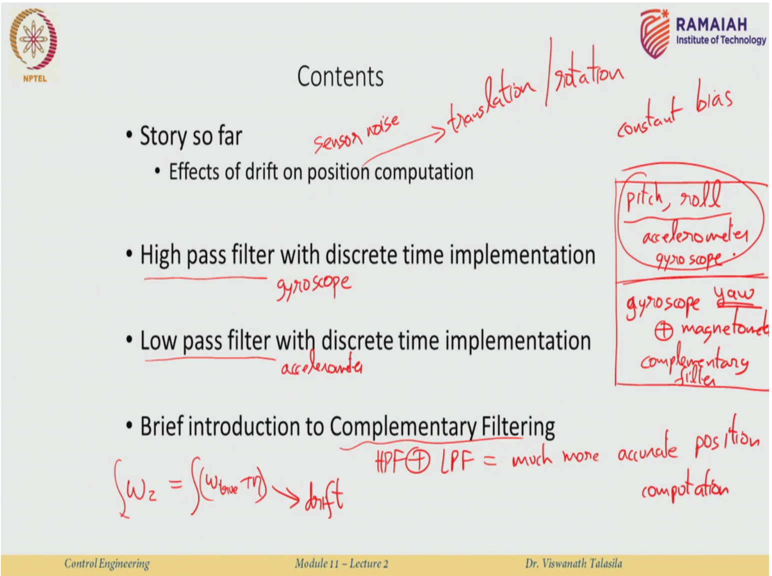
**Control Engineering**
**Dr. Viswanath Talasila**
**Department of Telecommunication Engineering**
**Ramaiah Institute of Technology, Bengaluru**

**Module – 11**
**Lecture – 02**
**Part – 02**
**Filter Design to attenuate inertial sensor noise**

Welcome to the nexus of lectures in control engineering. So, we are going to be looking at module 11 lecture 2 part 2, and it is a continuation of the study we of the discussion we had on the inertial sensors.

(Refer Slide Time: 00:40)




So, if you recall in the last lecture, I had discussed the various effects of a sensor noise specifically accelerometer sensor noise and gyroscope sensor noise on deposited on the position computation. And whenever I say position it could mean either translation or it could mean the rotation position, the angle rotation position right all right.

So, we saw a couple of things one was that there are various types of noise, there is a constant bias, the drifting bias angle random walks so on and so forth. For the kind of applications we typically deal with in consumer electronics and even sometimes to basic medical applications, it is enough for us to deal with the constant bias. So, if you are able to estimate the constant bias and then to remove that, you will get actually a very good position estimation ok.

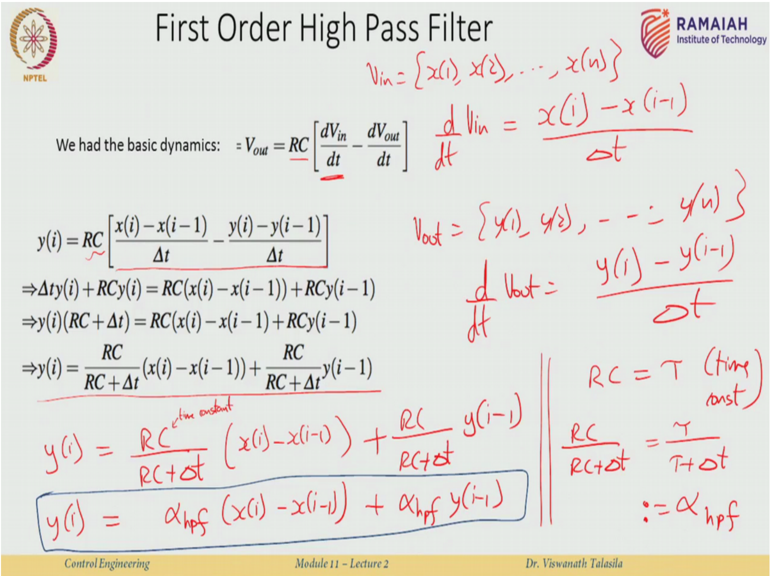
So, the way that we are going to do it in this lecture, which will be immediately followed by a discussion on the actual implementation of this in both MATLAB and on a microcontrollers all right. So, that will be done by my research associate Vinays reader. We are going to start with the discussion of a high pass filter, and we will see that we pass the gyroscope data which is the angular velocity through the high pass filter. We will then go to a low pass filter and basically we will be passing the accelerometer data through the low pass filter all right. And we will then see how to combine the high pass filtered angle output with a low pass filter angle output, in order to get a much more accurate position computation and again when I say position it could be either that x y z the transition position or the angular position the roll pitch and you are.

So, let us go ahead and one last thing you should be very careful to remember, this we are going to be looking at computing only two angular positions today. And the first will be the pitch the second will be the roll we will not be computing the yaw. There is a specific reason for this, width and it is basically that the accelerometer is a sensor which for various reasons, which I will not discuss here it is unable to compute the yaw angle which is the which is the rotation about the vertical axis right it is able to compute only the pitch or the roll, it cannot compute the yaw. So, and we know that we can compute the yaw from the gyroscope because it can measure the angular rotation angular velocity about the vertical axis, but we know that we cannot blindly use the gyroscope output because of drift right. So, whenever you integrate the gyroscope output we saw that before right.

