Data Acquisition with Comedi on linux

Kuei.Chen   Application Engineer

Advantech Industrial Automation Group

Agenda

- Abstract of linux kernel
- Overview of the Comedi project
- Comedi Function Reference
- Demo

The concept of Linux

- What is linux kernel?
- Compile kernel (building a Image file)
- Linux driver vs. Kernel

Why compile a new kernel?

- The key point: Do What?
- What situations?

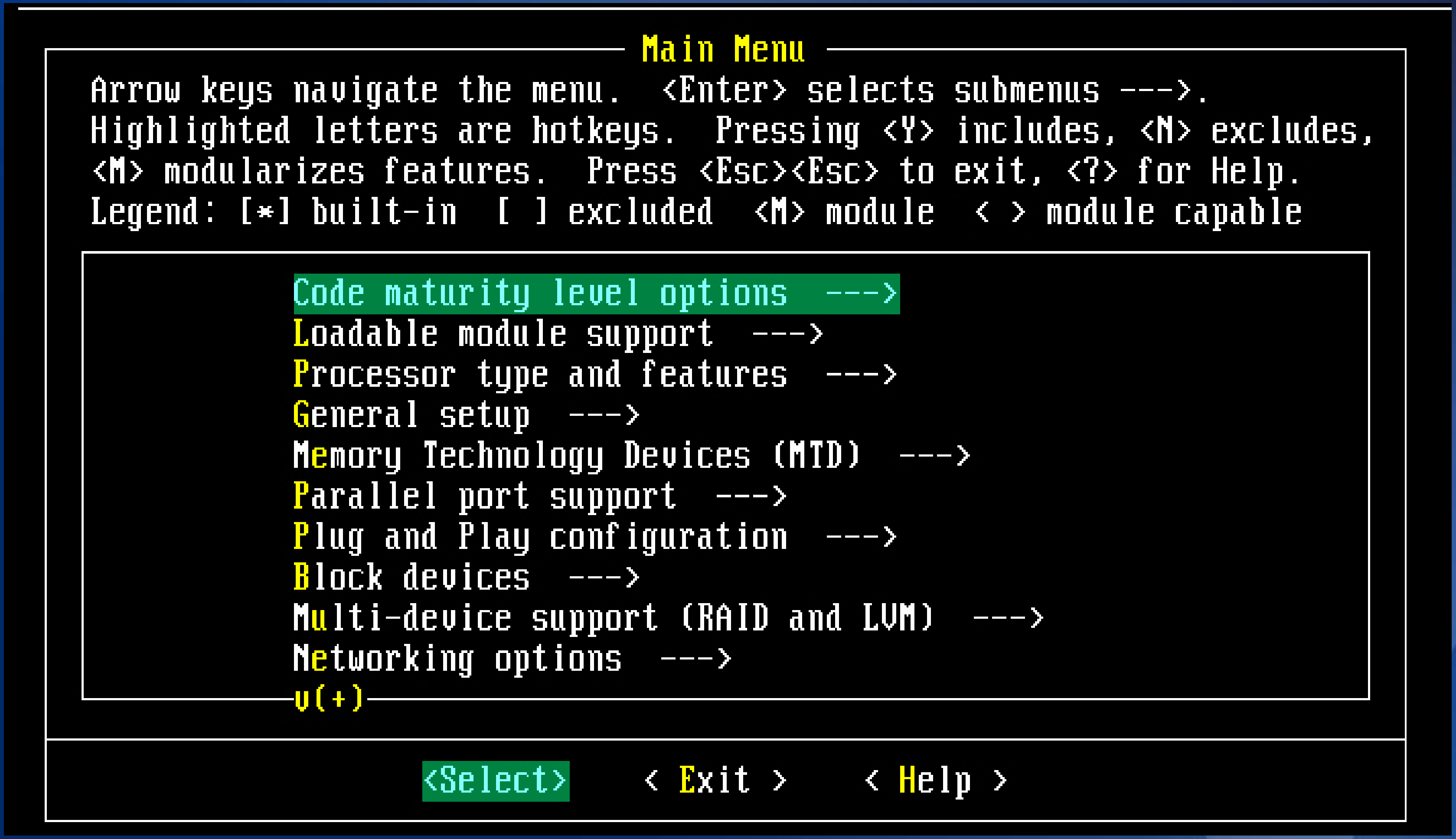
  - New function requirement (PCI , ISA)

  - Kernel too fat

  - Stability

  - Other

Kernel Configuration



```
                              Main Menu
 Arrow keys navigate the menu.  <Enter> selects submenus --->.
 Highlighted letters are hotkeys.  Pressing <Y> includes, <N> excludes,
 <M> modularizes features.  Press <Esc><Esc> to exit, <?> for Help.
 Legend: [*] built-in  [ ] excluded  <M> module  < > module capable

           Code maturity level options  --->
           Loadable module support  --->
           Processor type and features  --->
           General setup  --->
           Memory Technology Devices (MTD)  --->
           Parallel port support  --->
           Plug and Play configuration  --->
           Block devices  --->
           Multi-device support (RAID and LVM)  --->
           Networking options  --->
           v(+)

                  <Select>    < Exit >    < Help >
```

