Engineering Research Course, Boston Leadership Institute, 2017

Mrs. Diane Brancazio

The website I created for the course is http://designbuildcode.weebly.com/. There you will find Goals and Syllabus, resources for all projects including links to lesson plans, and some photos.

In summary -

Students started by learning electronics principles and skills. This part of the course includes several investigations, and 2 projects that students could then take home: **LightMeUp Tiles** (personalized LED displays), and **BumpBots** (simple robots that change direction when they bump into something).

Next is the **Programming and Robotics** part of the course. For this they need to build robot bodies, then learn how to use Arduino microcomputers. They program Arduinos to control the robots in a simple demonstration pattern, then added a variety of sensors and actuators so the robots can do some interesting things, such as follow a line, interact with each other, play a tune, navigate a maze, pick up and deliver an object, etc. Students also had the opportunity to use the Arduino microcomputers and variety of electronics materials to create a different project other than a robot

Students have fun while learning, share discoveries, and come up with ideas and creations that inspired the course leaders as well as the rest of the class.

Daily Agendas

Day 1 morning

Objective: Know each other and the plan for the course. Learn to solder. Design and begin building a Light-Me-Up Tile.

Classroom materials:

Project Materials: Corrugated plastic squares, copper tape, LEDs, magnetic strips, craft materials, markers, colored pencils, scissors, hot glue

Soldering equipment, stranded wire

Day 1 morning

1. Welcome (15 min)
    a. Students check in – clarify names, and take a seat at any table
    b. Go to website designbuildcode.weebly.com and do "Intro exercise" on Google Form
2. Introductions and housekeeping (30 min)
    a. Teacher and mentor introduction
        i. Training, Engineering Work, Teaching work, Sailing exp, Family
    b. Ground rules
        i. Honor yourself, your classmates, and the learning process:
        ii. Be safe, Stay on task, look out for others, ask questions, bring problems/concerns to someone who can do something about them, Have Fun!
    c. Course structure:
        i. All have projects can be done individually or in groups. Groups can make several projects together, 1 for each group member. All projects can go home with you
    d. Materials
        i. Laptop and charger
        ii. Pencils
        iii. Water bottle
        iv. Cell phone
        v. Good attitude
    e. Student Intros
        i. Your Name
        ii. Where you are from
        iii. What you want to get out of the course
    f. Review Overall Goals and Outline of course.
        i. Show/demonstrate example of each project
3. Soldering activity (60 min)
    a. Show video on soldering http://www.youtube.com/watch?v=BLfXXRfRIzY
    b. Make soldered rings of wire
4. Light-Me-Up tiles (75 - 90 min)
    a. Take a look at tile board
    b. Design tile with up to 4 LEDs
    c. Layout design
    d. Build tile with craft materials,
    e. Add magnetic tape – see examples for correct locations
    f. Add copper tape. See diagram for techniques
    g. Fasten LEDs with hot glue (not where they will be soldered)
    h. Save for soldering

Day 1 afternoon (2 hrs)

Objective: Learn electronic components, basic circuit principles, symbols and schematics.

Materials: Circuit investigations – 1

1. investigate components with set of simple challenges – see lesson plan Electronics Investigation1.doc for all activities

Day 2 morning 2 hr 45 min

Objective: Finish and debug and share Light-Me-Up Tiles (90 min) Build BumpBot Bodies (75 min) could use 90 min

Classroom materials:

Project Materials: Corrugated plastic squares , copper tape, LEDs, magnetic strips, craft materials, markers, colored pencils, scissors, hot glue

Day 2 afternoon 2 hr 15 min

Objective: Wire prep for BumpBots (75 min – some needed 30 min more time) and circuit debug

Followed drawings on whiteboard (need better drawings – paper handout was wrong) and made wires for the BumpBot, installed on BumpBots such that they were ready for soldering. Teacher checked wires and tested functionality. Biggest issue is not enough insulation on wiring.

