

A Low-Cost, Scalable Mechatronics Course Centered on Tinker-and-Build Laboratories and Project-Based Learning

Pranav A. Bhounsule*, Daniel Torres, Salvador Echeveste

*Corresponding author: pranav@uic.edu

Department of Mechanical & Industrial Engineering
University of Illinois Chicago

Abstract

Mechatronics education integrates mechanical systems, electronics, and computing to prepare students for modern interdisciplinary engineering practice. Mechatronics education achieves its greatest impact when theoretical instruction is reinforced through integrated laboratory experiences. Two challenges limit the effectiveness and scalability of existing laboratory models: insufficient alignment between lectures and laboratories, and the high cost of setting up laboratory experiences. This paper presents a low-cost, scalable laboratory framework centered on a “tinker-and-build” philosophy that leverages the do-it-yourself (DIY) ecosystem of the Arduino community, and browser-based design and circuit simulation platforms built on TinkerCAD. By tightly integrating textbook content with hobby-grade electronics and open-source development environments, the proposed approach enables reproducible, affordable, scalable, and adaptable laboratories that directly reinforce course concepts. The framework reduces financial barriers while maintaining pedagogical rigor, thereby expanding access to experiential mechatronics education to institutions with limited resources.

Introduction

Overview: Mechatronics, which integrates mechanical systems with electronics and computer programming to create intelligent systems, has become increasingly important as modern engineering systems grow more interdisciplinary. In response, many textbooks have emerged to support undergraduate mechatronics education. Although these texts are effective at conveying theory and principles, mechatronics is fundamentally a hands-on discipline, and meaningful learning ultimately depends on laboratory experiences that allow students to translate concepts into functional systems.

Two Major Challenges: Developing effective mechatronics laboratories presents two major challenges. First, there is a lack of laboratory content that tightly couples with lecture material, often resulting in labs that are disconnected from the course concepts. Second, the cost of establishing and maintaining mechatronics laboratories can be prohibitively high, particularly for institutions with limited resources. Specialized hardware, costly software, and low student-to-equipment ratios limit scalability and accessibility, hindering programs' ability to offer enriching experiential learning experiences.

Our Contribution: This work addresses both challenges by leveraging the growing do-it-yourself (DIY) and hobbyist ecosystem. Online communities have produced extensive educational resources, such as TinkerCAD for teaching mechanical design and simulating electrical circuits, and the Arduino community for electronic integration and programming using free, open-source

Integrated Development Environments (IDEs). The large user base has driven low cost, rapid development, and robust community support, making these tools reliable and accessible. This paper integrates existing mechatronics textbook content with low-cost, hobby-grade electronics and microcontrollers to develop a scalable and inexpensive laboratory framework that aligns directly with lecture material. The resulting approach enables institutions with limited resources to implement reproducible, affordable, scalable, and adaptable hands-on mechatronics laboratories.

Background and Related Work

Mechatronics Labs: Early efforts in mechatronics laboratory development were supported through U.S. National Science Foundation–funded initiatives, which enabled the creation of comprehensive laboratory infrastructures but typically relied on specialized hardware, dedicated laboratory space, and substantial institutional investment [1]. To mitigate cost concerns, some approaches emphasized the use of reusable or modular components to extend the lifespan of laboratory equipment and reduce per-student costs [2]. To address space limitations and increase flexibility, portable laboratory kits were developed, often in conjunction with flipped-classroom pedagogies [3].

Alternative approaches have explored using cloud-based services to further reduce hardware costs and physical space requirements; however, such solutions may be ill-suited for courses that rely heavily on hands-on experimentation as a core component of the learning experience [4]. More recently, some universities have partnered directly with industry to develop automation workstations aligned with contemporary industrial practices. While these platforms expose students to realistic tools and workflows, they often present a steep learning curve for novice instructors and require significant instructional support [5]. In a similar vein, commercial providers such as Quanser offer high-fidelity mechatronics systems that integrate seamlessly with MATLAB/Simulink; however, individual systems typically cost several hundred dollars, making them a substantial financial investment for many educational programs [6]. The authors have skimmed multiple mechatronics books, but few offer a cohesive lecture-laboratory integration, which impedes instructors from developing a lab-integrated mechatronics course.