So, if you take the vertical axis as a z axis, and then you integrate the angular velocity which you get about the z axis, you know that you are integrating two quantities. The true value about the z axis plus some noise, and you know that this leads to drift. And what we are trying to claim in this lecture is if you combine the gyroscope with an accelerometer by using these filters, you can actually reduce drift. But because we cannot compute the yaw angle using the accelerometer, we will not compute the yaw angle directly with the gyroscope. There is actually another way of doing it which is you combine the gyroscope with a magnetometer and a magnetometer basically gives you the yaw angle or the heading angle.

So, this can again be done in the same complementary filter technique that I am going to explain today, except that instead of the accelerometer we use a magnetometer all right. And there will be basically two combine a gyroscope and magnetometer in order to get the yaw angle. So, yaw is obtained from these guys, for today's lecture we are going to be focusing only on combining the gyroscope and the accelerometer data in order to compute the pitch and the roll ok let us see how we are going to do that.

(Refer Slide Time: 05:48)




So, before we go ahead what we need to do is to first look at what is a basic first order high pass filter and then we look at a low pass filter and then a complimentary filter. Now you may be surprised that we are trying to do a real application over here and we are still just using first order filters, well it turns out that in most practical applications in the industry as far as possible people try to prefer to use only first order, and maybe if necessary second order filters typically we do not like to go much higher than that at least in the area of navigation ok.

So, what is the first order high pass filter? A physical representation of that could be something as simple as this. So, you have a RC filter basically. So, the capacitor followed by a resistor R. So, if I then measure the voltage across this over here that will be I will just call it as V out and we apply a voltage of the input terminals we call it b n. Now this functions like a high pass filter and you already know that a high pass filter is nothing, but what is called as a lead compensator. If you do not know this that is not an issue, but for those of you who are going to go into design of control systems the term lead compensator essentially means a high pass filter and you will see later that a lag compensator is a low pass filter anyway.

So, this is the physical realization of the high pass filter. So, the first thing we will try to do is to derive the transfer function of this filter. So, transfer function of the high pass filter, some of you would have already seen this with the basic derivation of electric circuits and transfer functions. So, the way that we are going to do is as follows. So, let us look at V out, the output voltage across the resistor terminals it is nothing, but let us say a current I is flowing clockwise in the circuit it is I times R ok.

So, let us keep the suicide all right. So, this I times R. Now we know the basic one of the basic constitutive relationships between the charge and the voltage drop across the capacitor, which is basically Q is C times V. Now if we take a look at our capacitor over here, the voltage drop across the capacitor. So, let me just write this down. So, the voltage drop across my capacitor C is basically V in minus V out ok.

So, this then implies that Q is nothing, but V in V out this I will call it a step number 2. Now we go to step number 3, where we know the basic relationship between the charge in the capacitor and the current flowing through the capacitor which is that I is equal to dQ by dt and then we substitute this expression for Q over here back into this derivative term and you basically get d by dt c of V in minus V out. So, this can be written as C d dt V in minus V out. So, that is the expression for the current through the capacitor.

Now, substitute into one what are we going to substitute, the expression for the current into one. So, substitute 3 into 1, this would give V out is nothing, but C d dt V in minus V out times R right. So, this is RC dV in by dt minus dV out by dt that is basically the expression that you are going to get. So, if I now write all of this in a Laplace transform domain. So, transform I will call this as this entire equation as equation number 4. So, transform 4 into a Laplace domain and what do we get well? We get V out of S, I know that on the right hand side you would get RC times s V in of s that is because the d by dt term in the Laplace domain it is s all right minus RC minus RCs V out of S this expression we get.

So, take this term on to the left hand side and take the V out of first common, and you will basically get V out of s into 1 plus RC s on the left hand side is equal to RC s into V in of s. Now we know the definition of the transfer function it is a ratio of the output response to the input signal. So, we get G of s equal to V out s divided by V in of s all right and that will be nothing, but the transfer function of a first order high pass filter. So, this is the cool little guy we have been trying to derive all along ok.

So, that is a transfer function of a sort of high pass filter now for those of you who were already comfortable in MATLAB, what he can go ahead and do is to well let me erase make some space over here. So, that I can write that is enough let me not spoil this anymore. So, for those of you who are comfortable with MATLAB by now what you can do is to take different values of R and C and you can use the command tf in MATLAB in order to define your system the high pass filter.

So, we can actually call it system or you can call it a filter if you like that is a better word. So, we can call it h p f filter is equal to transfer function of the numerator comma denominator. In this and the way you would write it down in MATLAB is basically numerator is. So, you have a s term over here. So, it would be RC and there is no constant term. So, that is a 0 that is a vector right and the denominator term would be you have a constant term 1 and you have RC over here ok.

