ECE352: Signals and Systems II - Winter 2014

Laboratory 1 Assignment

DATA ACQUISITION, IMPORT, AND EXPORT

Lab Dates: two weeks (1st and 2nd)

Lab Report Due Date: At the beginning of the lab on Jan 21st for Tuesday session; Jan 22nd for Wednesday session; Jan 23rd for Thursday session; Jan 24th for Friday session.

1. Introduction

This laboratory will familiarize you with the processes for (a) acquiring digitized data from three sources, (b) importing the acquired data into Matlab, (c) generating graphics of the acquired data using Matlab, and (d) exporting the Matlab graphics to a file for later importing into your lab re-port. The sources of acquired data include (1) digitized voltage waveforms observed on the lab oscilloscope, (2) digitized acoustic waveforms from a microphone, and (3) imagery from a digital video camera. All acquired data and processed byproducts (graphics in this assignment) will be stored as files in an ECE352-specific lab folder created within your per-sonal engineering server directory. This will permit access outside the lab to prepare your lab report.

Note

The video functions will require the R2010b64 version MATLAB. When started, you must log on to the ENGINEERING domain using a username of ENGINEERING\accountname and your engineering password.

2. Objectives

2.1 Acquire oscilloscope waveforms and observe quantization effects.

2.2 Acquire stereo audio signals from a microphone and observe quantization effects.

2.3 Acquire USB camera imagery and display as color image and as separated red-green-blue (RGB) image planes.

3. References

Course text: Section 4.5 (sampling) and page 429 (quantization).

Following reference documents may be found on the ECE352 web site:

1 ♦ Tektronix CFG250 function generator user manual (TekFunctionGen_User_Manual.pdf)

2 ♦ Tektronix TDS210 digital oscilloscope user manual (TekDigScope_User_Manual.pdf)

3 ♦ Dell PC user guide (Dell_user_guide.pdf)

4. Pre-Lab Tasks (0 pts)

Read and familiarize yourself with this entire assignment prior to start of lab.

Bring scope probe from your prior labs.

Bring some headphones to listen to recorded audio (lab PCs may not have built-in speakers).

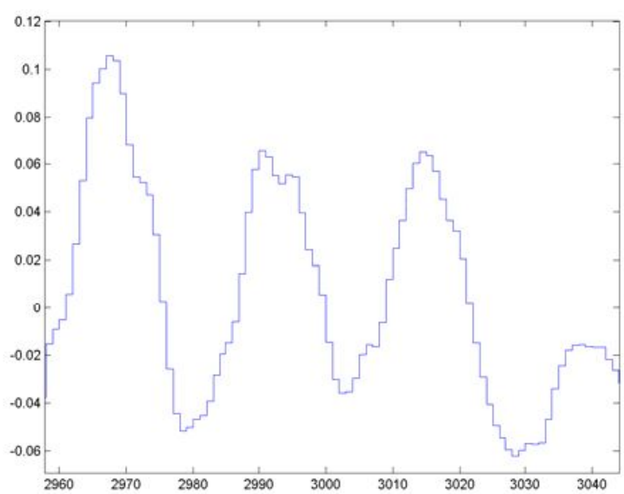
5. Lab Tasks/Experiments

Future laboratory assignments will involve acquiring data, performing operations on the acquired data related to concepts covered in the course textbook and lectures, and making comparisons between measured and theoretical signal responses. This laboratory assignment will acquaint you with data acquisition processes for three signal sources and will make you aware of some acquisition issues related to data fidelity.

5.1 *Oscilloscope Displayed Waveform Acquisition*.

Connect two probes to channels 1 and 2 of the oscilloscope. Select probe attenuation settings to either 1X (no attenuation) or 10X (attenuation by a factor of 10). Make sure your interpretation of the voltage levels seen on the scope screen reflect your choice of either the 1X or 10X setting. Perform probe compensation as needed, using the procedure discussed on page 6 of the oscilloscope manual.