Project-based Learning (PBL): PBL approaches grounded in societal and industrial challenges require students to integrate technical knowledge from coursework with professional skills such as communication, collaboration, and project management [7]. In early undergraduate implementations, students were tasked with identifying knowledge gaps, formulating project concepts, selecting components, and iteratively refining their designs. These activities led to significant improvements in problem-solving, creativity, and teamwork compared with more prescriptive instructional approaches [8]. Furthermore, students exposed to PBL have shown to exhibit substantially higher levels of preparedness for capstone or graduation projects, with faculty supervisors reporting marked improvements in technical proficiency and problem-solving skills relative to non-PBL cohorts [9].

PBL has also been implemented through a wide range of project modalities, highlighting its flexibility across instructional contexts. Examples include hands-on projects focused on

assembling and controlling radio-controlled aircraft to teach aeronautics, plant modeling, and mechatronics principles [10], as well as competition-based formats—such as sumo robot challenges—that motivate student engagement through structured constraints and peer comparison [11], [12]. Collectively, these studies demonstrate that PBL can be effectively adapted to diverse mechatronics topics and instructional levels, making it a powerful framework for experiential and integrative engineering education.

Overview of the Course:

Course Offerings and Structure: The mechatronics course was developed and taught by the author on five occasions in two large public universities. It was offered three times as a core senior undergraduate mechanical engineering course (Fall 2015, Spring 2015, and Fall 2018) and twice as a cross-listed course, serving as a technical elective for senior undergraduate students and as a regular course for graduate students (Spring 2023 and Fall 2025). The core offerings enrolled approximately 90–120 students per term, organized into four-person teams for laboratory and project activities. In contrast, the cross-listed offerings enrolled 20–30 students per term, with students working in pairs on laboratory and project assignments.

The course met for two 50-minute lectures each week, complemented by a two-hour laboratory session. Student assessment was based on a combination of homework assignments from the text, laboratory exercises, project work, and closed-book examinations created by the authors.

Availability of Materials: All course materials, including lecture notes, video recordings, lab write-ups, project write-ups, and other relevant materials, as well as the associated costs per student, are available, <https://tiny.cc/mechatronix> [13].

Textbook limitations that prompted the development of labs and projects: The textbook “*Introduction to Mechatronics and Measurement Systems*” by David Alciatore (5th ed., 2019) served as the primary reference for developing the lecture notes and assigning homework problems. The text provides comprehensive coverage of foundational topics, including resistors, capacitors, inductors, diodes, transistors, amplifiers, logic gates, flip-flops, analog and digital circuits, and microcontroller programming using PIC-series in BASIC.

While the textbook includes an accompanying laboratory manual, several limitations motivated the development of alternative laboratory and project components. *From the student perspective*, the laboratory exercises rely on PIC microcontrollers programmed in BASIC, a language that is typically not part of the standard mechanical engineering curriculum. Additionally, PIC microcontrollers are relatively challenging to interface, often requiring external breadboards and additional circuitry for programming, and they have a comparatively limited user community. *From the instructor’s perspective*, the laboratory activities are not clearly integrated with the lecture material presented in the textbook, making it challenging to align the theory with hands-on implementation. Furthermore, the laboratory framework does not provide clear guidance on how activities can be scaled to accommodate large enrollments, a critical consideration, especially for institutions with limited resources.

Lectures and Laboratories: While the textbook was retained for lecture content and homework, the authors developed a custom laboratory component that was tightly integrated with course delivery. The laboratory sequence was built around the Arduino microcontroller platform, which is inexpensive, widely adopted, and supported by a large open-source community. The Arduino environment uses a C/C++-based, open-source integrated development environment, making it more accessible and better aligned with engineering curricula.

The lectures and labs covered building and analysis of electrical circuits, resistors, capacitors, diodes, transistors, op-amp circuits) using function generators, oscilloscopes, and voltmeters/ammeters; Arduino-based integration and programming with sensors (ultrasonic, touch, tilt, photocell, and joystick) and actuators (servos, stepper, and DC motors); and essential DIY skills including soldering, mechanical design, and 3D printing.

The overall course design followed an incremental and scaffolded pedagogical approach. Each week, new concepts were introduced in lecture, reinforced through homework assignments, and then implemented in a corresponding laboratory exercise.

Figure 1 and Figure 2 gives two examples of lab instructions. Table 1 provides an overview of the course organization over the 16-week period. The weekly laboratories (Weeks 2–11) were designed to build progressively in complexity and capability, culminating in a comprehensive final project conducted during Weeks 12–15. The final project required students to integrate sensing, actuation, microcontroller programming, and mechanical design into a functional mechatronic system.