So, you can define a transfer function in MATLAB for different values of r and c and then you can go ahead and try to look at you know the bode the bode plot of your high pass filter. So, the bode power plot hpf of the high pass filter. So, this is a command in MATLAB. So, when you actually plot the bode plot of the high pass filter let me erase make some more space over here, right you are actually going to get you should if you do it carefully, you are going to get a bode magnitude plot you will not worry about the phase over here too much where this is in a log scale the frequency on the x axis and then you have the magnitude of G of s in decibels. And this being a high pass filter you would actually expect to see a magnitude plot of this type all right where this is the cutoff frequency let us see.

All of you know how we can calculate the cutoff frequency. So, we have the high pass frequency response plot over here and you see the cutoff frequency is shown over here actually it should be slightly over here around the 3 dB point. So, let me correct that let me draw this again that is a better plot. So, this is a cutoff frequency, and the cutoff frequency as all of you know you can compete with the expression 1 over 2 pi RC. So, given the values of R and C you know now what is the specific cutoff frequency which you can actually use all right.

So, what have we done in this slide we have derived the transfer function of a high pass filter we have seen what are the command we need to use to plot the bode plot to first of all define the transfer function in MATLAB, then plot the bode plot the bloody the frequency response in MATLAB and you will actually get something like this the magnitude part you can see a similar one for the phase. Now with this you need to remember that this is a continuous time filter right of course, what we would like to actually do is to use this in discrete time on a computer or in a microcontroller.

So, we will actually need to go for discrete time version of this continuous time filter, that is what will actually need to do and this is what I am going to do in the next slide. So, we have the same high pass filter. So, we had the capacitor and the resistor right and for this we got the basic expression for the output across the capacitor, as a function of the other variables. Now if you look at each of these variables let us just plot the input V in as a function of time and also the output Vout as a function of time. So, let us say we in a some signal like that and ok.

So, we have the output filtered version something like this, now this is a continuous time what you actually do is to pass this through a to d converter, and you take it into your microcontroller or into your computer or something like that, and when you do that you actually only have just discrete samples right. So, you may have this this this so on depending on how fast your sampling rate is right, and the same for the output as well. So, you will not have the continuous signal, you will just have discrete parts of that and so on for all the other points as well. So, how would we represent that well one way to do it is to actually use the notation over here. So, the input signal V in you can see here the input signal V in right. So, it is a collection of points. So, I will call the first point as x 1, I can call the second point as x 2, x 3 so on and so forth right.

So, that is what I have done over here. So, this n which you have will depend on the sampling rate and for how much time you have actually collected the data. So, it can be a really large number if you collect data for a few minutes. So, this will be essentially the discrete version of the continuous time signal. So, these collection of points is a discrete version of this continuous time signal. In the same way we measure the output we again collect all the discrete samples over here, and we call it as y of 1 y of 2 so on till y of n. Now when you do this we need to now basically plug these two expressions back into this equation, and we know that d by dt in the difference time would basically be converted to a difference equation and you have seen this difference equation come about in when we computed the angular position, if we had a measurement as angular velocity right or we computed the basic velocity when we had the acceleration measurement with us. So, we did a single integration in both cases.

So, that is basically your difference equation. So, how would this look if I actually write down the difference equation. So, the dV in by dt. So, remember that Vi n was nothing, but x of 1 x of 2 so on till x of n depending on how much time you have collecting your data. So, d by dt of d of Vi n would be nothing, but i the current sample minus the previous sample divided by the amount of time difference between these two samples and the time difference will be essentially your sampling rate which we are looking at ok.

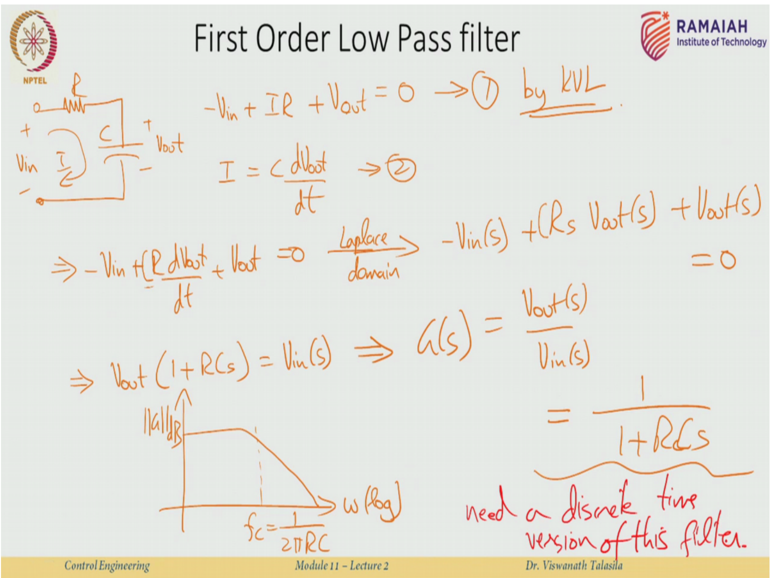
So, that is how we write d by dt V in and we know that we had written V out as y of 1 y of 2 or till y of n and in exactly the same way dv out would be nothing, but y the current sample which are measuring minus the previous sample divided by the time difference between the two samples and that is basically what we have over here. Now if I take this expressions, remember we are multiplying this with RC as we see here and you do very simple algebraic manipulations and the derivation is given over here you will end up with this final expression. So, what does the final expression look like let me just write this down again. So, the output at the current time is a function of some constants.

