Laboratory 12

Data Acquisition

Required Special Equipment:
- Computer with LabView Software
- National Instruments USB 6009 Data Acquisition Card

12.1 Objectives

This lab demonstrates the basic principals of analog to digital conversion and provides a brief introduction to LabView. A LabView VI file is created which utilizes the USB 6009 Data Acquisition Card to convert an analog signal into a digital signal, store it, and display it. The effects of sampling frequency will be explored for both a function generator voltage signal and an audio signal from a stereo amplifier.

12.2 Introduction

A data acquisition system is used to convert an analog signal into a digital signal that can be stored and processed on a computer. The most common type of analog signal acquired by a computer is a voltage output from a sensing device. Examples are voltages due to resistance changes in a strain gage Wheatstone bridge, voltages from an accelerometer charge amplifier, and voltages from a thermocouple amplifier.

A data acquisition system consists of a sample/hold circuit to capture an instantaneous value of a time varying analog voltage signal, an A/D converter to convert this voltage to a digital code, and a computer interface that allows storing and processing of the digital data. These components are packaged on a PC plug-in board, PCMCIA card, or USB device called a Data Acquisition and Control (DAC) card. These cards support various language programming environments including C, FORTRAN, and BASIC. Various software function calls are provided via a software library that gives easy high-level access to the board's capabilities. Acquiring data from the outside world on the computer is a simple matter of calling a function from a program. A DAC card can also be controlled with LabView, a visual programming interface where icons are selected and connected to achieve the desired functionality. A DAC card can support both input and output functions including binary (TTL) I/O, analog I/O, and counter/timer features.

12.3 A/D basics

An analog to digital converter converts a continuous analog voltage signal into a discrete digital signal. The digital signal is represented by a certain number of bits (n) and each combination of bits refers to an output state. Using more bits means that the digital signal can be discretized into

more states (2^n) resulting in a higher resolution. Therefore, the number of bits (n) is often referred to as the resolution of the A/D converter. The resolution of the A/D converter directly affects the quantization size. This and other features are discussed in the following sections.

Quantization Size

Quantization size is a measure of the minimum change in the analog input that can be measured (i.e. the size of the output states). If the change in the analog input is less than the quantization size, then its digital representation will not change (i.e. it will be assigned to the same output state). For example, if an A/D converter has a quantization size of 1V with an output state of 0V-1V, then an analog input of 0V will read the same as a 0.75V. The quantization size depends on the resolution of the converter (n) and the range of possible voltage values and is given by the following equation:

$$Q = \frac{V_{max} - V_{min}}{2^n}$$

where n is the number of bits used to represent the analog signal, Q is the quantization size, and Vmax/Vmin is the maximum/minimum voltage that the A/D converter can measure. The resolution (n) is determined by the specific device that is being used and can be found in the data sheet or user guide. Commercial A/D converters may have a resolution of 8-bits up to 18-bits. The voltage range ($V_{max} - V_{min}$) can often be set with the controlling software to give the desired quantization size.

Frequency Resolution

Another characteristic of A/D conversion is the sampling frequency. This is the rate at which samples are acquired (measured in samples/second or Hz). The sampling frequency has a large impact on how well the digital signal represents the analog signal. **Aliasing** can occur if the signal is sampled too slowly. Aliasing occurs when the frequency content of the digital signal is different from the analog signal. Figure 12.1 shows an example of aliasing. As you can see, the frequency of the measured signal does not represent the frequency of the actual signal.
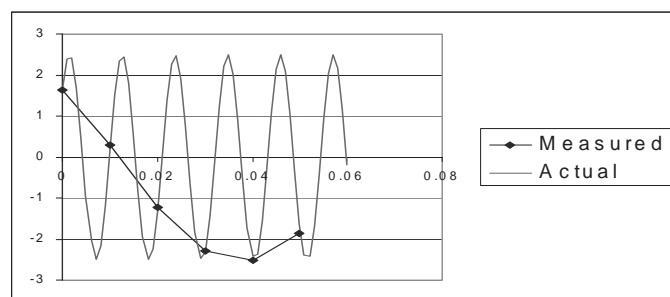

Figure 12.1 Aliasing due to sampling a 90 Hz signal at 100 Hz

Shannon's sampling theorem states that the sampling frequency must be greater than twice

the maximum analog frequency to avoid aliasing. This is restated by the following equation.

$$f_s > 2f_{max}$$

where f_s is the sampling frequency and f_{max} is the maximum frequency of the analog signal. The lower bound on the sampling frequency ($2f_{max}$) is referred to as the **Nyquist frequency**. It is important to note that the sampling rate must be greater than and not equal to the Nyquist frequency in order to retain the frequency information of the analog signal. To accurately represent the amplitude variation of the signal within a period of the waveform, the sampling rate must be well above the Nyquist frequency.

