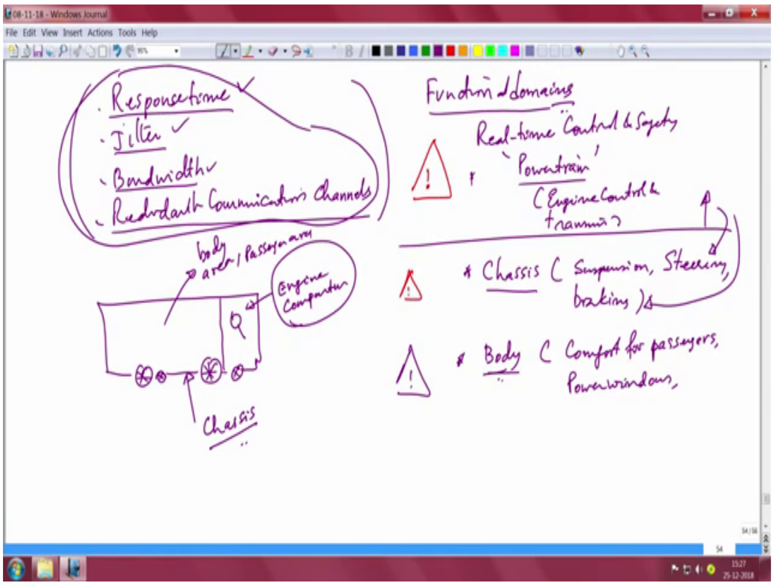
Advanced IOT Applications
Dr. T V Prabhakar
Department of Electrical Systems Engineering
Indian Institute of Science, Bangalore

Lecture – 20
On Board Diagnostics and protocols

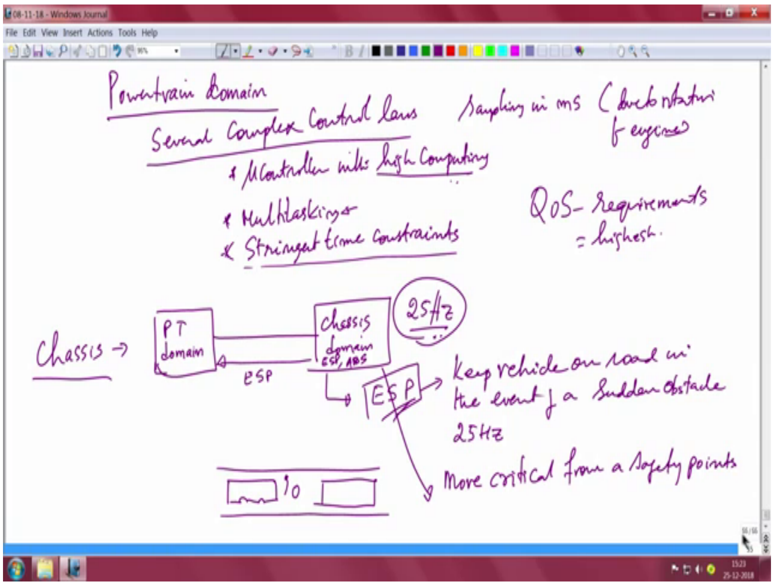
Let us get into a little more detail because we motivated ourselves into this requirement right response time, jilter, bandwidth, redundant communication channels.

(Refer Slide Time: 00:36)



We also divided the whole system into very broad areas, engine compartment we called it the powertrain part, then there is the chassis part and then there is the body part, so,powertrains, chassis and body.

(Refer Slide Time: 00:54)



So, we have to know a little more in detail about the powertrain itself. See the powertrain is very complex control law that is running inside. So, there are very complex control laws and sampling is happening at millisecond duration, because its all to do with engine rotation and all of that. So, you need a microcontroller which has very high compute, you must have OS or whatever software that is running to do multitasking, you will have to meet stringent time constraints as well. Which means, that the powertrain is a critical part there and which has certain quality of service (QoS) requirements, which perhaps is highest requirements, equally high is the requirement from the chassis part. But, what I have done is, I have tried to show you that the powertrain cannot work in independence, it cannot work in isolation, it has to work with the chassis domain. In the chassis domain the suspension is there, steering is there, braking is there, all of that is part of the ABS system. Also ESP, I never elaborated it too much to you, but indeed ESP is the Electronic Safety Program which some of the current day cars seem to come with.