Attach a cable with clips to the function generator, and attach the probes to these clips. Set the function generator frequency, amplitude, and DC-offset controls to create a 10 KHz symmetric triangle wave of -1V to +1V peak-to-peak amplitude range. Select a SEC/DIV setting that will display approximately 5 periods of the symmetric triangle wave. Adjust channel 1 POSITION and VOLTS/DIV controls such that the peak-to-peak waveform display covers 10 graticule vertical divisions, which should cover from the bottom to the top of the visible graticule markings. Adjust channel 2 POSITION and VOLTS/DIV controls such the peak-to-peak waveform display of the same waveform covers a smaller extent of only 1 vertical graticule division.



The waveforms that you see on the scope screen have been *digitized*. This means that not only has the waveform been sampled in time (temporal quantization along the horizontal axis, as shown in the figure above), but also quantized in amplitude level along the vertical axis using an electronic circuit known as an analog-to-digital converter (ADC). The quantization in the temporal direction is typically specified as a sampling rate in either samples per second (sps) or more commonly and equivalently in Hertz (Hz). The quantization in the amplitude direction is typically specified in number of bits used to quantify the amplitude levels. Frequently encountered bit lengths are N=8,10,16, 32 bits. The total number of levels quantized is 2^N, indexed from 0 to 2^N-1 for uni-polar waveforms, or -2^N/2 to +2^N/2 -1 for bipolar waveforms. Thus, an 8-bit quantization produces 256 quantized levels and a 16-bit quantization produces 65536 quantized levels.

In order to capture the digitized waveforms observed on the scope for insertion as a plotted figure in your lab report, a 3 step import/export procedure involving Excel, Matlab, and Word is required. The data will be saved as an Excel-formatted file *.xls (you will provide the * file name when saving). Your lab TA will provide instruction on how the dual-channel acquisition is done. A scope acquisition always captures a fixed horizontal time axis sampling of 10,000 samples to cover the 10 graticule divisions observed along the time axis. The vertical amplitude of each scope waveform is quantized to 8 bits (2^8=256 levels) such that each vertical graticule division seen on the scope display is represented by ~25 levels of quantization (it would require a displayed waveform, therefore, to extend over 10.24 divisions to see all 256 possible levels). After invoking an acquisition, the Excel spreadsheet will have three columns filled, each with 10,000 values. Column 1 contains the 10,000 time-axis sample times (in seconds), column 2 has the 10,000 scope channel 1 quantized voltage values (in volts), and column 3 has the 10,000 scope channel 2 quantized voltage values (also in volts).

To import the saved scope waveform samples of the triangle wave, which are in Excel format, into the Matlab environment, use the on-line Matlab HELP resource to investigate the use of the built-in Matlab function xlsread. Note that you may need to designate which Excel sheet number to import if you saved several scope acquisitions covering multiple pages in one Excel file. The xlsread function will convert the 3 columns of an Excel-formatted file into a Matlab-compatible array of dimension 10,000 x 3 floating-point values.

Once the digitized waveform data is imported into Matlab, all of Matlab's graphics capabilities are available to generate high-quality labeled figures. Use online help to investigate the built-in plotting function stairs. This plotting function creates a stair-step plotted result appropriate for representing quantized waveforms. Using stairs, create two Matlab plots of the same waveform as quantized on channel 1 versus that of channel 2. Which channel reconstructs the triangular waveform with greater fidelity (less observable stair steps)? How many levels can you see in the channel 1 waveform plot versus the number that you can see in the channel 2 plot? What does this suggest should be the practice (in terms of the number of graticules that a waveform occupies) when capturing digitized waveforms by the procedure of this section if one wishes to pre-serve the highest waveform fidelity?

5.2 *Dual (Stereo) Audio Signal Acquisition.*

Obtain a microphone assembly with attached line cable from your lab TA. Plug the line cable into the MIC jack on the front of the lab PC (consult the on-line Dell PC user guide for location). Place the microphone assembly on the lab bench in a convenient close position to record your voice. You will use the open-source application Audacity to acquire a few seconds of dual-channel audio from this assembly.

