

GPIO control in C, using sysfs

The examples below present how to control GPIOs in C language, through sysfs.

Set and read value from D09

Example overview:

- Sets [GPIO\(22\) number](#) for D09
- Sets value 1
- Resets latch for change to take effect on port
- Prints value of the port

```
/* Necessary includes */

#include <stdio.h>
#include <stdlib.h>
#include <string.h> //string manipulations
#include <unistd.h> //sleep

FILE      *file_gpio; //file pointer
int       value;

/*
The main function of program.
Sets GPIO number for D09
Sets value 1
Resets latch for change to take effect on port
Prints value of the port
*/
int main()
{

//Write to 'export' file, so we gain access to port
file_gpio=fopen("/sys/class/gpio/export", "w");
fprintf(file_gpio, "22"); //GPIO22 is D09
fclose(file_gpio);

//Write 'out' to this port 'direction' file, so it's output port
file_gpio=fopen("/sys/class/gpio/gpio22/direction", "w");
fprintf(file_gpio, "out");
fclose(file_gpio);

//Set D01 to 1
file_gpio=fopen("/sys/class/gpio/gpio22/value", "w");
fprintf(file_gpio, "1");
fclose(file_gpio);

//Write to 'export' file, so we gain access to latch
```

```
file_gpio=fopen("/sys/class/gpio/export", "w");
fprintf(file_gpio, "2"); //GPIO2 is one of two latches
fclose(file_gpio);

file_gpio=fopen("/sys/class/gpio/gpio2/direction", "w");
fprintf(file_gpio, "out");
fclose(file_gpio);

// Reset the latch
file_gpio=fopen("/sys/class/gpio/gpio2/value", "w");
fprintf(file_gpio, "1");
fclose(file_gpio);

file_gpio=fopen("/sys/class/gpio/gpio2/value", "w");
fprintf(file_gpio, "0");
fclose(file_gpio);

//Read and print value
file_gpio=fopen("/sys/class/gpio/gpio22/value", "r");
fscanf(file_gpio, "%d", &value);
fclose(file_gpio);
fprintf(stdout, "D09 value is: %d\n", value);

return 0;
}
```