ABOUT THIS SAMPLE

This sample provides a ready-to-use Excel file that will log data from an Opto 22 *groov* View data store. A sample *groov* View project containing a data store is ready to load to a *groov* Server. Once loaded, the Excel file needs only the network address of the *groov* Server, and automated data logging starts with the press of a button. The Excel file will communicate with *groov* View through the *groov* RESTful API.

The following files are included:
- Opto22_groovRestAPI_datalogging.xlsm (Macro-enabled Excel spreadsheet)
- groovProject_dataLogging_opto_opto22.tar.gz

REQUIREMENTS
- *groov* Server for Windows
- Microsoft Excel

GETTING STARTED

You'll need to do the following tasks, which are explained below.
1. Back up your current *groov* View project. Record the login credentials for when you reinstall it later.
2. Load the sample *groov* View project using the command "Restore Project from Computer"
3. Find and select the project file groovProject_dataLogging_opto_opto22.tar.gz.
4. Provide these credentials:
- Username: opto
- Password: opto22

(Use lower case letters as shown.)
5. Save your project.

You are now ready to configure the *groov* Server's IP address in ExceL
6. Open the Excel spreadsheet.
7. Open the Configure worksheet.
8. Provide IP address of your *groov* Server in cell B2.
9. Open the Data Logging worksheet.
10. Click the Start Data Logging button.

GROOV VIEW

The *groov* View project uses a data store that the Excel file reads. The project has one HMI screen that allows you to modify the data store variables so you can see changes as Excel logs the data.

EXCEL DATA LOGGING SHEET

Before starting, use the Configure Tab to provide the IP Address of your *groov* server.

Data will appear at line 3. Older data will be moved down as new data appear.
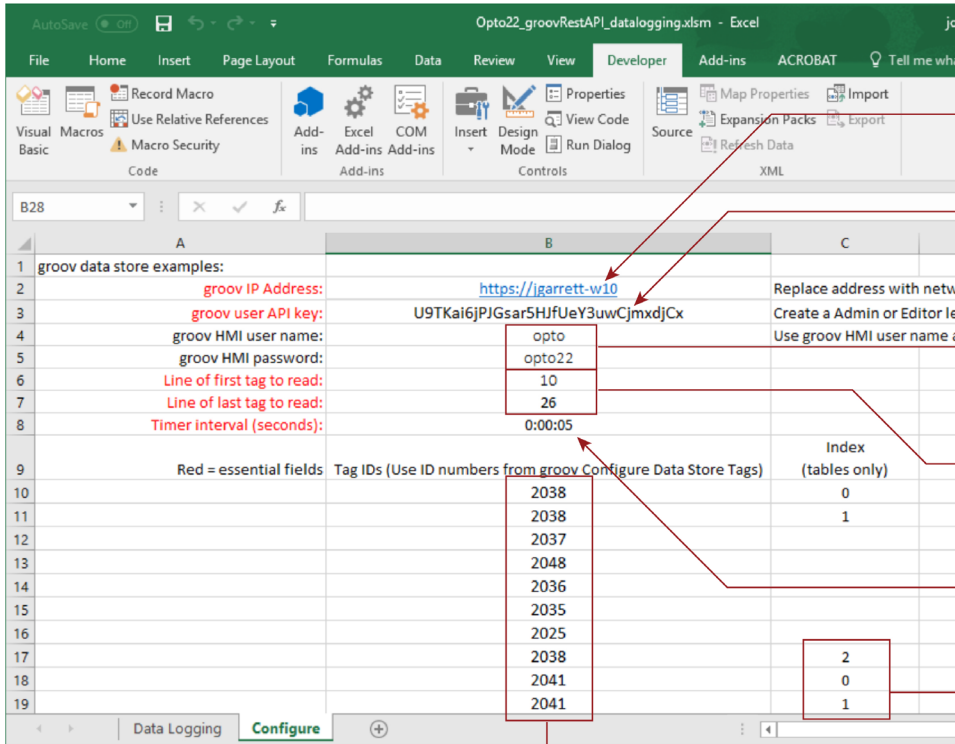
The PC's time stamp appears here.

Data will be written here in one column for each Tag ID listed on the Configure sheet.

Do not rename these tabs unless you also rename all references to the tabs in the Visual Basic code.

EXCEL CONFIGURE SHEET

The Configure sheet is set up to work with the Data Store of the sample project. To change *groov* Servers, change the IP Address and the user API key in fields B2 and B3. The API keys are found in *groov* View's Build mode -> Configure -> Accounts.

