Mathematical Aspects of Biomedical Electronic System Design

Professor Chandramani Singh

Department of Electronic Systems Engineering

Indian Institute of Science, Bangalore

Week – 12

Probability Distribution and Biomedical Systems Design

Lecture – 40

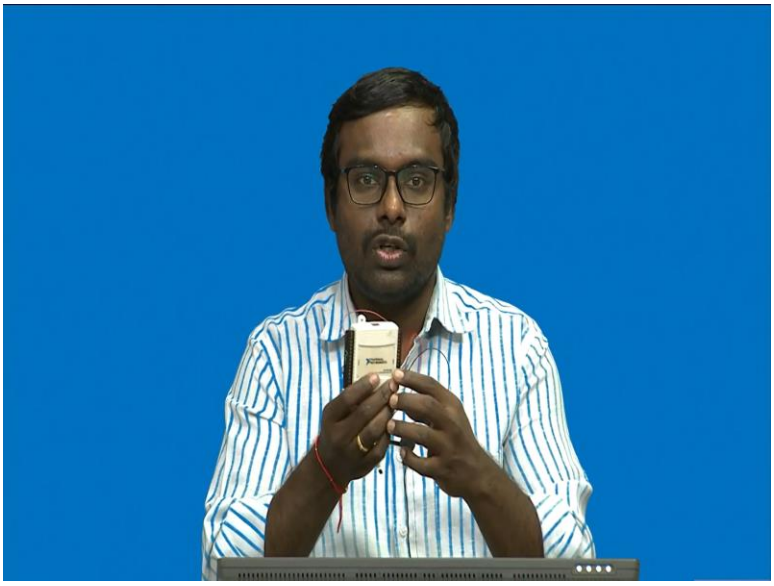
Demo of Temperature data acquisition system Using Lab view

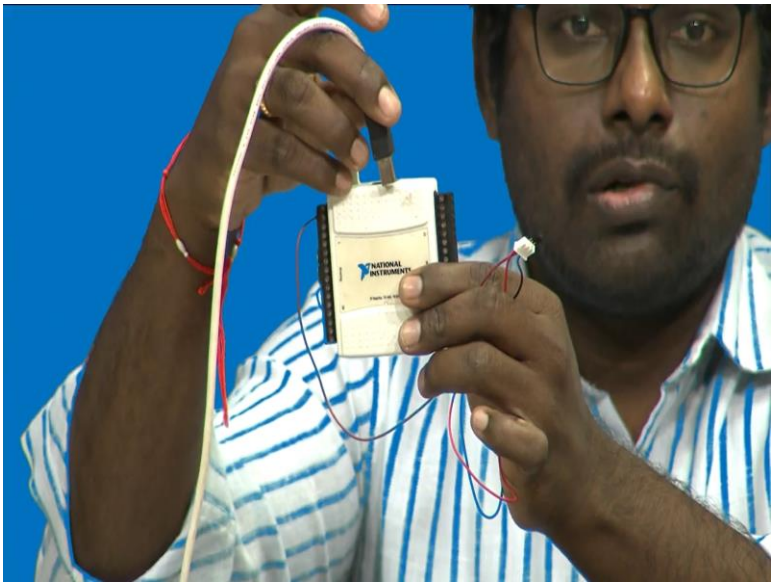
Hello all, welcome to the module. So, in the last video we discussed the datasheet of the temperature sensor that we are going to interface to the PC. We have also seen the datasheet of data acquisition device NI USB 6008 data datasheet of USB DAQ device. And we discussed about what different channels, how many channels that analog channels this DAQ device has at the same time how many digital ports that device has. And the internal construction or internal circuit of blocks internal circuit blocks of the DAQ has been discussed. Now, in the last video we have also seen the connections to interface the sensor to the ESP DAQ.

Now we will use a LabVIEW software, we will integrate the data acquisition device to the sensor and the data acquisition device to the laptop such that the change due to the temperature which is reflected as an output voltage will be acquired by a data acquisition to the PC and it displays the temperature corresponding temperature output in terms of voltage. Since our idea was to understand the room temperature, what we do, we will convert this corresponding voltage value into an output, we will represent it as any degree centigrade so that the room temperature can be displayed in the PC.

So, the process of integrating the process of developing a code or making a code using a LabVIEW software and the display of the temperature value will be discussed in this video session. So now, as we have already seen in the last video.

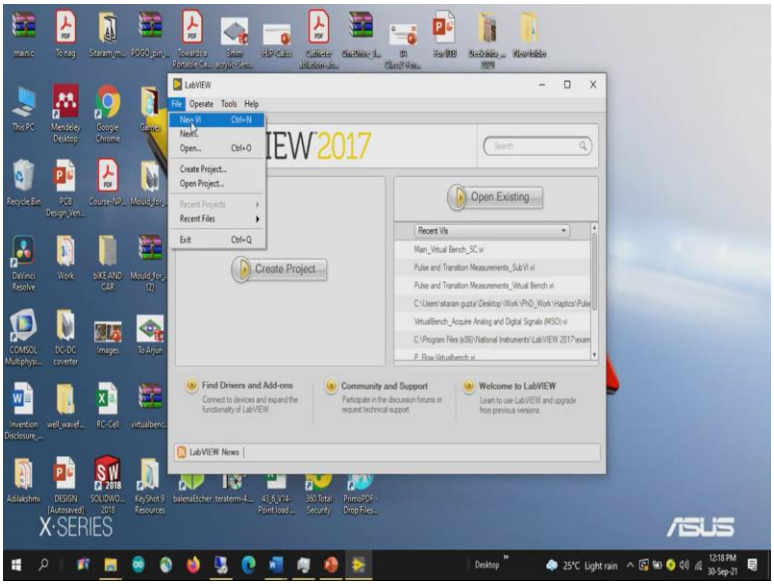
(Refer Slide Time: 02:35)

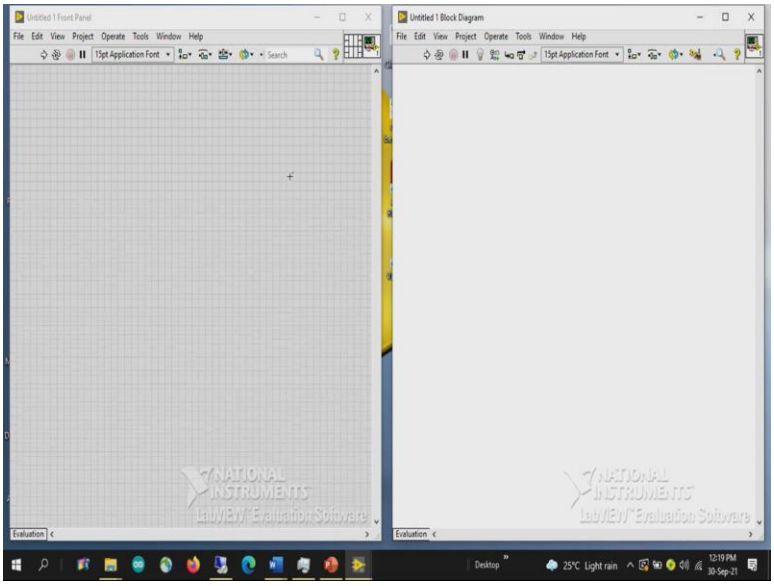




So, this is the USB DAQ we discussed, and we have already made the connections to connections from the sensor to the USB DAQ and this is our LM35 sensor which is connected to the RMC cable. Now, since the connections are made, this is a port the place where we will be integrating. So, this place so here we can see the LM35 temperature sensor which is connected to the RMC cable, and which is interfaced directly to the USB DAQ. Then the cable USB cable will be connected to NI DAQ and the other end of the USB cable will be connected to the laptop so that we can detect the USB DAQ in the laptop. Now we will see we will go to the laptop.

(Refer Slide Time: 03:31)



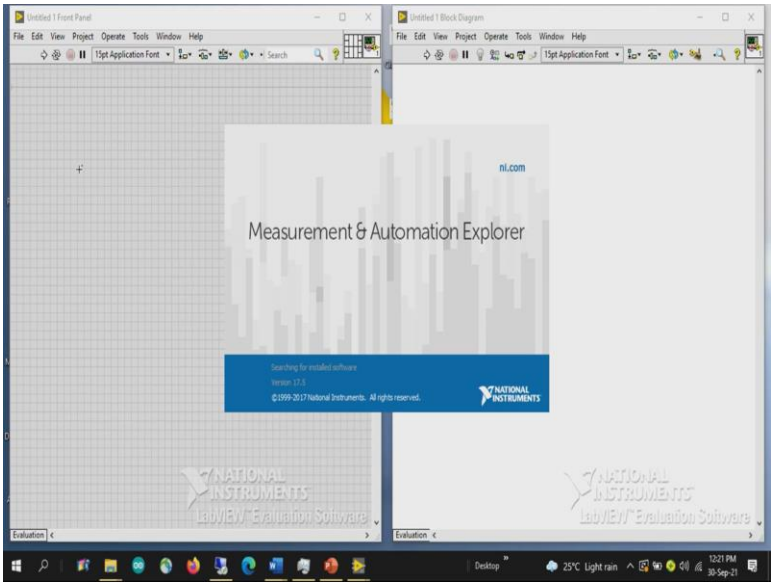


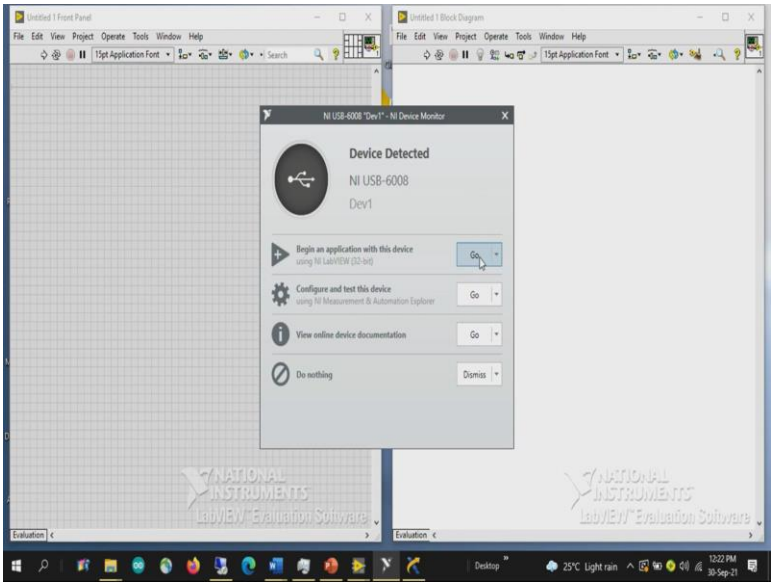
So, I am opening a new VI. So, in the lab view, here we have two different panels, this is called Front panel and this is called the right side panel is called Block diagram. So, the Front panel is called User Interface Module. So, when the program is running user will have interaction only with the display devices as well as a control which are placed on the front panel. So, based upon the selection of these parameters, the corresponding values will be taken by the block diagram and it will be given as an input to the next subsystems or next to functions or next blocks that we are going to place it here.

So, while the program is in running state, no user will have an interaction with the block diagram panels at all. So, this block diagram is nothing but lines of code which we write in any microcontroller programming. So, since it is a graphical user interface, since it is a G code. LabVIEW is also called as the G code. So, the lines of code will be completely eliminated, it will not be available in the LabVIEW software.

Instead, there are different icons available, which we have to drag and drop onto the block diagram and connecting them using a wire. So, the lines of text are very, very small, provided if you require or if you are interested to write in terms of a C code or a text code are a function based code or MATLAB code there are toolboxes available in order to do that. So, now, what I will do I will interface I will connect the DAQ USB cable to the laptop.

(Refer Slide Time: 05:23)

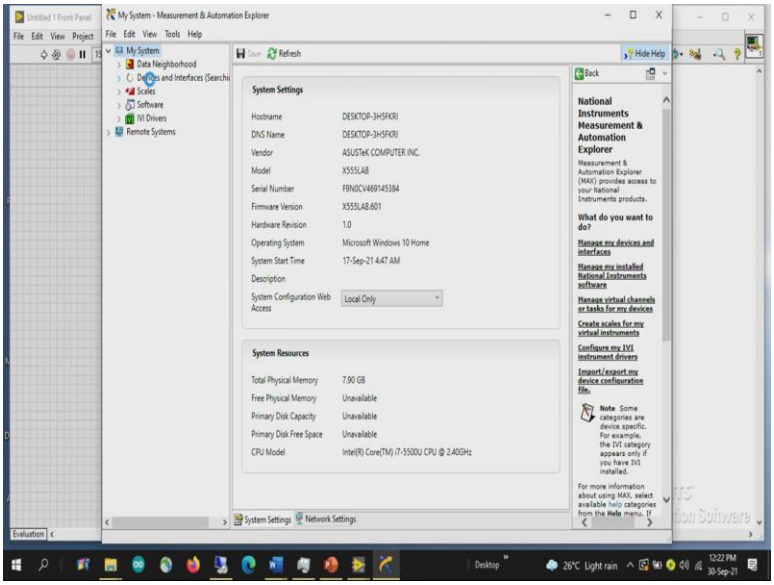


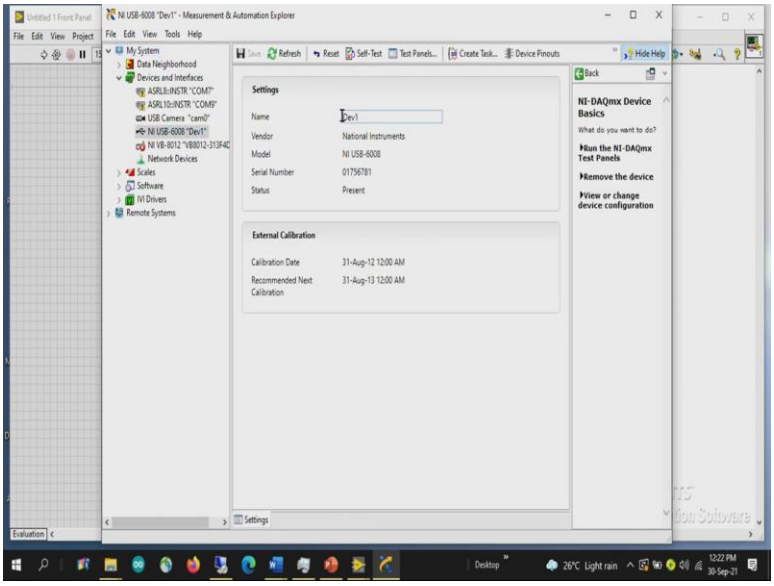


Then to understand the connection, we can open a Max, Max is generally sandwiched between the LabVIEW software and a PC. So, all the drivers all the instruments connected to the PC can be visualized in the measurement and automation explorer window. Now, since we have connected the device, the device has been detected by the software. So, here we can see a pop-up window has appeared indicating NI USB 6008 the device one NI device monitor.

So, the device which we have connected is NI USB 6008. So, if you want to write a code here, once you get a pop up before opening the LabVIEW software, we can even just click on this point to begin an application with this device or if you want to open a max window even that can be opened using the pop up window.