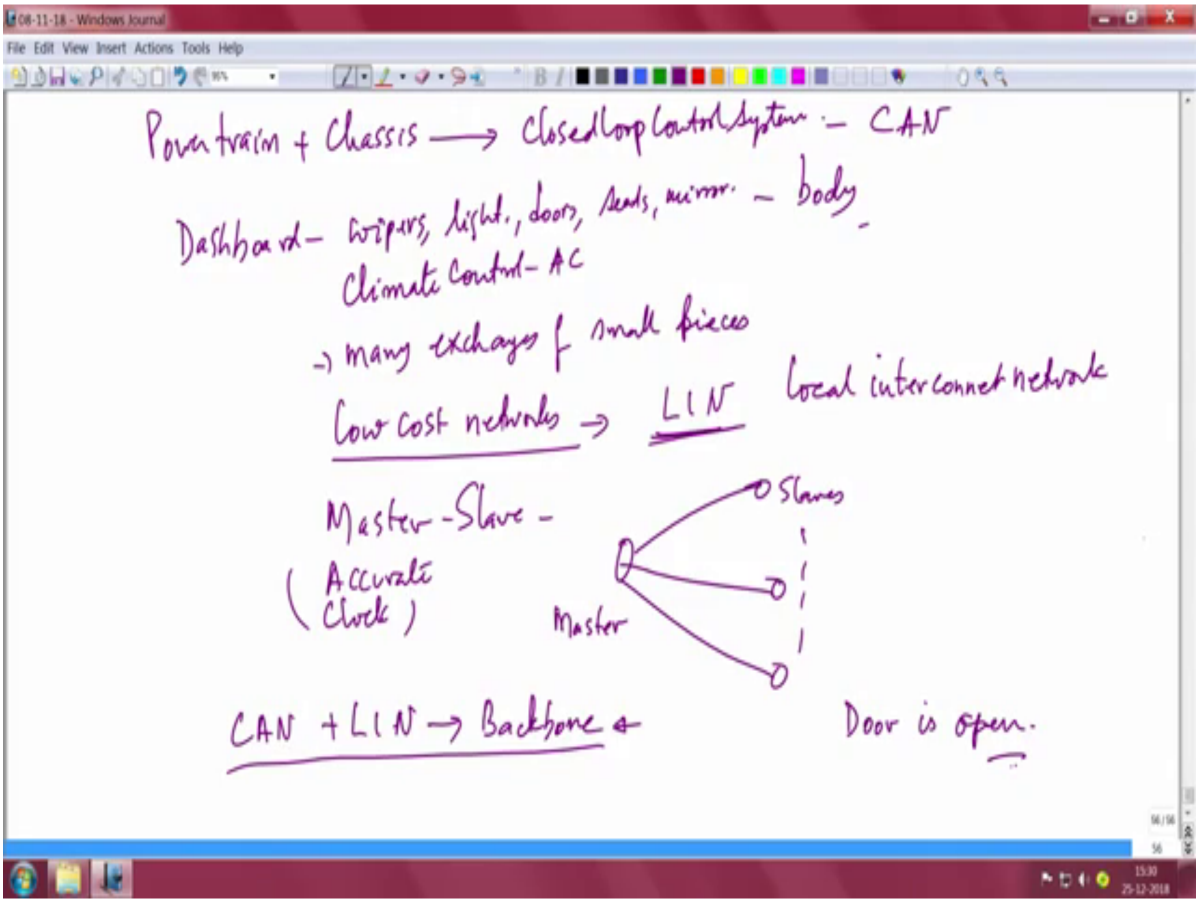
See again everything is moving in the autonomous part people are building, manufacturers vendors are building small modules, which will all of it culminate into a nice; the safe way of sitting inside vehicles. And please note I keep repeating this its not that autonomous driving has come today.

The whole traction towards building autonomous systems have happened over 30 40 years right, so, and that is what we are trying to discuss all through. So, this ESP is interesting because it allows you to keep the vehicle on road, think of a situation where there is a track and there is a big truck in front of you and you are on in your own way moving slowly towards that and then suddenly from nowhere there is an obstacle; you are going in your lane suddenly there is an obstacle.

So, the ESP program keeps the vehicle on the road by ensuring that it samples the condition of the road at something like 25 hertz. So, 25 times per second it keeps sampling using some sensor realizes that there is an obstacle and then commands back to the powertrain to cut speed and so on and apply brake or keep it on the track. In fact, cut speed in a manner that it does not skid when the steering is swiveled, when you swiver the steering just to avoid the obstacle, you are often at such a high speed that it may turn round, or you may have a collision, it can deal any dangerous situation. Even if you swiver with the steering the vehicle tends to have put the car on the road, put the vehicle back on road in the manner that it is safe for people inside.

