

Übungen zu imc FAMOS II – Digital Kurs

- Block 4 -

Doc. Rev.: 1.1- 27.08.2025



Gezielter Wissenstransfer - höhere Produktivität

Übung A

Übungsziel:

Neben dem Schreiben von kompletten Messdatensätzen in Excel-Dateien ist es außerdem möglich Excel ferngesteuert aus FAMOS heraus zu betreiben. In dieser Übung soll eine Auswertung aller Messungen in einem Ordner durchgeführt und die Ergebnisse zusammengefasst in eine Excel-Datei übertragen werden.

Aufgabenstellung:

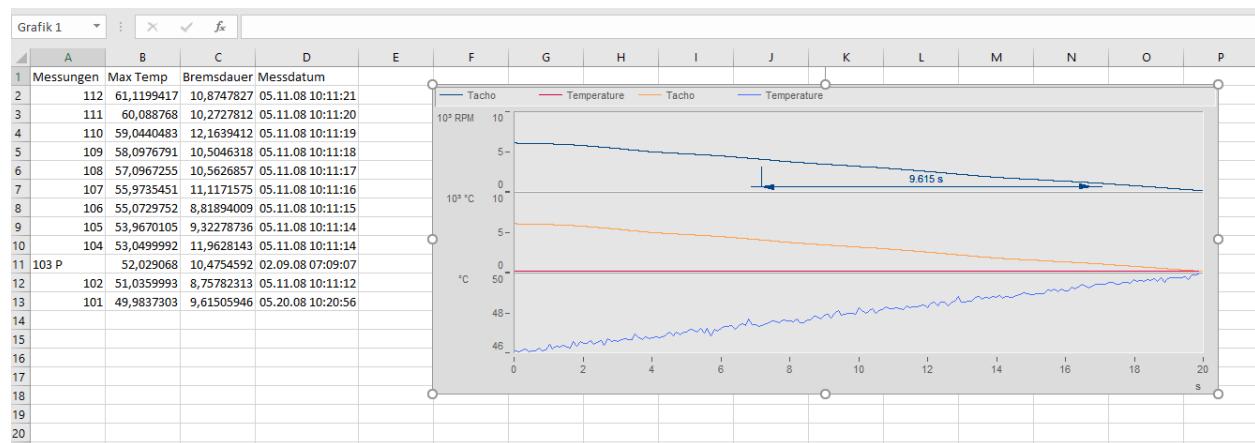
Bestimmen Sie für alle Messungen im Ordner **Test** die Maximaltemperatur sowie die Dauer des Bremsens von 4000 auf 1000 RPM und stellen Sie diese in einer Excel-Liste dar. Zudem sollen ebenfalls die Erstell-Uhrzeit des Kanals **Tacho** sowie der Verzeichnisname mit übertragen und in der Excel-Liste dargestellt werden.

Zusatz:

Fügen Sie ein Kurvenfenster mit dem Kanal **Tacho** und der **Temperatur** sowie einem Differenzmarker zwischen den beiden gesuchten Drehzahlen zur Excel-Tabelle hinzu.

Ergebnis:

Sie erhalten ein Excel-Worksheet mit den gewünschten Einträgen sowie ein Bild des erzeugten Kurvenfensters. Das Excel-Worksheet sieht in etwa wie folgt aus:



Übungsschritte:

- Öffnen Sie die Gruppe der Befehle für die Fernsteuerung von Excel in der Funktionsbibliothek. Sie finden diese unter **Dateien, Import/Export** in der Untergruppe **Excel (Erweitert)**. Erstellen Sie eine neue Sequenz, in der Sie mit geeigneten Befehlen zunächst Excel starten, sichtbar schalten und eine neue Arbeitsmappe erstellen.

```
err = XLStart()
XLVisible(1)
XLWbNew("")
```

- Erstellen Sie ein neues Texrarray, welches die Titel der Spalten (Messungen, Max Temp, Bremsdauer, Messdatum) beinhaltet.

```
titles = ["Messungen", "Max Temp", "Bremsdauer", "Messdatum"]
```

- Setzen Sie die Überschriften in der Excel Tabelle mit Hilfe einer Schleife und geeigneten Funktionen aus dem Excel-Kit:

```
For i = 1 To TxArrayGetSize(titles)
    reference = XLBuildA1Ref(1,i,1,1)
    XLSetText(reference, titles[i])
End
```

- Laden Sie nacheinander alle .raw-Daten aus den Unterverzeichnissen im Ordner **Test** aus den Beispieldaten. Orientieren Sie sich dabei an der Übung C aus Block 3.

