Class Project

Spring 2009

ECE554

Digital Engineering Laboratory

Project Assignment

- Design a non-trivial computer with an original instruction set architecture (ISA)
- Four key requirements
  - It must be an original ISA
    - Somewhat negotiable
  - It must be non-trivial
  - It must be tractable
    - everything takes at least twice as long as you expect
  - It must interface through the serial port with the terminal emulator on the lab workstations
    - Negotiable (always useful for development/debug)
- Be creative and have fun!
- Leave helpful hints for the future – use “wiki” and update it

Sample ECE554 Computers – Part 1

- A (pipelined) general-purpose 32-bit computer with
  - A conventional ISA
    - ADD, SUB, MUL, MOV, LD, ST, JMP, etc.
    - A gimmick that makes the machine non-trivial
  - Examples of gimmicks include
    - Superscalar, VLIW
    - Multithreading
    - Pipeline with dynamic branch prediction and data forwarding
    - Out of order execution
    - Floating point support
    - Single instruction, multiple data execution
    - Specialized instructions
    - Coprocessor support

Sample ECE554 Computers – Part 2

- A programmable special-purpose processor
  - Multimedia Processor
  - 3D Graphics Processor
  - Binary Coded Decimal (BCD) Processor
  - String Processor
  - Security Processor
  - Fault tolerance support
- Examples of very good projects from previous semesters are in lab
- Feel free to look at real ISAs, but don't copy
- Make sure your processor can be implemented on the board, but don't let quirks of the system limit your creativity

Sample ECE554 Computers – Part 3

- Projects should also have significant software components. For example,
  - Software simulator for your processor
  - Assembler for your processor
  - Simple compiler for your processor
  - Demonstration and testing software
- Also good to include hierarchical memory (e.g., cache on FPGA + memory on chip).
- Most useful processors will support interrupts
- Talk to me and TA about ideas you have

Incorporating Various I/O Interfaces

- The Vertex-2 Pro FPGA board has several interfaces that may be useful in implementing or demonstrating your project
  - Video Graphics Array (VGA) for displaying graphics on the monitor
  - USB interface – Avoid since it doesn't work well.
  - PS/2 interface for mouse/keyboard
  - Ethernet interface to network
  - Audio codecs for playing music

  Become familiar with the board and its features early to know strengths/limitations.

New FPGA Boards

- The new XUP Virtex-II Pro FPGA boards have the following features:
  - Virtex-II Pro FPGA with PowerPC™ 405 cores
  - Up to 2 GB of Double Data Rate (DDR) SDRAM
  - On-board 10/100 Ethernet PHY device
  - RS-232 DB9 serial port and two PS-2 serial ports
  - AC-97 audio CODEC with audio amplifier and speaker
  - XSGA video output
  - Support for various accessory boards

    http://www.xilinx.com/univ/accessory_boards.htm
- For additional information see:
  - http://www.xilinx.com/univ/xupv2p.html

Project Milestones – Part 1

- The major steps for the project include:
  - Form teams (finish doing this today)
  - Choose one team leader (required)
  - Define the architecture
    - Will need to meet outside of lab time
    - Use brainstorming and multivoting
- Dates for all the major milestones are listed on the course syllabus
- See website for further details and previous projects

Project Milestones – Part 2

- Project Proposal (2/12)
  - 20 to 30 minute presentation (8 to 10 slides)
  - Your project's targeted application
  - Processor type (superscalar, out-of-order, dsp, etc.)
  - Peripherals you plan on using (vga, keyboard, ethernet, etc.)
  - Other features/gimmicks you plan to include (interrupts, performance counters, etc.)
  - Supporting software you plan to write (assembler, simulator, etc.).
  - Gantt chart with your tentative project timeline (Microsoft Project is useful for this)
  - Good opportunity to get preliminary feedback
  - All team members attend

Project Milestones – Part 3

– Architecture review (2/19)

- 40 minute presentation to instructors (1 hour total)
- Full ISA Specification - registers, instruction formats, addressing modes, opcodes, interrupts, exceptions, flags
- Other details – gimmicks, peripherals and I/O, memory architecture, planned software, planned user interfaces, etc.
- Group leader + other members (all attend)
- Let us know if your group needs specific accessory boards (memory, USB, video, etc.)

