DAMP — Data acquisition

Bernd Porr

October 13, 2009

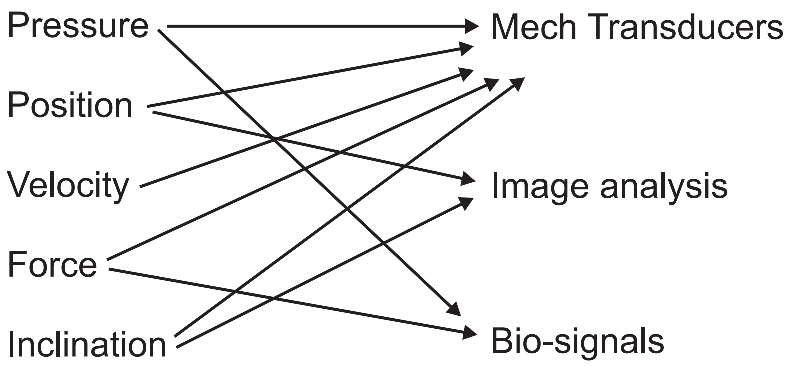

Figure 1: Physical quantities and how to measure them.

1 Introduction

The module is divided into 3 parts. This handout deals with the second part, namely how to measure performance data on your instruments. With performance data we mean anything but sound, be it the angle the bow is hold or how you pluck a string on your guitar. Consequently we are discussing now all different kinds of sensors which could be used for measuring performance data.

Fig. 1 gives an overview over the mapping from physical quantities to sensors. The general idea is to translate a physical quantity into a proportional voltage.

In the following section we are going to present different sensors.

2 How to measure...?

2.1 Pressure/Force

2.1.1 Piezo effect

The piezo effect uses the fact that some materials produce a voltage when pressure is applied. Simple piezo sensors can be found in greeting cards as sounders (yes they also work the other way round) and cost a pound whereas calibrated sensors cost easily hundreds of pounds. The problem is that the voltage output of a piezo is differential meaning that when pressed it produces a voltage but this decays quickly. In order to measure pressure this voltage needs to be integrated. (Farnell 1192520).

2.1.2 Pressure guages

These are Wheatstone bridges which need an instrumentation amplifier which transforms the differential signal into a voltage with ground reference. (Farnell 4982277)

2.1.3 Strain gages

These are usually strips glued on a surface and change their resistance when bent.

2.1.4 Resistive pressure sensors

These sensors are relatively new and not that easy to get. However they are superior to Piezo sensors and are very robust. The resistance is inversely proportional to the pressure (www.interlinkelectronics.com).

2.1.5 Muscle activity

Every muscle in the body can generate quite a high voltage up to 10mV and can easily be measured by ECG surface electrodes. Measurement requires a differential amplifier to cancel out noise. (Farnell 1648643).

2.2 Position

2.2.1 Linear voltage dividers

Use simple voltage dividers to measure position wherever possible. For example a string of a guitar has a certain resistance and can be used as a voltage divider.

2.2.2 Inclination

There are sensors around which measure the angle against the ground. Very often they just give a very rough estimate of the angle such as “vertical” or “horizontal”. These need to be combined with more precise measurements. (Farnell 178338).

2.2.3 Acceleration

Acceleration can be used to calculate position:

$$s(t) = \int \int a(t) dt dt$$

gives us the position. Note that integration in the digital domain is just “adding up” the samples. (Farnell 1566165).

2.2.4 Hall sensors

These sensors react on the magnetic field and change their output proportional to the magnetic field. (Farnell 1521708).

2.2.5 Visual tracking

A video camera can be used to track movements of markers or highly salient features in the scene. The most reliable marker is an LED which gives a distinct saturated dot in a video recording. Image processing can be done with the image library openCV (http://opencv.willowgarage.com).

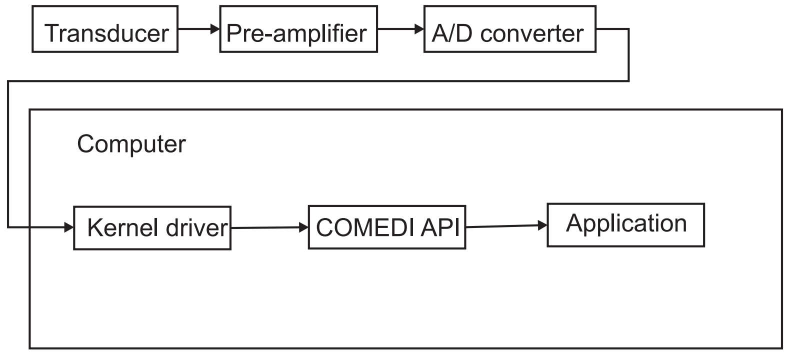

Figure 2: General data flow diagram for analoge/digital processing

3 Digital data acquisition with COMEDI

In this section we are describing briefly how the analogue voltages are digitised and how they can be processed under Linux.

Fig. 2 shows a typical flow diagram of how data is processed here. We are going to concentrate on the computer side which uses the so called COMEDI framework which provides a standard API for data acquisition.

3.1 synchronous acquisition

In this mode the application program sends a request through COMEDI to the DAQ card and then is forced to wait until the result is available:

1. request data
2. wait for data
3. get data
4. jump to 1.

In comedi synchrononous acquisition is done in the following way:

```
comedi_t *device;
lsampl_t data;
device = comedi_open(‘‘/dev/comedi0’’);
comedi_data_read(device,0,0,0,0,&data);
printf(‘‘%d\n’’,data);
```

The problem is that the time between measurements is not defined and therefore the samplerate is also not defined. Therefore we cannot apply any operation to the samples which imply a constant samplerate. For example integration assumes a constant sample rate and even plotting a curve needs equally spaced samples.

3.2 a-synchronous acquisition

In this mode data is acquired while the main program is performing other tasks. Or in other words the acquisition is running in the background.

1. Initialise the background process (number of channels, etc)
2. Start background/data acquisition process
3. Do something else
4. Get data in chunks
5. go back to 3.

This is the mode you should be working with. Comedi has a working example called “cmd.c” in comedilib and also comedirecord can be used as a working example.

Here are the essential commands to start an asynchronous acquisition:

```
static unsigned int chanlist[N_CHANS];
comedi_cmd cmd;
comedi_t *dev;
dev = comedi_open(‘‘/dev/comedi0’’);
chanlist[0]=CR_PACK(0,0,0);
chanlist[1]=CR_PACK(1,0,0);
comedi_get_cmd_generic_timed(dev,0,&cmd,1E9/frequ);
cmd.chanlist=chanlist;
cmd.chanlist_len=2;
comedi_command_test(dev,cmd);
ret=comedi_command_test(dev,cmd);
if (ret!=0) exit(1);
comedi_command(dev,cmd);
```

the last command starts the acquisition in the background. Reading data is done by:

```
while(1){
        ret = read(comedi_fileno(dev),buf,BUFSZ);
        if(ret < 0){
                /* some error occurred */
                perror("read");
                break;
        }else if(ret == 0){
                /* reached stop condition */
                break;
        }else{
                /* ‘‘ret’’ samples in the buffer */
                /* PROCESS DATA */
        }
}
```