

# Falcon BMS to Arduino Interface Tool (BMSAIT)

## Example program for solenoid switch



Author	Robin "Hummer" Bruns
Document version	1.0
Software Version	1.4.4
BMS Version	4.37u7
Date	21.04.2025

## 1. Overview

The present example program demonstrates the control of a simulated solenoid switch.

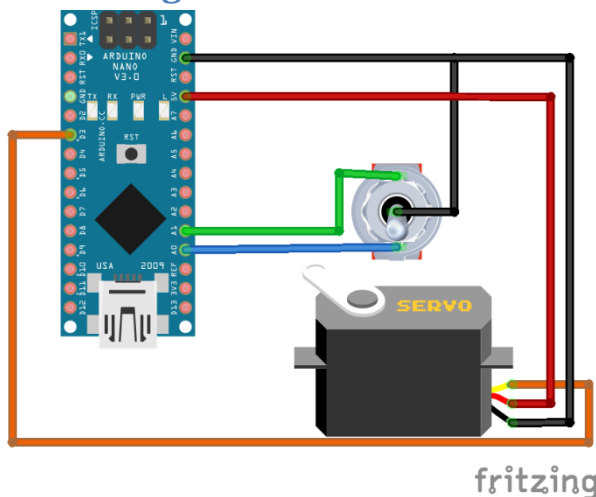
Normal solenoid switches work with an electromagnet that actively holds a switch in a deflected position – if the magnet is deactivated, the switch moves back to the starting position.

In this example, a reverse approach is taken – if a switch is actuated, it is held in that position by a permanent magnet. To bring the switch back to the starting position, the switch is pulled off the magnet by a servo motor.

To try out the example you need:

- An Arduino board (e.g. a UNO)
- A switch with a spring position (OFF-(ON) ; (ON)-OFF-(ON))
- A servo (type SG90g or similar devices)
- Connecting cables

## 2. Cabling



Connect the switch to pin A0 of the Arduino. If the switch has several positions (e.g. autopilot), connect the second position to A1.

The signal cable of the servo motor must be connected to PIN 3 (works with UNO, NANO or similar - it must be a PWM-capable connection of the Arduino!).

## 3. Programming the Arduino

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

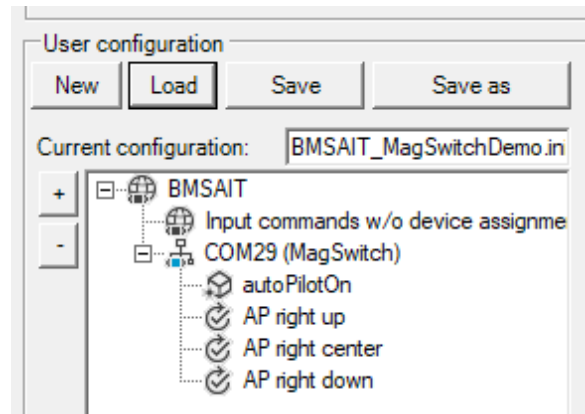
Now call up the .ino from the \Arduino Sketch\BMSAIT\_MagSwitch\ folder with a double-click. The sketch is loaded in the Arduino IDE. If you have done the wiring according to Chapter 2, no adjustments are required here.

Once you've chosen the correct Arduino board, upload the sketch to the Arduino.

## 4. Setting up the Windows program

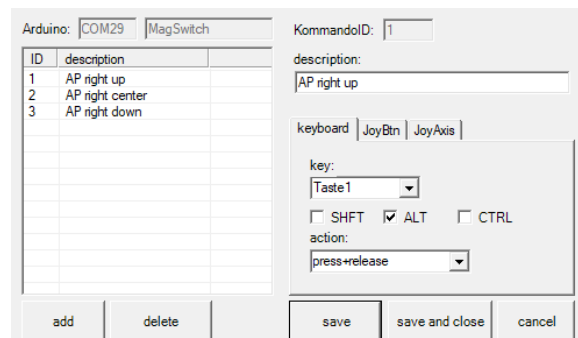
Install and start BMSAIT and make sure that the basic settings have been made correctly. It is particularly important that the reference to the variable definition (BMAIT-Variablen.csv) is made. Select the PUSH mode and turn off the autostart.

Then load the enclosed configuration (BMSAIT\_MagSwitchDemo.ini). BMSAIT should now display the loaded definition (only one COM port, one variable, 3 commands).



By right-clicking one of the commands, you can choose the "Edit input command" option. This is needed to change the input signals for the switch, if necessary (see right).

Check the key assignment of BMS and transfer the inputs here to be able to control the right autopilot switch via BMSAIT.



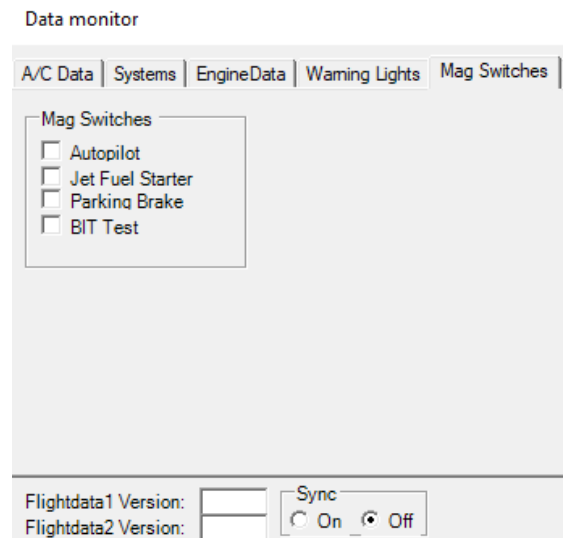
Now close the input command option window and return to the main window of the BMSAIT software. Right-click on the COM port and edit it. Select the COM port to which your Arduino is connected. If you are not sure which COM port this is, then either select the SCAN function or look in the Windows Device Manager.

I recommend saving the changes now ("Save as" and selecting a new file). Activate the test mode and launch the processing mode.

When you press the switch, the message should appear in the console of the BMSAIT in Windows that the command (here: AP right up/center/down) has been recognized.

Call up the data monitor and turn off the synchronization to the sharedMem of BMS. Then switch to the "Mag Switches" tab. The "Autopilot" option should now be accessible. Enables this option. If the solenoid switch is now flipped, the switch should remain in this position. Now remove the "Autopilot" option in the data monitor again. A short time later, the servo motor should be activated to pull the switch back into the original position.

Close the data monitor, disconnect from the Arduino, and disable the test mode.



## 5. Result

Now perform a live test in BMS. Launch the BMSAIT processing without the test mode. Launch BMS and start an Instant Action Mission. When you are in the 3D world, activate the autopilot via the magnetic switch on the Arduino. The switch should remain in the flipped position and the autopilot in BMS should be engaged. Now perform some hard maneuvers in BMS so that the autopilot is disconnected in the simulation. A short time later, the magnetic switch on the Arduino should also be released and jump back to the starting position.