Intro to Physical Computing Syllabus

ITPG-GT.2301.001	Tom Igoe	Tuesday, 9:05 AM - 12 PM
ITPG-GT.2301.002	Tom Igoe	Wednesday, 9:05 AM - 12 PM
ITPG-GT.2301.003	Jeff Feddersen	Wednesday, 9:05 AM - 12 PM
ITPG-GT.2301.004	Jeff Feddersen	Wednesday, 3:05 PM - 6 PM
ITPG-GT.2301.005	Benedetta Piantella	Monday, 11 AM - 2:55 PM
ITPG-GT.2301.006	Benedetta Piantella	Thursday, 3:05 PM - 6 PM
ITPG-GT.2301.007	Tom Igoe	Monday, 11 AM - 2:55 PM

Physical Computing is an approach to learning how humans communicate through computers that starts by considering how humans express themselves physically. In this course, we take the human body as a given, and attempt to design computing applications within the limits of its expression.

To realize this goal, you'll learn how a computer converts the changes in energy given off by our bodies (in the form of sound, light, motion, and other forms) into changing electronic signals that it can read interpret. You'll learn about the sensors that do this, and about very simple computers called microcontrollers that read sensors and convert their output into data. Finally, you'll learn how microcontrollers communicate with other computers.
Physical computing takes a hands-on approach, which means that you spend a lot of time building circuits, soldering, writing programs, building structures to hold sensors and controls, and figuring out how best to make all of these things relate to a person's physical expression.

Some time in weeks 1 - 3: Attend a tool safety session in the shop

General class structure:

- 45 minutes discussion, Q&A from the homework, discussion of reading material, current events, observations from the world.
- 5-10 minutes intro the exercise, any questions about it
- 90 minutes build time. Feel free to take a break as needed during this time. Please stay in class to work the majority of the time, however, as you'll benefit from questions that come up during yours and your classmates' progress.
- 30 minutes review and discussion. During this time, we'll highlight 3-4 projects as time permits. During the review time, please take written notes on your classmates' projects when they're being reviewed, and give them to each other afterward.

Readings and Labs:

Course notes for each week are linked so you can (and should) read them *before* class, to know what we're talking about. There are also lab assignments that illustrate the topics. You should read through the notes and labs and try as much as you can the week before the lab is discussed in class. There won't be formal in-class lectures on this material. Instead, we'll start each week with questions from the notes and labs.

In-class exercises: These are short (90-minute) projects for you to get hands-on experience building projects using the principles you're learning in the class. If you've done the readings and labs before each class, you'll be ready for them. If you haven't, you'll likely have a hard time.

For each exercise, you'll be paired up with a classmate or two the week before. You'll get some basic details the week before, but the final details of each exercise will be announced in class. Each in-class exercise will build on the material and lab exercises you will have read and tried prior to the class. Feel free to plan in advance. Bring in your own construction materials, extra sensors, and whatever other elements you might want to incorporate that will make your in-class project the best that it can be.

If there are any materials that you need that aren't available in a starter kit, you'll be given access to them for the class period.

Come to class with questions prepared about that Class's assignments. If you have no questions, be prepared to build a working project in class.

Commenting on each others' work: When watching your classmates present their work in class, we'll make some time for verbal comments, but you should take written notes on their presentations and on their interaction with their sample user. Offer suggestions on what they do well and should continue doing, what they should stop doing, and what they could add to their work and/or their presentation to make it better. Give them these notes at the end of class. You'll be getting the same notes from your classmates, so write in the same voice in which you'd like to hear feedback on your own work.

Blog assignments are assigned in response to some of the higher-level readings. Do them the week when they're assigned. These readings and your responses will come up in class from time to time, Read and comment on each other's writings as well. In addition to the assigned blog posts, use your blog for notes on your progress and documentation of your projects in progress. **Please make sure your blog assignments are online the night before class (by 8 PM) so that your instructors and classmates can read them before class.**

Class 1

What is physical interaction?

