Final Design Project: Multiple Object Tagging

Locating and reaching multiple objects in a given area

ECE2031 Fall 2016

Design Project Motivation

- ECE 2031 includes the sophomore-level team design experience
- You have developed a useful set of tools, including an entire computer within the DE2 board
- Using tools creatively to solve problems is what engineers and computer scientists do

ECE 2031 Project Components

- Propose a solution to a problem:
  - Given an area containing an unknown number of objects, how can the DE2Bot locate and reach as many as possible in a given time?
  - More details later in this presentation
- Implement the proposed design on the DE2Bot
- Demonstrate, present, and document your solution

Technical Communication Aspects

- Your project includes three major UPCP assignments:
  - A proposal outlining what you intend to develop
  - An oral presentation of your design and results
  - A final design report
- You will also maintain a design logbook using forms provided by the UPCP
  - Specific requirements will be detailed on Piazza

ECE2031 DE2Bot Past Projects

- Position/velocity feedback from wheel encoders
- Open-loop velocity control with PWM
- Processing of sonar transducers, and wall-following demonstration
- I^2C interface for battery monitoring and audio codec control
- Odometry (position estimation from wheel rotation)
- Audio codec interface and digital sound generation
- Robot Self-test
- “Corral” localization and navigation demonstrations
- Infrared signal detection and “remote control” demonstration
- UART for wireless communication and “warehouse robot” demo
- Implementation of hardware interrupts for SCOMP
- Complex mathematical functions in software (ATAN)
- Explorations of point-to-point movement methods
- **This semester: analyzing sonar data to locate objects**

Current Project Motivation

- Being able to interpret data about the environment is a central need for a mobile robot
- Several previous projects have used the sonar to:
  - Precisely follow a wall without odometry
  - Interpret the shape of the environment
  - Detect the presence of objects at known locations
- You will be looking at more complex situations
  - Future students will benefit from your findings about what does and does not work with this robot and these sensors
- Locating objects quickly and robustly will be beneficial to your final results

Your Design Task for Fall 2016

- Locate all objects within a prescribed area
  - Area will be unobstructed except for the objects
- ‘Tag’ each object by touching it and then returning to a ‘home’ area
  - The tagged object will be manually removed once the robot returns home
  - Only one object may be tagged at any time
- Demo score will be based mostly on the number of objects tagged in the allowed time
  - Specific demo and scoring details are described in a separate document

Design Space (factors that drive design choices)

Locating objects:

- Will you try to fuse data from multiple sensors, or focus on data from only one sensor?
- Will you try to keep track of multiple object locations (i.e. create a map), or only find one at a time?
- If you create a map, will you use an occupancy grid map or other method, and how will you resolve the many design choices and technical hurdles inherent in mapping using sonar sensors?

Movement:

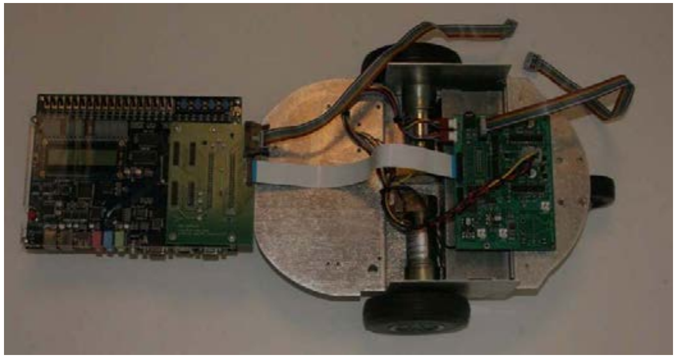
- How will you balance searching for objects with reaching them?
- How will you correct for imperfect motor control and odometry?

Hardware modifications:

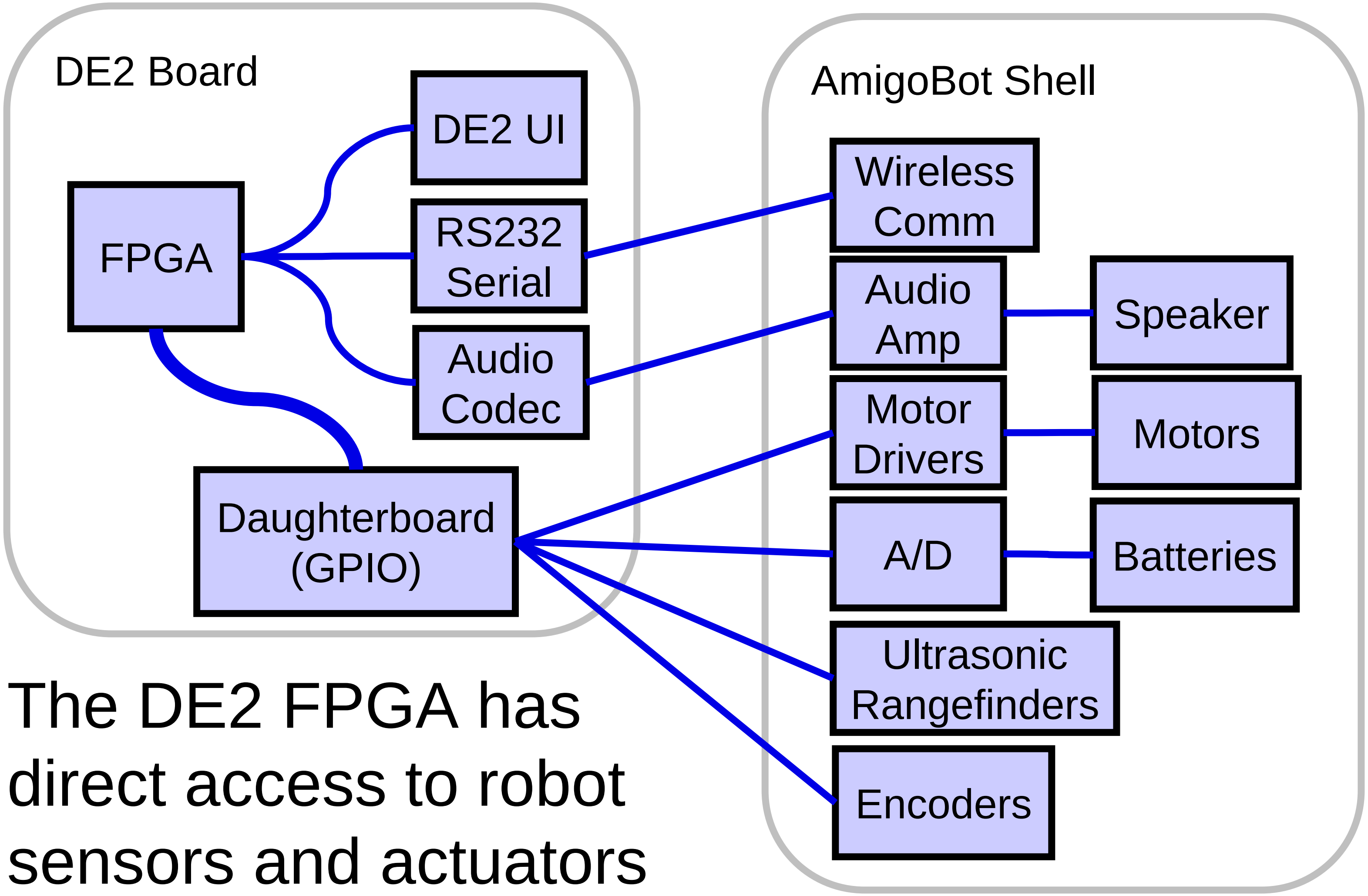
- Is there anything useful to add to SCOMP or the peripherals?

