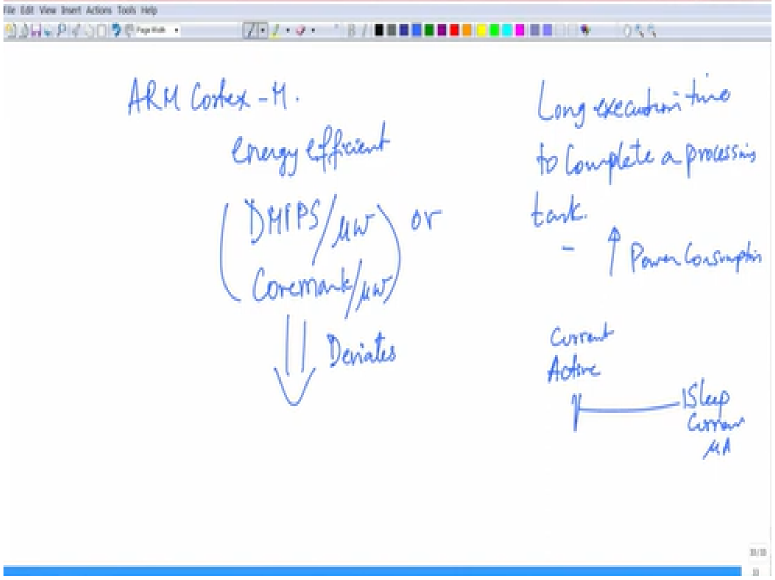
Design for Internet of Things

Prof. T. V Prabhakar
Department of Electronic Systems Engineering
Indian Institute of Science, Bengaluru

Lecture - 17
Power Management Systems – 02

(Refer Slide Time: 00:28)



So, one of the major benefits of using ARM Cortex processors, you are looking at ARM cortex processors in fact Cortex M series processors is the fact that they are highly energy efficient. They are indeed highly energy efficient. And just talking about the energy efficiency we even express them in terms of the DMIPS per microwatt or sometimes you express it in terms of CoreMark per microwatt. These are important.

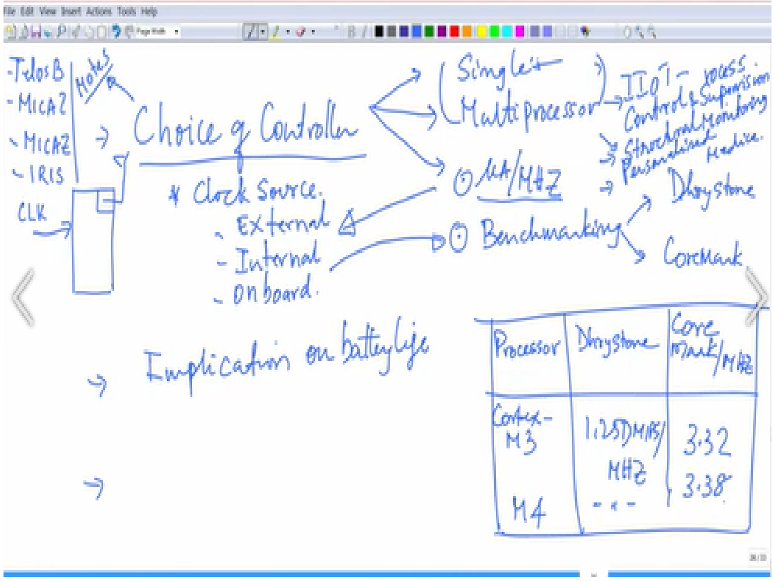
See, these two things clearly deviate from what does it mean? It deviates from the fact that you are focusing only on active current, sleep current you are not saying any more about just that which was the micrograms per megahertz that we knew. That is all gone now. You should consider the D MIPS per microwatt or the Core mark per microwatt which is much more important as a metric. So, point is just this.

For example, if you have many microcontrollers, they might need long execution times to complete a process. So, let us look at it, long execution time to complete a processing task that means you just taking too long. So, what will happen? This will increase the overall power consumption, surely it will. So, you must look at energy efficiency which is a very important aspect also. Now, controllers will have to, rather newer controllers, are actually trying to make bring the gap between, let us say, the active current and the sleep current values were far apart.

