**Course Overview**

**Course Overview**

**Welcome**

Hello, I'm [Nick Beser](https://jhu.instructure.com/courses/33443/pages/instructor-biography). Welcome to 525.612, the online offering of Computer Architecture. For now, I would like to give you an overview of what we will be doing this semester and where you should begin.

If you have any questions or concerns while going through this course you may post them in the [Problem Forum](https://jhu.instructure.com/courses/33443/discussion_topics/416773) discussion, located under Discussions, or you may contact me directly at [nbeser1@jh.edu](mailto:nbeser1@jh.edu). Additionally, you can refer to **Help** on the global navigation menu for a listing of resources and support available to you.

**About This Course**

This class is part of the Computer Engineering program, and provides a basis for understanding computer architecture. Coming from a design standpoint, the class goes into more hardware issues than a typical computer science oriented course. We are using Patterson and Hennessy's text: Computer Organization and Design The Hardware/Software Interface, 6th Edition. I also use other references that are being supplied in the eReserves section on the course menu. One of them, Yamin Li's text: Computer Principles and Design in Verilog HDL has some wonderful examples of computer arithmetic and MIPS processor design emulation. I mixed these examples with the material from Patterson and Hennessy. I started with the Li designs, and then ported them to the Nexys 4 DDR board. The Nexyx A7 is the same board (rebranded). We modified the designs to support a serial port debugging interface (PMODUSBUart). This interface will allow you to take an existing MIPS design that you compiled using Vivado, and then download a new memory file to change the program on the emulated computer. By the end of the class we will be adding new instructions to these emulations, and expanding our tools to include a C compiler. These tools when taken together provides a hands on learning experience in computer architecture.

In this class we will examine the design issues that drove the development of the Reduced Instruction Set Computer. We will cover processor design, and use the requirements of Reduced Instruction Set Computers to examine ways of implementing a theoretical design. We will also cover computer arithmetic which when coupled with the processor design sections defines much of the resources offered by modern architecture. We will also cover memory systems, including memory hierarchies such as multilevel cache design and virtual memory. Rounding out our design, we will look at a complete processor simulation (A pipeline MIPS processor with Cache and VM) which implements a complete RISC architecture. Along the way, we will simulate the designs, so that we will always be able to keep the path from theory to practice clear.

In order to develop grades for this class I have three elements that we will use: Homework Assignments (problems from the text), Forum Discussion: (Class problems that are worked as group problems rather than individual efforts), and a research project with a paper and presentation. Of the three element's the Discussion Forum is unique to the on-line type of class. I will suggest a topic for each week. As a student, you are expected to divide the topic up among the class, provide some individual input into the topic, and assist your fellow student's by commenting and contributing to their input. There is a pattern to this type contribution. I suggest that as a student, you should discuss the topic among the class, get your part in place early in the week, and then spend the rest of the week commenting and contributing to your and the classes comments. You should not wait until the end of the week to contribute. If you find that you only participated at the end, and you do not do the required number of posts, you will not get full credit. If you did not participate at all, you will get no credit. While I am looking for quality not quantity, if you find that you participated less than 6 times during the week and you only reacted rather than initiated discussion, you may not get full credit. If you typically wait until the end of the week to jump in, you may not get full credit.

For those who need a **rubric**:

Here is what I use:

The grading criteria is:

**5 - 6** you will probably get a 10 if you turned in a good post, and helped out the discussion

**4** - you will probably get a 9 if you turned in a good post, but If you didn't you would probably get a 7-8 depending on what you posted

**3** - If you turned in a good post, you will probably get a 7 (I have been very generous up to now). You can also get a 5-6 if you did not do a detailed post.

**2** - You will get a 6 if you turned in a good post, and a 5 if you didn't

**1** - you will get a 5 if you turned in a good post, and 4 if you didn't

**0** - You will get a 0

Most of the time, the class is not aware that they posted 6 or more posts. When you are in the middle of a discussion, it just happens. For those who can't get into posting. I fall back on the rubric.

I find that the discussion forum could be characterized as a conversation. If you participate in the conversation, and contribute some unique information, then it is easy to get full credit.