So, is ESP is a very interesting program. It has certain constraints, it has certain requirements, when you talk about that. So, you will essentially be talking about chassis functions are indeed very important and as important as the other one as I mentioned. So, essentially more critical from a safety point of view, very important is chassis. It is more critical I would say more critical from a safety point of view. All the ESP programs which are part of the chassis indeed gives you that feeling that it is very critical.

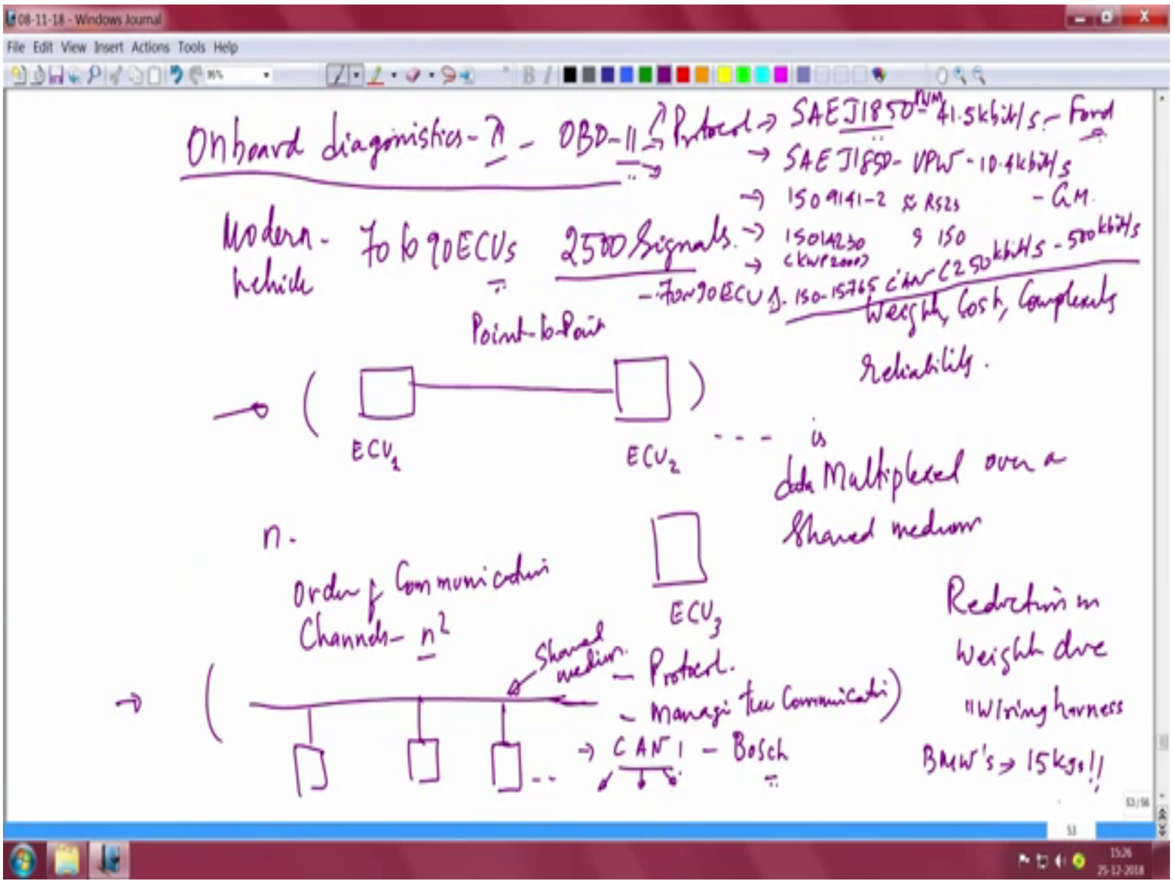
(Refer Slide Time: 05:24)



So, in summary powertrain and the chassis both operate in closed loop. This is so, important to understand that it really operates in closed loop. So, if you are familiar with controls you will find it exciting to work in this area and everything is moving towards time triggered and all that. Then there is the other one which is all about the dashboard, then you have wipers I mentioned to you ,the wipers, lights, doors, seats mirrors, you name it which are all part of climate control, these are all software based systems part of the body. Like climate control AC and so on, AC parts, blowers.

