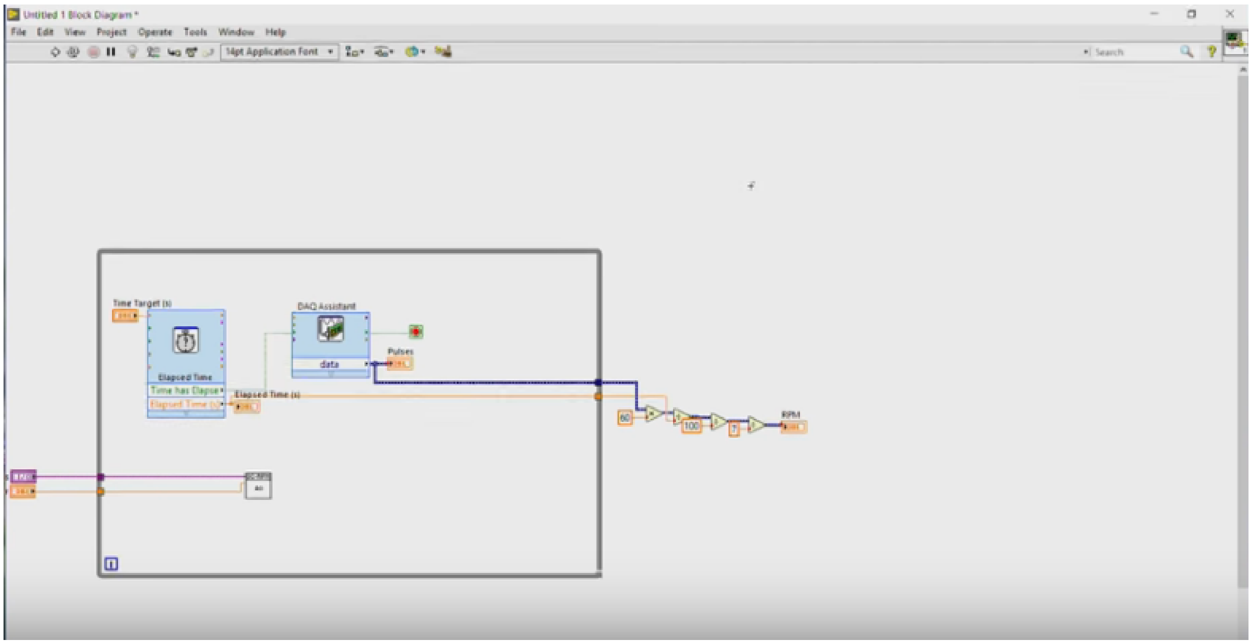
Lecture – 32
Design of Speed Control of a DC Motor using DAQ: Part 2

Refer slide time :(0:27)



So, let write a algorithm now.

Video start the time :(0:31)

What we have to see? So first thing, what I will be doing is a, I will be taking a graph. So this will show as, the input as well as the output. Also, we have to take, we have to provide an input to input, which is nothing but a set point. So for that, what I will be doing is, I will take horizontal slide, with a pointer or we can also take a vertical slide with pointer. So this vertical slide with pointer indicates the RPM, the set RPM by the user. So in the previous case, we have done the same thing, so this is RPM, Set Point or RPM, in terms of RPM. So in the previous case, we have set it in terms of voltages. But now, since it is a digital, we can consider, whatever the RPM that we have. And we also know that the maximum RPM the DC motor rotate up to 155, so what I'll do is that, I will change it to somewhere around 200 RPM. And, so since it is difficult to visualize to use it this way, I will also create a digital display, so any change in the set point, you can easily see it here. Right? The value can be easily visualized here. Okay? Now, other thing, at the same time, we also have to see, what is an actual RPM, the motor is rotating with. So, what I will be doing is that, I have to copy the same thing, but I'll have to change it as an indicator. So right click, change to indicator, now this will be actual RPM, so this is our actual RPM and this is our set point. But, the idea is that, we have to control it. So for that, what I'll be doing is that, so these are the things which, when we are, you know, placing the icons, when we are creating a control as well as indicators in the front panel, the appropriate, you know, icons has been created, even in log diagram. So this we can already seen here. So this is our actual RPM, this is the set point and this a Waveform Graph, but everything should be in our controlling simulation.