Module

- **What is a “device driver”?**
- **Module**
- **Loadable device driver**

  - reduce kernel process time

  - reduce memory space

The Comedi Project

Control and Measurement Device Interface

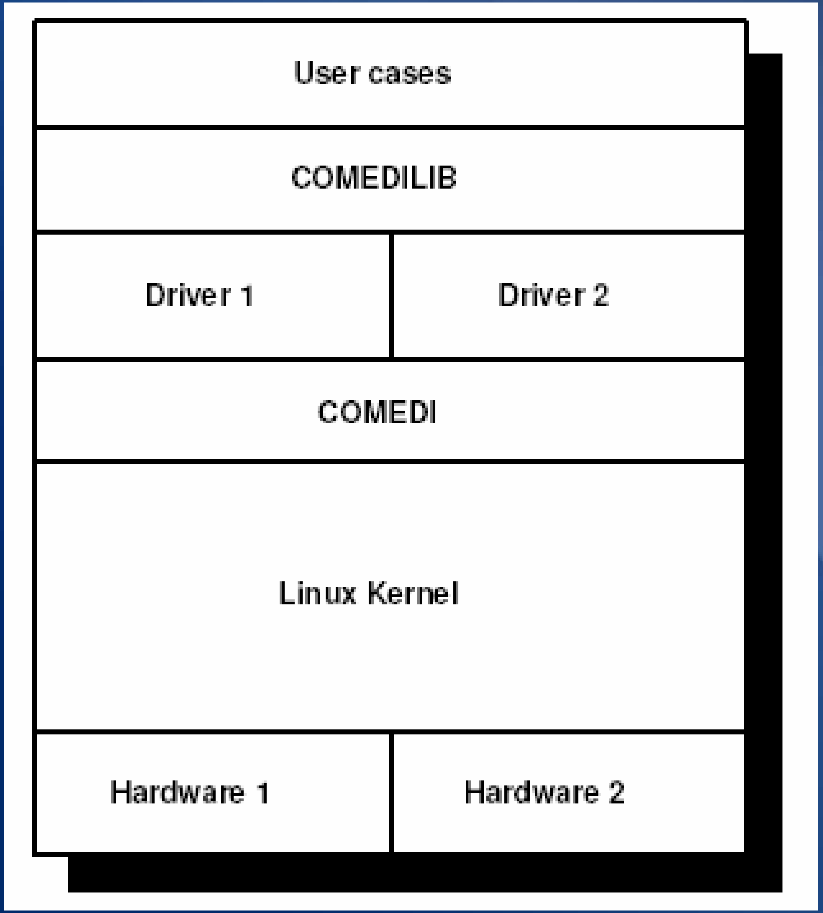
- Comedi is a collection of drivers for data acquisition hardware. These drivers work with Linux

- two packages:comedi and comedilib

- to support multiple vendors and models of cards through a common interface

- The COMEDI project develops open-source drivers, tools, and library for data acquisition

Comedi applications structure



User cases

COMEDILIB

Driver 1 | Driver 2

COMEDI

Linux Kernel

Hardware 1 | Hardware 2

Feature

- Integrated real-time support for most hardware.
- High-level library.
- Driver has been tested with Linux kernel for RedHat8(kernel2.4.18) and RedHat9(kernel2.4.20)
- Driver released together with comedi-0.7.60 and comedilib-0.7.19(for RedHat8) or comedilib-0.7.20(for Redhat9).

Supported Hardware

- **Manufacturer :**

ADLink, Advantech, Amplicon,
Analog Devices, Computer Boards,
Data Translation, ICP,
Keithley Metrabyte, Measurement Computing,
National Instruments