(Refer Slide Time: 06:20)



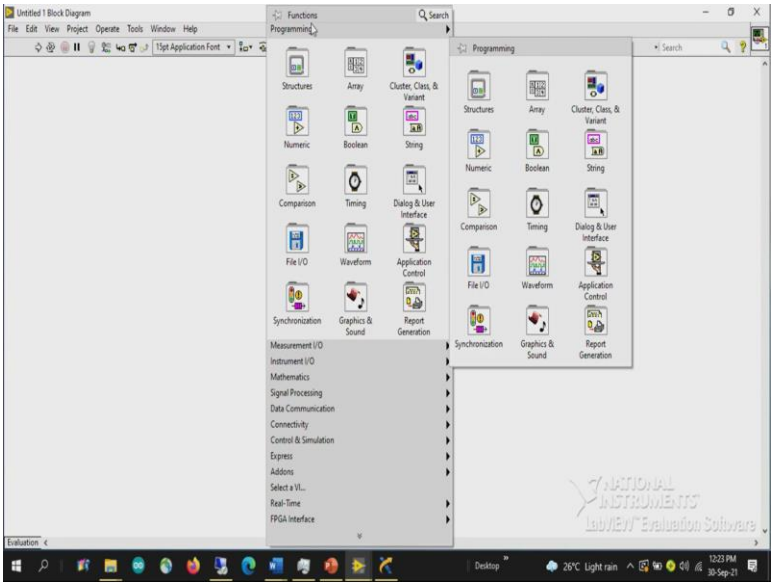


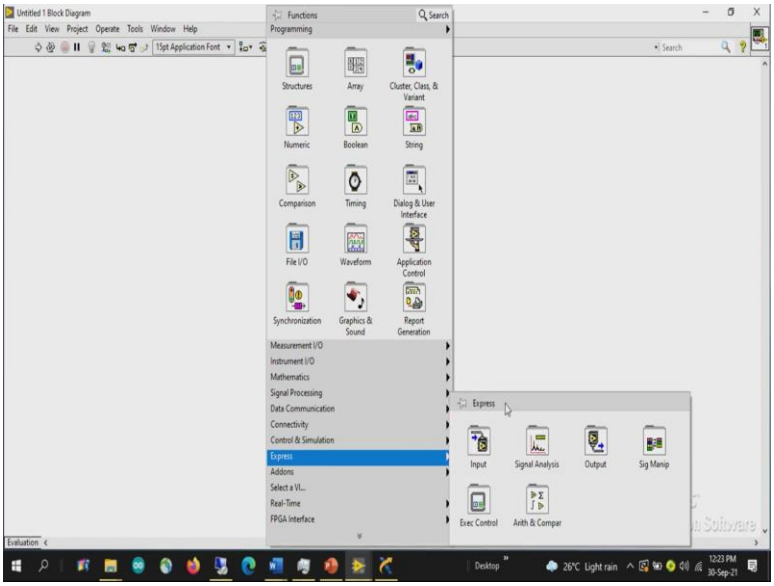
So, if you go to the max measurement and automation explorer in the second tab on the left side under my system, there is devices and interfaces column just expand this tab so that we can see what all the devices are that have been interfaced or connected to this particular system. So, here we can see USB 6008 is already interfaced we can see and it is represented with Dev1, the one of course in case if you want we can even change that now minimizing it.

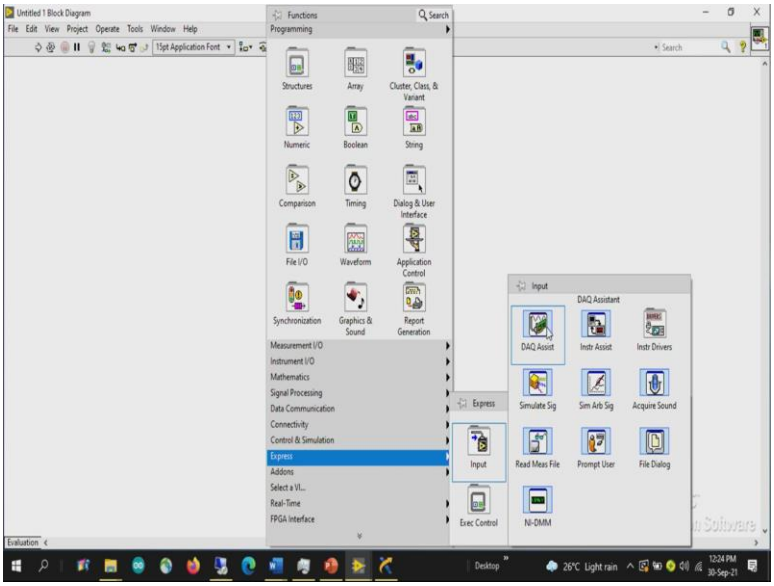
So, to acquire the data using the NI USB 6008 to the PC. So, we have to write lines of code, I will show you the simplest way using Express ways instead of going with the low level ways

where you have a more flexibility to acquire the data from the device. I will simply use Express VI, so where the name itself says that experts, within an instant you can create the file.

(Refer Slide Time: 07:31)

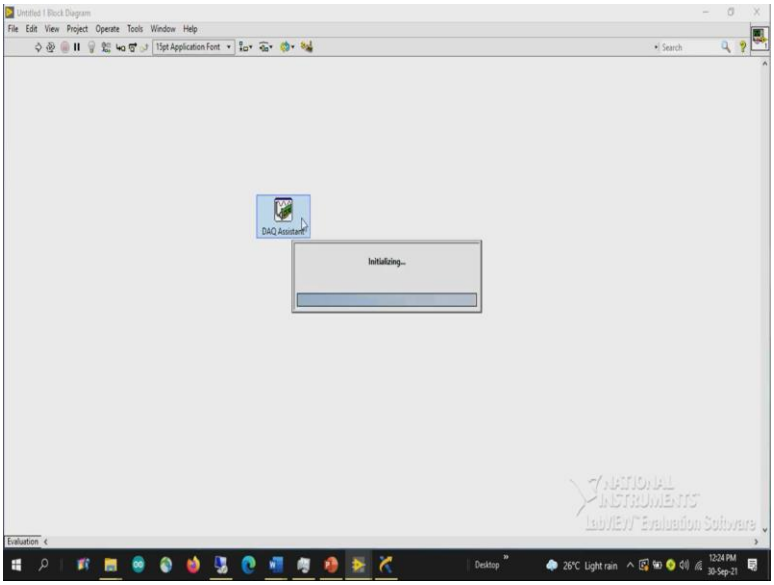


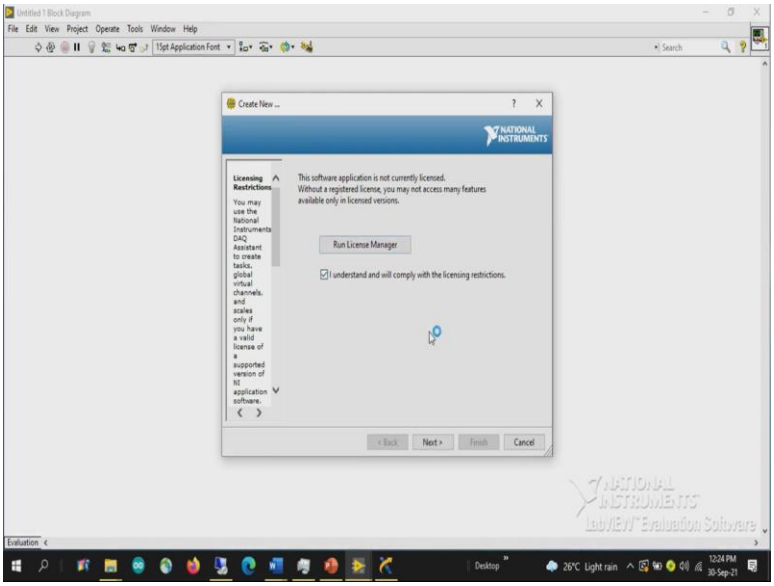




So, in order to access the Express VA related DAQ, you can right click on to the block diagram window. So, you can see a palette pop up when you right click on the block diagram window. This is called functions palette, under measurement I/O or under Express window you can observe input and output this. Input and output is relevant to what are all the devices it need not be only for DAQ input output, it can even for the visa input output and any other Express related VI or virtual instrument or sub programs or sub VI's will be available at this particular place. So since, it is a data acquisition device, which we have to acquire the data from, I will be selecting DAQ assist data acquisition assistant DAQ meaning data acquisition.

(Refer Slide Time: 08:30)

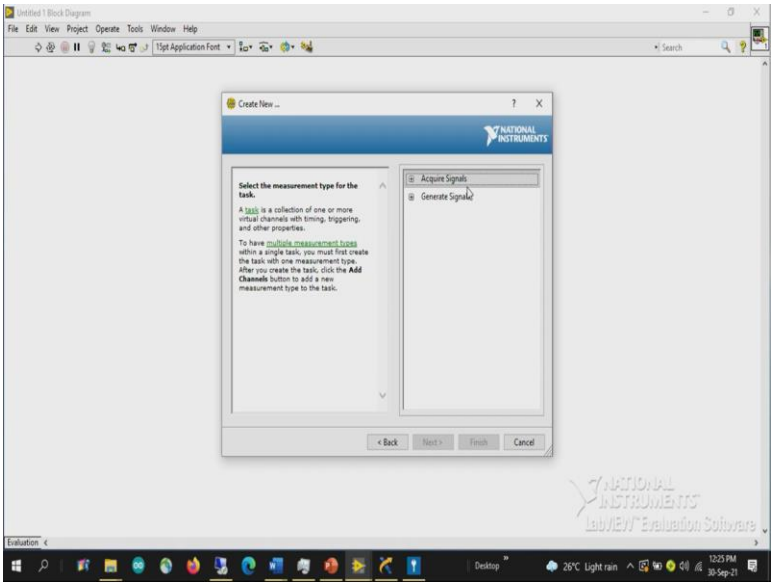


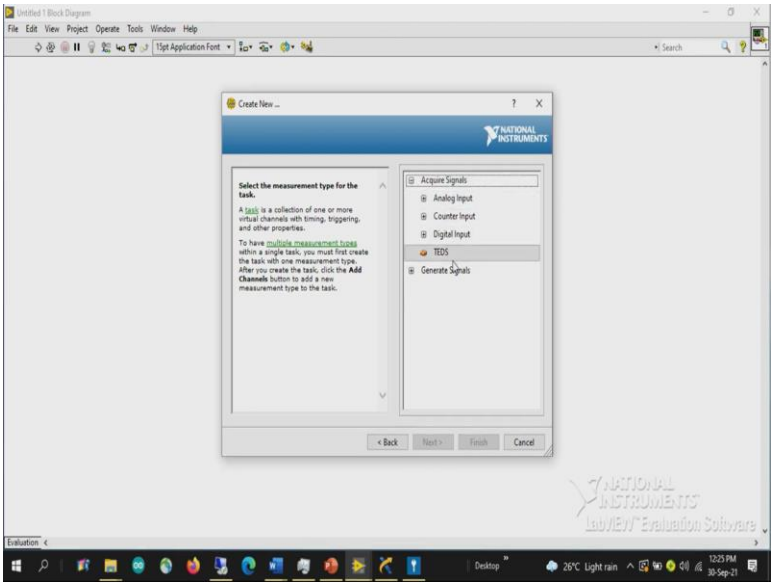


Select it once you click it, the window will be created here. So, since it is an express VI, the program will automatically pop up with a new window where user has to select the properties or different parameters in order to access the acquired the data from DAQ. So, what we have to do, any process in order to acquire the data from the real world to the PC, we have to give some instructions to the PC.

What type of a signal we want to acquire whether you want to acquire is first of all you have to indicate whether you want to acquire a signal or a generated signal meaning you are taking the data into the system or you are giving the data from the system to the output world.

(Refer Slide Time: 09:24)

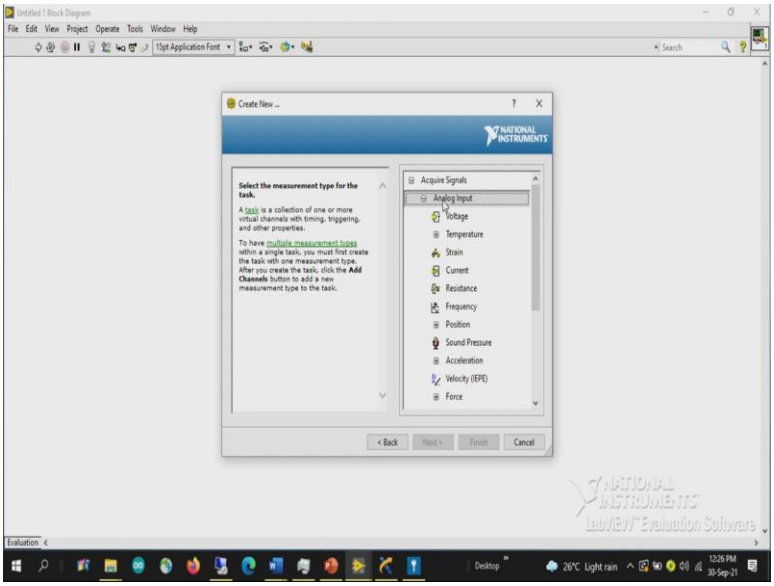


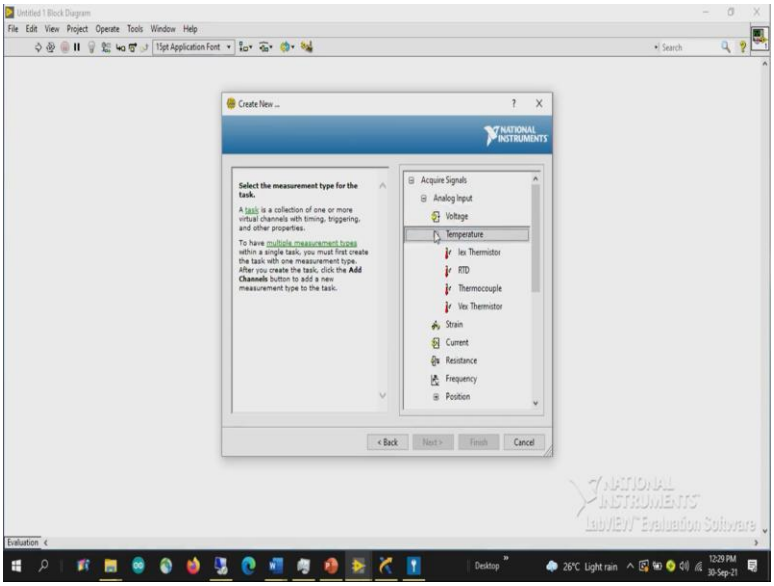


Then once you make a selection accordingly, to create a new task considering the fact that the signal has to be acquired by the PC. So, I am selecting an acquire signal. And there are different ways of different types of signals we can acquire. Since this particular USB DAQ supports with analog, digital and accountant input, so we should provide a few more information to the PC to create the background file that is required to acquire the data from the PC, from the USB 600.

So, since our intention was to acquire, the output voltage is produced by the active sensor in which in this case is a temperature sensor as a temperature sensor gives an output in terms of output voltages. So, we require to acquire analog input which is a voltage.

(Refer Slide Time: 10:24)





Even in the analog there are few other modules available, data acquisition devices available within NI or third-party modules too. So, when you interface it, what it does in the sense, it has an onboard signal conditioning modules to the data acquisition device because of the existence of presence of the onboard signal conditioning circuits. We can directly acquire the data from the temperature sensor or from the strain gauges or from say a mic or even from the IEP based sensors so that after acquiring the data it automatically process amplifies filters and gives in terms of the physical measure and parameter.

What I mean in in our today's experiment since we are using a simple DAQ system where it does not have any kind of signal conditioning circuits to convert from the resistance to voltage or from other physical electrical parameter to the voltage and amplify the signal and the filtering of the signal and represent in terms of degree centigrade. In our case we have to build all the systems.

But there are a dedicated data acquisition devices available in the market where interfacing particular sensor, it automatically converts the electrical parameter change into a voltage, acquires in terms of a voltage and again process the analog acquired analog input data. And converts with respect to the output value or displays in terms of the physical measurement value.