Background on DE2Bot

- In Summer 2010, older lab robots were gutted, adding a new internal controller board and a connected DE2 on top
- Beginning Fall 2010, each semester a new capability has been added, or a new application has been demonstrated

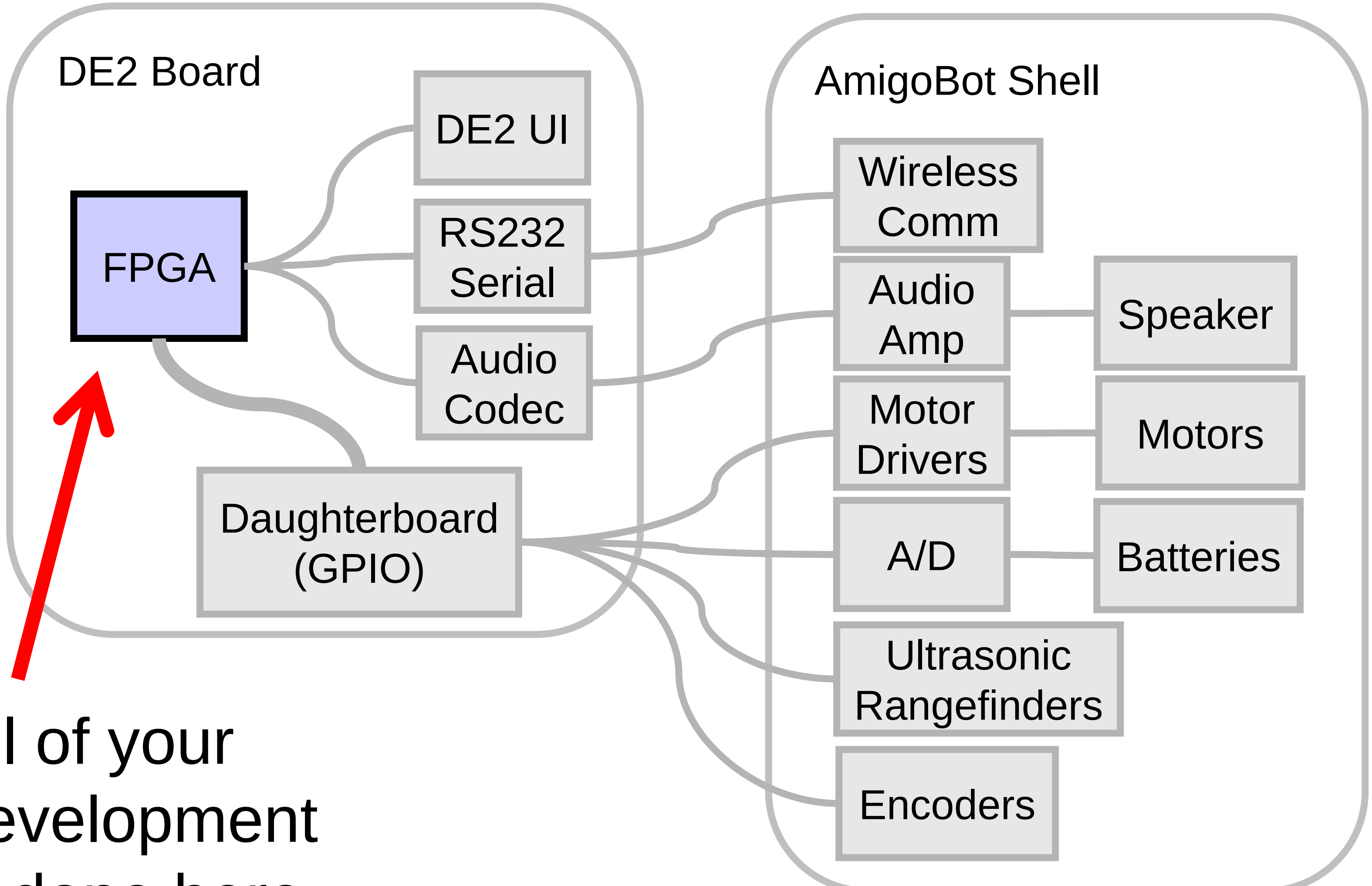


DE2Bot Hardware Architecture



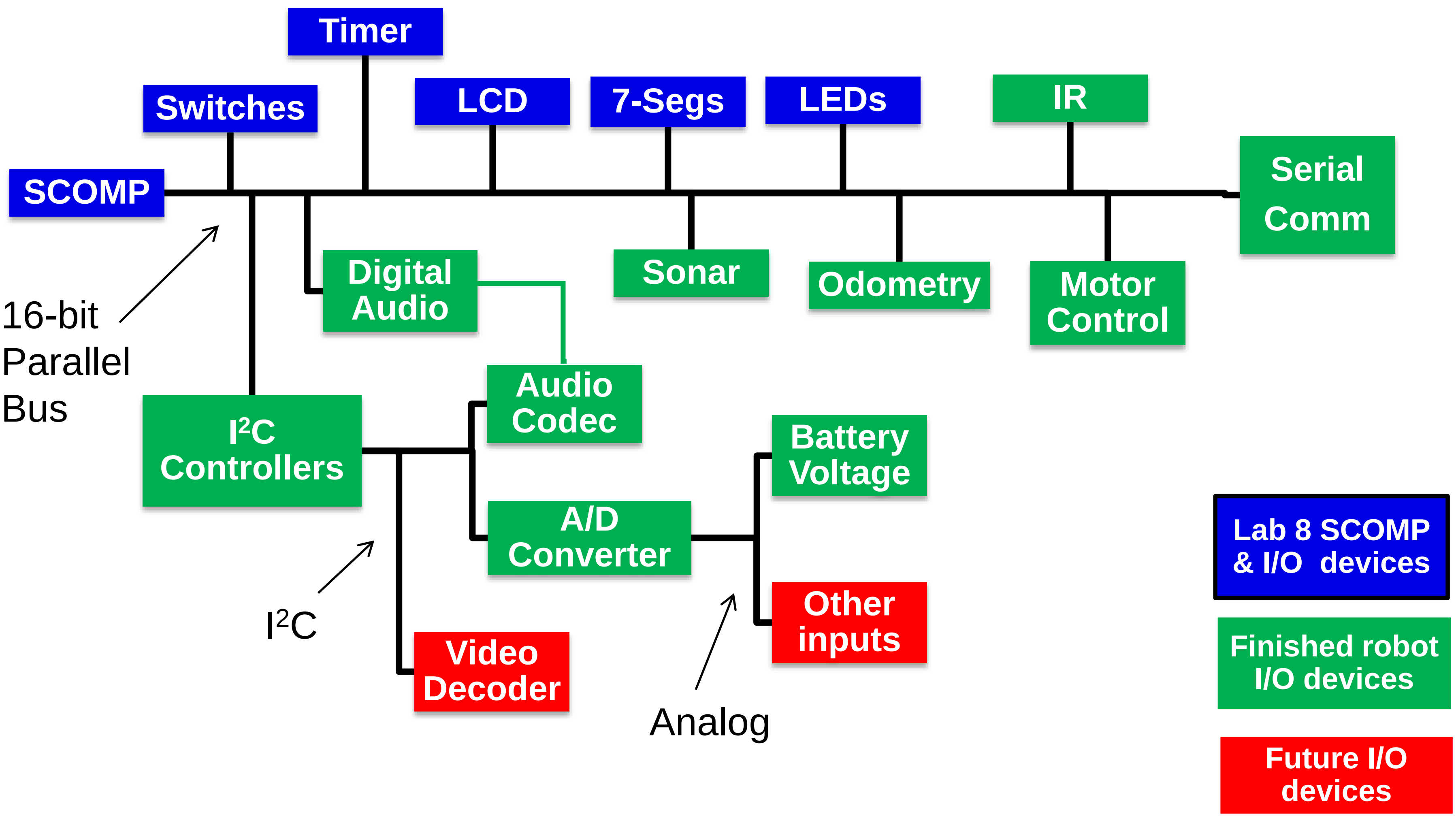
- The DE2 FPGA has direct access to robot sensors and actuators

Project Development



All of your development is done here

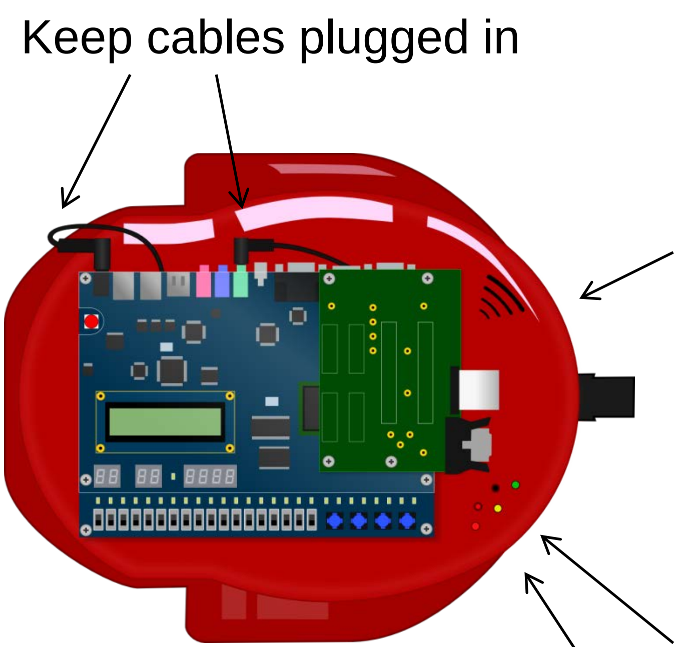
DE2 and FPGA System Architecture



Working with Complex Systems

- When something doesn't work, the robot gets blamed
- It is true – equipment CAN fail, but almost all problems are user-related:
  - FPGA design (possibly in bdf, possibly in VHDL)
  - SCOMP code (assembly errors can be elusive)
  - Careless errors (code not compiled, variables not initialized, something not reset)
- Use the robot self-test to identify any hardware problems with a robot, and show this to a TA

Robot Operational Details



- ONE robot per team at a time
- Teams in the current section ALWAYS get a robot
- Power switch is on bottom (directly beneath speaker)
  - When robot has power, DE2 has power, too
- Turn robot off when not in use, to conserve battery
- LEDs indicate hardware status (see manual for details)
- Red and black pushbuttons are not used at this time