Installation and Configuration

- Make sure that the Linux Kernel source is right

  support 2.2.X – 2.4.X

- Compile the Comedi Module

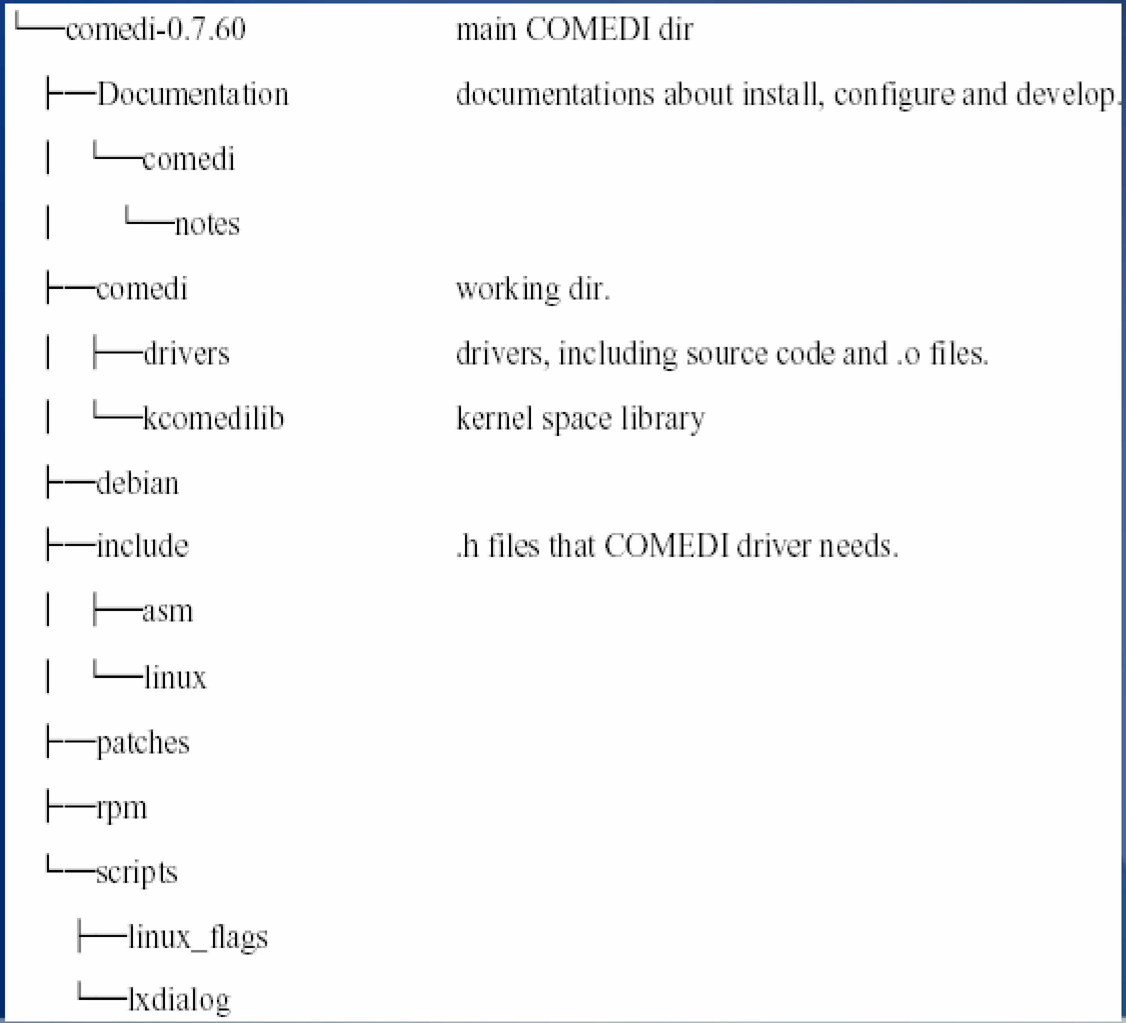
  version comedi-0.7.60

- Installing Comedi Lib

  version Comedilib-0.7.19 -> Comedilib- 0.7.20

- Starting up the Comedi Module

The structure of comedi-0.7.60



```
└──comedi-0.7.60            main COMEDI dir
   ├──Documentation         documentations about install, configure and develop.
   │   └──comedi
   │       └──notes
   ├──comedi                working dir.
   │   ├──drivers           drivers, including source code and .o files.
   │   └──kcomedilib        kernel space library
   ├──debian
   ├──include               .h files that COMEDI driver needs.
   │   ├──asm
   │   └──linux
   ├──patches
   ├──rpm
   └──scripts
       ├──linux_flags
       └──lxdialog
```

Device hierarchy

- The Comedi organizes all hardware according to the following generic hierarchy:

  - channel (the lowest-level component)

  - sub-device (a set of functionally identical channels)

  - device (a set of sub-devices that are physically implement in the same interface card)