Amplitude Resolution

Sampling at a rate greater than the Nyquist frequency ensures that the frequency of the analog signal is represented accurately but does **not** necessarily ensure that the amplitude of the analog signal is represented accurately. A criterion for accurately representing the amplitude can be developed by considering an upper bound on the change in the analog signal between samples. Between samples the analog signal will change by some amount ΔV. By approximating the derivative as constant over this interval the following equation can be used to relate the change in the signal to the sample time.

$$\Delta V \approx \left(\frac{dV(t)}{dt} \right)_{MAX} \Delta T_S$$

where ΔV is the change in the analog signal, ΔT_s is the time between samples, and $\left(\frac{dV(t)}{dt} \right)_{MAX}$

is the maximum of the derivative of the analog signal. Since ΔT_s is the time between samples it is the reciprocal of the sampling frequency (f_s). By substituting the sampling frequency, the equation becomes:

$$f_S \approx \left(\frac{dV(t)}{dt} \right)_{MAX} \frac{1}{\Delta V}$$

where $\left(\frac{dV(t)}{dt} \right)_{MAX}$ is determined by the signal and ΔV can be set to influence how finely the

amplitude is represented. A small ΔV means that the sampling frequency will be high and the signal will be represented well. The maximum derivative is known if the signal is known. If it is not, then a sinusoid can be assumed at the maximum frequency of the signal. The maximum derivative of a sinusoid of the form $A sin(\omega t)$ is $A\omega$. Using this, the previous equation becomes:

$$f_S \approx \frac{A\omega}{\Delta V}$$

where A is the amplitude of the signal and ω is the maximum frequency. Sampling at f_s ensures that

the signal will not change more than ΔV between samples. How accurately the amplitude is represented can therefore be controlled by selecting ΔV. It is important to remember that due to the approximations made above, ΔV is an upper bound and the signal will change less than ΔV. Also, using a very small ΔV will lead to a very fast sampling rate that may not be possible with some converters, and fast sampling will result in large amounts of data which can be an issue when using devices with limited memory like microcontrollers. ΔV should be chosen to be the largest value that gives adequate results meaning that the sampling frequency is the smallest possible that gives adequate results.

12.4   Introduction to LabView programming

LabView is a graphical programming environment with an intuitive user interface. It has many built-in features for data acquisition and works well with many commercial DAC cards. A very brief description of LabView is presented below.

There are two primary windows in LabView, the **Block Diagram** and **Front Panel** window (see Figures 12.2 and 12.3). Ctrl-T can be used to switch between the windows. The Block Diagram window contains the graphical program that you create, and the Front Panel window contains the user interface. The user interface is used to input control parameters, run the program, and visualize the results (e.g. plot of a waveform).

Additional windows called **palettes** contain the libraries of built-in LabView functions and are used to set the function of the cursor. The **Functions** palette is a library of blocks that can be used in the Block Diagram window (it is only available when the Block Diagram window is active). The **Controls** palette is a library of the functions available for the front panel (it is only available when the Front Panel window is active). The **Tools** palette sets the function of the cursor.

Different tools are used to perform different functions. For example, *connect wire*   is used

to connect blocks, and *operate value*   is used to change the value of a control (described

below). Alternatively, the *automatic tool selection*   will automatically change which

