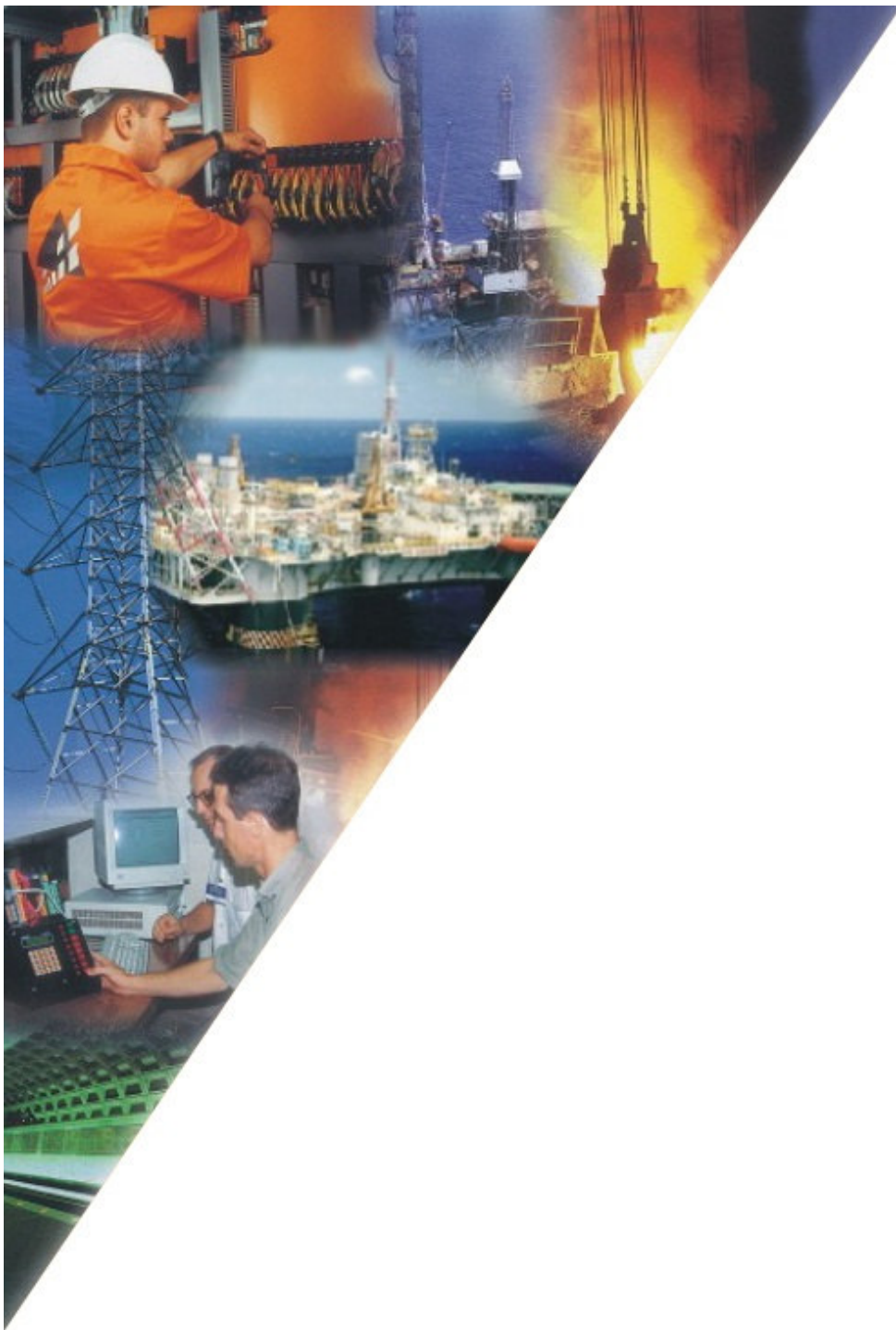




Treinamento DUO

Conhecendo o DUO e o MasterTool IEC
Carlos Santos e Rafael Morello
Setembro/2008
Rev. A



altus

evolução em automação



Cronograma

- 8:30 as 10:00
 - DUO - Recursos e características
- 10:00 as 10:15
 - Coffe Break
- 10:15 as 12:00
 - DUO – Configurando o DUO no MasterTool IEC
- 13:00 as 15:00
 - MasterTool IEC – Conhecendo a interface
- 15:00 as 15:15
 - Coffe Break
- 15:15 as 17:40
 - MasterTool IEC – Programação para DUO



Introdução

- DUO
 - Recursos
 - Configuração
- MasterTool IEC
 - Interface
 - Estrutura
 - Programação



Índice - DUO

- Recursos
 - Comunicação
 - Entradas digitais
 - Entradas rápidas
 - Entradas analógicas
 - Saídas digitais
 - Saídas rápidas
 - Saídas analógicas
 - Características gerais



Comunicação

- RS 232
 - Comunicação com MasterTool IEC
 - MODBUS RTU
 - Mestre
 - Escravo
 - Velocidade
 - 4800 a 115200 bps
 - Sinais de modem
 - TXD, RXD, RTS, CTS, DTR, DSR, DCD
- RS 485
 - MODBUS RTU
 - Mestre
 - Escravo
 - Velocidade
 - 4800 a 115200 bps
 - Sem terminação interna