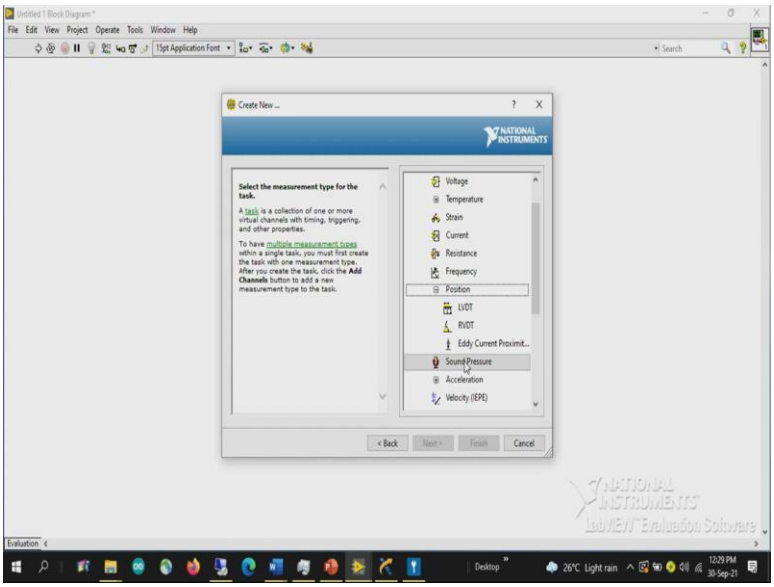
What I mean in case if you are interfacing a force sensor, which is a strain gauge based force sensor and as we discuss a two element varying bridge, single element varying bridge, four element varying bridge, which helps us to convert resistance to voltage and produces an output voltage. Using such a module when we interface a load cell to such a modules it automatically

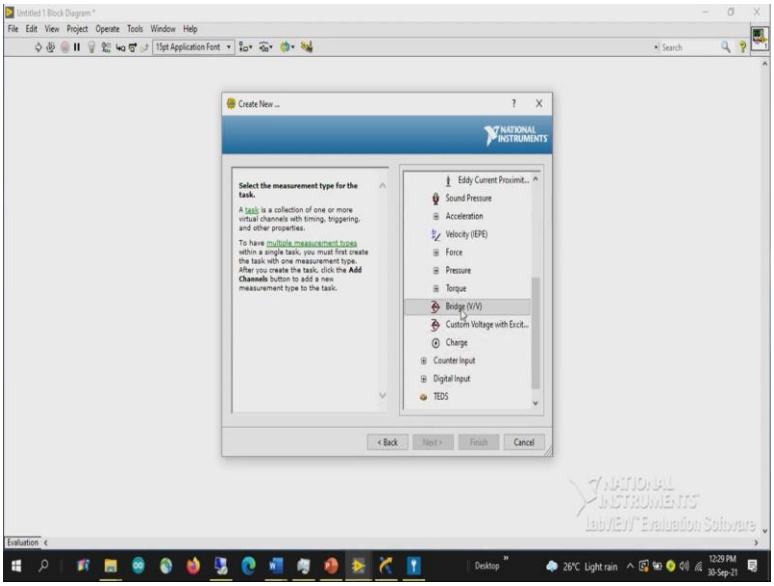
displays the force value instead of representing in terms of a voltages it automatically displays in terms of force values.

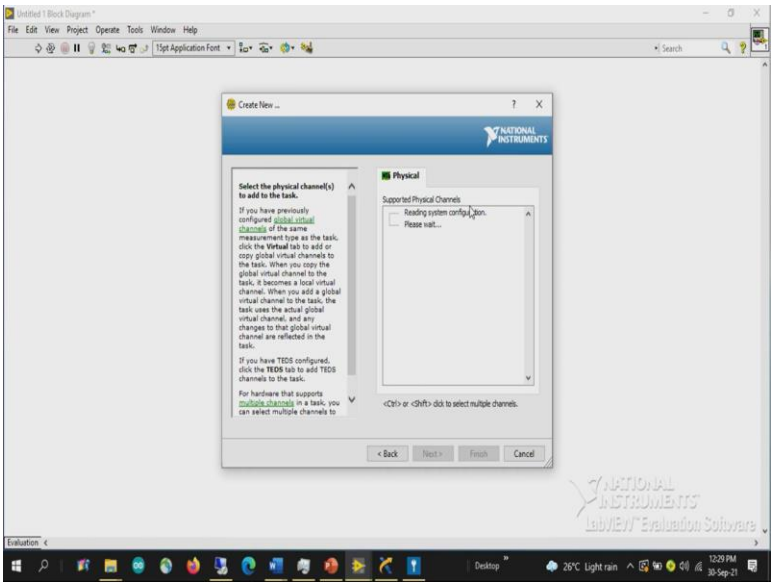
How it is being done, since we know the sensitivity factor right since we know the base resistance in that case since we know the sensitivity factor by providing the sensitivity factor which we observed from the datasheet as the user has to provide some information to the PC. It processes the data based upon that the analog acquired input voltage will be converted to the corresponding physical measurand or the force in that case and displays in terms of a force values.

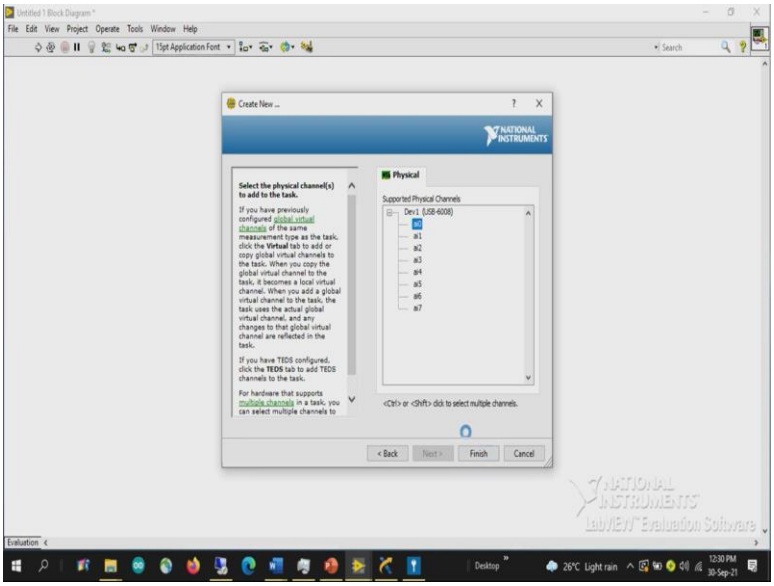
Since this is a simple DAQ all the procedure we have to follow up here. So, it can only measure in terms of voltages, it cannot measure the strain, but provided if you consider a proper signal conditioning circuits relevant to that with the C DAQ module or with other PXA modules or some other data acquisition modules you can even display in terms of temperature too. Similarly strain, current, resistance, frequency and some and some other position kind of sensors.

(Refer Slide Time: 13:39)




Untitled 1 Block Diagram *
File Edit View Project Operate Tools Window Help
15pt Application Font
Search
Create New ...
NATIONAL INSTRUMENTS
Select the measurement type for the task.
A task is a collection of one or more virtual channels with timing, triggering, and other properties.
To have multiple measurement types within a single task, you must first create the task with one measurement type. After you create the task, click the Add Channels button to add a new measurement type to the task.
Eddy Current Proximit...
Sound Pressure
Acceleration
Velocity (IEPE)
Force
Pressure
Torque
Bridge (V/V)
Custom Voltage with Excit...
Charge
Counter Input
Digital Input
TEDS
< Back
Next >
Finish
Cancel
NATIONAL INSTRUMENTS
LabVIEW Evaluation Software
Evaluation
Desktop
26°C Light rain
12:29 PM
30-Sep-21

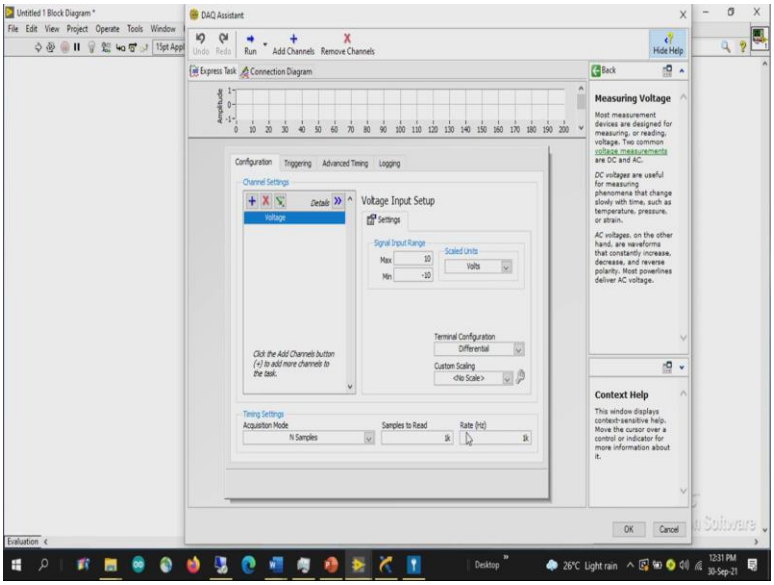

Untitled 1 Block Diagram *
File Edit View Project Operate Tools Window Help
15pt Application Font
Search
Create New ...
NATIONAL INSTRUMENTS
Select the physical channel(s) to add to the task.
If you have previously configured global virtual channels of the same measurement type as the task, click the Virtual tab to add or copy global virtual channels to the task. When you copy the global virtual channel to the task, it becomes a local virtual channel. When you add a global virtual channel to the task, the task uses the actual global virtual channel, and any changes to that global virtual channel are reflected in the task.
If you have TEDS configured, click the TEDS tab to add TEDS channels to the task.
For hardware that supports multiple channels in a task, you can select multiple channels to
Physical
Supported Physical Channels
Reading system configuration.
Please wait...
<Ctrl> or <Shift> click to select multiple channels.
< Back
Next >
Finish
Cancel
NATIONAL INSTRUMENTS
LabVIEW Evaluation Software
Evaluation
Desktop
26°C Light rain
12:29 PM
30-Sep-21

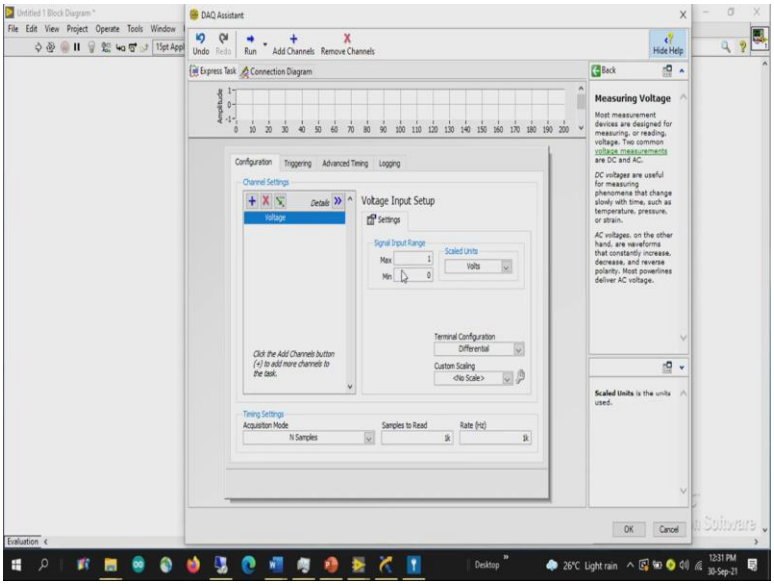


Also, we have Sound Pressure there are modules available for the sound pressure, acceleration and velocity sensor and so forth. And as we discussed about the bridges, V by V custom voltage with excitation. So, in this case, I am selecting your voltage. So, based upon the device that you connect the available number of channels with the device will be displayed at this point. So since, as we discussed our intention was to acquire a differential data. This USB DAQ can support up to 4 channels, but whereas if we are using a single ended it will be supported with 8 channels.

So, here if you see since we have not selected whether we are going with a single ended data or a or a differential data. So, all the available channels, 8 channels with reference to single ended ground is shown here. So accordingly, since the sensor is connected between second terminal as the third terminal is internally grounded. So, what we can do, we can take a single channel and finish it off.

(Refer Slide Time: 14:52)

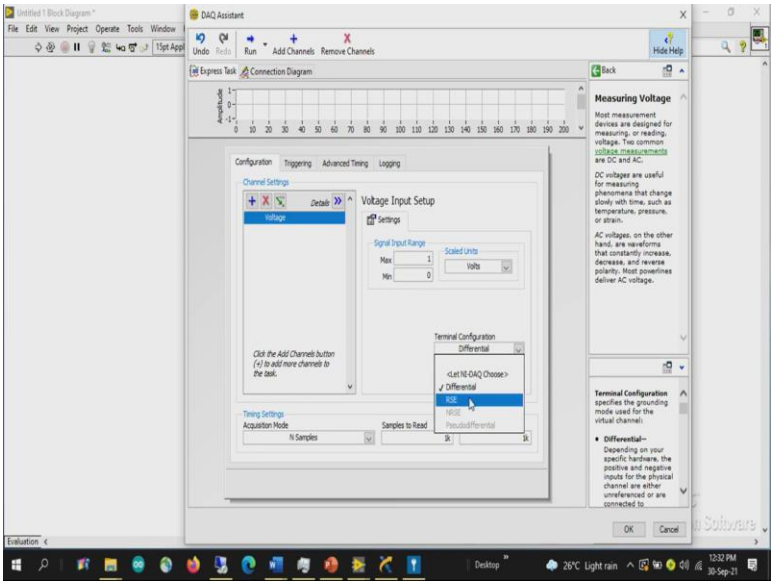


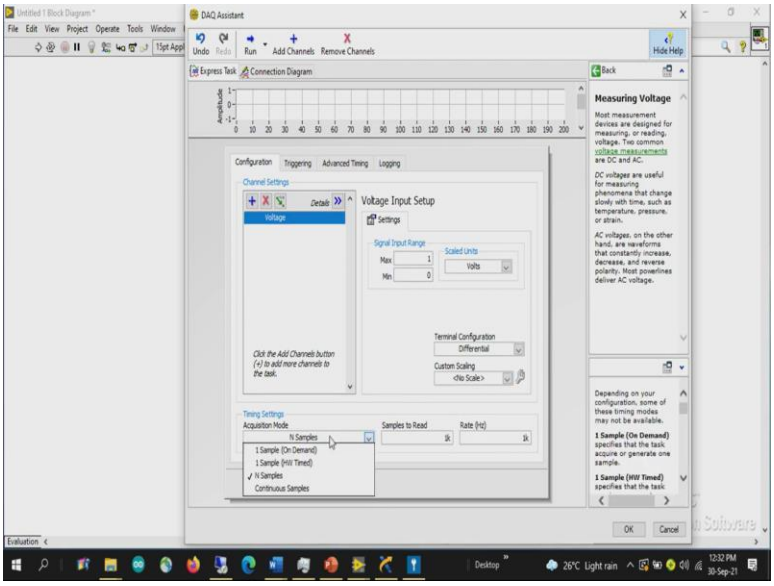


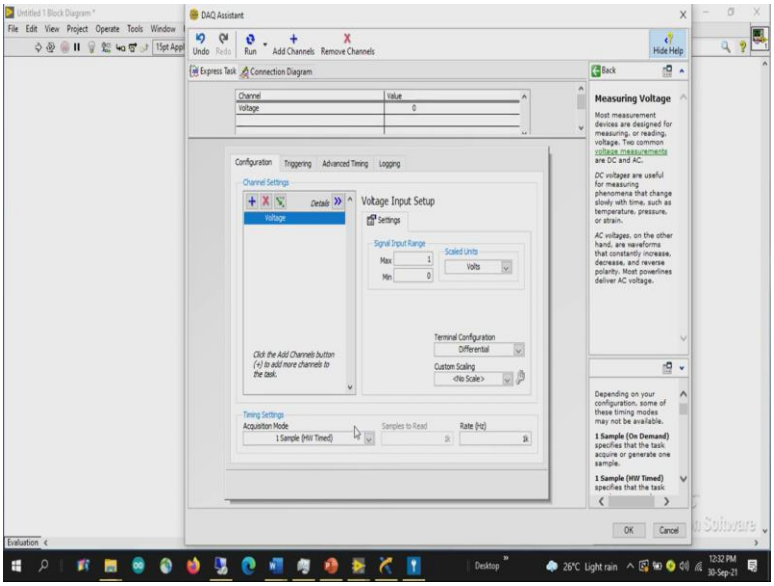
Now remember, we discussed about two different cases when we are acquiring the data, so, one is about reference range. So, the effect of the reference we have already seen. So, the number of levels the minimum voltage, since ADC resolution will decide the number of levels that voltage input reference voltage will be divided and the lowest value threshold for each level will be decided by the reference voltage.