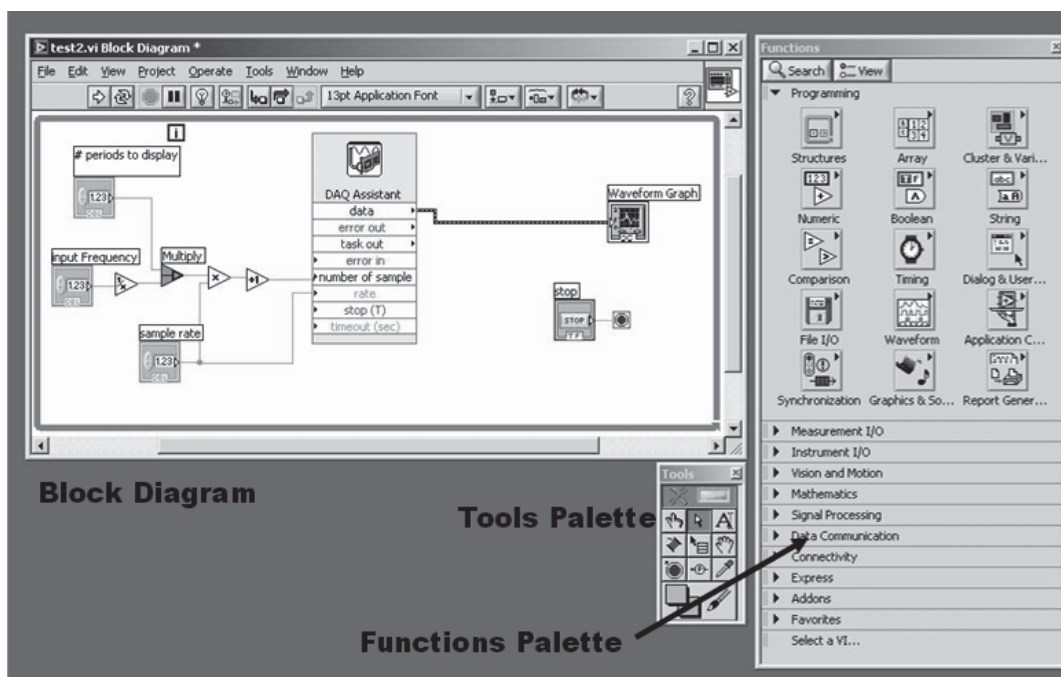tool you are using depending on the location of the cursor.
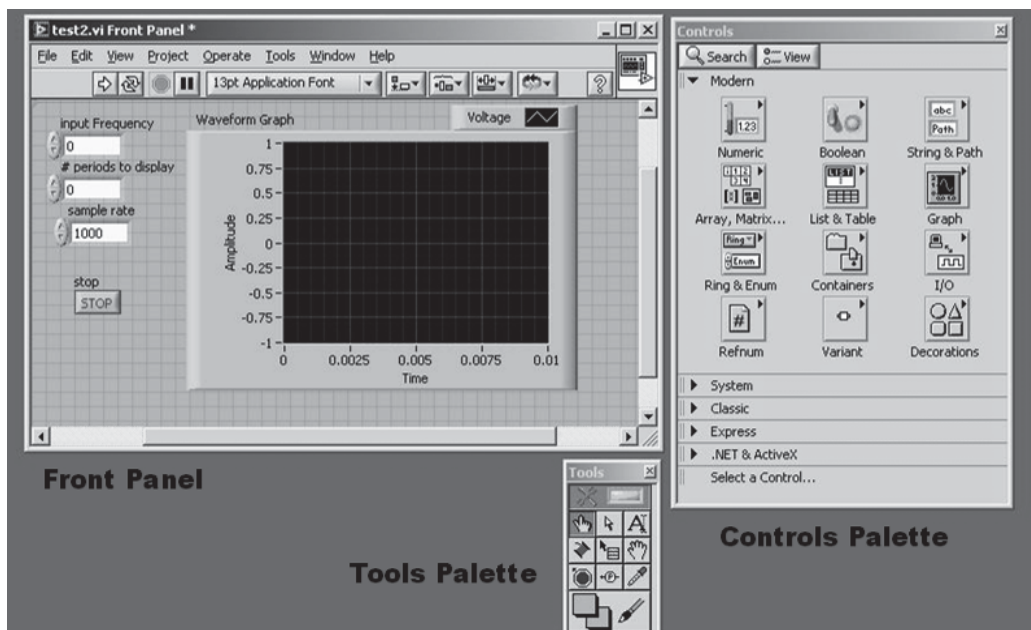
Figure 12.2 Example block diagram

