Computer Organization and Architecture: A Pedagogical Aspect

Prof. Jatindra Kr. Deka
Dr. Santosh Biswas
Dr. Arnab Sarkar
Department of Computer Science & Engineering
Indian Institute of Technology, Guwahati

Lecture - 23
Different Internal CPU Bus Organization

Hello and welcome to the last unit of this module of control unit and here in this unit we will be discussing about different internal CPU bus organization. So, throughout this module, on control unit we have seen how we can generate different type of control signals and then we have seen that how different instructions involve what type of control signals, and then we have discussed how these control signals can be generated using hardwired control, and then we have seen how we can do it using a micro programmed control unit.

But for everywhere we basically assume that the CPU is a single bus organization architecture. That is there is a single bus which carries the data and control signals there will be a single pair of busses. But now before we end this module, let us try to give you a very short idea that if there are multiple buses then what can be the advantages and what can be the disadvantages. Of course, one clear advantage as you can figure out if you have multiple buses to carry out data and control you require a much less control steps because many operations can be done in parallel.

So, this is a very simple which is logical argument; that if you have multiple paths then, obviously you can reach the destination much faster; because you can do things parallelly. But again also we see some stray cases where not much advantage is there, but also at the same time you have to appreciate that it involves more cost.

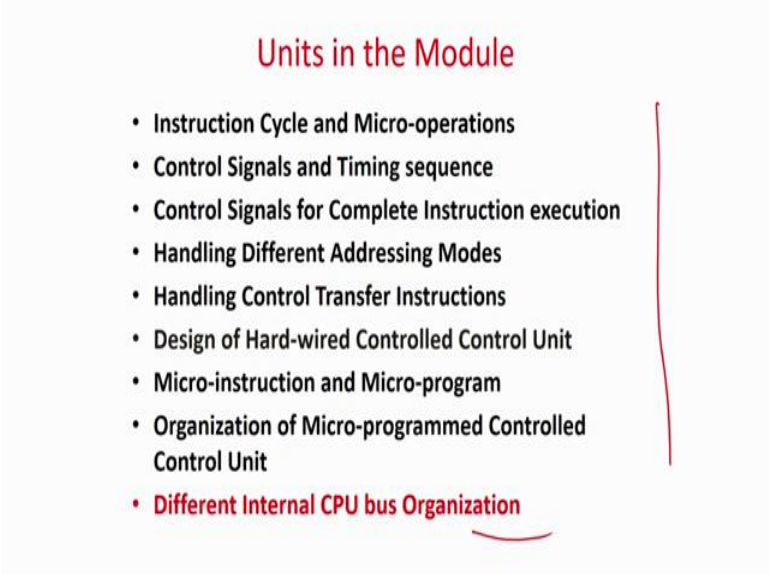
So, if we have very high number of system buses not only cost of the chip design or chip cost will increase, but also at the same time controlling it and will involve more number of circuit and overhead will increase. But we are not going to get give go to a vary in depth beside discussion on how different type of CPU buses, then how you can design control units for them rather we will try to give you an idea that what are the different type of control signals required if you have multiple bus architecture and what is the advantage and how things can move faster.

Based on this idea you can very easily recast if you have multiple bus architecture and then how can you recast and generate control signals for different instructions as well as also you will be able to generate circuits for control units and internal and micro program architecture.

Because you already know those concepts and here we will be show you what are the changes if you have multiple bus architecture, and then you yourself can think and make a merge and blend of it if somehow we have you have some point of time, you have to design a micro program control unit for three bus architecture or four bus architecture.

