

Falcon BMS to Arduino Interface Tool (BMSAIT)

Beispielprogramm X27-Stepper mit Controllerboard



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

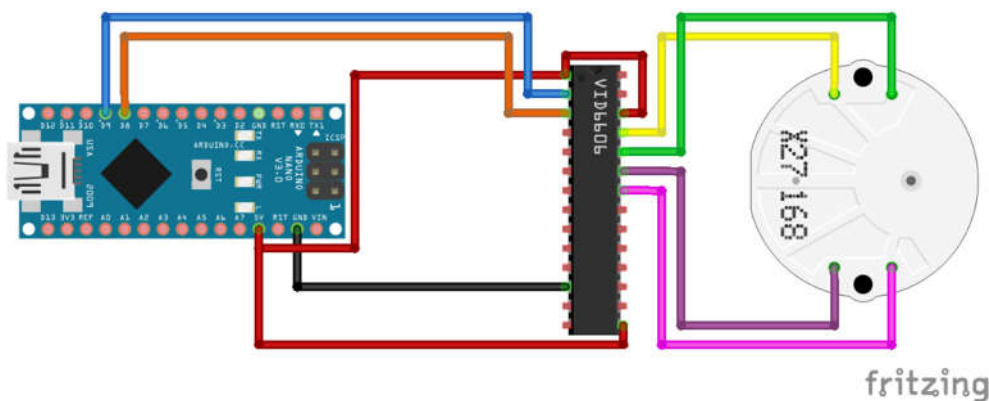
1. Überblick

Das vorliegende Beispielprogramm demonstriert die Abbildung eines Analoginstrumentes über einen X27.168 Stepper-Motor. Der Motor wird über ein Controllerboard VID6606 gesteuert.

Um das Beispiel auszuprobieren benötigt ihr:

- Ein Arduino-Board (z.B. ein NANO)
- Einen X27.168 Stepermotor
- Ein VID6606 Controllerboard
- Ggf. eine externe Spannungsversorgung
- Verbindungskabel

2. Verkabelung



Schließt den Motor wie dargestellt an die PINs des VID6606 Controller Chip an.

Der Controller Chip muss über eine Spannungsversorgung verfügen. Bei einem Motor reicht hier ein Anschluss an die Spannungs PINs des Arduino (siehe Grafik), bei mehreren Motoren sollte der Chip extern versorgt werden. Zudem sind die zwei Datenleitungen zwischen dem Arduino (hier: 8 und 9) und dem VID Chip zu verbinden.

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_StepperVID\ 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. 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 und schaltet den Autostart aus.

Ladet anschließend die beiliegende Konfiguration (BMSAIT_DemoStepperVID.ini). BMSAIT sollte nun die geladene Definition anzeigen (ein COM-Port und eine Variable).

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).

Startet nun eine Kalibrierung über die entsprechende Schaltfläche der BMSAIT Windows App oder dem Menüeintrag bei Rechtsklick auf den COM-Port. Wenn alles geklappt hat, sollte der Motor nun einmal den gesamten Anzeigebereich abfahren und anschließend in die Nullstellung zurückkehren. Wenn nicht, dann versucht die Kalibrierung nochmal anzustoßen.

5. Ergebnis

Druckt euch ggf. eine RPM Anzeige aus, die ihr für den Test auf den Motor legt, um die Anzeige besser prüfen zu können.

Startet nun die Verarbeitung im BMSAIT. Startet Falcon BMS und ruft einen freien Flug auf. Bewegt die Throttle und prüft, ob der Motor die Bewegungen der RPM Anzeige richtig abbildet.