The course will also use the same Xilinx tools that you used in the VHDL class: Xilinx Vivado 2020.2 or newer. I find that the newer tools seem to take up more disk space and present problems on older systems. These are available for download and should be downloaded as soon as possible. The software runs on PC's and linux. The software can run on a Mac using VMFusion running Linux. Please note that the new Mac's don't seem to run VMFusion or VirtualBox, While the tools run on the PC, I have discovered that the Nexys 4 DDR Board and Nexyx A7 are difficult to get running on the Windows computer. I suggest to keep the problems to a minimum, I recommend that if you don't have a native Linux system (Ubuntu 20.04 or newer), you can run a VM system on the mac or PC. I have documented the procedures for running VirtualBox, loading Ubuntu 20.04, loading Xilinx Vivado, and adding the additional libraries and tools required.  I find that a native linux system will execute faster, a VM system is not a bad way to go. You can use a newer version of Ubuntu, and the latest version of Xilinx, but you may need more storage to hold the code. While I have created a system with 100Gbyte of storage, you may need more storage (150 to 200 Gigabytes)

To get the class started, I am also loading two simple designs that will run with the Nexys board. The design simulates two MIPS architectures (Single Cycle with Interrupts, and Multi-Cycle). The instructions and additional zip file containing the processor designs and test assembly will be available on the Module 1 course file. We will be using a Git utility to distribute more code.

I will be discussing project topics through the first 9 weeks of class. At the end of Module 6, you should turn in a project proposal that details your topic. A template will be provided which follows the classic Industry Research and Development (IR&D) format. At the end of Module 9, you will turn in a status report that details your progress. Note that you could also turn in a re-scoped project proposal in the event your first was too optimistic.  Prior to Module 14 the class will pre-record their presentations, and then post them using TEAMS. The presentations will be organized into a 6 presentation TEAM, with multiple TEAMS. As an example if we have 36 students we will have 6 TEAMS. Each student will be expected to review at least 6 presentations and ask questions. You are not limited to asking questions to your own TEAM, and are encouraged to review other TEAMS presentations. The presentation recording will be posted at the start of Module 14. The report is due by the end of the module. It should also be noted that Module 12 and 13 will not have homework, and that the discussion topics will be very light for those weeks, so that the class may have sufficient time to complete their projects.

I will be holding office hours once a week on Wednesdays. Since the week begins on Monday, it makes sense to hold the first office hours early in the week so that comments can be factored into both the discussion topic and homework assignments.

**Student Project Archive**

Beginning with the use of the Canvas software, the Student Project Archive has been changed slightly. During the semester, all student projects of that semester will be provided to the class. Past projects will be provided only if the author has signed a FERPA form giving permission to include the project in the archive. The signed waiver is not a requirement for your grade, but I hope you will consider signing the form so that your work will be included in the archive. When you look at the past archive, all projects are listed. Only projects with a signed waiver will have links to read the content. Here is a blank form for you to fill out.

[FERPA release for EN.525.612 Computer Architecture.docx](https://jhu.instructure.com/courses/60023/files/9590520?wrap=1)

**How To Begin**

You should begin by reading the **Course Syllabus** and **Course Outline**. It is also a good idea to read the other information available in the **Course Information** area and to explore the global and course menu items to become familiar with the learning environment we will use this semester.

After you have read the **Course Syllabus** and **Course Outline**, it is your turn to tell me something about yourself. Please go to **Discussions** and respond to the [**Student Introductions**](https://jhu.instructure.com/courses/60023/discussion_topics/684784) discussion with your answers to the following:

* Tell us why you're taking this course.
* Share something interesting about you that will likely be unique in this class.
* Post what you think is a picture that represents you best—it can be an actual photograph of you, or some other picture that represents who you are or something about you.

After you have introduced yourself to the class, you should begin working your way through **Module 1** found under **Modules**. Look under **Modules** to find out what the current assignments are. You should also review the **Calendar** at least once a week for any items that are starting or are coming due. **Announcements** will be posted throughout the semester as reminders or to notify you of updates or changes in the course. You should log into the course at least 5 times a week to check for current announcements. You will find the **Announcements** on the course **Home** page when you login or by selecting **Announcements** on the course menu.

I would like to make sure that we have a good communications path setup for this class. If you are not normally using your JHU email or don't have it forwarded properly, you will find that you will be missing the announcements and messages. As a safety measure, I would like you to send me an email with a primary email address that you always read. I will use this address if I can't seem to reach you by the Canvas Inbox which uses the your jhu.edu address. Please add 525.612 in the subject heading.

I will also be using your email to enable your access to bitbucket.org. Verilog design files are provided on bitbucket as well as example code and tools.