

Excel VBAとオシロスコープによる データ測定システムの構築

常三島技術部門 計測制御システムグループ^a

大学院社会産業理工学研究部 理工学域電気電子系^b

北島 孝弘 (KITAJIMA Takahiro)^a 桑原 明伸 (KUWAHARA Akinobu)^a

安野 卓 (YASUNO Takashi)^b 鈴木 浩司 (SUZUKI Hiroshi)^b

Keywords: Excel, VBA, Windows, Oscilloscope, Data-acquisition

1. はじめに

本稿では、Tektronix製オシロスコープ (TDS2014B) での測定値をエクセル (Excel 2016) で読み込むためのプログラム (VBA) について紹介する。

2. 測定器用制御ライブラリ

プログラムと測定器の通信は、測定器メーカーで共通規格となっている制御ライブラリ (VISA-COM) を介して行う。ライブラリには、測定器に対してデータを送受信するための関数が実装されており、測定器メーカーおよびインタフェース (USB, LAN, GPIB) に関係なくプログラムの作成ができる。本稿で使用するオシロスコープはTektronix製であるが、サンプルプログラムや参考資料の豊富さから、ライブラリはKEYSIGHTのIO LIBRARIES SUITEを利用する。なお、データ取得等の測定器を制御するためのコマンド (SCPI コマンド) のうち、機器固有のサブシステムコマンドは測定器により異なるので、製品マニュアルを参照して確認する必要がある。

3. 測定準備

今回は、パソコンとオシロスコープをUSB接続し、測定を行うための前準備として下記の項目を事前に行う。

1. KEYSIGHT Connection Expertで測定器のVISAアドレスを確認
2. Excelの開発タブを有効化
3. Excel Visual Basic Editorの参照設定で、「VISA-COM x.x Type Library」を追加 (x.x部分はIO LIBRARIES SUITEのバージョンで異なる)

4. 測定プログラム

オシロスコープ (Ch1) の測定値をエクセルシートに読み込むためのプログラムを表1に示す。なお、エクセルシートにはActive X ControlsのCommand Button (ここでのボタン名はCommandButton1) を配置しておく (本稿で紹介するプログラムの都合上、ボタンはセルJ2より右下へ配置)。エクセルシートに配置したCommand Buttonをダブルクリックすると、Visual Basic Editorのコード作成画面に移動できる。

プログラムにおける処理の流れを以下に示す。測定値取得で使用するコマンドはメーカーにより異なるので、対象機器の仕様書 (Tektronixの場合、Programmer manual) の確認が必要である。図1に測定結果の例を示す。

1. オシロスコープと接続
2. 測定パラメータの設定
3. 測定値取得、シートへ入力
4. グラフ表示
5. 測定終了処理

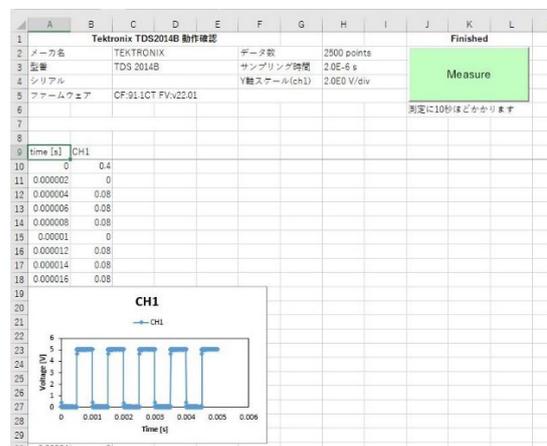


図1 測定後のエクセルシート画面例

表 1 プログラムソースコード

```

'オシロスコープから受信する文字列フォーマットは同型番でも異なる場合がある
'筆者らが試した機種では区切り文字でスプリットした後の配列で参照する要素の変更で対応した
Private Sub CommandButton1_Click()
    Dim ws As Worksheet      'Worksheet オブジェクトの作成
    Set ws = ActiveSheet     'アクティブシートオブジェクトを取得
    Dim RM As New VisaComLib.ResourceManager
                                'ResourceManager オブジェクトの作成
    Dim DMM As New VisaComLib.FormattedIO488
                                'FormattedIO488 オブジェクトの作成
    Set DMM.IO = RM.Open("USB0::0x0699::0x0368::C100370::0::INSTR")
                                'VISA アドレスを指定して計測器と接続

    On Error GoTo VisaComErr  'エラー発生時に実行
    DMM.IO.Timeout = 10000    'タイムアウトまでの時間(ms)

    ws.Cells(1, "J") = "Running" 'セルに現在のステータス表示
    ws.Cells(1, "J").Font.Color = RGB(255, 0, 0) '文字色(Red)を設定

    'オシロスコープの情報取得
    Dim id_str() As String, label() As String
    label = Split("メーカー名,型番,シリアル,ファームウェア", ",")
    DMM.WriteString "*IDN?" 'オシロスコープへコマンド送信(測定器の ID 取得)
    id_str = Split(Replace(DMM.ReadString(), vbLf, ""), ",")
                                'オシロスコープからデータ受信(Replace で改行コード削除)
    For i = 0 To UBound(id_str) '受信データをセルに書き込み
        ws.Cells(2 + i, "A") = label(i) '見出し
        ws.Cells(2 + i, "C") = id_str(i) '受信データ
    Next

    ' アクティブセルの取得(このコードでの測定値データの入力基準位置となる)
    Dim cell_row As Integer, cell_col As Integer
    Dim act_range As String
    cell_row = ActiveCell.Row      ' アクティブセルの行番号
    cell_col = ActiveCell.Column   ' アクティブセルの列番号
    act_range = ActiveCell.Offset(10, 0).Address ' グラフの表示位置(左上)

    ' 測定データ取得(CH1)
    Dim r_data As String
    Dim data() As String
    DMM.WriteString "DATa:SOUrce CH1" 'ソースはチャンネル 1 を指定
    DMM.WriteString "WFMPre:ENCdg ASC" 'ASCII フォーマットでデータ取得
    DMM.WriteString "DATa:STARt 1" '測定データの取得範囲(始点)
    DMM.WriteString "DATa:STOP 2500" '測定データの取得範囲(終点)