Obs.: As portas não são isoladas e os conectores deve ser RJ45



Comunicação

MODBUS RTU – Mestre

- Delay inter-frames
 - 5 ms a 1 s
- Timeout de resposta
 - 100 ms a 10s
- Retentativas
 - 0 a 10
- Relações
 - Máximo 16
 - Funções
 - Read Coils, Read Discrete Inputs, Read Holding Registers, Read Input Registers
 - Write Single Coil, Write Single Register, Write Multiple Coils, Write Multiple Registers
- Diagnóstico
 - Não aceita broadcast
 - Falha na tentativa
 - Endereço MODBUS não encontrado na tabela
 - Comando enviado não implementado
 - Recebimento de resposta de outro escravo
 - Erro de CRC na resposta



Comunicação

MODBUS RTU – Escravo

- Delay inter-frames
 - 5 ms a 1 s
- Coils
 - 1 a 1024 mapeadas em %QB0 a %QB127
 - 1025 a 2048 mapeadas em %MB5745 a %MB5872
- Inputs
 - 1 a 1024 mapeadas em %IB0 a %IB127
 - 1025 a 2048 mapeadas em %MB5873 a %MB6000
- Holding Registers
 - 1 a 2000 mapeados em %MW0 a %MW1999
- Inputing Registers
 - 1 a 870 mapeados em %MW2000 a %MW2869
- Diagnóstico
 - Comando recebino não implementado
 - Erro de CRC no pacote recebido
 - Pacote para outro escravo ou broadcast
 - Limites do comando inválido
 - Endereço MODBUS não encontrado na tabela
 - Endereço do host inválido



Comunicação

- Alternando protocolo da COM 1
 - CoDeSys ou MODBUS
 - Segurar tecla MAIN e pressionar a tecla DOWN
 - Tela de estado – aproximadamente 1 s
- Bootloader
 - COM 1
 - Segurar a tecla MAIN e a tecla LEFT
 - Energizar o DUO
- Parâmetros de configuração MasterTool IEC
 - Protocolo CoDeSys
 - Velocidade: 115200 bps
 - Paridade: sem paridade
 - Bits de parada: 1 bit de parada
 - Motorola byteorder: não
 - Controle de fluxo: não



Entradas Digitais

- 20 entradas digitais – “sink” tipo 1
 - 3 grupos de isolamento
 - I00 .. I08 – 9 entradas – Grupo 0
 - I10 .. I18 – 9 entradas – Grupo 1
 - I20 .. I21 – 2 entradas – Grupo 2
- Tensão de entrada
 - 11 a 30 Vdc em relação ao comum para estado 1
 - 0 a 5 Vdc em relação ao comum para estado 0
- Corrente de entrada
 - 5 mA – 24 Vdc em relação ao comum
- Impedância de entrada
 - 4,3 K Ω
- Isolação
 - 2000 Vac por um minuto entre cada grupo de entrada
- Tempo de resposta
 - 5 ms



Entradas Rápidas

- 6 entradas rápidas
 - 2 blocos de contagem
 - I00 .. I02 – Bloco 0
 - I10 .. I12 – Bloco 1
- Blocos de contagem
 - 2 entradas de contagem e 1 entrada configurável
 - 2 contadores unidirecionais ou 1 contador bidirecional
- Formato do dado
 - Inteiro 32 bits sem sinal
- Corrente de entrada
 - 15 mA – 24 Vdc em relação ao comum
- Impedância de entrada
 - 1,5 K Ω
- Tempo de resposta
 - 10 μ s



Entradas Rápidas

- Contador unidirecional
 - Progressivo ou regressivo
- Contador bidirecional
 - A incrementa, B decrementa
 - A contagem, B sentido
 - Quadratura 4x
 - Quadratura 2x
- Entrada configurável
 - Zeramento
 - Congelamento
 - Preset
 - Amostragem (HOLD)
- Frequências
 - Bloco 0
 - 20 KHz para Quadratura 4x
 - 40 KHz para demais modos
 - Bloco 1
 - 10 KHz para Quadratura 4x
 - 20 KHz para demais modos



Entradas Rápidas

- Formato do dado
 - Inteiro 32 bits sem sinal
- Registradores
 - Contagem
 - Preset
 - Hold
 - Comparador 0
 - Comparador 1
- Comandos
 - Zerar contagem
 - Desabilitar contagem
 - Carga – Preset
 - Amostrar – Hold
 - Limpar overflow/underflow
- Diagnóstico
 - Overflow e Underflow
 - Direção
 - Contador maior que registradores de comparação
 - Contador igual aos registradores de comparação
 - Contador igual 0



Entradas Rápidas

- Configuração bidirecional
 - Contador 0 e Contador 2
 - Utilizam entradas do outro contador
 - Contador 1 e Contador 3
 - Desabilitados
- Entradas configuráveis
 - Contador 0 e Contador 2
 - Comandos em software e hardware
 - Contador 1 e Contador 3
 - Comandos em software



Entradas Analógicas

- 4 entradas analógicas não isoladas
- Tensão
 - 0 a 10 Vdc
- Corrente
 - 0 a 20 mA
 - 4 a 20 mA
- Entradas diferentes para tensão e corrente
- Fundo de escala
 - 0 a 30.000
- Proteção com diodo
- Tempo de atualização
 - 60 ms



Entradas Analógicas

Tensão

- Filtros de primeira ordem
 - 90 ms, 140 ms, 1 s, 15 s
- Precisão
 - $\pm 0,3\%$ do fundo de escala @ 25°C
 - $\pm 0,015\%$ / °C do fundo de escala
- Impedância de entrada
 - 1,1 M Ω
- Tensão máxima e mínima
 - 12 Vdc e -0.3 Vdc
- Folga de escala
 - 3%
- Sensibilidade
 - 2,54 mVdc



