**Chapter Title**: Games & Playable Media (GAME) 202: Foundations of Alternative Controller Games

**Author Names**: Edward F. Melcer

**Course University**: University of California, Santa Cruz
**Course College/School**: Baskin School of Engineering
**Course Department/Program**: Games & Playable Media / Serious Games MS Programs (Computational Media Department)
**Course Level**: Graduate
**Course Credits**: 5
**Course Length**: 10 weeks
**Course Medium**: *Face-to-face*
**Course Keywords**: *Alternative Controllers, Novel Interfaces, Physical Computing, Interaction Design, Arduino, Prototyping*
**Catalog Description** (~125-250 words):
This class provides an introduction to physical computing and interaction design concepts critical for building novel physical interfaces. It employs a combination of theory and practice, ranging from gamepad, circuit, and interaction design to in-class activities, homework assignments, and group projects creating alternative controller games.

**Course Purpose and Objectives** (~250-750 words):
Increasingly low cost, availability, and ubiquity of sensors have made incorporating novel technologies into games and their controllers a viable practice for many developers. The resulting alternative controller games have become a popular phenomenon—1) utilizing emerging technologies to expand the scope of digital games through novel material mediums; 2) moving players and designers beyond the screen through diverse and customizable physical interactions; and 3) enabling innovative ways of engaging with both player and spectator through space, objects, and theming. Foundations of Alternative Controller Games provides an introduction to physical computing and interaction design concepts critical for students to build these novel physical interfaces. It employs a combination of theory and practice, ranging from the basics of gamepad, circuit, and interaction design to in-class activities deconstructing existing alternative controller designs and building circuits to test a variety of switches and sensors. Project-based learning is also applied in the form of midterm and final projects that guide students through building their own novel alternative controller game prototypes, combining numerous switches and sensors in tandem with Arduino and Processing to create unique player experiences. This course ultimately aims to improve its students' overall game design skills by broadening their perspective and understanding of how the physical affordances of a controller can drastically impact the interactions and design choices that best fit a game.
The specific course objectives to meet this purpose and goals are as follows:

- Develop a set of skills and experience necessary to design meaningful hybrid digital-physical interactions.
- Analyze context and possible applications of custom physical interfaces for digital games, toys and other interactive experiences.
- Understand the differences and relationship between physical and digital affordances with respect to games.
- Employ physical computing concepts (i.e., the application of various sensors and switches) in the creation of alternative controller games.

- Apply DIY methodologies in order to incorporate open source software tools and rapidly prototype physical interfaces/interactions.
- Create game prototypes that utilize customized novel interfaces and alternative controllers.
- Develop unique portfolio pieces (in the form of alternative controller games) that are submittable to relevant game festivals such as IndieCade, Come Out & Play, alt.ctrl.GDC, IGF, PAX, and so forth.

**Course Context** (~100-500 words):
Foundations of Alternative Controller Games is currently an elective course offered to graduate students as part of the Games & Playable Media and Serious Games professional masters programs within the Computational Media Department at the University of California, Santa Cruz. It has also been offered previously as an upper-level elective course simultaneously open to both the B.S. in Computer Game Design and B.A. in Art & Design: Games & Playable Media degrees. In both cases, the students that attend this class are generally quite experienced in making purely digital games and have some experience creating analog games in the form of board games or playground games. However, these students generally have very little (if any) prior experience designing or playing hybrid digital-physical games. They are also expected to have no real experience with physical computing or prototyping video game hardware. Therefore, the course experience focuses heavily on a broad application of interaction design and physical computing concepts in applied activities rather than diving deeply into electrical engineering or interaction design theory—as students have little prior background in either. I.e., instead of getting "buried in the weeds" on topics that would derail the ability of the course to enable students to quickly and successfully build alternative controller games, the teaching emphasis focuses on a practical understanding of how sensors map real-world phenomena into digital and/or analog output which can be mapped further into input for games. This is done through in-class analysis activities exploring existing systems to develop an intuitive sense of the broad design space for alternative controllers, design activities centered on understanding physical affordances and their relationship to digital affordance for both player and spectator, and in-class prototyping activities that create electronic components, circuits, arcade game interfaces, and so forth to understand how sensors work and can be applied to novel physical interfaces.

