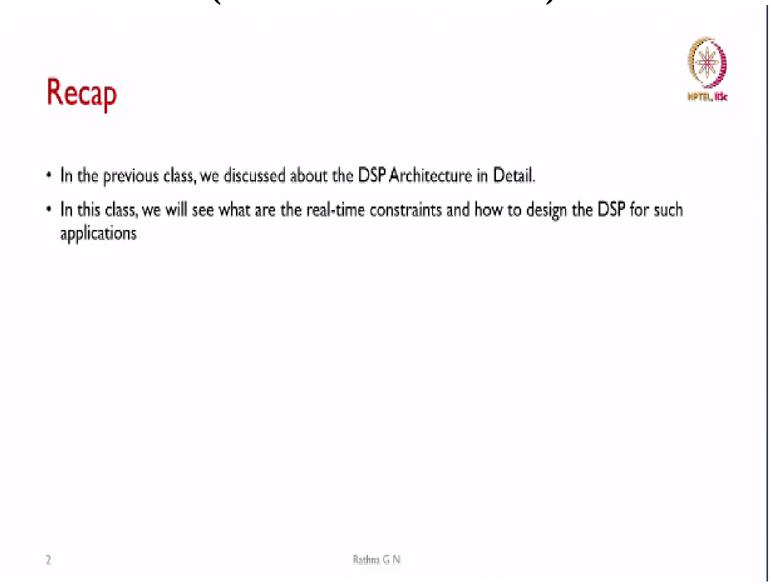
Real - Time Digital Signal Processing
Prof. Rathna G N
Department of Electrical Engineering
Indian Institute of Science - Bengaluru

Real – Time Constrains
Lecture - 07

So, welcome back to real time digital signal processing course, today we will discuss about some of the real time constraints, what we are going to have it and then how to solve and then look for the built in DSP processor or how to design our own DSP processor in FPGA? So, we will do the recap first, in the previous class we discussed about DSP architecture in detail.

(Refer Slide Time: 00:54)



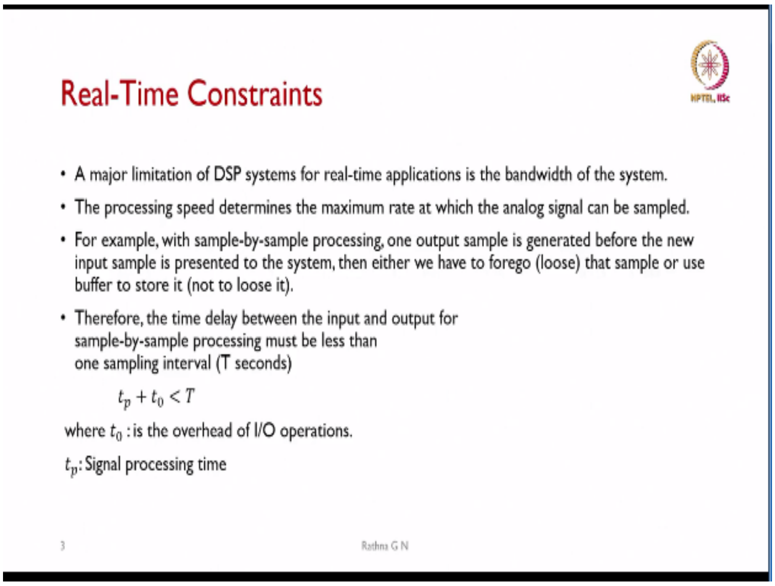
Recap

- In the previous class, we discussed about the DSP Architecture in Detail.
- In this class, we will see what are the real-time constraints and how to design the DSP for such applications

2 Rathna G N

So, hope you are comfortable with the architecture whatever we have used, in this class as I mentioned, we will be seeing some of the real time constraints and how to design the DSP for such applications.

(Refer Slide Time: 01:07)



Real-Time Constraints

- A major limitation of DSP systems for real-time applications is the bandwidth of the system.
- The processing speed determines the maximum rate at which the analog signal can be sampled.
- For example, with sample-by-sample processing, one output sample is generated before the new input sample is presented to the system, then either we have to forego (loose) that sample or use buffer to store it (not to loose it).
- Therefore, the time delay between the input and output for sample-by-sample processing must be less than one sampling interval (T seconds)

$$t_p + t_0 < T$$

where t_0 : is the overhead of I/O operations.

t_p: Signal processing time

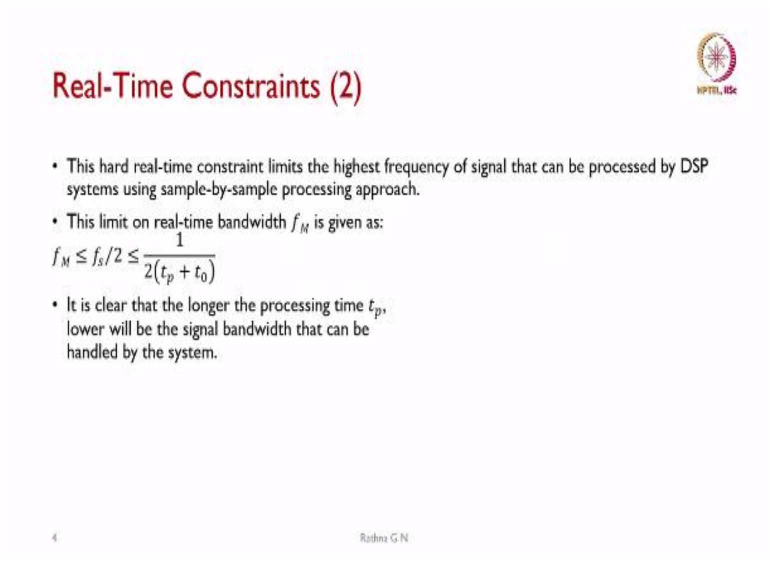
3 Rathna G N

What are the real time constraints? The major limitation of DSP systems for real time application is the bandwidth of the system. So, the processing speed determines the maximum rate at which the analog signal can be sampled. As an example, with sample by sample processing, so, if we use the interrupt driven service, one output sample is generated before the new input sample is presented to the system, then either we have to forego what we say is loose that sample.

If we are unable to compute our algorithm within that time or use buffer to store it that is if we do not want to loose any of the sample. So, we have to have a buffer to store that whatever sample which is coming much faster than our computation time. So, what it says is therefore, the time delay between input and output for our sample by sample processing must be less than one sampling interval. So, we have discussed that T seconds is my sampling time what a have it in seconds.

So, processing time t_p plus or I O overhead together should be less than our sampling period. So, as it is defined t_0 is overhead I O operations which takes that is the data which comes from input and then till the output is going to the system or wherever you are storing that we call it as t_0 will be the overhead and then t_p will be are signal processing time the algorithm which you are going to incorporate in your computation basically.