Entradas Analógicas

Corrente

- Filtros de primeira ordem
 - 2 ms, 90 ms, 1 s, 15 s
- Precisão
 - $\pm 0,3\%$ do fundo de escala @ 25°C
 - $\pm 0,015\%$ / °C do fundo de escala
- Impedância de entrada
 - 124,5 Ω
- Corrente máxima e mínima
 - 25 mA e -2mA
- Folga de escala
 - 4%
- Sensibilidade
 - 5,18 μA
- Diagnóstico canal aberto
 - Corrente inferior a 4 mA
 - Configuração 4 – 20 mA



Saídas Digitais

- 14 saídas digitais
 - 2 grupos de isolamento
 - Q02 .. Q07 – 6 saídas – Grupo 0
 - Q10 .. Q17 – 8 saídas – Grupo 1
- Isolação
 - 2000 Vac por um minuto entre cada grupo de entrada
- Modelos de saída
 - DU350 – saídas a transistor
 - DU351 – saídas a relé



Saídas Digitais

DU350 – saídas a transistor – tipo “source”

- Corrente máxima por ponto
 - 0,5 A
- Tempo de comutação
 - 600 μ s
- Frequência máxima de chaveamento com carga
 - 250 Hz, com carga externa mínima de 12500 Ω
- Proteção com diodo
- Tensão de operação
 - 10 a 30 Vdc
- Impedância de saída
 - 500 m Ω



Saídas Digitais

DU351 – saídas a relé – tipo “NA”

- Corrente máxima por ponto
 - 1 A
- Tempo de comutação
 - 10 ms
- Frequência máxima de chaveamento com carga
 - 0,5 Hz máximo com carga nominal
- Vida útil esperada
 - 10×10^4 operações com carga nominal
- Tensão de operação
 - 30 Vdc – Grupo 0
 - 30 Vdc ou 240 Vac – Grupo 1
- Resistência de contato
 - $< 250 \text{ m}\Omega$



Saídas Rápidas

- 2 saídas rápidas – tipo “source”
 - Q00 .. Q01
- Corrente máxima por ponto
 - 0,5 A
- Frequência máxima de geração de pulsos
 - 50 KHz
- Proteção com diodo
- Tensão de operação
 - 10 a 30 Vdc
- Isolação
 - 2000 Vac por um minuto entre cada grupo de entrada
- Impedância de saída
 - 700 m Ω
- Modos de saída
 - PTO, VFO, PWM, Comparador



Saídas Rápidas

PTO

- Frequência em regime
 - 0 a 50 KHz
- Características
 - Pulsos totais
 - Pulsos em aceleração/desaceleração
 - Tipo trapezoidal e curva S
- Comandos
 - Iniciar
 - Parar
- Diagnósticos
 - Em operação
 - Acelerando
 - Regime
 - Desacelerando
 - Não parametrizada
 - Erro de parametrização
- Reconfiguração no final da desaceleração



Saídas Rápidas

VFO e PWM

- Freqüência
 - 0 a 50 KHz
- Duty Cycle
 - 0 a 100%
- Comandos
 - Ligado ou desligada
- Diagnósticos
 - Em operação
 - Não parametrizada
 - Erro de parametrização
- Reconfiguração no final do ciclo



Saídas Rápidas

Comparador

- Tipos de comparação
 - Contador $>$ Comparador 0
 - Contador $<$ Comparador 1
 - Comparador 0 $<$ Contador $<$ Comparador 1
- Saída digital
 - Q02 .. Q03
 - Saídas de comparação lentas



Saídas Analógicas

- 2 saídas analógicas não isoladas
- Tensão
 - 0 a 10 Vdc
- Corrente
 - 0 a 20 mA
- Resolução do conversor
 - 12 bits
- Fundo de escala
 - 0 a 30.000
- Diagnóstico
 - Curto circuito
 - Laço aberto



Saídas Analógicas

Tensão

- Precisão
 - $\pm 0,3\%$ do fundo de escala @ 25 °C
 - $\pm 0,015\%$ / °C do fundo de escala
- Impedância de saída
 - 22 Ω
- Impedância mínima de carga
 - 600 Ω
- Proteção contra curto circuito
- Tempo de atualização
 - 1 ms
- Tempo de estabilização
 - 4 ms
- Resolução
 - 2,59 mVdc
- Folga de escala
 - 4%
- Indicação de sobre carga



Saídas Analógicas

Corrente

- Precisão
 - $\pm 0,3\%$ do fundo de escala @ 25°C
 - $\pm 0,015\%$ / °C do fundo de escala
- Impedância de entrada
 - 600 Ω
- Tempo de atualização
 - 1 ms
- Tempo de estabilização
 - 4 ms
- Resolução
 - 5,17 μA
- Folga de escala
 - 4%
- Indicação de laço aberto



Características Gerais

- Visor gráficos
 - Monocromático
 - 128 x 64
 - Backlight
 - Ajuste de contraste
 - Operando reservados
 - Tempo de backlight
 - Contraste
- Teclado
 - 25 teclas de membrana
 - Teclas alfanumérico
 - Teclas de navegação
 - 7 teclas de funções customizáveis
- RTC
 - Autonomia de 15 dias
 - Resolução de 1 s