How to use Multimeters – did for some – did not work out well. Need more structure and steps and activities here. Define terms and concepts before moving into meters: voltage, Current, Power and Resistance

Classroom materials:

Project Materials: Corrugated plastic squares , copper tape, LEDs, magnetic strips, craft materials, markers, colored pencils, scissors, hot glue

Day 3 morning 3 hours

Explain task – solder and strain relief. Take batteries out when soldering

Fix bodies as needed and solder BumpBots – debug and get them working. Most working by 10:30, 90 min. Decorate test, refine, play and clean up until 12:00

Day 3 afternoon

Discussion what we have learned and what we are doing (15 min) to 1:30

Technology/Engineering – we go back and forth between them

Projects up to now have included both technology and Engineering. Technology is using the tools and materials appropriately like wiring, soldering, building, learning electronic components and how to connect them. Engineering is using technology to solve a problem or accomplish a task such as making circuits with components when given a goal task, debugging BumpBots after building, refining BumpBots (antennas, body, performance)

Students now have new skills in technology and engineering: stripping wires, soldering, identifying components, making circuits from a description, using components in circuits (switches, motors, potentiometers, lamps, LEDs, buzzers), using LEDs in circuits, using copper tape as a conductor. Problem solving and debugging.

The scientific principles behind the technology should now make more sense – spend some time studying it at a scientific level

Electronic Principles (60 min)

See Electronics Principles 1 Lesson/activity will take 2 hrs with activities

Videos from http://afrotechmods.com/tutorials/category/tutorials/beginner-tutorials/

Do BumpBot circuit – with limit switch and motor and battery (15 minutes) better to do this right after they prepare wiring on BumpBot and before soldering

Breadboard lessons Intro to breadboard – 40 minutes

Show layout of breadboard and how connections work

Materials for each group of 2: Battery pack, breadboard, solid wires and wire stripper, 2 LEDs, 100 ohm resistor

Students make a circuit on breadboard with 2 LEDs and resistor. They figure out whether series or parallel will work. Use Power buss on breadboard for 3V power.

Add potentiometers, motors. Make some circuits, putting components in series or parallel to see if they work. Use potentiometer to change speed and brightness

Day 4 morning 3 hours

BumpBot 3 (~2 hours)

Check basic functioning and work on BumpBots as needed/desired 45 min

- some need soldering or strain relief
- some need new components.
- Some get decorated
- Some tinker with circuit – add LEDs and/or buzzers and/or other
- Some create course for demonstration and play

LED tile finish (30 min)

- Review LED tiles and share with group
- Students checked them in advance and debugged/fixed as needed

Day 4 afternoon 2 1/2 hours

Fill in on Electronics Lessons using Electronics Principles 1 (1+ hour)

- Multimeter video
- Voltage measurement Practice
- Current measurement practice
- Resistor measurement practice
- Voltage divider video and practice

Protoboard circuits – Light Sensitive circuits (1hour)

- Review Light sensitive circuits to show breadboards and voltage divider
- Groups of 2
- Handed out all materials – breadboards, resistors, transistors, jumper wires (pre-made), battery packs, LEDs (students helped distribute materials)
- Built night-light type circuit

Day 5 Field Trip – all day

Day 6 morning 2 1/2 hours

Electronics Lessons breadboards

- Do Toy Organ circuit (optional)
- Do Audible Light Probe circuit (optional)

Prep materials for Robotics –

- Neopixels soldering
- Reflectance sensor soldering
- Speakers – add wires
- Battery pack tinning
- Download Arduino IDE – check for latest version

Finished most work and cleaned up by 11:30

Last 30 min – Arduino intro –

- Watched inspirational videos – top 10, top 5,
- Watched Intro videos from https://www.youtube.com/playlist?list=PLYutciIGBqC34bfijBdYch49oyU-B_ttH
- Watched prelude and Hardware overview (10 min total)

All Arduino work for this day (morning and afternoon) is compiled in Arduino Lesson 1)

Day 6 afternoon 2 1/4 hours (started 1:15)

Arduino Introduction with videos and exercises

- Skipped download and install because it was already done at home or in morning
- Watched Tutorial #3 and made my own comments (15 min)
- Handed out Arduino Unos, Cables, and Servo Motors. Did Arduino Tutorial1 with Servos. Students copy and get page 1 to work. Comment on it and move on to page 2

