

# Falcon BMS to Arduino Interface Tool (BMSAIT)

Example program for simulating the  
speedbrake indicator on an OLED  
display



Author	Robin "Hummer" Bruns
Document version	1.1
BMSAIT version	1.3.1
BMS Version	4.35
Date	17.02.2021

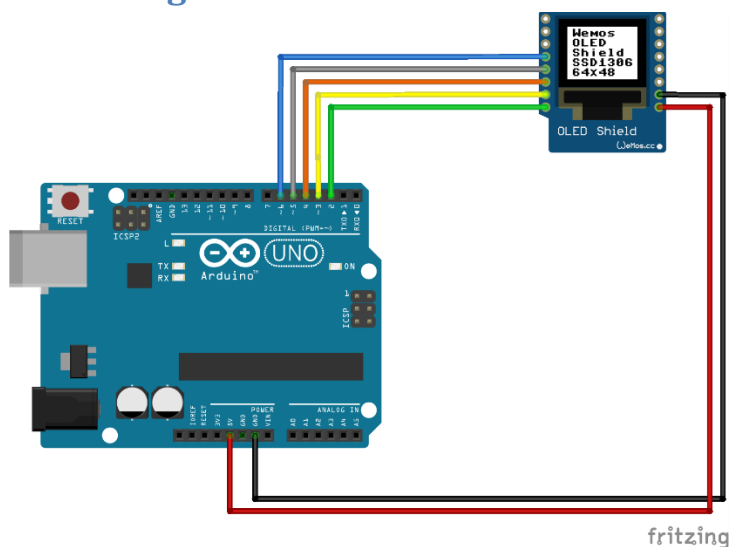
## 1. Overview

This example program demonstrates the image of the speedbrake Indicator on a monochrome OLED display.

To try out the example you need:

- An Arduino board (e.g. a UNO)
- An OLED display (in this example: SSD1306 128x64, size 0.96" with 4-cable SPI connection).
- Connecting cable

## 2. Cabling



PIN Arduino	PIN OLED
GND	GND
5V	Vcc
2	D0
3	D1
4	RES
5	DC
6	CS

Note that the wiring of the OLED display depends on which form of communication the OLED is set to. You can influence this by setting/removing contacts on the back of the OLED.

## 3. Programming the Arduino

If the Arduino IDE is not yet installed, please refer to chapter 4.1.4 of the BMSAIT documentation.

Now call up the .ino from the \Arduino Sketch\BMSAIT\_SBI\ folder with a double-click. The sketch is loaded in the Arduino IDE. If you have done the wiring according to Chapter 2, no adjustments are required here.

With OLEDs, it is important to choose the right "structure" for controlling the OLED. I have described the procedure in the documentation of BMSAIT.

After the constructor has been checked in the sketch and you have selected the correct Arduino board, upload the sketch to the Arduino.

After uploading, the display should briefly show the "off" display (strip). These disappear after 10 seconds.

## 4. Setting up the Windows program

Installs and starts BMSAIT and makes sure that the basic settings have been made correctly. It is particularly important that the reference to the variable definition (BMAIT-Variablen.csv) is made. Select the PUSH principle and turn off the autostart.

Then load the enclosed configuration (BMSAIT\_demoSBI.ini). BMSAIT should now display the loaded definition (one COM port and several variables).

Right-click on the COM port and edit it. Select the COM port to which your Arduino is connected. If you are not sure which COM port this is, then either select the SCAN function and see on which COM port the Arduino sends a response or you can look in the Windows Device Manager.

I recommend saving the changes now ("Save as" and selecting a new file).

Activates the test mode and starts processing. If everything has worked out, the "CLOSED" indicator should now appear.

## 5. Result

Disable the test mode and launch Falcon BMS in an instant action mission. Open the air brake and checks whether the display shows the open status (9 circles). Close the air brake and check if the display changes back to "CLOSED".

The "off" indicator (dashed display) appears briefly when opening/closing and when the power supply to the aircraft is not established (e.g. during ramp takeoff).

Note: As long as there is no connection to BMSAIT, the Arduino will put the OLED to sleep and turn off the display after 10 seconds. A connection must therefore be established for the test.