Tipp: Anstatt alle Dateien in den Unterverzeichnissen erst aufzulisten und anschließend in einer Schleife nacheinander zu laden, kann stattdessen die **FileLoad()** Funktion verwendet werden, um alle Dateien auf einmal zu laden. Auf ein Messungsbezug kann in diesem Fall verzichtet werden. Da die weitere Verarbeitung noch in derselben Schleife durchgeführt wird, können die entsprechenden Kanäle beim nächsten Schleifendurchlauf wieder überschrieben werden.

```
List_of_dirs = FsFileDialogNew(basepath , "*",1, 0, 2)
number_of_dirs = FsFileDialogGetCount(List_of_dirs)
For i = 1 to number_of_dirs
    act_dir = FsFileDialogGetName(List_of_dirs, i)
    FileLoad(act_dir + "\*.raw","",0)
    ; weiterer Code
End
FsFileDialogClose(List_of_dirs)
```

- Berechnen Sie das Maximum des Kanals **Temperature** sowie die Bremsdauer von 4000 RPM auf 1000 RPM. Ermitteln Sie außerdem den Triggerzeitpunkt von Tacho sowie das aktuelle Messungsverzeichnis:

```
maxTemp = Max(Temperature)
braking_duration = Pos(Tacho, 1000) - Pos(Tacho, 4000)
meas_date = Time?(Tacho)
measurement = FsSplitPath(act_dir, 2)
```

- Übertragen Sie die Werte in die Excel Tabelle unter Verwendung geeigneter Funktionen aus dem Excel-Kit. Beachten Sie dabei den jeweiligen Variablenotyp:

```
reference = XLBuildA1Ref(i+1,1,1,1)
XLSetText(reference, measurement)
reference = XLBuildA1Ref(i+1,2,1,1)
XLSetValue(reference, maxTemp,0)
reference = XLBuildA1Ref(i+1,3,1,1)
XLSetValue(reference, braking_duration,0)
reference = XLBuildA1Ref(i+1,4,1,1)
XLSetValue(reference, meas_date,1)
```

- Setzen Sie nach dem Abarbeiten der Schleife noch das Format für die Spalte **Messdatum**:

```
reference = XLBuildA1Ref(2,4, number_of_dirs,1)
XLSetCellFormat(reference, "tt.mm.jj hh:mm:ss")
```

- Speichern Sie die Excel Datei im Pfad **basepath** und vergeben Sie einen geeigneten Dateinamen:

```
XLsName = basePath + "\summary.xlsx"
XLWbSave(XLsName, 0)
```

- Schließen Sie die geöffnete Excel Datei am Ende der Sequenz und speichern Sie diese ab.

```
XLWbClose()
```

Zusatz: Übertragung eines Kurvenfensters in eine Excel Datei.

- Erstellen Sie ein neues Kurvenfenster und konfigurieren Sie dieses nach eigenen Wünschen. Falls Sie ein Kurvenfenster per Sequenz erzeugen möchten, finden Sie im am Ende dieser Übung einen beispielhaften Codeblock.
- Kopieren Sie das Kurvenfenster mit der Funktion **CwAction()** und fügen Sie es anschließend in der Exceltabelle ein. Diese Befehle müssen logischerweise vor dem Schließen der Excel Datei ausgeführt werden:

```
CwAction("clipboard.copy")
XLPaste("f2")
```

Codeblock zur Erstellung eines passenden Kurvenfensters zur Übung. Die Erläuterung der einzelnen Schritte ist nicht Teil der Übung, daher bleibt dieser Block an dieser Stelle unkommentiert:

```
CwNewWindow(Tacho, "show")
CwNewChannel("append Last axis", Tacho)
CwNewChannel("append new cosys", Temperature )
CwSelectByIndex("cosys", 1)
CwNewElement("marker.abs")
CwMarkerSet("x", Posi(Tacho, 1000))
CwMarkerSet("y", 1000)
CwMarkerSet("arrow", 0)
CwMarkerSet("text", "")
CwMarkerSet("LineLength", 0)
CwNewElement("marker.abs")
CwMarkerSet("x", Posi(Tacho, 4000))
CwMarkerSet("y", 4000)
CwMarkerSet("arrow", 0)
CwMarkerSet("text", "")
CwMarkerSet("LineLength", 0)
CwNewElement("marker.hori.dimline")
CwMarkerSet("ref.1", 2)
CwMarkerSet("ref.2", 1)
```

Hinweis: Sie finden in den Beispieldaten eine vollständige Lösungssequenz zu dieser Übung (**excel_transfer.seq**). In der Sequenz wird eine Gruppe **gr** verwendet, um nicht direkt notwendige Variablen in der Variablenliste zu verstecken bzw. nachträglich mit der Befehlszeile **Local gr** auch als lokale Variablen zu definieren.