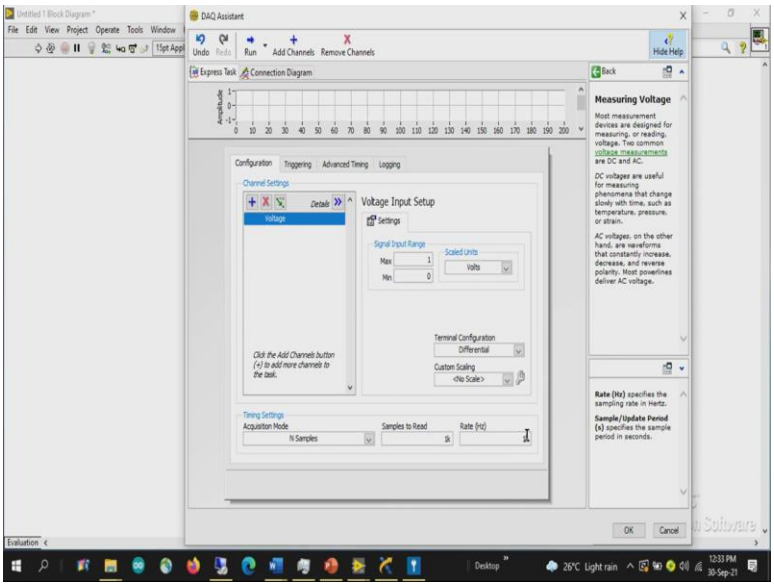
So, since here we are not going more than 1 volt. So, the maximum voltage and setting it to one in voltages and minimum is 0 we do not need a negative voltage here since we are connecting it to with reference to the ground. So, the voltage will be represented in terms of volts only.

(Refer Slide Time: 15:48)









Now, the configuration either you can go with RSE and differential configuration as we have already seen, so, I will show you with a differential configuration now. So, because of which the third terminal of analog the terminal of USB DAQ 6008 is connected to the fourth the terminal four terminal is ground. So, we do not need any custom scale. There are different types of acquisition one sample on demand means whenever you require it acquires the data and displays.

Then hardware time, if you can connect it to a hardware, where based upon that the timing signal receives from the hardware it can acquire n samples, how many number of samples and at what acquisition rate that you have to acquire will be decided samples to read how many number of

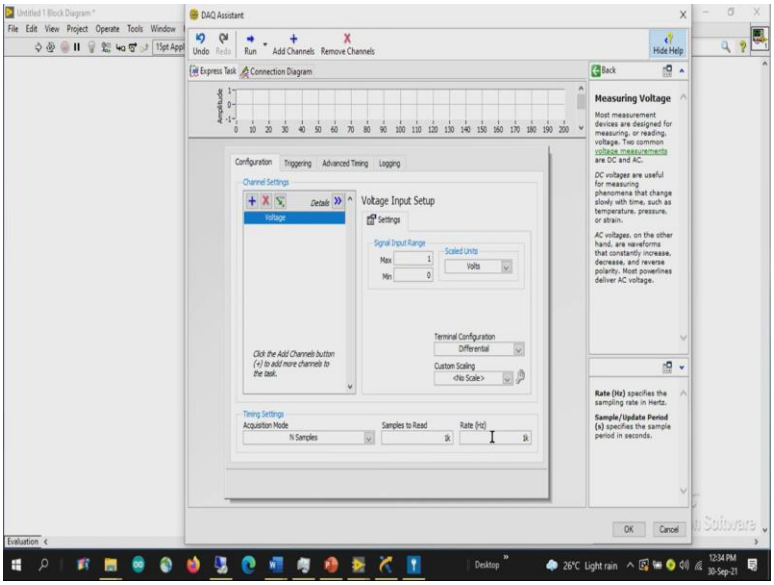
samples. The time to finish the acquisition depends upon how many number of samples that we require to acquire and what it is sampling rate.

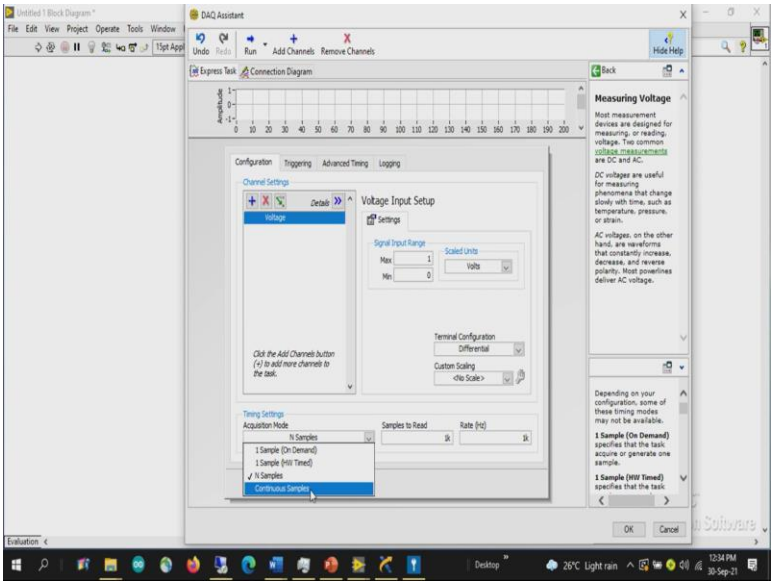
So, if the sampling rate is 1 kilo and the samples to read is 1 kilo it takes one second to acquire all the 1000 samples whereas if the sampling rate is 1 kilo and the sample student is 10 kilo it takes 10 seconds, it is understood because per second this is the rate this is how it will acquire the data. And since we require 10k it requires 10 seconds straightforward. And how does rate depends rate tells you what kind of a signal or what frequency of the signal that we can take we can record.

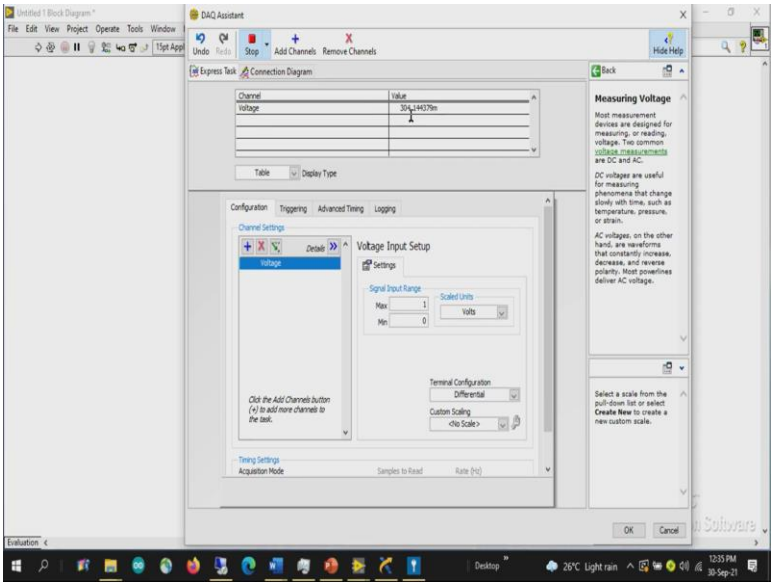
As per the Nyquist criteria, the minimum number of samples required to acquire input frequency signal is the twice that of the frequency of the input signal. But that is only the minimum number of which mean that to understand about it is a frequency very few parameters of the signal that two times the input frequencies. But if you want to extract the more information from the signal, it should be always 5 to 10 times the input frequency.

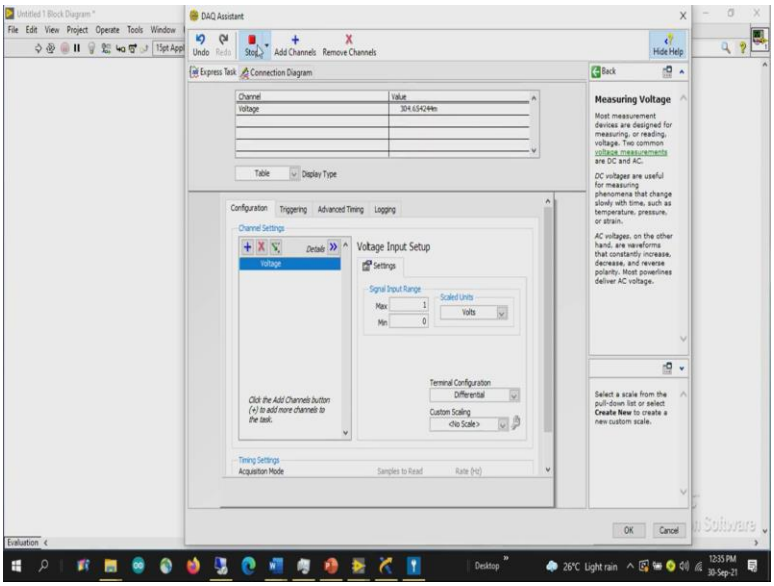
So, in this case, since it is a simple DC you can change with respect to your rate. And in case if you want to remove the noise what we can do, you can acquire at sampling rate of 1k or even 500 or even more than that, read first 100 samples or 200 samples average it out so that you can give a noise. More number of samples you are averaging it out so that the noise the noise will be very less.

(Refer Slide Time: 18:17)







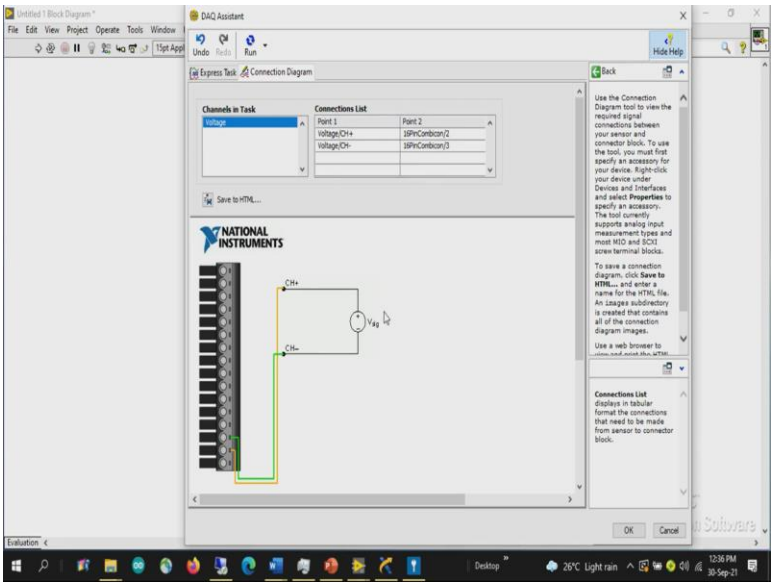


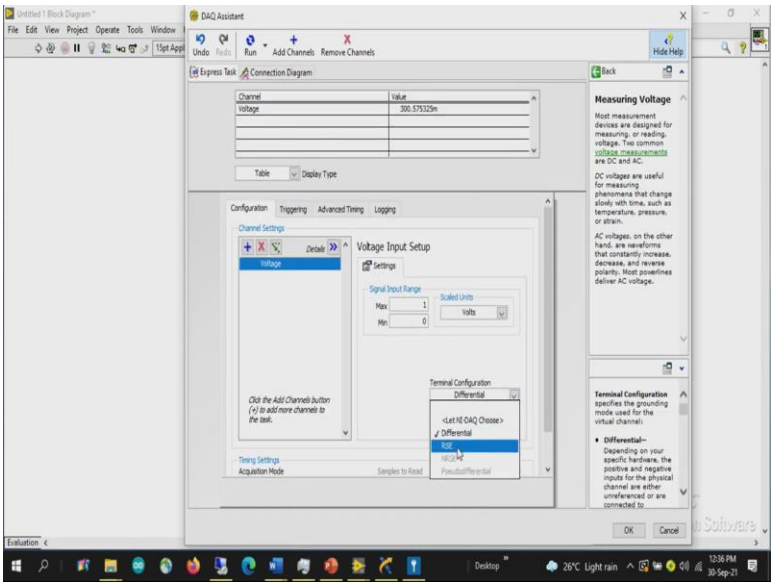
Then continuous sample. So, in this case once it reads these many number of samples the system will stop or the program will stop. Again, if you want to acquire the data you need to acquire. But where is the continuous samples? Once the acquisition is done again it starts the new acquisition that acquisition will never stop until unless you force stop the complete operation or you create a stop button and stop the operation. Continuous samples is nothing but putting n samples in a while loop or a for loop. So, I will show you the difference between all three.

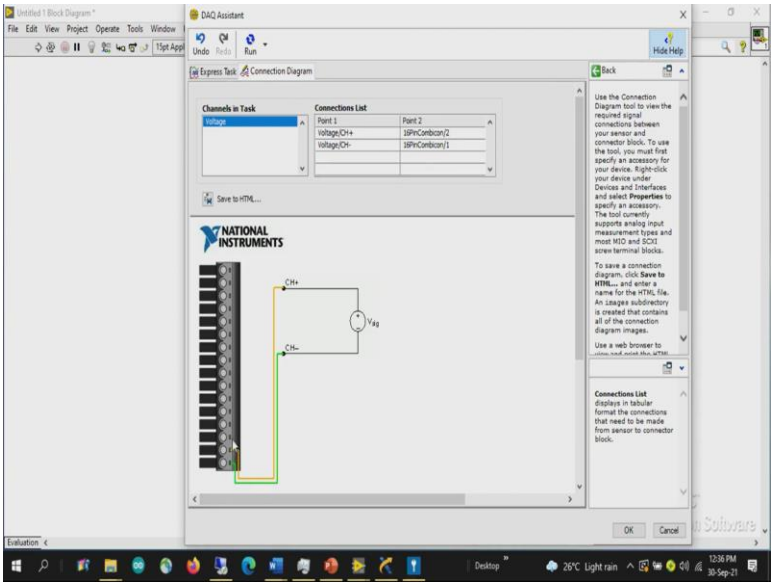
Since I am selecting one samples and one sample on demand and only when the user requires to acquire the data, only then it records the data and displays that instantaneous voltage value. In order to quickly look into that, we can simply run this to see the voltage value. It is at 300 milli

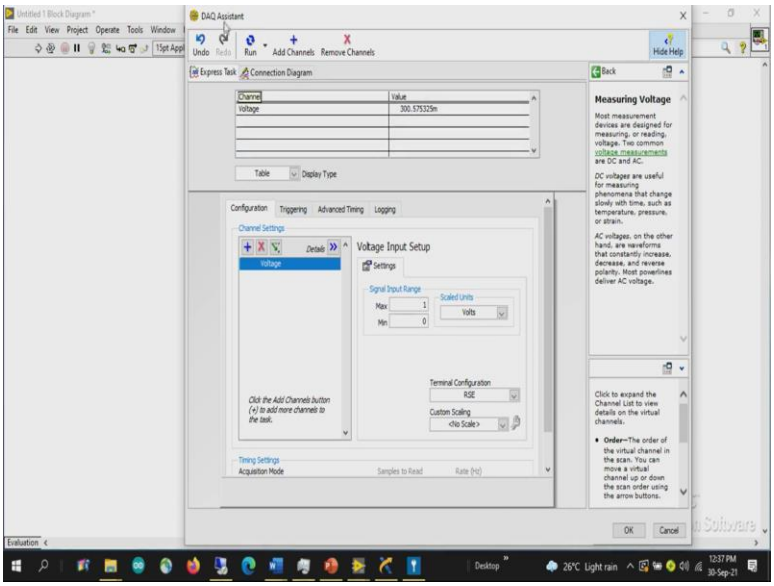
as we know 10 milli volts per degree centigrade, the room temperature is around 30 degrees now. Blowing air onto this it is randomly changing from 1 or 2 degrees as we cannot change more than that. So, that means suppose if I remove the sensor, great, yes, no value. So again I am connecting it back. We can see the temperature roughly around 30 degrees.