Figure 12.3 Example front panel

A LabView VI file is made up of **objects** (or blocks) with connections between the objects. There are two types of objects: **nodes** and **terminals**. The nodes perform functions such as

acquiring a digital signal from a data acquisition card, multiplication, and signal processing. Terminals are the connections between the block diagram and the front panel. Each component in the front panel appears as a terminal on the block diagram. Every object has inputs, outputs, and parameters that determine its function. For example, an analog to digital conversion block will have an analog signal as its input (hardware input), a digital signal as its output, and parameters such as sampling rate. The output of an A/D conversion block becomes an input to a block that graphically displays the waveform (terminal). The parameters of a block can be set in different ways. One way is to open the properties window for the block (by right-clicking and selecting properties) and enter the values for the parameters. Some of the parameters cannot be changed independently of other parameters, like parameters that define the configuration or mode of the block, and can only be set within the properties window. The parameters that can be set independently of the others, like sample rate and number of samples for an A/D converter block, can be set using inputs. This can be done using a **constant** or **control** (both are terminals). A constant is set in the block diagram and a control is set in the front panel. Figure 12.2 contains control blocks on the left labeled *# periods to display, input frequency* and *sample rate*. Figure 12.3 shows the corresponding controls on the front panel.

The VI file in Figures 12.2 and 12.3 performs A/D conversion using the National Instruments USB 6009 DAC card. LabView can be used to do much more and has many more advanced features that are not mentioned here.

12.5 The USB 6009 data acquisition card

The USB 6009 (see Figure 12.4) is a relatively small external data acquisition card that is connected to a computer through a USB port. It has A/D conversion capabilities as well as D/A conversion, digital I/O, and counters/timers. The I/O are connected with wire (e.g., 16-28 AWG wire) to the detachable screw terminals.
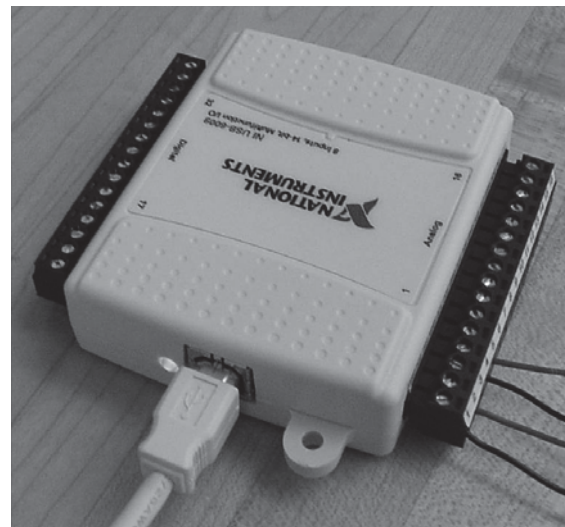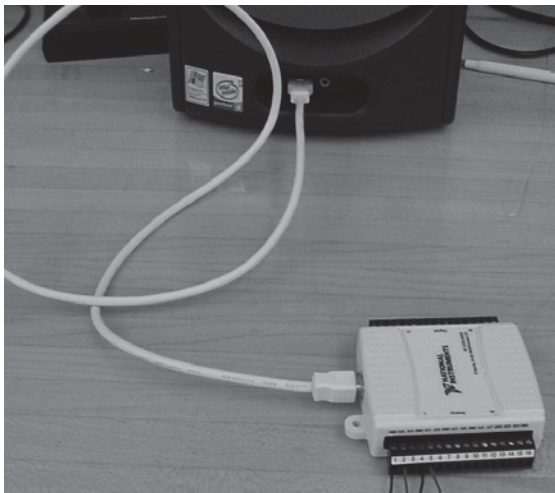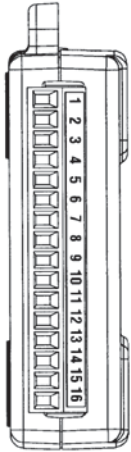

Figure 12.4 The USB 6009 connected to the computer with the screw terminals attached
(NOTE - only two signal wires are used in the Lab)

Terminals 1-16 are used for analog I/O, and terminals 17-32 are used for digital I/O and counter/timer functions (see Figure 12.5). Notice that the analog terminals are different depending on which mode the device is in, single-ended mode (also known as reference-signal-ended or RSE) or differential mode. In single-ended mode the positive voltage signal is connected to an AI terminal and the negative voltage signal is connected to a GND terminal. This mode uses two terminals allowing eight analog inputs (AI0-AI7). The maximum voltage range in this mode is -10V to 10V. Differential mode can be used to obtain a larger voltage range. This mode measures the difference between two signals, AI+ and AI-; each referenced to GND. A voltage range of -20V to 20V can be achieved, but the maximum voltage on one pin (AI+ or AI-) referenced to ground is ±10V. This means the 20V amplitude sine wave cannot be measured using a single pin (AI+ or AI-). A combination of two 10V amplitude sine waves that are 180 degrees out of phase would need to be applied to AI+ and AI-. Differential mode uses one more wire than single-ended mode, so only four analog inputs are available. A description of each signal is summarized in Figure 12.6.