They all come under the body part and they are all mostly software control and they exchange small pieces of information among themselves. They have numerous functions, and they exchange plenty of information, many exchanges of small pieces of information. Not all nodes require a high bandwidth, you can design low cost networks and you can put them under a low cost network like LIN, LIN is local interconnect network. So, most of the body related things will have a LIN network.

(Refer Slide Time: 07:53)



But, whereas, powertrain and chassis will have a CAN network so, you see what we are trying to show you, what we are leading to is that, you are going in or away from the point to point systems to shared communication based systems. Because, you want to reduce the weight due to wiring harness and all that put them into a bus systems a single bus is not going to work for you. A single protocol is not going to work for you, because of this requirement, which requirement? this requirement: Response time, jilter bandwidth and redundant communication channels. If you put them in the forefront you will say I cannot do this with this protocol.

But, I do not need this for this requirement. So, if I you match a requirement to what you want to achieve, you have a requirement and what you can put there on the vehicle, then you will have to necessarily design multiple protocols. And that is why the fact that a vehicle has a LIN protocol comes into picture. So, here we will get into the detail of the LIN, one node which is termed as a master, possesses an accurate clock and drives the communication by pulling other nodes.

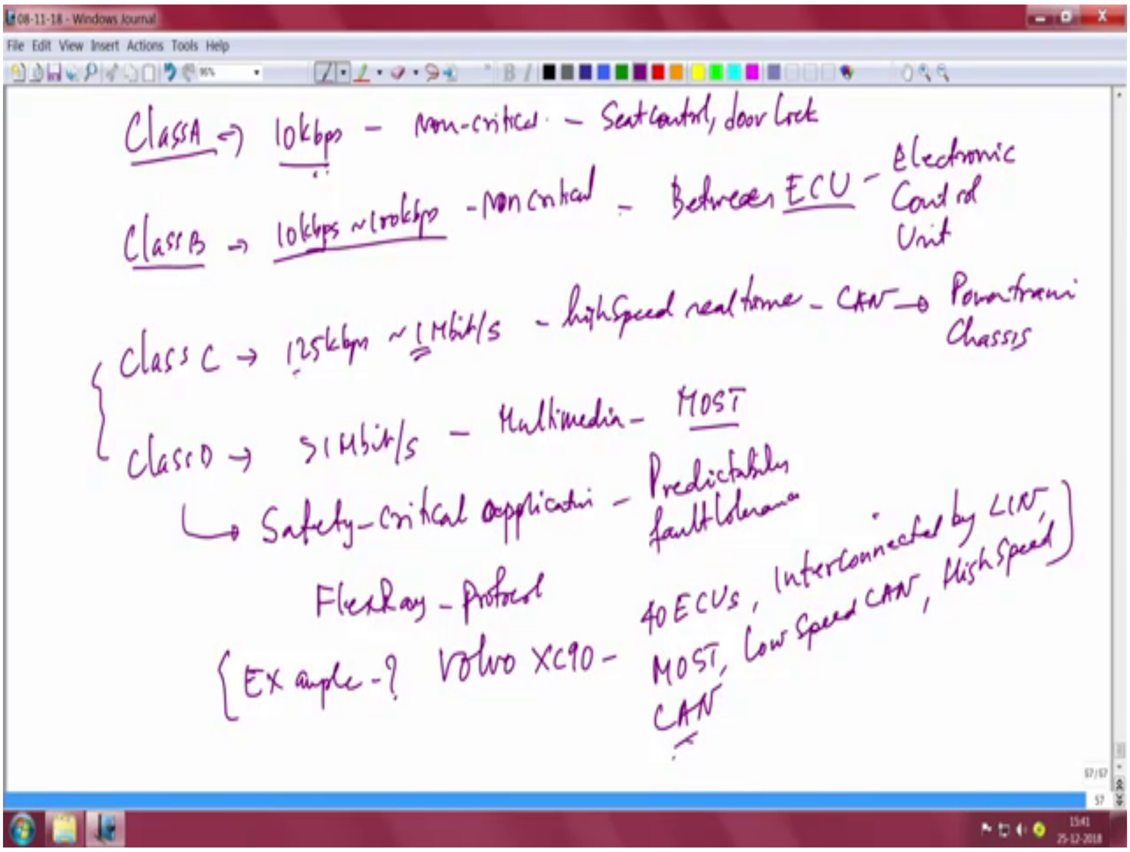
So, there is a master slave paradigm. Master holds an accurate clock and drives the communication by polling the other nodes, which are essentially slaves you poll them periodically.