(Refer Slide Time: 19:52)







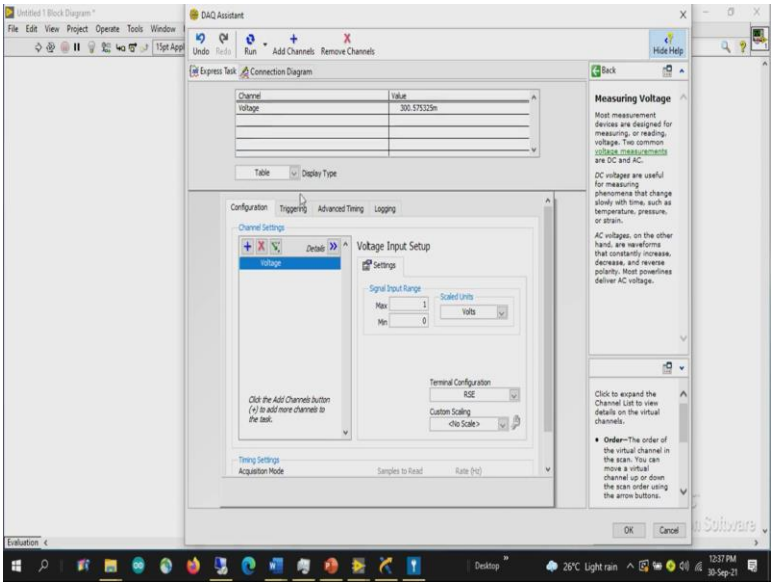


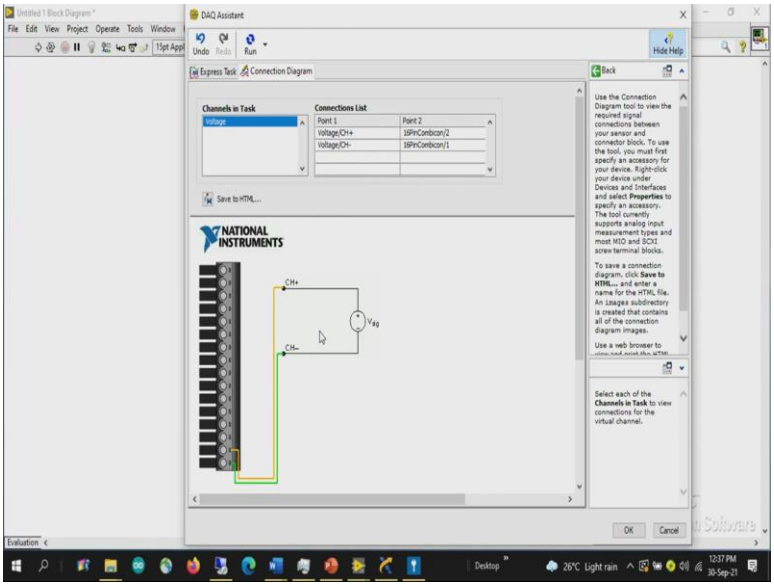
Now, if you want to understand the connection diagram, here you can see as we are configured in the differential mode. So, since it is a differential mode the positive and the negative terminal should be connected between second and the third channel, reversing the terminal that is swapping these terminals will display and in a negative value.

So, this is how. Whereas if the connection is made instead of differential, if the connection is made as RSE. Now you can see that as long as CH- is connected to any of the ground, need not to be connected to the third terminal and need not to connect with any channel. If we connect it right it acquires the data because it references to the ground. So, that is an intention in the previous video, we have made the output from the temperature sensor connected only to the

second channel since we are planning to connect to RSE configuration single ended reference ground.

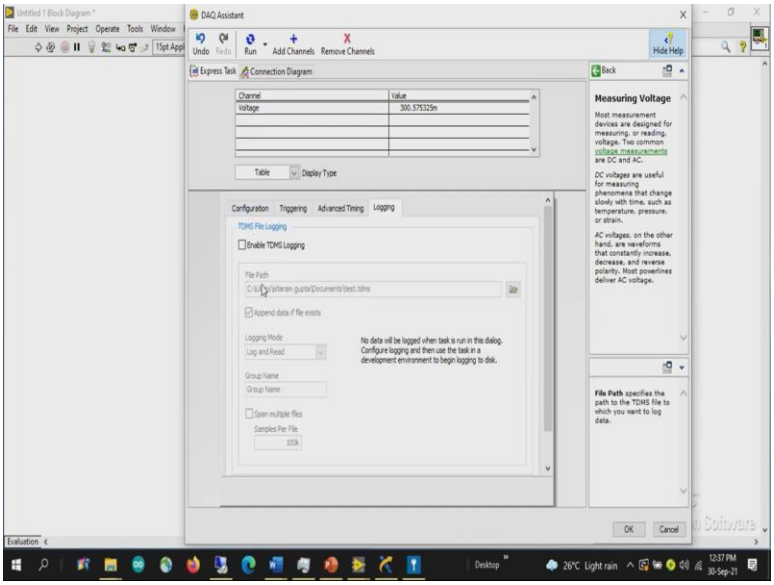
(Refer Slide Time: 20:53)

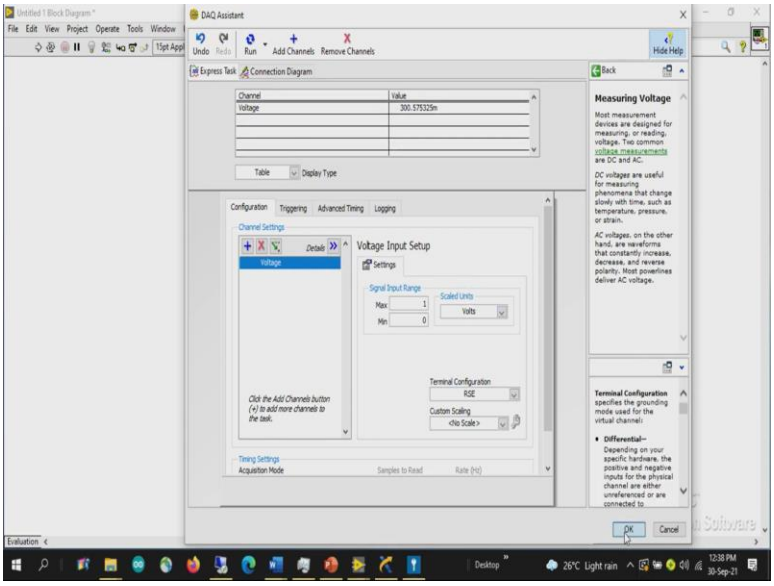


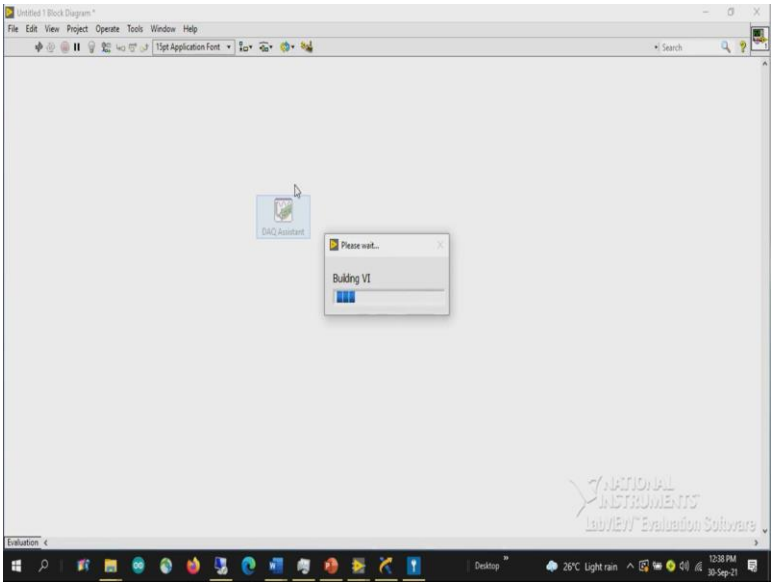


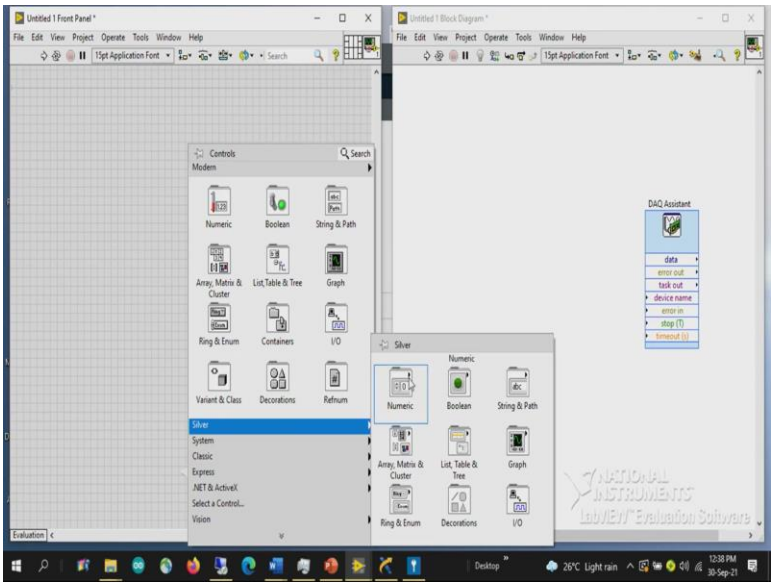
In case if you use a differential mode take a wire and connect between third terminal and the fourth terminal so that the fourth terminal is already grounded so that the third terminal is the negative value is also ground, which means that though the system is capable to measure RSE we are externally connecting the negative terminal of an instrumentation amplifier ground. So, as we discussed internally the gain is gain is fixed where we will not have an option to change more or much about the gain we cannot see any units related to the gain .

(Refer Slide Time: 21:43)





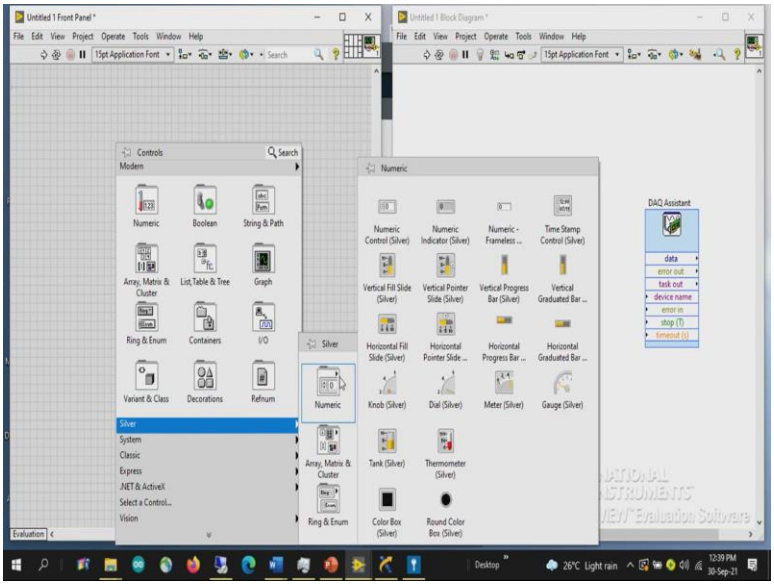


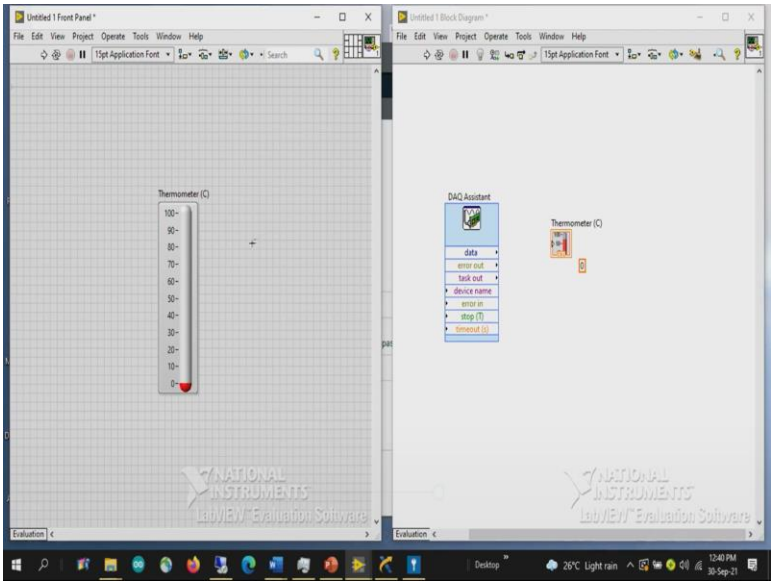


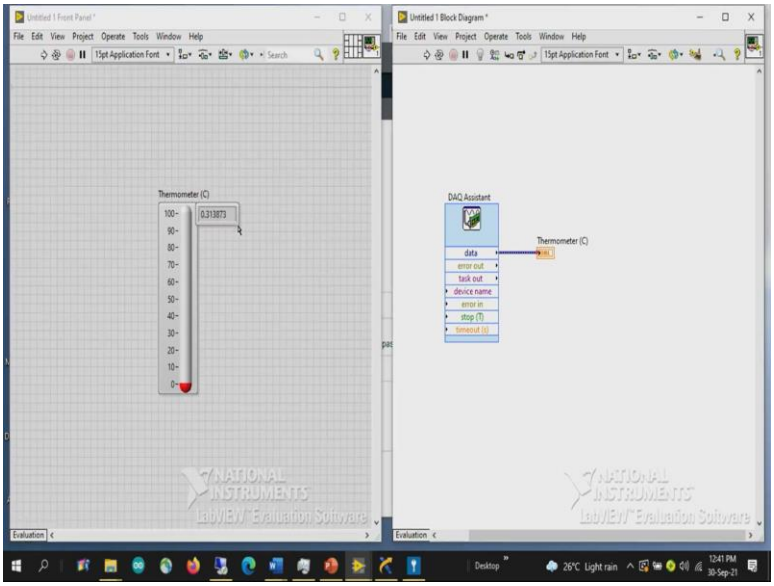
In case if we want to log and read the data while acquiring the data so here we can simply click Enable TDMS logging so that it automatically logs the data. So, once all the necessary parameters are set, press ok so it regenerates the program will build the code according to the user requirements once you do that, here we can see the data output.

So, let me connect this to what I will do I will go to controls palette in order to access the controls palette because user will have you know interaction while the program is running user can only interact with the front panels, front panel windows only. Front panel controls or front panel palettes only.

