

# Falcon BMS to Arduino Interface Tool (BMSAIT)

## Beispielprogramm Schalter und Taster



Autor	Robin „Hummer“ Bruns
Dokumentversion	1.0
Softwareversion	1.0
BMS Version	4.34u3
Datum	3.10.2020

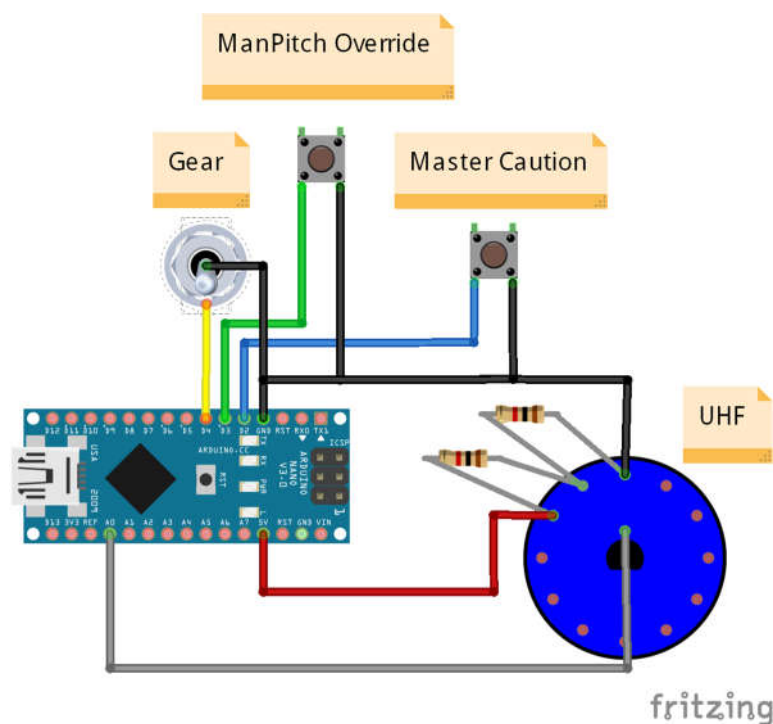
## 1. Überblick

Hier soll die Möglichkeit demonstriert werden, BMSAIT als Eingabegerät zu nutzen. Das Modul Switches ermöglicht es dabei, verschiedene Arten von Schaltern und Tastern auszulesen und in Signale für die Steuerung des FalconBMS umzuwandeln.

Um das Beispiel auszuprobieren benötigt ihr:

- Ein Arduino-Board (z.B. ein UNO)
- Zwei Taster
- Ein Kippschalter (ON-ON)
- Ein Drehschalter
- Zwei Widerstände (1kOhm)
- Verbindungskabel

## 2. Verkabelung



Für ein einfaches Beispiel zur Nutzung des Moduls benötigt ihr je einen Druckknopf an PIN 2 und PIN 3 sowie einen Kippschalter an PIN 4. Widerstände sind übrigens nicht erforderlich. Diese sind im Arduino eingebaut und werden durch die Software aktiviert.

Ein komplexeres Beispiel kann demonstriert werden indem ein Drehschalter angeschlossen wird, bei dem über nur einen einzigen PIN am Arduino alle Rastungen des Schalters ausgelesen und in Steuersignale umgewandelt werden können. Beim Drehschalter sind in dieser Demonstration drei

Rastungen vorgesehen. Die Kontakte der drei Rastungen sind über zwei Widerstände (1KOhm) zu verbinden, so dass diese Widerstände eine Kette bilden. Die Enden der Kette sind mit  $V_{cc}$  (+5V) und GND des Arduino zu verbinden, der mittlere PIN des Drehschalters wird an den PIN A0 angeschlossen.

### 3. Programmierung des Arduino

Falls die Arduino IDE noch nicht installiert ist, lest bitte das Kapitel 4.1.4 der BMSAIT Dokumentation.

Ruft nun die .ino aus dem Ordner \Arduino Sketch\BMSAIT\_Switches\ mit einem Doppelklick auf. Das Sketch wird in der Arduino IDE geladen. Wenn ihr die Verkabelung gem. Kapitel 2 durchgeführt habt, sind hier keine Anpassungen erforderlich.