Übung B

Übungsziel:

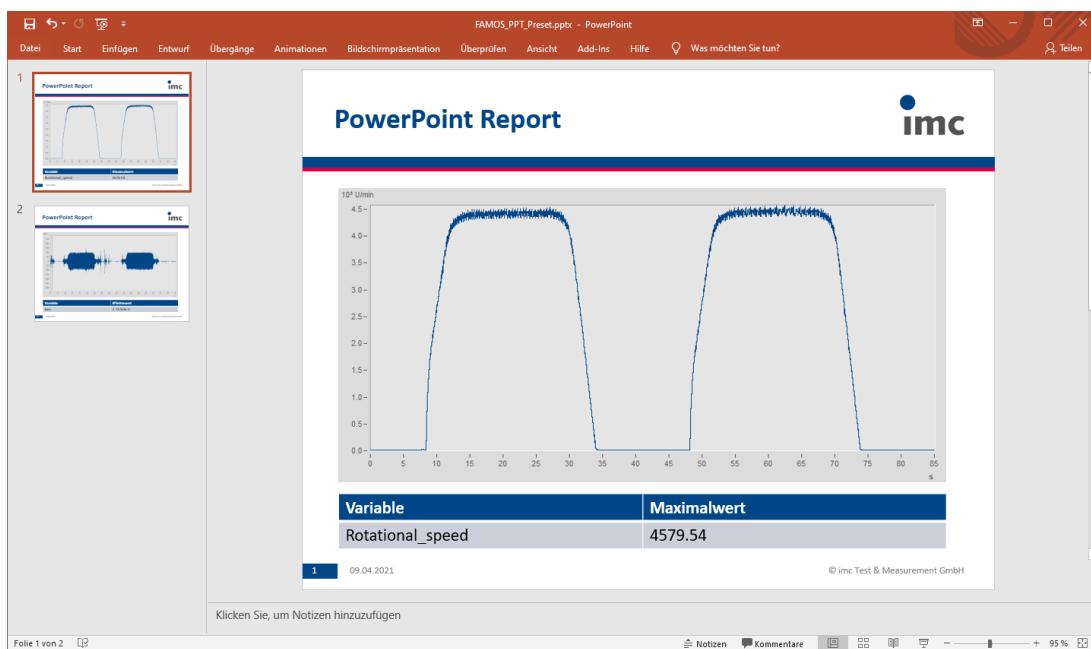
In dieser Übung lernen Sie wie man mit Hilfe des FAMOS PowerPoint-Kits Microsoft PowerPoint fernsteuert und Inhalte per Sequenz in eine PowerPoint Präsentation überträgt.

Aufgabenstellung:

Laden sie das erste Verzeichnis aus den Beispieldaten **Serial_Measurement\2020-11-06 08-54-21 (1)** und füllen eine Powerpoint (**FAMOS_PPT_Preset.pptx** aus den Beispieldaten) als Report mit einer Darstellung der Drehzahl **Rotational_speed** auf der ersten Folie, sowie dessen maximalen Drehzahl als Zahl in der darunter stehenden Tabelle. Löschen Sie dann die vorhandene Folie Nummer 2 und fügen die erste Folie erneut ein. Füllen Sie diese anschließend mit dem Datensatz **Geo** und dessen Effektivwert als Zahl.

Ergebnis:

Sie erhalten eine PowerPoint Präsentation mit 2 Folien, die jeweils ein Kurvenfenster und eine Tabelle mit Ergebnissen beinhalten.



Übungsschritte:

Um PowerPoint aus FAMOS heraus fernsteuern zu können, muss zunächst eine Präsentation als Vorlage erstellt werden. Die einzelnen Elemente auf einer PowerPoint Folie lassen sich über deren Eigenschaft **Alternativtext** mit dem PowerPoint Kit ansprechen. In dieser Übung kommt dabei die vorbereitete Vorlage **FAMOS_PPT_Preset.pptx** aus den Beispieldateien zum Einsatz.

