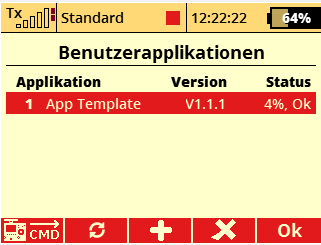
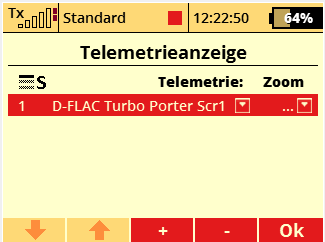
**Test Handling Turbinenmodelle:**

1. **Installation:**

Lupa App über setup batch installieren, bei Registrierung alle zusätzlich (auch leere Apps) deinstallieren. **Es darf nur die App Template aktiv sein!!!**

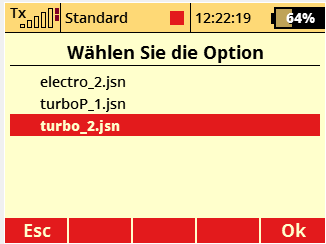


Dann Telemetriefenster registrieren



1. **Datafile laden:**

Für jedes Modell wird ein jsn file mit Modellname.jsn angelegt. In diesem File werden alle Sensorkonfigurationen gespeichert. Bei Inbetriebnahme wird das erste Datafile aud der Templateliste geladen.

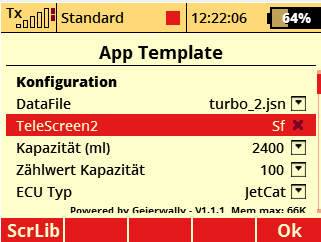


turboP\_1 ist ein Template für nur eine Telemetrieseite

turboP\_2 ist ein Template für aufgeteilt auf zwei Telemetrieseiten.

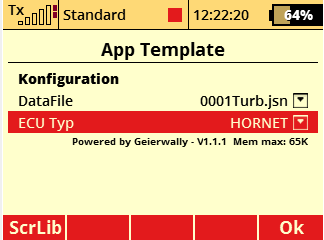
Electro\_2 ist ein Template für Elektromodelle auf zwei Telemetrieseiten

1. **Schalter zur Umschaltung der Telemetrieseite konfigurieren (steht nur bei Doppelseiten zur Konfiguration):**

****

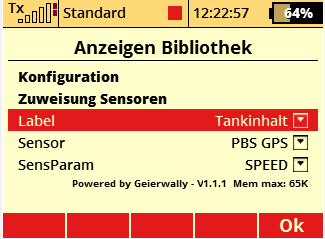
Konfiguration von zwei getrennten Telemetrieseiten ist aus Speichergründen auf den 14/16-er Sendern nicht möglich. Aus diesem Grund wurde das Konzept über Schalter implementiert. Das hat auch den Vorteil, dass man z.B. zwei Seiten in Abhängigkeit des Flugzustandes schalten kann.

1. **ECU Config laden (nur bei turboP Modellen):**



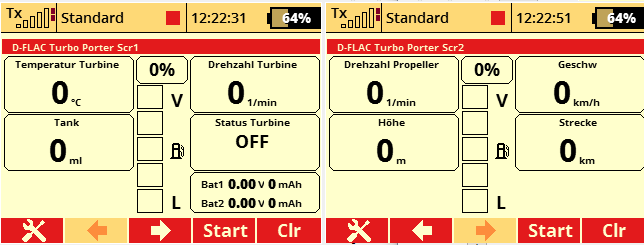
Turbine im ECU Config auswählen für Statusanzeige

1. **Sensoren zuweisen:**

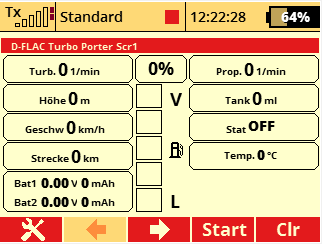


Jedem Rahmenlabel den entsprechenden Sensor zuweisen. Bei Erstinbetriebnahme ist zum Einlesen der Sensoren erforderlich, den Empfänger einzuschalten. Dabei kann kommt es bedingt durch fehlende Sensorzuweisung zu akustischen Alarmausgaben. Diese werden zurückgesetzt wenn alle Sensoren gebunden sind.