In-class: As a class, we'll make a list of fantasy devices; things that don't exist, but that we wish did. For the moment, we'll ignore technical feasibility, and concentrate on how you'd operate these devices if they did exist. You'll be divided into groups of 3. Pick a fantasy device from the list. Design a control interface for this device and demonstrate it in-class. Think about what a person has to see, hear, touch, etc. while operating the device. Think about what cues they need to understand the interface. Think about what parts of their body they need to have free, what gestures are best for triggering the device's behaviors, and so forth. Build a mock-up of the controls from anything you can find on the floor or nearby. At the end of class, one or two of you will act out the use of the device using your mock-up while the third person narrates.

Reading for next class:

- Crawford, The Art of Interactive Design, chapters 1 and 2 (note: you will need to sign into NYUHome to view this. From your NYUHome home page, click "Research" then "books24x7.com" then search for "The Art of Interactive Design" by Chris Crawford. Alternately, try this link.)
- Bret Victor, "A Brief Rant on the Future of Interaction Design"
- Understanding Electricity
- Understanding the breadboard
- A short video in a light-hearted vein on some electrical characteristics.

Labs for next class:

- Lab: Components
- Lab: Setting up a breadboard
- Lab: Electronics and using a Multimeter
- Lab: Switches
- Lab: Transistors

Blog: After this class' discussion and exercise, and reading Chris Crawford's definition and Bret Victor's rant, how would you define physical interaction? What makes for good physical interaction? Are there works from others that you would say are good examples of digital technology that are not interactive?

Class 2

- Understanding Electricity:
  - Ohm's Law
  - Parallel vs. serial circuits
  - Identifying electrical components
  - Making a circuit

In-class: Work in pairs. Using the electronics you learned about in the readings and labs for this Class, make a zoetrope or a magic lantern. Bonus points for giving the user the ability to control the speed of the lantern, and/or controlling the timing of the light turning on and off to create a persistent moving image.

Reading for next class:

- Norman, Design of Everyday Things, ch. 1
- Norman, Emotional Design, Chapter 1, "Attractive Things Work Better".
- Igoe, Physical Computing's Greatest Hits (and misses)
- What is a Microcontroller?
    - Microcontrollers and sensors in the everyday environment
- Analog vs. Digital
- Digital Input and Output
- Intro to Arduino and first program.
- analog input

Labs for next class:

- Lab: first Arduino program. Download the latest version of the Arduino software for this lab.
- Lab: Analog in
- Lab: Mouse Control
- Lab: Mouse Control with Push Buttons
- Lab: Mouse Control with Joystick
- Lab: For more info on the Mouse and Keyboard control with Leonardo

Class 3

Introduction to microcontrollers

- Digital, analog in
- digital out
- serial.print, println

In-class: work in pairs. Make a game controller with mouse and/or keyboard libraries. You'll be given a keyboard/mouse controlled browser-based game (chosen at random on class day from one of the games below), and you'll create a game controller for it, using whatever sensors you have at your disposal. Bonus points for being able to turn on and off control of the mouse, for increased functionality and ability for the user to win the game fairly. Make sure your controller gives local feedback as well as sending data to the computer, so the user knows that it's working properly.

Possible games:
http://www.ponggame.org/
http://chrome.angrybirds.com/
https://chrome.google.com/webstore/detail/90%60s-games/illbbfoihflomkbpcaaakhijinbnejom?hl=en-US
https://chrome.google.com/webstore/detail/build-with-chrome/lbbbhbjeecagnlfgggogfclkdjamoapf?hl=en-US

Reading for next class:

- Graham Pullin, *Design Meets Disability*
- Example: Ramps (see also this video) Digital Wheel Art, Modal Kombat, Red Urchin, Bricolo
- Analog Out
- High Current Load Control - Transistors and Relays

Labs for next class:

- Lab: Tone output
- Lab: servo/analog out
- Lab: Controlling High current loads with transistors

Blog: Observation. Pick a piece of interactive technology in public, used by multiple people. Write down your assumptions as to how it's used, and describe the context in which it's being used. Watch people use it, preferably without them knowing they're being observed. Take notes on how they use it, what they do differently, what appear to be the difficulties, what appear to be the easiest parts. Record what takes the longest, what takes the least amount of time, and how long the whole transaction takes. Consider how the readings from Norman and Crawford reflect on what you see.

