EGR430 **PARALLEL PROCESSING Lecture & Lab**

Spring 2024

Lectures include design of embedded systems, microcontrollers, microprocessors, and supercomputers. Cache designs, parallel processing topics, instruction set designs, neurocomputer designs, IBM research. Major digital design laboratory projects require students to design, build, test, and demonstrate prototype computer hardware. Custom lab manuals

**Prerequisite(s): EGR/CS332 (EGR330 after Spring 2022) Hours: lecture 3, laboratory 3*

COURSE-CREDIT / CONTACT-HOURS = 4/6 (*6 x 50 = 300 minutes per week of "contact")*

PROFESSOR

Joseph T Wunderlich PhD Website: http://users.etown.edu/w/wunderjt

Offices: E274E and E273, *or remote (text, or call my cell and leave a message so I know it's not spam)*

Office Hours: http://users.etown.edu/w/wunderjt/schedules/Wunderlich%20Schedule%20Card%20Spring%202024.pdf

Office: 717-361-1295 Cell: 717-368-9715 Email: wunderjt@etown.edu

MEETING-TIMES: **Mondays & Wednesdays 3:30PM to 6:00PM**

PREREQUISITE TOPICS

EGR330 COMBINATIONAL DIGITAL CIRCUIT DESIGN

- Example with **Wunderlich 8 STEPS OF COMBINATIONAL** Digital Circuit Design
- Digital Circuit Basics
- Boolean Algebra. Circuit Re-Design Proofs *(all else moved into new course in "Applied Discrete Math")*
- Design using K-Map simplification for Two, Three, Four, & Five variables; Derivation of Maps
- Digital Circuit Simulators (Logisim, Xilinx) vs. Real-Time circuit implementations
- ETOWN **MANUAL** 2018 IC's, Circuit Trainer, and Power Supply PDF MP4 YouTube (Fan-out, etc.)
  - Pull-up and LED Current-limiting Resistors, Floating Pins, etc. (1,2)
  - List of 7400-series integrated circuits
- Don't-Care conditions, Code Convertor
- XOR-Pattern non-SOP functions *Just for Reference: Non-linear-Separability (slide#8) in Neural Networks PPTX-w/audio PDF MP4 YouTube*
- Voting Machine, Display Controller

---------- **VIDEO REVIEW 1; FOR ALL ABOVE: MP4 YouTube**

- **AD-HOC DESIGNS** (i.e., no Design "**STEPS**")
  - 2-bit-Adder into Bit-Sliced, Scalability **AD-HOC** Full Adder Half-Adder
  - 2-bit-Subtractor into Bit-Sliced, Scalability **AD-HOC** Adder/Subtractor
    - Two's Complement Number Representations PPTX-w/audio PDF MP4 YouTube *(IEEE BFP Binary Floating Point just for reference)*
      - *Just for Reference: IEEE BFP PDF PPTX-w/audio MP4 YouTube, Fractional-part PDF PPTX-w/audio MP4 YouTube*
  - 2-bit-Multiplier into Scalability **AD-HOC** Multiplier
- **FUNCTIONAL BLOCKS** (to build larger circuits) – *without internal designs: :*HalfAdder,FullAdder,Multiplier,Decoder,Multipexor(MUX),Encoder,DeMultiplexor
  - *Internal designs:* Decoder, Multiplexor(MUX)

---------- **VIDEO REVIEW 2; FOR ALL ABOVE SINCE LAST REVIEW: MP4 YouTube**

EGR330 SEQUENTIAL DIGITAL CIRCUIT ANALYSIS & DESIGN

- Concept *(Sequential w / Combinational nested within)*
- WaveForms, Analysis & Design (**F**inite **S**tate **M**achines, **Wunderlich 13 STEPS OF SEQUENTIAL** Digital Circuit Design) More problems
- Design w/unused states More problems
- Digital Circuit Simulators vs. Real-Time circuit implementations
  - De-Bouncing switches(1,2,3)
- Counters More sample problems
- CPU Pipeline -- *driven by a Finite State Machine*
- CPU Design & Assembly Language intro

---------- **VIDEO REVIEW 3; FOR ALL ABOVE SINCE LAST REVIEW: MP4 YouTube**

EGR330 INTRO TO MICROCONTROLLERS & ASSEMBLY LANGUAGE

- Wunderlich, J.T. (1999). **Focusing on the blurry distinction between microprocessors and microcontrollers.** In *Proceedings of 1999 ASEE Annual Conference & Exposition, Charlotte, NC*: (session 3547), [CD-ROM]. ASEE Publications. PDF MP4 YouTube
- Intel 8051 Microcontroller
  - ETOWN **VIDEO MANUAL** 2022 Intel 8051 Microcontroller **Simulator** ETOWN **VIDEO EXAMPLE** 2022 Intel 8051 Microcontroller **Simulator**
  - Overview/architecture, Memory Map, PSW, Stack
  - Instruction Set (1,2), MOVES, LOGIC, MATH, JUMPS&CALLS (PPT,text), 80251

EGR330 CIRCUIT ISOLATION

- ETOWN **MANUAL** 2019 Relays Phoenix Contact Relays
- Buffer-chips

EGR330 PROGRAMMABLE LOGIC CONTROLLERS (PLC'S)