(Refer Slide Time: 22:49)







So, either you can take any display or the control devices from the modern classic system or silver so since I am interested to go towards the silver, it looks really good. So, I am going to the silver tab and that there is something called numeric tab. So, when you when you click on it or when you place your cursor on the numeric tab, you can see a new window containing different controls and indicators relevant to the numeric.

So, just to display I will go with since it is a temperature I will just go with a thermometer silver since I placed that on to the block diagram. So, it is considered as a constant. So, it should be always placed on the front panel. So, when I place it, when I place the difference between a control indicator and a constant is that the control will have an interaction between the front panel and the block diagram so that on the run any parameter has to be given as an input to or any parameter to be updated while the system is running can be modified using those controls.

Whereas a constant will not have any change during the run. So, since our intention was to display the temperature in terms of degrees centigrade, so I am considering its degrees centigrade thermometer. But as of now, we have not provided the conversion factor yet. So, when I placed the particular control here, double click, you can see the corresponding icon already placed onto the block diagram.

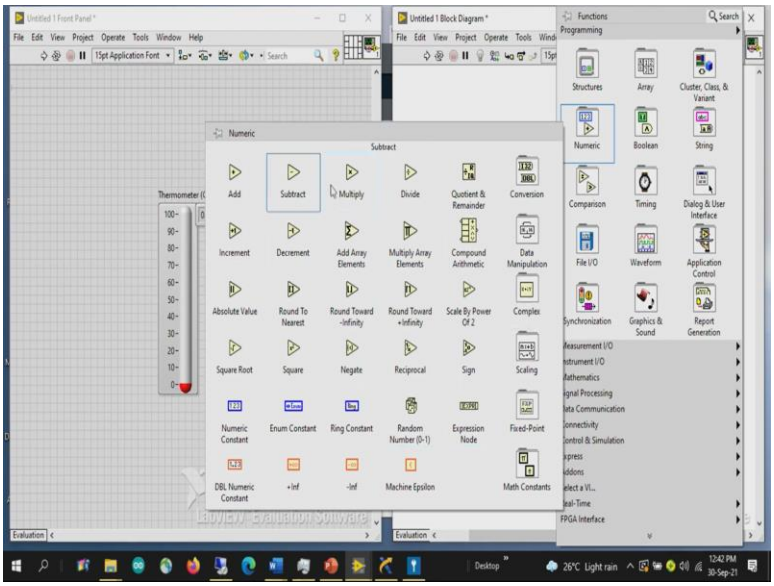
So, I am just viewing it as I can so that the space will not be constrained. What I will do the output will be directly connected to this display. Though it shows that you know in a degree centigrade whatever it is going to display is not in degree it is in millivolts. So along with this, I

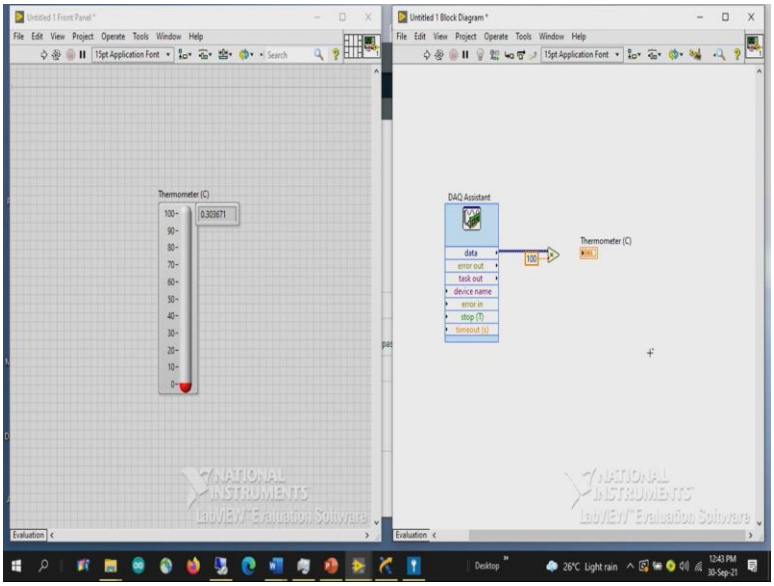
can also make one I can also keep one a digital display to see the exact value that is being displayed on the thermometer.

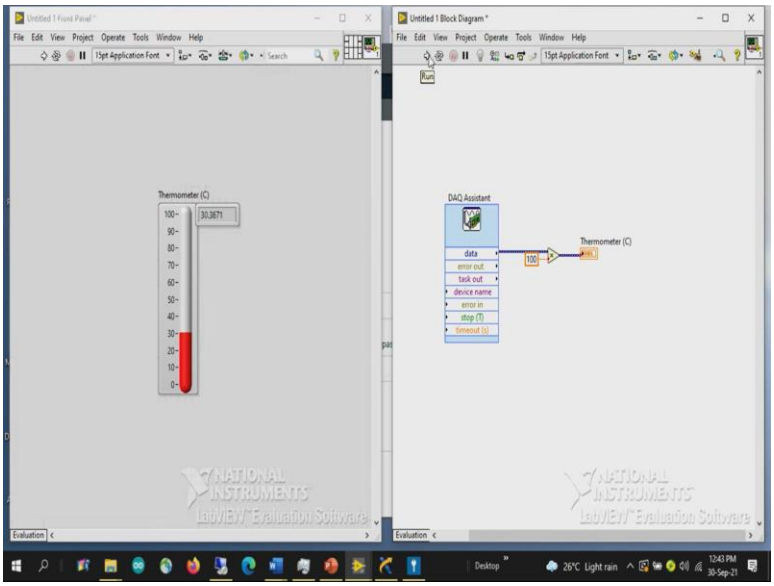
Now, since we have selected as one sample per demand pressing the button you can see the value. So, we have seen 300 millivolts, 300 millivolts which is nothing but 0.3 right. So, now the value is 0.3, continuously can see that variation So, almost it is at 30 degree to 31 degrees. How I am saying it is a 30 degree or 31 degree? As we know the sensitivity factor which is nothing but 10 millivolt per degree centigrade one degree will be always represented with the 10 millivolt.

So, that means 0.01 will be represented with this. 300 millivolt will is nothing but your 30 degrees, to represent this value in terms of a degree centigrade what we can do? Instead of directly connecting to the thermometer or to this particular icon, we use the scaling factor so, we will multiply with a factor of 100 to represent the temperature

(Refer Slide Time: 26:13)

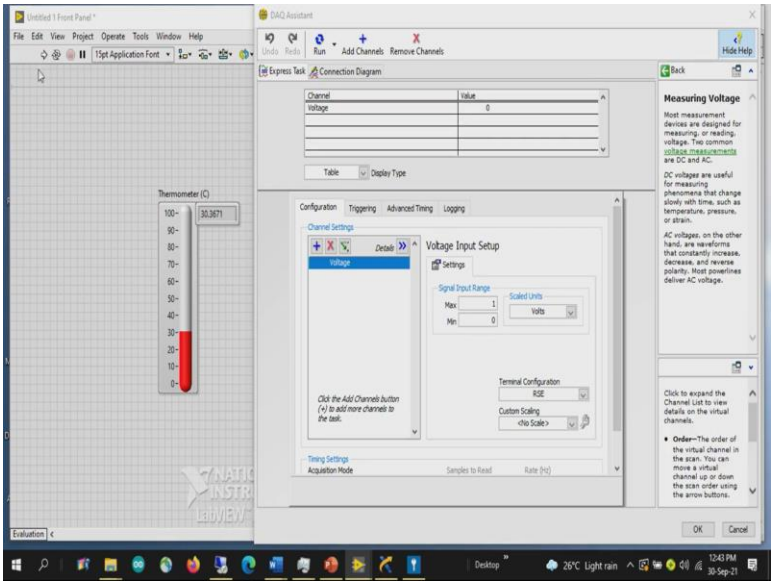


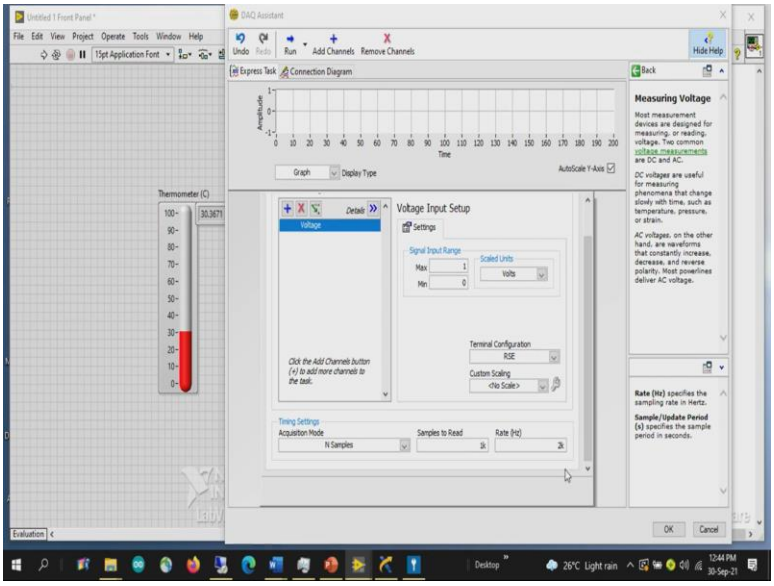


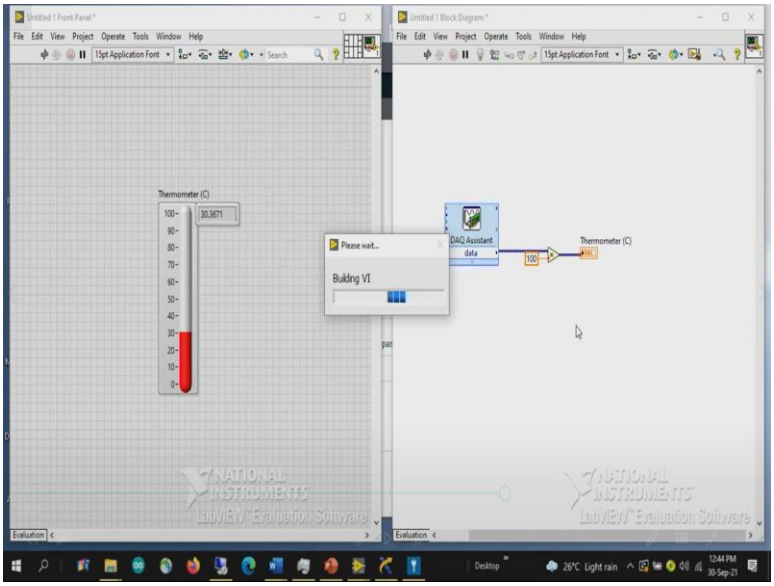


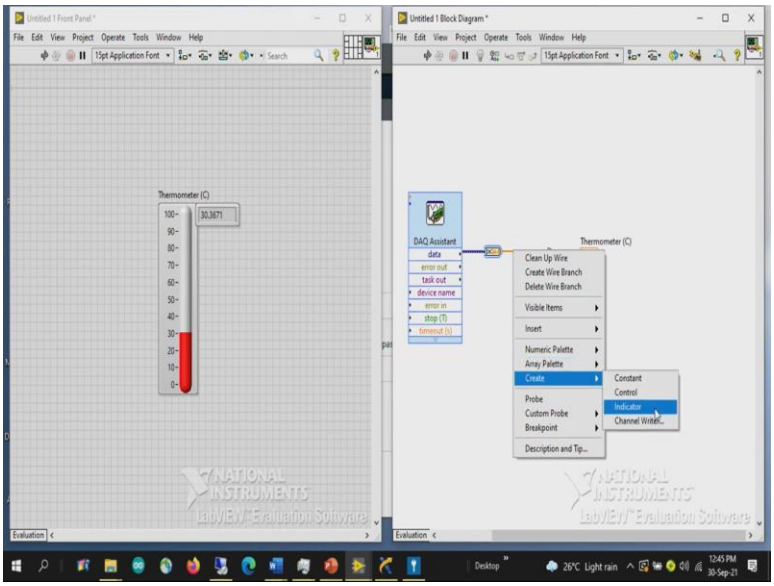
So, in order to do that, since we require to multiply with 100, I will go to the functions palette under the programming you can see a numeric block when you place your cursor on that you can see different numeric blocks minimum functions that is that is required to perform. So, I will be selecting a multiply here whatever the data I get will be multiplied with a factor of 100 why with 100 because 10 millivolt per degree centigrade, 1 degree corresponds to so 10 millivolts is nothing but 1 degrees centigrade 300 millivolts is nothing but 300 into 1 divided by 10 which is nothing but 30 degrees. So, the scaling factor is 100 multiplying with 100 you can represent in terms of temperature. So, here you can see.

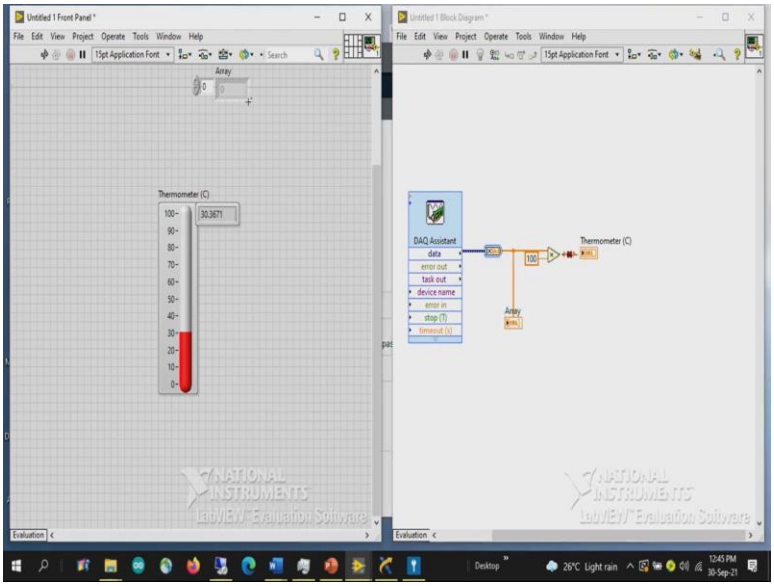
(Refer Slide Time: 27:17)






Untitled 1 Front Panel *
Untitled 1 Block Diagram *
Thermometer (C)
30.3671
Please wait...
Building VI
DAQ Assistant
data
Thermometer (C)
100
NATIONAL INSTRUMENTS
LabVIEW Evaluation Software
Evaluation
Desktop
26°C Light rain
12:44 PM
30-Sep-21


Untitled 1 Front Panel *
Untitled 1 Block Diagram *
Thermometer (C)
30.3671
DAQ Assistant
data
error out
task out
device name
error in
stop (T)
timeout (s)
Thermometer (C)
Clean Up Wire
Create Wire Branch
Delete Wire Branch
Visible Items
Insert
Numeric Palette
Array Palette
Create
Probe
Custom Probe
Breakpoint
Description and Tip...
Constant
Control
Indicator
Channel Writer...
NATIONAL INSTRUMENTS
LabVIEW Evaluation Software
Evaluation
Desktop
26°C Light rain
12:45 PM
30-Sep-21