Características Gerais

- Tempo de inicialização
 - 10 s
- Cão de guarda
 - 2 s
- Telas de erro
 - 13 telas de erro grave
 - Retornar a matriz
- Estado seguro das saídas
 - Sempre que ocorrer um erro no DUO
 - Durante o download de aplicativo
 - Enquanto o DUO estiver no estado PARADO
 - Saídas digitais a transistor e rápidas em nível lógico 0
 - Saídas a relé com contato aberto
 - Saídas analógicas em 0 Vdc ou 0 mA



Características Gerais

- Telas especiais
 - Segurar a tecla MAIN e pressionar a tecla UP
 - Navegação utilizando as teclas UP e DOWN
 - Informações gerais
 - Modelo, versão de firmware, ..ATUALIZAR
 - Estado das entradas digitais
 - Estado das saídas digitais
 - Entradas e saídas analógicas
 - Valor dos contadores
- Teste de teclado
 - Janela de tempo durante a inicialização do DUO
 - Segurar a tecla MAIN e pressionar a tecla RIGHT
 - Tela mostrando a disposição do teclado
 - Preenche a caixa correspondente a tecla que está pressionada
- Inicializar sem aplicativo
 - Manter pressionada a tecla ESC durante a inicialização



Índice – MasterTool IEC

- Norma IEC 61131
- Linguagens
- Identificadores
- Tipo de dados
- Variáveis
- POU's
- Tarefas
- Configuração



Norma IEC 61131

- IEC 61131-3
 - A International Electrotechnical Commission (IEC) é uma organização interenacional, formada por representantes de todo mundo.
 - A norma IEC 61131 foi desenvolvida para atender as demandas da comunidade industrial.
 - Principais aspectos: Linguagens de Programação, Recursos Multitarefa e Reutilização de Software.
 - Um objetivo é portabilidade de software entre diferentes fabricantes.



Norma IEC 61131

- É dividida em 8 partes:
 - Parte 1: Informações Gerais
 - Parte 2: Requisitos de Equipamentos e Testes
 - **Parte 3: Linguagens de Programação**
 - Parte 4: Guia de Usuário
 - Parte 5: Comunicações
 - Parte 6: Reservada
 - Parte 7: Controle de Programação Fuzzy
 - Parte 8: Guia para Aplicação e Implementação das Linguagens de Programação



Linguagens

- Suporta todas linguagens do padrão IEC
 - Duas linguagens textuais
 - Instruction List - IL
 - Structured Text - ST
 - Três linguagens Gráficas
 - Sequential Function Chart - SFC
 - Function Block Diagram - FBD
 - Ladder Diagram - LD
 - ...uma adicional baseada em FBD
 - Continuous Function Chart - CFC



Linguagens

- Lista de Instruções- IL
 - É constituída por uma série de instruções. Semelhante a assembly.
 - EX:

```
LD      saida1
EQ      3
NOT
JMPC    else3_0

LD      TRUE      (* realiza um IF/ELSE em ST*)
ST      Q00
JMP     end3_0

else3_0:
LD      FALSE
ST      Q00

end3_0:
```



Linguagens

- Texto Estruturado - ST
 - É uma linguagem constituída por algumas palavras reservadas, muito semelhante as linguagens de programação de alto nível.
 - » IF, RETURN, FOR, WHILE, CASE...

– EX:

- **ST**

```
IF saida = 4 THEN
    Q00 := TRUE;

ELSE
    Q00 := FALSE;

END_IF
```

- **IL**

```
LD        saida1
EQ        4
NOT
JMPC      else4_0
LD        TRUE
ST        Q00
JMP       end4_0

else4_0:
LD        FALSE
ST        Q00

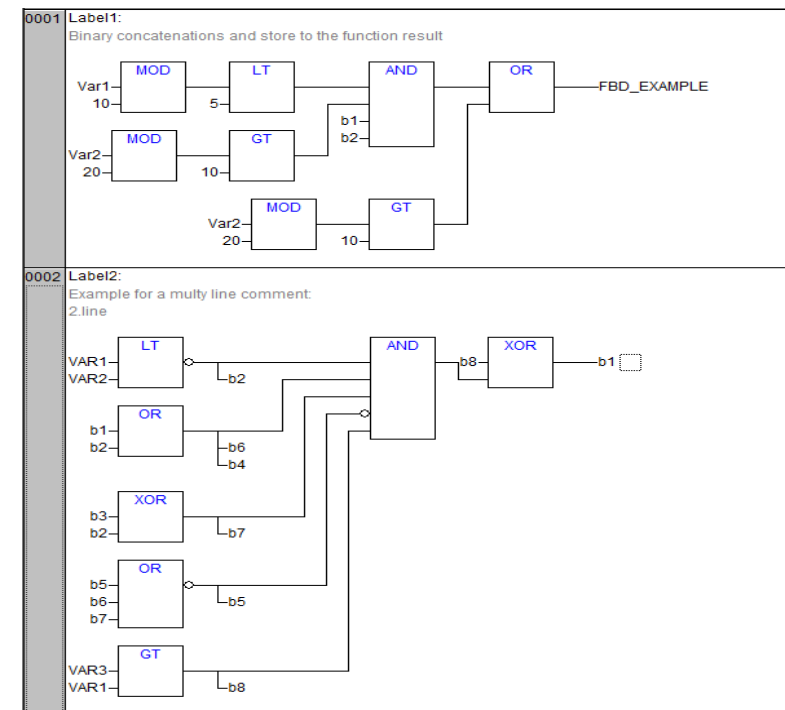
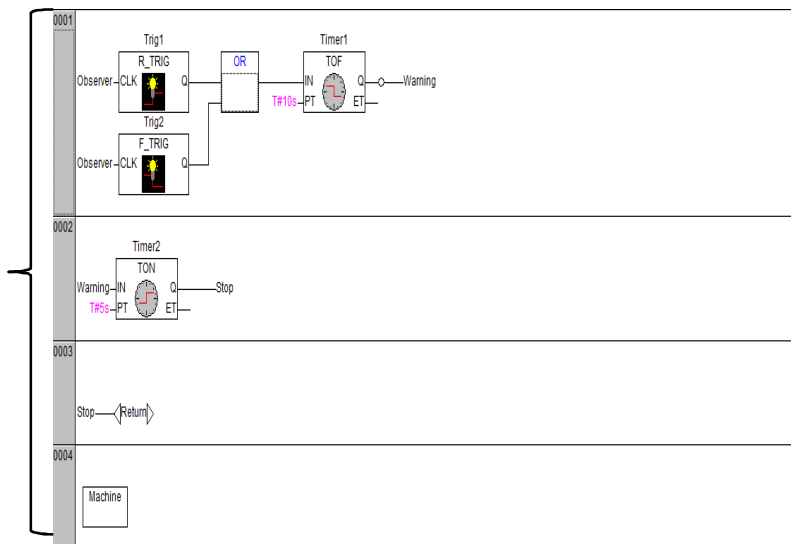
end4_0:
```