(Refer Slide Time: 03:05)



Real-Time Constraints (2)

- This hard real-time constraint limits the highest frequency of signal that can be processed by DSP systems using sample-by-sample processing approach.
- This limit on real-time bandwidth f_M is given as:

$$f_M \leq f_s/2 \leq \frac{1}{2(t_p + t_0)}$$

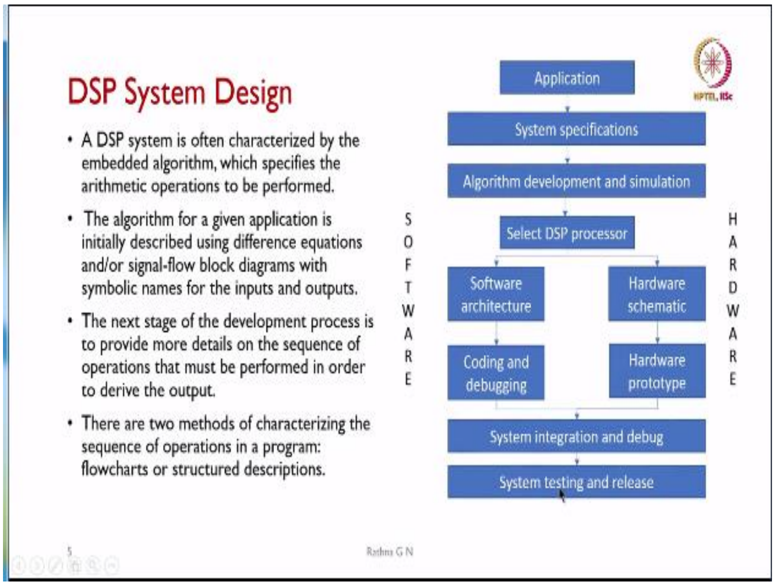
- It is clear that the longer the processing time t_p, lower will be the signal bandwidth that can be handled by the system.

4 Rathna G N

So, the other constraint what we are going to see, so once we say that it is hard real time constraint, which is going to limit our highest frequency of signal that can be processed by DSP systems using sample by sample processing approach. That means to say that, I am unable to hard real time system constrained is one such that, where I cannot offer to loose even a single clock cycle, I have to process and then my results should be ready before the next sample comes.

In that case, what happens to bandwidth, we say f_M is the maximum frequency I can allow in my input sample. So, that should be less than or equal to half the sampling rate which is driven by sampling theorem constraint, which should be $\leq \frac{1}{2(t_p+t_0)}$ that is my processing time as well as my overhead for I / O operations. So, it says that the longer the processing time t_p that is my processing time is much greater then what happens lower will be the signal bandwidth that can be handled by the system as we can see from this equation.

(Refer Slide Time: 04:28)



DSP System Design

- A DSP system is often characterized by the embedded algorithm, which specifies the arithmetic operations to be performed.
- The algorithm for a given application is initially described using difference equations and/or signal-flow block diagrams with symbolic names for the inputs and outputs.
- The next stage of the development process is to provide more details on the sequence of operations that must be performed in order to derive the output.
- There are two methods of characterizing the sequence of operations in a program: flowcharts or structured descriptions.

5 Rashmi G N

So, coming to how to do the system design that is digital signal processor system design, how we are going to take it? So, the first one is we will be seeing that DSP system is going to be characterized by the embedded algorithm which specifies the arithmetic operations to be performed. So, the next step is the algorithm for a given application is initially described using difference equations and or signal flow block diagram with symbolic names for the inputs and outputs.

The next stage of the development process is to provide more details on the sequence of operations that must be performed in order to derive the output. So, there are 2 methods, which characterize the sequence of operations in a program, one is using the flowcharts or structured descriptions one can give it. So, in this case you will be giving the structured description it derived from your application for whichever application you are designing your processor you will be specifying that.

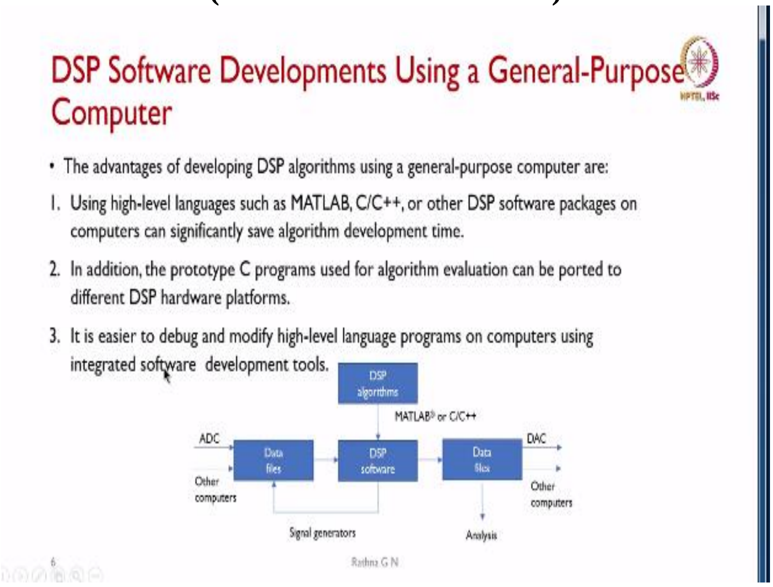
And then you have to consider all the system specifications in this case, then you will be going for the algorithm development and then initially we will be doing the simulation so that all our algorithm is working correctly, then we will be seeing that which DSP processor have to select, or which DSP processor have to design if I am using the FPGA, then we will be coming for the software architecture, how to steal this hardware, or I can generate my hardware schematic and then I can generate the hardware prototype here.

Then in the software side, we will be seeing that I had to do the coding at that debugging. So that the complete system is working as nowadays, we will be seeing that hardware and then software code design is the popular one in the present scenario so both of them will be going hand in hand if it is possible. Otherwise, we have to do them separately and then merge here in the integration system here.

So, we have to integrate both software and hardware and then do the debugging, see that my software and then hardware are going correctly according to my application, then we will be doing the system testing and then we will be releasing my hardware. So, even the system testing involves lot of tests to be conducted, it depends on how you will be for all cases, you may do the testing and some of them may be left out, because exhaustive testing is going to take a lot of time. So, mostly some of the testing will be critical once will be done.

And then we will be releasing product or for whichever application we are doing hardware, which is going to be released.

(Refer Slide Time: 07: 44)



DSP Software Developments Using a General-Purpose Computer

- The advantages of developing DSP algorithms using a general-purpose computer are:

1. Using high-level languages such as MATLAB, C/C++, or other DSP software packages on computers can significantly save algorithm development time.
2. In addition, the prototype C programs used for algorithm evaluation can be ported to different DSP hardware platforms.
3. It is easier to debug and modify high-level language programs on computers using integrated software development tools.

Rathna G N

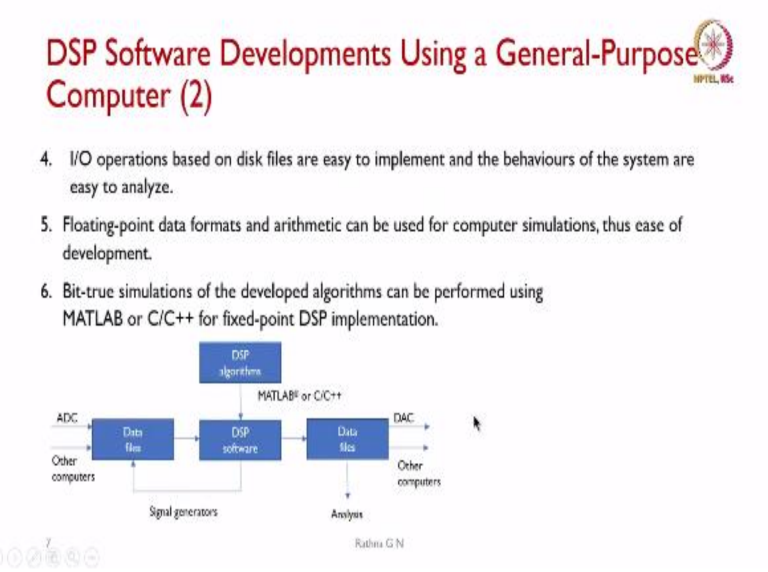
Coming to the software side. So, how general purpose processor is advantageous we will see it here. One is we know that we use the high level languages such as we can use the MATLAB or C or C++ or other DSP software packages on computers can significantly save algorithm development time. So, even nowadays, the python is the one of the popular one, one can use it and then test your prototype basically.