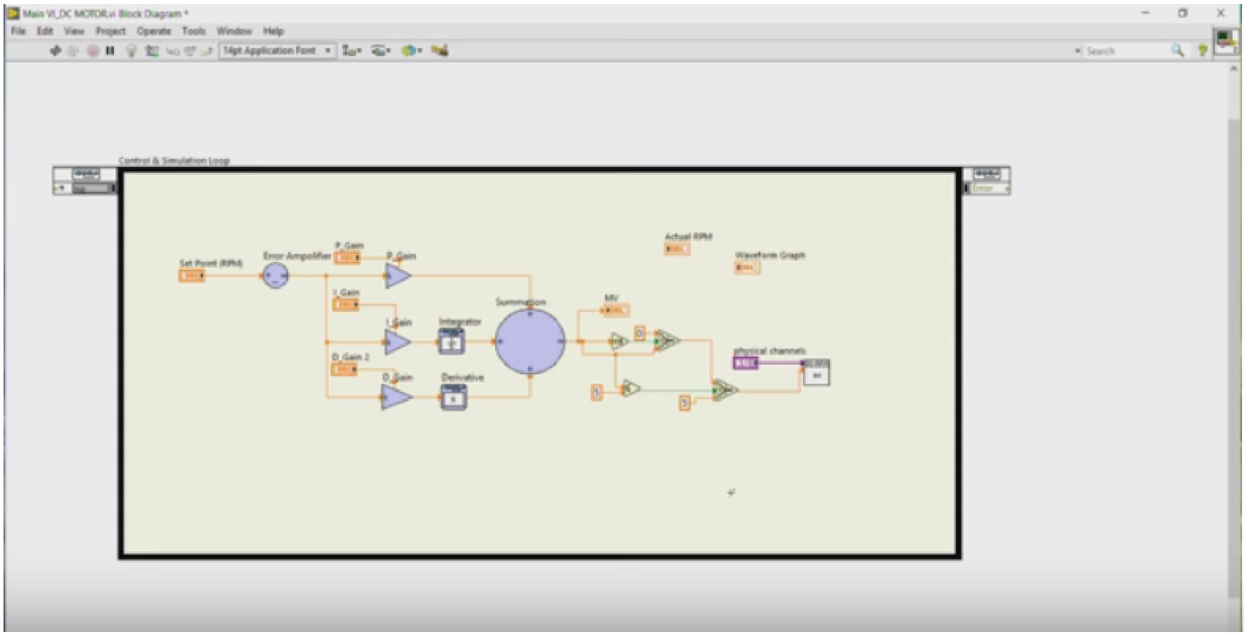
So what I will be doing is that, I will go to the Control and Simulation in the Block Diagram, I will take, the Control and Simulation loop. Right? So this is our set point, so after set point, what we have to do? We have to create, so I will drag, every, each and every point. So, one we need, a signal arithmetic, for summation, which access an error, in our case. Whereas the gain, both, P Gain, I Gain, as well as D Gain. And we need all three to be added, so for that again we need a summation. So all three blocks I've taken it. So what I'll be doing is that, the output, the set point, now I'll be connecting it to the first terminal of summation. Let me rename, which will look similar to the top of our terminology used in control systems. So this I will replace with error, error amplifier, then the output will be connected to gain, but since, the gain has to be externally R user, has to change during the operation. So what I will be doing is that, I will provide a control for this. So inorder to that, double click on it and here I will be changing it to a terminal, press [Okay]. Right? So here, everything now is terminal base. So this I will take one more, again, I will take one more here. So this is I will say, P Gain, P underscore Gain and whereas this gain is I Gain and this is D Gain. But incase of I Gain and D Gain, we have to, connect integrated as well as, differentiator, so what I will be doing is that, again I will go to the Control and Simulation tool box, here we have different blocks. One is Integrator Block, other one is a Differentiator Block. So if I pass the signal directly, to the summation block, this case, so I will activate this, as well as this one. Okay, let me zoom it. So this is summation, but basically this is PID summation. If this is directly connected it here, that means, it's, it's a P. Suppose, the output of our Amplifier, is passed through I Gain, P Gain has zero and then output is passing through, 1 by S, which is nothing but, integrated action, then if it is connected to summation, this is called, 'I Gain'. Whereas, let me make the terminals, connections, this is a D.