**Course Pedagogy** (250-500 words):
This course employs a constructivist pedagogical approach, focusing on learning-through-doing—i.e., in the form of group projects, live coding/prototyping lectures, and a variety of in-class activities applying the technology and theory learned earlier in the lecture—and helping students develop their own skills for learning and applying new technology/information outside of the traditional classroom setting. In this constructivist learning approach, rather than treating information and theory as a set of facts that students are expected to merely memorize for a grade, basic principles of game design, interaction design, physical computing, and so forth are taught as tools. These tools can in turn be applied as lenses to critically assess existing systems and/or combined as techniques to build novel interactions into one's own systems. Tying theory to practice also offers three major benefits to students: 1) it develops strong fundamentals for designing and creating interactive systems through practical experience; 2) it provides concrete, interactive examples to ground and enhance theoretical understanding; and 3) it offers the flexibility to creatively experiment with course material through projects that afford a deeper understanding of underlying concepts.

The learning-through-doing aspect of the course also inherently utilizes the DIY method from which its content draws heavily upon. Students are given homework assignments that tie the theory presented in class to practice, providing open prompts that allow for creation and exploration of custom physical interfaces as well as modification of existing games to incorporate novel physical interactions. Most importantly, students are encouraged through extra credit to be creative in their own explorations at the intersection of digital and physical, ultimately learning through failure as they attempt the challenge of simultaneously combining hardware design with game design in a unique way. In this way, students develop valuable practical skills for discovery learning which helps them move from structured-inquiry to guided-inquiry within the paradigm of inquiry-based learning. To that end, it is critical that this course merges theory, practical examples, and hands-on projects.

**Course Texts, Games, Software, and Hardware** (~200-500 words):

*Textbooks*

- None

*Hardware for Students*

- Arduino Starter Kit ($30 - $50) — There are a number of cheap Arduino hardware starter kits available online that provide both the Arduino microcontroller and a variety of sensors needed for the in-class activities, homework assignments, and group projects. Recommended starter kit is the ELEGOO UNO Project Super Starter Kit, however most are fine. The kits should come with a minimum of the following sensors to be useful for all class lectures/assignments (most do):
    - Breadboard
    - 30 x Jumper Wires
    - 10 x Resistors (1K or 10k preferred but most resistance values will work fine)
    - 4 x Pushbuttons/Buttons
    - 4 x LEDs
    - 1 x Potentiometer
    - 1 x Passive Buzzer
    - 1 x Photoresistor
    - 1 x Tilt Switch
    - 1 x Ultrasonic Sensor
- Laptop with Web Camera — Running software and web camera for AR.

Hardware for Instructors

- Arduino Starter Kit — Same as the students' hardware kits for live coding/prototyping during lectures.
- Additional Sensors and Materials — These sensors and materials typically don't come with starter kits, but can be purchased relatively inexpensively in bulk from most electronics websites (e.g., SparkFun, Adafruit, etc.) and returned/used over multiple years:
    - Electret Microphone
    - Piezo Element
    - Velostat
    - Single-sided Conductive Tape
    - Electrical Tape

*Software*

- **Arduino Web Editor (Free)** — Communication with the Arduino microcontroller.
    - https://create.arduino.cc/editor
- **Processing and/or Unity (Free)** — Prototyping of digital games and communication with Arduino microcontroller via serial communication.

    - https://processing.org/download/ | https://unity3d.com/get-unity/download
- **Fritzing (€8 for Executable, Free from Source Code)** — Circuit diagramming of hardware interfaces and prototypes.
    - https://fritzing.org/download/

**Course Assignments** (~500-1500 words):
*In-Class Activities* — Each in class activity is designed to build upon what was taught earlier in that class through lecture by providing a more creative context to apply the new piece of technology, sensing technique, design analysis, etc.

*Homework* — Each homework assignment is designed to provide additional practice outside of class that reinforces knowledge, skills, and design thinking learned from previous weeks. Homework will focus more heavily on brainstorming and understanding how to apply sensing techniques in novel ways within alternative controller games. Please see expanded course outline for individual homework assignments and objectives.

