

# UnicViewAD - Resumo

---

Sugestão das funções gerais utilizadas no software

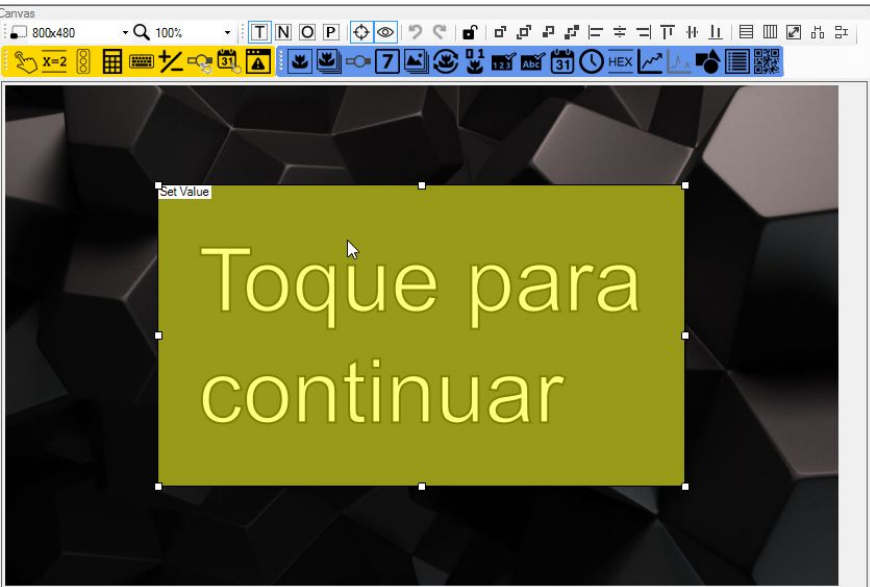
Vinícius de Souza Miralhas - 10728289

# Documentação geral

Para mais informações, consulte o .pdf associado a estes slides, além dos links abaixo:

- [Treinamento UnicViewAD](#)
- [Tutoriais UnicViewAD](#)
- [Exemplos UnicViewAD](#)
  - Estes exemplos são do UnicViewAD puro, sem Arduino
- [Biblioteca UnicViewAD.ino: exemplos, referências, etc.](#)
- [Documentação UnicViewAD](#)
  - Esta documentação é do software UnicViewAD puro, sem Arduino

# Set Value



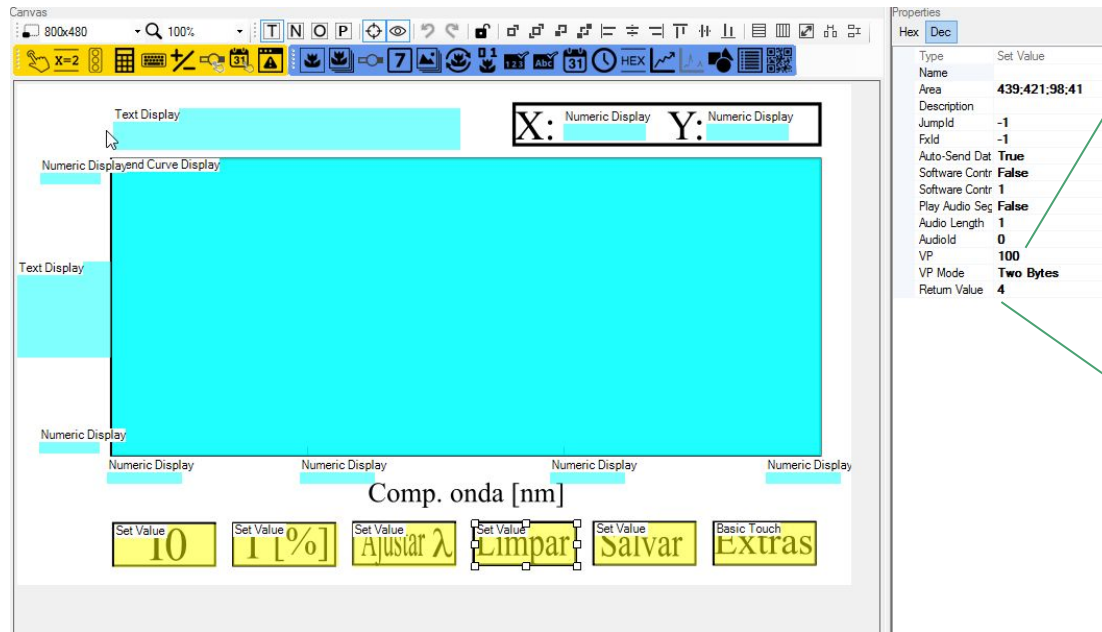
Properties	
Hex	Dec
Type Set Value	
Name	
Area	147,96,506,289
Description	
JumpId	-1
FxId	-1
Auto-Send Dat	True
Software Contr	False
Software Contr	1
Play Audio Seg	False
Audio Length	1
AudioId	0
VP	10
VP Mode	Two Bytes
Return Value	1

```
// LockScreen  
LcmVar startButton(10);
```

```
while (true) {  
  if (startButton.available()) {  
    if (startButton.getData() == 1) {  
      SDbegin(); // Inicia o cartão SD  
      displayMessage("Abra o serial mor  
adjustLambda(false); // Requisito:  
Lcm.changePicId(10);  
Localizar();  
delay(1000);  
NovaPosicao(lambda_min);  
break;  
    }  
  }  
}
```