As in the case of the case of the scope waveform acquisition, you control the fidelity of the audio waveform acquisition by adjusting three parameters: the signal sampling rate (temporal quantization) in Hz, the number of bits (amplitude quantization), and the volume level (the gain control, analogous to selecting the vertical waveform height adjustment on the scope). The microphones have a roughly 10 KHz response range, so a sampling rate of at least 20 KHz would be needed to fully capture all the frequency range of the microphones. However, most speech nominally falls under 5 KHz in frequency content. For this task, you will evaluate the quality/fidelity of stereo audio recordings made for the following parameter selections using PCM (pulse code modulation) audio format:

1 ♦ fs=22050 Hz, nbits=16, volume full

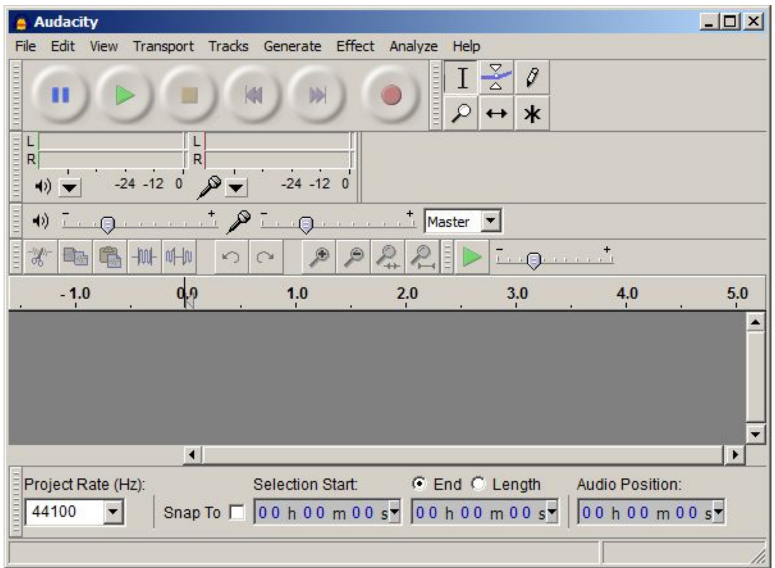
2 ♦ fs=22050 Hz, nbits=16, volume half

3 ♦ fs=22050 Hz, nbits=8, volume full

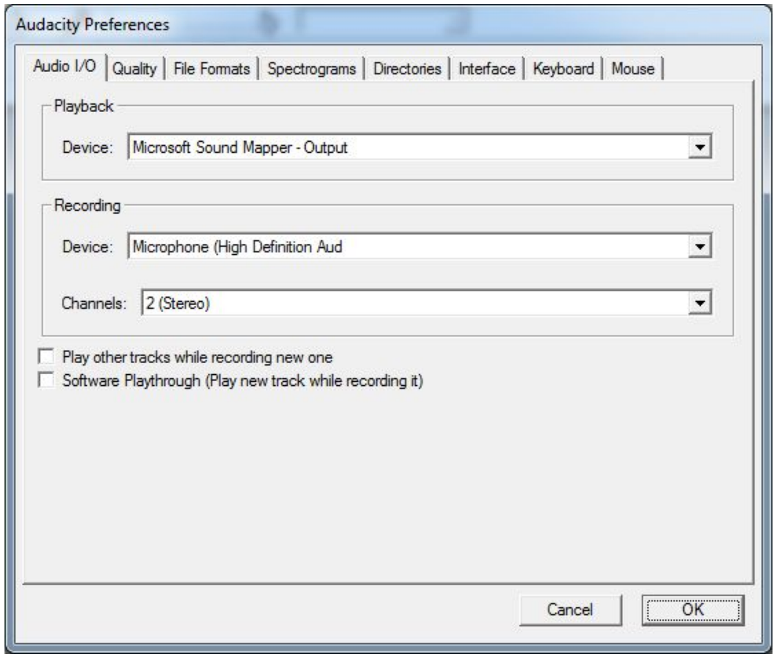
4 ♦ fs=11025 Hz, nbits=16, volume full

5 ♦ fs=11025 Hz, nbits=8, volume full

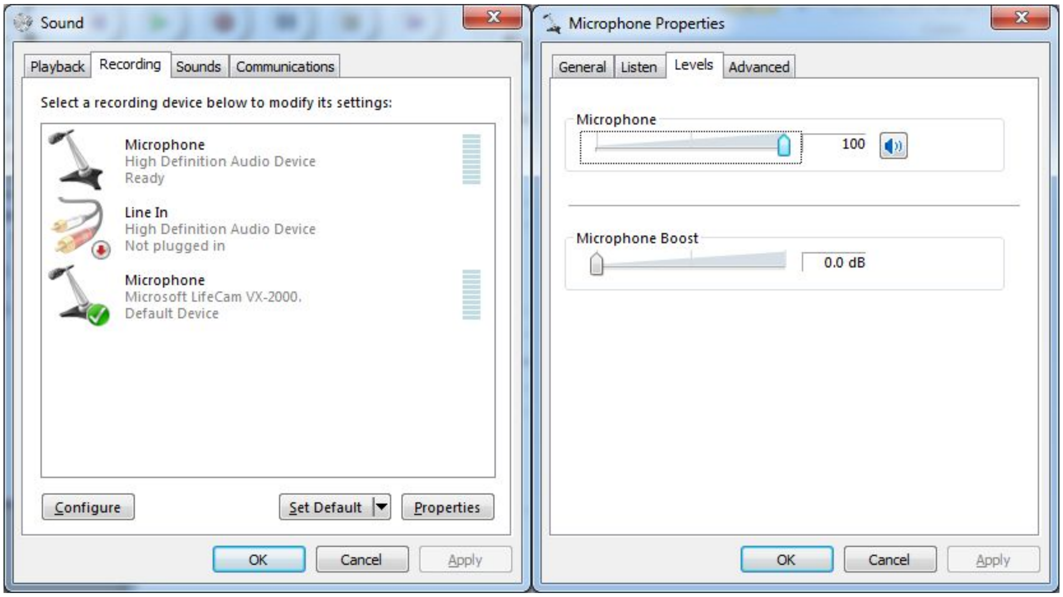
To make an audio acquisition, first start Audacity from the Windows environment. A window as shown below will appear:



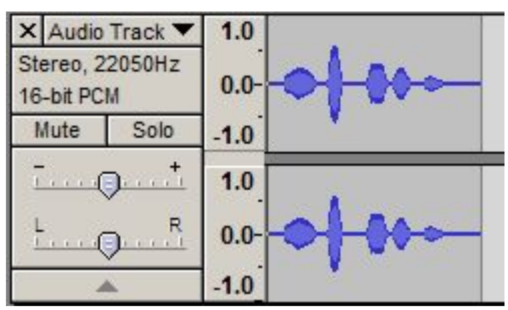
The fs rate is selected in the Project Rate box in the lower left corner. The microphone volume is near the top of the window, for setting the recording volume. To prepare for audio recording, select Edit -> Preferences. A window as shown below will appear:



Under Audio I/O, set Channels to Stereo, and Device to Microphone (High Definition Audio Device) to select the external microphone, and not the one built in to the camera. Under Quality, set Default Sample Format to 16-bit, and then select OK. Under File Formats, you will choose whether you are saving nbits=8 or nbits=16. For 8-bit, set Uncompressed Export Format to "WAV (Microsoft 8 bit PCM". For 16-bit, set it to "WAV (Microsoft 16 bit PCM)". Before starting the recording process, make sure the microphone volume is set to maximum in the Windows volume control. Right-click on the volume in the taskbar and select Recording Device, which will bring up a window like the one shown below on the left. Right-click on the High Definition Audio Device microphone and select properties. Go to levels and slide the volume all the way up.



You are now ready to create and save an audio recording. While talking in the direction of the microphone, click the record button to start recording. Record a few seconds of your voice using a common phrase (for example, "This is a test of audio recording.") at the same level and distance from the microphone for each set of selections, then click the STOP button to end recording. Make sure you are recording in stereo, as evidenced by two audio tracks being shown:



Plug some headphones into the headphone jack on the front of the PC. Click the triangular PLAY button in the Sound Recorder window to playback the recorded stereo sound to validate the quality of the recording. Finally, save your recording by selecting File -> Export As WAV. Select a filename and location and click Save.