*Presentations* — One core aspect of being a game developer that creates alternative controller games is submitting and presenting such games at festivals since festivals are the primary medium to distribute and showcase such work. Learning to give an elevator pitch and quickly present any game (but especially an alternative controller game) is a crucial skill that is typically only developed through experience of doing so. Therefore, most of the homework assignments also come with an in-class presentation component to help students develop their presentation skills and ability to discuss the design of an alternative controller game.

*Midterm Project* — The midterm project focuses on tying learning content from multiple weeks together in order to produce more complex alternative controller games (i.e., using Arduino and a variety of sensors with processing). It also allows small teams of students to think more broadly about how the various sensing technologies they have learned can be incorporated into games and enables them to apply this knowledge in the creation of a more ambitious **and functional** game prototype than on the homework assignments.

*Final Project* — The final project builds upon development of applied skills from the midterm, enabling student teams to polish their designs, games, and hardware from either the midterm or a student's previous homework assignment. Student teams are also able to create a new game from scratch for the final projects if they desire, but this is not advisable without a strong creative vision for the game before starting. This project also focuses on teaching students how to present this kind of alternative controller work to the outside world, and the best ways/venues to do so.

**Course Assessment** (potential bulleted list):
- Class participation 10% of grade (attendance, discussions, and in-class activities).
- Presentations 10% of grade (presenting certain homework assignments to the class).
- Homework 30% of grade (each homework assignment counts equally).
- Midterm project 25% of grade.
- Final project 25% of grade.

**Expanded Course Outline** (500-1500 words/whatever necessary to complete the table): *Use this space to provide an expanded outline of the course materials. This should be in line with the course length* *highlighted above. Add more rows to the table as you need to. Consider linking to outside materials (e.g., videos, open access Dropbox or Drive files, readings, websites, etc.) to strengthen your chapter. Write to an audience that wants to attempt to replicate your work; the more materials you provide, the better.*

| Week or Module # | Topic | Class Topics/Activities | Assignments |
|---|---|---|---|
| 1 | *Introduction & Overview of Alternative Controllers* | • *Class introductions*<br>• *Introduction to the syllabus*<br>• *Class overview*<br>• *Introduction*<br>  ○ *What is a controller?*<br>  ○ *What are alternative controllers?*<br>• *Alternative game controllers and interfaces*<br>  ○ *Overview of commercial alternative controllers and hardware for making them*<br>• *Alternative controller games*<br>  ○ *Examples, analysis, and discussion*<br>• *Understanding and designing for affordances of the controller, environment, and body*<br>  ○ *Broadening our understanding of affordances*<br>  ○ *Why physical affordances matter to digital game design*<br>  ○ *Designing for spectacle/spectators*<br>• *Embracing the DIY mindset*<br>• *Introduction to Arduino and Processing/Unity, your tools for the quarter* | ***Homework 1 – Gameshow Interface Analysis***<br>• *Look up an existing game show and identify the interface(s) it provides for contestants.*<br>• *Analyze how different aspects of the design impact player and spectator experiences. E.g., think about narratives, affordances, physical interactions, gameplay, and emotions that the interfaces evoke.*<br>• *Create 3 - 5 slides highlighting some of these aspects*<br>  ○ *The first slide should explain how the game show works or have a link to a video that shows the core gameplay.*<br>  ○ *The remaining 2 - 4 slides should highlight some of the points above.*<br>  ○ *Use images, gifs, or videos of the gameshow to help illustrate your point.*<br>• *Present these slides at the beginning of class.*<br>• ***DUE Week/Module 2*** |
| 2 | *Basic Circuits, Digital Input, & Digital Output* | • *What is a circuit?*<br>• *Introduction to currents*<br>  ○ *AC vs. DC*<br>• *Ohms Law*<br>• *Conductors vs. Insulators* | ***In-class Activity 1 – Building a Family Feud Interface***<br>• *Watch videos of people using the Family Feud button/buzzer interface* |

