PPI Project Description : 8255A - Programmable Peripheral Interface

----

- Marks : 5 Marks
- # Students : 3 Students Per Project. Doing it alone is not recommended, and will not give you any extra appreciation. We want you to help each other and learn from each other.
- Final Submission Date : **3 -12 - 2018**, **However** if you submit the project to me on Monday 26-11 face to face) in a perfect state, well commented code, and with a perfect test-bench covering all the requirements, **you will get extra 3 marks each**.
  A condition for getting those extra 3 marks is that you do everything on your own as a team, without requesting help from the TAs.

- For the remaining students who won't deliver on 26-11, we will tell you how exactly to submit later.

Please use this sheet to write your team names and IDs. You must be 3 students in every project.
https://docs.google.com/spreadsheets/d/1-RSaaKvCKRGYenNsAZJOC4GUnTKy4SSHhbF0xtpw4qE/edit?usp=sharing

Description :

You will implement the 8255 PPI chip in verilog. Learn about the 8255 chip from here :

- https://www.tutorialspoint.com/microprocessor/microprocessor_intel_8255a_programmable_peripheral_interface.htm
- And here : https://en.wikipedia.org/wiki/Intel_8255
- You can also check the datasheet here, but it is more professional and might be slightly harder to start with. Read it after you see the tutorials before. Here is the link : http://pdf.datasheetcatalog.com/datasheet/Intel/mXuttts.pdf
- There are Lectures from nptel here :
  https://nptel.ac.in/courses/108107029/module9/lecture49.pdf
  https://nptel.ac.in/courses/108107029/module9/lecture50.pdf
- There are also many video tutorials out there explaining the chip(mostly indians), you can learn alot from them.
- In general I think you will need to learn from a mix of all of those sources in addition to of course some googling before you can get it right. This is a real chip, so things can be slightly complicated. However, this is much better than just doing toy-examples. And after all it is still a very simple real chip, and we want you to model even a more simplified version of it. Complete reading this document to exactly know what we want you to model in verilog.

Note the following: The 8255 chip has two modes of operation, BSR mode (see wikipedia for this) and I/O mode(Both Wikipedia and the tutorial explain this).
Inside the I/O Mode, there are three modes of operation. You are required to implement only (**Mode 0 - simple I/O**) as described in the wikipedia page. So, you need to implement the BSR mode, and the Mode 0 from the I/O mode only.

Note : You will need bidirectional ports that can act as inputs sometimes and as outputs sometimes. They are called "inout" in verilog. To see how to use them, see this document where I wrote a simple verilog code example.
https://docs.google.com/document/d/1RuQjonuT5VQVAgQRSKMDe7pln3IrcMX5Lc9J4ixDLqw/edit?usp=sharing

Note: There is no clock involved here at all. This is not a synchronous chip. This doesn't mean that it doesn't have registers. It does have registers, but it uses latches that don't depend on clock edges. The difference between a flip-flop and a latch can be summarized as follows :

For latches, the outputs are constantly affected by their inputs as long as the enable signal is present. When they are enabled, their content changes immediately when their inputs change. Flip-flops have their content change only either at the rising or trailing edge of the enable signal. This enable signal controls the clock signal. After the rising or trailing edge of the clock, the flip-flop content remains constant even if the input changes.

So basically, a latch will change its input if the enable signal is high, and will preserve the old state if the enable signal is low.

More Details :
You need to write the verilog code for this chip. You are responsible to test all required modes of operation and make sure it works correctly. You must write a testbench, and we will not accept the work of those who test their designs manually without well-written test-benches. We require your work to be accurate and since we believe projects are the most important part of this subject, we will not give you marks unless we believe your work really deserves them. We (Might) give you a test bench similar to the one we will test your work with in a week or so, but don't depend on that. Create your own testbench, and if we happen to give you our testbench, you will need to use our test bench. Anyway, your chip couldn't shouldn't change at all if you change your testbench and use ours. We will treat your chip as a black-box in our testbench. More on that later isa.