There are slaves and there is the master, and then you start periodically, and then the mixture of different communication channels is now very clear motivation. Somehow you may also want interconnect. On the one side you need a different network, but you may want to interconnect and then integrate the subsystems. So, all of that you would do by mixing LIN with CAN on a backbone, CAN plus LIN, can happen only by at the backbone level. So, you will have multiple networks all of them connected at the backbone level.

I will tell you in a peculiar way, why you need to interconnect this.? For instance if a door is open, you may not want to start the engine at all, which is part of the powertrain or you may not want to release the brake if it is part of the chassis. So, unless you have an integrated solution, ECU is talking amongst themselves, but each one of them having their own specific requirements, this is not going to give you a full solution, cannot give you safety. Therefore, if you ask yourself a question door is open what should you do, the ECU simply signals that a door is open. And then the powertrain ECU says I am not going to allow this car to start.

Now, the chassis may say, that I am not going to allow the brake to release. The brake will be ON and the vehicle cannot move till the door is closed. You can have that safety consideration. So, all these things essentially means this ECUs have to communicate amongst themselves, that is a very important thing. All of this leads to different types of classes of networks.

(Refer Slide Time: 11: 39)



You will have something called class A, which is multipurpose and which is essentially very low data rate requirement., something like 10 kilobits per second. Then you will have class B networks which are typically 10 kilobits per second to about 100 kilo bits per second. And these are also non critical I would say non critical. Seat control door lock., all of them need some low cost technology. Then, you can have class B between ECUs. You may want to exchange data and they will essentially be 10 to 100 and 25 kilo bits per second, and they essentially use low speed CAN., you can be using low speed CAN. So, there essentially I would say for gluing between class A and the class C.

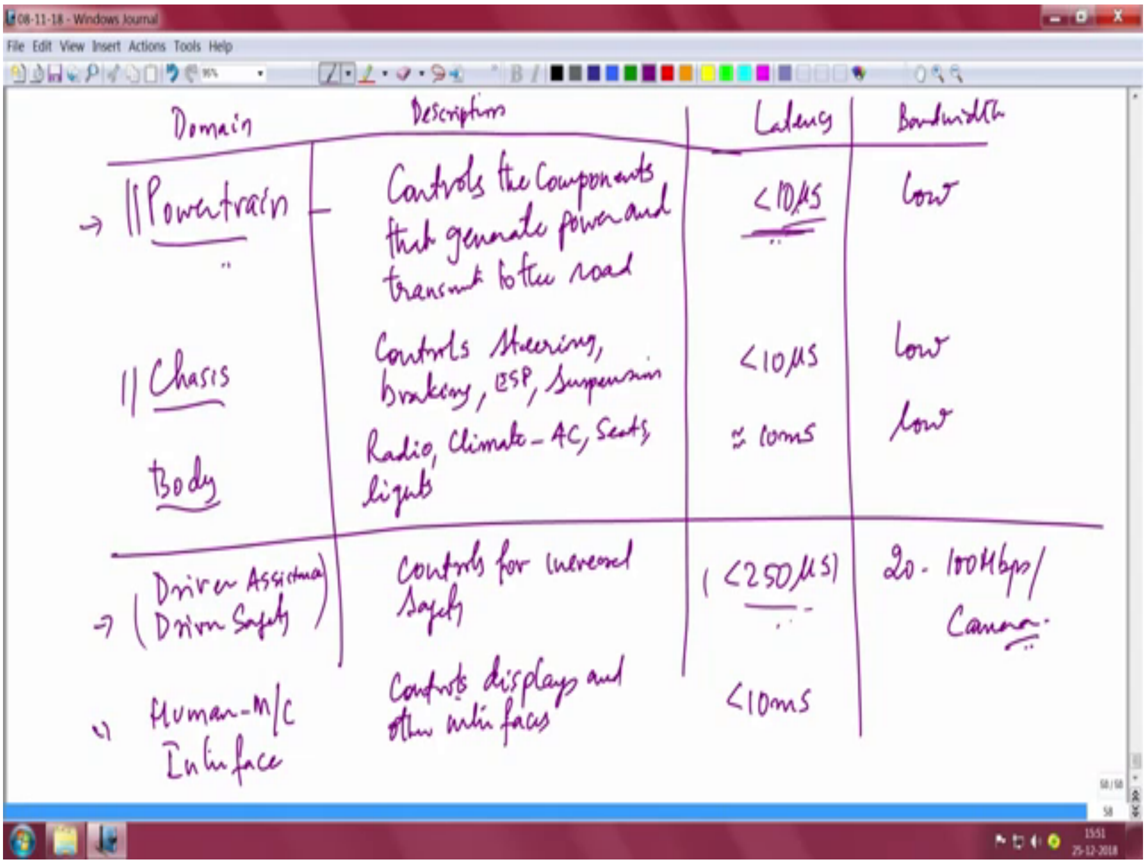
Suppose you have another class C, class B is more for gluing purpose. So, I will say between ECUs. So, when I say ECU I mean electronic control unit. If, there are multiple electronic control units, communication between them is something which you can use with 10 to 125 or 100 kilo bits per second, but is still non critical. It can also be for gluing between classes, because it is essentially between ECUs, not necessarily within the same class, but also across classes.