Crie uma variável LcmVar nas definições globais do código-fonte do Arduino. Ao tocar, o código lê os dados.

# Set Value com muitos botões

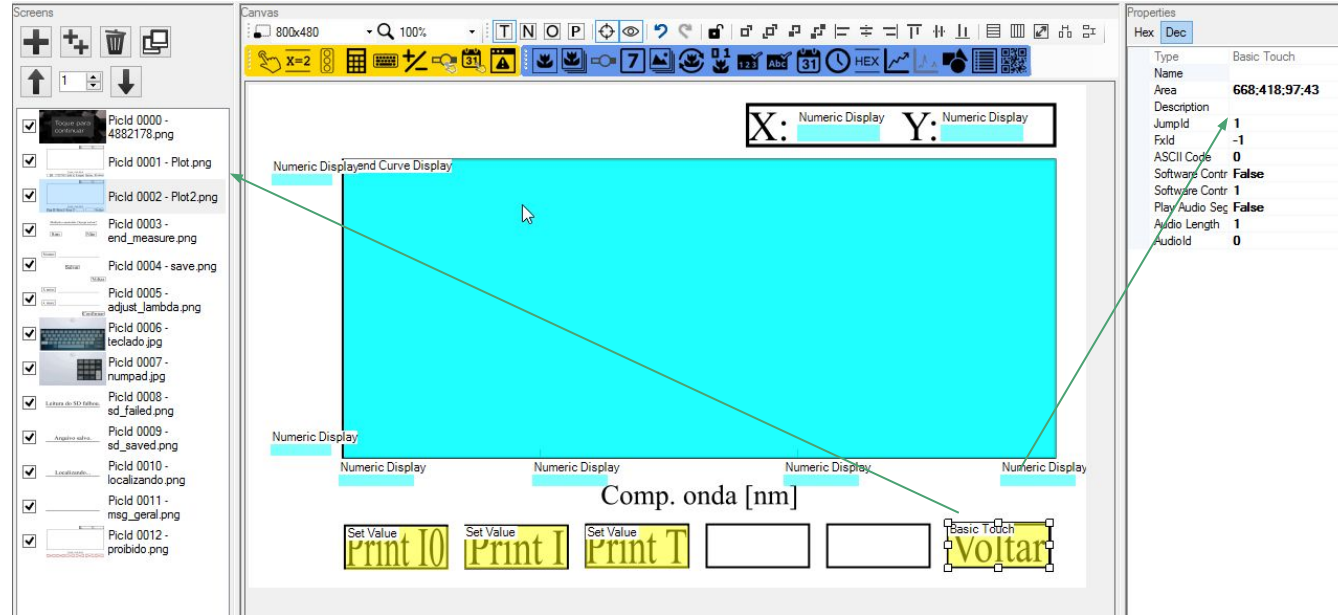


```
// Botão do plot:  
LcmVar plotButton(100);
```

```
if (plotButton.available()) {  
  
    byte opcao = plotButton.getData();  
    Serial.println(opcao);  
    switch (opcao)  
    {  
        // Botão 1: Lê e plota I0  
        // Limpa o TrendCurve. Insete os 1  
        case 1:  
  
        case 4:  
            // Limpa o plot com os comprim  
            cleanPlot(0, 0, 0);delay(50);  
            cleanPlot(0, 0, 1);delay(50);  
            cleanPlot(0, 0, 2);delay(50);  
            break;  
    }  
}
```