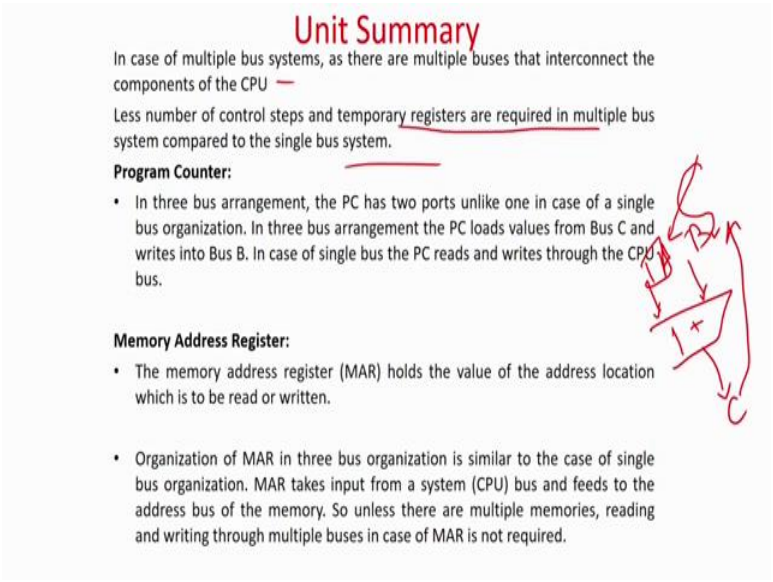
So, let us in the unit have a very brief a broad overview, that how things change if they have different internal bus organization compared to a single bus architecture. So, in this discussion mainly we will not be considering a two bus architecture, we will rather go for a three bus architecture, which will give you a more elaborate feel that if you have multiple I/O's possible together what is the advantages.

(Refer Slide Time: 03:07)



So, as you see that is the last part of this unit, that is how all these basically all these units basically covered the means whatever we covered in all these one units, basically was on single bus architecture. Finally, in the last we will just give you an overview of three bus architecture so that you can intuitively think that how other concepts will change, if you have a multiple bus architecture.

(Refer Slide Time: 03:28)



So, again in some pedagogical sense let us try to see what is going to be we will try to summarize in this unit what we are going to expect. So, in case of multiple bus systems there are multiple buses that connects the different components of the CPU that is obvious. So, there will be not a single data an address, bus there can be multiple buses to transfer the signals ok. Obviously, less number of control signals and temporary registers will be required if there are multiple buses as we will see that is quite obvious also, because if you have two lines you send the input and two lines to get the outputs you may not require any temporary registers because there will be two direct lines which will feed the ALU.

If you consider single bus architecture if you remember then if you want to add two numbers A+ B, but there is only one path to give the values then you have to intermediate store the value of A at some temporary register, the second operand that is B can be directly fed to the ALU through the bus and then you have to add and this is answer you also be have to be stored temporary in a register, because the bus at present is carrying the value B. Once the addition is done then you can make B free and then you can feed the answer of the A+B to the bus.

But if you have three buses then two bus will fed A+B, and the third bus will just give the value will take the value of the output that is A+B. So, in one go you can do A+B=C. It is as simple as there will be one input A, one input B, there are two buses, there will be an ALU and this will be third bus. So, this is A, B and C if there are three buses and, but if there is a single first one bus will dump the value of A and store it in in a register. Then the same bus will now divert

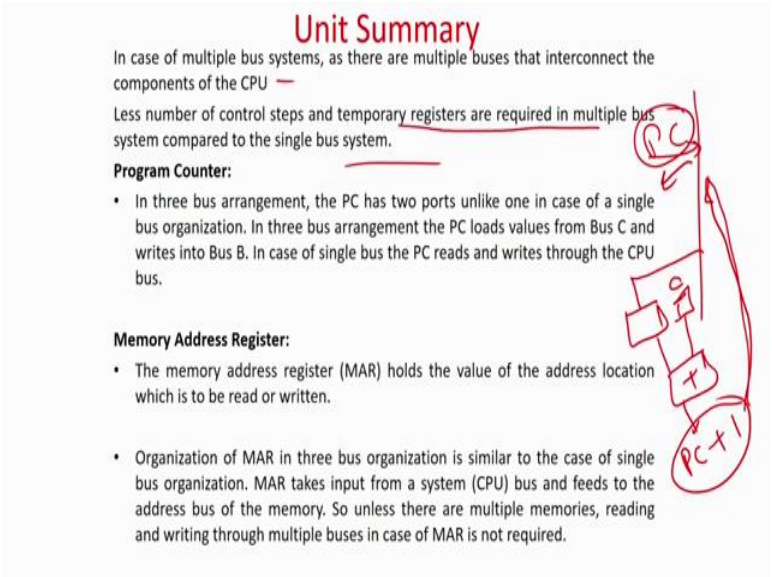
to B and it will be hold the B value temporarily. Then you will do A+B=C and then if in the third part third state you have to erase the value of B from that bus and dump the value of C to that bus. So, that it can be saved into the memory location.