- Öffnen Sie die Vorlage **FAMOS_PPT_Preset.pptx** aus den Beispieldateien in PowerPoint. Diese enthält eine Folie mit einem Titel-Textfeld sowie eine Grafik als Platzhalter. Öffnen Sie die Bearbeitungsleiste für den **Alternativtext** und finden Sie die vergebenen Werte für die beiden Platzhalter heraus.

- Schließen Sie PowerPoint und wechseln Sie zu FAMOS.
- Laden Sie alle Dateien aus dem ersten Verzeichnis aus den Serial Measurement Beispieldaten **Serial_Measurement\2020-11-06 08-54-21 (1)\:**

```
FileLoad(basepath + "\Serial_Measurement
2020-11-06 08-54-21 (1)\*.raw", "", 0)
```

basepath ist dabei eine Variable, die den Pfad zu den Beispieldateien auf Ihrem PC enthält.

- Das PowerPoint-Kit finden Sie in den Funktionen unter der Gruppe **Präsentation**. Wechseln Sie zum Eingabefenster und öffnen Sie die Vorlage mit dem Befehl:

```
res = PptOpenPresentation(basepath + "\FAMOS_PPT_Preset.pptx", 1)
```

Bei erfolgreicher Ausführung sollte sich PowerPoint geöffnet haben.

- Suchen Sie nach dem Index der Folie, welche das Element mit dem Alternativtext **CurveWindow** enthält:

```
index = PptFindSlideByAlternativeText(1, "CurveWindow")
```

Da die geöffnete Vorlage nur eine Folie enthält, kommtt hier logischerweise 1 für den Index heraus.

- Ersetzen Sie den Text des Textfeldes mit dem Alternativtext **Headline** auf der zuvor gefundenen Folie mit dem Text **PowerPoint Report**:

```
PptSetText(index, "Headline", "PowerPoint Report")
```

- Öffnen Sie den Kanal **Rotational_speed** in einem Kurvenfenster. Setzen Sie anschließend die Optionen zum Selektieren des aktuellen Kurvenfensters und ersetzen Sie die Grafik auf der PowerPoint Folie durch ein Bild des aktuellen Kurvenfensters:

```
Show Rotational_speed
CwSelectMode("newest")
CwSelectWindow("")
PptSetCurve(index, "CurveWindow", 1, 0, "screen")
```

- Berechnen Sie das Maximum von **Rotational_speed** und befüllen Sie die Tabelle auf der PowerPointFolie mit den entsprechenden Werten:

```
;Maximum berechnen und in Tabelle in Präsentation eintragen
maximum = Max(Rotational_speed)
res = PptSetCellText(index, "Table", 2, 1, "Rotational_speed")
res = PptSetCellText(index, "Table", 1, 2, "Maximalwert")
res = PptSetCellText(index, "Table", 2, 2, TForm(maximum, ""))
```

Das Kurvenfenster kann im Anschluss wieder geschlossen werden.

- Löschen Sie die zweite Seite der Präsentation:

```
res = PptDeleteSlide(index + 1)
```

- Fügen Sie eine neue Folie ein, indem Sie die erste Folie der Vorlage noch einmal an die geöffnete Präsentation anhängen:

```
res = PptAddSlides(index, basepath + "\FAMOS_PPT_Preset.pptx", 1, 1)
```

- Ersetzen Sie den Titeltext der neu hinzugefügten Folie wieder durch **PowerPoint Report**:

```
index = index +1  
PptSetText(index, "Headline", "PowerPoint Report")
```

- Stellen Sie den Kanal **Geo** in einem Kurvenfenster dar, berechnen Sie dessen RMS Wert mit der Funktion **RMS()** und fügen Sie die Ergebnisse auf der 2. Seite der PowerPoint Präsentation ein. Orientieren Sie sich dabei an den vorherigen Schritten:

```
Show Geo  
CwSelectMode("newest")  
CwSelectWindow("")  
PptSetCurve(index, "CurveWindow", 1, 0, "screen")  
rms_value = RMS(Geo)  
res = PptSetCellText(index, "Table", 2, 1, "Geo")  
res = PptSetCellText(index, "Table", 1, 2, "Effektivwert")  
res = PptSetCellText(index, "Table", 2, 2, TForm(rms_value, ""))
```

- Speichern Sie die erstellte Präsentation unter neuem Namen ab (z.B. Report.pptx):

```
res = PptSavePresentation(basepath + "\Report.pptx")
```

- Schließen Sie die Präsentation:

```
res = PptClosePresentation()
```