The Comedi Configuration

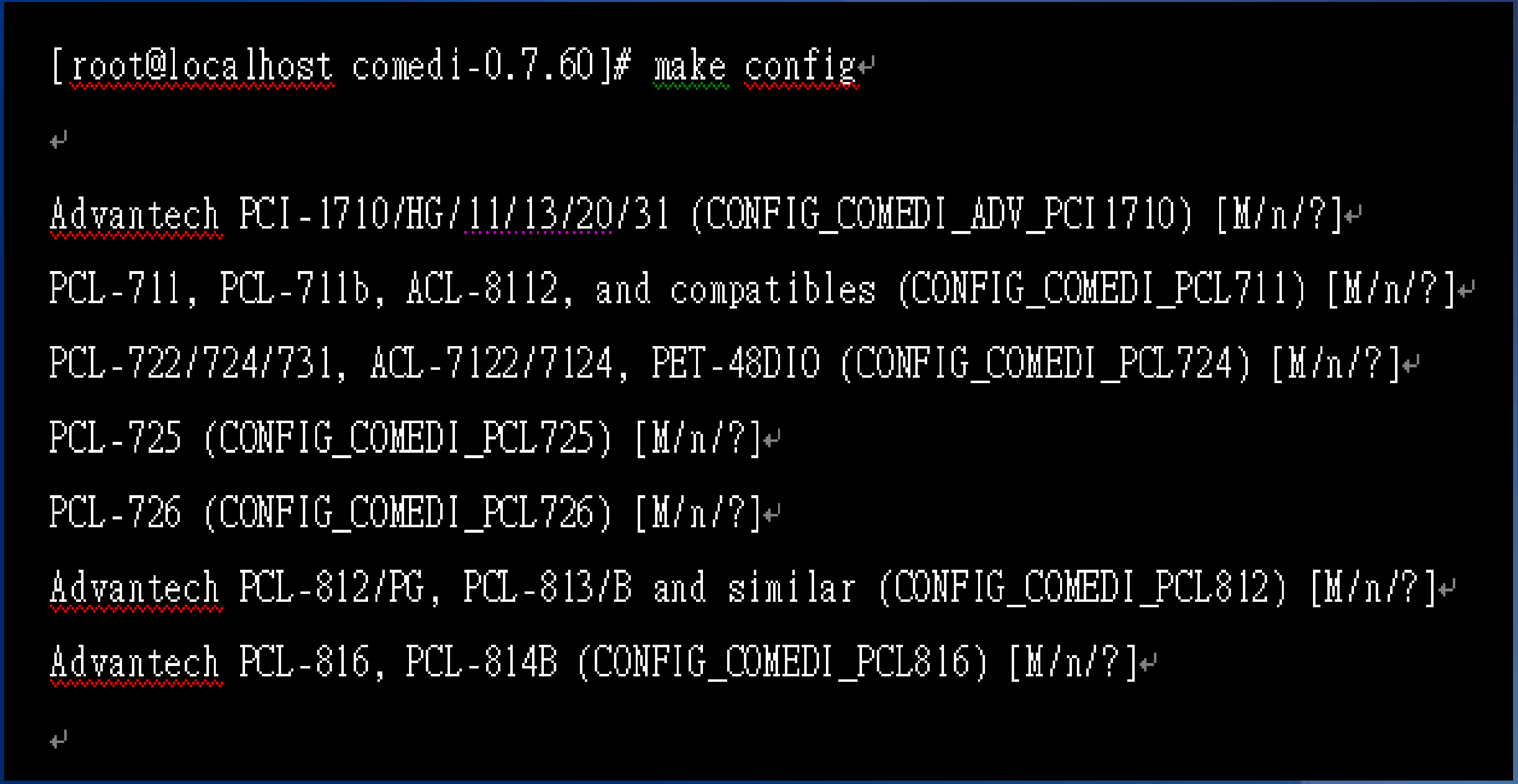
- Make config
- Compile module
- Installation

/lib/module/<<kernel version>>/misc/comedi.o

/lib/module/<<kernel version>>/misc/kcomedilib.o

/lib/module/<<kernel version>>/misc/driver files.o

Make config

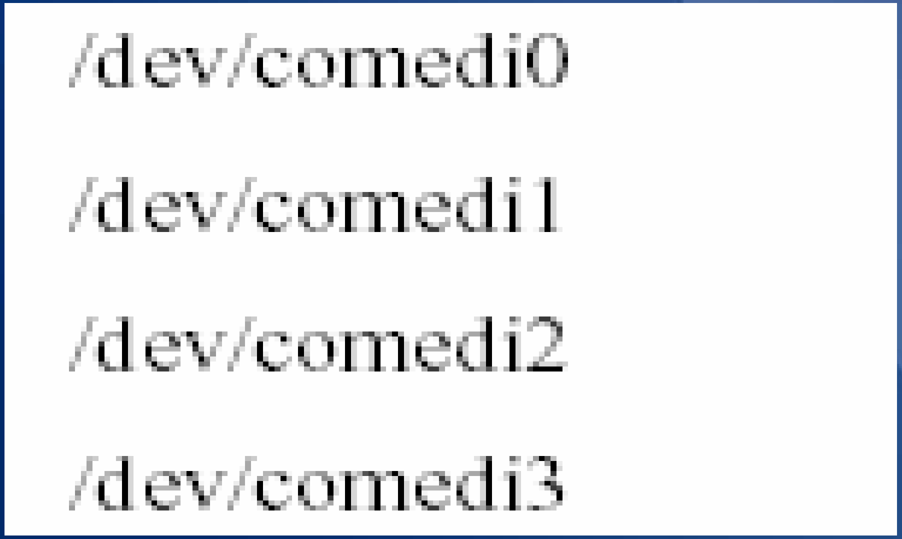


```
[root@localhost comedi-0.7.60]# make config↵

↵

Advantech PCI-1710/HG/11/13/20/31 (CONFIG_COMEDI_ADV_PCI1710) [M/n/?]↵

PCL-711, PCL-711b, ACL-8112, and compatibles (CONFIG_COMEDI_PCL711) [M/n/?]↵

PCL-722/724/731, ACL-7122/7124, PET-48DIO (CONFIG_COMEDI_PCL724) [M/n/?]↵

PCL-725 (CONFIG_COMEDI_PCL725) [M/n/?]↵

PCL-726 (CONFIG_COMEDI_PCL726) [M/n/?]↵

Advantech PCL-812/PG, PCL-813/B and similar (CONFIG_COMEDI_PCL812) [M/n/?]↵

Advantech PCL-816, PCL-814B (CONFIG_COMEDI_PCL816) [M/n/?]↵

↵
```