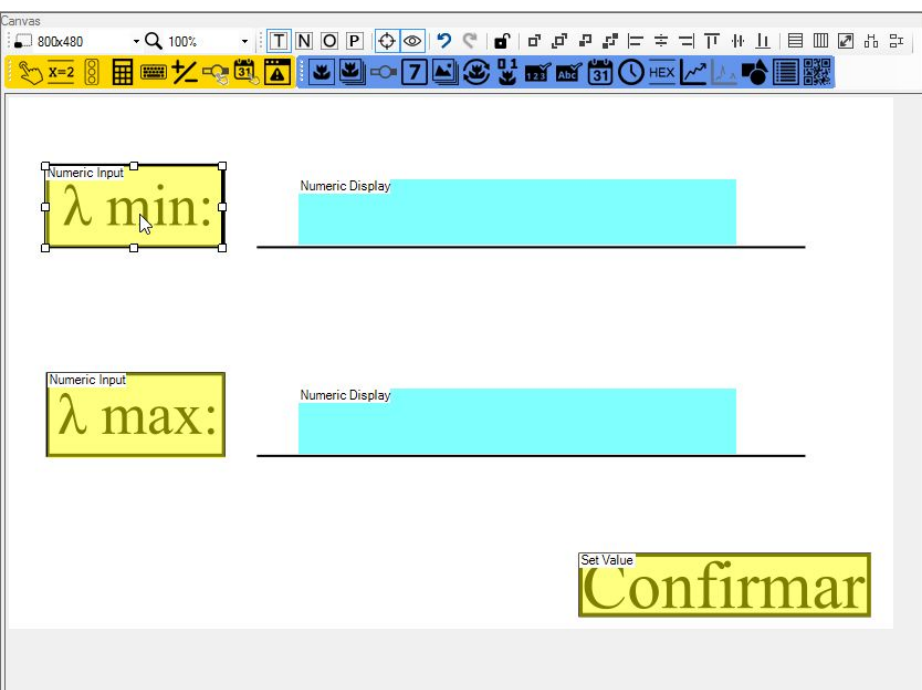
Crie vários botões para uma mesma LcmVar.  
Isto funciona bem com switch case

# Basic Touch



O basic touch não interage com o Arduino. Ao pressionar o botão, a tela será mudada ao PicId número 1, que é o menu principal do plot. Mude o JumpId como quiser.

# Numeric Input



Properties	
Hex Dec	
Type	Numeric Input
Name	
Area	33;62;160;74
Description	
JumpId	-1
FxId	-1
Auto-Send Dat	True
Software Contr	False
Software Contr 1	
Play Audio Seg	False
Audio Length	1
AudioId	0
VP	502
Library	Microsoft-Sans-Seri
Font Width	12
Font Color	0
Cursor Color	Black
Source PicId	7
Source Area	0;0;800;480
Target Top-Lef	0;0
Cursor Origin	272;38
Integer Digits	4
Decimal Digits	0
Show Character	True
Limit Value Rar	False
Max Value	0
Min Value	0
VP Mode	Two Bytes

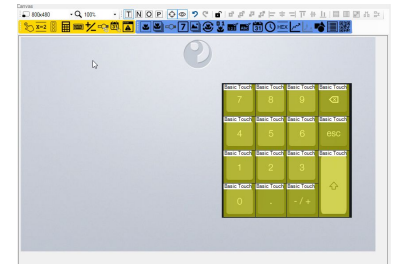
```
// Ajustar lambda
```

```
LcmVar lcmLambdaMinInput(502);
```

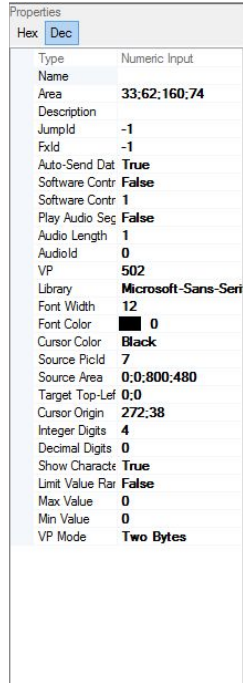
```
LcmVar lcmLambdaMaxInput(501);
```

```
// Lê o lambda_min do usuário
```

```
if (lcmLambdaMinInput.available()) {  
    lambda_min = lcmLambdaMinInput.getData();  
    if (lambda_min > lambda_max) lambda_min = lambda_max;  
    if (lambda_min < 300) lambda_min = 300;  
}
```

