Project 1

Project 1 Objectives

At the end of this project, you should be able to:

- create simple microcontroller-based circuits,

- read digital information from a switch,

- compute errors between desired and actual state variables, and

- convey information about sensors using a set of LEDs.

Part 1: Circuit

- Mount Arduino board to your solderless breadboard. Mount both to frisbee

- Connect and mount the compass. Compass must be 12" above the ground (at least)

- 4 LEDs in circle: will use to display heading or heading error

- 10-LED bar: will use in next project to display distances

- Add a switch

Part 2: Compass Interface

Must implement:

- int16_t get_heading(void)
  - Returns heading in 10ths of a degree.
  - Range: -1799 to 1800 (0 is North)
- Main function: include a while(1) loop that:
  - Gets the heading
  - Displays the heading (part 3)
  - delay_ms(100)

Part 3: Sensor Processing and Display

Must implement:

- int16_t compute_heading_error(int16_t heading, int16_t heading_goal)
  - Returns the difference between **heading** and **heading_goal**
  - Return value range: -1799 to 1800 (0 means heading is at heading_goal)
- void display_heading(int16_t theta)
  - Changes the 4 LEDs to indicate theta
  - Must encode at least 8 different orientations

Part 3: Sensor Processing and Display

Must implement:

- In main function: switch state determines whether you display the heading or the heading error

- You may hard-code the heading goal in your main function

Part 4: Hovercraft

- Mount lift fan
- Start mounting batteries and other lift fans

Demonstration/Presentation

- All group members must be present
- 4-5 slide presentation (see project spec)
- Demonstration
- All group members must be able to answer questions about the hardware or software
- Code review

Code

- Check in code to subversion tree (it should be clear which files are for project 1)

- Code must be documented (see project specification for an example)

Other Components into Subversion Tree

- Presentation file

- Circuit diagram
  - Must use EagleCad
  - Deadline for training is Friday (each group must send one envoy to Andy for this). Also – get hyperterm or "screen" training if you have time

Personal Reports

- Next week you will receive a request from catme to fill out an evaluation of you and your group

- Must be filled in to receive project grade

Group Grade

- 35%: Project implementation

- 30%: Demonstration/presentation of working project (to either of the TA or the instructor)

- 35%: Code documentation and circuit diagram

Personal Programming Components

- Must accumulate at least two during the semester
- To receive credit, you must be the primary designer, implementer and debugger of the component
- Your other group members should still help!

Personal Grade

- For all parts not including the personal programming components: each group member should contribute about equally

- If this is the case, then your personal grade will be equal to the group grade

- If not, then the personal grades will be adjusted appropriately

Next …

- Finish sign-offs of:
  - Subversion
  - Programming the Atmel processor
  - Group: attach LEDs to the Atmel and control them (could be the same ones for the project)

- Start on project 1