Device file

- **Create 4 device files(default)**

  #make dev



```
/dev/comedi0
/dev/comedi1
/dev/comedi2
/dev/comedi3
```

Makefile

- If you plan to install more than 4 card, need to edit relative Makefile

```
dev:
        -mknod -m 666 /dev/comedi0 c 98 0
        -mknod -m 666 /dev/comedi1 c 98 1
        -mknod -m 666 /dev/comedi2 c 98 2
        -mknod -m 666 /dev/comedi3 c 98 3
        -mknod -m 666 /dev/comedi4 c 98 4
        -mknod -m 666 /dev/comedi5 c 98 5
```

Command Set

- **Running Comedi**

  #modprobe comedi

- **Driver installation**

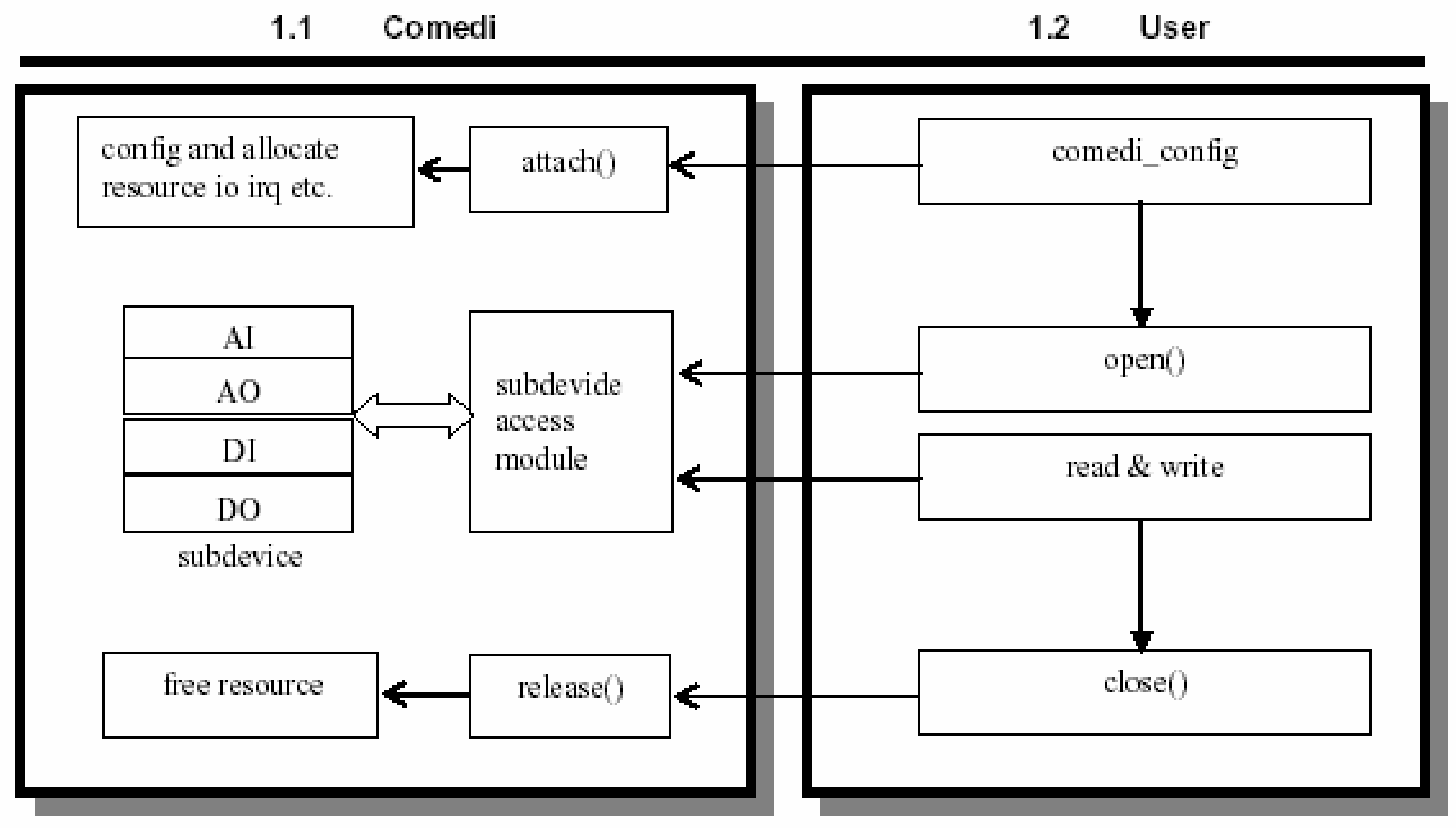
  #modprobe pci17xx (driver name)