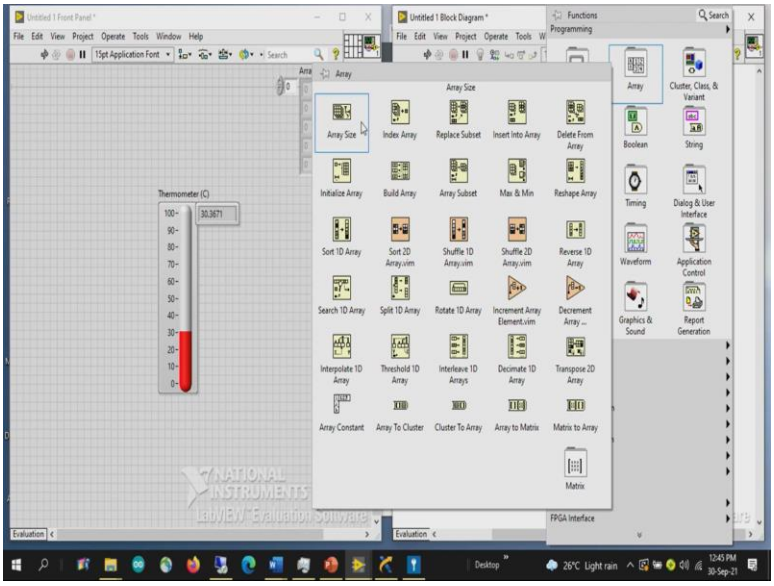


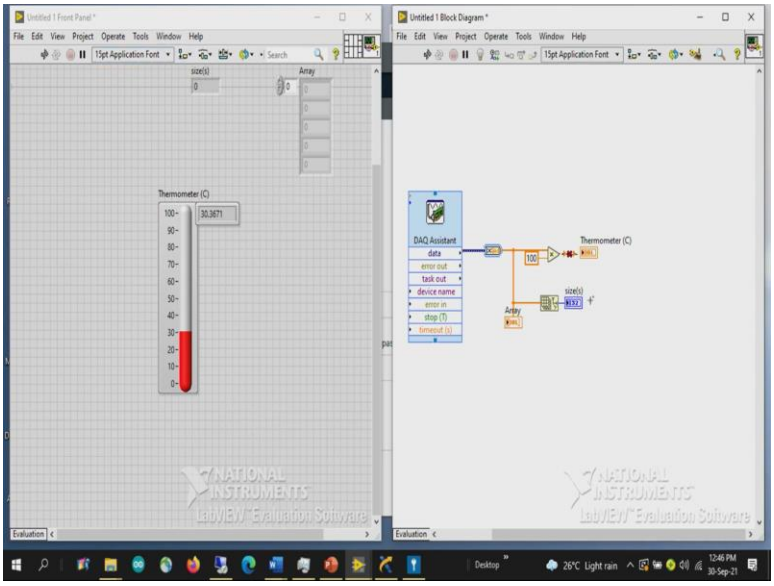
Now, since the DAQ assist was given or made. He selected in a single sample every time in order to update the value, the user has to press a run button. So, this is the Run button. Now, what I do, I instead of changing one sample per demand, I will go with n samples. So 1,000k 1k. Samples to read is 1000 samples and the rate is of 1k. So, changing this value, the updation rate will be faster so 2k 1k samples to read.

So, this is the sampling rate as from the datasheet it can go up to 10,000 samples per second. So, we have certain range for that too. And since it is only 1k samples, since the acquisition per second it takes 20,000 samples, it reads for 0.5 second then it stops.

Now, remember the previous case it is a single element it is reading from the DA cases, but now it is array because it contains more than one sample. So, in order to average the value whatever the values that you get what I do let me convert the dynamic data to array from dynamic so here you can see if you want to display all the values right click the easiest way to display the things is this right click on the wire you can see a pop up go to create then you select the appropriate window appropriate option that you are looking for right. So this is the array as you have already seen. So, all 500 samples or 1k samples can be seen here.

(Refer Slide Time: 29:30)



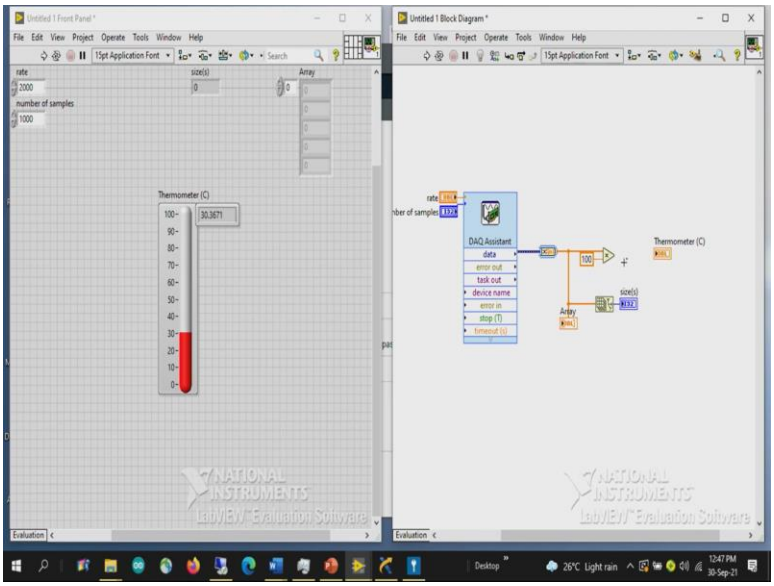


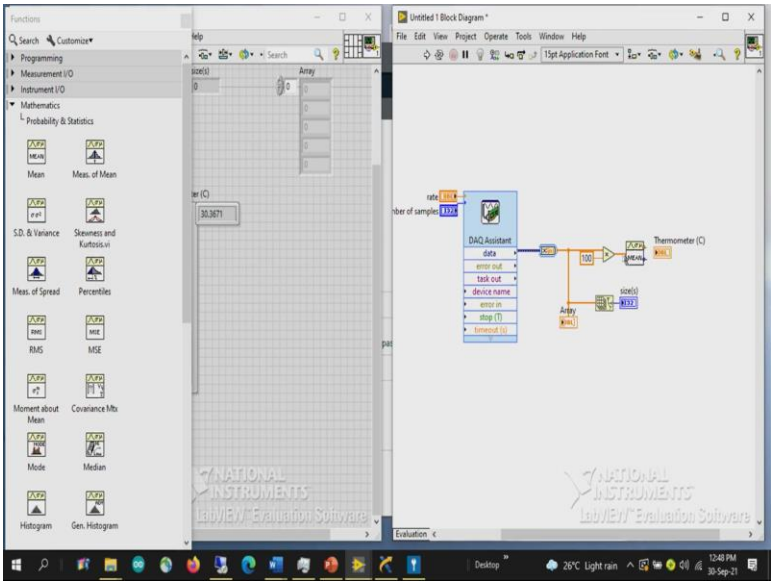
If you want to understand how many number of sample has been acquired. In the functions palette under programming, under the array sub palette you can see the array size, there is a function called array size. Click on this array size and select it here so that we can see what the size of the array is. Since we need to display the array size create indicator, so this gives you the array size.

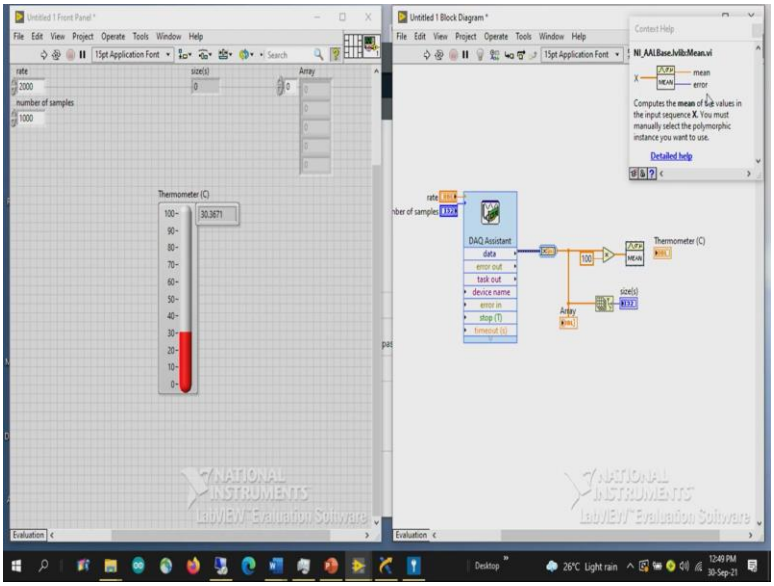
So, since we are we have asked to acquire 1000 samples it will display. In case if you want to modify on go while the program is running in case if you want to modify the sampling rate as well as there are a number of samples to read, right clicking on to the appropriate terminals of

the DAQ will provide a flexibility to control these parameters when the program is running. So, this is sampling rate here we can see when I click on the corresponding block you can see where it has been placed on the front panel or vice versa.

(Refer Slide Time: 30:46)







Now, there is an error, the reason is that this multiplication or addition or any kind of a program or subtraction, what it does in the sense it is a polymorphic. So, which means that single element since it is having 100 elements or 1000 elements, each element will be multiplied with 100 and produces another array and the array dimensions will also be the same. So, because of which since that thermometer is in say a digital display or it can only display single element, whereas the output from this function is an array it cannot display all the values using this.

Either you have to use an array indicator as we have seen here or we can convert this array values into single values since our intention of taking arrays and array data is to average it out so, that the noise can be minimized. In order to the average there is a function called mean. So, if you know the name you can directly search here or in case if you want to find out where it is it is available under the mathematics, probability and statistics you have a sub VA called mean, select drag and drop that place.

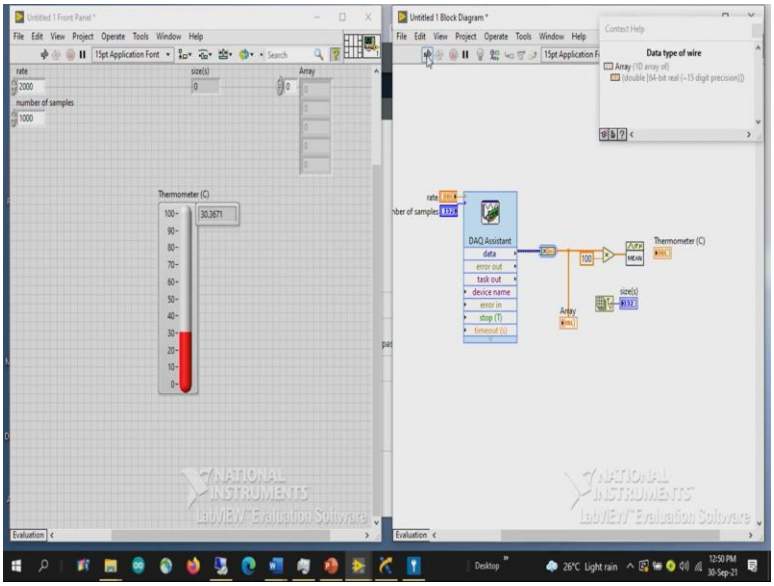
So, if you want to understand how many number of inputs and how many number of outputs this VA has press Ctrl H, Ctrl H which is a help window. Here you can see the input to the system is always x and the output to the system is mean and error. So, whichever is represented on the left side is always considered as an input represented on the right side is mean is the output. So, the LabVIEW thing you have to remember the data flows always from left side to the right side no business of flowing from the right side to the left side provided if the feedback loop is created.

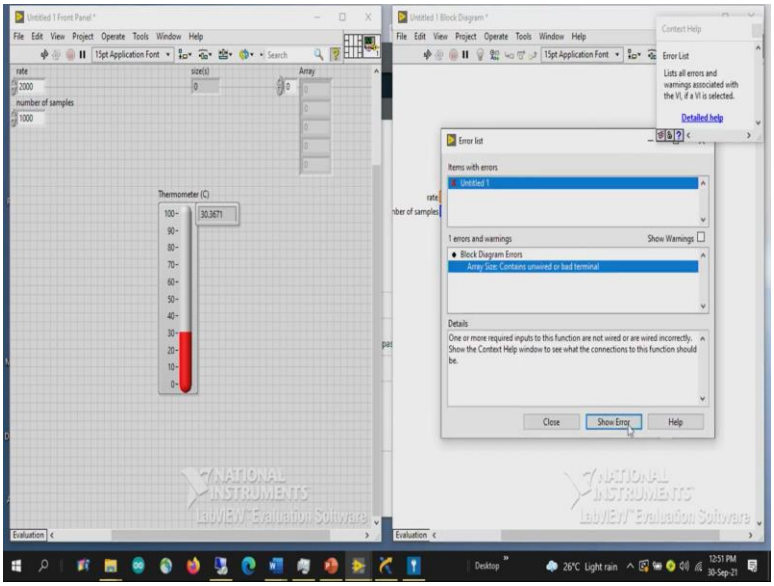
So, here even if you consider a few things you have to remember that when you consider few different blocks suppose say if I consider array size, in case of array size the left is represented in a bold and the right side one is represented with a simple way. Whereas, even if you go with a mean the left-hand side is represented with a normal a text without any bold or anything even right side is also represented, what does it mean?

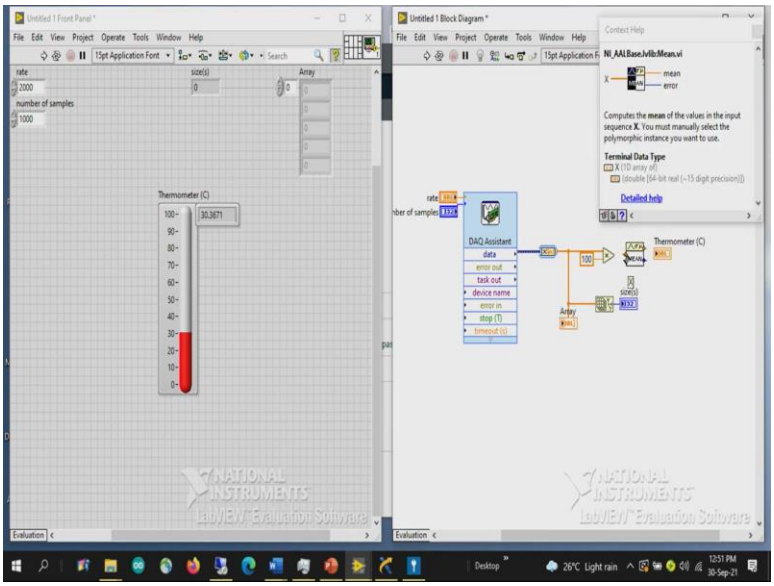
It means that the bold, informs the user that this particular terminal has to be conducted compulsory, it is required, without this input, this particular function cannot execute. Whereas the right side whether you look at the value or whether you do not look, it automatically acquires the data and performs the function displays. Whether to see or whether not to see, it's up to you.

That is why it is recommended, but it is not. It is the users wish to show it or whether or not to show it. But even in this case, the same in order to do the operation, it performs only when you connect the input if you do not connect it will not throw any error. But whereas if you consider array size, if you do not connect it to an input, it always throws an error because it is a required terminal.

(Refer Slide Time: 34:03)







Here we can see any error in the program will be represented with a broken arrow which means that a user will not have the ability to or user will not have access to run the program. So, you can list other terminals too. But whereas this is not the case, when you remove the connections between this, see you can run the program because the input is recommended. Since it is in normal font, it is not required. But whereas there is another way of representing the terminal, which is nothing but Italics, Italics indicates that it is it is optional. Even if you cannot or if even if you do not connect it will not hamper the programming, it will not change the results at all.