DE2Bot I/O Devices

- Your program interacts with the robot using IN and OUT instructions to peripheral devices
- The downloadable **DE2Bot Manual** includes detail about every I/O device

Name	IO Address	IN/OUT	Description
SWITCHES	0x00	IN	Read DE2 switch
LEDS	0x01	OUT	Write to DE2 LE
TIMER	0x02	IN/OUT	Read 10Hz timer
XIO*	0x03	IN	Read PB3-PB1,
SSEG1	0x04	OUT	Write to left 4-di
SSEG2	0x05	OUT	Write to right 4
LCD	0x06	OUT	Write to LCD (16
XLEDS	0x07	OUT	Write to DE2 LED
BEEP	0x0A	OUT	Write 1-7 for be
CTIMER	0x0C	OUT	Configurable ti
LPOS*	0x80	IN	Read the current
LVEL*	0x82	IN	Read the curren
LVELCMD*	0x83	OUT	Write the desir
RPOS*	0x88	IN	Read the current
RVEL*	0x8A	IN	Read the curren
RVELCMD*	0x8B	OUT	Write the desire
I2C_CMD*	0x90	OUT	Write configurati
I2C_DATA*	0x91	IN/OUT	Read or write d

Position Control in the DE2Bot

- You can read the (X, Y, θ) estimate from the odometry module, and the cumulative rotation counter of the wheels
- The only things you can control are the velocities of the left and right wheels
- Neither the feedback nor the control are perfect, which you will have to account for if you want your project to be robust

Odometry in the DE2Bot

- The wheel position encoders provide total distance moved by each wheel. They can not keep track of X, Y, and heading of the robot as a whole.
- A separate I/O device (from ECE2031 Summer 2012) added this capability:

$$x_i = x_{i-1} + \Delta U_i \cos \theta_i$$
$$y_i = y_{i-1} + \Delta U_i \sin \theta_i$$

where

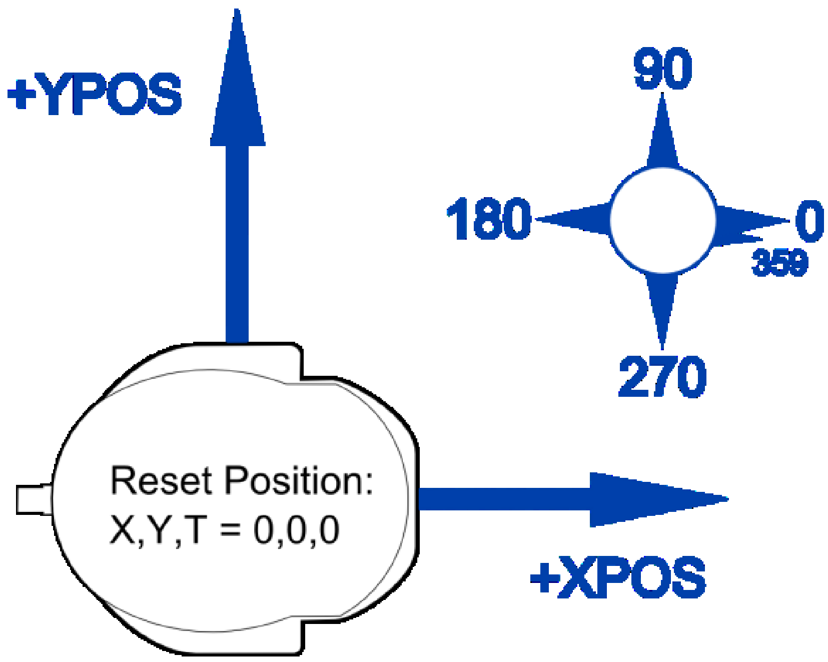
θ_i = heading of robot at instant i,

ΔU_i = distance travelled in i^{th} interval,

x_i, y_i = relative position of the robot's centerpoint c at instant i.

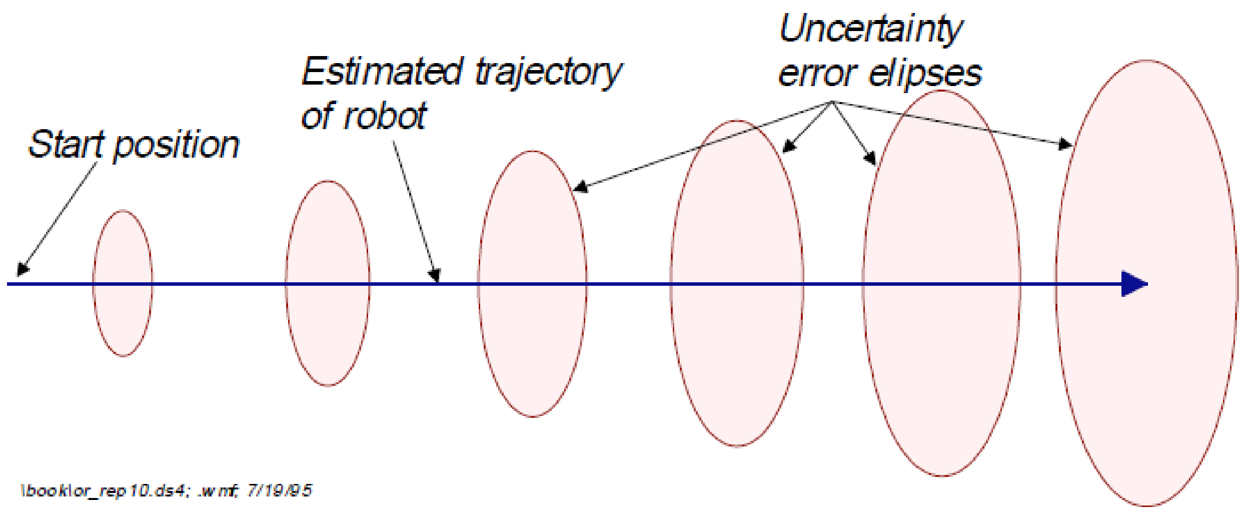
Odometry Coordinate System

- At reset, the front of the robot is defined as the positive X direction, and positive Y is to the robot's left.
- θ increases CCW, as would be the normal convention.
- For as long as you don't reset it, the odometry module will estimate movement relative to the reset position.