Linguagens

- Diagrama de Blocos Funcionais— FBD
 - É uma linguagem de programação gráfica. Pode-se formar uma lista de redes.
 - Cada rede pode ser formada por uma estrutura de blocos.
 - Cada bloco pode conter lógicas, expressões aritméticas, chamada de funções...
 - EX:

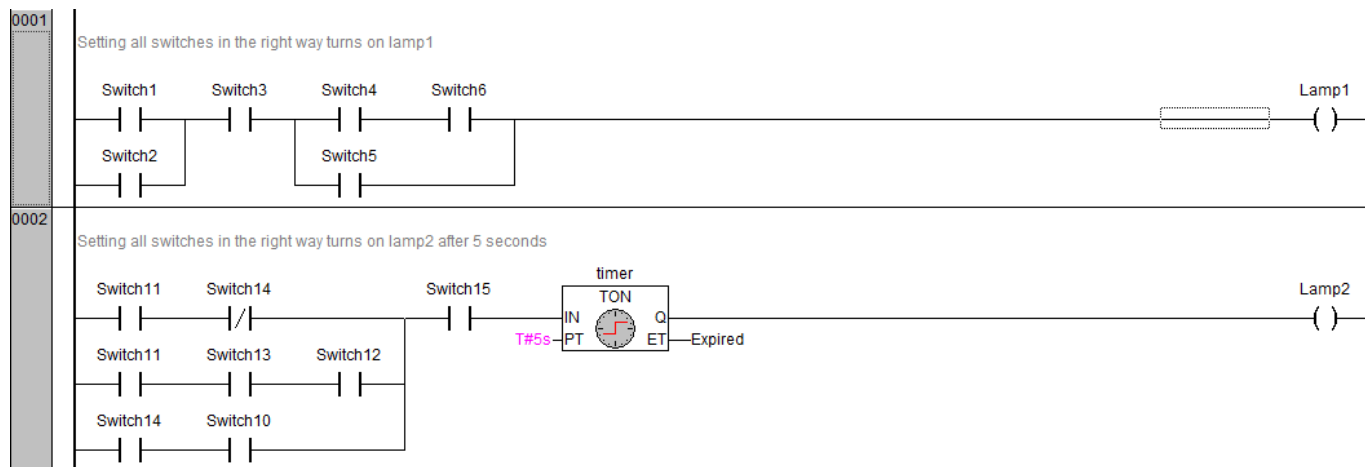
*Lista de
redes*





Linguagens

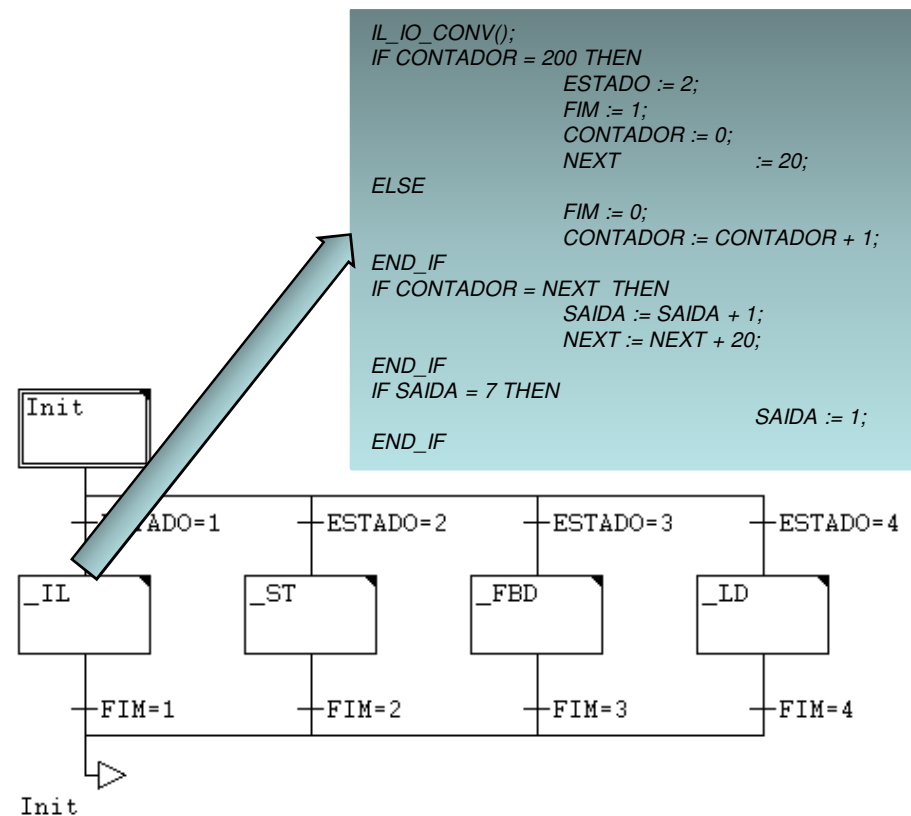
- Diagrama Ladder— LD
 - Também uma linguagem gráfica de programação.
 - Pode criar redes como no FBD.
 - EX:





