

Autor: Luciano Lettnin;

Finalidade: Software para plotar o valor de temperatura lido pelo CI LM35;

Data: 21/06/2013.

```
//-----
```

```
#include <vcl.h>
```

```
//includes acrescentados para utilizar a geração de arquivos e captar a hora e data atual
```

```
#include <stdio.h>
```

```
#include <dos.h>
```

```
#include <math.h>
```

```
#pragma hdrstop
```

```
#include "ColetaDAQUnit1.h"
```

```
//-----
```

```
#pragma package(smart_init)
```

```
#pragma link "CPort"
```

```
#pragma link "CPortCtl"
```

```
#pragma resource "*.dfm"
```

```
TForm1 *Form1;
```

```
//variáveis globais
```

```
int cont=0; //contador do eixo do tempo do chart
```

```

int cont_linhas=0; //contador das linhas do buffer retorno

int n_pt_visible = 0; //pontos visíveis da janela no chart

int n_pt_serie = 0; //pontos escritos na série

float data_conv, data_conv1, data_conv2, data_conv3,data_conv4,data_conv5,data_conv6,data_conv7; //valor do sinal convertido para volts

//-----

__fastcall TForm1::TForm1(TComponent* Owner)
    : TForm(Owner)
{
    ComPort1->Open(); //abre a comport

    ComPort1->ClearBuffer(true, true); //limpa buffers da serial

    Series1->Clear(); //limpa o chart
}

//-----

void __fastcall TForm1::Button1Click(TObject *Sender)
{
    ComPort1->ShowSetupDialog(); //mostra a janela de configuração da comunicação serial
}

//-----

void __fastcall TForm1::BitBtn1Click(TObject *Sender)

```

```

{
    ComPort1->Close(); //fecha a comport
}

//-----

void __fastcall TForm1::ComPort1RxChar(TObject *Sender, int Count)
{
    char retorno[1024];
    unsigned char data_0, data_1; //parte MSB e LSB do sinal vindo da serial
    float data; //valor do sinal MSB+LSB
    int num_carac; //número de bytes na entrada da comport
    num_carac = ComPort1->InputCount(); // lê número de bytes disponíveis no buffer de entrada da serial
    if (num_carac == 2)
    {
        ComPort1->Read(retorno,num_carac); // pega os bytes
        data_0 = retorno[0];
        data_1 = retorno[1];
        data = ((data_1 & 0x03) <<8) | data_0; //valor do conversor A/D
        data_conv1 = ((data*160)/1024); //converte o valor do conversor A/D para tensão e multiplica para obter o valor em temperatura
        data_conv2= data_conv1;
    }
}

```

```

data_conv3= data_conv2;
data_conv4= data_conv3;
data_conv5= data_conv4;
data_conv6= data_conv5;
data_conv7= data_conv6;

data_conv = ((data_conv1+data_conv2+data_conv3+data_conv4+data_conv5+data_conv6+data_conv7)/7);//Média de aquisições para suavizar o gráfico em
função do problema de oscilação apresentado por um lote da placa de aquisição de dados CD2000.

Series1->AddXY(cont,data_conv, "", clTeeColor); //acrescenta o ponto com as duas coordenadas

cont++; //incrementa o valor da posição no eixo do tempo

}

}

//----- * inicio do código do gráfico

void __fastcall TForm1::Timer1Timer(TObject *Sender)
{
    if (ComboBox1->ItemIndex==0)
    {
        ComPort1->Write("k",1);

        Label1->Caption =data_conv;
    }

    if (ComboBox1->ItemIndex==1)

```

```
{  
    ComPort1->Write("l",1);  
    Label1->Caption =data_conv;  
}  
if (ComboBox1->ItemIndex==2)  
{  
    ComPort1->Write("m",1);  
    Label1->Caption =data_conv;  
}  
if (ComboBox1->ItemIndex==3)  
{  
    ComPort1->Write("n",1);  
    Label1->Caption =data_conv;  
}  
if (ComboBox1->ItemIndex==4)  
{  
    ComPort1->Write("o",1);  
    Label1->Caption =data_conv;  
}
```

```
}
```

```
//-----
```

```
void __fastcall TForm1::Button4Click(TObject *Sender)
```

```
{
```

```
Series1->Clear(); //limpa o chart
```

```
cont = 0; //zera o valor do eixo do tempo
```

```
}
```

```
//----- * fim do código fonte do gráfico
```