Limitations of Odometry

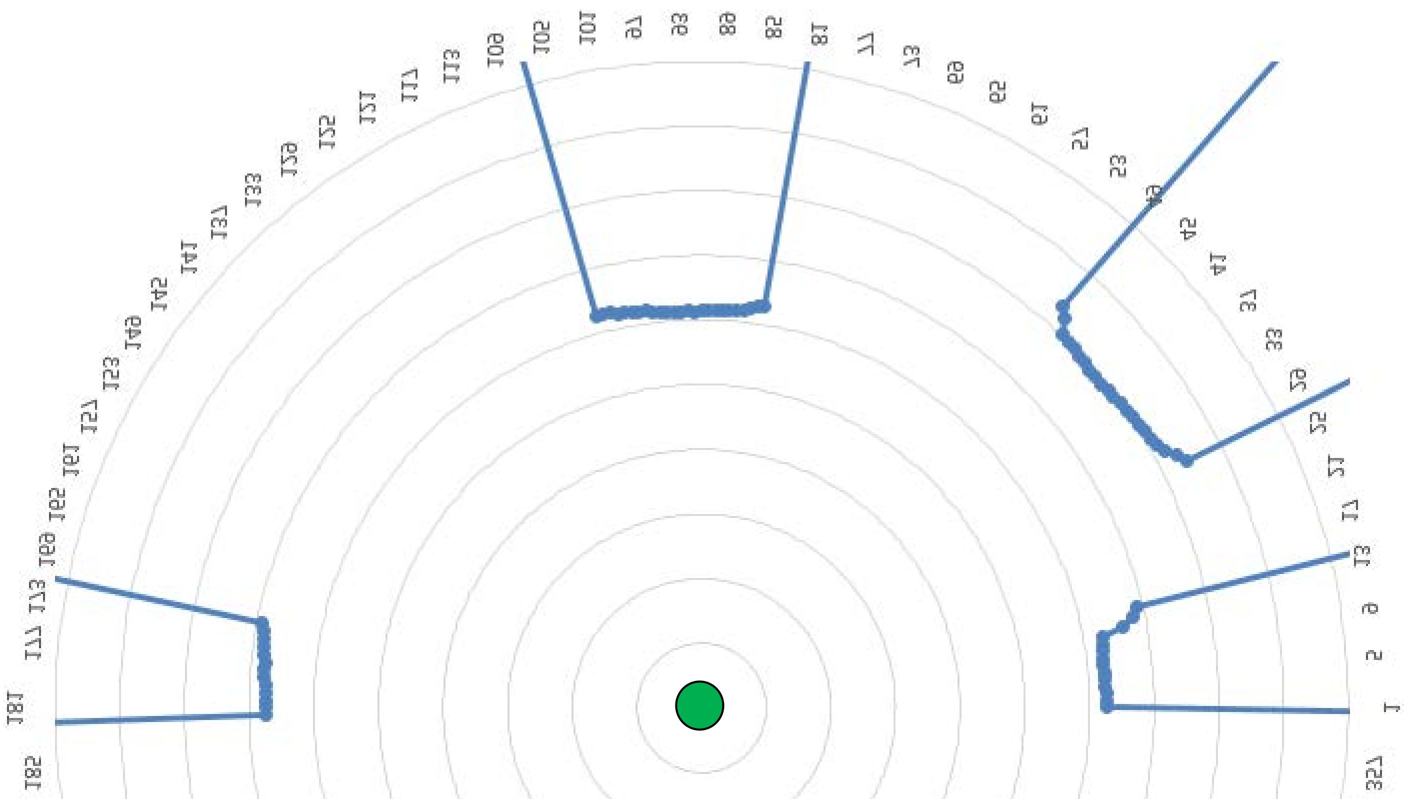
- Errors accumulate as robot moves
- Inaccurate heading (orientation) results in incorrect incrementing of X, Y components



