

BPL001 Introduction to Robotics

Instructor: Carlos Ambrozak

Spring 2016

General Information

Class Time

Monday 4-530PM

Classroom Location

Main Library Tech Lab, 480 Primrose Rd. Burlingame, CA 94010

Contact Info

Kim Day day@plsinfo.org, Carlos Ambrozak carlos.ambrozak@gmail.com

Website

<https://github.com/cambrozak/bp001> (<https://github.com/cambrozak/bp001>)

Course Description

The field of Robotics is at a fascinating and exciting point in its evolution. Robots are no longer confined to the realms of industry and research, they are rapidly becoming a part of daily life.

This course provides Middle School students with an introduction to fundamental concepts and practical application of tools and techniques for building robotic systems in order to inspire further study and participation in High School and College programs.

Hands-on laboratory exercises provide experience with embedded systems, programming, circuits, motors, sensors and servos.

Students collaborate on their own autonomous robot platform to implement dynamic physical behaviors such as: edge detection, line following, collision detection, orientation, proximity sensing, audio & lcd text output, image sensing and servo controlled arm manipulation.

Short lectures connect the laboratory exercises to current state of the art happenings across the field of Robotics and beyond.

Prerequisites

None! This class is for motivated beginners.

Objectives and Outcomes

Discuss experience with robots in the real world in order to understand what a robot is and how it relates to general purpose computers and embedded systems.

Know the characteristics of four main classes of robots: Pre-Programmed, Autonomous, Remote and Augmenting.

Employ simple tactics to identify and overcome two major obstacles to hi-tech learning: FUD & Buzzwords.

Write and upload source code programs for execution on a physical robot.

Utilize source code program libraries to understand how libraries and APIs work and why they are important.

Maintain source code in a versioned repository to understand revision history.

Understand some general fundamentals of programming languages: expressions, syntax, high-level vs low-level.

Write programs with typical programming forms to control hardware: variables, if/then, loops, functions.

Write programs that use variables in different scopes to understand the general concept of scope in programming languages and how it is implemented in the C language for Arduino.

Use a Simulator and Step-Thru Debugger to troubleshoot programs with the 'EIPOC' meta-tactic.

Observe how electrons move through a conductor using a physical marble-madness model to understand how current flow affects circuit components such as lights LEDs and Motors.

Build a simple battery from scratch, power an LED and probe it with a multimeter to observe and understand Electric Current in Circuits and how it relates to Voltage and Resistance.

Add cells to the homeade battery in Parallel and Series configurations, probe with a multimeter to observe and understand how these arrangements affect Voltage and Current.

Construct a Reference Circuit on a breadboard composed of a battery, fuse, control switch, an LED a Resistor and Common Ground to explore and understand how to troubleshoot the three types of circuit problems: Short Circuits, Grounded Circuits and Open Circuits

Provide the Robot with an identity by writing a program to store and display its name and unique ID on a LCD display and play a distinguishing tune with a Piezo Buzzer.

Provide the Robot with edge detection awareness by writing programs to process input from IR LED/Phototransistors and control motor outputs. This keeps the robot from roaming

beyond the test area.

Provide the Robot with path finding abilities by writing programs to process input from IR LED/Phototransistors and control motor outputs.

Provide the robot with autonomous roaming capabilities and collision detection by writing programs that process proximity sensors, accelerometer, gyroscope and compass inputs and control motor outputs.

Provide the robot with an articulated arm by connecting a pan-tilt mechanism to the chassis and writing programs to control servo movements and point the arm at objects detected by proximity sensors.

Provide the robot with the sense of sight by attaching an image sensor to the pan-tilt mechanism and writing programs to react to different image patterns.

Teaching and Learning Style

The primary objective of this course is to inspire middle school students to take an interest in Robotics regardless of prior skills and experience.

A broad top-down approach to the overall subject combined with a hands-on laboratory setting with bottom-up emphasis on select problems enables students to learn by discovery and be effective with the technology immediately. This teaching style provides just enough detail to learn foundational concepts and hold the interest of budding beginners.

Each classroom session follows a *Lab-Lecture-Lab* format:

A simple lab exercise is completed as a group to start the session. Students follow along with the instructor to load prewritten software and wire up components onto the Robot and observe resulting physical behavior. The session does not proceed until every Robot is successfully performing as expected.

A short lecture explains how and why the lab exercise worked and answers open questions. The lecture also connects the lab exercise to examples in the real world.

A final lab exercise challenges the students to implement their own original ideas by combining the first lab with previous learning.

Class Rules

1. Respect each other.
2. Respect the lab.
3. Respect the equipment.

Schedule

Session One: Introductions, Orientation, First Program

Lecture: What's a Robot? Hi-Tech Learning

Exercise: Give the Robot a name. Edit, compile and load a program, LCD and Audio output. Save code in GitHub.

Session Two: Programming, Orientation, Motors

Exercise: Move the Robot, Edge Detection, Program motor control using libraries & APIs.

Lecture: Expressions, Syntax, Variables, if/then, loops

Exercise: Turn the Robot, Speed control, orientation with the gyro and compass, move in a pattern.

Session Three: Programming, Debugging, Line Following

Exercise: Line following. Program processing of IR LED/Phototransistor inputs.

Lecture: Scope, Functions, High Level vs. Low Level Languages, The Art of Debugging

Exercise: Step-Thru Debugging with Simulator for Arduino

Session Four: Electricity, Circuits

Exercise: Build a Battery from scratch, light an LED, measure Current, Voltage and Resistance with a Multimeter

Lecture: Electricity in 30min: Electrons, Current, Voltage, Resistance, Series, Parallel, Ground

Exercise: Build a Reference Circuit using a battery, LED, Switch and Ground. Debug the three types of circuit failures: Short Circuits, Grounded Circuits, Open Circuits

Session Five: Roaming

Exercise: Detect collisions with accelerometer, Detect objects in space with proximity sensors.

Lecture: Why Accelerometers and proximity sensors work.

Exercise: Advanced Roaming, navigating terrain, avoiding moving obstacles.

Session Six: Servo Mechanical Control

Exercise: Attach Pan-Tilt servo arm to the Robot

Lecture: Servo safety, how servos work

Exercise: Point the servo arm at detected objects

Session Seven: Pattern Recognition

Exercise: Attach Pixy CMUcam5 image sensor to servo arm

Lecture: Image Sensors and Pattern Recognition

Exercise: Program robot for follow/flight when it recognizes certain image patterns.

Session Eight: Open Lab

Exercise: TBD

Lecture: TBD

Exercise: TBD

Equipment

All equipment is kept in the lab at all times.

Laptop

- Dell, Windows 10

Software

- Arduino IDE (<https://www.arduino.cc/en/Main/Software>)
- Simulator for Arduino (<http://www.virtronics.com.au/Simulator-for-Arduino.html>)

Hardware

- Zumo 32U4 (<https://www.pololu.com/product/3126/resources>)
- Mini Pan-Tilt Arm and Micro Servos (<https://www.adafruit.com/products/1967>)
- Pixy CMUcam5 Sensor (<https://www.adafruit.com/products/1906>)

Text and Reference Material

Purdum, Jack J. *Beginning C for Arduino* New York: Apress, 2012. Print.

Atmel ATmega32U4 (<http://www.atmel.com/devices/atmega32u4.aspx>)

Arduino (<https://www.arduino.cc/>)

License & Copyright

BPL001 Introduction to Robotics by Carlos Ambrozak is licensed under a Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>) .