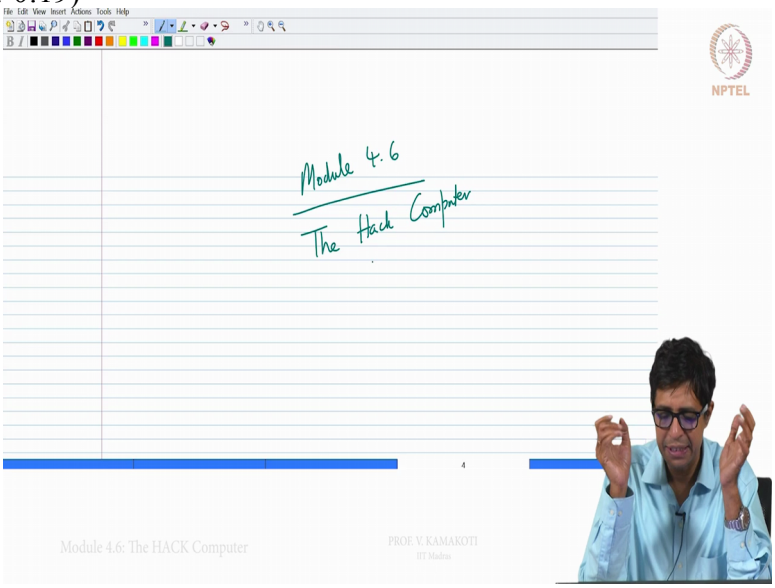
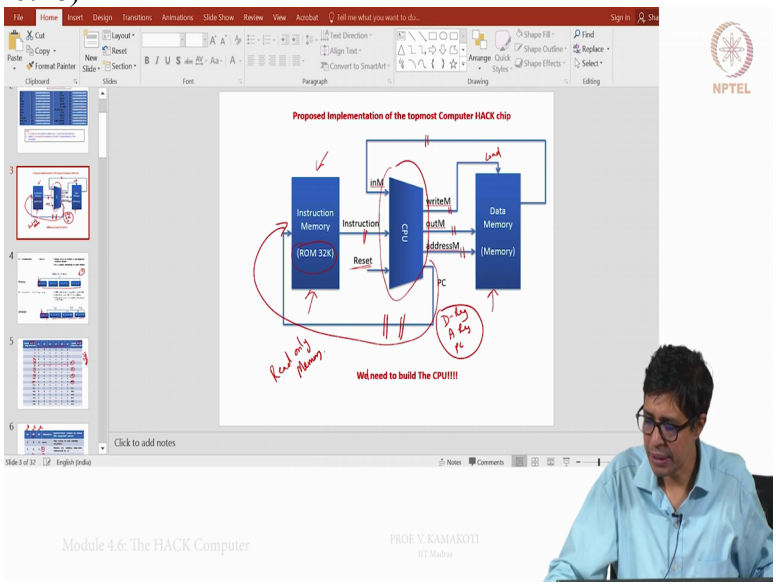
Foundations to Computer Systems Design
Professor V. Kamakoti
Department of Computer Science and Engineering
Indian
Module 4.6
The HACK Computer

(Refer Slide Time: 0:19)



So welcome to module 4.6. In this module, will see the final, the Hack computer. How does the Hack computer look like?

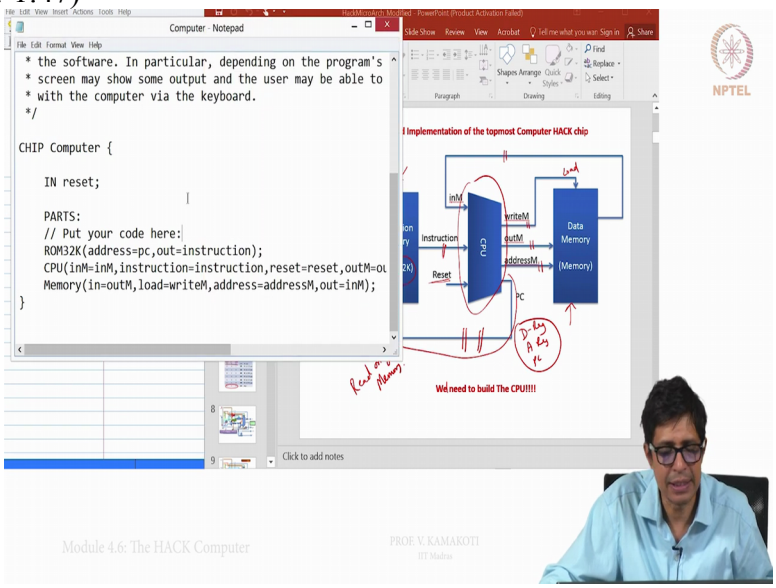
(Refer Slide Time: 0:28)