| | | | |
|---|---|---|---|
| | | ◦ *Examples of surprising insulators and conductors (such as the Jello piano)*<br>• *Understanding a breadboard*<br>• *What is a circuit diagram?*<br>◦ *Fritzing introduction and example*<br>• *Debugging a circuit (various approaches)*<br>• *Building circuits with Arduino*<br>• *Understanding digital input and output*<br>◦ *Using LEDs for output*<br>◦ *Using switches for input*<br>◦ *Live coded example of both*<br>◦ *In-class activity to reinforce concepts* | • *Recreate that interface in class to practice wiring and coding buttons as input and LEDs as output*<br>• *The Family Feud interface should have the following:*<br>◦ *Two buttons (one for each contestant)*<br>◦ *Two LEDs (one for each contestant)*<br>◦ *The first contestant to hit their button will cause their LED to light up*<br>◦ *The other contestant's LED can no longer light up until the Arduino is reset*<br><br>***Homework 2 – Simple Button & LED Game***<br>• *Using the Family Feud Interface created during class as inspiration, sketch the design of and build a game interface that uses* ***multiple*** *switches and LEDs. For example, a Jeopardy game interface with 3 buzzers, Auto Race, or a Simon game.*<br>• *Before working with any hardware, first sketch circuit diagram for the game interface using Fritzing.*<br>• *Build a working interface using that sketch as the starting point.*<br>• *1 minute presentation of the game at the beginning of class.*<br>• ***DUE Week/Module 3*** |
| 3 | Thinking in Analog: Diversifying Input & Output | • *Switch vs. Sensor*<br>◦ *Switches provide discrete digital values (on/off or 1/0)*<br>◦ *Sensors provide continuous ranges*<br>• *Understanding analog input and output* | ***In-class Activity 2 – Create a Flex sensor from scratch***<br>• *Flex sensors are expensive to get on their own, but luckily it is possible to create our own homemade flex sensors from scratch using velostat, single-sided conductive tape, foam, electrical tape, and hot glue.* |

| | | | |
|---|---|---|---|
| | | ◦ *Visual/video examples*<br>• *Using variable resistors (Potentiometers)*<br>◦ *Live coded example*<br>• *Using Flex and Pressure sensors for input*<br>◦ *In-class activity making a custom flex/pressure sensor from scratch*<br><br>• *What are piezo elements?*<br>• *How does sound work?*<br>◦ *Understanding your speakers*<br>• *Using Buzzers for output*<br>• *Controlling tones with the Arduino*<br>◦ *Live coded example* | • Follow the instructions here to create the flex sensor in class. |
| 4 | Things are Getting Serial: Communication Between Arduino and Processing | • *Serial data, communication between Arduino and Processing*<br>◦ *Introduction to the Serial Monitor*<br>◦ *Debugging Arduino code using the Serial Monitor*<br>• *Sending serial messages from Arduino to Processing*<br>◦ *Using sound in processing*<br>◦ *Live coded example*<br>• *Sending serial messages from Processing to Arduino*<br>◦ *Live coded example*<br><br>• *Using a sensor as a switch*<br>• *Bouncing in a switch (debouncing to avoid the wobble effect)*<br>• *Using photoresistors and Piezo elements as input*<br>• *In-class activity: Infinite Whack-A-Mole game* | ***In-class Activity 3 – Infinite Whack-A-Mole game***<br>• Build a simple single button interface with Arduino that sends a message to Processing when the button is pressed or released.<br>• Build a simple infinite Whack-A-Mole game in Processing that tries to whack the mole whenever it receives a button press message from Arduino via serial.<br><br>***Homework 3 – Brainstorming Arduino with Processing Games***<br>• *Brainstorm 2 potential games or interactive experiences that use Arduino and Processing for your midterm project. These can be entirely original or (*substantial*) expansions on existing prototypes you made for past homeworks. You can use any hardware for the Arduino, even if we haven't* |