In addition, the prototype C programs used for algorithm evaluation can be ported to different DSP hardware platforms. So, one of the advantage of using a higher level language is it is independent of the platforms and with a little bit of fine tuning. So, we can port from one DSP hardware platform to the other DSP hardware platforms. And then the other one is it is easier to debug and modify high level language programs to computers using integrated software development tools.

As an example, you are seeing the DSP algorithm which is there, I can specify either in MATLAB or using C or C++. And then this is the DSP software, what we will be using it and data inputs for this software, which will be coming from the external ADC or any other computers what you can take it as an input and you will be storing it with that data file. And then you perform your software operations and see if you are meeting all your requirements for that particular algorithm.

So, if everything is done, so you can specify some of the signal generators back you can store it in the data file and then it can be reused. So, if the complete algorithm is satisfactorily working, then what you will be doing is you will be generating the data files, you can send it for either for analysis or if you want to have the analog output you will be converting this digital data into analog and sending it out to other hardware or you can save them as digital data in other computers also for later use if you want to have it.

(Refer Slide Time: 10:30)



DSP Software Developments Using a General-Purpose Computer (2)

4. I/O operations based on disk files are easy to implement and the behaviours of the system are easy to analyze.
5. Floating-point data formats and arithmetic can be used for computer simulations, thus ease of development.
6. Bit-true simulations of the developed algorithms can be performed using MATLAB or C/C++ for fixed-point DSP implementation.

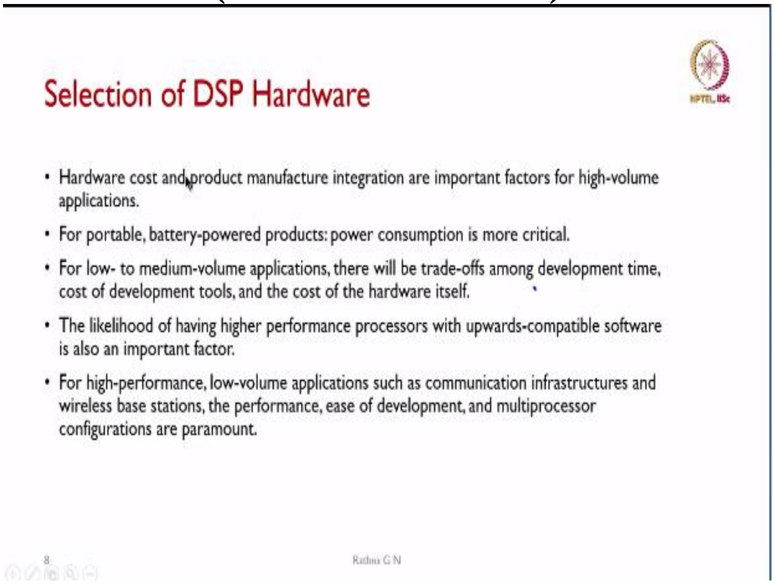
7

Rathna G N

Continuing with development of software basically on the general purpose computer, what happens to I O operations, so, it will be based on disk files. So, it is easy to implement and behaviours of the system are easy to analyze all these operations. And we know that we have a floating point data formats and arithmetic can be used for computer simulations, thus ease of development. So, as we can see, even the fixed point file system is available in MATLAB.

So, if I want to test my floating point data if it is working and then I want to convert it into fixed point and then see that my code is running perfectly, almost equivalent to my floating point operations, I can check it and then use them in my hardware development. So, we will see that we call that as the bit-true simulations of the developed algorithms, which can be performed using MATLAB or C or C++ for fixed point DSP implementation. So, algorithms will be running depending on what software I will be using it on the general purpose computer.

(Refer Slide Time: 11:59)



Selection of DSP Hardware

- Hardware cost and product manufacture integration are important factors for high-volume applications.
- For portable, battery-powered products: power consumption is more critical.
- For low- to medium-volume applications, there will be trade-offs among development time, cost of development tools, and the cost of the hardware itself.
- The likelihood of having higher performance processors with upwards-compatible software is also an important factor.
- For high-performance, low-volume applications such as communication infrastructures and wireless base stations, the performance, ease of development, and multiprocessor configurations are paramount.

8 Rathna G N

So, the next one is how we are going to decide on the selection of DSP hardware. So, we know that the hardware cost and product manufacture integration are important factors for high volume applications. So, one is concentrating on high volume so we have to see the cost and then the manufacture integration. How it is going to happen? So, we know that for portable, that is battery powered products, power consumption is more critical.

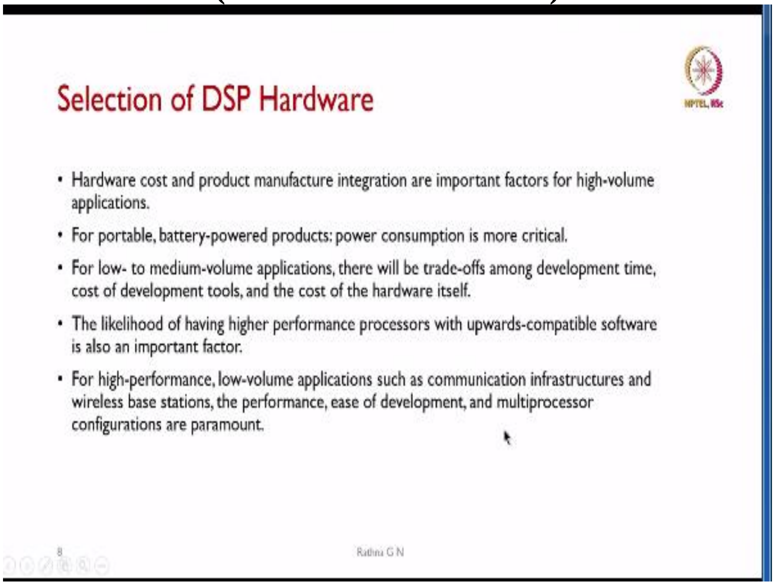
All of us use our mobiles, they have DSP processor built in them, all of us are power hungry, as you know about it, I want to save my power and I should not be charging my mobile and then I want all the features in them to be working for full day, so that I need not have to use the power banks in these cases. So, this becomes that critical component of our portable devices. And then for low to medium volume applications, there will be trade off among development time, cost of development tools and the cost of the hardware itself.

So in this case, I do not want to have high volume products to be coming out, but mine is going to be low and medium, but at the same time I have to look at it my costs should not be going up at the same time I should provide whatever features that are required at the low cost. So, the likely would having higher performance process with upwards compatible software is also an important factor.

So, one has to look at it that they have to be programmable using higher level software basically that is like either I can interface with MATLAB or I can use C or C++ programming so that one is at ease in programming. So, for the high performance or low volume applications, such as communication infrastructures and wireless base stations, the performance, ease of development and multiprocessor configurations are the paramount in this case. So, in the communication field, you will be seeing that all of us will be hungry with our network.

So, I want immediate connectivity as we connect to other person or any other devices or whatever may be the thing. I do not want to loose my time in that. So that way we say that this becomes the performance becomes the paramount in that and even the development of that and how we will be using the multiprocessor in those cases, I have to increase my hardware and then the cost may little go up, but try to see that we try to keep the cost also at a low of whatever possible.