Exams: The authors implemented and evaluated two examination models: (i) a single midterm examination followed by a comprehensive final examination, and (ii) two non-cumulative term examinations. Both models were found to be effective in assessing student learning and were successfully implemented across various course offerings. The examinations consisted of a combination of multiple-choice questions and traditional long-answer problems requiring analytical reasoning and problem-solving.

For all examinations, students were permitted to use a single letter-size formula sheet, with content allowed on both sides. This policy was intended to reduce rote memorization and emphasize conceptual understanding and problem-solving skills. Gradescope, through a university license, was extensively used to enable efficient grading of the examination.

2.4 (20 pts) Ultrasonic sensor, a distance sensor

This webpage gives theory behind ultrasonic sensor and how to hook it up: <https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/>. Hook up the sensor to the arduino and calibrate it to read the distance. Demonstrate that the sensor is able to measure the distance and show it to the TA.

2.5 (20 pts) DC motor

This webpage shows how to interface and program the DC motor. <https://create.arduino.cc/projecthub/ryanchan/how-to-use-the-l298n-motor-driver-b124c5>. Demonstrate that you are able to spin a single DC motor back and forth using the H-bridge and various speeds.

2.6 (30 pts) Application as a backup sensor

Create a prototype of an automated backup system for an electric car. The DC motor is the proxy for the rear wheel. The DC Motor should be freely spinning. Then, as the distance between the sensor and the obstacle decreases, the speed of the DC motor should decrease till it reaches zero when speed reaches a critical value. Demonstrate the prototype to the teaching assistant.

Figure 1: Lab 10 is one of the two labs introducing students to sensors and actuators. In sections 2.4 and 2.5, students learn about interfacing the sensor and actuator, respectively. In section 2.6, they apply this knowledge to a practical application. Note the extensive use of tutorial links generated by other users to help guide their work.

2.1 Function Generator and Oscilloscope

2.1.1 (20 points) Simulation in TinkerCAD

Watch this video on using the TinkerCAD Function generator and Oscilloscope. <https://youtu.be/wfvYz1L0Qkk>. Connect the Function generator to the oscilloscope. Play with the frequency, amplitude, DC offset, and function on the function generator and note the output on the oscilloscope.

1. Generate a square wave of frequency of 10 Hz and amplitude of 10 V. Show the output on the oscilloscope.
2. Now add an offset of 5V. What happens to the output?

Show your work to the teaching assistant.

2.1.2 (20 points) Hardware

Repeat the above TinkerCAD exercise but using the oscilloscope/function generator in the lab. You could also use 2 Hantek; one as a function generator and one as an oscilloscope. Here is a short video showing how to connect and use a function generator/oscilloscope <https://youtu.be/jWDJeiH6vcQ>. Show your work to the teaching assistant.

Figure 2: Lab 3 on capacitors. First, students learn how to use a function generator & oscilloscope to measure the response. Learning is first through a simulator, TinkerCAD, and then in hardware. Thereafter, they proceed to assemble, test, and evaluate the RC low & high pass filter in hardware. By learning to simulate a circuit, the students are better able to understand the hardware experiment.

Table 1: Schedule for Lecture, Labs, Exams, and Project. There was HW assigned each week starting from Week 1. The Labs covered ABET outcomes 1,5,6, and the Project covered ABET outcomes 1,5,6,7. Also see <https://tiny.cc/mechatronix>

Wk	Lectures	Labs
1	Basic Electrical Components; Kirchoff's Current/Voltage Law	No Lab
2	Analyzing Circuits; Practical Electronics; Characteristics of a Good Measurement System	Lab 1 – Tinker CAD
3	Diodes (Junction, Zener, LED, Photodiode)	Lab 2 – Resistors (Voltage/Current Divider)
4	Bipolar Junction Transistor (Theory & Examples)	Lab 3 – Capacitors (High/Low Pass Filter)
5	Amplifiers (Op-amp Basics, Inverting/Non-inverting, Applications)	Lab 4 – Diodes
6	Op-amp Applications (Instrumentation, Comparator, Sample & Hold); Real Op-amp Specs	Lab 5 – Bipolar Junction Transistor
7	Exam 1	Lab 6 – Integrated Circuits
8	Digital Circuits; Logic Gates; Boolean Algebra	Lab 7 – Soldering