So, this is actually a time constant let us call as RC, a function of the of these constants times the current input minus the previous input plus again we have these constants over here and the previous output, which we measured last time all right. So, what we do is to replace this node with a different notation. So, we will typically call RC as tau the time constant and we will call because this is a high pass filter. So, we will have RC divided by RC plus delta t which in our case is nothing, but tau by tau plus delta t and this entire term we are going to denote by alpha high pass filter ok.

So, it basically means the current output after the filtering right the current output is alpha high pass filter times x i minus x i minus 1 plus alpha high pass filter y of i minus 1. This is the expression that we are going to be using in the discrete time implementation and we are actually going to show an example of this in MATLAB as well as in the microcontroller code. So, this is what we are going to do with the high pass filter, we have seen how to compute the frequency response we have seen how to compute the cutoff frequency by ok.

So, if I go back we have seen that the cutoff frequency let us use a different color. So, the cutoff frequency was over here 1 over 2 pi RC. Now we know that tau is basically RC right which is nothing, but the time constant. And we will see in the next lecture how to specifically to choose this value of tau based on the design constraints that we have to consider which may depend on how fast your motion is what kind of filtering you want to use and so on. So, this will be done in the next lecture by Vinays for either all right very good. So, that is a first order high pass filter and just to recall the gyroscope angular measurement data after computing the angle is going to be passed through this high pass filter, we will see that.

(Refer Slide Time: 25:00)




Now, let us look at a low pass filter. So, how would a physical realization of a low pass filter look like well you would have a resistor followed by the capacitor over there, and then we of course, would write V out a signal over here this is your R and this is your C.

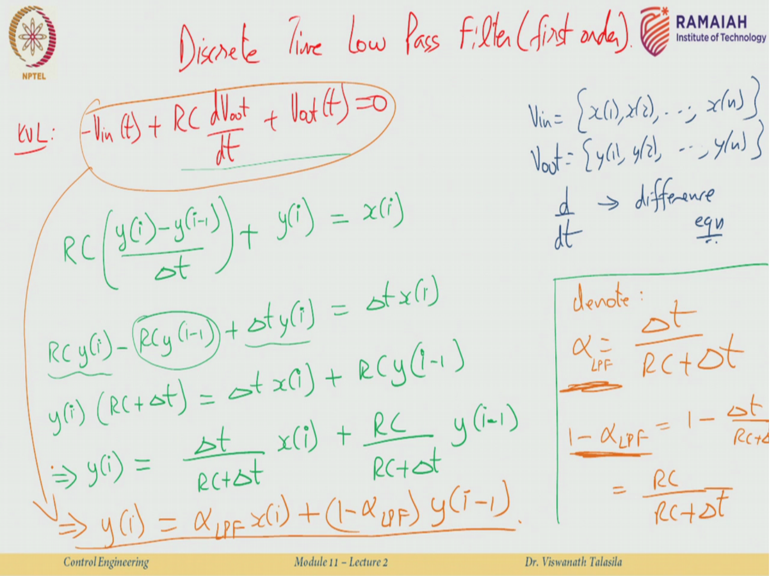
Now why do not we apply KVL and see what we are going to get. So, let us say you have a current over here and if I apply KVL you would actually get minus V in because I am going in a clockwise direction plus I times R plus the voltage drop across the capacitor which is the same as V out is equal to 0 this is by KVL.

Now, the current in this loop is the same as a current which is passing through the capacitor and what is the expression of the current which is passing through the capacitor I is basically c dV out by t right that is the expression number 2 and we substitute now 2 in 1. So, this gives us minus V in plus R dV out by dt plus V out equals 0. Transform this to the Laplace domain and remember that d by dt in the Laplace domain is basically S. So, when we transform this to the Laplace domain you are going to get minus V in of S plus R times s times V out of s plus V out of us is equal to 0 and now all you need to do is to simplify this expression.

So, we take V in on the right hand side and take the V out term common over here. So, this would imply V out into I am sorry they should be C over here, because I is equal to c dv by dt. So, in this expression when I replaced I had forgotten to include the C anyway. So, the C comes over here you will also have a C over here all right good. So, now, you take V out common and you would basically have one plus RC s, On the left hand side on the right hand side you will just have V in this of course, gives us the transfer function which is the output response to the input signal, which is one over one plus RCs, this is the transfer function of the low pass filter. And as before if you try to compute the frequency response of this and you look at the magnitude plot, you will get a response like this. So, this is omega in a log scale this is the magnitude in decibels and this would be your cutoff frequency, once again it is equal to 1 over 2 pi RC very similar to the high pass filter derivation so, good.