- Ladder-Logic PLC's Fundamentals Evolution
- ETOWN **MANUAL** 2019 NanoLC PLC previous version:2018 Example design
- ETOWN **MANUAL** 2019 AXIOLINE PLC
  - 2021 PLC-NEXT VIDEO Tutorial
  - 2021 connecting PLC-NEXT to AXIOLINE using PROFINET Comm. VIDEO Tutorial

CS and MATH COURSES (e.g. CS/EGR230): Number Representations (PDF,PPTX-w/audio,MP4, **YouTube**); IEEE Binary Floating Point Example (PDF,PPTX-w/audio,MP4,YouTube); PC Design 1&2 *(PDF PPTX-w/audio MP4 YouTube) (PDF PPTX-w/audio MP4 YouTube);* Computer Graphics & Boards (*PDF PPTX-w/audio MP4 YouTube) (PDF PPTX-w/audio MP4 YouTube)* Display Technologies PDF PPTX-w/audio MP4 YouTube

EGR430 **PARALLEL PROCESSING** Lecture & Lab

J. Wunderlich PhD

COURSE OUTLINE

Spring 2024

CASE STUDIES

SC-16H CPU Design by Glen G Langdon PPTX-w/audio PDF MP4 YouTube
2021 Etown GPU Design by J. Freaney (YouTube) – first review "Computer Graphics & Boards" from CS230
2020 Etown CPU Design by J. Freaney and E.Schneider a-(MP4 YouTube), b-(MP4 YouTube)
2018 Etown Vector-Register Processor w/neural-network function Video1,Poster1, Video2,Poster2, Video3,Posters3(a,b); MP4_of_all
2016 Etown Super-Scalar Dual-Core Processor
2014 Etown Application-Specific Computer Design Competition (Team Videos: 1, 2, 3, 4)

INTRO to SUPERCOMPUTERS & NEUROCOMPUTERS

- Overview of J. Wunderlich **IBM S/390 & Z-Series** R&D + previous Neural Network Processors PPTX-w/audio PDF **MP4 YouTube**

COMPUTING FUNDAMENTALS PPTX PDF MP4 YouTube

- Symmetric Multiprocessing (SMP), Vector-Register, vs. Massively Parallel Processing (MPP) machines, SISD, SIMD, MIMD, Interconnect overview
- "Levels" of Computing, Microprocessor vs. Microcontroller Design
- Cache & Virtual Memory overview, CISC/RISC overview, Parallel Processing overview

PARALLEL PROCESSING FUNDAMENTALS PPTX-w/audio PDF MP4 YouTube

- Dependencies (Data, I/O, Resource, and Control)
- Grain Size, Hardware vs. Software Parallelism
- Scheduling, Partitioning, and Grain-Packing Superscalar and Multiple-Processors/Cores
- Out of Order Execution Dynamic Scheduling, Score-Boarding

PARALLEL INTERCONNECT ARCHITECTURES *of Cores, Processors, and Networks* PPTX-w/audio PDF MP4 YouTube

- Connectivity Typologies (Static vs. Dynamic)
- Scalability
- **IBM S/390 & Z-Series** Parallel Sysplex PDF

CACHE DESIGN PPTX-w/audio PDF MP4 YouTube

- Spatial and Temporal Locality of Reference
- Mapping to Memory (Direct, Associative, Set-Associative, Sector-mapped)
- Cache Block Size Design
- Replacement policies after a Cache Miss
- Cache Coherency

VIRTUAL MEMORY DESIGN PPTX-w/audio PDF MP4 YouTube

- Translation look-aside Buffer (cache virtual address translations)

CISC vs. RISC INSTRUCTION SET DESIGN, PERFORMANCE & SCALABILITY PPTX-w/audio PDF MP4 YouTube

- Addressing Modes, Instruction-Format Design
- Scalability & Performance
  - *Quick Review of general performance and decision analysis (application to monitors) from EGR/CS230*: PDF PPTX-w/audio **MP4 YouTube**

SCALABILITY in HIGH PERFORMANCE COMPUTING MP4 YouTube

"Simultac Fonton: A Fine-Grain Architecture for Extreme Performance beyond Moore's Law" by M. Brodowicz and T. Sterling

NEURAL NETWORK PROCESSOR DESIGN

- **Neurocomputer Reference material** from my *EGR434 Robotics & Machine Intelligence* course
  - Wunderlich, J.T. (2004). **Top-down vs. bottom-up neurocomputer design.** In *Intelligent Engineering Systems through Artificial Neural Networks, Proceedings of ANNIE 2004 Int'l Conference, St. Louis, MO.* H. Dagli (Ed.): Vol. 14. (pp. 855-866). ASME Press. [Paper Award]
  - Wunderlich Deep Learning Book Chapter Draft & Neural Network Processor Designs PPTX-w/audio PDF MP4 YouTube
  - Wunderlich Machine Intelligence History -- Draft book chapter PPTX-w/audio PDF MP4 YouTube
  - Calculus for Machine Learning PPTX-w/audio PDF MP4 YouTube
  - Machine Intelligence Intro -- Symbolic AI vs Neural Networks PPTX-w/audio PDF MP4 YouTube
  - Neural Network Code MP4 YouTube