Nachdem ihr das richtige Arduino-Board ausgewählt habt, ladet ihr das Sketch auf den Arduino hoch.

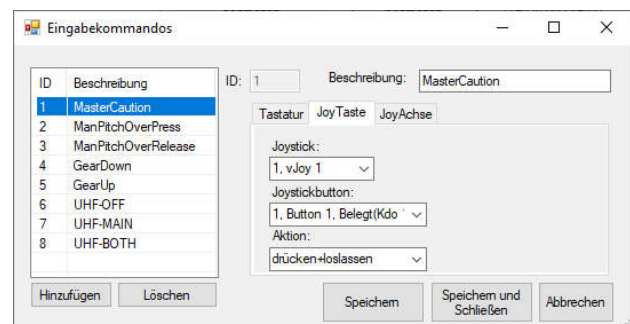
### 4. Installieren und nutzen eines virtuellen Joysticks

Für dieses Beispiel muss vJoy installiert und ein virtueller Joystick aktiviert sein (siehe Kapitel 3.3 der BMSAIT Dokumentation).

### 5. Einstellung des Windows-Programms

Installiert und startet BMSAIT und stellt sicher, dass die Basiseinstellungen richtig vorgenommen wurden. Wichtig ist insbesondere, dass der Verweis auf die Variablendefinition (BMAIT-Variablen.csv) hergestellt wird. Wählt das PUSH-Prinzip, aktiviert die Joystickverarbeitung und schaltet den Autostart aus.

Ladet anschließend die beiliegende Konfiguration (BMSAIT-DemoSwitches.ini). BMSAIT sollte nun die geladene Definition anzeigen (nur ein COM-Port, keine Variablen). Über einen Klick auf die Schaltfläche „Eingabekommandos bearbeiten“ sind die eingerichteten Kommandos sichtbar (siehe rechts).



Prüft, ob den Kommandos die richtigen Signale zugewiesen sind.

1	MasterCaution	Joystick1, Taste 1 (drücken und loslassen)
2	ManPitchOverPress	Taste „o“ (nur drücken)
3	ManPitchOverRelease	Taste „o“ (nur loslassen)
4	GearDown	Taste „g“ (drücken und loslassen)
5	GearUp	Shift und Taste „g“ (drücken und loslassen)
6	UHF-OFF	Joystick1, Taste 2 (drücken und loslassen)
7	UHF-MAIN	Joystick1, Taste 3 (drücken und loslassen)
8	UHF-BOTH	Joystick1, Taste 4 (drücken und loslassen)

Schließt nun die Eingabeverarbeitung und kehrt zum Hauptfenster der BMSAIT Software zurück. Macht einen Rechtsklick auf den COM-Port und bearbeitet diesen. Wählt den COM-Port aus, an dem euer Arduino angeschlossen ist. Wenn ihr nicht sicher seid, welcher COM-Port dies ist, dann wählt entweder die SCAN Funktion oder ihr schaut in dem Windows-Gerätemanager nach.

Ich empfehle die Änderungen nun zu sichern („Speichern unter“ und Auswahl einer neuen Datei).

Aktiviert den Testmodus und startet die Verarbeitung. Wenn ihr einen Button oder Schalter betätigt, sollte in der Konsole des BMSAIT in Windows der Hinweis erscheinen, dass ein Kommando erkannt wurde.

## 6. Einstellungen FalconBMS

Deaktiviert den Testmodus und startet die Verarbeitung des BMSAIT.

Startet Falcon BMS und öffnet das Setup der Eingabegeräte. Betätigt hier nacheinander die Taster/Schalter und prüft, ob die Eingaben erkannt werden. Ihr könnt diese nun einem Callback in BMS zuordnen. Das gilt insbesondere für die Joystickkommandos. Wenn einem Callback bereits ein anderes Tastenkommando zugeordnet ist, könnt ihr die Koppelung des Schalters mit dem Callback in BMS natürlich auch durch Anpassung der Eingabekommandos in BMSAIT erreichen.

## 7. Ergebnis

Startet einen Freien Flug und prüft, ob die Kommandos, die ihr im BMS Setup einem Callback zugeordnet habt, auch in der 3D Welt funktionieren.