

Projektarbeit LabVIEW Workshop  
Kleinprojekt(2er Teams)

# Batterietester

Daniel Begert

10.11.2022

Fachhochschule Nordwestschweiz  
Hochschule für Technik  
Institut für Automation

## 1 Eckdaten

**Aufgabe:** LabVIEW Workshop Kleinprojekt, «Batterietester»

**Abgabe:** Die Abgabe erfolgt bis spätestens Sonntag 18.12.2022 23:59:59 GMT+1 per E-Mail

**E-Mail:** daniel.begert@fhnw.ch

**Abgabeform:** Zip-komprimierter Ordner mit LV Projekt, allen Files inklusive Software-Diagramm. Benennen Sie den Ordner wie folgt: Name1-Name2-1Sa-labw22HS.zip

**Teams:** Die Aufgabe wird in Einzelarbeit oder als Zweier-Team durchgeführt

**Bewertung:** Die Bewertung erfolgt nach den Bewertungskriterien in Kapitel 5

**Plagiate:** Werden Codeteile Dritter, die über die in LabVIEW implementierten Funktionen hinausgehen, übernommen (Internet Foren, Codebeispiele, etc.), müssen diese zitiert werden. Ein Fehlen einer solchen Quelle, wird als Plagiat gewertet und hat die Note 1 zur Folge

**Plagiate bereits existierende Projektarbeiten «Batterietester»:** Werden alte Projektarbeiten des Batterietesters (oder Teile davon) kopiert, wird die Note 1 vergeben

## 2 Aufgabe

Erstellen Sie eine LabVIEW-Applikation, mit der Sie den Ladezustand einer AA oder AAA Batterie bestimmen können. Die Applikation umfasst ein einfaches GUI, die Datenerfassung und die Datenauswertung. Die Grenzwerte (Strom, Spannung, Innenwiderstand), in welchen ein bestimmter Batterietyp noch als gut/schlecht gilt, sollen in einem ASCII File gespeichert sein. Nachdem die Batterie getestet wurde, soll dem User sofort visuell dargestellt werden, ob die Batterie noch gut oder schlecht ist, inklusive der verbleibenden elektrischen Kapazität in Prozent.

Nach jeder Messung soll gewählt werden können, ob das Resultat in einem File archiviert werden soll. Wird die Wahl getroffen, so soll das Messresultat ins File geschrieben werden (alle Resultate in einem File). Des Weiteren soll ein Zähler alle durchgeführten Messungen zählen. Auch wenn das Projekt geschlossen und neu geöffnet wird, soll der Zähler auf dem letzten aktuellen Zählerstand weiterzählen. Dieser Zähler kann zurückgesetzt werden, jedoch nur durch die Eingabe eines Passwortes (admin2022).

An die Aufgabe werden folgende **Rahmenbedingungen** gestellt:

- Es ist ein NI myDAQ und der Batterie-Print zu verwenden
- Die Umsetzung muss in einem LV-Projekt erfolgen
- Es ist eine State Machine Architektur zu wählen
- Es sind mindestens 3 SubVIs zu erstellen
- Der Benutzer steuert den Batterietester über das GUI
- Das GUI soll das FHNW Logo enthalten
- Das Testresultat soll vom User einfach ablesbar sein (Batterie Gut/Schlecht?, Ladung in Prozent)
- Es ist min. ein Plot zu erstellen (z.B. Entladekurve)
- Pfade sollen alle relativ sein, sodass sich der User nicht um die Wahl des richtigen Files kümmern muss
- Dokumentieren Sie den Code sinnvoll
- Zusätzlich zum Code muss ein Software-Diagramm (Zustandsdiagramm) erstellt und abgegeben werden

### 3 I/O

Die I/O-Liste für den Print:

Batteriespannung:	<b>AI 0</b>
Batteriestrom:	<b>AI 1</b>
Batterie belasten:	<b>Port 0 / Line 0</b>

### 4 Vorgehen

Bevor Sie mit der Programmierung starten, definieren Sie diese Punkte:

- Wie ist der Print aufgebaut?
- Wie könnte damit der Ladestand ermittelt werden?
- Welche Anzeige- und Bedienelemente benötigt der Benutzer? Wie erfolgt die Bedienung?
- Skizzieren Sie den Messablauf
- Wie gehen Sie mit Fehler um? (Software- und Bedienungsfehler)

## 5 Bewertungskriterien

Benötet werden folgende Aspekte:

- Funktionalität der Lösung
- Übersichtlichkeit der Programmierung
- Einhalten der Programmierregeln
- Umsetzung der Programmstruktur
- UI Design
- Usability
- Kreativität
- Error Handling
- Dokumentation (Code)
- Software-Diagramm