2:15 – Started Arduino Tutorial 2 on Serial I/O

- Reviewed purpose of tutorial – using Serial Monitor to get info and display to user
- Mentioned data types: int and float
- Mentioned camelCase convention for variables
- Did page 1 of tutorial 2 by 2:45
- BY 3:00 most did page 2 and some combined it with servo code
- Worked till 3:15

Reviewed Arduino references – guides like programming notebook, and Arduino.cc

Day 7 morning

- Robot chassis construction, motor wiring, soldering (Instructions on website)
    a. Build robots on stock bases
    b. Motor drivers/H-Bridge tutorial
    c. Wire to L298N board Motor controller, and battery pack
    d. Power in, Motors out. Check direction and fix wiring if needed
- Wire to Arduino to check power supply from Arduino

Day 7 afternoon

- Arduino programming lesson - compiled in Arduino Lesson 2
    - Practice with syntax, IDE, variables, etc.
    - Controlling LEDs
    - Logic with For Loops

Day 8 morning (2 ½ hours)

Materials: Wiring chart, starter code ready to download for motor controllers – robot goes through all speeds moving forward, then through all speeds in reverse. Robots built and wired. For each robot, a set of 6 wires with M-F ends, (best in a ribbon cable format), not including Red and Black

- Overview of task – get starter code and program your robot to do a demonstration pattern (such as an infinity sign, a box, out and back, or spinning). Learn about the robot, how the wheels and motors behave, how it balances. There will be an opportunity to make a new chassis and mount components as desired. This should be done after there is an understanding of how the base system works.
- Review starter code (make sure all are close and can see the screen, not in front of computers) for content and for programming conventions
    a. Description and wiring notes in header
    b. #define for pin definitions
    c. digitalWrite() vs analogWrite()
    d. Use of functions to modularize code
    e. Use of tabs in IDE
    f. Using “hit any key to continue” to pause code
- Demonstrate code on teacher robot
    a. Watch values on Serial Monitor
    b. Unplug Arduino from computer and show it moving on ground
    c. Discuss why it may not be going straight
    d. Discuss how it might turn
- Brief intro to functions
    a. Show similarities/differences between code with functions and without

    b. Demonstrate how to create a function with existing section of code (e.g. create a function called forwardTest or motorsOff).
- Brief intro to tabs in Arduino
    a. Create a new tab, call it “functions”, and move functions there, then save program
    b. Show how the sketch folder now has another .ino file in it for the code in the tab
- Download starter code from website
    a. Demonstrate how to download, unzip, and put folders into Arduino folder
    b. Check that all students can access and store code
    c. Open the code and do Save As to get a new copy
- Connect motor controller to Arduino and use starter code to confirm hardware
    a. Wire up motor controller with 6 new wires as specified in header and on wiring sheet. Do not break wires apart.
    b. Run demonstration code to confirm that robot wiring is correct
- Work on demonstration pattern and robot setup
    a. Create original code for demonstration pattern of choice
    b. Mount components to chassis as desired
    c. Debug wiring and software as necessary
    d.
- Whole class demonstration of all robots (15 – 20 min)
    a. Software review – what did the robots do?
    b. Hardware review – how were the components mounted?

Day 8 afternoon (2 ½ hours)

- Materials: Wiring chart, starter code ready to download for reflectance sensors, each with a set of 5 wires with M-F ends, including Red on Vcc and Black on Gnd. Speaker with solid wires.
- Overview of task – get starter code for reflectance sensors (line-following sensors). Test thqt the sensor works. Mount the sensor on the robot such that it reliably reads data sensors. Lots of Testing and chassis construction

Not much coding, except for testing. Several refl sensors had bad soldering and needed to be fixed

Several chose to build new bases

Needed to modify the motor code to fix pins. Have not yet combined motor and sensor code. – will need to provide this!

Goal today was to get the line-following sensors mounted and all other components mounted on the chassis. Some built new chassis.