Class 4

Analog Out: Pulse width modulation vs frequency modulation

In-class: work in pairs. Build a musical instrument. Use what you've learned about physical input and combine it with what you learn this Class about output in order to make a musical instrument. Since you haven't learned about synthesizers and data protocols yet, your instrument will most likely be an acoustic instrument, but it doesn't have to be. At minimum, you should be able to play different pitches. Bonus points for being able to vary not only pitch, but also volume, and play multiple notes simultaneously. Make sure your instrument gives local feedback as well as sending data to the computer, so the user knows that it's working properly.

Reading for next class:

- serial communication
- More serial communication
    - multiple sensors
    - Interpreting bytes: ASCII vs. binary
    - handshaking/call-and-response

Labs for next class:

- Lab: Serial Output
- Lab: Multiple Serial Output

Class 5

Serial

In-class: Take one of your animation projects from Introduction to Computational Media and give it a physical interface. Use readings from physical inputs on a microcontroller to control changes in your Processing sketch.

Consider the behavior of the thing you've animated, and choose sensors that let you create that kind of behavior interactively. For example, if your animation featured a bouncing ball that moved with a "squishy" feeling, make sure the physical control has a similar feel. You might get this by embedding force sensors in foam, or attaching pressure sensors to a balloon. If your animation has a sharp feeling to it, you might need sensors that have a precise, sharp feel, such as pushbuttons with a good solid "click".

If there is more than one parameter to be controlled, you will need to send data from multiple sensors to control each parameter. For example, if you're controlling an object that moves in three dimensions on the screen, you'll need sensors that give you control over all three dimensions. You might use a 3-axis accelerometer, or three photoresistors, or some other combination.

If you or your partner on this exercise are waiving Intro to Computational Media, that means you're good enough programmers to create this project in Processing on your own. Good for you! You'll be in good shape for this assignment.

Reading for next class:

- Buxton, *Sketching User Experiences: The Workbook* (note: you will need to sign into NYUHome to view this. From your NYUHome home page, click "Research" then "books24x7.com" then search for "Sketching User Experiences: The Workbook" by Saul Greenberg, Bill Buxton, et. al.. Alternately, try this link.) Read the summary of *Sketching User Experiences*, then browse the book for usefu prototyping and testing methods.
- Igoe*,* Making Interactive Art: Set the Stage, Then Shut Up and Listen
- Example: Happy Feedback Machine

Class 6

Tangible Controls and real-time control interfaces

One of the most interesting challenges for a physical interaction designer is to create a multi-modal control interface. These are used in situations which require a person to both see and hear what she is controlling and operate the controls at the same time. The person is using all of her senses to do her job: she's watching the action, sometimes listening as well, and feeling the state of her controls. Think about driving a car, or piloting a plane, or operating a crane. On the small scale, think about the controls to operate laproscopic surgical instruments during surgery. Controllers for live entertainment also fall into this category, as the operator needs to have her eyes on the stage, not the controls.

These kinds of situations require you as the designer to think carefully about all of the user's senses. You have to know where the user's eyes are focused, what she can hear, and what information she can gain about the state of the system by feeling tangible controls.

For the next two classes, you're going to make more fully-realized real-time control interface. You can take one of your previous projects and build a more finished version of it. You can also make a new project if you prefer, as long as it involves real-time control and feedback.

For inspiration, consider applications that require real-time control and tangible control elements so that the operator doesn't have to use her eyes to operate the controls. Vehicle dashboards, live media controllers, remote control interfaces, or something else you come up with in discussion with your instructor and classmates. You can design for a blind operator if you prefer as well. Make sure you know who would use such an interface, as you'll need to interview them about the controls.

This week, you'll come up with an initial sketch for the controls of your device in class, and test it on your classmates. For homework, you'll interview someone who's a likely target user using your sketch model, and you'll build an initial prototype based on what you learn from them. Next week, you'll fabricate a housing for the controls.

You won't be required to make a fully functional system, but you will be required to design and build the control interface for it. You need to consider the layout of the controls and displays as well as the feel of the tangible elements. That means (for example) if there are knobs, you need to design basic feedback when the user turns them. If there are switches, there should be feedback when they are pressed. Whatever actions your user performs should be rewarded with feedback in some physical form.

The feedback might consist of LEDs that indicate the state of the controls; speakers that make a sound to indicate when a control has changed; vibrating motors that give tactile feedback; and so forth. Your control interface for next week is likely to be rough around the edges. You could simply mount the controls on a piece of cardboard or foam-core. The point is to give your user something to see, hear and feel besides a breadboard, some sense of where the controls are located relative to each other, and some idea of what immediate feedback the device will give them to acknowledge their actions.

