



Informática Industrial

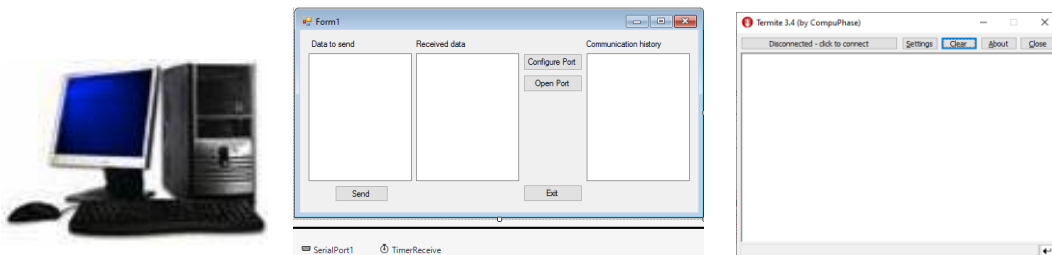
Aula Prática 5

Controlo de um reservatório por RS232

Jorge Almeida, José Paulo Santos

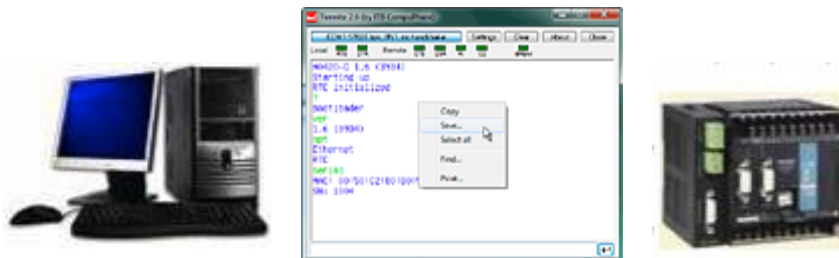
1. Descrição do trabalho

Na aula prática 3 foi feita uma aplicação em Visual Basic que comunicava por RS232. Para poder receber e enviar mensagens sem a necessidade de portas físicas foi instalado um emular de portas COM que nos permitiu colocar a nossa aplicação a enviar e a receber dados para o Terminate.



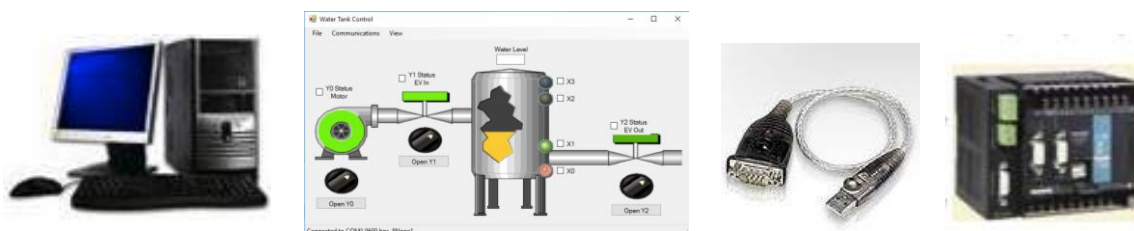
Aula 3 - Comunicação RS232 entre PC (Visual Basic) com Terminate

Por outro lado, na aula 4 foi desenvolvido um programa para PLC que permitia ao PLC comunicar com o PC. Do lado do PC foi usado o programa Terminate (semelhante ao hyperterminal) para verificar as mensagens enviadas pelo PLC. Também foi possível usar o Terminate para enviar mensagens para o PLC, como por exemplo ligar Y0. A figura seguinte ilustra esse trabalho.



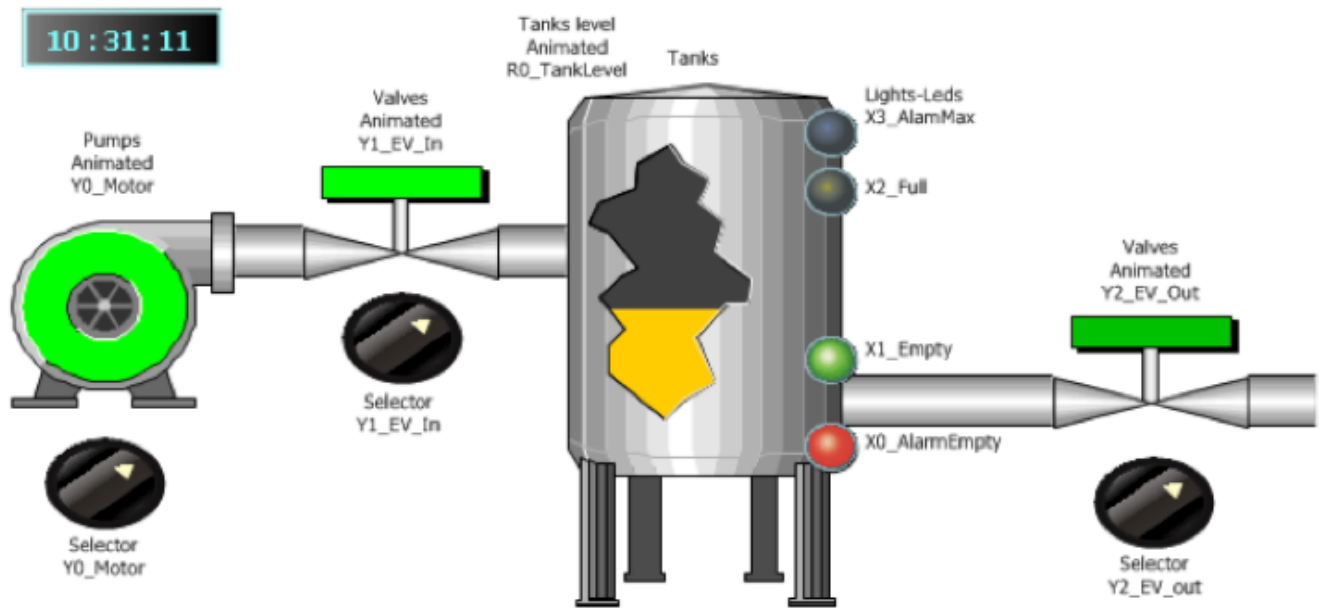
Aula 4 - Comunicação RS232 entre PC (Termite) e PLC