(Refer Slide Time: 15:13)



Selection of DSP Hardware

- Hardware cost and product manufacture integration are important factors for high-volume applications.
- For portable, battery-powered products: power consumption is more critical.
- For low- to medium-volume applications, there will be trade-offs among development time, cost of development tools, and the cost of the hardware itself.
- The likelihood of having higher performance processors with upwards-compatible software is also an important factor.
- For high-performance, low-volume applications such as communication infrastructures and wireless base stations, the performance, ease of development, and multiprocessor configurations are paramount.

8

Rathna G N

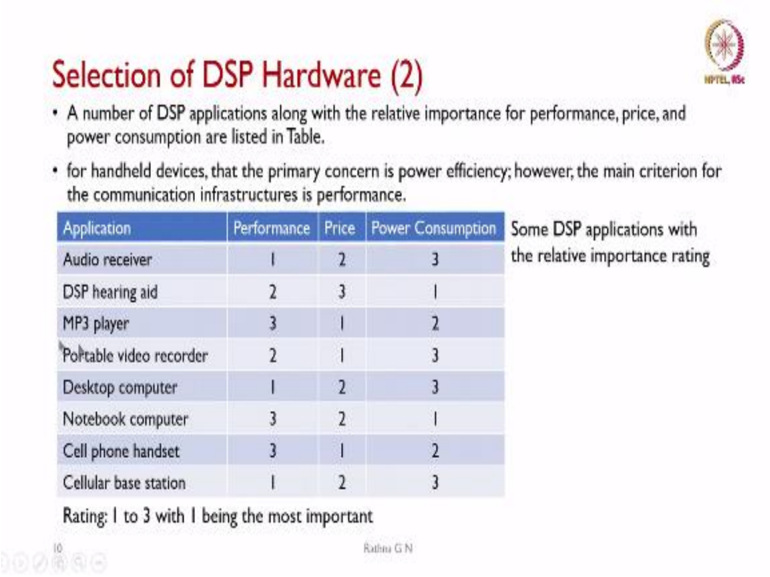
Continuing with the hardware selection, so some of the ways one processor execution speed is compared is as follows. So, usually we say it is MIPS that is millions of instructions per second, how much the processor is doing it or we say MOPS it is millions of operations per second otherwise, it is going to be millions of floating point operations per second we call it as mega flops basically or the clock rate should be in the megahertz range what we wanted or in terms of my multiply and accumulate.

I should be able to do millions of them that is MMACS what will define. So, these are the earlier comparisons one used to make it but now, it is any of the DSP processor if you go to bug link site actually, you will be seeing how the proceeds are going to be tested it is based on the application previous to application it was on the algorithms whether you want to run the filter algorithm or discrete Fourier transform what you want to run it or any other adaptive filter applications wise basically that is algorithm wise the testing used to be done.

Because each one will not have the same hardware built in, but how well they react to these algorithms was getting tested. So, but later on they found that even the algorithm level is not sufficient, it is in the application level one is going to test that how fast these applications can run in their hardware. So, you can go to the bug link website and then see that, how the processors are tested for their operations?

In addition to these low level actually, we need the other metrics to be considered that is we call it as milliwatts. So, you will be seeing that power consumption always have to be in milliwatt for measuring, what are the things power consumption that is a MIPS per milliwatt, what we will call it instead of instructions per second I want to consume based on my power or MIPS per dollar also. So, you will be seeing the cost is also coming into picture. So, these numbers provide a simple indication of performance, power and price for the given applications. So, depending on it, you will be deciding on your hardware.

(Refer Slide Time: 18:07)



Selection of DSP Hardware (2)

- A number of DSP applications along with the relative importance for performance, price, and power consumption are listed in Table.
- for handheld devices, that the primary concern is power efficiency; however, the main criterion for the communication infrastructures is performance.

Application	Performance	Price	Power Consumption
Audio receiver	1	2	3
DSP hearing aid	2	3	1
MP3 player	3	1	2
Portable video recorder	2	1	3
Desktop computer	1	2	3
Notebook computer	3	2	1
Cell phone handset	3	1	2
Cellular base station	1	2	3

Some DSP applications with the relative importance rating

Rating: 1 to 3 with 1 being the most important

10

Rathna G N

So, how will do, the selection of the DSP hardware will continue with the thing? So, we have number of DSP applications along with the relative importance for performance price and then power consumption as we have already said, so which is given in this table. So, for handheld devices that the primary concern is power efficiency, however, the main criterion for the communication infrastructure is performance. So, these are the one low end to the high end what you will be traversing here.

So, the first application we will be considering was the audio receiver. So, the performance what we want is 1 and price should be medium and power consumption should be low. So, you will be seeing the rating is 1 to 3. 1 being the most important one and the 3 is the lowest one. So, we want to have low power consumption in this. So, we know that hearing aid have DSP processor. So, what is the constraint in this the performances what we are claiming is medium and then my price has to be very low.

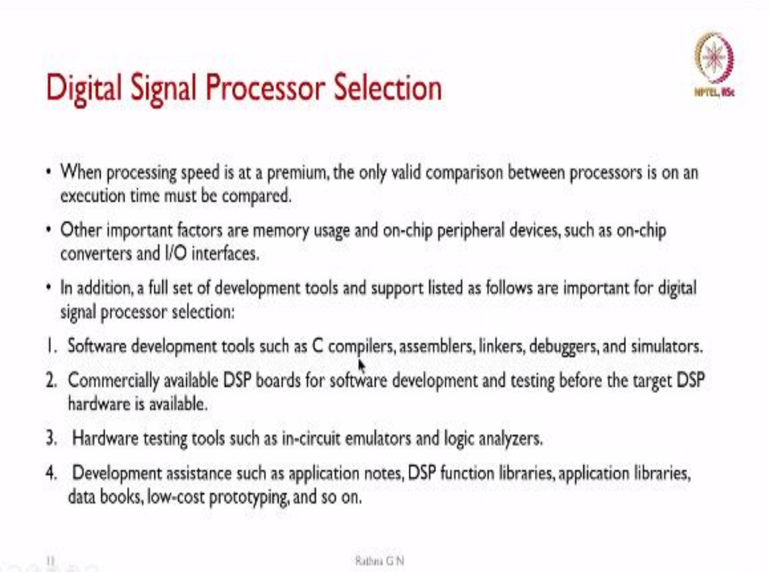
And power consumption can be little more what we say, but still most of the cases that is power consumption has to be very low in this case, because one person will be wearing it. So, if it is a middle class or low class family who is using the hearing aid, so, you can see that the battery that they have to replace in a day or in a week or in a month, what you will be calculating that, so, the next one now, I think mp3 players are getting outdated, but they are more important earlier days for all our music and other things.

Now, because of the mobile all the players are available for us in the mobile system itself. So, in this case, what we say is the performance is not such criteria and then price has to be as minimum as possible, it is very important then power consumption is in between because it is not being used continuously. So, we can take it that it can have a power consumption of medium rate. Coming with the portable video recorder one uses the thing so, which has to have the middle performance what we call it.

And then the price has to be minimum and then our power consumption has to be a little more or whatever. So, desktop computer, we know that performance is the critical part of it. And then price is a middle what we compare and then power consumption because it is connected to the mains basically so not much botheration in this case. So, you will be seeing like the notebook or cell phone handset and then last one is cellular base station, which is power hungry for your competition.

So, we say the performance rating is 1 and then price is 2 and then your power consumption is going to be 3. So, that is how we have the list says that how the DSP applications with a relative importance rating in this table.