"Deep Learning using Rectified Linear Units (ReLU)" by A. Agarap – Neural Network Transfer Function Hardware

IBM S/390 & Z-Series PARALLEL PROCESSOING (supercomputers)

J. Wunderlich Ph.D. Advisory Level Engineer & Researcher **MP4 YouTube**

- Machine Evolution PDF PPTX
- Processor Design including early Cache and Out-of-order Execution Design PDF
- Principles of Operation ("POPs") PDF
- Quick Reference PDF
- **BRANCH PREDICTION** with Branch History Table **(BHT)** – to *cache* branch addresses PDF
- **VIRTUAL ADDRESS PREDICTION** with Translation Lookaside Buffer **(TLB)** - to *cache* Virtual Address Translations PDF
- **QUALITY CONTROL / VERIFICATION** with Wunderlich "Controlled Randomness" *patented by IBM* PDF PPT
  - Wunderlich, J.T. (2003). **Functional verification of SMP, MPP, and vector-register supercomputers through controlled randomness.** In *Proceedings of IEEE SoutheastCon, Ocho Rios, Jamaica,* M. Curtis (Ed.): (pp. 117-122). IEEE Press. PPTX-w/audio PDF MP4 YouTube

CRITICAL TIMINGS IN PROCESSOR/CORE DESIGN

- J. Wunderlich Neural Network Chip Design #2, Fabrication, & Testing (including timings)
- J. Wunderlich IBM Research

LEARNING OBJECTIVES

1. Qualitative and quantitative ANALYSYS and DESIGN of advanced computer architectures and parallel processing systems
2. Architectural Models of Parallel Systems & Topologies *(superscalar, multi-core, multi-processor, vector-register, etc.)*
3. Scalable Performance of Parallel Systems & Topologies *(multi-core, multi-processor, vector-register, etc.)*
4. Advanced Memory and Cache Design
5. RISC and CISC Instruction Set and Pipeline Design
6. Neural Network Processor Designs for Machine Leraning
7. Hands-on Gate-level Design & Testing of parallel systems *(simulations and built systems)*
8. Quality Control & Verification *(from simplest lab designs to the most complex supercomputer designs)*

LECTURES INCLUDE EXCERPTS FROM:

[1] Yan Solihin, ***"Fundamentals of Parallel Multicore Architecture, Chapman & Hall/CRC Computational Science,"*** 1st Edition, November 24, 2015. ISBN-10 : 9781482211184 ISBN-13 : 978-1482211184

[2] John Paul Shen, ***"Modern Processor Design: Fundamentals of Superscalar Processors,"*** Waveland Press, Inc, 1st Edition, July 30, 2013. ISBN: 1478607831

[3] K. Hwang, ***" Advanced Computer Architecture: Parallelism, Scalability, Programmability,"*** Boston: McGraw-Hill, 1992. ISBN: 0070316228

[4] My YouTube Channel "Advanced Hi-Tech LECTURES" playlist

LAB MANUALS *-- written by students, often edited by J. Wunderlich*

[1] **2018 IC's, Circuit Trainer, and Power Supply**
[2] **2019 Relays**
[3] **2019 NanoLC PLC(Programmable Logic Controller)**
[4] **2019 Rasberry Pi and ARM Microcontroller**
[5] **2022 Intel 8051 Microcontroller Simulator VIDEO Tutorial**
[6] **2019 Intel 8051 Microcontroller** (2015, 2014, Pre-2013)
[7] **2019 AXIOLINE PLC**
[8] **2021 PLC-NEXT VIDEO Tutorial** **2021 connecting PLC-NEXT to AXIOLINE using PROFINET Comm. VIDEO Tutorial**
[9] **2022 FPGA Verilog HDL & Gate-level Simulator (and waveforms) "Vivado" VIDEO Tutorial** (IDE requires 100 GigaBytes on Harddrive)
[10] **2019 FPGA** for HDL (Hardware Description Language)
[11] **2018 FPGA** for Gate-Level Circuit Design, and critical Waveforms/Timing design (2018A, 2013, pre-2013)

GRADING

45% Laboratory projects
10% HomeWorks and Reading Assignments/Discussions
25% Midterm Exam
20% Final Exam

Up to –10% of course grade for absences, tardiness, or lack of participation; HOWEVER, IF YOU'RE SICK, DON"T COME TO CLASS

GRADE: (60-62)=D-, (63-67)=D, (68-69)=D+, (70-72)=C-, (73-77)=C, (78-79)=C+, (80-82)=B-, (83-87)=B, (88-89)=B+, (90-92)=A-, (93-100)=A
(with any fractional part rounded to the nearest integer)

ACADEMIC HONESTY

Elizabethtown College Pledge of Integrity: *"Elizabethtown College is a community engaged in a living and learning experience, the foundation of which is mutual trust and respect. Therefore, we will strive to behave toward one another with respect for the rights of others, and we promise to represent as our work only that which is indeed our own, refraining from all forms of lying, plagiarizing, and cheating."* See the 2016-17 Elizabethtown College Catalog, "Standards of Academic Integrity http://catalog.etown.edu/content.php?catoid=10&navoid=507#Academic_Judicial_System or Academic Integrity at Elizabethtown College, 11th ed. https://www.etown.edu/offices/dean-of-students/files/academic-integrity-handbook.pdf