Module	Terminal	Signal, Single-Ended Mode	Signal, Differential Mode
	1	GND	GND
	2	AI 0	AI 0+
	3	AI 4	AI 0–
	4	GND	GND
	5	AI 1	AI 1+
	6	AI 5	AI 1–
	7	GND	GND
	8	AI 2	AI 2+
	9	AI 6	AI 2–
	10	GND	GND
	11	AI 3	AI 3+
	12	AI 7	AI 3–
	13	GND	GND
	14	AO 0	AO 0
	15	AO 1	AO 1
	16	GND	GND
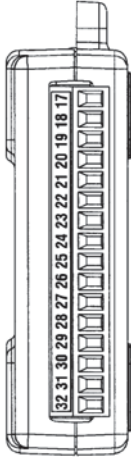
Module	Terminal	Signal
	17	P0.0
	18	P0.1
	19	P0.2
	20	P0.3
	21	P0.4
	22	P0.5
	23	P0 6
	24	P0.7
	25	P1.0
	26	P1.1
	27	P1.2
	28	P1.3
	29	PFI 0
	30	+2.5 V
	31	+5 V
	32	GND

Figure 12.5 Analog (1-16) and Digital (17-32) pin assignments of the USB 6009

Signal Name	Reference	Direction	Description
GND	—	—	**Ground**—The reference point for the single-ended AI measurements, bias current return point for differential mode measurements, AO voltages, digital signals at the I/O connector, +5 VDC supply, and the +2.5 VDC reference.
AI <0..7>	Varies	Input	**Analog Input Channels 0 to 7**—For single-ended measurements, each signal is an analog input voltage channel. For differential measurements, AI 0 and AI 4 are the positive and negative inputs of differential analog input channel 0. The following signal pairs also form differential input channels: <AI 1, AI 5>, <AI 2, AI 6>, and <AI 3, AI 7>.
AO 0	GND	Output	**Analog Channel 0 Output**—Supplies the voltage output of AO channel 0.
AO 1	GND	Output	**Analog Channel 1 Output**—Supplies the voltage output of AO channel 1.
P1.<0..3> P0.<0..7>	GND	Input or Output	**Digital I/O Signals**—You can individually configure each signal as an input or output.
+2.5 V	GND	Output	**+2.5 V External Reference**—Provides a reference for wrap-back testing.
+5 V	GND	Output	**+5 V Power Source**—Provides +5 V power up to 200 mA.
PFI 0	GND	Input	**PFI 0**—This pin is configurable as either a digital trigger or an event counter input.

Figure 12.6 Signal descriptions for the USB 6009

Another difference between differential and single-ended mode is the resolution of the analog inputs. Differential mode has a resolution of 14 bits where single-ended mode has a resolution of 13 bits.

The analog input converter type is successive approximation and the maximum sampling rate is 48 thousand samples per second (kS/s). The device contains one analog to digital converter that is multiplexed to each input (which is one of the reasons the small package is possible). For more information on the device specifications, refer to the user guide and specifications document.

12.6   Creating a LabView program that utilizes the USB 6009

This example assumes that LabView 8.0 is being used. If an older version is used the specific commands will be different but the general procedure will be the same. This procedure also assumes that the USB 6009 is already set up to interface with the computer according to the instructions that come with the device.

Opening a blank VI file

(1)   Start LabView [Start > Programs > National Instruments > LabView 8.0 > LabView].

(2)   Click [Blank VI] to open a new program. The *Block Diagram* window and the *Front Panel* window should appear. If only one is open, then under the Windows menu click [Show Block Diagram or Show Front Panel]. Some other small windows may also be open.

(3)   Open the *Functions* palette if it is not open. From the *Block Diagram* window, under the View menu, click [Functions palette] to open the *Functions* palette.

Creating node blocks

(1)   From the *Functions* palette select [Measurement I/O > NI-DAQmx]. Drag the *DAQ Assist* icon onto the *Block Diagram*. A DAQ Assistant window should appear.