Day 9 Morning (Friday)

1. Check in on all being able to move and read sensors (separately). Refer to list of things to do and code/experiment for 90 min until frustration builds up. Do lesson with demonstration of robot and writing code
2. List of things to have done:
    - Robot built and wired
    - Robot can do demonstration pattern with MotorTest code
    - Reflectance sensors are mounted on robot, with correct wiring for Right A0, Middle A1, Left A2
    - Mount Arduino, battery pack and motor controller fairly securely (can change later)
    - Reflectance sensor work with ReflTest code
3. Create/obtain combined code LineFollowReady code which sets up combined code
    - Header- comments with brief description of code and wiring notes
    - Declarations – constants and variables
    - Setup – initialize Serial monitor, set motor controller pins to output
    - Loop
        - Read in Data from refl sensors
        - Print out to monitor for verification of data
        - Logic to determine speed of each motor
        - Drive motors (send speed value to enable pins), and set direction to forward

Students work on code on their own – much frustration – hardware and software problems

Call a stop and do a lesson on the board with all students close up. Write the logic that will make a robot work (come up with this as a demo)

(Add a lesson on functions earlier so the concept can be used here in creating function code)

Combining code

- Do demo on projector
- Start with motor code and "Save as" Linefollow1
- Add in reflectance sensor code from other starter code, as needed
-

Introduce challenge – follow a line with just 1 sensor (any of the 3)

Introduce Algorithms and flow charts – brief discussion of their importance – don't start coding without figuring out a line following algorithm

Day 9 Afternoon (end of 2nd week)

spent working on code

A couple of successes – shared code in a folder.

Lots of frustration

Pointed out how much they now can do in Arduino coding, robot construction, and working with electronics. Happy campers.

Day 10 Monday

Morning – get more hardware and starter code. Pay with it during the day, start thinking of a project that uses a variety of hardware. Sample flowcharts to help with algorithms

1. Review the new starter code in google folder and demonstrate as working through it (took a while – should be broken up)
    a. motor balance,
    b. line follow template (combined code for motor controller and refl sensor.
    c. working line following code (from me),
    d. better tests for reflectance,
    e. new code for neopixel, servomotor, photoresistor (never worked well)

    Should have had a uniform wiring systems by now – did not – check that all components can be used together, and if not, which cannot be used together. Spent time working that out with class
2. Review algorithms and flow charts – show sample flowcharts and graphic method for writing statements in flowchart boxes. Discuss different algorithms that can be used for line-following – edges, etc.
    Should also be highlighting programming techniques and styles while showing sample code
    Consider having students fix a piece of code and add comments to check style

Afternoon –

start thinking of a project that uses a variety of hardware. Program, play around with components.

Day 11 Tuesday

Morning – Get a project, work on code, test it out

Evaluate – its okay to change the project

Afternoon - everyone working

Day 12 Wednesday

Morning – Get a project, work on code, test it out

Evaluate – its okay to change the project

Start the maze – cut foamboard and show students how to build it

Afternoon –

Lesson on

- logic and programming – while loops vs if statements vs for loops
- Functions
- Commenting
- How to find things out
-

Individual work with students to fix/debug code and hardware. Troubleshooting/debugging – not done with the group - need to make up a list with a troubleshooting procedure

More of everyone working

Day 13 Thursday

Morning – Get project working today – can start on presentation in afternoon

Trim down your project to something reliable. It is better to have it do the thing you say it will do, and and do it reliably, than to do a little of everything badly. Students will create a presentation that they could show to a technologically literate person. Should show what they want to show about their learning and what they created. Think of it like a portfolio showing off who you are as a designer, engineer, problem solver, programmer,

Build maze so robots can be tested (in the future do this in the very beginning)

Afternoon –

Refine projects. Test/evaluate, document

Do end-of-course survey – good preparation for creating presentations

Worked with students on everything, helping them get something to work reliably.

When possible – have students hep each other

Friday Day 14

Morning

- do survey
- share results of survey
- make huge maze

- troubleshoot/debug guide
- student advice to others just starting to code and “I wish I knew this”/”I wish I listened to this”
- work on presentations - shared folder
- clean up (tape, scraps) and sort materials
- greetings to Ryan
- review resources on webpage

Afternoon

Presentations