(Refer Slide Time: 21:51)



Digital Signal Processor Selection

- When processing speed is at a premium, the only valid comparison between processors is on an execution time must be compared.
- Other important factors are memory usage and on-chip peripheral devices, such as on-chip converters and I/O interfaces.
- In addition, a full set of development tools and support listed as follows are important for digital signal processor selection:

1. Software development tools such as C compilers, assemblers, linkers, debuggers, and simulators.
2. Commercially available DSP boards for software development and testing before the target DSP hardware is available.
3. Hardware testing tools such as in-circuit emulators and logic analyzers.
4. Development assistance such as application notes, DSP function libraries, application libraries, data books, low-cost prototyping, and so on.

Coming to the processor selection. So, when processing speed is a premium, the only valid comparison between processor is on an execution time must be compared how fast because I need the algorithm my application running at utmost speed. So, then I have to see that how execution of my algorithm is going to have it being done in my processor. Other important factors are memory usage and on chip peripheral devices, such as on chip converters and I / O interfaces.

So, some of the cases as we say that memory usage is very much you will be seeing that our portable device like I will be giving example of mobile always what we want is the memory usage we want to have there is no end for asking for more memory every time. So, whatever memory you want to add on with other devices or try to save it in someplace or in cloud or whatever you will be doing the thing if it is overflowing and then whether can I have on chip peripheral devices.

So in the mobile itself, what are the things I needed to have to answer my call, I want to see my movie I want to listen to my music and then what else, lot of applications what I need it. So, you will be seeing lot of whatsapp messages, so many peripherals which are connected to this device, what you can have it and then I need the on chip converters also, because you will be seeing that we have different network providers, how will I be switching from one vendor to the other one?

If I have a BSNL sim model, know what I will call it as Airtel sim. So, we both are incorporated in the same mobile, I should be able to change or I must be able to have converters for me to change from one vendor to the other one so these are the important ones. Next one in addition to this we need full set of development tools and supports listed as follows are important for digital signal processor. Why do we need the tools because we are in the development stage?

So, we have to see that our algorithm is going to run and then our application is met and then all the testing is done. So, these are the capabilities that development tools should have. The first one we will say is the software development tools, such as C compilers, assemblers, linkers, debuggers and simulators, this is C compilers, we know that it is the highend the software what I need it so that I can have the I need not have to worry about the hardware on which I am going to run.

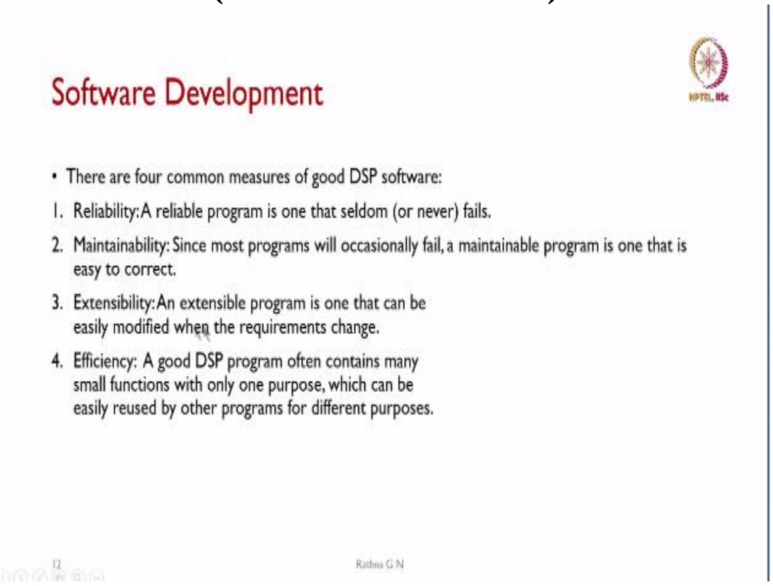
The next one is assemblers because all DSP processors, we say that assembler is equivalent almost our machine coding. So, they usually we do C to assembly conversion or one can do the programming written in assembly language itself. So, disadvantage of assembly language, it is particular to one of the hardware and this is not portable. So, one person who is writing the code or software should be knowing the in and out of the hardware or the architecture, so that he is going to use them in a better way.

The other one is linkers, because I will be using some of the libraries which are present because or if I develop my own library, I must be able to save them or combine them so that I will be working on it. So, it will be linking all the files and then next is debuggers. So, this is one of the important thing because if I have done a mistake, or if it is not running based on my intuition, then how I am going to do the debugging and see where it is getting stuck or it is taking more time.

So that, that part of the code whether I can optimize on it and why do we need the simulators, because I would not be able to go and buy all the DSP processors for my applications. So, I have to see that whether it is going to run on this platform. So, if I have a simulator, I can simulate and then check if it is doing the performance in this particular DSP processor, then I can go and then buy and then install the thing or this chip is suitable for me that is what, what I can say if it is not possible then I have to try other vendors or I have to design my own algorithm. Once I have tested it that it is working then I can go for my own design in FPGA or VLSI. The commercially available DSP boards for software development and testing before the target DSP hardware is available, that is one of the criteria what we are looking at. The other one is hardware testing tools such as in circuit emulators and logic analyzer should be able to pinpoint where the error is coming from.

The other one is development assistance such as application notes and DSP function libraries and application libraries, data books, low cost prototyping and so on are the criteria for the process selection.

(Refer Slide Time: 28:12)



Software Development

- There are four common measures of good DSP software:

1. Reliability: A reliable program is one that seldom (or never) fails.
2. Maintainability: Since most programs will occasionally fail, a maintainable program is one that is easy to correct.
3. Extensibility: An extensible program is one that can be easily modified when the requirements change.
4. Efficiency: A good DSP program often contains many small functions with only one purpose, which can be easily reused by other programs for different purposes.

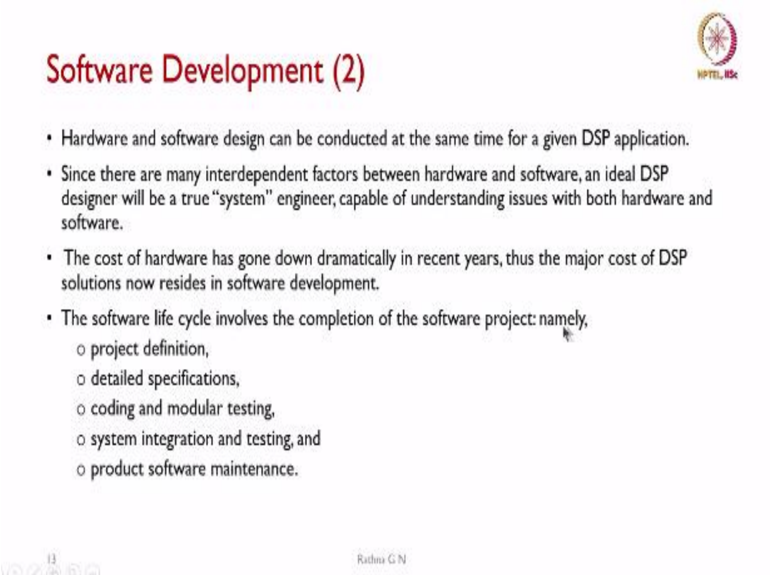
12 Rathna G N

Coming to the software development, we have there are 4 common measures for good DSP software, what are they? The first one is what we call it as reliability, the reliable program is one that seldom or we say never fails. So, that is what one of the thing. So, you will be seeing that Google never goes down or whatever may be the thing or you can say that it has gone down in once in 20 years or what you call it, I think you will be seeing even the banking system, you will be seeing that whenever they are doing the updation or something like that it goes down for 1 or 2 hours that is what you will be getting the message, but we do not want this basically happen or seldom what we call it. The next one is the maintainability since most programs will occasionally fail. We know that although it has been considered you can think of 2000 case nobody had considered that our computers will sustain so many years so, all of them Y2K problem.

