t level_D)
468 {
469     switch (level_D)
470     {
471         case 1:
472             if (calcSensitivity_D != 10)
473             {

```

```

475     }
476     digitalWrite(TSL_S0_D, HIGH);
477     digitalWrite(TSL_S1_D, LOW);
478     calcSensitivity_D = 10;
479     break;
480 case 2:
481     if (calcSensitivity_D != 100)
482     {
483         Serial.println("Now at medim sensitivity (D).");
484     }
485     digitalWrite(TSL_S0_D, LOW);
486     digitalWrite(TSL_S1_D, HIGH);
487     calcSensitivity_D = 100;
488     break;
489 case 3:
490     if (calcSensitivity_D != 1000)
491     {
492         Serial.println("Now at high sensitivity (D).");
493     }
494     digitalWrite(TSL_S0_D, HIGH);
495     digitalWrite(TSL_S1_D, HIGH);
496     calcSensitivity_D = 1000;
497     break;
498 }
499 return;
500 }

```