So whatever the output that we get from the summation block, is nothing but, error amp, manipulated variable. Right? So, but we have not set any gains, which can be set from outside. So we have just chosen a property as a terminal. So we will be creating a terminal from the front panel. So what I will be doing is, I will go to Silver, there, here we have different slide bars. So how, now this time I will take a horizontal slide bars and the maximum, I will go with 1. So we also need, so this I'm renaming it to, P Gain. And to visualize the exact value, I'll create a digital display for this. Either we can set from the digital display or we can, we can understand, what is the gain which has been set there. So say, time being I will go with 0.05 and the maximum is of .1, right now I'm going with. Then I also need, I Gain, so I'm just driving and dropping it. Also, D Gain, so this I name it as, I Gain and this I name it as, D Gain. So, what I will be doing is that, this particular block, the icons have been created in the, Block Diagram, so I'll connect the P Gain with the P Gain, I Gain with I Gain and P Gain with that particular block. So one thing it is clear that, if, if the I Gain as well as D Gain values are zero, so, 'Let me make it as zero'. That means, the controller which we have set is only P. If P zero I has some value and D has some value, which means, it is IT controller. If D is also zero, only I has some value and P is also zero, it means its only Integrator. So depends upon, what type of controller that we want to realize, we set those gains or we make the gains as zero or we can set to some particular value and we can understand the working of it. So whatever the output that we get, is nothing but, the manipulative variable. So inorder to understand that, what I will do is that, I'll create one more block at the output side, which is nothing but the manipulative variable. So what I will do, I'll create a numeric output indicator, so renaming it as, 'Manipulated Variable (MV)', so let me connect is here. But, we have done, we have realized, Error Amplifier, we have realized, a Controller, P as well as, as well as D and we have done a summation. Similar to that, what we have seen in our, Board 2? We have done, we have implemented design and implemented, using operational amplifiers of, error amplifier, as well as P controller, as well as, PI controller, P plus I, as well as PID, which has both P, I and D. And so which means that, it uses a summer, which is OpAmp based summer and it adds all the output that we get and gives us an output. So idea is that, the output of this, particular block, which is nothing, to manipulate a variable, has to be given as an input to our, DC motor, driver. So, since we have already created a VI for that, let me call that VI, so analog output DC, so let me save this first, Main VI, VI for DC Motor, Speed Control. So the physical channel we have to connect it.

So let me create; Control, so A00 (A zero, zero), so these are physical channel, the output whatever we're getting, has to be passed through this. But, but we know that, we know that, the operating voltage of our motor is only 0 to 5 volt, 0 to 5 volt. So, inorder to, not to provide, more than 5 volts, as, and, not only that, our data questions device can only provide a maximum output voltage of 5. So inorder to restrict those things, what I will be doing is that, I'll create a smaller logic here, which says, that is the in, the output voltage, from the summation block, is higher than 5 volts, then, the, the, the value that has to be passed through the analogue output channel, is only 4. If it is less than, 5 volts and if it is greater than, 0 volts or 0 urens, that means the value has to be directly passed. If it is again less than 0, again it has to pass only 0. At, so, now to implement that logic, what I will be doing is, I will take a select block and I also require, comparative, caparison, we have to see, whether, whether the manipulated variable is less than 0, if it is less than 0, 'Yes' then, the output should be 'True'.

So create constant, if it is less than 0, then, the output has to be passed, output to be passed is of 0. If it is 'False', then, the same manipulated variable can be passed. Okay? And other logic is that, we have to observe the manipulated, whether it is greater than 0 or not. So for that, again I will take one more select and I'll take greater than operations, if this value is greater or equal to 5, if that is 'True', then output should be only 5. Or, so to make our connections easy, I just replaced this with less than or equal to, less than or equal to, if the value is less than or equal to 5, then, pass the manipulated variable directly, if not only pass 5. But, the idea is that, we can only provide an input, only single input to the system. So, this logic, we can clearly understand that, okay, to understand, whether the logic is working or not, I will copy, I'll create a new VI and I will paste it here and I'll create a control, numeric control, connecting it here and we can see, what is the value it is being sent. So let me run continuously. So 0, okay, minus 1, still 0. So 1, it is 1, 4, it is 4, 5, 5. What if it is greater than 5? So, 6. 4, or if it is not working 10, so still it is passing this. So, when it is went? Where it is going wrong? So, if this is greater than 0, I mean, if it is less than 0, which it is passing 0, here, this should be, 5. Now, reason once again, right. Suppose if it is 0, 0, 1, if the value is 1, 1, if the value is minus 5, 0, if the value is 10, 5, if the value is 4, 4. So that means, 0 to 5, it will pump, it will give the, the value. If it is greater than 5, even then it will, it will set the, input of DC Motor to only to 5 volts. If it is lesser than 0, the motor, the input to the motor will be 0 volts. Which means the motor will not run. So that means, the logic whatever we implemented, is working correctly, only thing is, this has to be replaced with 5. Then the output of this, the out of this, has to be connected to, this block. So providing an output to plant, till that part is done, but we have to even acquire a signal.

