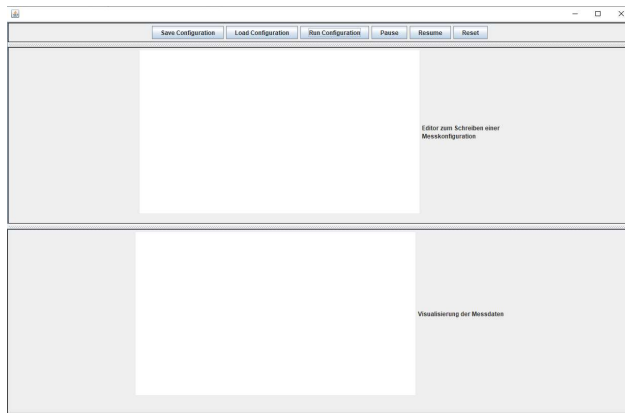


Aktivitäten der letzten Tage

Linus

Aufwertung der GUI. Jetzt sind folgende Buttons vorhanden:

Save, Load, Run, Pause, Resume, Reset



Ausblick über die nächsten Schritte:

1. Konfigurationen im Editor schreiben. Diese werden dann in Yaml Files übersetzt.
2. Angeben welche Blöcke man verwenden will. Dieser Aufbau wird dann in ein Yaml File übersetzt.
3. Vollständige Konfiguration kann im Editor erstellt und gespeichert werden.

Jan

Wie kann man die Daten von der Konsole auslesen? Diese Frage beantwortet, aber noch nicht mit einem Raspberry Pi ausgetestet.

Das Backend liest aktuell nur Daten aus Dateien aus. Die Implementierung von Threads läuft gerade.

Leon

Der Controller kann jetzt die grundlegenden Funktionen:

Save, Load, Run

an das Model weiterleiten.

Stefan

Der Cache ist jetzt in Bearbeitung. Wenn er fertig ist wird er sich um die zeitliche Steuerung des Messablaufes kümmern.

Beschreibung des Integrationstest mit den verschiedenen Mocks, die verwendet wurden.

David

Das Model kann jetzt Datenpakete verarbeiten und durch eine Messkonfiguration laufen lassen.

Als nächstes wird das Laden und Speichern von Yaml Files in Angriff genommen werden.

Ein Problem, das noch besteht ist das direkte Weiterleiten von Datenpakete an die GUI. Vorschlag von Stefan: Da man auch von Dateien auslesen kann, in die geschrieben wird, könnte man das alles über die Datei laufen lassen.

Die Repräsentation, die aktuell existiert, nimmt erst alle Messdaten entgegen und schreibt sie anschließend in eine .csv Datei.

Problem der Unabhängigkeit der Konfigurations Yaml Files

Aktuell sind in den Konfigurations nur die Verweise auf die Building-Block-Yamls. Das heißt die Konfigurations-Yamls können nicht ohne die Building-Block-Yamls funktionieren. Hier sollten wir überlegen ob es nicht eine Lösung gibt, das Problem löst.

Pausieren Problematik

Wie schaffen wir es, dass der Messlauf pausiert werden kann?

Unterschiedliche Ansätze:

1. Daten vom Raspberry Pi ignorieren und die Pausierung vollständig in FreeJDAQ abhandeln.
2. Pausier-Aufforderung an den Raspberry Pi weiterschicken und die Pause-Funktionalität von PhyPiDAQ nutzen.

In dem Zusammenhang ist es noch wichtig zu entscheiden, was mit den bis dahin mit den bereits im Cach befindlichen Daten passiert.

ToDo's

Die neuen ToDo's können im Trello eingesehen werden.