Linguagens

- Seqüenciamento Gráfico de Funções - SFC
 - É uma linguagem gráfica que torna possível descrever uma ordem cronológica para diferentes ações dentro de um programa
 - EX:





Identificadores

- Os identificadores servem para dar nomes aos diferentes elementos de software, tais como variáveis, tipos de dados, POUs e tarefas.
- Regras a serem observadas:
 - O primeiro caractere não pode ser um dígito;
 - Caracteres underline “_” repetidos em sequência não são permitidos;
 - Identificador não pode incluir espaço;
 - Identificadores podem ter qualquer tamanho, pelo menos os 6 primeiros devem ser considerados para unicidade.



Tipos de Dados

- Para cada tipo de dado a norma define número de bits, faixa de valores, tamanho e forma de representação.
- Todo tipo de dado deve ser definido para informar ao compilador como alocar espaço de memória.



Tipos de Dados

- **Inteiros:**

Tipo IEC	Descrição	Bits	Faixa
SINT	Short Integer	8	-128 a 127
INT	Integer	16	-32768 a 32767
DINT	Double Integer	32	-2^{31} a $(2^{31} - 1)$
LINT	Long Integer	64	-2^{63} a $(2^{63} - 1)$
USINT	Unsigned Short Integer	8	0 a 255
UINT	Unsigned Integer	16	0 a 65535
UDINT	Unsigned double Integer	32	0 a $(2^{32} - 1)$
ULINT	Unsigned Long Integer	64	0 a $(2^{64} - 1)$

- Não necessita suportar todos os tipos de dados.



Tipos de Dados

- Uso de literais para inteiro:
 - Decimais: -30 12 +52 23_584_444
 - Binário: 2#1111_0101
 - Octal: 8#735
 - Hexadecimal: 16#C3D



Tipos de Dados

- **Ponto Flutuante.**

Tipo IEC	Descrição	Bits	Faixa	Precisão
REAL	Real	32	$\pm 10^{\pm 38}$	$1/2^{23}$
LREAL	Long Real	64	$\pm 10^{\pm 308}$	$1/2^{52}$

- **DUO**

- REAL

- Não tem LREAL

- O formato utilizado é da norma IEC 559

- Uso de literal:

4.65

+5_011.6

-0.62

-0.584e-5



Tipos de Dados

- **Duração/Tempo**

Tipo IEC	Descrição	Bits	Uso
TIME	Duração do tempo	Depende da implementação	Armazena duração do tempo após o evento

- Usado tipicamente para medir duração de intervalos ou definir time-outs.
- Duas formas para uso de literais.
 - Forma curta
 - Forma longa.



Tipos de Dados

- Ambas formas utilizam as letras para representação.

d = dias

h = horas

m = minutos

s = segundos

ms = milisegundos

- Forma curta:

– T#15d5h12s



15 dias, 5 horas e 12 segundos

– T#6d5.5h



6 dias e 5.5 horas

- Forma longa:

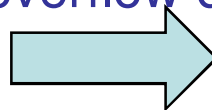
– TIME#22d_2h_2m_2s



22 dias, 2 horas, 2 minutos e 2 segundos

É permitido o overflow da parte mais significativa.

T#65m12s



Equivalente a T#1h5m12s



Tipos de Dados

- **Data e Hora**

Tipo IEC	Descrição	Bits	Uso
DATE	Data Absoluta	Depende da implementação	Armazena datas de calendário
TIME_OF_DAY ou TOD	Hora do Dia	Depende da implementação	Armazena real time clock
DATE_AND_TIME ou DT	Data e Hora	Depende da implementação	Armazena data e hora

- **Duas formas de literais**

Tipo de dado	Forma Curta	Forma Longa
DATE	D#	DATE#
TIME_OF_DAY ou TOD	TOD#	TIME_OF_DAY#
DATE_AND_TIME ou DT	DT#	DATE_AND_TIME#



Tipos de Dados

- Ex:

- D#2008-09-20 20 de setembro de 2008
- DATE#2008-09-20 20 de setembro de 2008

- TOD#12:15:18 12 horas, 15 minutos e 18 segundos
- TIME_OF_DAY#12:15:18 12 horas, 15 minutos e 18 segundos

- DT#2008-09-20:12:15:18 20 de setembro de 2008 às
 12 horas, 15 minutos e 18 segundos