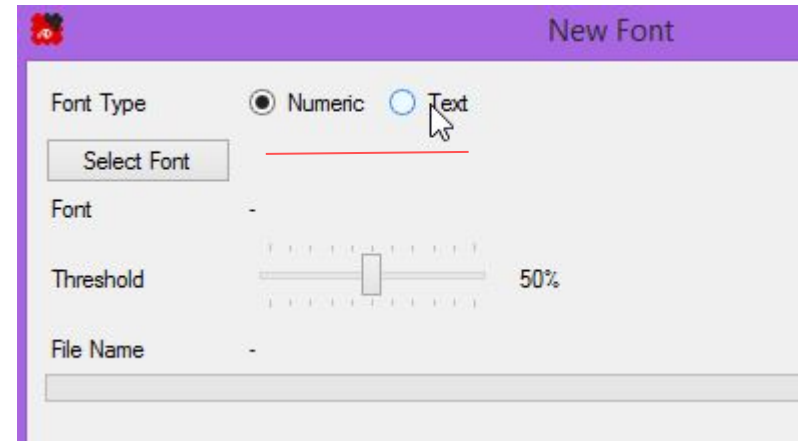
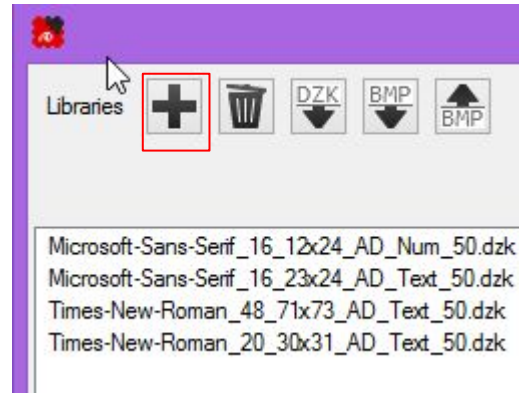


# Numeric Input



Para o numpad, simplesmente copiei o teclado numérico de um dos exemplos do UnicViewAD, com todos os endereços das teclas automaticamente mapeados.

Como o input era int, usei e limitei os caracteres a 4 dígitos inteiros. Adicione também uma biblioteca de fontes numéricas:



# Numeric Display

The screenshot shows a LabVIEW front panel and back panel. The front panel features a large cyan numeric display area. Above it, there are labels 'X: Numeric Display' and 'Y: Numeric Display'. Below the main display, there are several smaller numeric displays and a row of buttons labeled 'Set Value', '10', '1 [%]', 'Ajustar  $\lambda$ ', 'Limpar', 'Salvar', and 'Extras'. The back panel shows a 'Properties' window for a 'Numeric Display' object. The 'Dec' tab is selected, showing properties like Name, Area, Description, VP, PP, Library, Font Width, Font Color, Text Alignment, Unity, Value Type, Integer Digits, Decimal Digits, and Initial Value. A green arrow points from the 'Initial Value' property to the text below.

Property	Value
Type	Numeric Display
Name	
Area	524;25;79;29
Description	
VP	320
PP	-1
Library	Microsoft-Sans-Seri
Font Width	12
Font Color	0
Text Alignment	Left
Unity	
Value Type	Two Bytes
Integer Digits	4
Decimal Digits	0
Initial Value	0

```
LcmVar xBox(320); // Estas são as c  
LcmVar yBox(321);
```

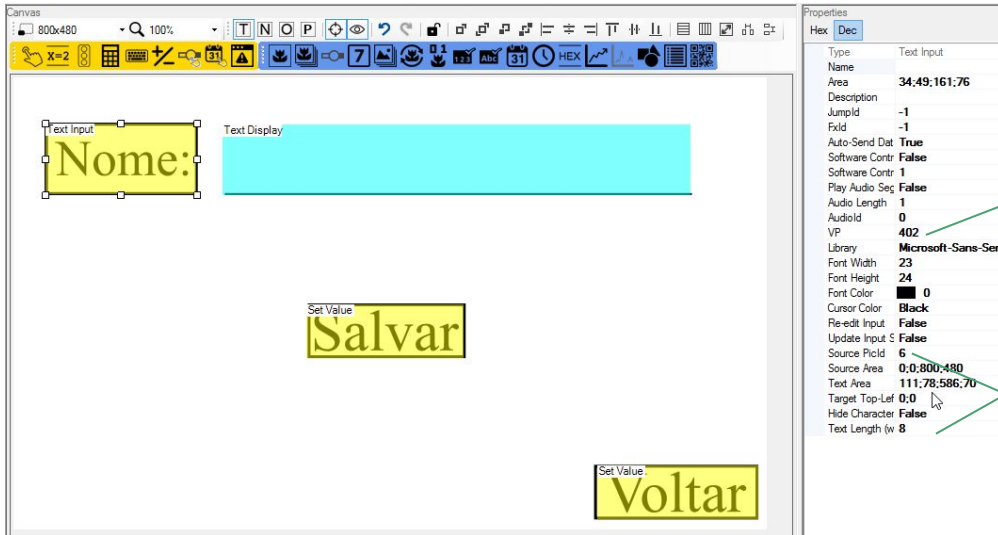
```
y = vector[imed];  
yBox.write(y); //Atualiza o y atual
```