To import the saved recording in wave format into the Matlab environment, use the on-line Matlab HELP resource to investigate the use of the built-in Matlab function wavread. Use script line:

```
[x,fs,nbits]=wavread('filename');
```

to import your recording into a Matlab array x of dimension #samples x 2. The first column contains channel 1 audio samples and the second column has the samples from channel 2 audio. If you get only one column, then the recording was incorrectly recorded in Mono rather than Stereo. Look at the values of fs and nbits to verify that they correspond to the attributes that you selected when creating the file with Sound Recorder. To have Matlab reproduce the recorded two-channel sound so that you can listen with either headphones or powered speakers, use the script

```
sound(x,fs);
```

to hear the stereo recording. If you want to listen only to channel 1, then use

```
sound(x(:,1),fs);
```

or change 1 to 2 to listen only to channel 2.

5.3 *Camera Image Acquisition.*

Obtain a usb camera with attached USB cable from your lab TA. Plug the camera into a USB port on the front of your lab PC. Windows drivers have already been installed on the lab PC so that it should automatically recognize the camera. In MATLAB, enter the following two commands to set up a video object with which to capture individual frames in RGB color:

```
vid = videoinput('winvideo', 1, 'YUY2_160x120');
vid.ReturnedColorspace = 'rgb';
```

Once the object is ready, you can preview the output of the camera using:

```
preview(vid);
```

To acquire an image from the Matlab command window, type in the command

```
pic = getsnapshot(vid);
```

This will automatically capture and load a 160 x 120 x 3 array of uint8 pic (you may select another variable name of your choice) with the captured image. The image consists of 120 rows x 160 columns of pixels, each pixel of which is expressed in the three RGB (red-blue-green) colors of 8-bit un-signed integers (uint8) per color (0 is dark and 255 is brightest intensity for a color)[total of 24 color bits per pixel]. To see the image within Matlab, use on-line Matlab HELP for utilization of function image, and simply type in the command image(pic). An image comparable to the image of a Rubick's cube below will then be generated within a Matlab figure window.

You will need to save the images in your working directory. You can use the MATLAB command imwrite to save images in bitmap format.

```
imwrite(pic, 'filename.bmp', 'BMP');
```

You can also load previously saved bitmaps using imread.

```
pic = imread('filename.bmp', 'BMP');
```

After experimenting with the camera and the effect of lighting conditions on the quality of the camera imagery, your task here is for each lab partner to acquire and save the image of the other lab partner's face.

6. Post-Lab Tasks

6.1 *Oscilloscope Waveforms.*

Generate labeled and captioned Matlab-created figures for each of the two waveforms saved from Task 5.1. Use the same vertical and horizontal scaling for both figures. Avoid numeric labeling of the time or amplitude axes that involves exponential powers of ten by selecting sec, msec, or μsec for the time axis and V or mV for the horizontal axis. Recall, from ECE351, the process of exporting Matlab figures into other applications like MS Word. While in the Matlab window, select File -> Export and save the Matlab figure window as a *.tif file (recommended; al-ternative formats are *.jpg, *.bmp, and *.emf).

Question: Examining the two plots of the waveform taken at different vertical scope gain set-tings, how does the selection of the Volts/div setting affect the fidelity of the saved waveform?

Question: What are the implications for saved scope waveform data that you may make in future labs if you want to achieve saved scope waveforms of highest fidelity?

6.2 *Dual Audio Signal.*

Generate labeled and captioned Matlab-produced figures for a 0.1 second audio segment of the waveforms you recorded in Task 5.2. Use the Matlab subplot function to create figures in which the top half is a plot of channel 1 audio and the bottom half is a plot of channel 2 audio for the 0.1 second audio segment; use a common vertical scaling and a common time axis interval. The Matlab function wavread scales the signed 8-bit or 16-bit quantized integer sample values pro-duced by the Sound Recorder application to floating-point values in a range between -1 and +1 (max) by dividing the integer values by 2^{N-1}. To restore the original integer sample values, scale the floating values by 2^{N-1}, for either N=8 or N=16.