So, three steps are required, but if you have more buses very easily you can in a single stage you can do multiple operations, which will be parallel and faster. So, that is what is going to be the basic summary of this of this unit. But now let us look at different components like program counter, memory address registers etcetera and how they will change if there are multiple buses.

So, what is the program counter? So, program counter actually points to the current instruction, and then it will do program counter plus the next address. So, that it can program counter plus increment, which will point to the next address of the instruction so. In fact, that is very simple. So, you always do PC=PC+constant.

So, in a single bus if you remember you require two stage; first the program counter will be dumped to the will be in the bus and in the second stage you have to do program counter plus program counter plus constant by the ALU, and that has to be stored in a temporary register. And then only in the second stage the temporary register value will be dumped to the bus, which will again go and save it to the program counter because this is single bus which does this.

(Refer Slide Time: 06:20)



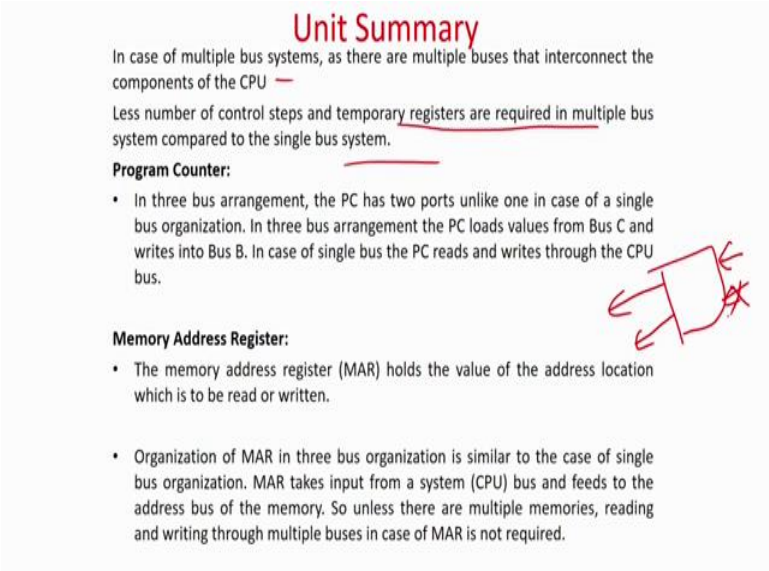
And first program counter is feeding this value here, which you are storing in a temporary register and this may be your constant and then you are going to add it by the ALU. So, this is equal to PC+1 then again it has to go and write back to the program counter.

But there is a single bus. So, in stage 1 the program counter value will go as an operand to the ALU, addition will be done as stored in a temporary register, which is this one is a temporary register and then in the next stage you will dump the value of PC+1 or the PC+constant into the bus it will again write back to the PC already we have seen the concept. So, two stages are required.

But in a three or multiple bus system mainly you are keeping our discussion here in a three bus system, and we are assuming that there are two buses will check take the output and one bus will give the input to the registers. We will see in the details, but of course, in two bus system all the registers actually have multiple ports because this is very obvious, because if you only one gateway and therefore, multiple paths then there will be no advantage. So, here different registers have different input and output port somebody has multiple output ports, then the register can give output to three parallel buses together and it can; obviously, read only from one port, it cannot read from multiple ports together.

So, there will be multiple in a multiple output ports which we basically assume.