O tipo da variável pode ser int ou float. Pelas definições, aqui é int.



# Text Input

```
// Salvar SD:  
LcmString fileNameInput(402, 16);
```



Teclado

2x

Text Input é mais complicado. O total de caracteres é 2x TextLength. Use SourceArea total para pegar toda a área do teclado. TextArea é onde o cursor irá inserir o texto



# Text Input

```
String fname = "trans.csv";  
...  
if (fileNameInput.available()) {  
    fname = "";  
    while (fileNameInput.available()) {  
        fname += (char) fileNameInput.getData();  
    }  
}
```

Recomendado utilizar variáveis declaradas como String para manipular a entrada de dados com TextInput.

Para o teclado, simplesmente copiei as imagens e os basic touch dos exemplo providos pelo UniconViewAD

# Text Display

The screenshot shows a graphical user interface editor. The canvas contains three text components: a yellow box labeled 'Text Input' with the text 'Nome:', a light blue box labeled 'Text Display' which is empty, and two yellow boxes labeled 'Set Value' with the text 'Salvar' and 'Voltar'. The properties panel on the right shows the 'Text Display' component selected, with the following properties:

Type	Text Display
Name	
Area	222,50;497,75
Description	
VP	422
PP	-1
Library	Microsoft-Sans-Ser
Font Width	23
Font Height	24
Font Encoding	8 Bit
Font Color	0
Text Length	16
Fixed Spacing	False
Character Spar	0
Line Spacing	0
Value Type	String
Initial Value	

```
LcmString fileNameDisplay(422, 16);
```

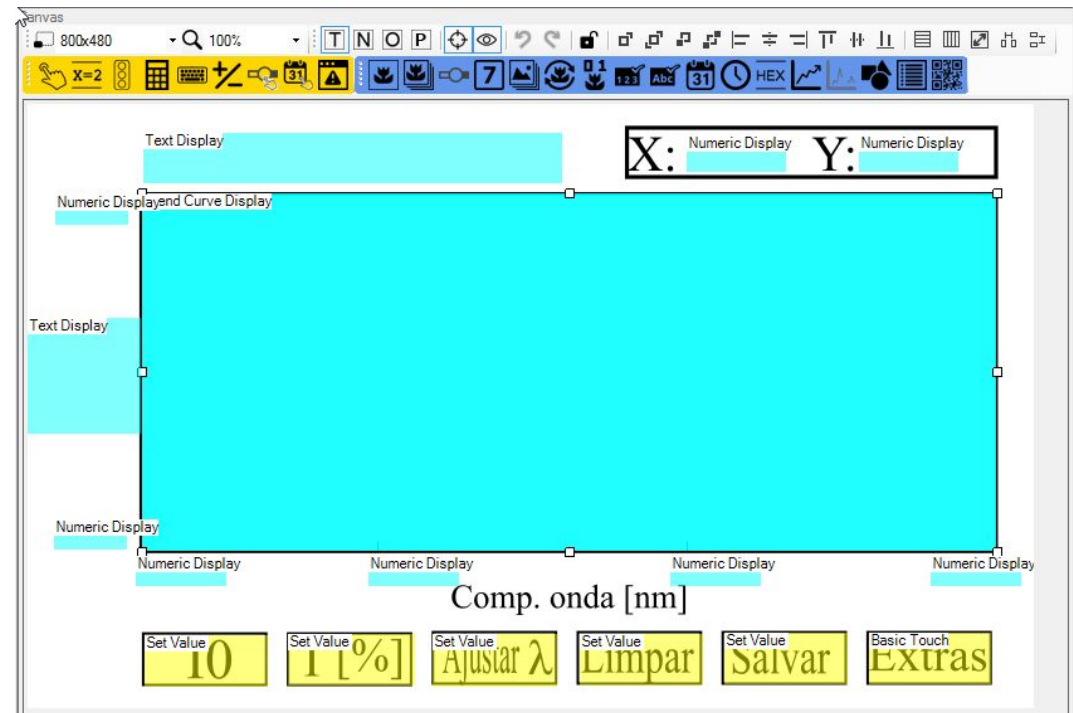
```
String fname = "trans.csv"; // 1  
while (true) {  
    fileNameDisplay.write(fname);
```