Wk	Lectures	Labs
9	Logic Gates (Problems & Boolean Expressions)	Lab 8 – Arduino (Hardware)
10	Sequential Logic; Flip Flops; Applications	Lab 9 – Design and 3D Printing
11	Counters, Decoders, Timers, Data Sampling; Project: Interactive Animatronics	Lab 10 – Sensors and Actuators 1
12	Data Quantization; A/D and D/A Converters	Lab 11 – Sensors and Actuators 2
13	Sensors (Emitter/Detector, Ultrasonic, RFID, Encoders); Measurement (Strain, Force, Temp, Vibration)	Makeup Lab – Students can make up any missed lab, Project week 1
14	Actuators (Solenoids, DC Motors, Stepper, Brushless); Hydraulics & Pneumatics	Project week 2
15	Exam 2	Project week 3
16	Public Demonstration of project	

Final Project: The authors implemented two distinct final projects across different course offerings to evaluate alternative modes of project-based learning.

(1) Sumo-Bot Project

In this project, student teams designed and built a wheeled mobile robot capable of pushing an opposing robot out of a defined arena with a clearly marked boundary. The project incorporated realistic engineering constraints, including limits on robot weight, overall dimensions, battery capacity, and motor selection. These constraints required students to balance mechanical design, actuation, sensing, and control while considering trade-offs related to power and robustness.

(2) Animatronics Project

In this project, student teams designed and built an animatronic face representing a well-known personality, with interactive behaviors enabled using sensors and actuators. Typical implementations utilized ultrasonic sensors to detect user gestures (e.g., a waving hand), triggering motor-driven facial motions such as mouth movements synced with voice recordings or exaggerated movements of eyes, ears, or other props, like a hat. This project emphasized human-machine interaction, creative mechanical design, and the integration of sensing, actuation, and embedded control.

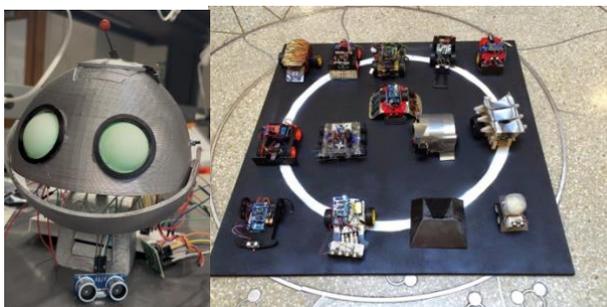


Figure 3: Spring 2025, Animatronic of Clank (PS2), and video: <https://www.instructables.com/Clank-Animatronic/> and <https://youtu.be/2su75F3SnYI> Fall 2015, Sumo-bots.

See Figure 3 for some students' projects. Reports from Fall 2018 Animatronics project are here: <https://github.com/pab47/roboprof> and the YouTube Playlist is here: https://www.youtube.com/playlist?list=PLc7bpbeTIk77kdx4O8HoIr_Ep76HiSwil. A video of sumo bots from Spring 2015 is here: https://www.youtube.com/watch?v=dk8n1J9m_4Q. Both projects were supported by clearly defined project descriptions, expectations, and assessment criteria, including explicit milestones and weekly deliverables. The project timeline spanned five

weeks: the first week focused on background research and a brief literature review; the second week emphasized conceptual design and system architecture; Weeks 3–4 was dedicated to fabrication, integration, and testing; and the final week culminated in a public demonstration and presentation.

The final project report was a major graded component and required detailed documentation, including step-by-step implementation procedures, a complete bill of materials, embedded code, demonstration videos, and a reflective discussion of lessons learned and suggested improvements. This emphasis on documentation reinforced engineering communication skills and promoted reproducibility and design iteration.

Student Feedback: Despite the breadth and rigor of the material covered, the course consistently received high student evaluations across multiple offerings. Qualitative feedback and course evaluations indicated that students particularly valued the hands-on laboratory and project components. These activities enabled students to directly apply theoretical concepts and develop practical skills, thereby reinforcing learning.

Student feedback further suggested that the opportunity to gain exposure to electrical and electronic concepts was especially beneficial for mechanical engineering majors, for whom such topics are often underrepresented in the core curriculum. The integration of electrical systems, sensing, actuation, and embedded programming within a mechanical design context was perceived as a key strength of the course and a significant contributor to its overall effectiveness. The course consistently had a high course/instructor rating and drew higher enrolments relative to other technical electives.

Discussion

Placement of the course in the curriculum: Although the course was offered at the senior (fourth-year undergraduate) level, the authors strongly recommend positioning a mechatronics course at the junior (third-year undergraduate) level. At this stage, the course can serve as an effective feeder for senior design and capstone projects focused on automation, robotics, or controls. Since students are introduced to basic electrical circuits during the sophomore year, offering mechatronics in the junior year allows students to build on this knowledge while it is still fresh, thereby improving the retention and integration of electrical, mechanical, and programming concepts.