So, you know that how everybody has to go and then redo their checking and then redesigning all their software and other things to take care of beyond 2000 and beyond. A maintainable program is one that is easy to correct. So, that is what the thing one needs it, so that I can modify it so that I can maintain whenever I want to change some of the things which I found out that it has got error in it. The other one is extensibility when do we say that program is extensible that is from one platform to the other platform, I can extend it.

So, whenever the requirement changes also I can easily modify so that the extension of the program is easy in that case. The other one is efficiency that is a good DSP program often contains many small functions with only one purpose which can be easily reduced by other programs for different purposes that is what, what we call it as efficiency.

(Refer Slide Time: 30:48)



Software Development (2)

- Hardware and software design can be conducted at the same time for a given DSP application.
- Since there are many interdependent factors between hardware and software, an ideal DSP designer will be a true "system" engineer, capable of understanding issues with both hardware and software.
- The cost of hardware has gone down dramatically in recent years, thus the major cost of DSP solutions now resides in software development.
- The software life cycle involves the completion of the software project: namely,
  - project definition,
  - detailed specifications,
  - coding and modular testing,
  - system integration and testing, and
  - product software maintenance.

13

Rathna G N

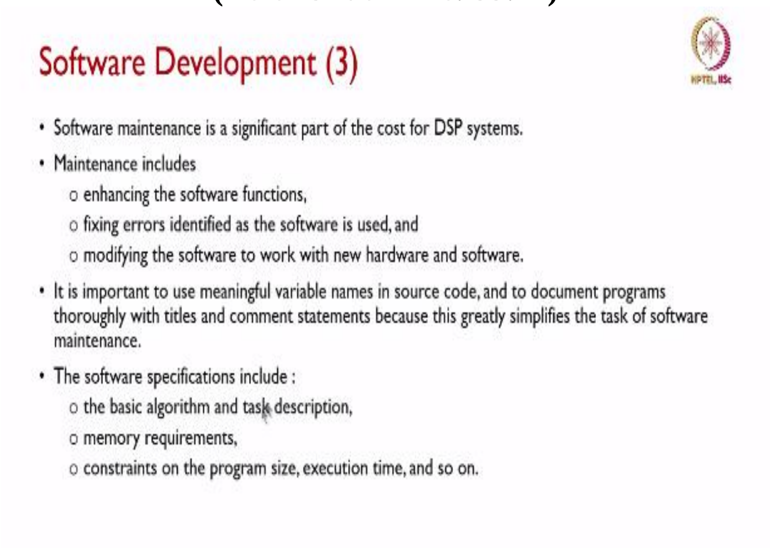
Coming further with the software development, we know that hardware and software designed can be conducted at the same time for a given DSP application, that is what I discussed when we are doing the DSP system design. So, we said since there are many independent factors between hardware and software, which restricts them actually what we call it as may not be a true system engineer should be capable of understanding issues with both hardware and then software.

So, most of the people you will be seeing that IT software industry, most of us or something we say that we are capable of solving the software but not in the hardware. So, one person as you can see that he can understand both the hardware and software he can do a real justice to the hardware which has come out so which is going to be late. So that is why we will be having a little bit of difference between the 2 so, as we know VLSI has come a long way.

So, hardware has the cost of it has gone down dramatically as you will be seeing it whatever buying of your laptop or system or your mobiles or any product for that matter. So, you will be seeing that your cost is going down but major cost of DSP solutions it resides in the software development. So, as I can point out, we used to have software cost was much higher than in 1000 dollars earlier for our emulators that is JTAG interfaces compared to around system which was in 400 dollars or something like that.

The software lifecycle involves completion of the software project. So, namely that is project definition, detailed specifications one has to have it how you are going to do coding and then you will be having the coding also you can do modulararly so that testing becomes easy. The next is system integration, all of them have to be your hardware and then software has to be integrated. And finally, how you are going to test it and later on you will be seeing that your product software maintenance is one of the important criteria in your software development.

(Refer Slide Time: 33:22)



Software Development (3)

- Software maintenance is a significant part of the cost for DSP systems.
- Maintenance includes
  - o enhancing the software functions,
  - o fixing errors identified as the software is used, and
  - o modifying the software to work with new hardware and software.
- It is important to use meaningful variable names in source code, and to document programs thoroughly with titles and comment statements because this greatly simplifies the task of software maintenance.
- The software specifications include :
  - o the basic algorithm and task description,
  - o memory requirements,
  - o constraints on the program size, execution time, and so on.

14 Rathna G N

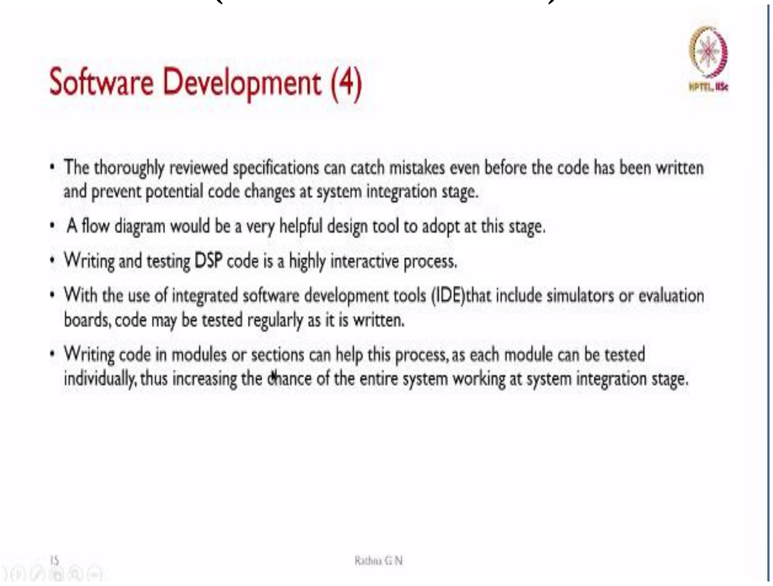
So, continuing with this, maintenance is a significant part of the cost for DSP systems, how we are going to do that what are the things that are there in maintenance? Basically, one is enhancing the software functions, fixing errors identified as the software is used, as you will be seeing even in your mobile you will be getting updates for your software. So, from one stage to the other one within one year you may be having multiple software updation which is going to come so you should be able to have more functions incorporated.

And then if there are bugs in the earlier cases, how you are rectified and then identification and then how you will be updating them in your next software? And then modifying the software to work with new hardware and then software. So, you will be seeing that iPhone has its own operating system, whereas Android has its own operating system, but the devices may be from different vendors basically. So, how the android platforms will be having the same software which is running on different platforms?

So, it is important to use meaningful variable names in source code and do document programs thoroughly with the titles and comment statements because this greatly simplifies the task of software maintenance, it is not just writing and then leaving it like or solving own problem or a math problem, we have to document how it has been done. So, it will be easy for someone else to take over and then continue from that, so that it need not have to be repeated.

So, the specifications for the software include the basic algorithm and task description and how much of memory I need it and constraints on the program size, execution and time and then so on what you can keep on including it these are the major ones, rest of it will be coming as and when you will be developing it.

(Refer Slide Time: 35:40)



Software Development (4)

- The thoroughly reviewed specifications can catch mistakes even before the code has been written and prevent potential code changes at system integration stage.
- A flow diagram would be a very helpful design tool to adopt at this stage.
- Writing and testing DSP code is a highly interactive process.
- With the use of integrated software development tools (IDE)that include simulators or evaluation boards, code may be tested regularly as it is written.
- Writing code in modules or sections can help this process, as each module can be tested individually, thus increasing the chance of the entire system working at system integration stage.