Sleep current, I will write it far apart. This distance, this may be in milli ampere, this may be in micro amps and sometimes even in usually, it is in micro amps. Now that is coming down, because this guy is going closer. You see now this is active current. That is fantastic. Newer controllers, they are pushing it to the right. So, that you are going closer, you are reducing the gap between the active current and sleep current.

So, that is a very important change that is happening in the choice of controllers. This is one part of the story folks. We will have to look at this discussion even more in broader detail. But before that, we must complete the story of where we started. If you look at this picture, we started here of choice of controller.

(Refer Slide Time: 04:52)



Processor	Dhrystone	Core Mark/MHz
Cortex-M3	1.25DMIPS/MHz	3.32
M4	- " -	3.38

We went and did quite a few things and then came back. I want to summarize with this table that we see here. Look at Cortex M 3, it is1.25 D MIPS per megahertz, CoreMark is 3.32 megahertz. M 4 is the same as the Dhrystone is the same benchmark, but it CoreMark per megahertz is 3.38. This is what people report, but do not forget, I meant what I mentioned to you on how the newer CoreMark is to be mentioned not just in terms of megahertz.

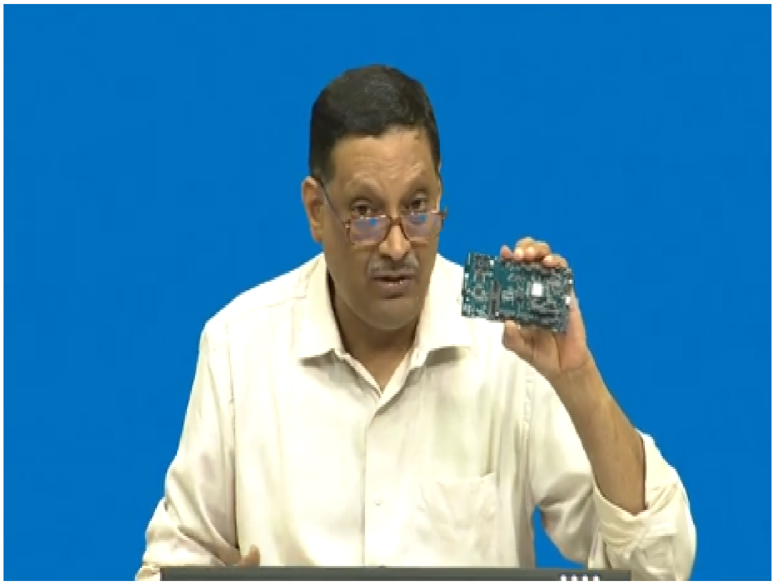
But also, in terms of power consumption which is the micro watts. This is the most important aspect of what I wanted to drive home to you. Now the discussion continues to be incomplete because we have not touched upon this multi processor in detail. I will touch upon this multi processor and then I will close this discussion. Then later we will take up the question of implication in battery, battery life, implication of all what we are saying in terms of battery life.

So, let us move on to understand multi processors. So, folks if you are tired of listening to me, let me show you are demonstration. In fact, I will not demonstrate. I will show you a picture of what we have in the labs. And maybe that will help you to connect. Look at this picture here.

(Refer Slide Time: 06:48)

Look at what I have in my hand are the single Cortex M three controller, single controller, small little one, has some ports here. Through some flexible cable I can connect a few sensors and so on. And I can do some sense and send and I can run it and you know, ensure that lifetimes of these systems are quite close to the 10 year kind of lifetimes that we are talking of.

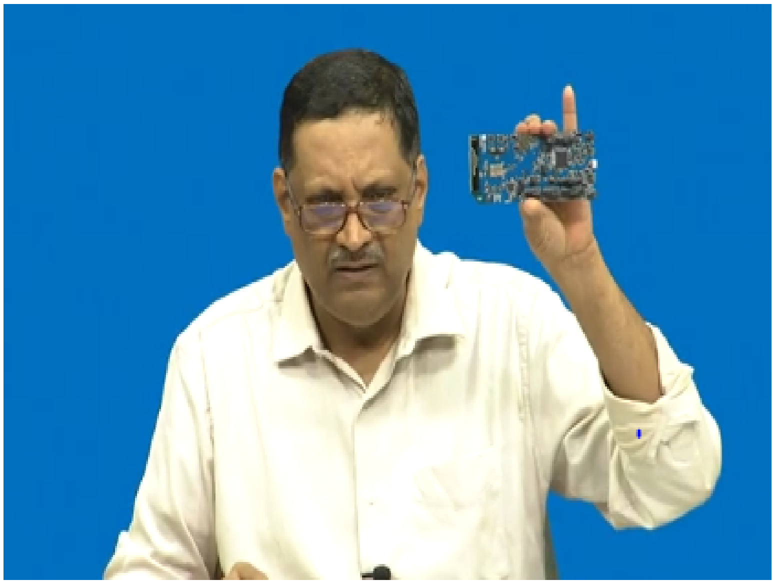
(Refer Slide Time: 07:16)