| | | | |
|---|---|---|---|
| | | | *covered it in-class. Make sure to bodystorm the interactions to ensure that they are physically reasonable for the player.*<br>• Submit sketches of each interface and a 1 – 3 paragraph description for how each game would work.<br>• ***DUE Week/Module 5*** |
| 5 | All About the Beat: Detecting Sounds in Arduino | • *Faking sensing - degrees of separation between perceived input method and actual sensor detection*<br>  ○ *Examples of faking sensing*<br>  ○ *E.g., the Nintendo DS detecting how hard a player blew on the microphone*<br>• *Understanding analog sound and transducers*<br>• *Using electret microphones*<br>  ○ *Live coded example*<br>• *Students form teams for midterm project*<br>• *Second class, work on midterm project in class* | **Midterm Project**<br>• Form teams of 2 – 4 students per project.<br>• *Choose one design from Homework 3 to develop out into an alternative controller game that uses both Arduino and Processing/Unity.*<br>• Make sure to create a Fritzing sketch before starting prototyping of the actual circuit.<br>• Submit the fritzing circuit diagram for the alternative controller and source code for Arduino and Processing.<br>• Present the game at the beginning of class in week/module 7.<br>• ***DUE Week/Module 7*** |
| 6 | Guest Lecture & Midterm Project | • *One guest lecture covering development or use of alternative interfaces for games in first class*<br>• *Work on midterm project in second class* | |
| 7 | Sensing Motion and Distance | • *Playtest session showcasing midterm projects for first class*<br>  ○ *All teams take turns going around and playing each other's alternative controller games*<br>  ○ *The instructor visits each team to test* | ***In-class Activity 4 – Building a Custom Motion Controller***<br>• Accelerometers are expensive and not the easiest piece of hardware to work with. Luckily, if we just care about the direction of motion (and not the speed of it) then it's possible to instead use tilt sensors to create a motion controller. |

| | | | |
|---|---|---|---|
| | | *game and determine midterm grade*<br>○ *Teams must give instructor 1 minute presentation on the game and how to play (i.e., elevator pitch)*<br><br>• *Discussion of how sonar works*<br>• *Using ultrasonic distance sensors to detect distance of objects*<br>○ *Live coded example*<br>• *Using tilt switches as an alternative to accelerometers*<br>• *In-class activity: Building a custom motion controller* | • Use 2 tilt sensors to detect when the breadboard is tilted left, right, or is level.<br>• Print the state of the breadboard’s motion to serial.<br>• If there is extra time, create a simple Processing sketch that allows the breadboard motion to control the movement of an onscreen object. |
| 8 | Sensing Objects through Computer Vision: Engaging with Objects and the Body | • *Form final project teams and start working on final projects*<br>• *What is augmented reality?*<br>○ *Examples and discussion of AR games*<br>• *Window-on-the-World (WoW) vs Word-as-Support (WaS) interaction paradigms for AR [1]*<br>• *Understanding infrared and computer vision*<br>• *Commercially available technology for broader distribution (camera and Kinect)*<br>• *Detecting the body with Microsoft Kinect*<br>○ *Video examples*<br>○ *Discussion of why it failed*<br>• *Face detection and tracking with Ketai OpenCV and Processing* | ***In-class Activity 5 – Building an AR Mask Application***<br>• Use the Ketai library in Processing to perform basic face detection with a web camera.<br>• When the application detects a face, draw a mask image over it.<br><br>**Final Project**<br>• Form new teams of 2 – 4 students per project (or keep the original team from the midterm project).<br>• *Either continue working on finishing/polishing the midterm project, choose a past homework to develop out further into an alternative controller game, or create a new alternative controller game from scratch (not advisable unless there is a clear creative direction for the game).*<br>• *Games must use both Arduino and Processing/Unity.* |

<table>
<tr><td></td><td></td><td><ul><li>Fiducial markers and tracking objects with reacTIVision, TUIO, and processing</li><li>In-class activity: Building an AR mask application</li></ul></td><td><ul><li>Make sure to create a Fritzing sketch before starting prototyping of the actual circuit.</li><li>Submit the fritzing circuit diagram for the alternative controller and source code for Arduino and Processing.</li><li>Extra credit for creating a video trailer of the game (for documentation and submission to festivals in the future).</li><li>Extra extra credit for submitting the game to an actual festival.<ul><li>Good festivals for submission include: alt.ctrl.GDC, IndieCade, IGF, Come Out & Play, and A MAZE.</li></ul></li><li>Present the final game for the last class of week/module 10.</li><li>DUE Week/Module 10</li></ul></td></tr>
<tr><td>9</td><td>Making Robust Alternative Controllers: Soldering and Conductive Thread</td><td><ul><li>What could go wrong? Challenges in public display deployments [2]<ul><li>Taxonomy of six categories for things that can go wrong with public displays: weather, events, surroundings, space, inhabitants, and vandalism</li></ul></li><li>Discuss ways to make alternative controller games more robust</li><li>Basics of soldering<ul><li>Explain various soldering tools</li><li>Discuss proper way to solder and soldering techniques</li></ul></li><li>The LilyPad, wearables, and conductive thread as an alternative to solder</li></ul></td><td>In-class Activity 6 – Soldering Practice<ul><li>Setup several protoboards, soldering irons, solder, soldering fans, jumper wire, and helping hands.</li><li>Allow students to take turns soldering the jumper wire to the protoboards under direct teacher supervision.</li></ul></td></tr>
</table>