J. Borenstein, *Sensors and Methods for Mobile Robot Positioning*, 1996.

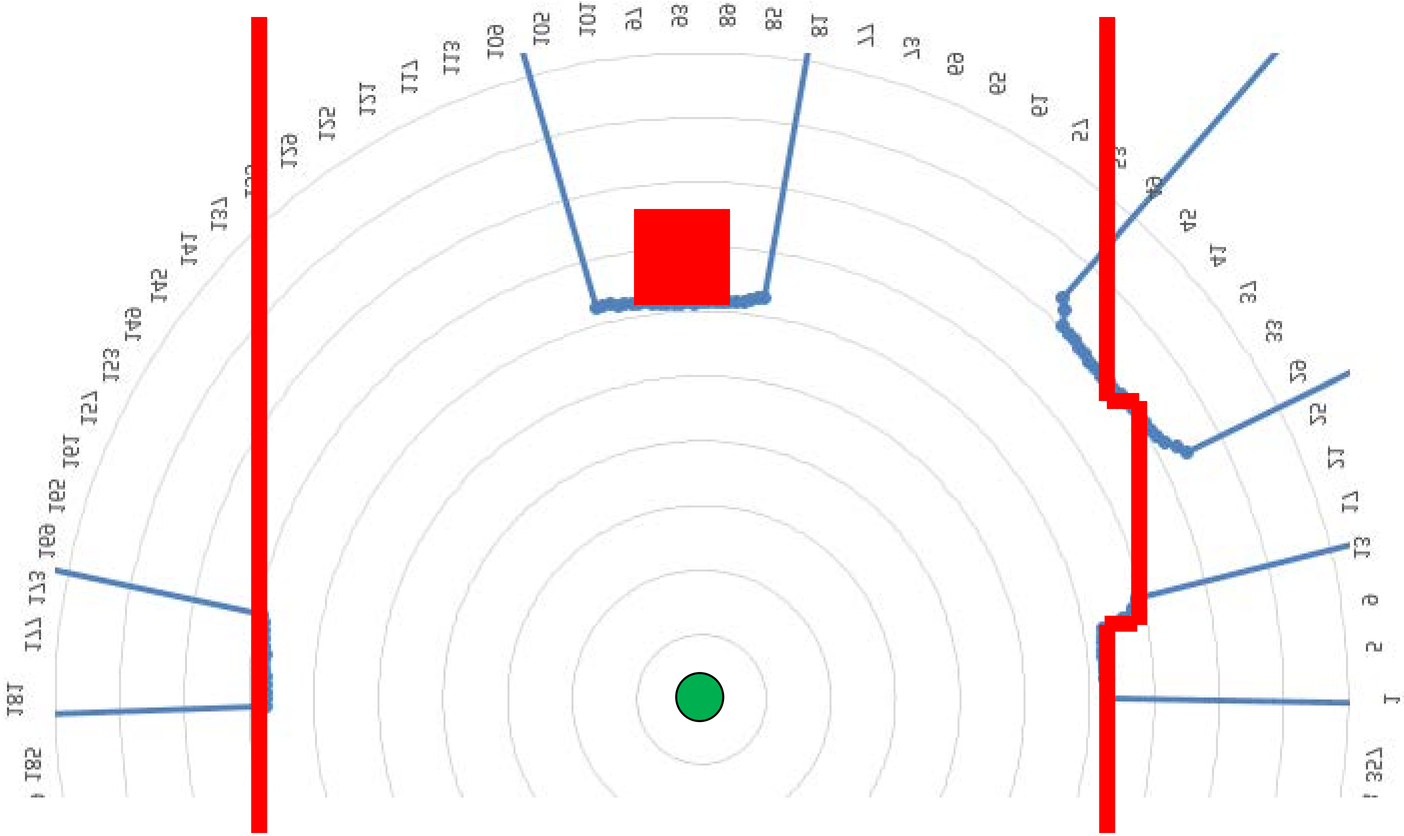
Interpreting Sonar Data

- You already know that sonar sensors aren’t perfect, which means you need to test.



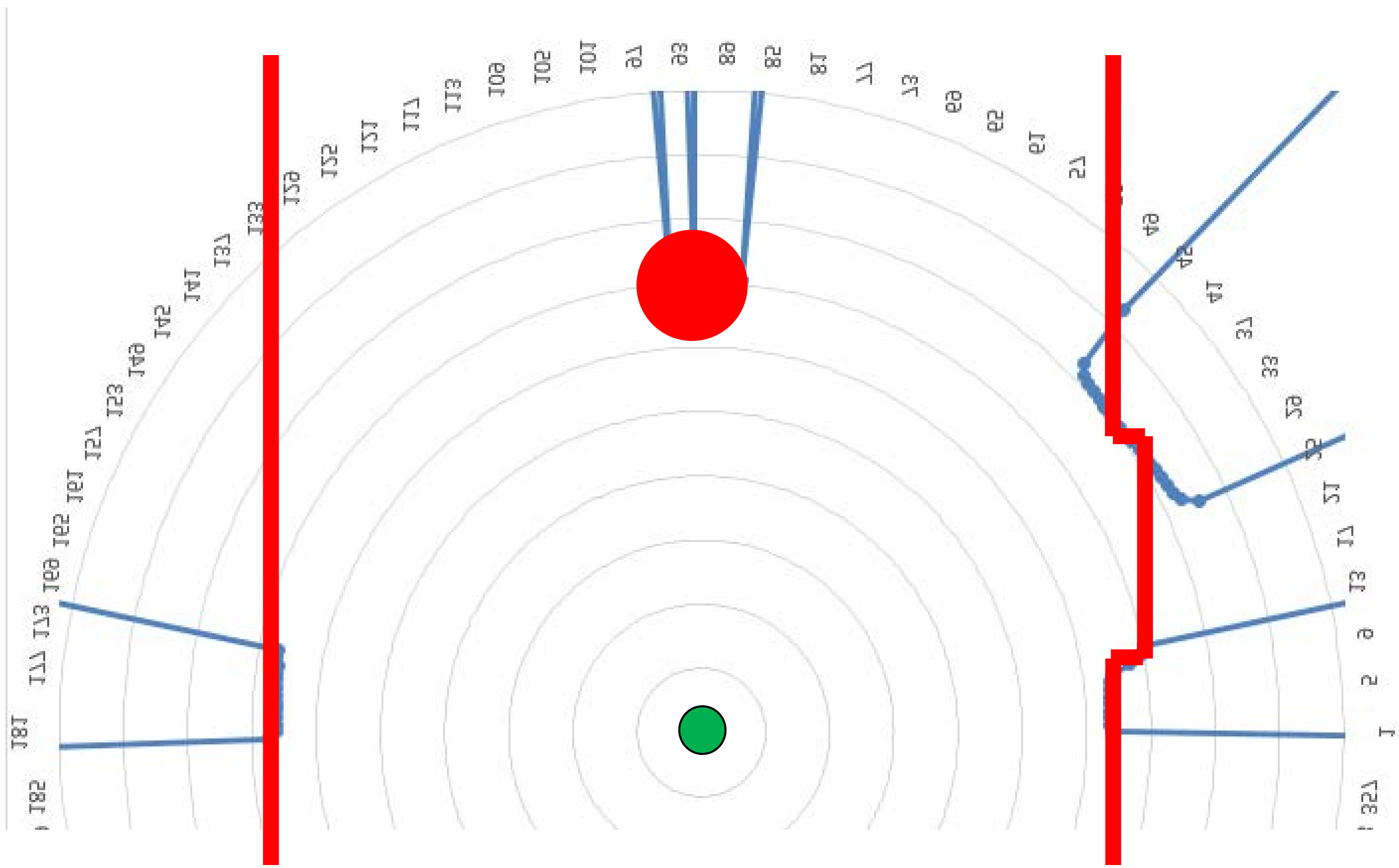
Sonar Test Results: Box

- Example: flat side of box is easy to see.