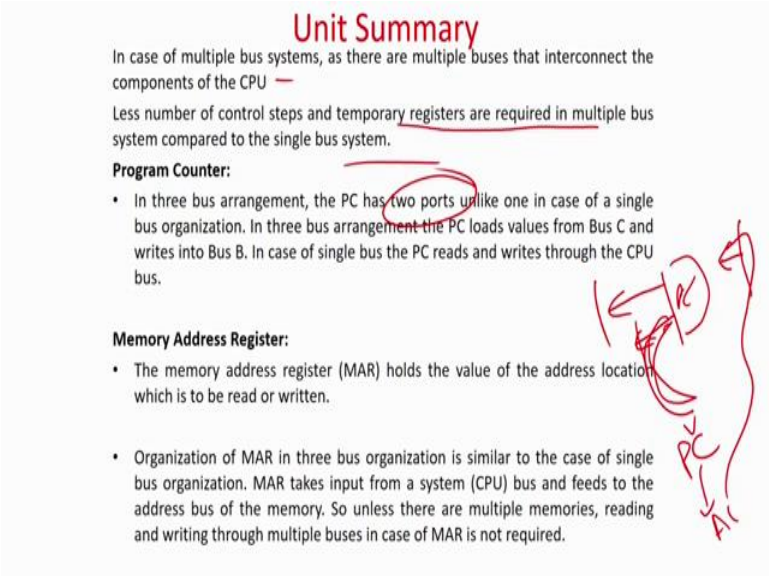
(Refer Slide Time: 07:33)



So in fact, it can be something like this one register can give multiple outputs to two different bus or more number of buses, and there will be also one input port which means takes the data in parallel.

So, the all these three activities can be done in together. So, one data can come in or the previous data can be sent out together and so forth. Of course, you cannot have two inputs that does not have any meaning basically. So, the program counter has two ports. So, what happens we will see later in details? So, we will see that how program counter can be made more efficient using two ports.

(Refer Slide Time: 08:05)

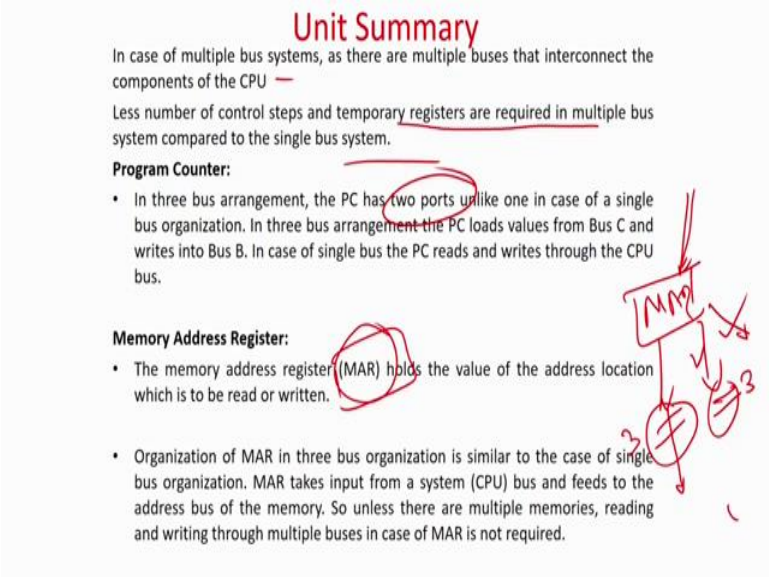


Because in one case the program counter will give the value out as this one as PC, which will be pointing to the current instruction and the same time it will also who has the operand to the ALU, you will have ALU, in the ALU we will do PC=PC+ 1, and it need not wait that you have to erase this value and then only you can dump the value of PC=PC+ 1 because there are two buses here in fact, multiple three buses. So, you can directly feed the value of PC=PC+ 1 in the same unit of time.

So, that therefore, there will be two ports and we can actually do this in single stage we will see later in with the figure later. So, but first with the summary is that we will study how program counter efficiency can be increased using two ports. Memory address register so this is very interesting.

So, here the memory address register in case of multiple bus as well as single bus will be similar now why? Because we have a single memory and memory address register what it does? It base basically tells that from which location of the memory data has to be brought in.