Neste trabalho pretende-se desenvolver uma aplicação em Visual Basic que substitua o Terminate da aula 4. Por outras palavras, a aplicação desenvolvida deve comunicar com o PLC, enviando mensagens de comando e recebendo (e processando) mensagens vindas do PLC. Isto está ilustrado na figura seguinte.



Aula 5 - Comunicação RS232 entre PC (Visual Basic) e o PLC

A aplicação será utilizada para efetuar monitorização e controlo de um reservatório de água como mostrado na figura em baixo. Recomenda-se a leitura do documento que descreve o controlo do reservatório que está no elearning. A figura em baixo está também no elearning caso seja necessária para inclusão no programa a desenvolver.



Controlo de um reservatório de água.

2. Funções para manipulação de strings

Para fazer a extração de mensagens recebidas pelo PLC será necessário fazer um conjunto de operações de manipulação de strings. Segue uma descrição das funções de manipulação de strings mais comuns.

Função “Mid”

Permite separar uma *substring* a partir de uma *string* original.

A função recebe três parâmetros: uma variável do tipo *string*, contendo um conjunto de caracteres; um número com a posição do primeiro caractere a ler; e o número de caracteres consecutivos a ler

Mid-Exemplo:

```
Dim strOriginal As String = "0123456789"
MessageBox.Show(Mid(strOriginal, 1, 2)) 'devolve o valor "01"
MessageBox.Show(Mid(strOriginal, 5, 3)) 'devolve o valor "456"
```

A função “Split”

Divide uma *string* em *substrings*. A função recebe uma *string*, um caractere delimitador e devolve um array de *strings*. Em cada posição do array devolvido vem uma parte da *string* original. Por exemplo, se a *string* “String_original” possuir um espaço em branco entre cada letra e o utilizador quiser separar as letras obtendo em cada posição do array uma só letra, pode indicar à função “split” para usar o caractere “espaço” como divisor da *string* original. A este caractere, usado para dividir a “String_original”, chama-se delimitador.

Split-Exemplo:

```
Dim strOriginal As String = "o l a"
```

```
Dim arrayDeStrings As String()
arrayDeStrings = Split(strOriginal, " ")
MessageBox.Show(arrayDeStrings(0))
MessageBox.Show(arrayDeStrings(1))
MessageBox.Show(arrayDeStrings(2))
```

A função “InStr”

Permite localizar a posição de uma *substring* numa *string* maior.

A função recebe duas *strings*: a *string* maior e a *substring* e devolve um inteiro com posição da *substring* na *string* maior. O valor devolvido pode ser (0) se a *substring* não for encontrada na *string* maior, ou (≥ 1).

InStr-Exemplo:

```
Dim strOriginal As String = "abcdefghi"
Dim strChave As String = "cd"
MessageBox.Show(InStr(strOriginal, strChave)) 'devolve o valor 3
```

As funções “Right” e “Left”

Devolve uma *substring* que contém um determinado número de caracteres, a contar da direita/esquerda, da *string* original.

Por existir uma propriedade do form com o mesmo nome é necessário utilizar o caminho completo *Microsoft.VisualBasic.Left* e *Microsoft.VisualBasic.Right*.

Right e Left-Exemplo:

```
Dim strOriginal As String = "abcdefghi"
MessageBox.Show(Microsoft.VisualBasic.Left(strOriginal, 3)) 'devolve o valor "abc"
MessageBox.Show(Microsoft.VisualBasic.Right(strOriginal, 2)) 'devolve o valor "hi"
```

A função “Len”

Devolve um inteiro com o número de caracteres que existem numa *string*.

A função recebe uma *string*, da qual queremos saber o seu comprimento, ou seja, o número de caracteres que a constituem e devolve um número.