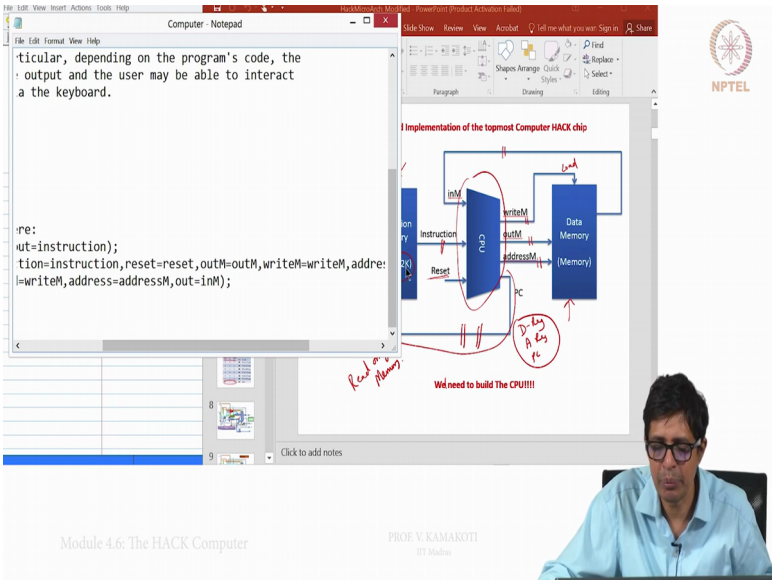


It is going to be pretty straightforward as you see here, the Hack computer basically has the data memory that you have already created. The data memory has a load, out and address. So the data memory basically has data plus screen plus keyboard. The output of the data memory will be fed back to the CPU, there will be a reset here. The PC will go to an inbuilt chip called ROM 32 and the instruction from there will feed to the CPU. So from the CPU, you will have a write M which will basically say whether I want access memory or not, there is a load input to your data memory.

Then whatever we want to access memory, that out M, the address on which I want to write and the PC. The PC comes back to the ROM 32K while the output of the data memory basically goes back as input some memory into the CPU and the instructions flow from the ROM 32K and then that is a reset pulse which basically resets the CPU. So this is what we need to create here.

(Refer Slide Time: 1:47)



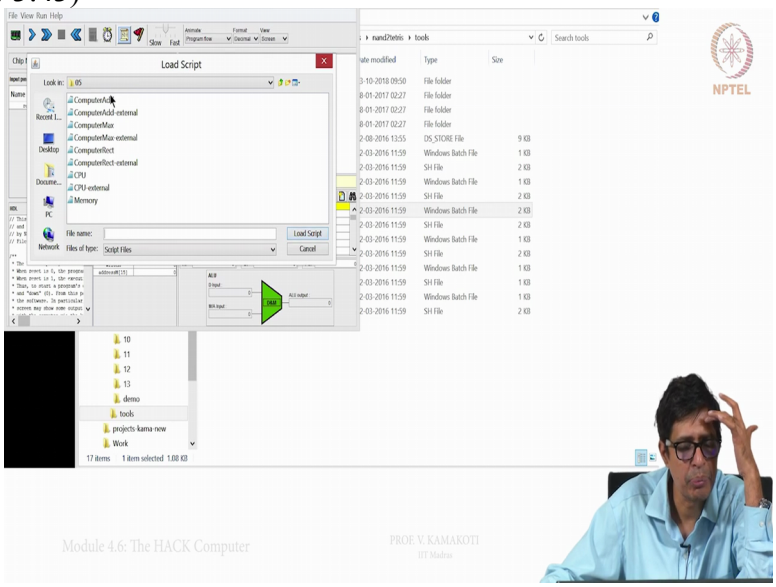


So this is just a so the entire CPU that we are trying to build just keep this in mind, so we will just we will just have this and we will open up the actual project. Project 5. We will open the computer. This is the HTL file and this is the exactly the 3 lines that we need to add to make this entire computer. What we have done here, we can just see, there is a ROM 32K to which the input address comes from the PC and the output is the instruction is what exactly what we are doing in ROM 32K.

And for the CPU, there is a input which is inM as you see, then the instruction comes from the output of the ROM, that is instruction. Then reset comes from the input of the computer, outM goes to the memory, writeM goes to the data memory. And then the addressM goes to the data memory. And then the PC comes out of this. This PC is fed into the ROM 32K. As you see here, the PC that is for the CPU is fed back into the ROM 32K. For the memory, the input is outM from the CPU and load is writeM that is coming from the CPU, address is addressM and out equal to inM.

So now that we have made these 3 modules, this is of course an inbuilt chip, the ROM 32K, well the CPU we have done and the data memory we have done. It is just a 3 line code to build this entire computer. Now we will just execute this compute this on the module to show that it works.

(Refer Slide Time: 3:43)



We go back to the tools, we just take the hardware simulator. So we just load the computer. Now let us load some of the scripts. So there are 3 scripts-computerAdd, computerMax and computerRectangle. Just for the thing we will just do the computerAdd as load script. Let us run the script at full speed. That is it. We can we can also have this computerMax. So computerAdd external and load this script also. Run it. And also have the computer max and then the computerMax external, computerRectangle.

A small rectangle should be drawn here. If we go and view the screen here. And similarly let us also see the computer. We will go to the screen here. Yes. So all the 3 programs, one for drawing a rectangle, finding a maximum and add, all these 3 programs have worked. So this is how we build a computer. So with this what we have done is we have finished the first part of building up an architecture, a machine on which we have understood the assembly languaged or the machine language and we have made the machines understand the machine language and execute.

And now we have built a small computer by itself. Now what we will do in the subsequent modules is to write softwares for this computer. So we will write an assembler, then we will write the virtual machine, interpreter, then we will write the compiler. We will compile again and run the game on this. So that is what we will be doing in the subsequent 4 modules, modules 5, 6, 7 and 8. So please complete the project 05 as a part of this thing so that your clear understanding the project 05 is very important for us to do the remaining projects and understanding the

remaining modules in great detail. And if you have any doubts, please do post on the forum and we will answer those doubts. Thank you very much.