(Refer Slide Time: 09:03)



So, even if I have multiple ports or even if you have your memory address register can write to multiple outputs and similarly take data, and it can write simultaneous with the output this of course, we can do we can give the value in one go and other values it can give out.

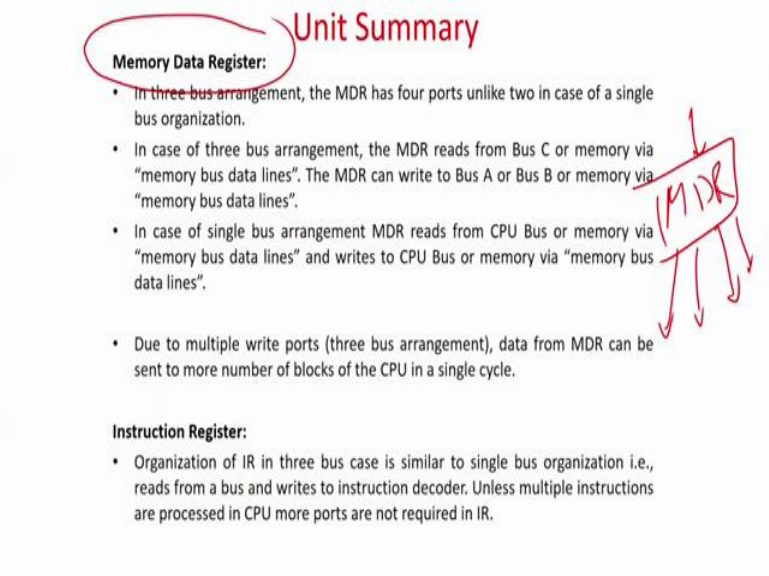
But you see this one will have no sense here, because if there is multiple memory; then only this is going to be a very advanced then it will be a very advantageous concept. Because you can read from one from one main memory location here of course, this is a memory location value we can also need from another memory. So, of course, the memory location will be same, but you can fetch two different data from two different memory blocks and you can get to two different registers. Of course, the memory location if it is 3 over here, it is 3 over here, also in this memory block but as there are two different independent memory data can be different.

So, you can make a program like that. So, that you can fetch from two different memory locations oh sorry from two different memory blocks you can fetch data together and you can use it parallelly. But in this context we are not handling processors which we are having multiple memory blocks as well. So, we are restricting to three bus architecture, but there is single memory. So, another that case you can you need not modify your memory address register. Because the output of the memory address register goes to the memory. So, if you

have a single memory of course, multiple outputs like this from the memory address register will not help.

So, we will also appreciate this fact in this unit, that when memory address register has no extra advantage of having more output lines if you have a single memory. Because, but if you have multiple memories then you of course, you can feed it together and get the advantage.

(Refer Slide Time: 10:34)



Then the memory data register. Of course, there from very intuitively you can feel that memory data register means it will take some data from the memory and of course, you have distribute to some others places like, if you have an instruction called $load\ R_1, M$. So, data from the memory will be dumped to register R_1.

So, in this case maybe if I have multiple ports of course, it is very advantageous because you can quickly transmit this data from memory load memory R_1 maybe there can be another same instruction $load\ R_2, M$ so in fact, if you have multiple ports from which you can do the output. So, you can very quickly read from the memory and send the value through two wires, because you have multiple buses that is R_1 and R_2 can be directly fed.