But then I also have this - I am not sold to Nordic. But we have been buying Nordic process in the lab. So, because of our comfort of using Nordic processors nRF 5340, if you go and Google this further, that is this board. If you go and look up this board, you will realize it has two processors inside. You may now ask, is this meant for beyond the sense and send, applications? Yes, indeed. It gives you good performance.

All the conflicting results requirements I was talking to you about particularly the CPU resource is out. Because you are one CPU doing one part which is perhaps high frequency sampling. And there is other CPU which is just looking at the communication interface. Now the question is how do you make the two processes work together? And that is an IP which nobody is going to tell you, how they end up making these things work.

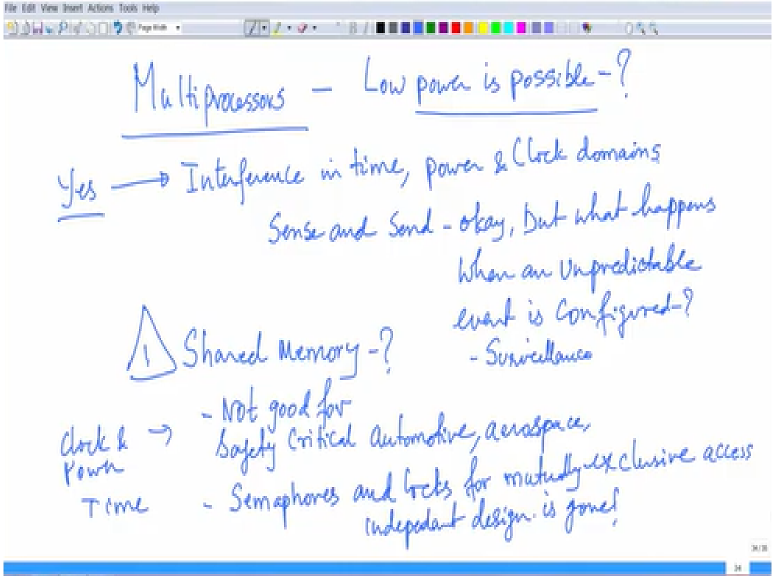
(Refer Slide Time: 08:35)



Also has moved the world has moved on to have on these kinds of development boards, these are development board. This is based on Nordic 9160; this is a Nordic 9160 kit and this board is quite advanced, because it can essentially connect to the internet over a narrowband IoT connection, NB IoT connection. So, we got this you can put a SIM card and you can take the service of an operator.

And you can connect sensors to it directly over cellular infrastructure, terrestrial infrastructure you can start uploading data or you can have a cluster of sensor nodes which communicate to this central system is like a gateway. This in turn communicates to the outside world using the SIM connection that we have. So, things are moving forward folks. And I am sure you appreciate that these are important developments happening going away from the Talas B and the Iris and the Mica and the Mica Z modes to something which is a lot more advanced various multi processors.

(Refer Slide Time: 09:58)



Multiprocessors - Low power is possible -?

Yes → Interference in time, power & Clock domains

Sense and Send - okay, But what happens
When an Unpredictable
event is Configured-?
- Surveillance

Shared Memory -?

Clock & Power → - Not good for
Safety Critical automotive, aerospace,

Time - Semaphores and locks for mutually exclusive access
independent design is gone!

So, let us discuss a little bit about multiprocessors. And also, the hard problems that are there in using a multiprocessor yet maintaining ultra low power. That brings us to this - what I wrote on the screen here. Is low power possible at all is the question? The answer is a big yes, provided you remove the interference in time power and clock domains this is very important. That is, you are talking of separating the two processes in terms of time in terms of power and also in terms of clock domains.