- DATE_AND_TIME#2008-09-20:12:15:18
 20 de setembro de 2008 às
 12 horas, 15 minutos e 18 segundos



Tipos de Dados

- **Strings**

- Cadeia de caracteres

Tipo IEC	Descrição	Bits	Uso
STRING	String de caracteres	Depende da implementação	Armazena texto

- Cadeia de bits

Tipo IEC	Descrição	Bits	Uso
BOOL	Bit string de 1 bit	1	Digital, estados lógicos
BYTE	Bit string de 8 bits	8	Informação binária
WORD	Bit string de 16 bits	16	Informação binária
DWORD	Bit string de 32 bits	32	Informação binária
LWORD	Bit string de 64 bits	64	Informação binária



Tipos de Dados

- **Inicialização variáveis**

- Pela norma IEC, toda variável deve ter um valor inicial
- Em caso de não ser inicializada a variável pelo usuário, é utilizado valor padrão.
- Valores padrões:

Tipo	Valor Inicial
Default	0
String	“
Datas	D#0001-01-01

- A variável pode ser inicializada na declaração



Tipos de Dados

- **Tipos de dados Derivados**

- Novos tipos de dados podem ser definidos a partir dos tipos elementares.
- A declaração de um novo tipo de dado é feita através do uso das palavras-chave TYPE e END_TYPE.

– Ex:

```
TYPE  
    PRESSÃO :      REAL;  
END_TYPE
```

- Pode ser definido um valor inicial para os tipos de dados derivados.



Tipos de Dados

- **ESTRUTURAS**

- Tipos de dados compostos podem ser derivados pela definição de uma estrutura.
- Uma estrutura é um agrupamento de elementos de diferentes tipos.
- As palavras chaves para definir uma estrutura são STRUCT e END_STRUCT.
- Ex:

```
TYPE SENSOR_PRESSAO:
    STRUCT
        ENTRADA           : PRESSAO;
        ESTADO            : BOOL;
        ...
    END_STRUCT
END_TYPE
```



Variáveis

- Todas as variáveis devem ser declaradas no início das Unidades de Organização de Programas (POUs).
- Tipos de variáveis:
 - Internas ou Locais
 - De uso interno ou escopo local.
 - Declaradas utilizando as palavras chaves VAR e END_VAR.
 - Existem somente em tempo de execução.
 - EX:

```
VAR  
    VELOCIDADE          : REAL;  
END_VAR
```



Variáveis

– De Entrada

- Supridas por uma fonte externa.
- Declaradas utilizando as palavras chaves VAR_INPUT e END_VAR.
- Ocupam posições de memória.
- Ex:

```
VAR_INPUT
    POSIÇÃO_X      : REAL;
    POSIÇÃO_Y      : REAL;
END_VAR
```



Variáveis

– De Saída

- Fornecem valores que serão escritos em uma variável externa.
- Declaradas utilizando as palavras chaves VAR_OUTPUT e END_VAR.
- Não utilizadas em funções.
- Ocupam posição de memória.
- Ex:

```
VAR_OUTPUT
    SAIDA_1 : BOOL;
    SAIDA_2 : BOOL;
END_VAR
```



Variáveis

– De ENTRADA/SAÍDA

- Agem tanto com parâmetro de entrada como de saída.
- Declaradas utilizando as palavras chaves VAR_IN_OUT e END_VAR.
- Utiliza passagem pela referência.
- Ex:

```
VAR_IN_OUT  
    RESULTADO      : INT;  
END_VAR
```



Variáveis

– Temporárias

- Declaradas somente em blocos funcionais e programas.
- Alocação temporária de variáveis na memória.
- Utilizadas para armazenamento de dados intermediários.
- Palavras chaves para declaração VAR_TEMP e END_VAR.
- Ex:

```
VAR_TEMP  
    TAXA      : REAL;  
END_VAR
```



Variáveis

– Globais

- Declaradas no nível de configuração, recurso ou programa.
- Acessada por POU's por meio de uma declaração de variável externa.
- Palavras chaves para declaração VAR_GLOBAL e END_VAR.
- Ex:

```
VAR_GLOBAL  
    VELOCIDADE_M    : REAL;  
END_VAR
```



Variáveis

– Externa

- Declaradas dentro de POU's.
- Acesso a variáveis globais definidas no nível de configuração, recurso ou programa.
- Nome das variáveis externa e global devem ser idênticos.
- Palavra chave para declaração VAR_EXTERNAL e END_VAR.
- Ex:

```
VAR_EXTERNAL  
    VELOCIDADE_M    : REAL;  
END_VAR
```



Variáveis

– Representação direta

- Permitem acessar diretamente as posições de memória.
- Começam com o caractere %, seguido de uma ou duas letras.
- A memória de um CP pode ser dividida em três regiões lógicas.

Primeira Letra	Interpretação
I	Entradas (Input): Recebe os valores das variáveis analógicas e discretas dos módulos de entrada.
Q	Saídas (Output): Para armazenar os valores a serem escritos nos dispositivos externos.
M	Memória interna: Armazena valores intermediários.



Variáveis

Segunda Letra	Interpretação
X	Bit
B	Byte (8 bits)
W	Word (16 bits)
D	Double word (32 bits)
L	Long word (64 bits)

- Demais dígitos representam a posição de memória.