ARTIFICIAL INTELLIGENCE USE

"In this course, students shall give credit to AI tools whenever used, even if only to generate ideas rather than usable text or illustrations. When using AI tools on assignments, add an appendix showing (a) the entire exchange, highlighting the most relevant sections; (b) a description of precisely which AI tools were used (e.g. ChatGPT private subscription version or DALL-E free version), (c) an explanation of how the AI tools were used (e.g. to generate ideas, turns of phrase, elements of text, long stretches of text, lines of argument, pieces of evidence, maps of the conceptual territory, illustrations of key concepts, etc.); (d) an account of why AI tools were used (e.g. to save time, to surmount writer's block, to stimulate thinking, to handle mounting stress, to clarify prose, to translate text, to experiment for fun, etc.). Students shall not use AI tools during in-class examinations, or assignments unless explicitly permitted and instructed. Overall, AI tools should be used wisely and reflectively with an aim to deepen understanding of subject matter." Source: https://ctl.utexas.edu/chatgpt-and-generative-ai-tools-sample-syllabus-policy-statements

DISABILITY SERVICES, RELIGIOUS OBSERVANCES, and COVID

https://elizabethtown-my.sharepoint.com/:w:/g/personal/ouimetc_etown_edu/EfZ-QooKt_VPjgwsWJz230wB3Rb6ClHsPvE0xuqWCpr-UA?e=4%3acZzjpW&at=9

DISABILITY SERVICES: Elizabethtown College welcomes otherwise qualified students with disabilities and is committed to providing access for all students to courses, programs, services, and activities. If you have a documented disability such as a learning disability or chronic illness or a new circumstance such as a concussion and would like to request accommodations, please contact the Director of Disability Services by phone (717-361-1227) or e-mail (daviesl@etown.edu). The Office of Disability Services can provide resources to you and facilitate communication with faculty about reasonable accommodations. After meeting with the Office of Disability Services, please set up an appointment to meet with me, the instructor, to discuss the accommodations as they pertain to my class.

RELIGIOUS OBSERVANCES: The College is eager to facilitate individual religious beliefs and practices whenever possible while retaining course student learning outcomes. It is your responsibility to meet with the class instructor in advance to request arrangements related to your religious observances that may conflict with this class, and to make appropriate plans to make up any missed work.

COVID-RELATED EXPECTATIONS: All students are expected to adhere to the established community expectations around safety, including: daily digital health reporting, physical distancing, proper wearing of facial coverings within buildings and classrooms and when within six feet of individuals outdoors, frequent handwashing, and participation in cleaning and sanitizing protocols as requested. You will be turned away from class if you do not have a face covering. Students diagnosed with a health condition that precludes mask wearing can contact Lynne Davies in Disability Services (daviesl@etown.edu) to request remote learning as a reasonable accommodation. **If you are exhibiting any symptoms of COVID or fail to pass the daily health screen, do not come to class.** Failure to adhere to the established community expectations around safety will result in notification of Campus Security and application of the student conduct process for failure to comply, endangering the well-being of others, and/or disorderly conduct. The student code of conduct applies also to participation in all virtual activities, including Zoom sessions and discussion boards.

MENTAL HEALTH & COUNSELING RESOURCES

Counseling Services provides a broad range of counseling and mental health support services that facilitate our students' personal, social, and academic development. Our licensed mental health professionals provide short-term individual counseling, group counseling, crisis intervention, and consultation to currently enrolled students for no additional charge. Counseling services are provided in a confidential and diversity-affirming environment to help students address a variety of mental health, situational, and developmental concerns. Our office is located in the Baugher Student Center, Suite 216. Appointments can be made in person or by calling 717-361-1405. Urgent walk-in services are also available. To access our after-hours crisis services, please call the 24/7 Campus Security number of 717-361-1111. For more information, please visit www.etown.edu/offices/counseling located in the Baugher Student Center, Suite 216. Appointments can be made in person or by calling 717-361-1405. Urgent walk-in services are also available. To access our after-hours crisis services, please call the 24/7 Campus Security number of 717-361-1111. For more information, please visit www.etown.edu/offices/counseling

SAMPLE LAB PROJECT

LAB #5-6 4-WAY SUPERSCALAR + VECTOR-REGISTER PROCESSOR DESIGN with Parallel Neuron Transfer Functions Circuit Simulations, Assembly language, FPGA-w/HDL, TTL Chips, and Dual PLC's

EGR430 Parallel Processing Lecture & Lab

JT Wunderlich PhD

LATE PENALTY: Minus 10% per day for each late item until last day of Final exams, then no credit

LAST REVISED:

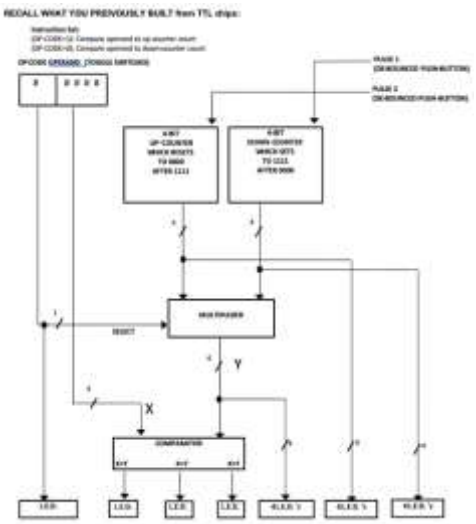
PRE-LAB

Read as needed:

- Video (26 min): Combinational Digital Circuit Logisim Tutorial (a 1-bit ALU) https://www.youtube.com/watch?v=dYZ-Hwbcnq4
- Video (37 min): 16 bit Full Adder Digital Circuit Simulation using Logisim software https://www.youtube.com/watch?v=81SrA0qSA98
- Video (65min): Synchronous Sequential Digital Circuit Logisim Tutorial (a Finite State Machine) https://www.youtube.com/watch?v=QE3byJdrWls
- 2019 Intel 8051 Microcontroller (2015, 2014, Pre-2013)
- 2019 FPGA (2018B, 2018A, 2013, pre-2013 VIDEO by Jeff Edmonds
- IBM QUALITY CONTROL / VERIFICATION of Supercomputers (your lab reports must show "verification") -- part of Dr. W.'s IBM supercomputer research & publication:
  IEEE: "Functional Verification of SMP, MPP, and Vector-Register Supercomputers through Controlled Randomness" (PPTX-w/audio,PDF,**MP4,YouTube**)
- Neural Network Processor Design Paper, with Rectified Linear Unit Transfer Function Hardware, *"Deep Learning using Rectified Linear Units (ReLU)" by A. Agarap*
- Neurocomputer Reference material from Dr. W's EGR434 Robotics & Machine Intelligence
  - Dr W Machine Learning Hardware Book Chapter & and his Two Neural Network Processor Designs (PPTX-w/audio, PDF, MP4, YouTube)
  - Dr W Machine Intelligence History -- Part of draft book chapter (PPTX-w/audio, PDF, MP4, YouTube)
  - Calculus for Machine Learning (PPTX-w/audio, PDF, MP4, YouTube)
  - Machine Intelligence Intro -- Symbolic AI vs Neural Networks. (PPTX-w/audio, PDF, MP4, YouTube)
  - Neural Network Code (MP4,YouTube)
- **2018 IC's, Circuit Trainer, and Power Supply** http://users.etown.edu/w/wunderjt/Chips_PSU_Trainer_Lab%20Manual%20Flores_Csongradi_Schick_Wunderlich%20VERSION%201.2.pdf
- **2019 Relays** http://users.etown.edu/w/wunderjt/LAB%20MANUALS%20post-2018/2019%20MANUAL%20Relays.pdf
- **2019 NanoLC** http://users.etown.edu/w/wunderjt/LAB%20MANUALS%20post-2018/2019%20MANUAL%20NanoLC%20PLC.pdf
- RESISTORS: http://users.etown.edu/w/wunderjt/333_RESISTORS.pdf **and** http://users.etown.edu/w/wunderjt/333_RESISTORS_Supplement%202.pdf
- DE-BOUNCING SWITCHES: http://users.etown.edu/w/wunderjt/333_BOUNCE_2.pdf **and** http://users.etown.edu/w/wunderjt/Debounce_toggle_Circuit_1.jpg **and** http://users.etown.edu/w/wunderjt/Debounce_toggle_Circuit_2.jpg

DURING LAB

- Implement the **Machine Instruction Set** shown
- Create **four identical Parallel SCALAR FUNCTIONAL UNITS** as shown
- Create **four identical Parallel VECTOR-REGISTER UNITS** to yield Vi, Vj, and Vk
  - created from outputs of scalar functional units
- Create **four identical Parallel NEURON TRANSFER FUNCTION UNITS**
  - Use the simple hard-limiting Perception Transfer Function, outputting a 0 or 1
  - Extra credit for a more complex transfer function.
- Create **one CODE STACK & DISPATCH UNIT**
  - Controlled by a PROGRAM COUNTER via a MASTER CONTROL unit with a Finite State Machine that dispatches one instruction at a time, sequentially, to each of the four scalar functional units; i.e. a four-way superscalar design with four pipelines (with approximately simultaneous execution)
- **Embed a carefully crafted assembly language code segment** in your CODE STACK
  -To demonstrate the functionality of your instruction set and circuitry in the minimal amount of time that you can defend as providing **Comprehensive Testing** of all scalar, vector, matrix, and neuron machine instructions and hardware.
- **Also**, **implement at least 50% of the previous lab on the right, OR at least 15% of new lab:**
  - as an equivalent **Assembly Language** routine running on the 8051 simulator, like in this VIDEO by Jeff Edmonds, but change some things to make it your own.