Refer slide time :(19:31)



So in order to acquire a signal, we'll see different ways of acquise, acquiring a signal. As we've already told you, I've, as I've already discussed with you, so one way of acquiring a signal is a counter. Another way of acquiring a signal is, using analogue input voltage. So considering it as an analogue input and measuring, you know, time duration pulses. Right? But the problem that comes with a counter is that, every time, we have to make the counter to set it to 0. So that means, we have to acquire, for a particular time, then we have to the manipulation or calculations, such that, which will be correspondent to the RPM value, meaning, suppose if you are acquiring a signal for 1 second, multiplied with 60, to get actual, then we have to compare it. But, since it is a control and simulation loop, problem if we do is, you know, the selection. So it has to wait till we receive the signal from the RPM. So as a result, sometimes if the error value is higher, where the control and simulation loop will be continuously working, so there are chances of make, you know, making the motor to go to unstable region, since the error value will keep on increasing, so the controller output, will provide an output of 5 volts, so that the motor will be going to higher. So as a result, it will be always fluctuating from high, low, high, low and again it is resetting also takes time. So, so that option is not feasible inorder to, implementing in this case. Another way of doing it is, by using, acquiring a signal, using analogue way.

So what we'll be doing is that, I'm taking a new block, so as I told you, we have another of acquiring, which is by acquiring, by connecting the output of encoder pulses to, to our analogue input channel, such that, whatever the signal, the pulses that we are acquiring, will be in the form of analogue input voltage. But the sampling rate, that we are acquiring, should be higher than that of, higher than, the maximum sampling, maximum frequency at which, our encoder is working. So from the previous analysis, we know that, the maximum frequency that the encoder is working when the input voltage is at 5 volts, to a DC Motor, is of somewhere around, 1.5 kilo. So if we can properly choose our set point, choose our sampling rate, such that, somewhere around 5 times to 10 times. So which will sat, which should satisfy the nikris criterion**.** So we don't miss any of the signal, which we can accurately measure. Why don't we do that? So what I will be doing is that, I'll take, I'll create one acquisition system, acquiring analogue input voltage control and this was, physical channel should be, Day 1 / AI 0. AI indicates out analogue input 0 channels. So if you want to understand, where is that analogue input channel? So here we can see A02, second channel. So what I'll be doing is that, I'll replace my connections to second terminal, so this is the counter block. So I'll, I'm removing this counter, then I'm connecting it to the second. So which is, now it is acquiring using analogue input channel. Now inorder to see, what kind of signal we are requiring, we also have to create logic here. I, I can simply go with an express VI, so I'll take an Express VI, for acquiring the data, data assist. So here, I'll be acquiring analogue input, which is of, in the form of voltage. And since I have connected, replaced the connections from the counter signal to AI 0, I'll go with AI 0, then finish. So let me run one, so that we can understand, what type of signal we are getting it there. 'Run' right, here we can see.