That is where the whole issue of sense and sense applications are very simple. But the same kind of controller which is used in sense and send, if it is now used for let us say surveillance or even tracking. These systems cannot respond to unpredictable events, that is the problem. That is why you must motivate yourself to move towards the multiprocessor environment. If that is indeed the case, how do multiprocessors communicate among themselves?

Well, very well known is the shared memory architecture. The shared memory architecture is something that is been used for several decades now. And the energy consumed grows by the amount of useful task execution that is performed systems use different power and clock domains. So, from an energy perspective, coupling of power and clock domains is important. That is okay. But then shared bus couples these domains - the power and the clock domains there is a shared bus.

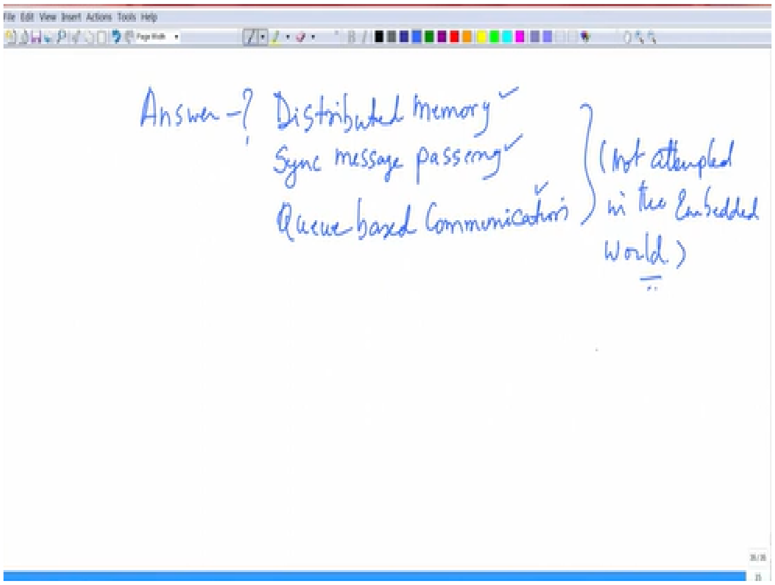
And this shared bus coupling these two power and clock is not at all good for safety critical automotive applications or automotive or even aerospace applications. So, that is a big problem . That is one thing. So, which essentially means the principles of separation of concerns and isolation of independent functionalities are actually violated. And power management requires interaction with many hardware and software layers here that is the problem.

Now, as a rule, unnecessary interference and dependencies introduced on the hardware layer can rarely be decoupled by means of any other layer software layer that comes into picture. So, in a way the existing mechanisms of shared memory are not good in terms of separating the clock and power. What about interference in time domain? Well, that is true, you can use semaphores and blocks for mutually exclusive access to resources.

But the problem with that is you lose your independence in design; independent design is a goal. So, the issue of addressing multi process in the market even for manufacturers is not a straightforward thing. They count how much can they bring in the separation between time power and clock domains is indeed the real crux of the problem. And you have to study the datasheet of any multiprocessor systems in careful and great detail from this perspective.

This is important. Now, is there a solution at all is a question the answer is yes, why not? There are new things that have come up not in the embedded domain, but well known in the distributed computing world.

(Refer Slide Time: 13:59)



Answer -? Distributed memory ✓
Sync message passing ✓
Queue based Communication ✓ } (Not attempted in the Embedded World)

Where people talk about distributed memory, synchronized message passing, queue-based communication and so on. All these are not really attempted in the embedded world. So, now you see, you can open up a new area, of looking at, can I take two commercially available processors which are heterogeneous by nature. Think about something like Atmel processor and Nordic processor and you are putting them together.

You have to manage to somehow connect them together. That means you have to build in that intelligent layer which does synchronize message passing for instance. Queue based communication or any one of the things you have to read them in detail understand what kind of construct you had like to do between the two paths to sort of connect them together. And that becomes a very interesting piece of hardware which could be used in industrial applications.

And we listed the several industrial applications in the past and I will connect you back. What were the multiprocessor applications? Let me read out. Industrial IoT applications, control and supervision, process control and supervision, P is gone for some reason I will put it back. Structural monitoring, personalized medicine, all these systems can use potentially multiprocessor environments. Understanding the benchmarking part, we said there are two.