Course Delivery and Pedagogical Approach: The instructor found that segmenting lectures into shorter conceptual “chunks,” supplemented with carefully selected online mechatronics videos, increased student engagement. In addition, the course deliberately encouraged real-time information seeking, such as estimating component costs or looking up datasheets during lecture hours. This approach helped connect learning with professional engineering practices, reinforced self-directed learning, and modeled how engineers acquire information in real-world settings.

Alignment with ABET student outcomes: The seven ABET student outcomes may be summarized as: (1) engineering problem solving; (2) engineering design under constraints; (3) effective technical communication; (4) ethical and societal considerations; (5) teamwork and leadership; (6)

experimentation and data analysis; and (7) lifelong learning. When offered as a technical elective with the animatronics project, the course most directly addressed Outcomes 1, 5, 6, and 7. With targeted modifications, additional outcomes can be emphasized. For example, Outcome 2 was explicitly addressed in the Sumo-bot project through constraints on size, weight, power, and components. This flexibility demonstrates that the course structure can be readily adapted to meet a broader set of accreditation objectives.

Use of TinkerCAD for circuits and design: Several homework and laboratory exercises incorporated TinkerCAD (<https://www.tinkercad.com>) for circuit simulation and preliminary design prior to physical implementation. These activities introduced students to the value of simulation as a design and debugging tool, allowing them to test circuits virtually before building them on a breadboard, thereby reducing trial-and-error time and component damage. TinkerCAD was also used for 3D design and printing; however, students were free to use alternative CAD platforms if preferred.

Costs and Scalability: There are two parts of the cost: 1) One-time infrastructure costs for shared equipment, which included soldering stations, oscilloscopes, multimeters, and function generators, were approximately \$2,500 for a lab with 10 groups and small consumables such as resistors, junction diodes, and soldering wire which were replenished as needed; 2) The per-student, per-course laboratory cost was approximately \$65, including a \$45 Arduino-based Elegoo kit (<https://us.elegoo.com/collections/arduino-kits/products/elegoo-uno-r3-super-starter-kit>) and additional consumables such as capacitors, transistors, and perf boards for soldering (see Table 2). The student course fee of \$50 per semester covered a part of the costs, while the remaining \$15 per student was covered by the Department. Note that the course fee could be increased to \$65 if needed. Note that the supplies provided through the course fee were given to the student, who could then use them for other courses, including their senior design project. All software used in the course was free and open source. This low-cost model enabled scalability to large enrollments without sacrificing hands-on depth.

Table 2: The laboratory setup had two types of costs. (1) Student kits: These cost \$65 per student and stay with the students. These are covered by the Course Fee and are in the table below. (2) Borrowed Equipment and small consumables: Oscilloscope/Function Generator, Voltmeter, Solder, Wire Strippers, and Helping Hand are one-time cost items that stay with the University, while soldering wires, resistors, and junction diodes are replenished as needed. All costs are here: https://github.com/pab47/mechatronics/blob/main/s23/Mechatronics_Equipment.xlsx

Item	Which lab	Qty	Price	Total
ON/OFF switch	Arduino	30	0.95	28.5
TIP31C power diode	BJT	30	0.73	21.9
Photo interuppter	BJT	30	0.35	10.5
10 uF capacitor	Capacitors	30	0.5	15
LM741 op-amp	IC	30	0.9	27
555 timer	IC	30	1	30
Zener diode	Diode	30	0.35	10.5
L298 H-bridge	Sensors/Actuators	30	2.2	66
Barrel jack	soldering	30	0.77	23.1
DC motor	project	30	2.95	88.5
Power adaptor	project	30	8.95	268.5
Perf board	soldering	2	8	16
Elegoo UNO R3 Super Starter Kit		30	45	1350
			Total	1955.5
	Total	30	Cost per	65
	Students		Student	

Personnel support: The lecture materials were taught by the instructor, while the lab and project were handled by the teaching assistant. We recommend one teaching assistant for 10 groups (each group can have between 1 and 4 students).

Lab Reports as Instructables: Students documented their project work as public-facing Instructables (<https://www.instructables.com/>), a website where users can create and share step-by-step instructions for their projects. This format emphasized clear documentation and communication while fostering reproducibility. Their Instructables could potentially be used to support their job applications and/or personal portfolios.