- **Config module onto specified device file** (such as : comedi0)

  #comedi_config /dev/comedi0 pci17xx (board name)

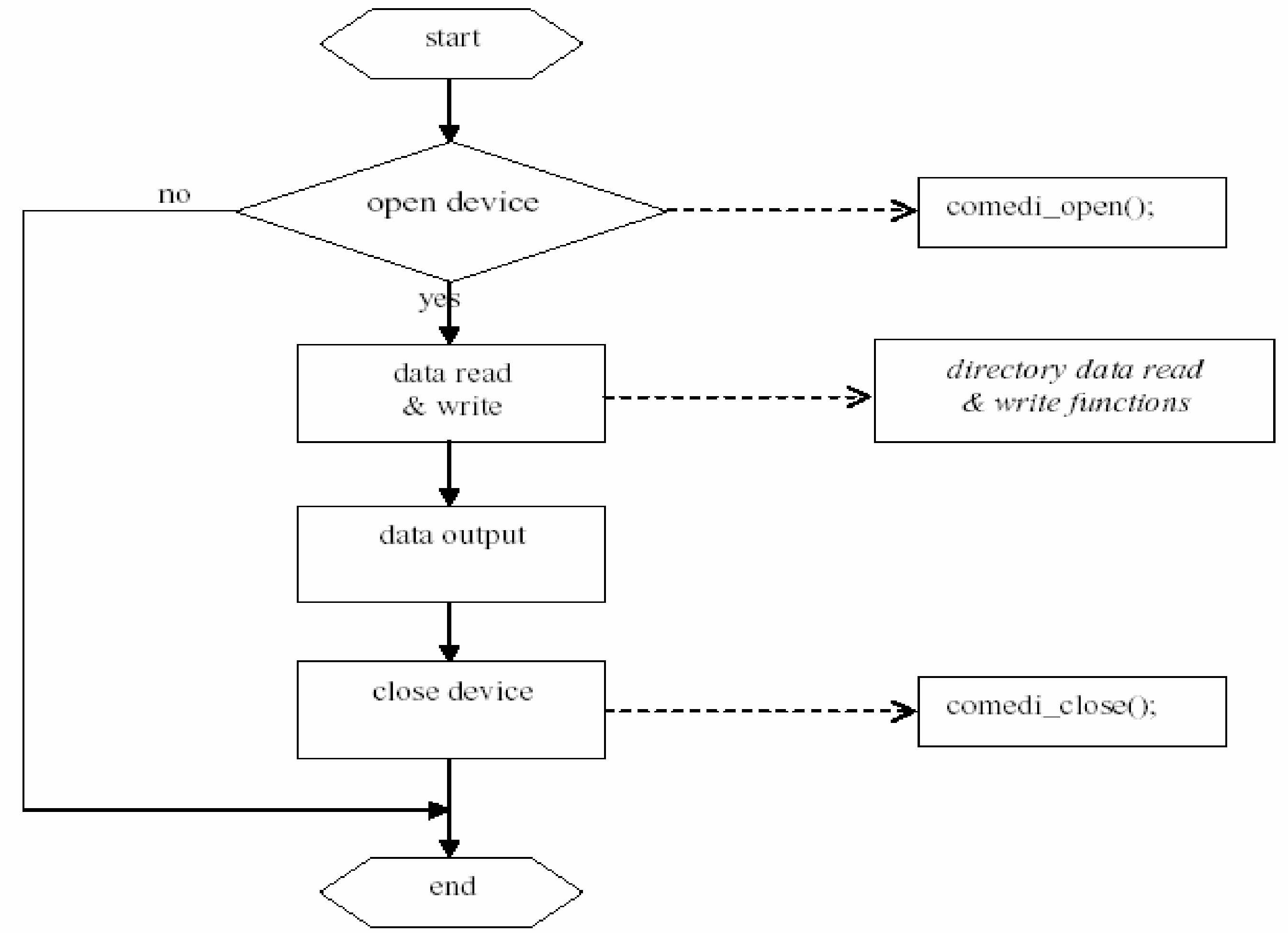
Programming with comedi

- **Relationship between user program and device driver**



Programming with comedi

- **Call flow of user's program**



Comedilib function description

- **Device file operation function :**

  - **comedi_t *comedi_open(char *it)**
    - Opens a comedi device specified by the filename **it**
    - **it** is a device filename, such as “/dev/comedi0”