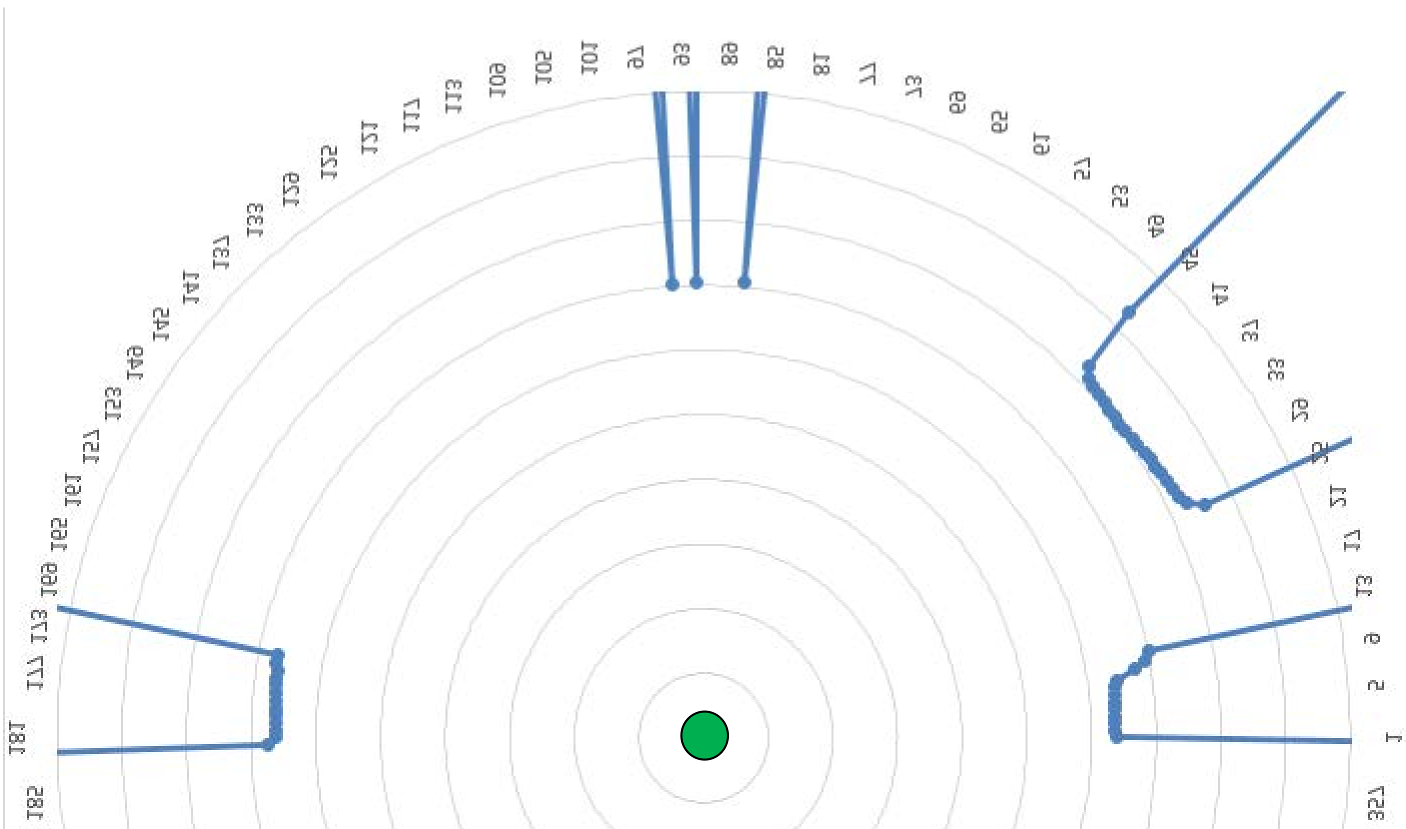
Sonar Test Results: Cylinder

- Different objects have different responses.



