Guidelines for ECE337 Project Ideas

(10 points towards total project documentation score)

1. Choosing a Project Idea:

Project ideas will be judged on the following criteria:

- There needs to be some reasonably compelling reason why the selected idea would not or could not be done in software on a microcontroller, such as the HC12 you work with in ECE362.
- The idea needs to be useful as an ASIC for reasons due to requirements for small size, low power, and/or high performance.
- The idea needs to be something you have a chance of completing in the available time. This hard to judge up front, so it is a good idea to keep in mind some ways that one can either simplify or expand the design if it turns out to be either too complicated or too simple.
- The proposed design needs to make use of moderately complex standard interface formats such as MIDI, I2C, USB, etc. or it needs to be designed to work with commercial parts that make use of common interface formats that are de-facto standards such as the serial or parallel interfaces used with LCD controllers.
- Some project ideas may be rejected because they have been done many times before. Or, if it is an idea that has been done many times, we will require you to incorporate features that will make your design unique.
- In many cases, your choice of project will require you to study a topic, an interface standard, or existing commercial devices that you haven't already encountered in ECE337 or prior courses. On the other hand, other courses could be a good source of inspiration for projects that would be useful as an ASIC.

2. Some general categories of project ideas that are appropriate for ASIC or FPGA

- Implementation of functions that would be useful in a larger chip as add-ons to an existing CPU design. ARM is an extremely popular processor in industry used for this kind of things. Some examples of useful add-ons include: an encrypted memory interface, some kind of image processing or signal processing accelerator, a digital video interface, music or audio synthesis hardware, interface modules for a wide range of industry standard busses - USB, CAN, Ethernet, etc.
- Image processing functions including blurring, sharpening, contrast enhancement, effects, perhaps for use in a video or still camera.
- Signal processing functions such as equalizers, noise reduction, spectral analysis (FFT), cross-correlation, distortion analysis.
- Encryption, decryption (have been done a lot)

- Adapters or translators to go between different bus standards. One example - convert between USB and I2C.
- Very small scale 2D or 3D graphics processing units (GPUs). Modern GPUs go way beyond what can be done in ECE337, but one could implement GPU functionality on a small scale.
- Some kind of very high-speed feedback control system

3. Interface Standard and Algorithm Difficulty/Workload Estimates

- Serial Communication Standards
  - USB (Physical Layer only):
    - 2 team members (when there is no USB design lab)
    - 1.5 team members (when there is a USB design lab)
  - USB (up to Bulk Data Transfer):
    - 3 team members (when there is no USB design lab)
    - 2.5 team members (when there is a USB design lab)
  - UART: 0 team members given UART design lab
  - SPI: ~0 team members
  - I2C:
    - 1.5 team members (when there is no I2C design lab)
    - 1 team members (when there is a I2C design lab)
  - MIDI:
    - 2 team members (when there is no I2C design lab)
    - 1.5 team members (when there is a I2C design lab)
- Direct Storage Device Standards
  - SD Card (Normal Speed/Single SPI): 1.5 team members
  - SD Card (High Speed/Quad SPI): 1.75 team members
  - IDE/PATA: ~0 team members (Digital logic side is pretty trivial)
  - SATA 1.0 (w/ slowed data rate): 3.5 team members
- Direct Memory Standards
  - Asynchronous SRAM: 0.25 team members
  - Synchronous SRAM: 0.25 team members
  - SDRAM: 3 team members
  - DDRAM: not feasible with tech library we have
  - NAND Flash (simple direct mapped): 3 team members
  - NAND Flash (simple log-based remapping): 4.5 team members
- System-on-Chip (SoC) bus standards
  - Altera Avalon (Pipelined): 1 team member
  - ARM AMBA APB: 0.5 team members
  - ARM AMBA AHB-Lite: 1 team member
  - ARM AMBA AXI: 1.5 team members

