



IIO libio



Contents



CLASS: TECHNICAL / PRODUCT / REVISION: TECHNICAL / PROD

A quality version of this page, approved on 11 February 2019, was based off this revision.

Template:ArticleMainWriter Template:ReviewersList Template:ArticleApprovedVersion

Summary

Libiio is a complete library which offers tools and an interface to develop an application using IIO subsystem.

Contents

1 Article purpose	4
2 Introduction	5
3 API description	6
4 Tools	7
5 Source code	8
6 Installation on your target	9
7 References	10



1 Article purpose

The purpose of this article is to:

- briefly introduce the *libio* main features and API
- provide few examples, using *libio* tools



2 Introduction

- *Libiio* is a user space library that provides an **interface** for user space applications. It is basically a wrapper that resides above the following interfaces:

1. `/sys/bus/iio/devices` sysfs interface (for configuration/setting)

2. `/dev/iio/deviceX` device interface (for data)

- *Libiio* also provides **tools** that can be used for testing

- *Libiio* design goals:

1. Interface with the kernel, to access IIO^[1] devices

2. Provide proper data structures and functions to the user application

3. Support for local and remote backends allowing applications to access the devices when running on a local or a remote machine

The full description of the IIO library is provided by the author of the library, see below references:

- What is libiio^[2].

- About libiio^[3].



3 API description

The API description can be found here: <https://analogdevicesinc.github.io/libiio>



4 Tools

Libiio offers tools such as:

- *iiod* server daemon
- *iio_info* to dump attributes

```

root@stm32mp1:~# iio_info
Library version: 0.8 (git tag: v0.8)
IIO context created with local backend.
Backend version: 0.8 (git tag: v0.8)
Backend description string: Linux stm32mp1 4.14.0-00004-gafe4a31 #778 SMP PREEMPT Tue Aug
28 14:02:25 CEST 2018 armv7l
IIO context has 3 devices:
    trigger1: tim6_trgo
        0 channels found:
        3 device-specific attributes found:
            attr 0: sampling_frequency value: 100
            attr 1: master_mode value: reset
            attr 2: master_mode_available value: reset enable update
compare_pulse 0C1REF 0C2REF 0C3REF 0C4REF
    iio:device0: 48003000.adc:adc@0 (buffer capable)
        2 channels found:
            voltage0: (input, index: 0, format: le:U16/16>>0)
                3 channel-specific attributes found:
                    attr 0: raw value: 72
                    attr 1: offset value: 0
                    attr 2: scale value: 0.044250488
            voltage1: (input, index: 1, format: le:U16/16>>0)
                3 channel-specific attributes found:
                    attr 0: raw value: 1746
                    attr 1: offset value: 0
                    attr 2: scale value: 0.044250488
...

```

- *iio_readdev*^[4] (to read or scan from a device)

```

STM32AP [rc=0]# iio_readdev -t trigger1 -s 8 -b 8 iio:device0 voltage0 voltage1 | hexdump
00000000 0068 055a 0058 0520 00b4 03df 0070 055f
00000010 0096 03d6 0089 038f 0077 05c8 0096 03b3

```

See also: [How to use the IIO user space interface](#)



5 Source code

Libiio can be downloaded on a public github^[5]. It can be cloned using git command:

```
git clone https://github.com/analogdevicesinc/libiio.git
```

Tools source code can be found under libiio "tests" directory.



6 Installation on your target

Libiio and the tools it provides are embedded by default in OpenSTLinux distribution.



7 References

- IIO overview, IIO subsystem overview
- <https://wiki.analog.com/resources/tools-software/linux-software/libiio>, What is libiio
- https://wiki.analog.com/resources/tools-software/linux-software/libiio_internals, About libiio
- https://wiki.analog.com/resources/tools-software/linux-software/libiio/iio_readdev, iio_readdev
- <https://github.com/analogdevicesinc/libiio>, libiio download link