  - **void comedi_close(comedi_t *it)**
    - Close a device previously opened by comedi_open()
    - **It** is the values return from comedi_open()

Comedilib function description

■ Data acquisition function

- int comedi_data_read(comedi_t *it, unsigned int subd, unsigned int chan, unsigned int range, unsigned int aref, lsampl_t * data);

  - **it** is the value return from comedi_open(),
  - **subd** is the index of the subdevice,
  - **chan** is the index of the channel,
  - **range** is the range index of a subdevice,
  - **aref** is analog reference type,
  - **data** contain the result data

Comedilib function description

Data acquisition function

- **int comedi_data_write(comedi_t *it, unsigned int subd, unsigned int chan, unsigned int range, unsigned int aref, lsampl_t * data);**

  - **it** is the value return from comedi_open(),
  - **subd** is the index of the subdevice,
  - **chan** is the index of the channel,
  - **range** is the range index of a subdevice,
  - **aref** is analog reference type,
  - **data** contain the result data

Comedilib function description

■ Data acquisition function

- **int comedi_dio_read(comedi_t *it, unsigned int subd, unsigned int chan, unsigned int *bit);**

  - **it** is the value return from comedi_open(),
  - **subd** is the index of the subdevice,
  - **chan** is the index of the channel,
  - **range** is the range index of a subdevice,
  - **bit** contains the result, 0 or 1,

Comedilib function description

■ Data acquisition function

- **int comedi_dio_write(comedi_t *it, unsigned int subd, unsigned int chan, unsigned int *bit);**

  - **it** is the value return from comedi_open(),
  - **subd** is the index of the subdevice,
  - **chan** is the index of the channel,
  - **range** is the range index of a subdevice,
  - **bit** contains the result, 0 or 1,

Comedilib function description

■ Subdevice operation function

- int comedi_get_n_subdevices(comedi_t *it)

- This function returns the number of subdevices associated with the comedi descriptor it

- **It** is the values return from comedi_open()

- int comedi_find_subdevice_by_type(comedi_t *it, int type, unsigned int start_subdevice);

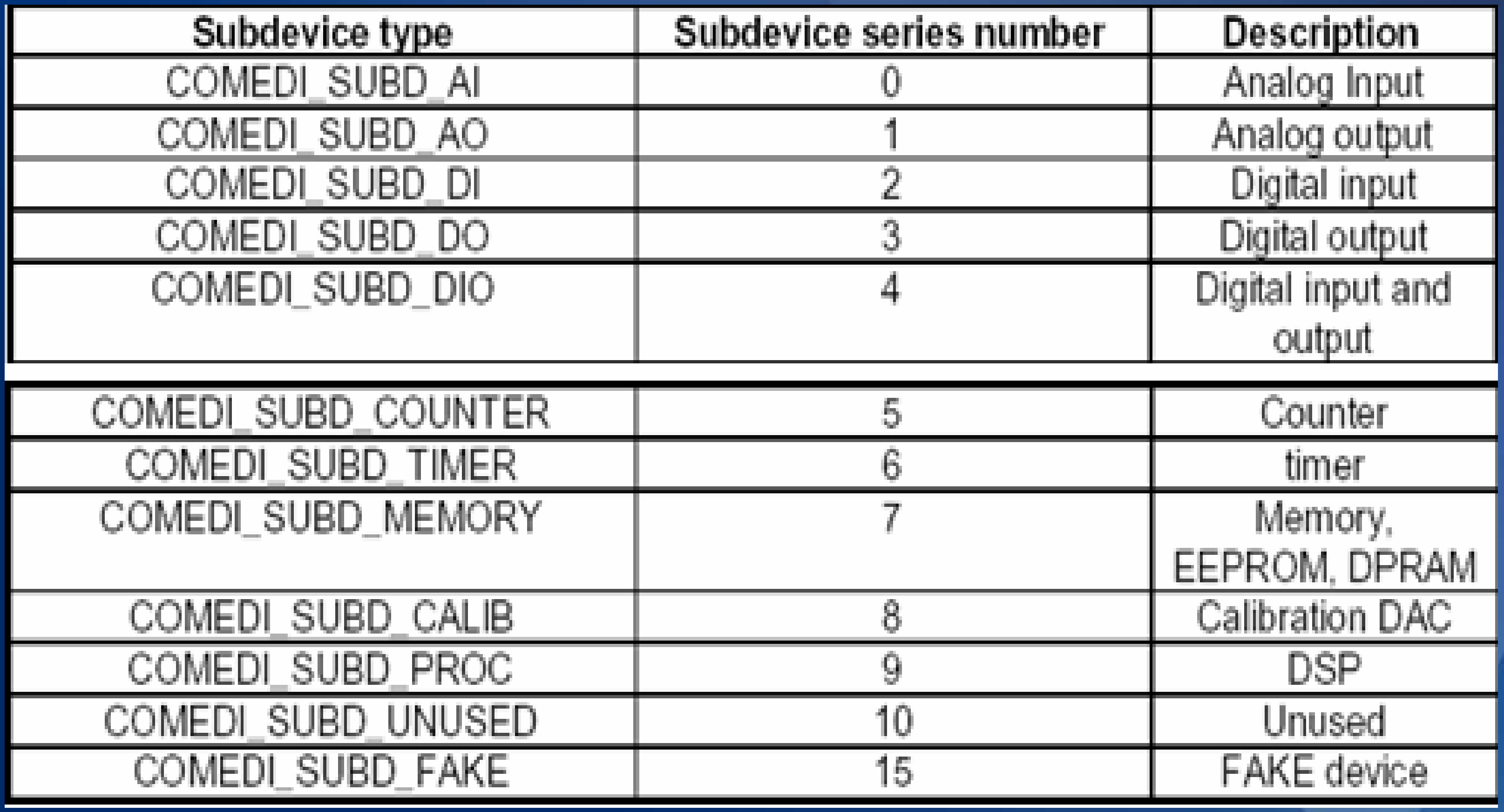
- This function tries to locate a subdevice belonging to comedi device it