(2)   Connect the USB 6009 device to the computer (Figure 12.4). The green light should be blinking. From the *DAQ Assistant* window select [Analog Input > Voltage > ai0 > Finish]. If the ai0 does not appear, then press the plus next to "Dev1 (USB-6009)" to display the available analog input channels.

(3)   A new window will open displaying the properties of the DAQ Assistant block (see below).
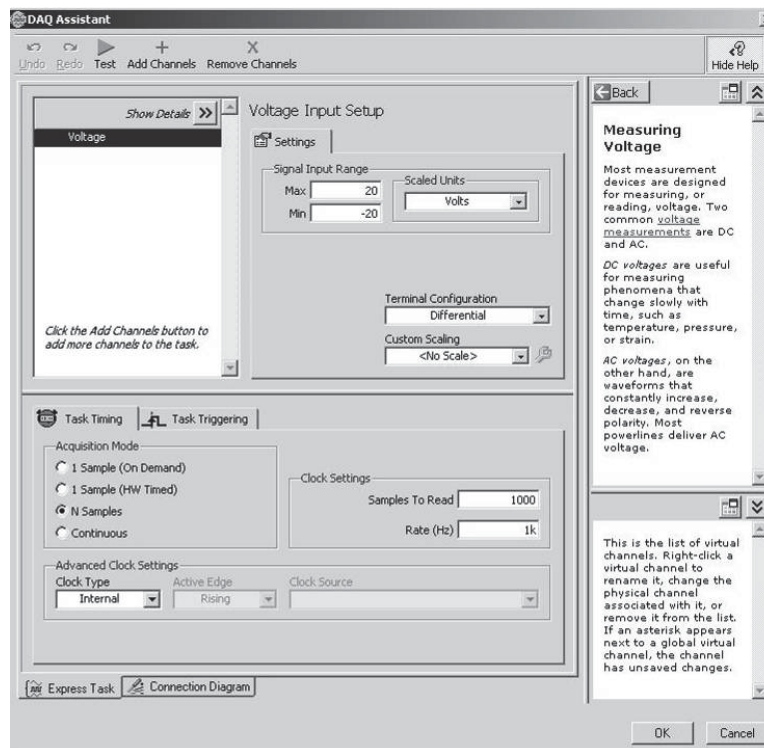
(4)   Under *Settings* set the maximum and minimum values for the Signal Input Range based on the amplitude of the input and the desired quantization size.

(5)   Under *Settings* set the *Terminal Configuration* to *RSE* (single-ended mode).

(6)   Under the *Task Timing* tab set the acquisition mode to *N samples*. The wiring diagram can be viewed by selecting the *Connection Diagram* tab towards the bottom of the window assuming an appropriate range was selected (see step 1 in the Lab Procedure to follow).

(7)   Select [Ok] to close the *DAQ Assistant* properties window. This may be opened later by right-clicking on the *DAQ Assistant* block.

Creating terminal blocks

(1)   Select the spool of wire icon   from the *Tools* palette (accessible under the *View* menu). Right-click on the *rate* input (arrow on the side of the block) on the DAQ Assistant block and select [create > control]. A block labeled rate should appear with a wire connected to the *DAQ Assistant* block.

(2)   Repeat this to create a control for the number of samples input. These two controls will appear in the *Front Panel* window.

(3)   Activate the *Front Panel* window and open the controls palette if it is not open. From the Front Panel window under the *View* menu, select *Controls Palette* to open the *Controls* palette.

(4)   From the *Controls* palette select [modern > graph] and drag the *Waveform Graph* icon onto the *Front Panel* window. A block labeled *Waveform graph* will appear in the *Block Diagram* window.

(5)   Right-click on the graph and select [properties]. Under the *Scales* tab select [Amplitude (Y-axis)] in the top pull-down menu. Deselect *Autoscale* and set the maximum and minimum to the values used for the signal input range on the *DAQ Assistant* block. Click [OK] to close the properties window.

(6)   Select the *Block Diagram* window and select the wire spool icon   (or automatic icon) on the Tools palette. Click on the data output of the *DAQ Assistant* block and on the *Waveform Graph* block. A wire will now connect the two blocks.

Running the program

(1)   Connect the analog signal to the USB 6009. The positive voltage signal is connected to screw terminal 2 (AI0) and the negative voltage signal is connected to screw terminal 1 (GND).