You will have to see what a careful decision, how you make this careful decision? But, what should be connecting at what rate between them? So, then there is class C. So, the class C essentially real time. So, if it is 100 I will say this at 125 kilo bits per second to about 1 megabit per second, 1 mbps up to 1 mbps or so, this is high speed and real time; so, that is what you would say. And then there is class D, which is more than 1 mbps greater than 1 Mbit per second this is high speed real time CAN. And then associate it with powertrain and chassis.

It is greater than one megabit per second and is mostly for multimedia. They use another protocol called MOST which is another protocol. Media oriented system transport MOST. So, that is going under class D; very high data rates. Sometimes, you may also want to use this class D for safety critical applications. Currently you can say that they are mostly centered around class C and class D, most of the modern cars are under either class C, but not the lower side, it has to be around the 1 mbps range. For all safety critical applications, where you need predictability, fault tolerance and so on, which will be using higher rates. So, here actually you will talk about another protocol, which is flex ray, MOST and flex ray. What is a good example today? look at Volvo XC 90, it has roughly 40 ECUs interconnected by LIN bus. There is also a MOST, you will see low speed CAN. Between the class C around the 500 kilo bits per second to 1 megabit per second, you will see a low speed CAN, you will also see a high speed CAN, high speed CAN all of that integrated into the vehicle.

Is that it is reality that such networks exist inside a vehicle? So, many ECUs exist, so many different type of protocols exist, inside a very complex system, complex protocol many protocols all of that if you slowly start dissecting you realize that, there are good possibilities for inserting your algorithms by using different sensors and move towards the autonomous vehicles. So, now, that we routed ourselves into these 3 broad I would say domains of a vehicle powertrain, chassis and body. We did not get answer at least to design any system that we need, what are the latency requirements of these systems?

(Refer Slide Time: 18:14)



So, I put down a small chart here, shown above; which will help you to do that. Powertrain, what does it do? it controls the components that generate power and it does do that to converts that into transmission on the road and you are talking about 10 microseconds, but not so much of any great bandwidth requirement; latency is important, end to end latency is most critical here. What is the chassis requirement, what does it do? It controls the steering, it controls the braking., I mentioned to about ESP, then I also mentioned to about active suspensions and so on.

This also there is no way by which it should be anything less than the first one right, because equally important and also the bandwidth requirement is quite low. Look at body. Body is not a problem really; you have radio, you have climate control, climate and simply call it AC, then you have seat lights, seats, lights. All of this can be anywhere up to about 10 milliseconds perhaps you do not care so, much and it also has a low requirement. These 3 sort of give you an overview. Then, if you are now going towards autonomous, this is what we know very well. If, you want to do driver assistance and you want to do safety, driver safety, which are all coming up now.

So, these are all controls design for increased safety. It should all be in the order of 250 microseconds. You can do it in about 250 microseconds and you will need 20 to about 100 megabits per second per system. These are all different sensors which you are going to put and you obviously need a lot of time to sort of process that signal and give it to you in less than one millisecond right.