After next week's fabrication exercise, you'll also be expected to build a panel and housing for the tangible controls.

Playtesting

In-class: For the first half of class, you'll come up with an initial design for your user interface and develop a plan to user test your design with users with an interactive sketch. For the second half, you'll explain what it does to your classmates, and get their feedback. We'll do this in parallel, with half the groups as presenters and half as users, and then change sides so everyone gets feedback on their first interface sketches.

Prepare a list of questions you want to answer with this test: are the controls immediately understandable to someone who's never used the thing before? Can she successfully operate the interface? Can she learn the physical actions necessary and internalize them, so that she can concentrate on what the device does, rather than her own actions?

You don't need to have a fully working device or system in order to have a successful play test. You can use paper mockups, pictures, and whatever else works in order to explain to the tester what happens when she takes certain actions. You benefit the most from play tests when you build very little, get the tester to act out a lot of actions, and ask a lot of questions.

Make a set of instructions to introduce your test, and list of questions to ask your testers after they've interacted with your device or system. Your mock-up should include as little as you need so that you can have other people perform your system. When they do, they will have questions, or will tell you what doesn't make sense. Take note of those things and make changes to your plan accordingly.

Reading for next class:

- Materials: what to choose and where to get
- Making Things

Blog: You've been thinking about and building physical interfaces for a few weeks now. Hopefully you've encountered examples of tangible vs intangible interactions, and have seen examples of implicit vs explicit interactions as well. Has learning to build tangible interfaces changed your view of what constitutes good physical interaction, or has it strengthened your initial ideas?

Class 7

Fabrication

In-class: We'll learn about how to plan a project housing. We'll look closely at the sensors, switches, and knobs we've been using, and measure them to figure out how to mount them in a panel or box. We'll talk about how to use the drill press and the laser cutter, what kind of pre-existing boxes you might use, and how to get a lot out of a little bit of fabrication knowledge.

Homework for next class: Finish the model of your system from last Class. Figure out the size of the full housing needed to hold all the components, measure the components' mounting elements, draw out a plan for your panel, and make it. You can use existing boxes in conjunction with custom pieces. Your model doesn't have to operate fully, but we have to be able to get the look and feel of it, and what immediate responses to controls might be.

If you get the fabrication done and want to prototype some of the behavior of the system, think about incorporating indicator LEDs, buzzers, or vibrating motors that respond to the controls. These help give a sense of the look and feel of a system even if the system itself isn't fully operational.

Reading for next class:

- *Making Things Talk,* 2nd edition, Chapter 6, especially pp 181-187, 190-205. You can read this for free via http://proquestcombo.safaribooksonline.com/ through your NYU Home account. Go to the research tab and search for Safari Books Online. Within Safari, search for Making Things Talk, 2nd edition.
- A little more on a different use of induction: high current loads and motors
    - controlling DC Motors
- Make article on How a Relay Works

Lab for next class:

- Project 10 from *Making Things Talk*, 2nd edition. XBee radios, adapters, and shields should be available for checkout from the Equipment room.

Inspiration for next class:

- Nearfield.org, Nearness video
- Fischli & Weiss, The Way Things Go video

Final project: Create a physically interactive system of your choice. Your focus in this assignment should be on careful and timely sensing of the relevant actions of the person or people that you're designing this for, and on clear, prompt, and effective response. Any interactive system involves listening, thinking, and speaking from both parties. The interaction should be iterative (according to Crawford's definition). Don't just make a system where the user takes one action, the system responds, and it's over. Make a system where the user sees the system's response, and takes more action in response, in a continued loop.

This will be due in week 12. Work in groups or alone. You will have interim assignments related to it in the coming weeks.

Document your work thoroughly online as you go. Include details of all phases of the project. Include a project summary as well, explaining what the system you built is, what it does, and what purpose it's intended to serve. Your summary should introduce the project.

All projects will be presented in the final week of class.

For next class: Blog your initial concepts before next week's class.

Class 8

Present your look-and-feel model

Wireless

In class: *The Telephone game*. The classic version of the telephone game goes like this: person 1 has a message. She whispers it to person 3, who whispers it to person 4, in turn. This continues through several people, and the last person speaks the message out loud. The end message is seldom the same as the first, due to mis-hearing, interpretation, and so forth.