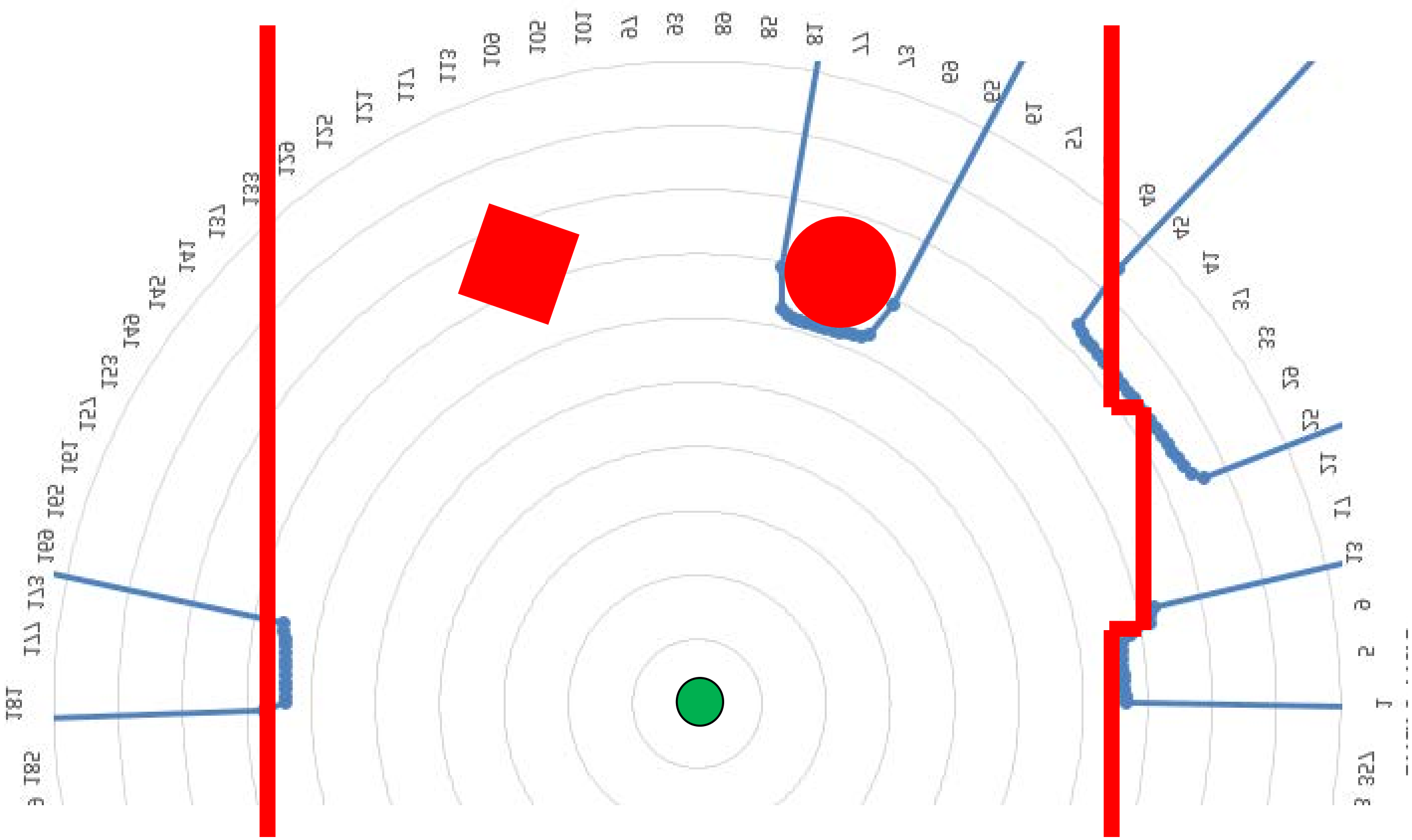
Sonar Test Results: Cylinder Data

- Cylinder is more difficult to see.



Sonar Test Results: Filtered Data for Cylinder and Angled Box

- Filtering can help in some cases, but some objects will just be invisible at some angles.



Project Starting Point

- You will have a complete SCOMP
  - Implements all instructions in Table 7.1 of lab manual
  - Implements additional instructions detailed in robot manual
  - Implements a 10-level subroutine call stack
  - Has twice as much program memory (2048 words)
  - Supports hardware interrupts from four sources
- You will have a complete DE2Bot Quartus project
  - Has working interfaces with all DE2 I/O (switches, LEDs, etc.)
  - Has the full complement of robot I/O devices
- You will have example ASM programs
  - An introduction to the robot, for your exercises next week
  - Basic project starting point, including relevant sample code

Project Phases and Key Dates

- Introductory exercises (next week in lab)
  - Focus on understanding DE2Bot and its capabilities
- Turn in proposal on Nov 18th by 3pm
  - That's three weeks from now, and you should be nearly <u>finished</u> by then
- Complete your design by Nov 28th
  - You will not be able to work in the lab after this day
- Final demonstrations Nov 29th to Dec 1st
  - Demonstrate your solution in your section
  - Make a PowerPoint presentation explaining your design
- Turn in final report by the following Tuesday, Dec 6th
  - The course calendar incorrectly says Monday; the design report assignment sheet will show the correct date

Project Schedule

	30-Oct	31-Oct	1-Nov	2-Nov	3-Nov	4-Nov	5-Nov
Assignments Due			Lab 8 Results, Brainstorming sheets				
Lab Activity	CLOSED	OPEN HRS	Project Initial Exercises*			OPEN HRS	CLOSED
Lecture Topic						Proposals	
	6-Nov	**7-Nov**	**8-Nov**	**9-Nov**	**10-Nov**	**11-Nov**	**12-Nov**
Assignments Due			Logbook Pages*				
Lab Activity	CLOSED	OPEN HRS	Practical Exercise 2 & Project Work*			OPEN HRS	CLOSED
Lecture Topic						Project Questions	
	13-Nov	**14-Nov**	**15-Nov**	**16-Nov**	**17-Nov**	**18-Nov**	**19-Nov**
Assignments Due			Logbook Pages*			Proposal Due**	
Lab Activity	CLOSED	OPEN HRS	Project Work*			OPEN HRS	CLOSED
Lecture Topic						Oral Presentations	
	20-Nov	**21-Nov**	**22-Nov**	**23-Nov**	**24-Nov**	**25-Nov**	**26-Nov**
Assignments Due				NONE			
Lab Activity	CLOSED	OPEN HRS		Fall Break			CLOSED
Lecture Topic						No Lecture	
	27-Nov	**28-Nov**	**29-Nov**	**30-Nov**	**1-Dec**	**2-Dec**	**3-Dec**
Assignments Due			Logbook Pages*				
Lab Activity	CLOSED	OPEN HRS	Project Demos & Presentations			CLOSED	CLOSED
Lecture Topic						Report Tips	
	4-Dec	**5-Dec**	**6-Dec**	**7-Dec**	**8-Dec**	**9-Dec**	**10-Dec**
Assignments Due			Final Reports (ALL DUE MONDAY)				
Lab Activity	CLOSED	CLOSED				CLOSED	CLOSED
Lecture Topic						No Lecture	

Next Week in Lab

- You will have some guided exercises to perform
  - Robot self-test
  - Basic robot movement
  - Associated check-offs count towards your Logbook grade
- If you complete the exercises before your lab period is over, **don't waste that extra time**
  - You have <u>three</u> lab sessions (<u>including</u> next week) and a handful of open hours to complete this project
  - The lab will be **busy** in those last few days
  - Robots may have to be rationed
  - Don't count on completing significant work during that time

Clarifications

- Additional announcements and clarifications will be posted **on Piazza**
  - You are responsible for information posted there
  - Could include changes to rules or assignments
  - Make sure you are monitoring it!
- Use Piazza to ask questions
  - If a general question is asked, everyone can benefit from the answer
  - If your question contains details specific to your design, you can limit the visibility to only instructors.