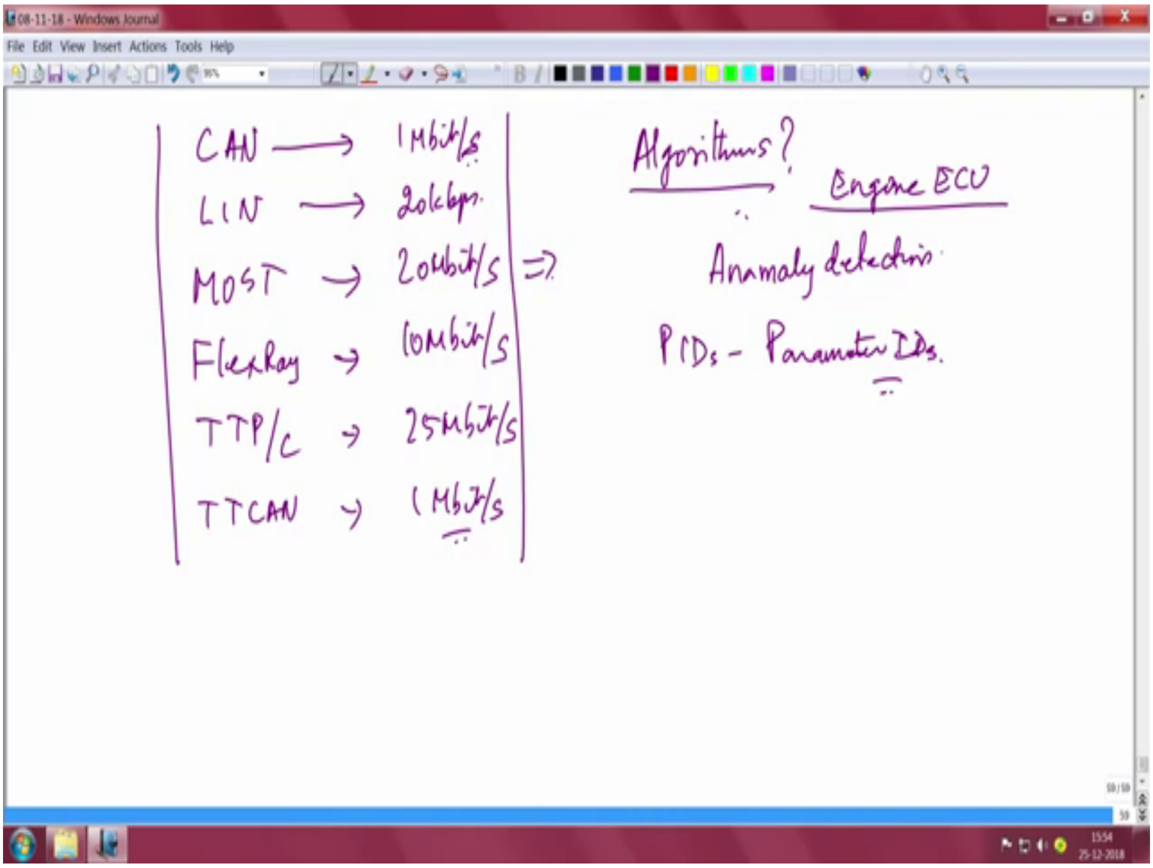
So, it is one fourth of a millisecond. Essentially this is a sufficiently high requirement compared to what you see here, which is 10 microsecond. This you can say is running at a much slower loop the driver assistance and driver safety is running at a much slower loop. Because it is looking at safety, it may have sufficient margin to decide, because we are putting a camera system. It has an ability to look ahead, to look in the night, it to look and take action, and provide you a feedback. That feedback loop is typically in the order of 250 microseconds.

Once the feedback comes which broadly falls under the domain of driver assistance and driver safety. The actuation will happen via the powertrain, which in which case it will go back to 10 microseconds. And these are add-ons, which can run at a slightly slower loop, but once detected will have to give commands to the powertrain and the powertrain itself will have to have a latency of less than 10 microsecond from the time the command is issued.

This driver assistance is not slower compared to powertrain. These are the basic requirements and this is an add on that you have put, the add on has to compute and give you in sometime and that will have to intern go back to the powertrain. Because, ultimately driver assistance or whatever you do in terms of your algorithms. The actual mechanism will have to fall under powertrain or chassis only right and they will have to be stringent latency requirements.

If, you do HMI; Human Machine Interface, which controls displays and other interfaces, this is controls displays and other interfaces with the driver passengers and all that. And this can be even in milliseconds in the order of milliseconds.