Question: What percent of the maximum integer discretization range did your recorded wave-forms use in each of the 5 recording conditions?

Question: Are you able to hear (and see in the plots) perceptible quality differences among the 5 recordings due to the differences in the degree of quantization? The sampling rate? The volume control setting? What are the implication for future audio acquisitions in order to maintain highest fidelity recordings?

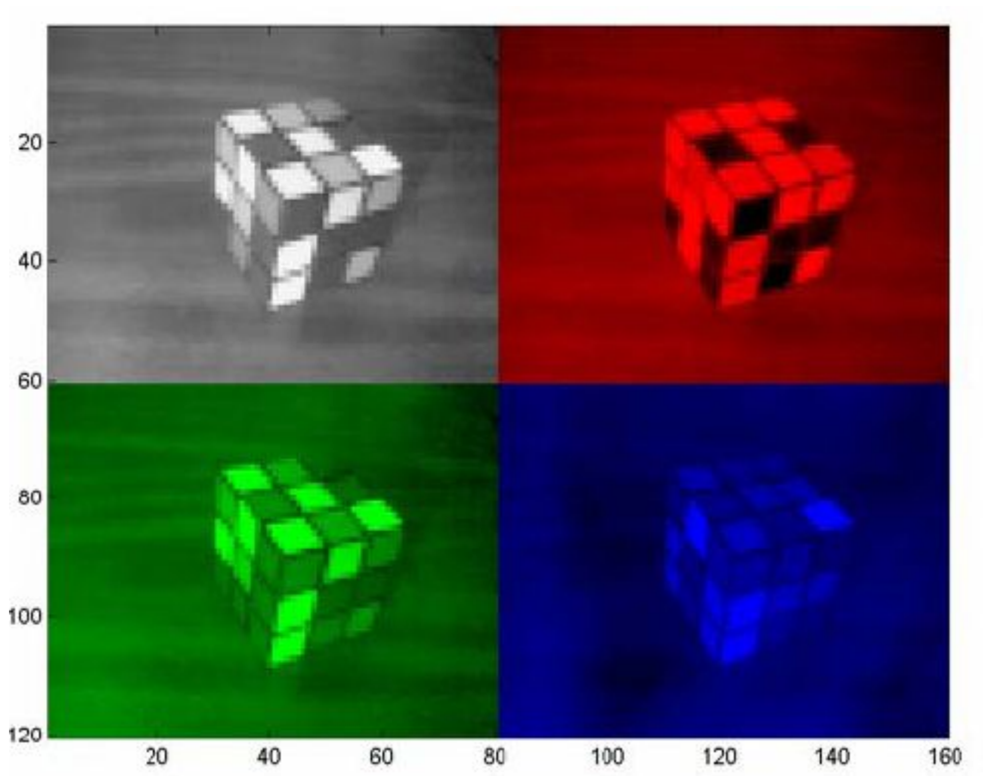
Question: Based on your listening tests, what are the implications for obtaining the highest fidelity audio recordings in terms of the three parameters that can be selected (fs, nbits, volume level)?

6.3 *Camera Imagery.*

The analysis of imagery often is simplified if each color is processed separately, or if the colors are combined into a single image in which only the illumination level changes (gray scale). In this task, you will devise Matlab operations that will separate the imagery you acquired into its individual color planes and also combined to create a gray scale image. Use the Matlab HELP facilities to examine the use of the command rgb2gray (located in the Matlab Image Processing Toolbox) for converting an RGB-based image to a gray-scale-based image. Your task is to create a new Matlab function (name it color_separation.m) that takes your original saved image and (1) creates individual gray scale and single color images of each of the separated red, green, and blue image planes, and (2) constructs a large image array

```
combined_pic= color_separation(your_pic);
```

composed of these four sub images. This can then can be displayed [image(combined_pic)] as illustrated below for the cube image.



7. <u>Lab Report</u>

Use the lab report format to be provided by the TAs.