EXTRA CREDIT (up to 15 points): Real-Time **PIPELINE CLOCKING stressor:** Connect your previous lab **TTL and PLC** circuit up so that it not only has the one-bit Op-code cycling from one PLC, but also the 4-bit operands cycling through all 16 combination of bits, in addition to the two parallel counters simultaneously running off of the clock generator on the circuit Trainor; And push all three of these frequencies to their maximum limits and note when errors begin to occur

GRADE PERCENTAGES (submit just one video with everything in it)

(**up to 15%** EXTRA CREDIT): Demonstrate in video (3 minutes max) **PAD-234 Circuit Trainer circuits** [primarily for **5 volt** TTL chips]
AND **Phoenix Contact NanoLC PLC simulations** AND **Phoenix Contact NanoLC PLC real-time systems**

none Demonstrate in video (3 minutes max) **Old Circuit Trainer circuits** [primarily for **5 volt** TTL chips]
none Demonstrate in video (3 minutes max) **RadioShack Circuit Trainer circuits** [primarily for **~3 volt** CMOS chips]
(55%) Demonstrate in video (3 minutes max) **Logisim Circuit simulations**
Demonstrate in video (3 minutes max) **Phoenix Contact NanoLC PLC simulations**
Demonstrate in video (3 minutes max) **Phoenix Contact NanoLC PLC real-time systems**
none Demonstrate in video (3 minutes max) **Phoenix Contact Advanced Axioline PLC real-time systems (PCworks software)**
none Demonstrate in video (3 minutes max) **Phoenix Contact Advanced PLCnext PLC real-time systems**
none Demonstrate in video (3 minutes max) **2018 Field Programmable Gate Array (FPGA) simulations & Real-time system (Gate-level-Design + Waveform/Timings**
(**15%**) Demonstrate in video (3 minutes max) **2019 Field Programmable Gate Array (FPGA) simulations & Real-time system (HDL Hardware Descriptive Language)**
(new IDE needs 100 Gigabytes on hard-drive, so we first need to upgrade computers in Lab)
(**15%**) Demonstrate in video (3 minutes max) **INTEL 8051/80251 or ARM microcontroller simulations**
none Demonstrate in video (3 minutes max) **INTEL 8051/80251 or ARM microcontroller real-time systems**
none Demonstrate in video (3 minutes max) **Raspberry Pi real-time systems**
none Demonstrate in video (3 minutes max) **Arduino real-time systems**
none Demonstrate in video (3 minutes max) **Basic-Stamp real-time systems**
none Demonstrate in video (3 minutes max) **Direct PC-port real-time systems**
none Demonstrate in video (3 minutes max) **Remote mobile-device real-time systems**
none Demonstrate in video (3 minutes max) **LabView real-time systems**
none Demonstrate in video (3 minutes max) **Isolated high-voltage bench-test** (with low Voltage electronics disconnected)
(15%) Written Report
(15%) **Other: POSTER**: Use template in my public folder

MACHINE INSTRUCTION SET

SCALAR ARITHMETIC ADDITION

00h (OP-CODE = 00000000) Ri + counter#1 →Rk
01h (OP-CODE = 00000001) Ri + counter#2 →Rk
02h (OP-CODE = 00000010) Ri + Rj →Rk
03h (OP-CODE = 00000011) counter#1 + counter#2 →Rk

SCALAR ARITHMETIC SUBTRACTION

04h to 07h (OP-CODE = 000001XX) *Reserved for future subtraction instructions*

SCALAR ARITHMETIC MULTIPLICATION

08h (OP-CODE = 00001000) Ri x counter#1 →Rk, overflow →R(k+1)
09h (OP-CODE = 00001001) Ri x counter#2 →Rk, overflow →R(k+1)
0Ah (OP-CODE = 00001010) Ri x Rj →Rk, overflow →R(k+1)
0Bh (OP-CODE = 00001011) counter#1 x counter#2 →Rk, overflow→R(k+1)

SCALAR ARITHMETIC DIVISION

0Ch to 0Fh (OP-CODE = 000011XX) *Reserved for future division instructions*

SCALAR ARITHMETIC COMPARISON

10h (OP-CODE = 00010000) Compare Ri with counter#1 →Rk
11h (OP-CODE = 00010001) Compare Ri with counter#2 →Rk
12h (OP-CODE = 00010010) Compare Ri with Rj →Rk
13h (OP-CODE = 00010011) Compare Counters

SCALAR LOGICAL AND

20h (OP-CODE = 00100000) Ri AND counter#1 →Rk
21h (OP-CODE = 00100001) Ri AND counter#2 →Rk
22h (OP-CODE = 00100010) Ri AND Rj →Rk
23h (OP-CODE = 00100011) AND counters →Rk

SCALAR LOGICAL OR

24h (OP-CODE = 00100100) Ri OR counter#1 →Rk
25h (OP-CODE = 00100101) Ri OR counter#2 →Rk
26h (OP-CODE = 00100110) Ri OR Rj →Rk
27h (OP-CODE = 00100111) OR counters →Rk

CLEAR

30h (OP-CODE = 00110000) Clear Ri

VECTOR/ARRAY / MATRIX and NEURON INSTRUCTIONS

Vi, Vj, and Vk are created from outputs of the four parallel scalar functional units