- Ex:

%I100 (* Memória de entrada bit 100 *)

%IX100 (* Memória de entrada bit 100 *)

%IW122 (* Memória de entrada palavra 122 *)

%QB100 (* Memória de saída byte 100 *)

%MW132 (* Palavra de memória 132 *)



Variáveis

- Mostrar Hi,Lo e Tabela de memória



Variáveis

– Atributos de variáveis

- As variáveis podem possuir atributos definidos na declaração que informam ao CP como considerar o comportamento do valor da variável.
- Sem atributo é uma variável normal.
- Atributos permitidos:
 - Retenção:
 - » Serão colocadas em memória retentiva.
 - » Mantêm o valor em caso de perda de alimentação do CP.
 - » Declaradas utilizando RETAIN.



- Unidade de Organização de Programas.
 - São as formas de implementar o programa do CP através da associação de variáveis e instruções.
 - Determinadas POU's podem ser utilizadas como parte de outras POU's, com o objetivo de estruturar a programação e reutilizar código.
 - Três tipos de POU's:
 - Função
 - Blocos funcionais
 - Programas



POUs

– Principais características:

- Possibilidade de fornecimento pelo fabricante;
- Possibilidade de desenvolvimento pelo usuário;
- Não permite recursividade.

– Hierarquia entre POUs:

- Programas podem conter qualquer tipo de POU;
- Blocos Funcionais podem conter outros Blocos Funcionais e Funções;
- Funções somente podem conter Funções.



- Funções
 - Elementos comuns para a definição de novas POUs.
 - São elementos reutilizáveis de software.
 - Produzem um único dado como resultado, que é próprio nome da função.
 - Não são instanciáveis.
 - Só existem em tempo de execução.
 - Deve ser feita a declaração do tipo antes do uso.
 - Não possui memória de estados, se chamar uma função com as mesmas entradas deve apresentar o mesmo resultado
 - Chamada em linguagens textuais pelo nome e em linguagens gráficas pelo bloco.



- Funções

- Funções definidas pela norma.

- » Numéricas, conversão de tipos de dados, equivalência, manipulação de bits, de seleção, comparação, manipulação de string, de tempo e data. (verificar help)

- Ex:

```
FUNCTION AREA      : REAL
  VAR_INPUT
    L_1, L_2      : REAL;
  END_VAR

  AREA :=          (L_1 * L_2);

END_FUNCTION
```



- Blocos Funcionais
 - Mais importante dentro do modelo de reutilização de código.
 - Um dos principais elementos para estruturação de programas do modelo top-down bottom-up.
 - Elemento encapsulado de software que pode ser reutilizado.
 - Somente tipos de dados definidos podem ser utilizados para interface externa.
 - Conjunto básico de blocos funcionais definidos pela norma.
 - Valores das variáveis mantidos entre uma execução e outra



- Blocos Funcionais

- Representado por um retângulo com pinos de entrada do lado esquerdo e saída do lado direito.
- Possui variáveis de entrada, interna e saída.
- Cada instância declarada é reserva uma área de memória.
- Devem ser declarados o tipo e as instâncias do Bloco Funcional.
- Pode produzir uma ou mais variáveis de saída.
- As variáveis internas não são acessíveis, somente pelo algoritmo interno do bloco.
- Uma instância de um Bloco Funcional pode ser utilizada na declaração de outro Bloco Funcional ou Programa.



- Bloco Funcional
 - Declaração é feita utilizando FUNCTION_BLOCK e END_FUNCTION_BLOCK.
 - Não é permitida a declaração de variáveis globais dentro de um Bloco Funcional.
 - Para o projeto de um Bloco Funcional, deve ser considerada a possibilidade de uso futuro.



- Programas
 - É a maior forma de POU.
 - Pode ser declarado a nível de recurso.
 - Agrupamento de Funções e Blocos Funcionais associados para realizar a função de controle.



POUs

- Unidade de Organização de Programas.
 - Três tipos de POUs:
 - Função
 - Blocos funcionais
 - Programas
- Características
- Hierarquia



POUs

- Unidade de Organização de Programas.
 - Três tipos de POUs:
 - Função
 - Blocos funcionais
 - Programas
- Características
- Hierarquia



POUs

- Unidade de Organização de Programas.
 - Três tipos de POUs:
 - Função
 - Blocos funcionais
 - Programas
- Características
- Hierarquia



Tarefa

- Escalonamento
 - Não preemptivo
- Declaração
- Associando programas e blocos funcionais
- Figura página 84



Configuração

- PLC Configuration



Índice - Programação

- ST
- FBD
- LD
- IL
- SFC



- Introdução
 - Onde é útil
- Característica
- Chama de Blocos Funcionais
- Comandos



- Introdução
- Características
- Saltos e Rótulos
- Avaliação da Rede
- Portabilidade ST <-> FBD
- Declaração
- Blocos padrões



- Introdução
- Características
- Metodologia
 - Diferente do MasterTool XE
- Uso de blocos funcionais
- Uso de funções
- Saltos e Rótulos
- Avaliação da rede
- Portabilidade ST e FBD <-> LD
- Aplicabilidade



- Introdução
- Estrutura da linguagem
- Semântica
 - Frizar que cada instrução sempre trabalha com o acumulador, não há instruções diretas.
- Operadores
- Chamada de funções e blocos funcionais
- Deficiências
- Portabilidade IL <-> Outras



- Introdução
- Estrutura da rede
 - Seqüências alternativas
 - Seqüências simultâneas
 - Passos
 - Transições
 - Ações
- Execução de um SFC
- Regras
- Projeto seguro e inseguro



altus