SS4ARD

SIOC SERVER 4 ARDUINO

SS4ARD.EXE

SS4ARD lets you use Arduino boards to interface LEDs, buttons, knobs, encoders, 7-segment displays to build you PMSG B737/B777/B747 home cockpit.

SS4ARD.INI

Here you list all SIOC variables you want SS4ARD to get info about. The COM ports for both Arduino Inputs and Outputs are also listed here.

HOW TO USE SS4ARD

SS4ARD can be used either with OC4BAv4 or as a stand-alone program. A total of 4 Arduino UNO /MEGA boards can be used as Inputs and a total of 4 boards can be used as Outputs. By expanding the Arduino boards SS4ARD can handle a total of 720 pins as Inputs and 720 pins as Outputs. That is more pins that you need to build your PMDG B737/B777/B747 home cockpit.

1. Decide which SIOC variables you want to monitor and add these variable numbers to SS4ARD.ini. The data of these variables will either be sent to your ARDUINO boards to activate LEDS, Servos and 7-segment displays or received from switches, buttons and encoders.

2. Write the ARDUINO programs with the ARDUINO IDE and download these programs to your connected ARDUINO boards.

3. Find the PC's COM ports that are connected to your ARDUINO boards and add these COM ports to SS4ARD.ini

4. Make sure SIOC.exe is up and running with a OC4BAv4 script

5. Start SS4ARD.exe

SS4ARD SUPPORT
Technical questions about SS4ARD should be addressed to
www.flightsim4fun.com/Forum

www.flightsim4fun.com will host some Youtube videos on how to use ARDUINO with OC4BAv4.

ARDUINO SUPPORT

rksoftware will not handle support about how to use ARDUINO or ARDUINO programming issues. Go to https://forum.arduino.cc

SS4ARD.INI

This is the file that tells SS4ARD which SIOC variables that should be sent to the ARDUINO OUTPUT boards. I recommend to have this list as short as possible to make the CPU usage as low as possible. Typical CPU usage is 0.01% with 800 variables defined in SS4ARD.ini, so no worries.

Let us say you are going to expand you home cockpit with an EFIS Captain module built with ARDUINO and are using the OCP4NGX_PP_MIP_v4.0.

As there are no SIOC Variables for EFISs (the EFISs only have Events, like buttons, switches and rotaries) you just add in 0000:0001 before the COM part.

It should then look like this:

```
0000:0001
[ARDUINO BOARD IN]
COMx
COM
COM
COM
[ARDUINO BOARD OUT]
COM
COM
COM
COM
```

In the COM section of the SS4ARD.ini you should have (the x should be replaced with the actual COM port number that your ARDUINO card uses on your PC. You find this number by checking Control Panel in your PC.

If expand your cockpit with a module that has leds, displays, servos etc. you need to find these SIOC variables that are used by the module. Go to OC4BAv4/Script, select B737NGX and make a text file of the OCP4NGX_PP_MIP_v4.0.ssi. Then find all SIOC variables by going to Viewer (right click the OCP4NGX_PP_MIP_v4.0.txt in the Text script window and select inputs. Then find all "module" inputs and use these numbers in the SS4ARD.ini. Now all changes to these variables will be send to the ARDUINO Output boards.

It could look like this in your SS4ARD.ini:
2405:2400:2382:2397:2401:2387:2396:0075:0078:0081:0082:0083:0084:0085:0086:0087:0088:0089:0090:0091:0092:0093:0094:0095:0096:0099:0102:0106:0109

Important, no LF or CR to be used here, only a long line of numbers separated with :

When you expand your ARDUINO home cockpit you just add in the needed SIOC variables used by the function you are building and add in more COM port numbers.

An updated version of OC4BAv4 will simplify the making of the SS4ARD.ini file.

Examples on how to build cockpit modules and how to program your ARDUINO will become available at www.flightsim4fun.com.