In this class, you're going to play a version of the telephone game that combines wireless communication and mechanics or physical interaction. You'll be divided into groups of 2, each group will be given two radios and two addresses. Group 1 will be given a short message. They will transmit to group 2, who will transmit to group 3, and so forth. The last group will repeat the message to the class. Every group will build a simple system that receives the wireless message, converts it to an analog action, then converts it back into digital form to re-transmit on a second radio. You can blink the bits on an LED and read them back with a photoresistor; you can play them on a speaker and "listen" to them with a microphone; you can use a servo to tap the bits into a pushbutton. You can display them to a human being and have her type them back into the sending radio somehow. Your goal is to come up with the most interesting conduit for the message. Bonus points for involving a human interactively in your design.

For next class: Continue to flush out your final project concepts and document any changes on your blog.

Class 9

Project Planning: This entire class will be given over to planning and beginning to execute your final project. At the end of this class period, you'll present to the class a one- to two-minute description, a system diagram, a timeline of development, a testing plan, start on a bill of materials for your project.

In discussing projects, we'll talk about project construction and planning issues for larger projects, such as CNC construction, advanced circuit planning and construction, communications issues, and other topics related to development of larger scale projects.

Final project: Present your initial concept for your final project to the class.

Reading for next class:

- stepper motors
- Dustyn's writing on:
  - Motors
- Rory Hamilton's notes on preparing presentations and giving presentations

Labs for next class:

- HBridge Lab
- Lab: stepper motors

Homework for next class:

- Complete the Self Test
- Put your project plan, including a timeline, system diagram and brief description and testing plan, on your blog.
- Come up with an initial Bill of Materials (BOM) for your project, and put it online.

Class 10

Motors and mechanics: steppers, gears, and mechanics

Questions on your system diagram of bill of materials.

In-class: Work in groups of three. Make a mechanical pointer. Keep it simple. Make it respond to a physical input or to serial input. It can be a temperature display, an applause-o-meter, a light meter or a direction indicator.Your pointer might point to information on a display like the schmobblometer (http://ithaibenjamin.com/shmobblometer.html) or it might point to a particular person in a room. It might be a lottery indicator. It should be motorized using one of the motor techniques you've learned.

Homework: Develop a prototype of your project that you can use for user testing next week. Use paper or cardboard mockups where possible, show initial versions of any media as needed. Your goal is to use next week to answer any remaining user interaction questions before you build the final version of your prototype.

Class 11

User test your project. In this class, you'll bring the current working version of your project and test it with your classmates. Even though you may not have the whole system working, you should have enough to be able to present a relatively high-fidelity mock-up that you can use to test the interaction. If you plan this well, you'll get invaluable feedback from your classmates about your project *before* you build the whole thing, so you can make changes easily.

We'll split the class into two groups of projects, each of which will test for half the class. When you are a tester, you should try to get to each project. When you are testing your project, you should use the time to gather as much feedback as you can about how the project works.

Take written notes on everything you test, and give it to your classmates for whom you're acting as a tester.

Class 12

Final presentations. You'll present your final prototype in class and give some of your classmates the chance to interact with it. How long you have to present will depend on how many project there are. We'll divide the available class time such that each project gets equal time. In addition to the class critique, take written notes and hand them to your classmates after they present as well.

Grading

Participation & Attendance: 40%
Production Assignments: 40%
Journal: 20%

Participation & Attendance

Showing up on time, engaging in the class discussion, and offering advice and critique on other projects in the class is a major part of your grade. Please be present and prompt. Lateness will hurt your grade. If you're going to be late or absent, please email your instructor in advance. If you have an emergency, please let your instructor know as soon as you can. Please turn in assignments on time as well.

Laptops

Laptop use is fine if you are using your laptop to present in class, or if we're in the middle of an exercise that makes use of it. Whenever classmates are presenting or we're in the midst of a class discussion, however, please keep your laptop closed. The quality of the class depends in large part on the quality of your attention and active participation, so please respect that and close your lid.

Mobile Phones

Please put them on vibrate or turn them off before you come to class unless they are part of your project. If you have an emergency that requires you to answer your phone during class, please tell your instructor ahead of time.