But, we don't need a differential, because we are considering all grounds are same, so I'm going with RSE, then sampling rate is another one, so change it to 10k and the samples to read, this is fine, yeah, here we can see. So we are getting a digital pulse, in analogue fashion. But, as long as, number of sample, samples is higher, we won't have any problem. So we can easily visualize the signal. That's okay. Now, create graphic indicator. Here we can see. Right? We are getting digital pulses. But the idea is that, if we can find out the frequency of the signal, since we have N number of signals, if we can find out the frequency of the signal. And convert into, divided by 7, because, per revolution we are getting 7 pulses, multiplied by, 60, gives us RPM. So, inorder to measure frequency, we have a blocks in signal processing, which is nothing but, way for measurement, where it gives pulse measurement value. So I will go with pulse measurement value, let me, there are express ways, available even for measuring Wave Form. So here we can see Amplitude and Level, Timing and Transaction, I'll change it to frequency. Okay? Press, [Okay]. So, I will be connecting the signal here and frequency. Create indicator, create indicator, Continue. So this is the frequency. Right? Now the motor is rotating with 1 volt. So we can see the frequency of the signal is, 163 hertz. Switch OFF. So the error is because of the slit middle heat. Let me 'Run', once again. 175, 172, 178. So the idea is that, this only gives the frequency, but we have to convert into RPM. So the way of doing it, is that. So I'll change, I'll use some comparison operators, so like, sorry, I will be going with numerical operators, where the, the frequency should divided by, divided by 7 and multiplied by 6. As well as, 100, again the value should be divided by 100, because whatever the frequency, whatever the RPM we are getting is from the motor, without the gear ratio. So since the gear ratio, deduction ratio is 100, I'm dividing with 100, then multiply with 60. So when I see the RPM value, I'm creating numeric indicator, which is RPM here. So when it is rotating in 1 volt, we can see 17.7783 is an RPM and it is keep on fluctuating. Right? Suppose if I change my samples to reach to 500, 14.315, 14.7, 15.3. But why there is a fluctuation? When you look into the, this, this value is not steady, it is keep on changing. The reason is that, whatever output the that we are getting, when you look into this, configuration. Okay. Let me look into the help, tutorial help. The problem is that, the sig, inorder to understand, the frequency or different timing parameters, like frequency period, pulse duration, duty cycle. So what it does is that, it will take only a single pulse. Either it can be a starting pulse, or it can be ending pulse or it can be between the pulse. Right? It cannot take the average. So as a result, if the pulse value, because of, you know, if the pulse duration has changed, then automatically, the whatever RPM signal that we get, will also be continuously changing. Observe it. So 14.3, 15.3 so keep on fluctuating it. So one way to avoid is that, if I, if we can take multiple samples, average it out and finding out an RPM. But with this particular Wave Form measurement blocks, we cannot do that. Because, even if you look into other wave form measurement block, which we have seen previously, pulse and measurement, even here we can see, it will help here, so pulse number specifies, which pulse are polarity, high or low, LabVIEW measures.

For pulse number and with polarity low. The VI measures the pulse duration and center of the nth, low pulse. So which means, that it depends upon, what is the pulse, that you're providing it? Pulse number. So only to the pulse number, depends upon, which pulse number you are looking for. Only that particular pulse numbers, duty cycle periods and pulse width will be measured. If you change the pulse number to 2, the second pulse, third one, third pulse. So which means that, it will, cannot give us average of pulse. So inorder to get an average of pulse, one way, to go is, rather than using a wave form measurements, if we can take, if we can measure, if we can use FFT of the signal. So now if you look into the signal, in processing here, Wave form measurement, using a spectral analysis, which is nothing but, where you can do, FFT measurement. So power spectrum, FFT, Linear, dB, press [Okay]. Now, if I connect the signal here, the output from this, then, let me keep everything aside, for time being. Then, create indicator, graph indicator, so let me change it to, 1500, this is the maximum frequency at which, we can go and this one, X Scale, I will change it to, or remove the Auto Scale. Right. Now if we can see, let me change this to 1k, 1000 samples. Right? We're getting multiples, you know, multiple pulses. So here we can see, let me replace, change some settings, auto scale to 0 and maximum frequency is 1500. Now it is clear that, depends upon at what RPM the motor is at, we can find out the frequency, because which use the average of the Input Signal Frequency.