1. **Prüfung der Telemetriehauptseiten:**

****

Die Hauptseiten bei geladenem turboP\_2.jsn Template (zwei Telemetrieseiten)



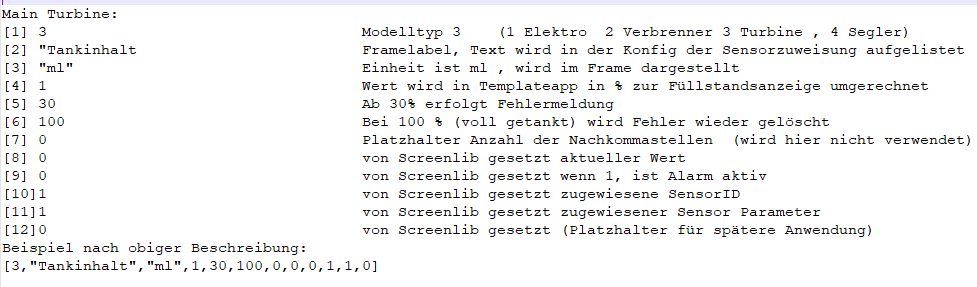
Hauptseite bei geladenem turboP\_1.jsn Template (eine Telemetrieseite)

**Von jeder geladenen Konfiguration bitte zur Ermittlung des Speicherbedarfes Screenshots der Konsole speichern**

1. **Hinweise zur Datafile – Struktur der konfigurierten Frames:**

Die verwendeten Frames sind wie folgt definiert:

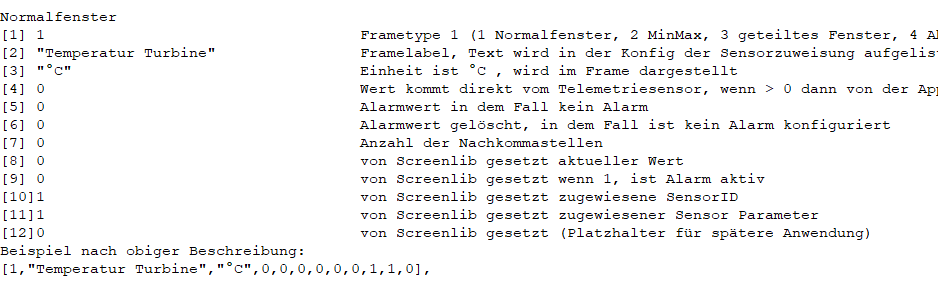
**Mittelteil Tankanzeige:**

****

[3,"Tankinhalt","ml",1,30,100,0,0,0,1,1,0]

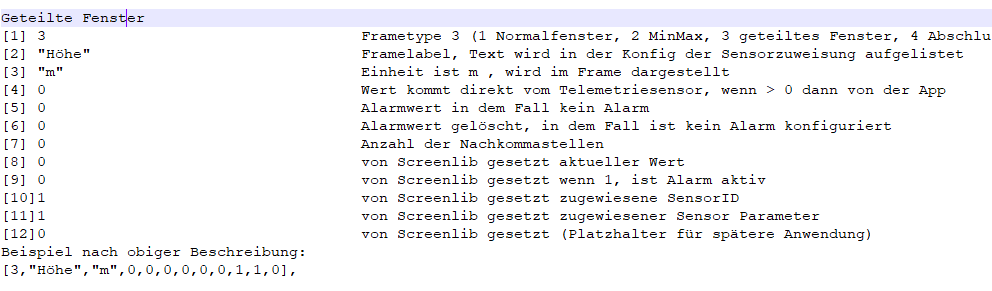
****

**Frametype 1 Normalfenster :**

****

[1,"Temperatur Turbine","°C",0,0,0,0,0,0,1,1,0],



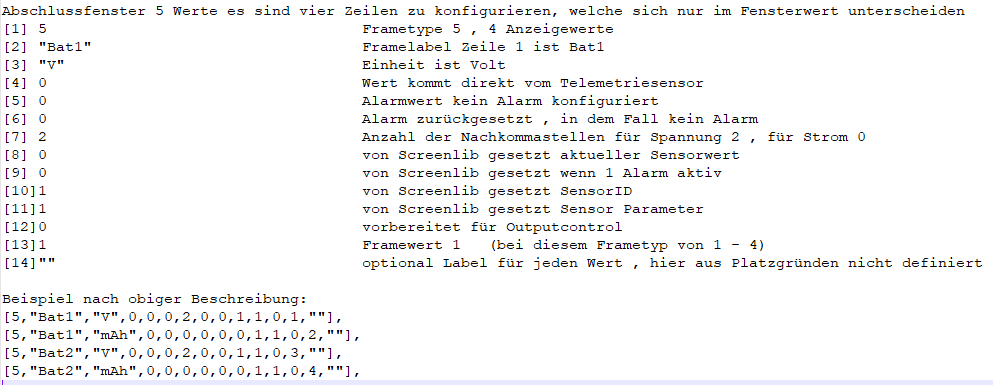
**Frametype 3 geteiltes Fenster :** 

[3,"Höhe","m",0,0,0,0,0,0,1,1,0],



**Frametype 5, Abschluss mit 4 Anzeigewerten:**

Jedem Frame ist in der scrlib config ein Sensorparameter zuzuordnen!!!

****

[5,"Bat1","V",0,0,0,2,0,0,1,1,0,1,""],

[5,"Bat1","mAh",0,0,0,0,0,0,1,1,0,2,""],

[5,"Bat2","V",0,0,0,2,0,0,1,1,0,3,""],

[5,"Bat2","mAh",0,0,0,0,0,0,1,1,0,4,""],

****