So, that is the low pass filter and now let us. So, as we discussed before all of this is a continuous time implementation, and what we will actually need is to have. So, we need a discrete time version of this filter, because we want to put all of this into a microcontroller or a computer.

(Refer Slide Time: 29:14)




So, how are you going we going we going to get it discrete time version? So, discrete time low pass filter and to be very specific this is a first order low pass filter. So, we are going to use this basic expression of the continuous time filter to compute our discrete time filter ok.

So, let us recall that, minus V in s plus R Cs V out plus V out of this. So, we change the colors. So, we had was this KVL what we used yes indeed. So, this was. So, we had the expression from the Kirchhoffs voltage law and it was derived as minus V ins plus RC let us do this again. So, with this expression from the Kirchhoffs voltage law which was minus V in of t plus RC d V out by dt minus r plus V out of the this was equal to 0. Let us try to write down the discrete time equivalent of this one in a similar way as we did before. So, let me choose blue. So, we have V in described by a sequence of points x 1, x 2. So, until right V of n.

Similarly, we had V out described by a sequence of points y 1, y 2 so Yn all right why do not we substitute this in this and we will see what this looks like, remembering that the derivative of the continuous time will become a difference equation in the discrete time. So, let me choose a nicer color now let us choose funny green. So, what will happen first let me take V in onto the right hand side, you will only be left with these terms on the left hand side. So, RC obviously, constants and the d by dt now becomes a difference equation, which will basically be us let us not say V out let us use the term y. So, we will actually say y of i which is the current sample, that is y of i minus 1 the previous sample divided by delta t, plus you have V out over here. So, it will just be again y of i the current sample and we had taken V in of t on to the right hand side.

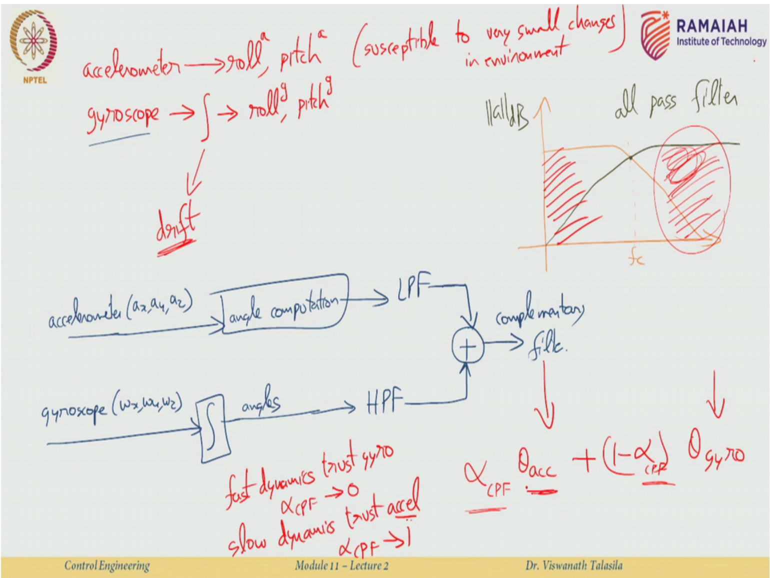
So, the V in of t will simply be x of i. I hope you have noticed I made a small mistake over here which all right. So, this is. So, this was i and this was i. So, let us see how we actually compute this relationship. So, we multiply everywhere with delta t and we will get our c y of i minus RC y i minus 1, now delta t goes over here multiplying with y of i and also on the right hand side multiplying with x of i. So, from this not that one let us take the other one. So, from this term and this term, let me take y of i common out. So, I get y of i times RC plus delta t, and let me take this term over here on to the right hand side. So, this will be nothing, but delta t times x of i plus RC times y of i minus 1.

Now you can see that I can divide with RC plus delta t, will go on to the right hand side and then eventually I will have my expression for y of i. So, this will be nothing, but delta t divided by RC plus delta t times x of i, plus RC divided by RC plus delta t times y of i minus 1. Now as we did before let me simplify this expression by denoting alpha to be delta t divided by RC plus delta t all right. So, alpha is delta t by RC plus delta t, I will also call this as the low pass alpha. In that case 1 minus alpha LPF will be nothing, but 1 minus delta t by RC plus delta t which is the same as RC by RC plus delta t ok.

So, let me use this expression and this expression in this equation and we get the final discrete time realization of a first order low pass filters. So, that is y of i is equal to alpha lpf times x of i plus 1 minus alpha l p f times y i minus 1. So, we started off with a continuous time version of the filter over here and through these steps we have now arrived at the discrete time version of the filter and again RC is nothing, but a time constant delta t is the time difference between two between two samples. So, depending on what data you have you will appropriately choose the RC and we will see this in an example by Vinays reader in the next lecture very good.

