

eTivity D – Aufgabestellung Software

Modul 11 / SoSe 2024

Hintergrund: das Prüfgerät LilonCheck9V125mA führt eine automatische Kapazitätsmessung an einem zu prüfenden Akku („device under test“, DUT) durch. Es sendet die Messdaten über eine serielle Schnittstelle an einen Laptop. In eTivity D soll eine Software-Applikation realisiert werden, welche folgende Aufgaben erfüllt

- a)** Eingabe des Identifiers für den Akku („DUT-ID“) durch den Benutzer.
- b)** Lesen der Messdaten über die serielle Schnittstelle.
- c)** Automatische Prüfung der Messdaten (Werte innerhalb der Spezifikation? Plausibel?)
- d)** Automatische Erstellung & Speicherung eines kurzen, übersichtlichen Prüfprotokolls (pass/fail?)
- e)** Optional – Speicherung der Rohdaten (offene Diskussion ob zweckmäßig?)

Fehler in der zu realisierende Software-Applikation sind kritisch. Sie können zu falschen Entscheidungen (Verwendung fehlerhafter Akkus, Ausschluss von guten Komponenten) führen. Es ist daher notwendig, die Software-Applikation zu testen („testen der Prüf-Applikation“). Um dies zu bewerkstelligen, werden Testdaten zur Verfügung gestellt, welche über Files eingelesen werden können.

Somit kann die Software-Applikation z.B. über eine virtuelle serielle Schnittstelle oder ein reales serielles Buffer-Device (z.B. ein Arduino Board, welches die Testdaten empfängt und wieder über die serielle Schnittstelle zurückschickt) getestet werden. Es müssen also zwei Dinge realisiert werden: i) die Software-Applikation selbst und ii) die Testumgebung, welche Testdaten in die von ihr *getrennte* Applikation einspeist.

Aufgabenstellung

- I)** Realisierung der Software-Applikation (Tipp: eine eindeutige Bezeichnung ermöglicht eindeutiges Referenzieren in der Dokumentation)
- II)** Realisierung der Testumgebung (Tipp: Bezeichnung)
- III)** Dokumentation der entwickelten Software-Applikation (Zweck, Anforderungen, Implementierung, Testfälle und Ergebnisse Schlussfolgerungen → z.B. ein kurzer Engineering Report bietet sich an).

Anmerkung: es wird im Team gearbeitet → einigen Sie sich auf die Aufgabenteilung und Zuständigkeiten

Einige praktische Hinweise:

Ad a) Es braucht nur ein Fenster, in dem die wesentlichen Daten in einer Zeile eingegeben werden können. Zum Beispiel kann die DUT-ID `BatXDevYyyyymmdd` verwendet werden, um die Prüfung des Akkus `X` mit dem Gerät `Y` im Jahr `yyyy` Monat `mm` am Tag `dd` zu dokumentieren. Es empfiehlt sich, die Eingabe schlank zu halten. Wichtig ist, dass die DUT-ID automatisch und korrekt ihren weiteren Weg in die gesamte Dokumentation findet.

Ad b) Prüfgerät LilonCheck9V125mA ist im zugehörigen ER ausführlich beschrieben.

Kapitel 5.1 beschreibt die Testphasen.

- In der Waiting-Phase wird die DUT-ID eingegeben. Erst danach überträgt das Prüfgerät Daten.
- Nach dem Voltage Test sendet das Prüfgerät in einer Zeile zwei Messwerte (Spannung Strom) im ascii Format.
- In der Discharge Phase werden alle acht Sekunden zwei Mittelwerte (Spannung Strom) im ascii Format gesendet.
- In der finalen End-Of-Test Phase werden keine Daten mehr übertragen.

Ad c) Table 2 listet die relevanten Spezifikationen. Die Software-Applikation soll prüfen:

- Liegt die Spannung immer im spezifizierten Bereich?
- Liegt der Strom immer im spezifizierten Bereich?
- Wird die Kapazität erreicht?
- Gibt es eine erkennbare Unplausibilität in den Daten? (Halten sie diesen Punkt einfach.)

Ad d) Das Ergebnis soll automatisch auf einer DINA4 Seite dargestellt werden (muss nicht schön formatiert sein). Folgende Infos müssen ersichtlich sein.

- Bild der Entladekurve (z.B. ähnlich wie Bild 7 im ER-LilonCheck9V125mA)
In weiteren Zeilen soll folgende Info stehen:
- Mindestkapazität, Ist-Kapazität -> automatische Entscheidung pass-/fail
- Grenzwerte Leerlaufspannung, Ist-Wert -> pass/fail
- Grenzwert Leerlaufstrom, Ist-Wert -> pass/fail
- Maximale Entladespannung, Ist Wert -> pass fail
- Minimale Entladespannung, Ist Wert -> pass fail
- Gerne können wenige Zeilen für Meldungen vorgesehen werden (z.B. unplausibler Anstieg der Spannung)
- Die Protokolle sollen mit Unterschriften freigegeben werden können.

Es werden Files mit Messdaten zur Verfügung gestellt, um die Software-Applikation prüfen zu können. Diese Daten wurden Großteils aus Appendix E übernommen. **Wichtig:** Es wird **nicht** gefordert, dass erkannt wird welcher Batterietyp, welche Daten erzeugt. Es geht darum, ob die Daten innerhalb oder außerhalb der Spezifikation liegen.

Überlegen Sie sich, welche möglichen Fehlerfehler durch die zur Verfügung stehenden Testdaten nicht prüfbar sind. Wie könnte das verbessert werden?