We said about Dhrystones and also, we mentioned about the Core mark, we must go and have a look at the core mark website because there is a wealth of information and that will help you to understand even further on performance of a controller or an SOC. What did we say about an SOC? SOC has peripherals, has the core CPU and the CPU has sleep modes and active modes, all are there. Now you should be able to pull out these components.

And say all the peripherals how do those perform? How does the CPU perform? How does the CPU - what is the kind of power consumption that the CPU utilizes when it is in deep sleep? All this performance related issues are there in the website. So, let us go to this website and have a look.

(Video Starts: 16:27)

I will just give you an overview. This EMBC actually stands for Embedded Microprocessor Benchmark Consortium. And spend enough time understanding this in great detail because all about core mark is out there and what actually the as far as power conserve power consumption is concerned. If you are interested there which is of interest in this course is ULP Mark suite offers a CoreMark variant.

That clearly defines how to measure and record CoreMark power in a uniform way making comparisons a snap. So, let us look at this ULPMark. So, if I go here to this website, it simply connects to a ULPMark. Again, you read these things I will tell you about the benchmark quantifies the many aspects of ultra low power MCUs and so on. If you read it carefully, you will find these three are very interesting.

This is the ULPMark family of Benchmarks. ULPMark core profile is true energy cost of deep sleep. Now, ULPMark peripheral profile common peripherals energy impact on deep sleep. The third one is ULPMark Core Mark, active power using CoreMark as the workload. So, he has defined in this here it says they define what is known as a workload. So, now, let us see as I mentioned CoreMark is very specific to CPU.

(Video Ends: 18:07)

This is the key takeaway. Unlike Dhrystone which was heavily dependent on the you know; you could tweak the compiler related. So, that is not the case here, it is completely CPU related.

(Video Starts: 18:20)

So, let us look at this ULP-Core-profile. What is this Core profile tell you? It tells you some interesting things, this is an interesting site. So, you must read this very carefully because this picture that you see is what we are going to use heavily in our battery life calculations. What do you have on the x axis? Time. What do you have on the y axis? Energy. Often, we plot not energy, but current we plot the current, current versus time.

So, when you get current versus time and you know the operating voltage you can easily calculate the energy, here has abstracted that. So, here has given you one number and he is plotting energy consumption over time. He has given you this. But whenever we do a measurement in the lab, it is usually i along the y axis time along the x axis then you calculate each of these peaks nicely and find out what is the area under this peak then you calculate the area under this section, add that.

So, take this, calculate the area of this plus area of this plus area of this plus area you will get one number. That number you will take and use it for your further operations. That is the core idea behind this picture. And that is also the reason why you must visit this website to get a very clear view. Let us read it, it says you will be ULPCoreMark profile focuses on MCU score, specifically energy cost in sleep and the transition to and from active mode. Folks, do not forget this.

(Video Ends: 20:16)

I mentioned several times in this course that if the controller is sleeping and you give it a timer interrupt so that it wakes up. It is going to take a huge spike of current. We mentioned that that spiky nature of current depends on the type of oscillator that you use. If it is a crystal oscillator, remember, the oscillator takes a long time to settle down and current consumption is quite high. However, the clock is stable.

Whereas if you use internal RC oscillator it might be fast, it cannot support very high frequencies, maybe the power consumption is also low. So, now design means you will have to think again where you want to apply internal oscillator or where you want to apply the external oscillator. So, anyway coming back to the focus here.

(Video Starts: 21:15)

This typically could be because of that energy consumed here could be because it has woken up. It has woken up, done some activity and gone back to sleep and then again it has come back to wake up state, it is active now. It has consumed some activity and gone back to sleep. Now what all are possible in this active state is the question. You wake up the crystal oscillator, you apply a timer, you wake up the controller it comes back to full frequency of evaporation.

Clock is fed to several of the peripherals, clock is ready now. Clock is available, you may do sense little process and send, sense process send, sense process send that means send means what radio. Radio is also shown here, it will also be shown here in this active. So, wake up, compute, sending all that will do then it will go back to sleep, again the same thing will happen. This can be equated to something like a periodic sensing of the environment.