(2)   Select the *Front Panel* window and select the *operate value*   icon from the *Tools* palette (or automatic icon).

(3)   Set the *rate* and *number of samples* controls to appropriate values.

(4)   Under the *Operate* menu select [run] to run the program (or use the large right-arrow on the toolbar under the menu bar). A waveform should appear on the *Waveform Graph*. A picture of the waveform can be saved to a file by right-clicking on the waveform and selecting [Data Operations > Export Simplified Image…].

12.7   Laboratory Procedure / Summary Sheet

Group:_____   Names:_____

(1)   Calculate the maximum allowable voltage range resulting in a measurement accuracy (quantization size) of 1 mV. The resolution of the USB 6009 is 13 bits in the mode that we will be using.

$V_{max} - V_{min} =$ _____V   (round down)

Select a maximum and minimum voltage to be used in the LabView program if a $2\sin(200\pi t)$ V signal is to be measured.

$V_{max} =$ _____V $V_{min} =$ _____V

(2)   For the following input signal $2\sin(200\pi t)$V, what is the Nyquist frequency? What is the sampling frequency that ensures the signal changes less than 0.4V (1/10 of V_{p-p}) between samples?

Nyquist frequency = _____Hz $f_S =$ _____Hz

Fill out the following table to display 4 periods of the analog signal. Round up and remember to account for the starting sample (i.e. add 1).

Sampling Rate (Hz)	Number of Samples
90	
150	
175	
200	
500	
1000	
2000	
3000	
4000	

(3)   Follow the procedure from Section 12.6 to create a LabView VI file.

(4)   Use a function generator to create the analog $2\sin(200\pi t)$ and connect the output to the USB 6009. Use the values from parts (1) and (2) as parameters in the LabView program and sketch the resulting waveforms on separate paper.

(5)   In this portion of the lab you will sample music at different sampling rates and listen to the resulting waveform.

Add the *Play Waveform* block to the block diagram which can be found in the Functions palette [Programming, Graphics and Sound, Sound, Output]. Press [OK] when the configuration dialog window appears. Wire the *data* input of the *Play Waveform* block to the data output of the *DAQ Assistant* block. Create a constant for the timeout input of the *DAQ Assistant* (using the same method used to create a control for the rate input) and set it to 30 (the letter icon will need to be selected from the Tools palette). Setting the timeout to 30 allows up to 30 seconds of music to be recorded. The block diagram should now look like the following figure.
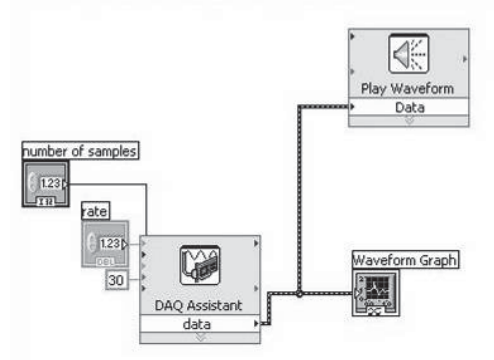
Complete the following table in order for 15 seconds of music to be recorded for each sampling rate.

Sampling Rate (samples/sec)	Number of Samples
40,000	
20,000	
15,000	
10,000	
8,000	
6,000	
4,000	
2,000	
1,000	

Connect the two wires from a 3.5mm audio plug to the inputs of the USB 6009 and insert the plug into the output jack of a music player. Play and sample some music staring with a sampling rate of 40,000. At which frequency did the music start to sound bad? The maximum frequency that people can here is 10-20 kHz, but most of the frequencies in an audio signal are well below this.

Lab 12

LAB 12 QUESTIONS

Group:_____   Names:_____

(1) For part (2) at what frequencies did you see aliasing?

(2) For part (1), what would the voltage range need to be to get a quantization size less than 0.2mV? Is it possible to measure $2\sin(200\pi t)$ at this quantization size?

(3) What resolution A/D converter should be used to measure a signal that can range between -8V to 8V with an accuracy of 10mV?

(4) To measure the signal $2\sin(200\pi t)$, draw a wiring diagram showing how you would connect the function generator to the USB 6009 in Differential Mode.

(5) With the USB 6009 in Differential Mode, what two signals would need to be connected to AI+ and AI- to give a $15\sin(2\pi t)$ waveform? (Hint: see Section 12.5.)

185