So, now that we have a high pass filter and we have a low pass filter, what is the idea of this why are we doing all this? Well the idea of this is that we want to create something called a complementary filter, and the complementary filter takes the best of both worlds it takes the best that the accelerometer data can give us and it takes the best that the gyroscope data can give us ok let us see what I mean by that is as follows.

(Refer Slide Time: 36:31)




Using the accelerometer data and w will go more into detail in the next lecture by Vinays reader, we can actually compute two angles we can compute the roll and we can compute the pitch. Using the gyroscope data and after we numerically integrate the angular rotation which we get over here. So, there is a numerical integration, we can also compute the roll and the pitch. So, this is from the axle and this is from the gyro ok.

Now, we have seen a specific problem with the gyroscope, that because of numerical integration I get the problem of drift. So, if I just use the gyroscope alone in order to compute these angles, eventually if my real angle is supposed to be some 30 degrees, my actual angle may just keep drifting away forever and you can have an unbounded error which is not good. So, why do not we use just an accelerometer; the problem with an accelerometer is that it is susceptible or you can say corrupted, it is susceptible to very small changes which is basically I am saying high frequency components, it is susceptible to very small changes in the environment and it picks up everything it is it is a very sensitive sensor. So, if I place an accelerometer on my hand right just suppose this pin is an accelerometer, and I am doing these motions. So, the accelerometer will pick it up I do want to track this.

Now, while the pen is on my hand even if I just tap very lightly very very lightly, the accelerometer is sensitive enough to pick up those kind of vibrations as well, and those start to come into the equations and you are going to get erroneous answers. Furthermore accelerometer data as explained just now is susceptible to high frequency noise. So, if you keep an accelerometer at a perfectly stable condition and we would expect the data to be something like this, along one axis of the accelerometer because of this high frequency noise and vibrations in the atmosphere all around you actually would be getting data, like this is high frequency data and that is not good. It does not drift, but it does not give you good accuracy. What you can do to remove this high frequency drift this high frequency noise is to actually pass all of this through a low pass filter right and then you can actually get a much cleaner response, and what we will do with the gyroscope is to pass it through a high pass filter primarily for the reason that a gyroscope is very accurate in computing the high speed dynamics it is not so good when your dynamics are really low speed.

So, and then when you combine the gyroscope with the accelerometer the axle meter ensures that the gyroscope does not keep drifting away, because it tries to bring it back because accelerometer does not drift. So, the basic construction which Vinay will be talking in more detail in the next lecture is that, we have the accelerometer data in all three axis. So, that is a x, a y and a z all of this going into let us say an angle computation block we will see exactly how to do that, and this will be then passed through a low pass filter. Then we have the gyroscope data, which will again you will take omega x, omega y, omega z the angular velocities do a numerical integration and pass it to an angle come well and that actually gives you the angles directly there is no angle computation block as such.

So, this gives you the angles directly after the numerical integration with drift and you pass it through a high pass filter. And what you then do is to combine both the outputs in order to get a complementary filtered output, and how does this look like? It is a little interesting. So, let me raise this stuff over here. So, we know what the frequency response of an accelerometer would of a low pass filter looks like right. So, if the low pass if the frequency response of the low pass filter looks like this where this being the cutoff frequency FC, the frequency response of a high pass filter would actually be like this and let us assume that we are clever enough to choose exactly the same cutoff frequency ok.

So, this is we are looking at the magnitude plot, what this tries to behave is like an all pass filter. So, if you pass in exactly the same signal, through this filter we will just call some signal as q, you pass q through a low pass filter you pass the same to a high pass filter both with the same cutoff frequencies, look and then you add them up the combination of this behaves like an all pass filter. We would not go more into details of this except to point out at this stage, that we are looking at filters with gains or magnitudes of one. So, that we do not amplify or attenuate these signals ok.

So, given that we will see in the next lecture by venise reader, how the use of this complementary filter the combination of a low pass and high pass, how the use of this will actually lead to much more accurate say computation of the angles, and we will also see that if you have really high speed motion I want to trust my gyroscopes more. If I have low speed motion I want to trust my accelerometer more right. So, if you look at it in the frequency response context if you have high frequency I want this part of the frequency response curve to be activated right because if I have high speed dynamics, I really want this to be activated. If you have low speed dynamics meaning my bandwidth of the system is low I do not really want to use this region, it is it is pointless it starts to bring in noise and other problems.