So let me change, the input voltage, to 2 volts, by using some, our test panels, analogue output to 2 volts and let me update. So right now the motor is rotating at 2 volts and when I acquire the signal. Right? So when it is a 2 volts we can see this particular peak is somewhere around 500 hertz. Not exactly 500, but somewhere around. Suppose if I change it to 3 volts, let me update and run this. Right? It is a, greater than750. And if it is at 4, update, run this, greater than 1000 hertz, then 5, run this, somewhere around, close to 1500, somewhere around 1400, something. But if we can measure this particular value at, what is the peak value, that we are getting and at what frequency that we are getting? Our problem is solved. Right? So if it gives us average signal and no matter what, how many number of times you are rotating, the value will remain same, average value will remain same. We can see, so let me run it to continuously and change the value to 1, 'Updating', here we can see. So it is continuously rotating, the average value it is giving it there. Right? So, let me change it to 2 volts and update it. Right? In the previous case, when we have seen, the RPM value was keep on fluctuating, because, as the signal, no the signal which is being measured is not an average value. Right? So, going with 3 volts, so no matter what, how much time that we arrive at this peak, peak is always at that particular point, with very little change. 4, updating it, 5, updating it. Right? So no matter what, one thing we, we are clear that, rather than going with a simple signal measurement, only of one peak, one, one signal, if you go average signal, it will be good. But, how do we measure the peak, how do we measure this particular peak? And how do we measure the RPM of this peak, the frequency of this peak? So inorder to that, what we can do is that, what we can do? We can do some; we can create a logic here. So one logic is that, we have to find out, the peak of it. So we have, we have a block called, 'Peak Measurement', measuring a peak of a signal. So I'm taking a peak detector here, connecting the input here and threshold, I'll set the threshold to somewhere around, 2 volts. Because whatever the signal that we have seen, the maximum threshold is 2, so I will change it to 1.5 and width peaks, how many peaks we want? We, whether we need peaks or valley. So because our intention was only to find out a peak, not a valley. So incase if we want to understand completely about this particular function, which is available in the LabVIEW, you can look into the detailed help. Then indicator, this gives, whether it has found any peaks or not? How many number of peaks it has found, above the threshold value? And, if I want to move those locations, our intention was to find out the locations. So create indicator, so this particular, this is what, helpful for us, so let me [Run]. Right?

So we found only one peak, which is greater than 1.4, 1.5, because remaining things are smaller. So it's a kind of a filtering that we are doing, thresholding as a filtering. And the value, it is showing somewhere around 20.96, but here it is showing 200, but the location of it, it is showing to 2, 20.96. What it indicates? Okay. Inorder to understand that, let me change the voltage, I'm again creating a test panel, running it continuously, going to test panel, Analogue output, changing it to, 2. Now we can see the location is somewhere around 52.7, 52.8 signal, so which makes sense. But it is, it is also, you know, somewhere 500 something. Keeping it to 3, 84.7, 80 point, 84.89, so somewhere around 890, 4. 112.8, 112.6, this is also somewhere around 1120, 1130, somewhere. So going to 5, so 1400 and here if you see, somewhere around 1004, close to 1500. So which is 1460, 70 which is also close to 500. But it is thou, 147. But if you see that, we can understand clearly that, there is a fraction of, there is a scaling factor of 10 that is because of, the sampling the samples that we are considering at. So incase, here, let me stop. Suppose if I use the data acquisition samples to be, 10k, but it has to wait for one second, inorder to acquire the signal and let me run continuously. Since we have so many number of signals, we can see accurately. 1487, fut., 77 minutes, is running at 5 volts. Right? This is also somewhere around that. I'm changing it to 1, let me update, we can see the signal, 200, 195, 196, 2 volts, 516. So that means, the value, the location value, entirely depends upon, what is the sampling that we have set, it also depends upon, how many number of samples that we are acquiring from the, sampling that, that we have set. So since in the previous case, we have set a sampling rate of 1k, 10k and the number of samples we are acquiring is only 10, 1k, so which is a factor of 10, so the output location is also a factor of 10. So inorder to map that, what I will do is that, one thing is clear is that, we have to know, our sampling rate, so [Create], [Control], I can set the sampling rate from here, then I also want to, I should know, how many number of samples. [Control], so I'm changing the number of samples to 1000, 1000. So depends upon, what sampling rate and sample that we are considering. The, the, the frequency of that also, should be understandable.