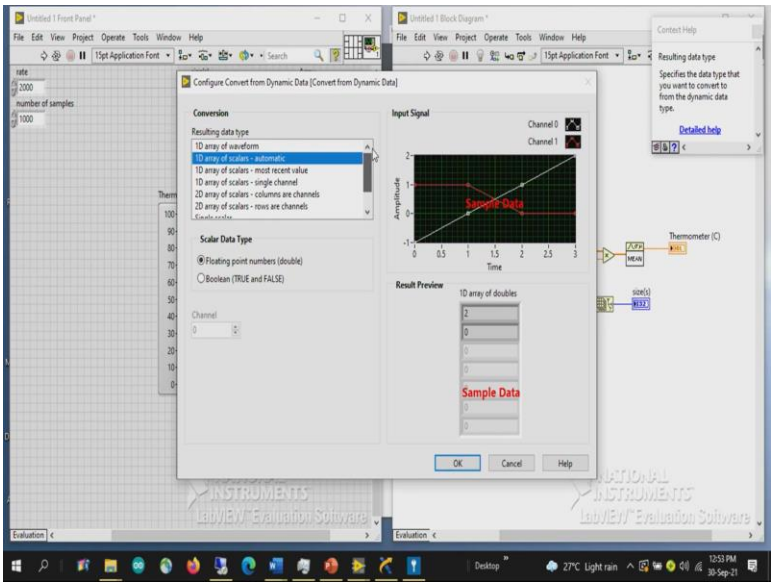
Now I will be connecting it since our idea was to check only for the mean I will take the mean and now what now if you look into the wire dimension, the thickness of the wire that decides

whether it is an array or whether it is a single element. Since the output the mean value is nothing but the average of all the inputs that you have connected, array of the inputs of connected. So, it gives us a single output single value.

So, that is why comparing the thickness of the input and output of this VI blocks sub VI block, you can understand whether it is an array or whether it is a single element. Even, the color always represents the data type here everything is in orange color, whereas here it is in blue color, whereas here it is in data wire like a thick wire with the helix kind of thing.

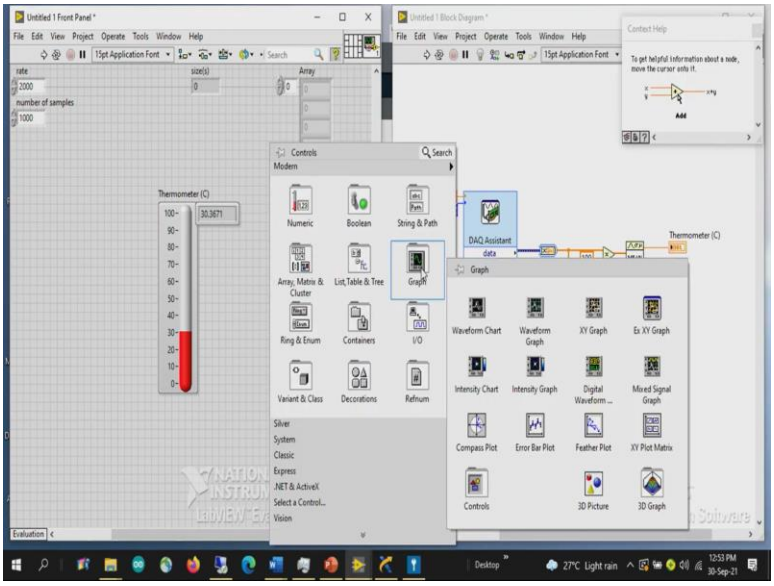
So, what this indicates in the sense, Orange is nothing but your D bill or a floating type data type. Blue indicates it is an integer data type. Since the thickness is in in form of very thin, which means that this is also single element is also single element. So, this particular conversion converts a floating type data to an integer type data. Whereas this indicates Express type, which means that along with the array, it also contains other information, what other information?

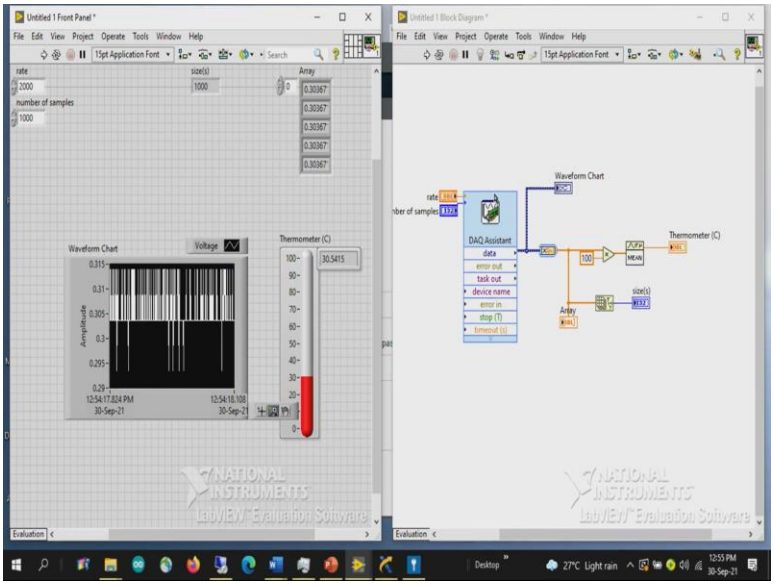
(Refer Slide Time: 36:19)

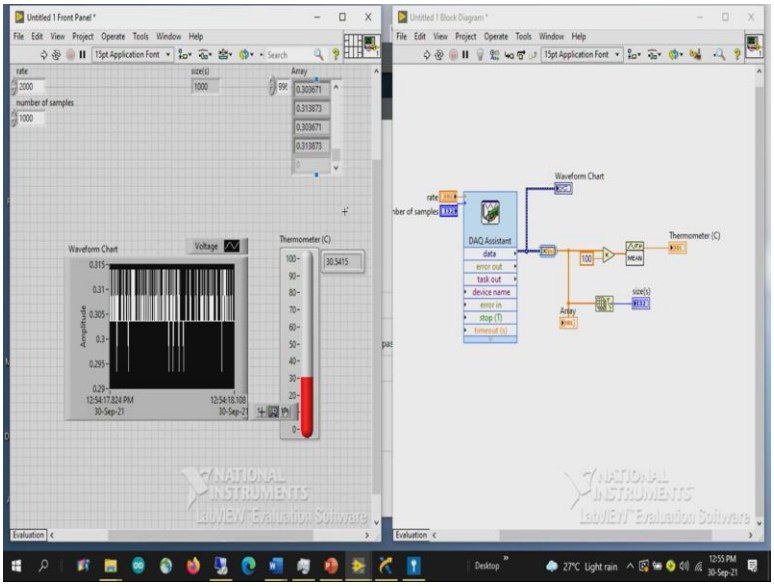


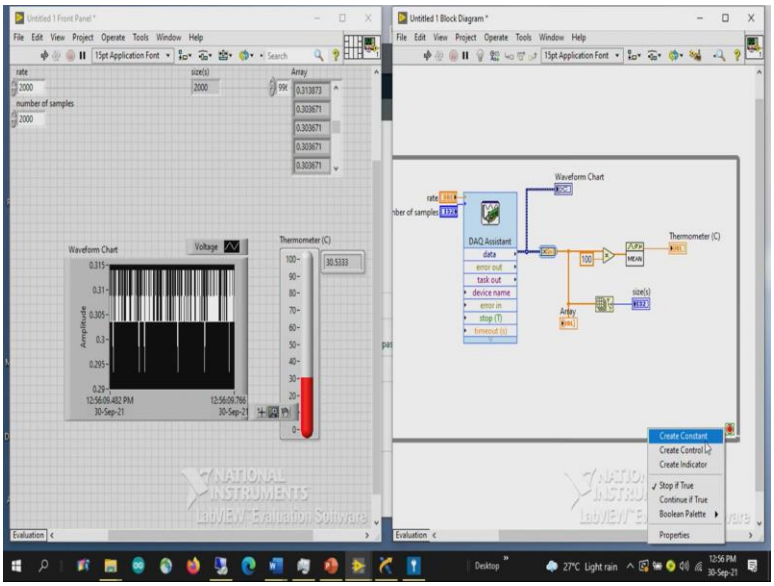
So since it is a dynamic data, it depends upon what data type that you want to use, by using this convert from dynamic data, it produces those, in case if you want a 1D array of waveform, where it will also have the information about your time sampling rate. So, you can even select that. Since our intention was only about scalars where I am not dependent on the time. So, I am just simply selecting D array of scalars. So, here you also have a single scalar or a single waveform. So, depends upon what how you want to represent the signal, you can select that and pressing it.

(Refer Slide Time: 36:57)









Now, the same thing even can be represented using a graph, waveform chart or waveform graph. So, I am taking a waveform chart and placing it here. Now, the corresponding acquisition signal will also be displayed after averaging it out using this display. At the same time, the intermediate values before averaging, how many number of samples that we have and it states will be displayed even in the chart.

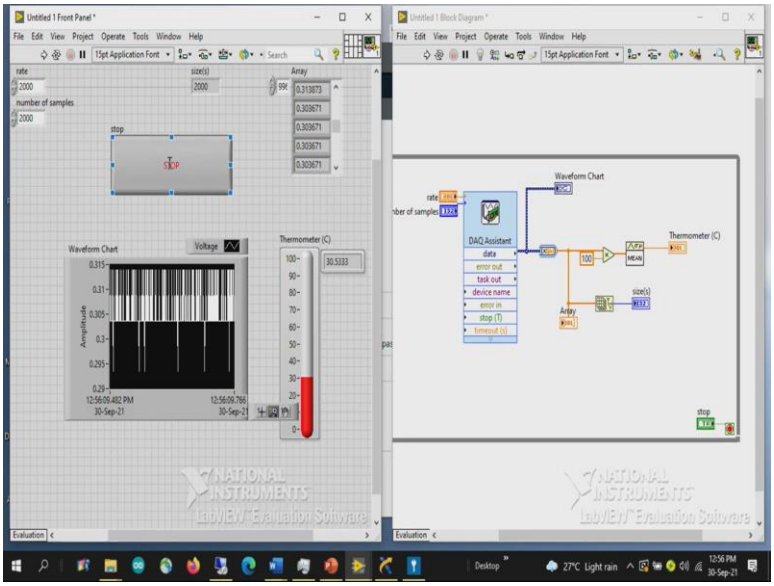
So, running this program, we can clearly see the number of the signals. So, since it is chart, which will also have the history of the previous information, and if we want to expand it further, we can go with different styles or interpolation type. So, I am creating some scale legend and graph palette. In the graph palette, I will expand these values.

So, here you can see the fluctuations between that so, the base value is somewhere around 0.3 to 0.305. So, some samples are going higher than 0.305, it is in millivolt because unit you have to define it since output since the output from the DAQ will be always in the volts. So, whatever the signal that we acquired in terms of 300 millivolts. So, this is amplitude in terms of volts it maximum goes to 0.315 or minimum is 0.295. So, the average signal is 30.5 that means the temperature of the room the current room where we are recording is having around a temperature of 30.5 degrees centigrade.

Now, after it acquires, so here you can see the sample size which is 1000 and all this corresponding samples can be visualized here. So, go to the right click and vertical scroll bar you can see all the 1000 samples available here. Since we have mentioned the number of samples to be acquired as 1000 and the rate at which they acquire is 2000. So, once it acquires these many number of samples say suppose say if I represent with the 2000, now it takes two seconds to acquire right all the 2000 samples has been updated and you can see the value here.

Since it is not in a continuous acquisition, it is mentioned in simple acquisition mode. This is how. Once it acquires n sample mode once it acquires this n samples, it stops execution. In case if you want to acquire continuously, you can go with a simple while loop, place a while loop here and create a control so that user will have an access when to stop it.

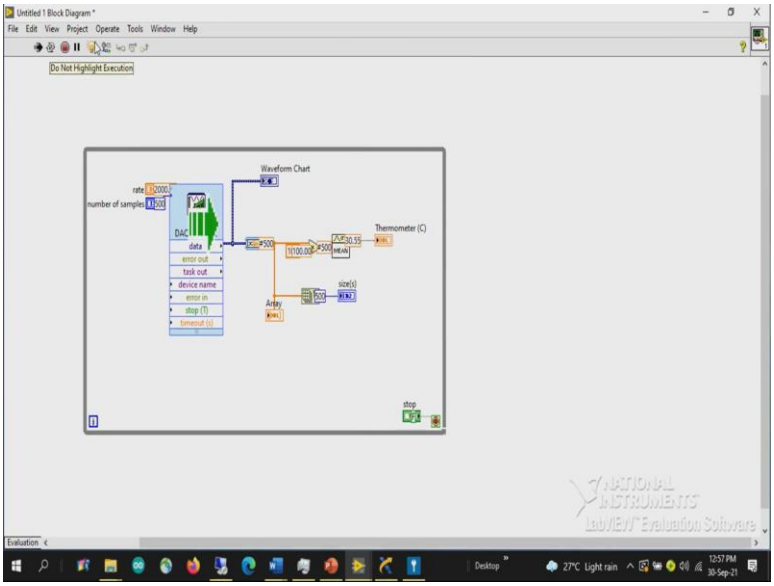
(Refer Slide Time: 39:52)



So, once you stop only then it will stop the execution as long as you do not want to stop it. It will keep on executing. So, just running a button it will acquire 2000 samples at a rate of 2000 rate at a rate of 2000 and it averages all the data; then it displays it here. So, the updation rate will be 2 seconds because the number of samples required is 2000 and the sampling rate is 1000.

In case if you require only 500 samples to acquire, it will acquire 500 samples among the first 2000, then it displays. So, you can see the size the rate at which it is updating is also higher now. So, it depends upon the user, it depends upon the application, this rate and this number of samples has to be decided.

(Refer Slide Time: 40:42)

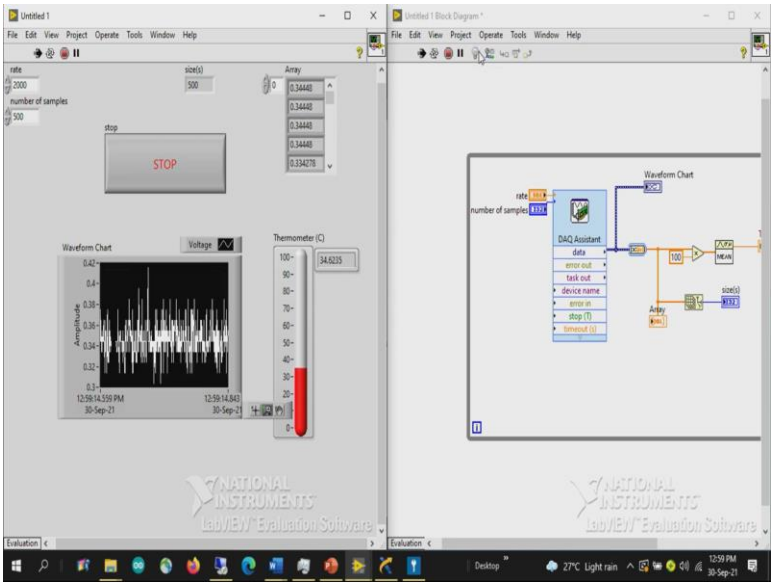


So, in case if you want to understand how the program is working, so, there are different debugging techniques available in the software, there is something called highlight execution, clicking on that, it slows down the execution process of the program. So, in the previous case, the execution rate entirely depends upon the processor. Now, the execution rate will be 1 icon per second. So, every one second it acquires the data, then after acquiring the data, it gives us an input to the next data and executes that particular sub VI.

So now, it is not executed, once the data is received and executed then it went to the mean and again executed and displayed it here. This is called Data Flow programming, so until and unless the data is available to all the terminals of the icon or the sub VIs is that we have placed on the block diagram window that particular sub VI will not will not execute. All the required inputs

have to be available for acquisition or for execution. This is the simplest way of representing your signal.

(Refer Slide Time: 41:54)



Now you may ask your question since we require these many number of samples the process will also take? Yes, of course there will be a delay, that delay is the debugging delay. It is not the execution delay, it is due to debugging, debugging the process of execution has slowed down. But in reality, it is really very fast. If any changes it will be recorded here.

So, I placed my finger on the thermometer. So, is representing the temperature of 37, removing it again it is coming back to the room temperature, again I am placing 37 degree to 40 degree. So, this is the simplest way of acquiring the signal using a data acquisition device from a temperature sensor unit. I hope it is clear to everyone in case if you have any doubts, please feel free to get in contact with us via forum. We will be happy to provide you the support. So, thank you very much.