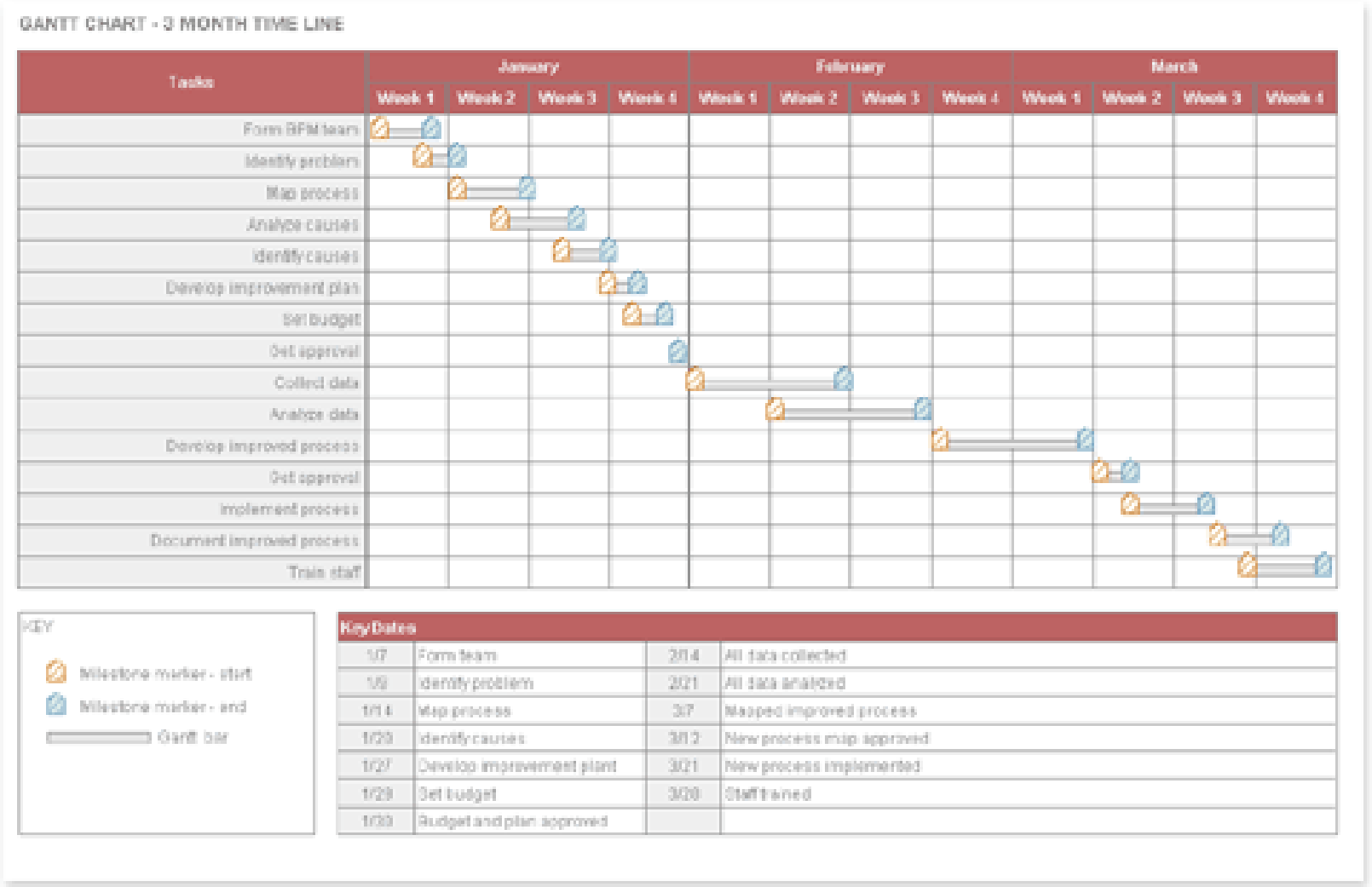
Project Milestones – Part 4

- ISA Report (2/24)
  - Description of visible registers and memory
  - Description of instruction formats
  - Descriptions of each instruction
  - Discussion of I/O interface (and interrupts)
  - Description of any gimmicks
  - Other things (e.g., code samples, software overview, diagrams that will help clarify the architecture of your processor)
- Define the Microarchitecture
  - Break the machine into subsystems
  - Carefully define the interface to each subsystem
    - signal names, directions, functionality, timing
  - Sketch out details of each subsystem

