

Hardware GPIO-IRQ

ODROID-N2 and **ODROID-C4** support the GPIO interrupt with edge triggers.

Usage

Download the source code and build with **gcc**.

target

```
gcc -o gpio_irq_test gpio_irq_test.c
```

Basic usage is in the below. You have to put the two arguments for **GPIO number and Edge type**. And you can set **Pull mode** optionally by third parameter.

target

```
root@odroid:~$ ./gpio_sysfs_irq_test  
usage : sudo sysfs_irq_test <gpio> <edge> [pull]
```

Source Code

gpio_irq_test.c

```
#include <stdio.h>  
#include <stdlib.h>  
#include <string.h>  
#include <unistd.h>  
#include <fcntl.h>  
#include <errno.h>  
#include <poll.h>  
  
#define SYSFS_GPIO_DIR "/sys/class/gpio"  
#define MAX_BUF 255  
  
int gpio_export(unsigned int gpio)  
{  
    int fd, len;  
    char buf[MAX_BUF];  
  
    fd = open(SYSFS_GPIO_DIR "/export", O_WRONLY);
```

```
    if (fd < 0) {
        fprintf(stderr, "Can't export GPIO %d pin: %s\n", gpio,
strerror(errno));
        return fd;
    }

    len = snprintf(buf, sizeof(buf), "%d", gpio);
    write(fd, buf, len);
    close(fd);

    return 0;
}

int gpio_unexport(unsigned int gpio)
{
    int fd, len;
    char buf[MAX_BUF];

    fd = open(SYSFS_GPIO_DIR "/unexport", O_WRONLY);

    if (fd < 0) {
        fprintf(stderr, "Can't unexport GPIO %d pin: %s\n", gpio,
strerror(errno));
        return fd;
    }

    len = snprintf(buf, sizeof(buf), "%d", gpio);
    write(fd, buf, len);
    close(fd);

    return 0;
}

int gpio_set_edge(unsigned int gpio, char *edge)
{
    int fd, len;
    char buf[MAX_BUF];

    len = snprintf(buf, sizeof(buf), SYSFS_GPIO_DIR "/gpio%d/edge",
gpio);

    fd = open(buf, O_WRONLY);

    if (fd < 0) {
        fprintf(stderr, "Can't set GPIO %d pin edge: %s\n", gpio,
strerror(errno));
        return fd;
    }

    write(fd, edge, strlen(edge)+1);
}
```

```
    close(fd);

    return 0;
}

int gpio_set_pull(unsigned int gpio, char *pull)
{
    int fd, len;
    char buf[MAX_BUF];

    len = snprintf(buf, sizeof(buf), SYSFS_GPIO_DIR "/gpio%d/pull",
gpio);

    fd = open(buf, O_WRONLY);

    if (fd < 0) {
        fprintf(stderr, "Can't set GPIO %d pin pull: %s\n", gpio,
strerror(errno));
        return fd;
    }

    write(fd, pull, strlen(pull)+1);
    close(fd);

    return 0;
}

int main(int argc, char *argv[])
{
    struct pollfd fdset[2];
    int fd, ret, gpio;
    char buf[MAX_BUF];

    if (argc < 3 || argc > 4) {
        fprintf(stdout, "usage : sudo sysfs_irq_test <gpio> <edge>
[pull]\n");
        fflush(stdout);
        return -1;
    }

    gpio = atoi(argv[1]);
    if (gpio_export(gpio)) {
        fprintf(stdout, "error : export %d\n", gpio);
        fflush(stdout);
        return -1;
    }

    if (gpio_set_edge(gpio, argv[2])) {
        fprintf(stdout, "error : edge %s\n", argv[2]);
        fflush(stdout);
        return -1;
    }
}
```

```
    }

    if (argv[3] && gpio_set_pull(gpio, argv[3])) {
        fprintf(stdout, "error : pull %s\n", argv[3]);
        fflush(stdout);
        return -1;
    }

    snprintf(buf, sizeof(buf), SYSFS_GPIO_DIR "/gpio%d/value", gpio);
    fd = open(buf, O_RDWR);
    if (fd < 0)
        goto out;

    while (1) {
        memset(fdset, 0, sizeof(fdset));
        fdset[0].fd = STDIN_FILENO;
        fdset[0].events = POLLIN;
        fdset[1].fd = fd;
        fdset[1].events = POLLPRI;
        ret = poll(fdset, 2, 3*1000);

        if (ret < 0) {
            perror("poll");
            break;
        }

        fprintf(stderr, ".");

        if (fdset[1].revents & POLLPRI) {
            char c;
            (void)read (fd, &c, 1) ;
            lseek (fd, 0, SEEK_SET) ;
            fprintf(stderr, "\nGPIO %d interrupt occurred!\n", gpio);
        }

        if (fdset[0].revents & POLLIN)
            break;

        fflush(stdout);
    }

    close(fd);
out:
    if (gpio_unexport(gpio)) {
        fprintf(stdout, "error : unexport %d\n", gpio);
        fflush(stdout);
    }

    return 0;
}
```

Example

Here is an using example with the following parameters.

- GPIO Number: 492
- Edge: rising
- Pull: up

target

```
odroid@odroid:~$ ./gpio_irq_test 492 rising up
.  
GPIO 492 interrupt occurred!  
...  
GPIO 492 interrupt occurred!  
.  
GPIO 492 interrupt occurred!  
.  
GPIO 492 interrupt occurred!  
.  
GPIO 492 interrupt occurred!
```

2020/03/19 10:02 · luke.go

From:
<https://wiki.odroid.com/> - **ODROID Wiki**

Permanent link:
https://wiki.odroid.com/odroid-c4/application_note/gpio/irq

Last update: **2020/04/23 16:09**