- Common Computation Algorithms
    - IEEE 74 STD Floating-Point ALU
        - ADD/SUB: 1 team member
        - MULT/DIV: 1 team member
        - SIN/COS: 0.5 team members
    - >32 & <256 Point FFT: 3 team members
    - 256+ Point FFT: 4 team members
    - 2D GPU (Basic Shapes): 2 team members
    - 2D GPU (w/ Alpha & Layer Blending): 3 team members
    - 3D GPU (simple fixed function pipeline through surface fill): 3 team members
    - 3D GPU (simple w/ multi-core): 4 team members
    - Huffman Encoding: 3 team members
    - Huffman Decoding: 2 team members
    - Simple Stream Compression: 1 team member
    - 1-D Integer Convolution: 1 team member
- Image Processing Algorithms
    - MP3 CODEC: infeasible
    - JPEG CODEC: 5 team members
    - Edge Detection (Canny, Sobel): 3 team members
    - Photo Die Demosaic: 2.5 team members
- Encryption/Decryption Algorithms
    - RC4: 1 team member
    - RC5: 2 team members
    - 3-DES: 2 team members
    - AES: 3 team members
    - Elliptical Curve (slow): 1 team member
    - Elliptical Curve (fast): 2-3 team member (Depending on target throughput)
- Display Standards
    - DVI (640x480 @ 60Hz): 2 team members (higher resolutions not feasible with course's tech/cell library)
    - VGA (640x480 @ 60Hz): 1 team member (no change due to resolution changes)
    - LCD (SPI): 0.5 team members
    - LCD (I2C): 0.5 team members + I2C rating

4. Past project ideas that ECE337 students have implemented:

- USB Flash Drive Controller
- Peer-to-Peer USB Data Switch
- Encrypted Ethernet (IEEE 802.3i) Controller
- Secure Digital (SD) Card RAID 4 Controller
- Sobel Edge Detection ASIC for Image Processing
- Canny & Sobel Edge Detection and Tracing ASIC for Image Processing
- USB Encryptor ASIC
- ARM AXI Memory Bus Encryptor ASIC
- Secure Digital (SD) Card to DVI ASIC
- Floating Point (IEEE74 std) Co-Processor
- A Packet Sniffer for packet logging and storage in a 100Mbps Ethernet medium
- Secure Digital (SD) Card Reader
- Secure Digital I/O (SDIO) to USB Adapter
- DVI Monitor Tester ASIC
- 3D Graphics Processing Unit (16-bit Fixed Point math only)
- Finite Impulse Response (FIR) Filter
- A custom Ether Gauge (that monitors the physical layer and checks for network activity, collisions, frame counts etc...)
- Huffman Encoding ASIC - Accept ASCII Data, Compress using a static binary tree
- A variety of FFT implementations for spectral analysis
- DVI to SD Card Frame Capture ASIC
- 100 Mbps Ethernet Controller SoC IP module

5. Project Idea Submission Requirements

5.1. Description of Idea (~1/2 page single spaced) (5 points)

Select partners to form a team of three to four people. Think up at least one idea for your semester project per team and prepare a one paragraph description per idea. Each description should discuss the following:

- Major functions to be performed by the chip.
- Who would want to use this chip and for what purpose?
- Why is this idea more appropriate for an ASIC (or FPGA) than for a microprocessor based design?

Your project will be expected to make use of some kind of industry standard data interface. You can re-use some of what you did for labs 5 or 6, but this will not count towards the required complexity of the design.

The minimum complexity (not including what you are able to re-use from labs 5 or 6) should be at least as much as the the USB Receiver/Transmit-only I^2C Slave you will be completing in lab 6. This can be a hard thing to judge up front, but when your TAs and instructor review the project idea they will give advice on expanding or simplifying your idea.

Please also keep in mind that while you will choose your I/O pinout and area constraints for the design, they must be approved by the course staff and need to be efficient and tight and justified. In addition, most of you don't know enough yet to judge the impact of the area constraint.

The due date is specified in the course syllabus.

Your instructor and TAs will return comments to you on the merits of your potential designs before you begin to write the actual project Proposal.

If you aren't able to find a team in time for this assignment, you need to submit a project idea of your own in this same format to receive credit. If necessary, course staff will work with you to find a team.

5.2. High Level Block Diagram **(up to 1 page) (4 points)**

Along with your project description, prepare a very preliminary high-level block diagram of at least one of your project ideas. Typical blocks in your diagram could include things like:

RS232 Serial interface

16 bit integer multiplier

Keypad decoder

Display decoder

Control Unit (probably representing a state machine and some timers)

Configuration Register

RAM interface

On the diagram, indicate the general type of external devices to which some of the blocks will be connected. Between blocks and for the external interfaces, indicate your best wild guess as to where connections will be required and the purpose of the connection. Following is a very simple example for a stopwatch (not a project idea that you will actually be allowed to use)

5.3. Format for each project idea: (1 point)

- Title of the idea in 14 point font centered at the top of the page
- List of partner names in 14 point font on separate lines with signatures of each partner to the right
- One paragraph description of project idea, single spaced, 12 point Times Roman or similar font.
- Computer generated architectural block diagram inserted into the document.
- The final document must be in pdf format.
- Total pages per project idea, 1 or 2 pages.