CG1111 Engineering Principles and Practice I

Basics of Signal Processing

(Week 10, Studio 2)

Time	Duration (mins)	Activity
0:00	15	Briefing
0:15	70	Activity #1: Sampling theorem and Nyquist rate (graded lab report)
1:25	30	The A-maze-ing Race – Project discussion
1:55	40	Work on the project with your team
2:35	15	Final discussions and wrap-up

Introduction

- A signal is a function of one or more variables that conveys information about some (usually physical) phenomenon
- Signals can be classified mainly into continuous signals and discrete signals
  - Continuous signals are signals in which the independent variable (usually time) is a continuous variable
  - Discrete signals are signals in which the independent variable takes on discrete values, for example, integers
- A discrete signal may represent successive samples of a phenomenon for which the independent variable is continuous
- The process of converting a signal from continuous time to discrete time by measuring the signal at certain intervals of time is known as sampling.
  - **Each measurement is referred to as a sample**
- The sampling rate or frequency is the number of times a signal is read in a second (eg. 100 times in a second, i.e. 100 Hz, which means the signal is sampled once every 1/100 seconds)
  - **Sampling period or interval = 1/sampling rate**
- The Sampling Theorem states that the information contained in a signal would be preserved if it is sampled at a frequency F, where F is equal or greater than twice the maximum frequency in the signal.
- The minimum sampling rate that is required to well represent a continuous time signal with highest frequency component f is given by $2\times f$ and this frequency is known as the Nyquist rate
- For a given sampling rate f_s, perfect reconstruction is guaranteed possible for a continuous signal whose highest frequency is $f_s/2$, which is known as the Nyquist frequency
- When the sampling rate is lower than the Nyquist rate, the signal reconstructed from samples (using DAC) is different from the original continuous signal. This effect is known as aliasing.

Materials Required

- Bitscope
- Arduino Uno and USB cable
- Computer/ Laptop

Objectives

- To understand the classification of signals – continuous time vs discrete time
- To understand sampling and sampling theorem
- To understand and realize the effect of aliasing
- To understand and realize ADC (Analog to Digital Conversion) in Arduino
- To verify the sampling theorem experimentally

Setup: ***Waveform Generator:***

a) Run BitScope DSO on your computer.
b) Turn on the power to your BitScope and turn on its waveform generator by clicking on "WAVE" in the "Scope Selectors (2)" panel (see Appendix for its position).
c) In the "FUNCTION GENERATOR" panel (top left), the top left brown box allows you to choose the type of waveform – TONE (sinusoidal waveform), RAMP (triangle waveform) or STEP (square waveform).
d) The red box beneath the brown box lets you choose and vary the frequency of the waveform. Left click it and drag the value downward to set the frequency to 10 Hz
e) For the peak-to-peak voltage, left-click the light green box below the red box and drag its value upward or downward to vary the peak-to-peak voltage to 3.3V.
f) For the DC offset (i.e., the box beside the peak-to-peak voltage setting), left-click it and drag its value downward to 0 V. The waveform will still have some DC offset.
g) In the "Channel Controls (7)" panel, change the voltage per division of CHA to "1V/Div".
h) In the "Timebase Control (6)" panel, change the Timebase to be "50 ms/Div".
i) Connect your "AWG" pin to "CHA" to observe the waveform.

Activity #1: Sampling theorem and Nyquist rate

In this activity, we would be verifying the Sampling theorem by generating a continuous signal using the function generator and sampling this signal using the Arduino Uno at different frequencies and observing whether the sampled discrete-time signal is a good reconstruction of the original continuous time signal.

Procedure

1. Setup the Function Generator of the DSO as described in **Setups: Waveform Generator**.
2. You may now use the Arduino IDE to write a program to sample the sine waveform generated by the function generator at the frequencies specified in Table I.
3. To which pins in the Arduino would you connect the function generator output terminals? Note this down in your journal. Connect the function generator 'AWG' and 'GND' pins to your chosen pins.
4. Use the **analogRead()** function to read the values from your chosen input pin. Provide the entire code for Activity #1 in your report.
5. Sampling corresponds to using the **analogRead()** function in this context. Based on the sampling rates specified in Table I, compute the corresponding sampling period for each of the sampling rates. You can set the sampling period as a delay in the loop() function.

6. Initialize the Serial monitor at a baud rate of 9600 and display the input values in the Serial monitor as and when they are read from the input pin.

   Note: Ensure that you have initialized the serial port in your Arduino code

7. Obtain 50 samples for each sampling rate for **the same 10 Hz sine waveform** and store the values in an Excel Spreadsheet (copy from Serial Monitor). **Do not change the frequency of the Sine waveform.**
8. Can you fill in the sampling time column for your samples assuming that the time taken by the **analogRead()** and **Serial.print()** functions are negligible?

   Hint: use your sampling period. Your excel spreadsheet should have 2 data columns for each sampling rate – Time and sampled values (from ADC).

9. The ADC in your Arduino converts 0 to +5V present in the Analog pins to digital values in the range of 0 to 1023 respectively. How can you scale your sampled values (from ADC) to obtain their corresponding voltage values using the ADC information provided? Provide the scaled values as the third column (in Excel) for each sampling rate's data.
10. Plot scaled sample values vs time for each sampling rate and note down whether aliasing is observed for each of them in Table 1. You may want to superimpose your sampled values on a sine wave plot.
11. What is an appropriate sampling frequency for the provided input waveform?
12. Calculate the Nyquist frequency for a sampling rate of 50 Hz.

Table 1

S. No	Sampling rate (Hz)	Sampling period	Aliasing observed?
1	5		
2	10		
3	20		
4	25		
5	50		

Please feel free to ask your TA or the Instructors any questions or doubts you may have.

Activity #2: The A-maze-ing Race – Project discussion (30 mins)

Activity #3: Work on the project with your team (40 mins)

END OF STUDIO SESSION