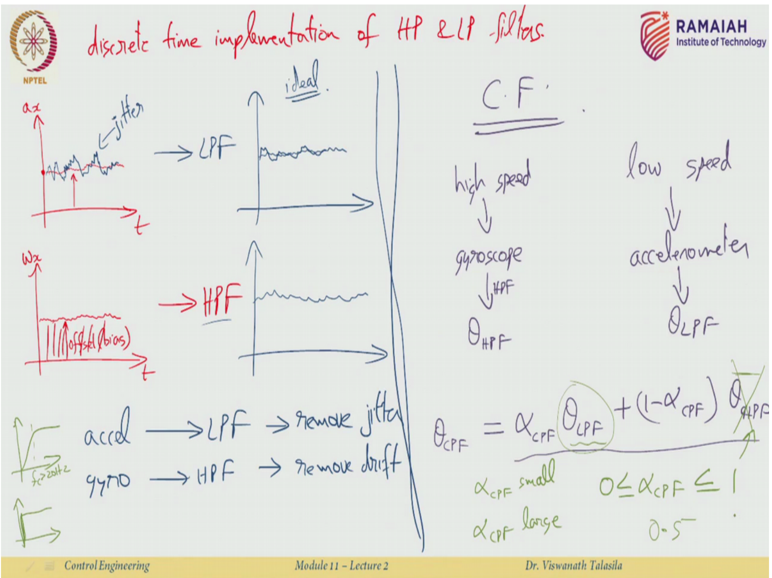
So, in that case I want to activate this part of the frequency response, how we would do that? While we would do that by writing the complimentary filter by using a special parameter called alpha, alpha complementary filter and you multiply this with the angle which is coming in from the accelerometer, and you multiply exactly 1 minus alpha with the angle coming in from the gyroscope all right and you would. So, again it is CPF. So, you would choose your alpha to be such that if my dynamics are really fast dynamics I would like to trust this one more. So, for fast dynamics I would like to trust the gyroscope and how would I tell the equation to trust the gyroscope, well you simply put alpha to be fairly low almost close to 0 for really fast dynamics. For slow dynamics I like to trust my accelerometer more.

So, you let alpha CPF almost get close to one you of course, in reality you would never use 0 and 1 because you remember the problem with drift that the gyroscope has. So, you would always want to make sure your accelerometer angles are always coming in to prevent that drift. So, it will not be really zeros or ones, it will be small numbers greater than 0 large numbers, but less than 1 and when I see that in the next lecture we will explain this in more detail with the couple of nice experiments as well ok.

So, to conclude this talk we had started out by recalling some of the effects of sensor noise on position computation translation and rotation position computation, we looked at two specific implementations the discrete time implementation of the high pass filter and the low pass filter, and we saw very briefly we will go more into detail into the next lecture, how the use of a complimentary filter can give us far more accurate whistles. What I want you to remember is that the MATLAB code and the microcontroller code that we have developed we will be actually sharing it with all of you on the forum. So, you are welcome to experiment with the codes that we are going to give you, the data that we are going to give you and you can actually design your own navigation filter ok.

So, we saw in the previous 2-3 slides, we basically wrote down what was that the discrete time realization or implementation of the high pass and the low pass filters right and the low pass filters.

(Refer Slide Time: 47:04)




Now, let us see once again why we would like to pass the gyroscope measurements through a high pass filter and axle load accelerometer measurements through a low pass filter, and then we will see the role of the complementary filter ok.

So, first let us take the accelerometer. If you actually look at the accelerometer data which we will also be providing to you on the forum, even when you keep the accelerometer in a perfectly stationary condition and we expect to see let us take along any particular axis, we can take along the x axis as a function of time. So, if it is stationary you would expect to see around the zero value, may be some minor variation like this right.

So, there is almost no variation and that is actually what we expect to see when the accelerometer is perfectly stationary. So, this is basically the noise primarily because of bias right. So, instead of being at 0 it gets offset to a certain value, which we can remove by calibration this is what we would expect to see. Now when you mount this on any stationary surface and let us say you are walking around that surface or you tap on your table or you shake your table very gently, any motion can be easily picked up by a accelerometer because it is very sensitive. So, instead of seeing the slight variation which is basically an offset that the bias offset with some and the noise parameters, the accelerometer is capable of picking up even minor vibrations in the environment and you may actually see the stationary accelerometer data to be something like this right.

So, this is the kind of actual measurements which you get from the accelerometer and this well it has many different terminology one of it we can say that it is called as jitter. So, these are the high frequency components which any minor vibrations in the environment the high frequency vibrations are easily picked up by the accelerometer and then and then measured and then you see them over there. Now because of the nature of these vibrations which are typically high frequency if you pass this through a low pass filter and if you do a really good job at that we would actually expect to see back of a nice little accelerometer data, but the high frequency components are removed of course, this is in an ideal condition it is never this perfect, you may actually get something like this.

But that is still far better than what you are seeing in the previous plot. So, that is for the accelerometer and that is a reason why we pass the accelerometer data through a low pass filter. Let us look at the gyroscope data. So, if you keep your gyroscope in a perfectly stationary condition, again you would expect remember this is angular rotation over there, it is this time you would again expect to see some fairly mild signal like that and it is not exactly at zero because you have offset or what is also called as a bias. Now unlike in the accelerometer the offset is present in all the axis.

So, when you compute the angles through the accelerometer, we would use trigonometric expressions of the type tan inverse acceleration x divided by square root of acceleration y square plus acceleration z square, and what actually happens then is you had have a small bias measurement in each one of them all right, but because you are dividing one by us with the other bias terms. So, this is the total bias contribution is quite small. So, although this is not the true angle it is still not too bad you are slightly erroneous, but it is not as if you are going to drift away completely from the true angle. In the case of the gyro as we have already seen before this is not true, and the reason for that is basically because we are integrating the signal and when you integrate the signal you also integrate this bias over there. So, what you would actually get is if the true angle is supposed to be somewhere like this.