```

```

'オシロスコープの設定値取得, セルへ入力
Dim r_param As String
Dim param() As String
label = Split("データ数, サンプルング時間, Y 軸スケール(ch1)", ",")
For i = 0 To UBound(label)
    ws.Cells(2 + i, "F") = label(i)
Next
DMM.WriteString "WAVFrm?" 'オシロスコープへコマンド送信(設定値取得)
r_param = Replace(DMM.ReadString(), vbLf, "") 'データ受信
param = Split(r_param, ";")
ws.Cells(2, "H") = LTrim(Split(param(6), ",")(4)) '測定データ数
ws.Cells(4, "H") = LTrim(Split(param(6), ",")(2)) 'Y 軸スケール(ch1)
ws.Cells(3, "H") = Split(param(8), " ")(1) & " s" 'サンプルング時間
DMM.WriteString "DATA:SOURCE?" '(チャンネル番号取得)
ws.Cells(cell_row, cell_col) = "time [s]"
ws.Cells(cell_row, cell_col + 1) = Split(Replace(DMM.ReadString(), vbLf,
""), " ")(1)

'測定値の時間情報をセルへ入力
Dim delta_t As Double
delta_t = Val(Split(param(8), " ")(1)) 'サンプルング時間
For i = 0 To 2499
    ws.Cells(i + cell_row + 1, cell_col) = delta_t * i
Next
'量子化データから電圧値に変換するために必要な係数(3 個)
Dim value_y As Double, yoff As Double, ymult As Double, yzero As Double
ymult = Val(Split(param(12), " ")(1))
yzero = Val(Split(param(13), " ")(1))
yoff = Val(Split(param(14), " ")(1))
'測定値取得, セルへ入力
DMM.WriteString "CURVe?" '測定値取得コマンド
r_data = Replace(DMM.ReadString(), vbLf, "")
data = Split(r_data, ",")
For i = 0 To UBound(data)
    If i = 0 Then
        value_y = Split(data(0), " ")(1) '文字列":CURVE"を削除, 測定値を残す
    Else
        value_y = data(i) ' data(1) 以降はそのままセルに代入
    End If
    ws.Cells(i+cell_row+1, cell_col+1) = ((value_y-yoff)*ymult) + yzero
Next

'グラフ表示
Dim chartobj As Object, chart As chart
Dim p_left As Double, p_top As Double

```

```

Set chartobj = ws.ChartObjects.Add(Range(act_range).Left,
Range(act_range).Top, 300, 200) 'グラフ表示位置とサイズの指定
Set chart = chartobj.chart
With chart
    .ChartType = xlXYScatterLines
    .SetSourceData Range(Cells(cell_row, cell_col), Cells(cell_row +
    UBound(data) + 1, cell_col + 1))
    .Axes(xlValue, 1).HasTitle = True           ' 縦軸タイトル
    .Axes(xlValue).Format.Line.ForeColor.RGB = RGB(0, 0, 0)
    .Axes(xlValue, 1).AxisTitle.Text = "Voltage [V]"
    .Axes(xlValue).HasMajorGridlines = False   ' 縦軸目盛り線
    .Axes(xlCategory, 1).HasTitle = True       ' 横軸タイトル
    .Axes(xlCategory).Format.Line.ForeColor.RGB = RGB(0, 0, 0)
    .Axes(xlCategory).TickLabelPosition = xlTickLabelPositionLow
    .Axes(xlCategory, 1).AxisTitle.Text = "Time [s]"
    .Axes(xlCategory).Format.Line.Visible = False ' 横軸表示
    .HasLegend = True                          ' 凡例
    .Legend.IncludeInLayout = True
    .Legend.Position = xlTop
    .PlotArea.Format.Line.Visible = msoTrue    ' プロットエリア枠線
    .PlotArea.Format.Line.ForeColor.RGB = RGB(0, 0, 0)
End With

DMM.IO.Close           ' IO Resource オブジェクト解放
Set DMM = Nothing      ' FormattedIO488 オブジェクトの解放
Set RM = Nothing       ' ResourceManager オブジェクトの解放
Set chart = Nothing
Set chartobj = Nothing
ws.Cells(1, "J") = "Finished" ' ステータス表示
ws.Cells(1, "J").Font.Color = RGB(0, 0, 0)
Exit Sub

```

VisaComErr: 'エラー発生時に実行されるコード'

```

DMM.WriteString "*CLS"
DMM.IO.Close           ' IO Resource オブジェクト解放
Set DMM = Nothing      ' FormattedIO488 オブジェクトの解放
Set RM = Nothing       ' ResourceManager オブジェクトの解放
Set chart = Nothing
Set chartobj = Nothing
ws.Cells(1, "J") = "Error" ' ステータス表示
MsgBox "VISA COM エラー" & vbCrLf & Err.Number & ":" & Err.Description
ws.Cells(1, "J") = "Ready" ' ステータス表示
ws.Cells(1, "J").Font.Color = RGB(0, 0, 0)
End Sub

```