InStr-Exemplo:

```
Dim strOriginal As String = "abcdefghi"
MessageBox.Show(Len(strOriginal)) 'devolve o valor 9
```

A função “StrComp”

Compara duas *strings*.

A função recebe duas *strings* e devolve um valor inteiro, igual a zero se as duas *strings* forem iguais.

StrComp-Exemplo:

```
Dim palavra1 As String = "123"
Dim palavra2 As String = "123"
MessageBox.Show(StrComp (palavra1, palavra2)) 'devolve o valor 0
```

A função “CStr”

Permite converter um número numa *string*. A *string* obtida pode então ser atribuída a variáveis do tipo *string* ou usada em funções de manipulação de *strings*.

CStr-Exemplo:

```
Dim texto As String
```

```
texto = CStr(123)
```

A função “Chr”

Converte um valor numérico no seu caractere correspondente, de acordo com a tabela ASCII

Chr-Exemplo:

```
MessageBox.Show(Chr(65)) 'devolve o valor "A"
```

A função “Asc”

Converte um caractere no seu valor decimal de acordo com a tabela ASCII.
Esta função recebe uma string e devolve um valor inteiro.

Asc-Exemplo:

```
MessageBox.Show(Asc("A")) 'devolve o valor 65
```



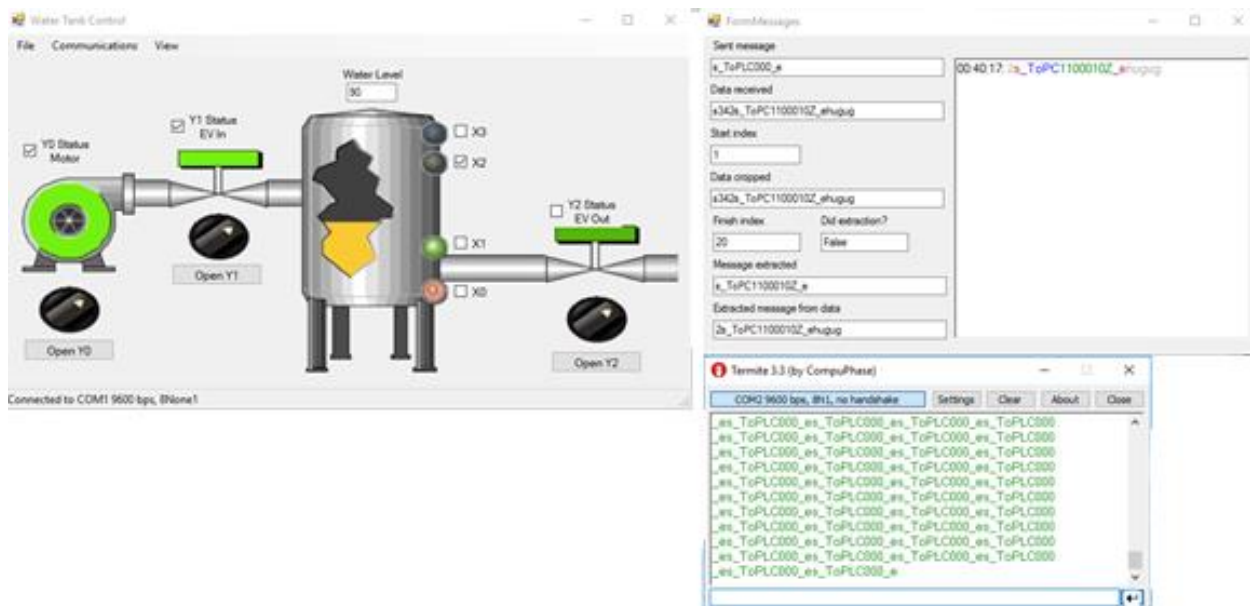
Informática Industrial

Trabalho prático 5 Controlo de um reservatório

Neste trabalho o nível de dificuldade aumenta em relação aos trabalhos anteriores, **uma vez que não é dado código que os alunos possam copiar. Cabe aos alunos definir e implementar o programa**, sendo que o mesmo deve, no final deste trabalho, ter as seguintes funcionalidades:

- Possuir um interface gráfico para configuração de uma comunicação RS232;
- Possuir um interface gráfico onde seja possível atuar as saídas Y0, Y1 e Y2 do PLC;
- Possuir um interface gráfico onde seja possível visualizar o estado das saídas e entradas e do nível de água transmitidos pelo PLC;
- Envio de mensagens para o PLC e visualização das mensagens enviadas;
- Receção de dados do PLC e visualização;
- Extração de mensagens a partir dos dados recebidos (estudar funções da secção 4);
- Implementação do modo de controlo automático do reservatório.

Recomenda-se que os alunos façam pequenas funcionalidades do sistema completo e as testem o mais frequentemente possível. Podem por exemplo seguir a lista indicada em cima. Segue um exemplo de um programa desenvolvido para o efeito.



Exemplo de programa a desenvolver (ligado ao Termite)