It could be like temperature; it could be like humidity and so on. Something which does not change drastically, but something that you need periodically, someone data point you need periodically. Now, this website is so critical. It tells you so many things. If you look at the down sentence here, he writes, while the active portion of the Benchmark is only running roughly 3% of the total run time, it requires data to be saved during deep sleep through the use of retention RAM.

Since it is rare that the sleepy edge node would clear its RAM after every sleep cycle. The exit and enter costs of retention RAM illustrate true energy costs of sleep modes. There is more of it than just a datasheet number. Amazing sentence, I will tell you why? Go back to your Nordic processor. There are several modes, which you can put the Nordic processor to low power modes of the Nordic processor.

One of the Nordic processor mode; you might have observed is with RAM retention. It is there, please read the datasheet - it is out there. And it is important you do that homework. And he is actually referring to that here. He is actually referring to that with the RAM retention. So, that is out there. So, this RAM retention is also and he tells you that it is not that. So, that means there is the RAM which is kept alive.

You cannot say I am going to switch out the RAM. Because it has sense put that stuff in whatever sense value is sitting in the RAM. So, he says it is, since it is rare that the sleepy edge would clear its RAM after every sleep cycle. The exit and enter costs of retention RAM illustrate the true energy cost of sleep modes. You cannot afford to clear the RAM, so it is out there. Now during the active portion of the test, what all he has done to Benchmark is the question.

I gave you an idea of crystal on. Then sense then transmit, sense and send. I have given you that idea. But when you are Benchmarking, that is not what you should be doing. What you should do is these things, these things is what you should do. This is the key difference between benchmarking and thinking that you want to use it for some application. Your application could be one thing, your benchmarking is important. So, what should you do?

Generate 20 GPIO pulses, perform an 8 bit linear interpolation, perform 16 bit integration, compute a seven segment LCD binary conversion, search for a substring in a string, perform a small bubble sort, permute the bits of a string based on input and previous state, do all these. These are the tests that you have to put. Take your multiple processors that are in front of you. Use those multiple processors, run these tests.

If these tests are done you can generate a number which you can corroborate on this website and use those numbers rather than worrying about your application. If you do this your application will automatically be energy efficient. You do not have to do you know crystal oscillator start and you do not have to do all of that. That is all be, it will be amortized into these tests. Moreover, you are more interested in the proper comparison between different curves.

Because our whole exercises, how do I choose a controller from a low power perspective from a performance perspective. So, you have to do these things much more importantly. So, I hope that part is clear. So, I will now show you that there are other possibilities of peripheral profile that means, it takes into account several other peripherals of an SOC and it gives you the slot here. For example, ADC is important.

So, conversion rate is one kilo hertz then you have the frequency which is 32768 hertz that is 32.768, the low frequency crystal oscillator which is out there and so on. So, all the other related you know, oscillators and frequency evaporation is also mentioned here. For example, if you look at nine, they use the one megahertz frequency here, it is all 32768 kilohertz and here it is the one megahertz frequency.

So, all the active parts are also mentioned here. And so, are the parts related to you know, the core controller also. So, let me go back and show you the third one is here. Yeah, I must go back here. You see the ULPCore, ULPCoreMark, you can obtain the benchmark of several controllers which are out there already. So, if you look at this Core profile, there is something even a little more interesting that you can look up.

So, let us see what that core profile is. Core profile is coming from here. This is the core profile. So, true energy costs of deep sleep, some standard controllers have been characterized and those will come here. You can choose the controllers of interest whichever you wish. And then basically, you can say compare scores. And then you can say, I want to do that. I also want to do RAM retention and choose your RAM retention parts.

If you remove you can see things change automatically. So, I would want you to do this, you can play around with this. You can click on Core, you can see SD micro cortex, M 4 cortex, M 33 so on, are all mentioned here. RAM retention, there you are. I will choose this. And then once you are done, are you going to choose any controller of interest, you just say compare scores. There you are, you get this course. And you can start examining the scores in great detail.

So, I would like you to explore whatever we discussed in the class, with respect to what is shown here. And we will see if we can look up some exercises as well, based on what you read in this data sheet.

(Video Ends: 29:42)

That is about what I wanted to say about benchmarking controllers for your application. So, please do spend time understanding CoreMark. Thank you very much.