A sample Data Store is included with the sample project. Tag IDs are found in groov View's Build mode -> Configure -> Devices and Tags. Select or create a new Data Store, then choose Configure Tags.

Provide the IP address of the groov server or box that is running the groov project provided with this sample.

This API key is configured in the sample groov project for the user name "opto" with password "opto22". No change is needed.

The user name and password to the groov HMI are provided here for reference. These fields are not used by the programming code.

These fields indicate which consecutive lines in column B include tag IDs to be read by Excel. Change these values if you add or remove tag IDs.

Timer interval in date format. This sample is set to a 5-second interval. Use a longer interval if you expect delays contacting your groov server, e.g, if it's remote.

This sample gets only one value per ID (to be programmatically simple). Provide an index for any tables. Leave this field empty if the variable is not a table.
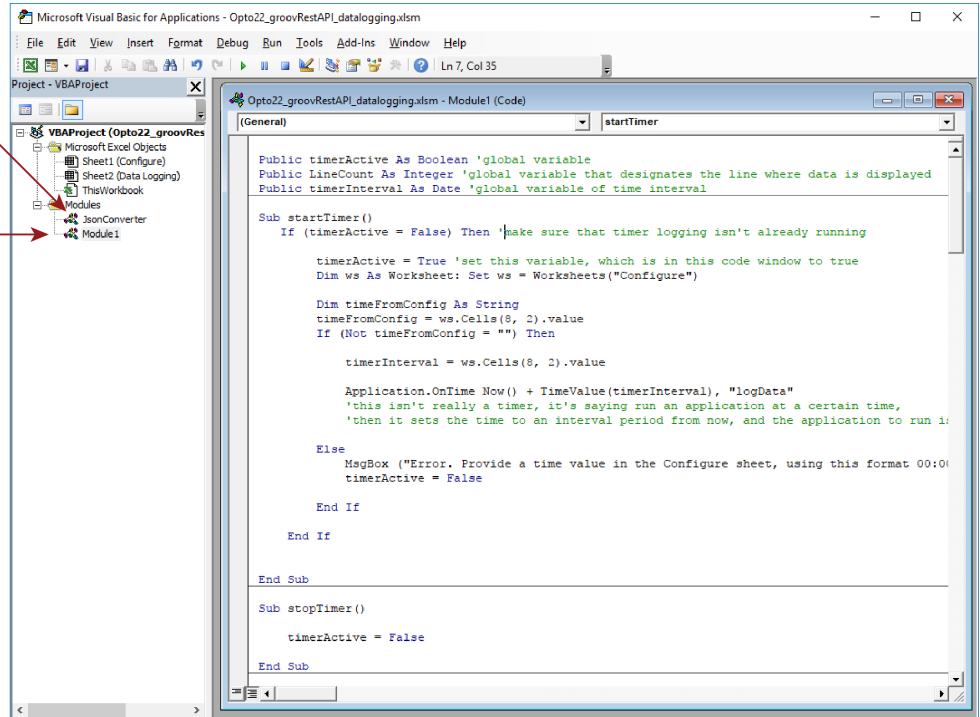
These are tag IDs from the sample groov project's data store. Change these to add or remove data fields from the data logging.

VISUAL BASIC

The Visual Basic code can be viewed by opening the Developer tab and selecting Visual Basic.

JsonConverter is a module written by MIT and made publicly available. It may be used according to terms of its copyright statement.

Module 1 contains all code for configuring and logging data.


```vba
Public timerActive As Boolean 'global variable
Public LineCount As Integer 'global variable that designates the line where data is displayed
Public timerInterval As Date 'global variable of time interval

Sub startTimer()
    If (timerActive = False) Then 'make sure that timer logging isn't already running

        timerActive = True 'set this variable, which is in this code window to true
        Dim ws As Worksheet: Set ws = Worksheets("Configure")

        Dim timeFromConfig As String
        timeFromConfig = ws.Cells(8, 2).value
        If (Not timeFromConfig = "") Then

            timerInterval = ws.Cells(8, 2).value

            Application.OnTime Now() + TimeValue(timerInterval), "logData"
            'this isn't really a timer, it's saying run an application at a certain time,
            'then it sets the time to an interval period from now, and the application to run is

        Else
            MsgBox ("Error. Provide a time value in the Configure sheet, using this format 00:0(
            timerActive = False

        End If

    End If

End Sub

Sub stopTimer()

    timerActive = False

End Sub
```

FOR MORE INFORMATION

Additional examples on using Excel with the *groov* RESTful API are available here:
http://developer.opto22.com/groov/examples/