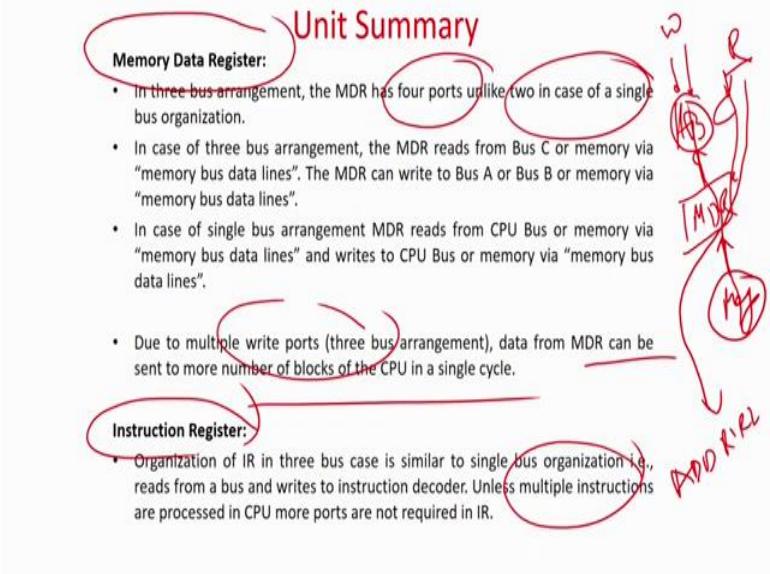
So, that is very true and very obvious that because you read one from the memory and then the data can go to multiple places like different registers, instruction registers, accumulators. So, if your memory data register this is coming from the memory of course, this will be 1, but you we can have if it has multiple outputs. So, it can feed different lines in one group

So, in this is one point which we are going to discuss in this unit that is how memory data register, if having multiple ports what will be the advantages. So, in a three bus architecture MDR has 4 ports unlike 2 in the case of a single bus organization. Single bus means so, the one input from the memory and the other will be to the other and the other will be towards the different places where it wants to send the data.

And of course, if this is the context of when the memory is writing to the registers. In fact, also for reading; what happens then actually things reverses? But in a nutshell I am not going to go into that type of intricacies in this unit, but in two single bus architecture from the memory to the registers and vice versa, but in case of multiple ports multiple buses, we have actually four ports four port memory. So, I have 4 port MDR.

So; obviously, in this case it will help, because in case of three bus arrangement MDR reads from bus C or memory, that is we are assuming there is a 3 bus architecture.

(Refer Slide Time: 12:42)



So, A, B and C. So, generally that means organization of the following is that everybody will dump the values to A, B and we will read that, then we will be writing to A and B and they will be reading from bus C. So, three buses will be there in the organization. So, all the registers etcetera we will write in bus A B and we will read from bus C and of course, because it is a memory data register. So, MDR it will it can also read from the memory.

So, basically in our previous signal bus architecture, it used to read from the memory and dump to bus A; because there is only one bus. But in this case actually they are increasing the number and then we will see how it becomes faster, because in memory data register you can read from the memory that is one, one extra bus C is there, that also can write to the memory data register and of course, it is going to give output to bus A and B. So, it has multiple ports.

So, in this case it is will be faster because you can take some data from the memory or you can take some data from some bus because in this case it actually writes from the some bus to the memory, and in this case it goes from the memory to the some registers etcetera through the wires A and B. So, multiple port means you read the data from the memory, and write to two locations connected to register two buses A and B simultaneously and it will be a faster operation.

So, due to due to multiple port writes data from MDR can go to more number of blocks in a single CPU cycle that is the bus word. More number of ports you read from the memory, you read from another bus called C or any other place and then dump directly to the components connected through bus A and B. Instruction register again this is very very similar to a single bus architecture, because even if you have multiple input output ports from a instruction register, it will not help you to parallelize because we are handling one instruction at a time.

But if it is a parallel processing kind of an architecture, then multiple instruction can be fetched and executed together. Like for example, if you have one instruction called say $add\ R_1, R_2$ another instruction say $add\ R_3\ , R_4$ and there is no inter interdependence in between. So, you can execute $add\ R_1, R_2$ and $add\ R_3, R_4$ in parallel. So, in that case if you have multiple instruction registers and multiple input output ports somehow it can help. But unless multiple instructions are there we are actually it is not better to have anything in the instruction register; it can be it is very similar to that of a single bus architecture.

So, here our assumptions are slightly different, it is not actually totally parallelized it is a single bus architecture like, but topology, but only the number of buses are 3 compared to 1. Memory is also not multiple and only single instructions are handled at a time. So, so in that context we are going to discuss this unit like arithmetic and logic unit.