Project Milestones – Part 5

- Microarchitecture review (3/24)
  - Present your microarchitecture to the instructors (40 minutes => 1 hour)
  - Present overall microarchitecture (team leader) plus each subsystem (team members)
  - Describe any changes to ISA or gimmick
  - Project schedule
    - Major project tasks (with start and end times)
    - Project subtasks (with start and end times)
    - Team members assigned to tasks/subtasks
    - Gantt chart (www.smartdraw.com or Microsoft Project)

Sample Gantt Chart



Project Milestones – Part 6

- Testing and demo plan (4/7)
  - Present the plan for integration, debug, and developing demonstration to the instructors (20 minutes => 40 minutes)
  - Present overall integration plan (team leader) plus status of each subsystem (team members)
  - Describe what kind of demo you will have to show your projects strengths and features
    - Describe the main features
    - Status of the implementation and integration (with start and end times)
    - Potential debug and test bottlenecks
    - "What if" some thing does not work – contingency plan
    - Updated Gantt chart (www.smartdraw.com or Microsoft Project)
- Informal Progress Reviews (see course syllabus – PR)
  - Informal meetings between your team and instructors during lab time to report on progress (10 to 20 minutes)
  - Driven by project schedule
    - Level of task/subtask completed to date
    - Recovery/fallback plan to get back on schedule
    - Chance to address problems and concerns

Project Milestones – Part 7

- Logic Design and Simulation
  - Begin after microarchitecture review (not necessarily)
  - Have each team member design a subsystem (two members very familiar with each subsystem)
  - Simulate each subsystem extensively (unit test)
  - Test subsystems on the board (don't wait till the end)
- System Integration
  - Begin after individual subsystems have been tested
  - Bring all subsystems together into a single module to run functional simulation, synthesis, implementation, and timing simulation
- Software Development
  - Typically start after ISA has been defined
  - Use to diagnose and demonstrate processor (e.g., simulator, assembler, assembly code, demo software, test generators,)

Project Milestones – Part 8

- Test and Debug (figure out everything you did wrong)
  - Architecture, subsystem, or logic errors (e.g., timing problems, gated clocks, asynchronous operation)
  - Synthesis/implementation problems (e.g. missing circuitry, incomplete routing) – examine reports
  - Download problems (e.g., poor VCC, GND, or CLK) – scope board and/or download simple design
  - Bad chips or boards
- Project demonstrations (5/5 in lab)
  - 20 minute overview of architecture & microarchitecture
  - 25 minute system demonstration (have software that thoroughly demonstrates the system)
  - Be ready for lots of questions and on-line evaluation

Project Milestones – Part 9

- Final report (due 5/14 by Noon)
  - Overview
    - Summarize architecture, microarchitecture, implementation, and software
  - Principals of Operation
    - Complete description of ISA
  - Machine organization
  - Detailed design for each subsystem
  - Annotated simulation results
  - Implementation reports (selected pages)
  - Description of software
  - Contribution of each team member (signed by all team members)
- See lab for previous final reports
- Catch up on sleep!

Teams in ECE554

- Each team should designate one team leader who helps the group stay organized and is the main contact person
- All team members must participate – if you are having problems with members not participating notify us early
- Keep the lines of communication open amongst all group members and with instructors
- Missing lab time more than once or twice is not acceptable. Any missed lab time should be made up.
- You will need to put in significant amounts of time outside of scheduled lab times. Start on projects early!
- Get feedback before project proposal and architecture/microarchitecture review
- Teams should cooperate, not compete

Brainstorming

- Goal - to examine as broad a range of options as possible
- Rules - Encourage free-wheeling - No discussion - No judgment - Allow hitch-hiking - Write visibly all ideas
- Sequence
  - Review the topic (as a question)
  - Minute or two of silence to think
  - Call out and write down ideas

Multivoting

- Goal - Select most important or popular ideas from a list with limited discussion and difficulty.
- Generate list and number.
- Combine similar items if agreed.
- If necessary, renumber.
- Have all members vote for several items to discuss by writing down numbers; about 1/3 of items per member.
- Tally votes using secret ballot if necessary.
- Eliminate items with fewest votes (less than about 25%).
- Repeat until only a few items - if no clear favorite discuss or vote again.

Evaluation in 554

- Individual
  - Effort Report in Final Report
    - Not just something you submit, but consensus of team members
  - Project Log
    - A detailed record of activities and accomplishments
    - Keep on-line (electronic copy) and up-to-date
    - May be requested periodically or at end of course
- Team
  - Project Demonstration
    - Technical Content and Presentation
  - Project Final Report
    - Technical Content and Presentation