Let us say zero degrees I will call this zero degrees, your bias is actually going the integrating the angular rotation rate is going to produce a bias of that nature all right because it is a fixed bias you have a linear drift over there. So, how to solve this problem well what you can do is to simply pass this through a high pass filter and what happens when you pass this signal through a high-pass filter, it would try to pass all of the gyroscope data except the bias which is a constant offset right. And a constant dc signal through a high pass filter is simply attenuated off right and when you do that. So, then you would not have too much of a drift your drift will be significantly reduced of course, depending on how will you choose your high pass filter coefficients.

So, accelerometer data is passed through a low pass filter and the reason for that is to remove jitter or the high frequency noise. The gyroscope data is passed through a high pass filter of course, after the integration to remove drift that is a basic idea of what we do with the high pass and the low pass filter. What do we do with the complementary filter, while we do something more interesting. So, now, let us look at the complementary filter let us say that we have high speed motion high speed dynamics say you are in a plane and then it is it is going at extreme speeds your rotation speeds or low speed motion right.

So, the aircraft is moving very gently, now because of the nature of the two sensors which we have, it turns out the gyroscope can actually capture the high speed motion very well. In fact, typical gyroscopes have fairly high bandwidth like what and it is in the order of a few thousand degrees per second whereas, accelerometers have very low bandwidth and it is meant to capture really the low speed dynamics. So, we would like to capture the low speed dynamics with the accelerometer and the high speed dynamics with the gyroscope.

So, how would you do this? So, we know that from the gyroscope we get an angle after doing the high pass filtering. So, I will just call it as theta high pass filtered angle and this has come. Out of the gyroscope out of the accelerometer we compute the angle and then we pass it through the low pass filter. So, we get another angle over here I have called theta LPf. Now how do we combine these two angles in such a way that we can capture both high speed dynamics, we can capture low speed dynamics or we can capture any dynamics in between because there is no guarantee your dynamics is always high speed or always low speed. And the way you would do that is by the use of the complementary filter, and it is a very simple equation.

So, the output coming out of the complimentary filter the angular position is nothing, but a coefficient which we will see how to calculate in later lectures by Vinays reader, the coefficient of the complementary filter this is multiplied with the angle coming out of the accelerometer plus 1 minus alpha times the angle coming out of the high pass filter which is the gyroscope after. How does this equation solve our problems? Well let us say that we have really high speed dynamics, I would like to trust my gyroscopes more because it can actually it has very high bandwidth it can capture these high speed dynamics far better. If I want to trust my gyroscopes more then I would like to actually use this far more than using the angle from the low pass filter from the accelerometer how would I do that well I will simply set the coefficient to be small; not zero, but small maybe ok.

If I have predominantly low speed dynamics then I would like to trust the angle coming out of the accelerometer more than the angle coming out of the gyro. Why is that we know that gyro integration is affected by bias, even though you have done a high pass filter of the gyro there is there is always a little bit of residual bias left and if the dynamics are really towards the low speed dynamics we are forced to choose the cutoff frequency of the high pass filter actually very close to the left axis. So, let us draw that over here, if you have very high speed dynamics a 100 hertz and above or 20 hertz and above, I can choose my high pass filter to have a cutoff frequency. let us say more than twenty hertz, But if I very low speed dynamics and I want to pass it through my gyroscope, I will need a cut off frequency much closer like this right and the problem with coming closer and closer to the lower frequencies is that I am actually going to pass in my bias that is going to get added up in the integration process and you are going to have the problem with drift. So, whenever you have low speed dynamics we really do not like to use the gyroscope we do not trust the gyroscope data, in that case what you would do?You would use the coefficient to be as large as possible of course, what do I mean by small what do I mean by large well the basic idea is that I will alpha of the complementary filter it lies between 0 and 1.

So, let us say we are absolutely sure that our vehicle will experience very high speed dynamics, in that case I would like to trust my gyro more right. So, I would actually set alpha to be almost close to zero maybe even 0. If I am always going to have very low speed dynamics 1 hertz 2 hertz kind of dynamics, I would start to prefer to trust my accelerometer more right in that case I will set alpha to be very close to one. So, that the gyroscope data is not being used. So, the interpretation of a complimentary filter is basically we choose the coefficient in such a way that we trust either the gyroscope measurements or we trust the accelerometer measurements; this is true of course, for extreme readings for any readings in between let us say that you have medium speed dynamics.

So, neither too fast not too slow then maybe we trust both the sensors equally well in which case I would choose alpha CPF to be maybe some way close to 0.5. So, that is the basic idea of the complimentary filter and we will see in the next lectures how Vinays reader is going to actually use this complimentary filter to actually compute angles on a real in a real experiment.