Lab Assignments

There is a lab activity for nearly every class in the first half of the semester. They are very short, simple activities. These are the basic steps you need to go through to understand the principle discussed in class each week. They're designed to help you not only to understand the technical details, but also to get a feel for what the technologies we're discussing can do, so that you can incorporate them into actual applications. You should at least complete the steps outlined in the lab activity each week, so that you understand practically what it is we're talking about. Document on your blog any discoveries you make, pitfalls you hit, and details not covered in the class or the lab that you think will be useful for your fellow students and future students in this class.

Production Assignments

For production assignments, you'll be expected to present your project in class on the day that it's due. If you're working in a group, all group members should be present, and should participate equally in the presentation.

Journal & Documentation

You are expected to keep an online journal of your progress. The purpose of the journal is twofold. First, it is a valuable way for you to communicate to your instructor that you are keeping up with the work in the class. We read the journals to see how students are doing, so you should update your journal regularly throughout the semester. At a minimum, reference to each week's work is expected, as well as reference to the readings, and thorough documentation of the production projects and technical research. Second, the journal is a way to document your work for your own use and that of others. Many ITP students have found themselves using their journals as a place to store notes, code samples, and more.

Good documentation habits for this class:

You may choose to document your major projects in a separate individual or group site if you choose, but you will be expected to link your site to the main site, and contribute to the class site as well nonetheless. Please avoid flash, shockwave, or other sites that are not text-searchable, as they won't show up on search engines for others to use.

Blogs are great for documenting your process, as they're usually defaulted to organizing the information chronologically. However, projects summarized in a blog can be confusing. It's often worthwhile to set up a separate page or pages to summarize your projects when they're done.

You should document your projects thoroughly. Plan in advance, and perhaps as a group, to have what you need to document at least your midterms and finals. Photos, video, drawings, schematics, and notes are all valuable forms of documentation. Explain the project at the beginning of your documentation, so that people who come to the site from outside this class will understand the overview before they get the explanation.

Don't overload your notes with code. If you've made a big improvement on an existing piece of code, post your new code, and link to the code you based it on (just as you would in citing a pervious author in a paper). If you only changed one part of an existing program, post only the part you changed, and link to the original. Make sure any code you post is well-commented, so you and others can understand what it does.

Always cite the sources of your code, the places you learned techniques from, and the inspirations of your ideas. This is the equivalent to citing your sources in a written paper, and copying code or techniques without attribution is plagiarism. few ideas come out of the blue, and your readers can learn a lot from the sources you learned from or were inspired by.

Good documentation should include a description and illustration of your project. You should include what it looks like, what it does, what the user or participant does in response. It should give enough information that someone from outside ITP, who's never seen the project, can get an understanding of what your project is.

You should also include a section describing the workings of the project, aimed at a more informed reader (me, or next year's classmates), so that the details of production are clear. It doesn't have to include every scrap of code and circuit diagram, but it should make clear what the major components of the system are, and how they communicate, and what the code, if any, does.

Work on this as you go, don't put it off until the end. Your fellow classmates will find your notes as useful too.

Pictures and video help a lot.

Some good project summary sites:

- http://www.tweenbots.com/ by Kacie Kinzer
- http://tomgerhardt.com/fireLight/ By Tom Gerhardt. Simple project, doesn't need a lot to introduce it.
- http://thomas-gerhardt.com/itp/FeltResistor/ Nice explanation of why, and summary of how it works, in a sentence or two.

A few good recent sample journals:

- Bona Kim's final project is explained well
- Matt Richardson's blog documents the weekly labs well and he wrote up his final project nicely as well.

- Nicely detailed blog throughout the semester from Roopa Vasudevan
- Another detailed narrative from Justin Lange
- Good narrative blog from Owen Roberts
- Useful questions at the end of each blog post, for next class:
- Nice example of the first Arduino lab turned into a game by Mimi Onouoha
- Moustache switch by Tak Cheung, nicely documented in video.
- Morgen Fleisig's intro to physical computing blog
- Jason Babcock's journal These are notes Jason kept throughout his time at ITP. Each section covers the technical details of a specific project. Sometimes the task is part of a larger project, and sometimes it's a project in itself. This is an excellent example of how to document the tech details of your projects.