| | | | |
|---|---|---|---|
| | | • *In-class activity: soldering to a protoboard (**ONLY under instructor supervision**)* | |
| 10 | Final Project Polish and Playtest Presentations | • *Work in Final Project groups for class 1*<br>• *Present Final Projects through class playtest session for class 2* | *Great job and good luck with finals!* |

**Course Best Practices** (250-1000 words):

Tips and Tricks

- Learning to prototype circuits is difficult for students with no prior experience to understand, even with pictures, videos, and circuit diagramming software such as Fritzing. One way to greatly help students in understanding how to build a certain circuit is to do "live wiring" during class lecture. I.e., setup an overhead camera (such as a document scanner camera), and use it to project the breadboard, circuit, and your hands during class while you are wiring a circuit.
- Providing real-world contexts and application areas for alternative controllers can serve to greatly motivate students and help them view their work as more relevant. Giving extra credit on the final project for submitting their alternative controller game is one nice potential way to add motivation for students and help them to build a quality game portfolio.

Pitfalls

- One assignment that didn't work extremely well in the past was to create a video trailer of the final project game. While this was beneficial as it helped students to document their games more formally and provided them with more submission material for festivals, it also provided difficult for a number of students to create a video trailer in a short one-week span. Making non-game design/development assignments optional (or for extra credit) can prove more flexible in providing additional challenge to students that are excelling in the class while alleviating pressure on those that are struggling a bit more.

Alternative Controller Game Examples from Past Courses

- *HyperMasculinity* by Fernando Tapia, Cory Super, and Charisse Lo
    - https://youtu.be/ox-30ISmpiM
- *Sengoku Rhythm* by Eisaku Imura and Hesiquio Mendez Alejo
    - https://youtu.be/Gk7FH0l9Wug
- *Beat Shift* by Bradley Matias
    - https://youtu.be/MVs-ZCFF6CY
- *Laser Archery* by Andrew Cousins and Mallory Strout
    - https://youtu.be/PGS1Gm5c1so

**Future Course Plans** (250-500 word):

Alternative controller games have continued to grow in popularity in recent years, with venues such as alt.ctrl.GDC and Night Games at IndieCade garnering major attention. With this rise in popularity comes additional polish and novel approaches that push the boundaries of what an alternative controller game is. One such direction that would be interesting to incorporate into the course would be an assignment on the repurposing of existing analog technology (e.g., adding sensors to a bike, couch, or television or even providing new ways to control a sewing machine or 3d printer). Similarly, teaching how to use fabrication technologies such as 3d printers and CNC machines within the course would enable students to create far more polished interfaces for their games and provide them with some valuable new skills. Additionally, escape rooms are a very common application of this alternative controller technology, so an assignment and class discussing the design of escape room puzzles/technology would be interesting to try. Finally, LARPing and the use of wearables has become increasingly popular, so a class discussing technologies currently employed in LARPing and assignment to build a LARP tool could prove interesting. Ultimately, this class addresses a wide range of games, communities, and applications that are generally ignored by most commercial game companies. I imagine this class broadening over the next 3 – 5 years in order to better address these often overlooked but quickly growing application areas.

**References**:


1. Malinverni, L., Maya, J., Schaper, M. M., & Pares, N. (2017, May). The World-as-Support: Embodied Exploration, Understanding and Meaning-Making of the Augmented World. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (pp. 5132-5144).
2. Mäkelä, V., Sharma, S., Hakulinen, J., Heimonen, T., & Turunen, M. (2017, May). Challenges in public display deployments: A taxonomy of external factors. In Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems (pp. 3426-3475).