Atenção: recomendado sempre avançar VP pelo total de caracteres da string anterior.

Exemplo: o TextInput do nome estava no VP 402. O texto tinha 16 caracteres, mas avancei 20 ao VP atual de 422. Se houvesse outro textinput/textdisplay, mas com 40 caracteres, eu avançaria uns 45 no VP por precaução

**Deve-se adicionar uma fonte de texto (volte ao slide 7)**

# TrendCurveDisplay



Properties	
Hex	Dec
Type	Trend Curve Display
Name	
Area	91:71:680:285
Description	
PP	-1
Curve Color	992
Channel	2
Vertical Origin	356
Vertical Origin 1	0
Vertical Zoom	72
Horizontal Incr	2

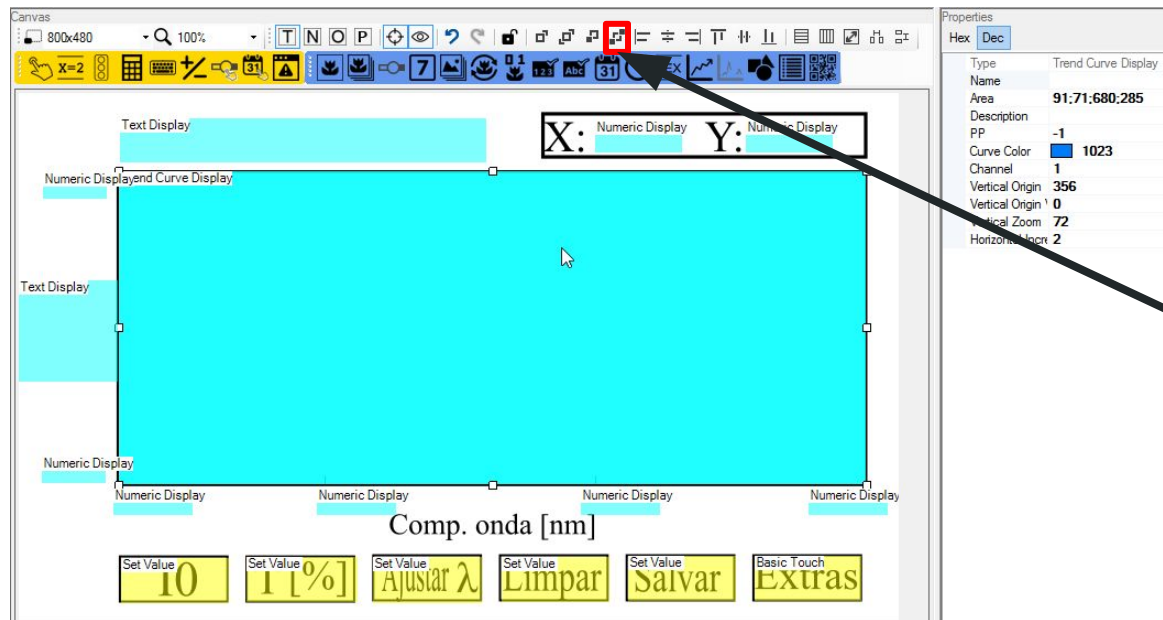
```
Lcm.writeTrendCurve2(y);
```

HorizontalIncrement define o total de pontos do plot:  $N = dx/horizontal + 1 = 680/2 + 1 = 341$

Vertical zoom é  $dy/delta\_valores * 256 = 285 * 256 / 1023 = 72$

O plot receberá valores inteiros na faixa de delta valores iniciando por 0

# TrendCurveDisplay - Plots sobrepostos



Clique no objeto TrendCurveDisplay, copie e cole no mesmo local e envie o objeto para trás apertando o botão em vermelho. Isto irá sobrepor os gráficos de canais diferentes em um mesmo espaço (o canal 2 é verde, o 1 é azul e o 0 é vermelho)