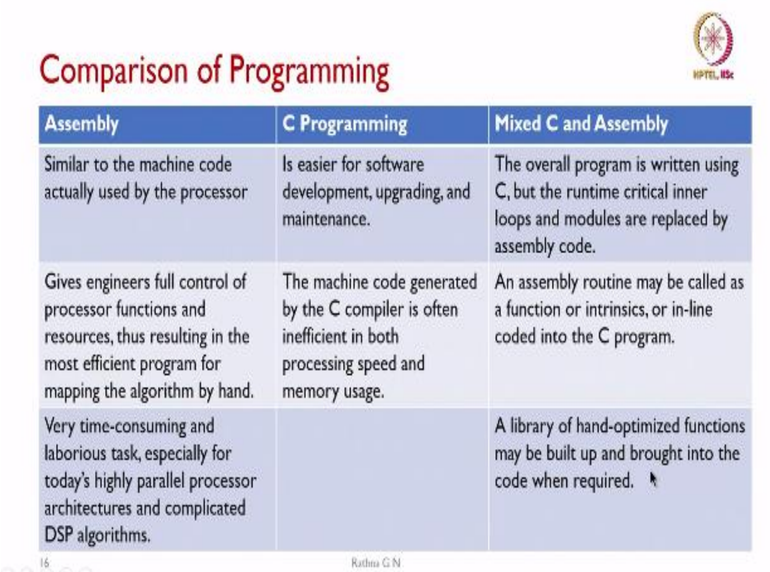
15 Rathna G N

So, the next one in the software development is we have to thoroughly review specifications or to catch the mistakes even before the code has been written and prevent potential code changes at system integration stage this becomes a little cumbersome so it is better to see that all the errors are fixed much earlier. And as always, I mentioned to students nowadays people would not write the flow diagram they directly go and then implement their algorithm in the software or whatever and then it becomes little tough to see that where the error has occurred.

If a flow diagram is prepared, it will help in design tool to adapt at any stage. So, which flow has gone bad, whether I am flowing in the right direction or wrong direction, I will be knowing about it. And writing and testing DSP code is highly interactive process, one has to discuss and then do the thing. So, with the use of integrated software development tools we call it as the IDE tool that includes simulators or evaluation boards, code may be tested regularly as it is written.

So, we will be seeing this, how we will be using the software exclusively in DSK boards, how we will be testing, how the input is coming, how the output is going, how are algorithm is running? We will be looking into that and writing code in modules as I have already mentioned, or sections can help this process because debugging base becomes easier which part of the section or module is not working. So, we can fix go back and then fix that and input and output to those modules are coming correctly or not.

(Refer Slide Time: 37:37)



Comparison of Programming

Assembly	C Programming	Mixed C and Assembly
Similar to the machine code actually used by the processor	Is easier for software development, upgrading, and maintenance.	The overall program is written using C, but the runtime critical inner loops and modules are replaced by assembly code.
Gives engineers full control of processor functions and resources, thus resulting in the most efficient program for mapping the algorithm by hand.	The machine code generated by the C compiler is often inefficient in both processing speed and memory usage.	An assembly routine may be called as a function or intrinsics, or in-line coded into the C program.
Very time-consuming and laborious task, especially for today's highly parallel processor architectures and complicated DSP algorithms.		A library of hand-optimized functions may be built up and brought into the code when required.

16 Rathna G N.

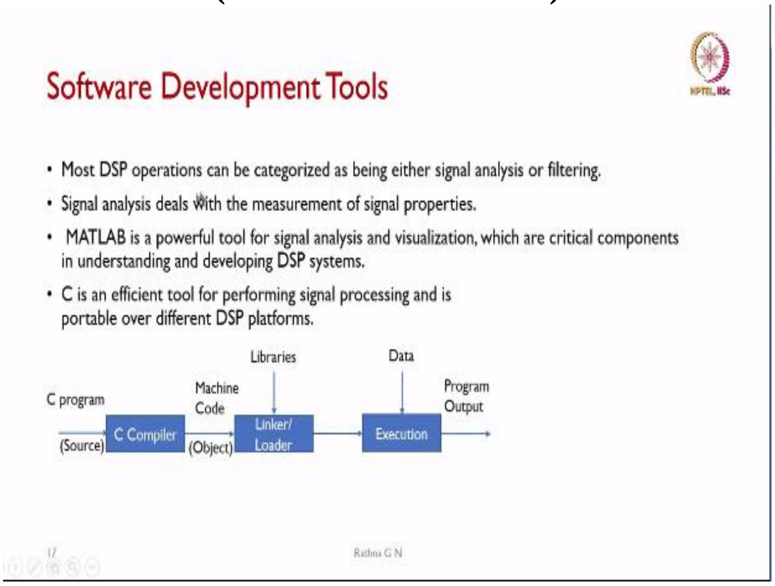
So, coming to the comparison, we know that already we said we have different programming parts of it. So, one of the DSP processor popular one in the earlier 80s and 90s was assembly programming. So, it is a as I have already mentioned, it is equivalent to machine code actually used by the processor and this gives engineers full control of processor functions and resources, thus resulting in the most efficient program for mapping the algorithm by hand.

This is the disadvantage what I have to say very time consuming because you should understand how one assembly programming runs and laborious a task it is going to be especially for today's highly parallel processor architectures and then complicated DSP algorithms it becomes an hindrance, so it is better to go we will come to the mixed C and assembly nearby, what is the advantage of C programming? We know that it is software development upgrading and then maintenance is easy.

And machine code is generated by the C compiler in this case, often what happens in this case it is inefficient in both processing speed as well as memory usage, which will be much more, we will see this in the lab. I will demonstrate how you can compare between just one addition what is going to happen. Whereas I can use both C and then assembly and then where I am going to use we will C the thing, overall program is written using C actually.

But the runtime critical inner loops and modules are replaced by your assembly code. So, routine may be called as a function or intrinsics we call it or in-line coded into the C program. So, you will be having the library of hand optimized functions may be built up and then brought into your code when required.

(Refer Slide Time: 39:49)



Software Development Tools

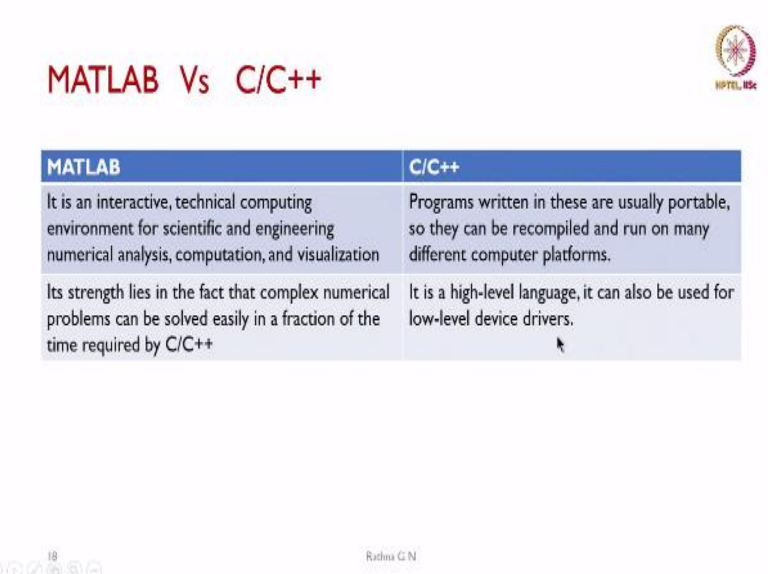
- Most DSP operations can be categorized as being either signal analysis or filtering.
- Signal analysis deals with the measurement of signal properties.
- MATLAB is a powerful tool for signal analysis and visualization, which are critical components in understanding and developing DSP systems.
- C is an efficient tool for performing signal processing and is portable over different DSP platforms.