VECTOR ARITHMETIC ADDITION

82h (OP-CODE = 10000010) Vi + Vj →Vk

VECTOR ARITHMETIC SUBTRACTION

84h to 87h (OP-CODE = 100001XX) *Reserved for future subtraction instructions*

VECTOR ARITHMETIC MULTIPLICATION

0Ah (OP-CODE = 10001010) Vi x Vj →Vk, overflow →V(k+1)

VECTOR ARITHMETIC DIVISION

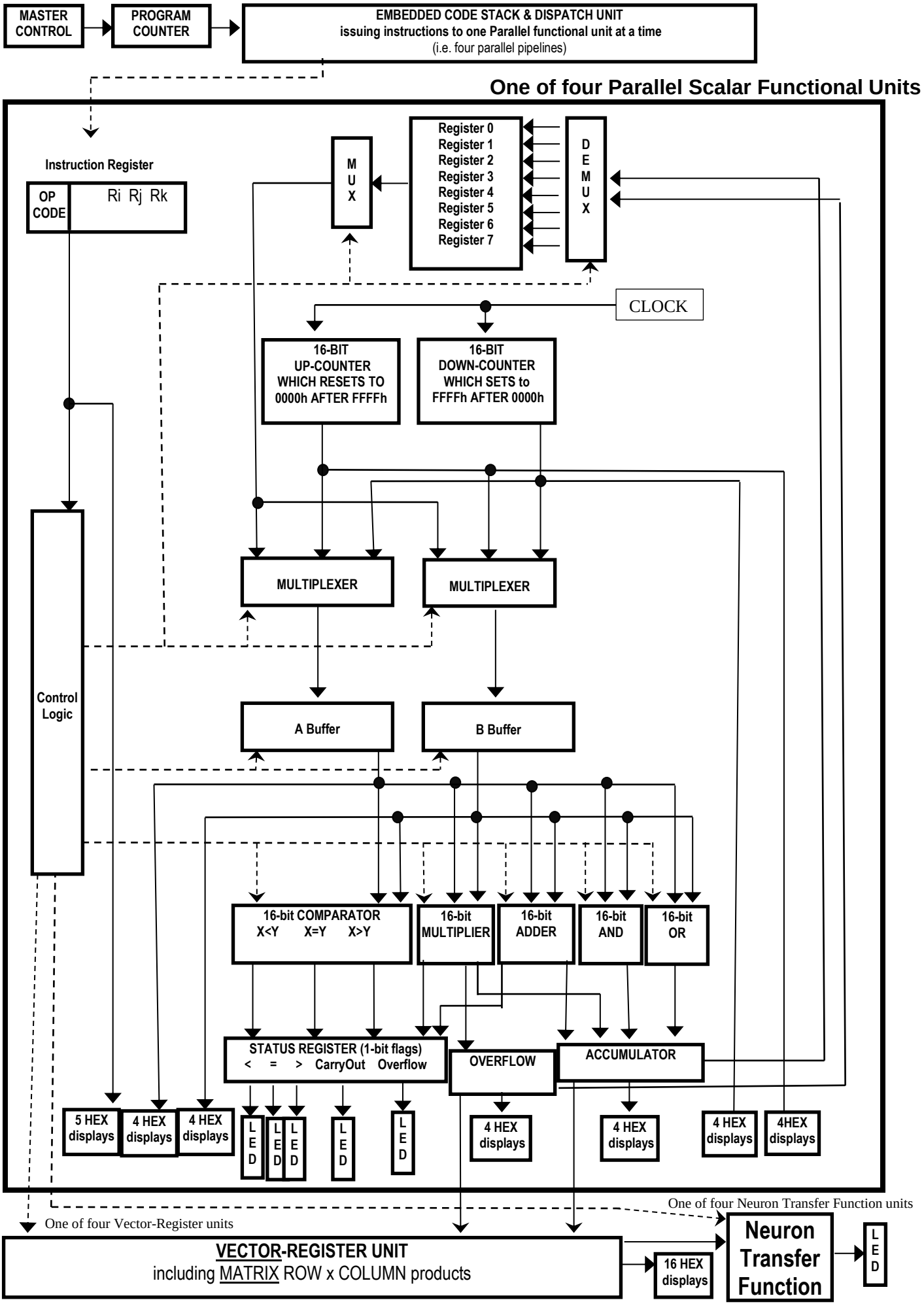
8Ch to 8Fh (OP-CODE = 100011XX) *Reserved for future division instructions*

MATRIX ROW x COLUMN (i.e., Dot-Product)

C0h (OP-CODE = 11000000) Vi x Vj →Vk, overflow →V(k+1)
Vk(1)+Vk(2)+Vk(3)+Vk(4) → 32-Bit Scalar Accumulator

NEURON TRANSFER FUNCTION

E0h (OP-CODE = 11100000) Vi x Vj →Vk, overflow →V(k+1)
Vk(1)+Vk(2)+Vk(3)+Vk(4) → 32-Bit Scalar Accumulator
32-Bit Scalar Accumulator → **Neuron Transfer Function**


