Project 1: Addressing Multiple Digital Lines
Questions?
Project 1

• Hardware:
  • Wire in a set of LEDs
  • (leave room for future components)

• Software:
  • Provide interface functions for the LEDs
  • Write a test main() function
Project 1: Heading Display

4 LEDs in a circle:
• Represent heading with 8 different illumination patterns

• Interface function:
  
  ```
  void display_heading(int16_t heading)
  ```

  • Heading = 10ths of a degree. Value between -1799 and 1800
  • Left-handed coordinate system

  • Do not deviate from this specification!
Project 1: Heading Velocity Display

10 LEDs in a line (use bar graph):
• Represent heading velocity
• Interface function:

```c
void display_heading_velocity(int16_t velocity)
• Velocity = 10ths of a degree/sec.
  Value between -3000 and 3000
• Illuminate the two center LEDs if rate is near zero
```
Project 1: Test Function

• Add switch to circuit

• In `main()` : while(1) loop
  • One switch state:
    • Slowly increment a simulated heading from -1799 to 1800
    • Display heading
    • When heading reaches 1800, reset to -1799
  • Other switch state:
    • Slowly increment heading velocity from -3000 to 3000
    • Display
    • When velocity reaches 3000, reset to -3000
Code Specifications as Contracts

• You are implementing code that will be used for future projects and by your other group members

• Implement exactly the functions that we ask for
  • Name of function must be as specified
  • Parameter types and names must be as specified
  • Return values must be as specified
Documentation

• Project-level documentation
  • Top of each C (and H) file
  • Project #
  • Date
  • Group number
  • Group members
  • Group member responsible for the software
Documentation

• Function-level documentation:
  • Summarize what the function does in a sentence or two
    • This is for future users of your function
  • Explicitly document the inputs and outputs of the function
    • Include variable names and meaning of the variables
    • Discuss any other effects that the function has (e.g., changing pin state)

• In-Line documentation:
  • Document the *meaning* of individual lines of code or small groups of lines
  • Document what you are doing and why

• See the project 1 specification for a link to an example

Andrew H. Fagg: Embedded Real-Time Systems: Digital IO
Project Groups

- Use assigned groups
- For each project, one person must take the lead on the software
Due by the Project Deadline

• Documented code (C, and eventually H, files) checked-in to the subversion tree
  • Useful for sharing with us and with your group members

• Demonstration/code review with me or the TA
Due Shortly after the Project Deadline…

Personal Report: Catme will ask you to fill out a survey
• This will be used to provide feedback to you and your group members
• I will also use this to detect asymmetries in group member participation
Grading

Personal programming component:
• Each group member must collect 3 personal programming components over the semester
• One is available for each of the 10 projects
• Grading is generally “done” / “not done”
Grading II

Group grade

• Assess circuit, program, functionality and documentation
• See the rubric for details

• In most cases, the group grade will be given to each group member
  • In cases of significant, repeated asymmetries, grades will be balanced to reflect contribution
Hardware to Borrow

- Double-stick tape
Next Time

Timers and counters