- **It** is the values return from comedi_open()

- **start_subdevice** define the starting subdevice number

Subdevice

- Type is one of subdevice type, include:



Subdevice type	Subdevice series number	Description
COMEDI_SUBD_AI	0	Analog Input
COMEDI_SUBD_AO	1	Analog output
COMEDI_SUBD_DI	2	Digital input
COMEDI_SUBD_DO	3	Digital output
COMEDI_SUBD_DIO	4	Digital input and output
COMEDI_SUBD_COUNTER	5	Counter
COMEDI_SUBD_TIMER	6	timer
COMEDI_SUBD_MEMORY	7	Memory, EEPROM, DPRAM
COMEDI_SUBD_CALIB	8	Calibration DAC
COMEDI_SUBD_PROC	9	DSP
COMEDI_SUBD_UNUSED	10	Unused
COMEDI_SUBD_FAKE	15	FAKE device

Comedilib function description

■ Driver information acquirement

char *comedi_get_driver_name(comedi_t *it);

- This function returns the name of a driver.

- **it** is the values return from comedi_open().

- Returns: a pointer to a string containing the name of the driver.

Comedilib function description

■ Driver information acquirement

char *comedi_get_board_name(comedi_t *it);

- This function returns the name of the device.

- **it** is the values return from comedi_open().

- Return: a pointer to a string containing the name of the device.

Comedilib function description

■ Driver information acquirement

int comedi_get_n_channels(comedi_t *it,unsigned int subdevice);

- This function gets the number of channels of a subdevice.

- **it** is the values return from comedi_open().
- **subdevice** is the index of the subdevice.
- Returns: the number of a subdevice channels.

Comedilib function description

■ Driver information acquirement

lsampl_t comedi_get_maxdata(comedi_t *it,unsigned int subdevice,unsigned int chan);

- This function gets the maximum valid data value for channel of a subdevice.

- it is the values return from comedi_open().

- subdevice is the index of the subdevice,

- chan is the index of the channel.

- Returns: the maximum valid data value of the channel or returns 0 on error.

Comedilib function description

■ Driver information acquirement

comedi_range * comedi_get_range(comedi_t *it,unsigned int subdevice,unsigned int chan, unsigned int range);

- This function gets the range for channel of a subdevice.

- **it** is the values return from comedi_open().

- **subdevice** is the index of the subdevice,

- **chan** is the index of the channel,

- **range** is the range index of a subdevice.

- Returns: a pointer to a comedi_range structure that contains information that can be used to convert sample values to or from physical units.

More Information

- Comedi supports a variety of data acquisition hardware; an incomplete list can be found in

  /Documentation/comedi/drivers.txt.

- Comedi also has a web page, at http://stm.lbl.gov/comedi. (New versions of comedi can also be found)

- Comedi may be freely distibuted and modified in accordance with the GNU General Public License.

The End

Thank you!