So, coming to other distinct development tools, we know that either signal analysis or filtering what we will be doing in the DSP operations. So, we know that signal analysis deals with measurement of signal properties and MATLAB is powerful tool basically all of us know for the analysis, visualization. So, which are the critical components in understanding and developing DSP systems, even in my regular course, I told my students so check using MATLAB and later on put it onto the DSP processor.

So that you can compare with the whether you are getting the same results. Any design, we do FPGA or any of the coding, we do the thing, we first tested in the MATLAB so that our algorithm is running correctly, then we will go to our hardware implementation and see that we are almost nearer to whatever MATLAB gives. So, we say C is an efficient tool for performing signal processing and is portable for different DSP platform what we have said.

Then what happens, my input is going to be a C program, which is a source code I will be giving it so I have to do use the compiler here. And then generate the object code, which we call it as machine code or assembly code, basically. And we will be linking with the libraries and then we will loading the code onto the hardware, then the data whichever has to be input, it is taken from the input and then we will be doing the execution here. So, I will be getting the program output here.

(Refer Slide Time: 41:32)



MATLAB Vs C/C++

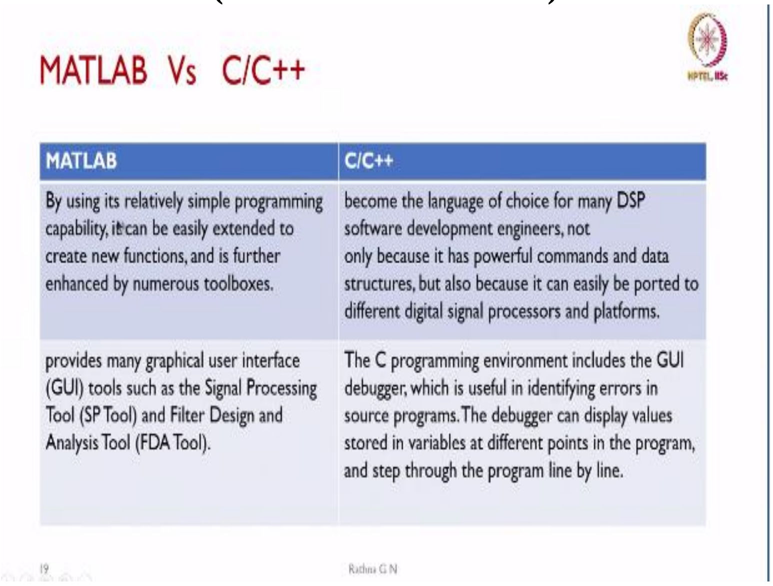
MATLAB	C/C++
It is an interactive, technical computing environment for scientific and engineering numerical analysis, computation, and visualization	Programs written in these are usually portable, so they can be recompiled and run on many different computer platforms.
Its strength lies in the fact that complex numerical problems can be solved easily in a fraction of the time required by C/C++	It is a high-level language, it can also be used for low-level device drivers.

18 Rathna G N

Coming to the comparison of MATLAB versus C, C++, we will see. So, we know that it is now MATLAB is an interactive technical computing environment for scientific and engineering, numerical analysis, computation and visualization. So, its strength lies in the fact that complex numerical problems can be solved easily in a fraction of the time which is required by our C or C++. And then program written in these are usually portable.

So, they can be recompiled and run on many different computer platforms. And it is a high level language, it can also be used for low level device drivers, you know that all kernel and other things are written using C and C++ nowadays, you will be seeing with python also.

(Refer Slide Time: 42:24)



MATLAB Vs C/C++

MATLAB	C/C++
By using its relatively simple programming capability, it can be easily extended to create new functions, and is further enhanced by numerous toolboxes.	become the language of choice for many DSP software development engineers, not only because it has powerful commands and data structures, but also because it can easily be ported to different digital signal processors and platforms.
provides many graphical user interface (GUI) tools such as the Signal Processing Tool (SP Tool) and Filter Design and Analysis Tool (FDA Tool).	The C programming environment includes the GUI debugger, which is useful in identifying errors in source programs. The debugger can display values stored in variables at different points in the program, and step through the program line by line.

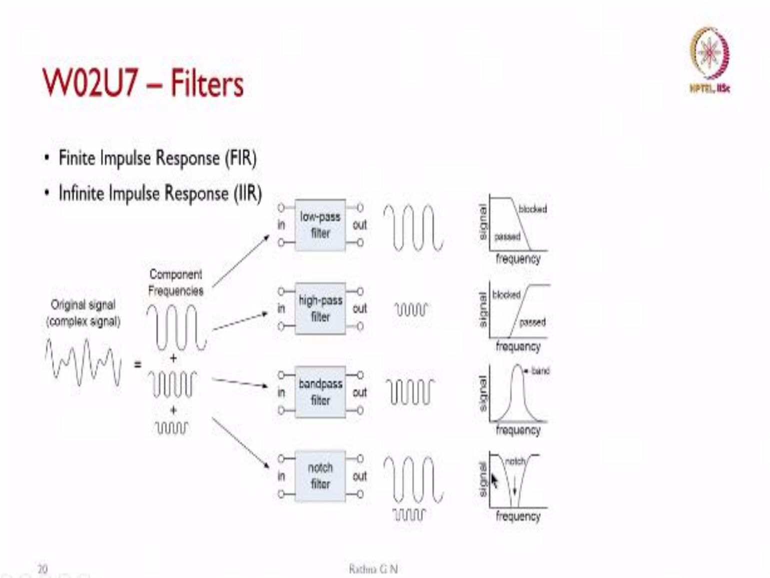
19

Rathna G N

The other advantage of model MATLAB is by using it is a relatively simple programming capability, it can be easily extended to your create a new functions and is further enhanced by numerous toolboxes. This is you will be seeing every year there will be new toolboxes introduced or updated basically in the tool box itself so that you have whatever is required at your tip of fingers, that is what I will call it.

And then they have the GUI tools to support whatever design you are doing it and then for the signal processing tool box, we have the SP tool box or FDA tool box for the filter design, analysis a toolbox so we will be using the FDA toolbox but nowadays they are naming it in 2020 and above as a filter design tool box basically so still this is supported so we will use them. The other one we say that C, C++ is a high level language and usually used for many DSP software development engineers not only because it has powerful commands and data structures, but also because it can easily be ported as we have been telling from the one we started considering the C. It includes even GUI you know that C++ debugger, which is useful in identifying errors in source programs. So, it can display values stored in variables at different points in the program and step through the program line by line.

(Refer Slide Time: 44:22)



W02U7 – Filters

- Finite Impulse Response (FIR)
- Infinite Impulse Response (IIR)

So, which will be taking it up in the lab and see so, this ends the constraints when we are designing the DSP processor for real time applications. So, in the next class, we will see that our module second, we will continue that how the algorithms for filters are going to be built around? So, we will be considering both finite impulse response and infinite impulse response, so we have some input. How I want to extract these components using different kinds of filters? It can be a low pass filter, high pass, bandpass and bandstop or knotch filter to eliminate one of the frequency and how you will be getting the output and their responses is shown. So, more in detail we will cover it in the next class. So, happy learning and then thank you for listening to this lecture.