Adaptability to other Engineering Majors: The course structure is readily adaptable to other disciplines, such as biomedical, civil, or chemical engineering, by substituting discipline-specific projects while retaining the core laboratories on sensing, actuation, embedded control, and prototyping.

Hands-on learning, prototyping, and experimentation: A defining feature of the course is its emphasis on rapid prototyping and iterative experimentation, practices that reflect modern engineering workflows. Unlike traditional laboratory courses that rely on expensive, prefabricated equipment with limited flexibility, this course empowered students to design, modify, and test their own systems. This openness not only reduced maintenance costs but also enabled students to create new experiments, fostering creativity, ownership, and deeper conceptual understanding. In addition, the lab component simulates creative troubleshooting, which is argued to be fundamental to authentic engineering learning [14].

Course modification: Although the course was offered in 16-week semester, it can be adapted for a 10–12-week quarter system by reducing or removing select labs, such as soldering (Lab 7),

TinkerCAD (Lab 1, moved to homework), or design and 3D printing (Lab 9). In addition, the Arduino labs (Labs 8, 10, 11) can be combined and streamlined to reduce the overall workload. Thereafter, appropriate reductions can be made to the lecture schedule. The course can also be upgraded to suit the robotics curriculum by adding a camera (an advanced sensor), a Raspberry Pi (a computer), and modifying the project to require an arm on a robotic car for tasks such as pick-and-place.

References:

- [1] Furman, B., Hsu, T., Barez, F., Tesfaye, A., Wang, J., Hsu, P., & Reischl, P. (1996). Laboratory development for mechatronics education. In *Proceedings of the ASEE Annual Conference & Exposition* (pp. 1-291).
- [2] Wild, P., Surgenor, B., & Zak, G. (2002). The mechatronics laboratory experience. *Mechatronics*, 12(2), 207-215.
- [3] Hurst, J., & Mishra, S. (2016). A flipped lab experience for mechatronics education. *Mechanical Engineering*, 138(06), S7-S11.
- [4] Çeven, S., & Albayrak, A. (2020). Design and implementation of modular test equipment for process measurements in mechatronics education. *Computer Applications in Engineering Education*, 28(2), 324-337.
- [5] Mikuska, R., Keczan, L., Katona, K., Szarka, A. V., & Sarvajcz, K. (2025). New laboratory extension and development for mechatronics education. *Measurement: Sensors*, 38, 101320.
- [6] Quanser Inc. (2025). *Quanser educational systems: Control and mechatronics teaching platforms*. Retrieved December 22, 2025, from <https://www.quanser.com/education/>
- [7] Ponce, H., Moya-Albor, E., & Brieva, J. (2024). From project-based learning to innovative technologies in mechatronics course: A case study in a private university in Mexico City. In *Data-driven innovation for intelligent technology: Perspectives and applications in ICT* (pp. 111-134). Cham: Springer Nature Switzerland.
- [8] Ramanathan, K. C., Mohan, M., & Lingampally, P. K. (2025). Project-based learning and its influence on teaching and learning in the first-year core mechatronic engineering course. *International Journal of Mechanical Engineering Education*, 03064190251322065.
- [9] Mohammed, R. B., Albarwary, S. A., & Sabeh, H. N. (2025, February). Project-Based Learning Investigation in Mechatronics Engineering Education at TIU: A Case Study. In *2025 IEEE 22nd International Multi-Conference on Systems, Signals & Devices (SSD)* (pp. 387-393).
- [10] Miyamoto, K., & She, J. (2021, October). Project-based learning using remote-controlled plane for students in mechatronics course. In *2021 IEEE 8th international conference on e-learning in industrial electronics (icelie)* (pp. 1-5). IEEE.
- [11] Hinkle, L. B., Welker, M. W., & Stevens, J. Incorporating Robotics into Electrical Engineering Capstone.
- [12] Peña-Montes De Oca, A. I., & Gallardo-De La Rocha, A. (2024). Supported learning in STEAM and PBL methodology; Using 3D CAD Software for the mechanical design of sumo robot. *Journal of Technical Education/Revista de Educación Técnica*, 8(20).
- [13] Pranav A. Bhounsule, *Mechatronics Spring 2023*, pab47.github.io, June 22, 2023. [Online]. Available: <https://pab47.github.io/mechatronics/spring23.html>. [Accessed: Feb. 23, 2026].
- [14] Dounas-Frazer, D. R., & Lewandowski, H. J. (2016). Nothing works the first time: An expert experimental physics epistemology. *arXiv preprint arXiv:1606.05389*.