So what I will be doing is that, for that, these are all everything clear to us, only the scaling factor term that we have to work out right now. So from the locations that we have found. So I'll be taking only the first value, since it is an array. So I'll go with Index Array, this Index Array gives me the first one, then to this, I have to multiply with a factor of sampling rate divided by, number of samples. So what I will do is that, I will take a division, operation, divide and one should be from sampling rater and another one should be number of samples, then, the output of this should be, multiplied with that factor. So I will take another multiplication factor, multiplication, where the output of this will be multiplied. So, whatever if gives me, when I do this particular thing, is a frequency. So this is our frequency term and this should be connected here. Now when I see, let me run continuously. Here we can see, okay, frequency is, oh, sorry, there is a misconnection, this should be connected here, not here, okay, let me run, continuously. Here we can see, the frequency is, fif, 528, so factor is also same, 45. What if the number of samples is 10,000? Updation rate will be, every 1 second. So we can see 45. No matter what, what's the rate that we set it here? So for example say, 500, the RPM will remain same and since we are taking an effective of the signal, so 45 RPM, still 45. But the input voltage is of somewhere around 2, so let me change it to 1 volt and see. So 17, so previously we got 16 something. So because that was not, that is our manual calculation, whereas in this case we can clearly see 17, 18. So change it to 1000, 1000 samples, 17.9.

Again I will change it to 2 volts 45, previously we got 44, 3, updating it, 72, we got also 72, 4, 100 RPM, we got 99, 5, 125 we got and we are getting here, somewhere around 124.5, 125. Right? So one thing it is clear that, no matter whichever the way that we are doing it, even with, Analogue acquisition or pulsing counting at, counting way, we are getting an accurate value. But only thing is that, which is feasible with our programming style that we have to understand. So in this case, the previous one, may not work properly, so where as this particular block will work accurately. So what I will be doing is that, so I'll be creating it as an VI, sub VI and I will using that sub VI there. So for that, I will also do small, okay, or I can copy and paste completely there, which gives us RPM, anything is fine. But, I'll create a sub VI, this gives us sampling rate, this gives us number of samples and the output is RPM, Frequency and this is a respulse.

We also have to create error in and error out. So let me make this to 1 volt. Okay? So for creating error in and error out, what I will be doing is that, nothing error out of this to here, create indicator, this is our error out block, connecting it here, then, so here we have error, create, control and taking it down, so this block is this and if I want to stop, create, control, we can use this. So let me save it. Acquire, as well as, signal acquisition and conditioning, for DC motor. So this part is also done for us, so only thing is that, now we have to call it. So I will also change the icon, I'm highlighting the icon, going to layers, deleting this part, so this is nothing but, signal acquisition, DC, acquire signal conditioning. So pressing [Okay]. So let me call this VI in our main VI, which is also corresponding to the same unit, which is RPM. So what I will do is, I'll select the VI, here I will be calling, Acquire DC Motor. Now the output of this has to be given to here. But to visualize it, better to also have the Graph Indicator, which is our pulses, so I'll also keep close to this and the same way, even the output is nothing but, our actual RPM and since we also have to set the sampling rate and everything, I'm changing it as, Control, this is a sampling rate, as well as number of samples, create control, number of samples, frequency, so if I want to see the frequency of it, but anyway, we already have RPM, this is good. Now, I have to pass the error of this to here and this is an error output. So the whole idea is incase if there is an error, then this has to switch OFF. So let me create a Y loop and I'm creating a logic, constant or control. So what I will do is that, I will take Control or Stop button, going to Silver, Boolean, Stop. So if I press Stop, it has to stop it. So I'll be keeping, keeping it somewhere. Okay. So, I've also created a Stop button, if the user wants to stop it, as well as, if there is any error or I'll create a Boolean logic or so whether user or the error, should make the system 'ON'. Okay? Then, so quickly if we observe, we have also created Set point, Error Amplifier, Gain, adding it, the logic for manipulating, sending the data and also acquiring the data. But we have, have to display everything on the wave form, inorder to understand, what is the set RPM and what is the actual RPM? So, for that, what I will be doing is , so I'll take array, Build Array and we have to plot the array, with respect to the simulation time, so that, both will be always in same units. Not the simulation, I yeah, with respect to the sim, the time, actual time, clean up Wire and the other one, from the actual input. So I can create, drag here. Okay? This is our Wave form graph, bring it here. So inorder to make the simulation time to be visible on the graph, I'll go with Control and Simulation, here we have some other block, called, 'Graph Utilities, Simulation Time Wave Form'. So it has its own block, so I don't have to do this, I'm just building it. So connecting the output of this to.