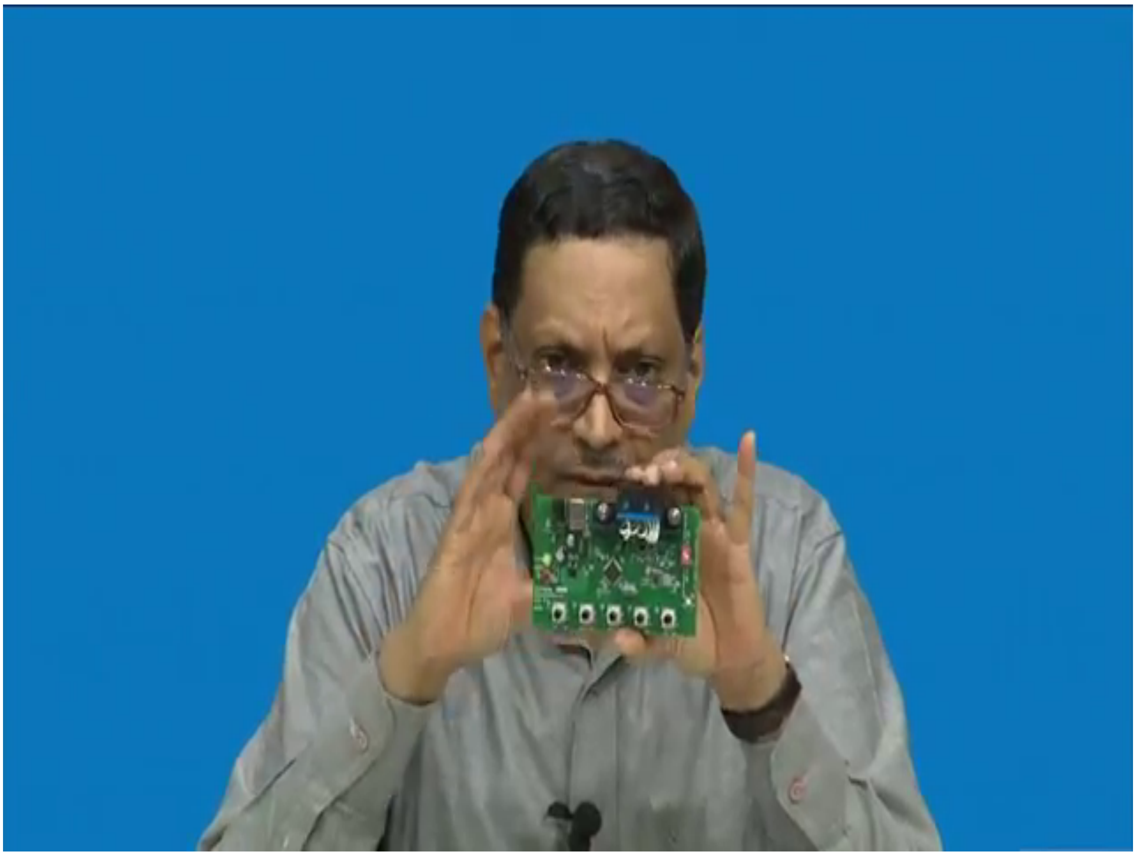
(Refer Slide Time: 23:52)



So, you can now sort of summarize the different buses which are the part of a modern vehicle. CAN gives you anywhere up to about 1 Mbit per second, LIN can give you as I mentioned not more than 20 10 to 20 kilobits per second. MOST media oriented stream transport 20 megabits per second, then flex ray you will get 10 megabit per second, then you have time triggered protocol and TTCAN and all of that. I think for completeness it is good to put 25 megabit per second.

And TTCAN also gives you time triggered CAN which gives you about 1 Mbit per second much more interesting TTCAN compare to other protocols. So, in summary these are all running inside a vehicle. We will have to see how exactly we can use these things and design your algorithm. So, where do your algorithms come in, where is the space for algorithms? The space in for algorithms is depends on what sensor you are likely to put.

(Refer Slide Time: 25:41)



For example, if you take this ECU., if you concentrate only on this ECU we have to come up with some way of detecting some anomaly. You may want to do an anomaly detection in the engine ECU.

So, and then there is the OBD port which I mentioned to you and somehow you should be able to read of this OBD codes. PIDs they are called PIDs, PIDs are also called parameter IDs. The parameter IDs are fixed as per a standard, but manufacturers can add their own PIDs as well. We can get into the detail when we go into OBD II.

But, this code for your algorithms is out in this space. If, you are indeed looking at the engine ECU; how do you connect everything back, you have to look at the powertrain. And, you have to look at something that happens in the reaction time of the order of 10 microsecond. Or, so anomaly you should detect quickly and moment the anomaly command is given to the powertrain, the powertrain system should respond in less than 10 microseconds. So, let us see how we can do this in terms of a simple example.