MASTER CONTROL
PROGRAM COUNTER
EMBEDDED CODE STACK & DISPATCH UNIT
issuing instructions to one Parallel functional unit at a time
(i.e. four parallel pipelines)
One of four Parallel Scalar Functional Units
Instruction Register
OP CODE
Ri Rj Rk
MUX
Register 0
Register 1
Register 2
Register 3
Register 4
Register 5
Register 6
Register 7
DEMUX
CLOCK
16-BIT UP-COUNTER WHICH RESETS TO 0000h AFTER FFFFh
16-BIT DOWN-COUNTER WHICH SETS to FFFFh AFTER 0000h
MULTIPLEXER
MULTIPLEXER
Control Logic
A Buffer
B Buffer
16-bit COMPARATOR X<Y X=Y X>Y
16-bit MULTIPLIER
16-bit ADDER
16-bit AND
16-bit OR
STATUS REGISTER (1-bit flags) < = > CarryOut Overflow
OVERFLOW
ACCUMULATOR
5 HEX displays
4 HEX displays
4 HEX displays
LED
LED
LED
LED
LED
4 HEX displays
4 HEX displays
4 HEX displays
4HEX displays
One of four Neuron Transfer Function units
One of four Vector-Register units
VECTOR-REGISTER UNIT
including MATRIX ROW x COLUMN products
16 HEX displays
Neuron Transfer Function
LED

LAB RULES & PROCEDURES

IF LABS ARE BUILT (AND POSSIBLY REBUILT), BUT DON'T FULLY FUNCTION: For **demonstrations** and **reports**, deduct depends on how adequately you identify problems. For example, make test set-ups to verify functionality of isolated simulation sub-parts, chips, circuit trainer elements, software, relays, other electronics, motors or other higher-voltage circuits and devices. PROVE THAT NO EASY FIX OR SUBSTITUTION WAS POSSIBLE or EASILY IDENTIFIABLE AT THE TIME. Discuss (1) How you identified problems, and (2) How you tried to fix them. Include evidence that you fully understand and have properly connected all pins on a given chip (including considering floating-pins, powering the chip, needed pull-up resistors, proper voltage levels, etc.), and that you have exhausted much time attempted to solve all problems.

DEMONSTRATIONS Alternate team members demonstrating lab in video.

GRADING For both demonstrations and reports, a **92** is for everything done very well and professional; to get more points, enhance things in creative ways

REPORTS must include:

- **PHOTO'S OF ALL CIRCUITS AND TEST SET-UP'S BUILT**
- Title Page with lab number, name of lab, your names, Majors, Year (e.g., Junior), who is demonstrating, and who is the designated TEAM LEADER
- Sections numbered and tilted as follows (always list all of these, and simply put "NA" if not applicable):
    1. **"Assignment"** (An exact copy of everything in this document -- exactly how it looks here)
    2. **"Equipment Used"** (A list of hardware and software) **INCLUDE PHOTO'S OF ALL EQUIPMENT**
    3. **"Methodology"** (including all design steps, analysis, **DECISIONS MADE**, etc.) **INCLUDE PHOTO'S OF ALL CIRCUITS BUILT**
    4. **"Options"** (if applicable, a comparison of each method used)
    5. **"Problems Encountered"** (including any debugging methodology) **INCLUDE PHOTO'S OF ANY TEST-CIRCUITS BUILT**
    6. "**Testing Methodology**" (including timing traces, test-vectors, and **RATIONALE FOR HOW YOUR METHODOLOGY ASSURES QUALITY**)
        a. **Including estimated PROBABILITY of satisfactory coverage by chosen test vectors**
    7. **"References"** (in standard IEEE format)
    8. **"Appendices"** (for spec sheets, etc.)
- **ALL DESIGN PROCESS STEPS MUST BE INCLUDED** for Digital Logic designs (**NUMBERED** as in EGR/CS 332/330). If design step not done, list as "N.A."

    For Combinational Digital Logic Design:

    Step 1: Define problem
    Step 2: Encode variables
    Step 3: Create truth table
    Step 4: Find simplified function(s)
    Step 5: Draw logic circuit
    Step 6: Convert to NAND's
    Step 7: Check assumptions
    Step 8: Chip circuit diagram

    For Sequential Digital Logic Design::

    Step 1: Define problem
    Step 2: Create state diagram
    Step 3: Encode variables
    Step 4: Minimize machine
    Step 5: Create state table
    Step 6: Append flip-flop inputs
    Step 7: Find simplified function(s)
    Step 8: Draw logic circuit
    Step 9: Convert to NAND's
    Step 10: Analyze any unused states
    Step 11: Revise state diagram
    Step 12: Check Assumptions
    Step 13: Chip circuit diagram
- **COLOR-CODED LOGIC DIAGRAMS** are required for any digital circuit (Breadboard, FPGA, etc.)
- **COLOR-CODED CIRCUIT SCHEMATICS** are required for any circuit implemented (Breadboard, PLC, ladder logic, etc.), color is a must, hand-colored is ok
- **FLOW CHART** is required for any program
- **COMMENT EVERY LINE OF CODE**
- **TEAM LEADER** has responsibility of coordinating all equipment problems and acquisition of needed parts with TA. Try to stick with same person for this role

SCHOOL CLOSURE / CLASS CANCELATION

Additional work assigned to cover any class cancelation