2018 ASEE Annual Conference & Exposition
June 24-27, 2018, Salt Palace Convention Center, Salt Lake City, UT
Paper ID #21722

# Design of a Modular Educational Robotics Platform for Multidisciplinary Education

**Zhen Wei**

I am Zhen, I got both my Bachelor and Master degree from Rose-Hulman Institute of Technology in Electrical Engineering. I am focus on Robotics and Embedded System area.

**Dr. Carlotta A. Berry, Rose-Hulman Institute of Technology**

Dr. Carlotta A. Berry is an associate professor in the department of Electrical and Computer Engineering at Rose-Hulman Institute of Technology. She is the director of the multidisciplinary minor in robotics and co-director of the Rose building undergraduate diversity scholarship and professional development program. She has been the President of the Technical Editor Board for the ASEE Computers in Education Journal since 2012. She is a member of ASEE, IEEE, NSBE, and Eta Kappa Nu.

# Design of a Modular Educational Robotics Platform for Multidisciplinary Education


## Abstract

Mobile robotics is inherently a multidisciplinary field due to the interaction of hardware, software, and electronics to create a machine that can sense its environment and then autonomously navigate in the world to achieve some goal or task. Due to its interdisciplinary nature, courses on mobile robotics draw students from various disciplines including computer science, computer, electrical, mechanical, and software engineering. However, teaching mobile robotics to students from multiple disciplines presents some unique challenges. For example, students in such a course may have divergent interests and skillsets. Computer science students may not take a controls course; electrical engineering students may not be familiar with kinematics; mechanical engineering students may not have electronic sensors experience. Therefore, the prerequisite knowledge and skillsets of the students will affect the course topics as well as how they are presented. These challenges also influence what types of assignments are given and how they are assessed. Although it is possible to teach robotics with a simulator, there are some important learning opportunities presented with real world hardware. For example, how to handle sensor error, odometry error, modeling errors, dynamic environments, mismatched motors, memory limitations, knowledge representation, mechanical failure, frame problems, and bandwidth limitations. Popular educational robots such as LEGO® MINDSTORMS® obscure some of these issues, which may not be ideal because there are valuable learning opportunities for students to learn how to resolve or work around these challenges. It would be ideal to have a robot platform with some flexibility such as in the programming language, interface, and programming device in order to

address the needs of diverse populations. It would also be desirable to have some flexibility in the robot controller such as in the number of I/O ports, communication ports, ADC, and DAC because this flexibility will enable the expert user to customize the system to suit their unique needs while also not being overwhelming for the novice user. This flexibility also allows students to use what they are most familiar with to reduce the learning curve and enables them to achieve small robotics successes sooner. This solution will take the focus away from the implementation tool and put it on the robotics educational objective.

This paper will present a solution to the need for an educational robotics platform that is suitable for divergent skill sets. It will describe the design of an economical plug and play robot to suit the needs of a mobile robotics course for students from multiple disciplines. This robot system can be programmed in JAVA, Python, Lua or C. It can also be programmed with various devices such as smartphones, tablets, or the traditional laptop computer. This mobile robotics course currently uses off the shelf or slightly modified off the shelf robots to teach robotics. The initial results will indicate that it is possible to use this modular platform in its various modes to create some of the basic behaviors required for the laboratory assignments.

**Introduction**

This paper will present the design of a modular educational robotics platform to handle the divergent skill sets of a multidisciplinary population in an introductory mobile robotics course. For the last 10 years, juniors, seniors, and graduate students in various engineering disciplines as well as computer science have taken this introductory mobile robotics course. This course has an integral laboratory component and it is necessary to include real hardware in order to meet the learning objectives. The course examines topics related to robotics history, robot components, effectors, actuators, locomotion, sensors, feedback control, control architectures, representation, localization, and navigation. The learning objectives include

- describing the basic components of a mobile robot and the three robot paradigm primitives,
- applying the fundamental principles of programming, mathematics, and science to implement several behaviors on a mobile robot,
- describing the difference between artificial intelligence and engineering approaches to robotics,
- functioning on a multidisciplinary team to complete mobile robotics projects on a hardware platform,
- comparing and contrasting the various robot paradigms including hierarchical, reactive, deliberative, hybrid, and behavior-based,
- analyzing and implementing metric and topological path planning on a mobile robot,
- analyzing and implementing subsumption architecture and potential field summation to implement obstacle avoidance on a mobile robot,
- describing the methods for localization and implementing the Kalman filter algorithm on a mobile robot, and
- describing the methods for mapping and implementing Histogrammic in motion mapping on a mobile robot.

The laboratory assignments are obstacle avoidance, path planning, wall following, light tracking, homing, docking, localization, and mapping. Over this period, the mobile robot platform, programming language and programming interface has been modified or completely changed at least 5 times to meet the needs of this diverse population. The author has determined that it would be ideal to have a robot platform with some flexibility such as in the programming language, interface, and device to address the needs of diverse populations. It would also be desirable to have some flexibility in the robot controller such as in the number of I/O ports, communication ports, ADC, and DAC. This flexibility will enable the expert user to customize the system to suit their unique needs. This paper will present a solution to the need for a robotic hardware platform for divergent skill sets by describing the design of an economical plug and play robot. It is possible to program this robot with various languages such as JAVA, Python, Lua or C. Software on smartphones, tablets, and laptop computers can program the robot. This robot can also use a diverse suite of peripherals, sensors, and motors. It is important to have this flexibility for programming and controlling the robot because these devices are popular, cheaper, simpler to use and easily accessible. They also allow the student to use what they are most familiar with to reduce the learning curve to robot control success for the student. This paper will present the design and implementation of this educational robotics platform as well as results of testing the initial prototype by discussing the creation of basic behaviors on the robot such as wall following and obstacle avoidance by using various languages and devices.

## Literature Review

Many educational institutions use robotics and mechatronics for multidisciplinary and problem based learning [1]-[8]. These activities are reasonable considering robots are ideal for illustrating connections between disciplines as well as for multidisciplinary teamwork, which is one of the ABET criteria for engineering program graduates. This literature review will identify the types of educational robot platforms that institutions use to teach robotics or mechatronics, in particular those with custom-made modular platforms. One of the first discoveries found was that many schools avoid designing and building an educational robot by using off the shelf robots such as LEGO® MINDSTORMS® or Parallax Boe-Bots [9]-[14].

Barrett et al. created a low-cost motivational robot platform, PROFBOT, to teach complex real time embedded system concepts [15]. The controller used RTOS, a computer operating system with a single processor. This controller was capable of responding to multiple events and tasks. It also used scheduling algorithms to manage the high, medium and low priorities within given processing time to execute. There were also interrupts to handle unscheduled events that occurred during the execution of the program. The students worked together to design the robot control system and had to determine the priority of when interrupts would occur individually or simultaneously. The controller used fuzzy logic to determine system output based upon trapezoidal membership functions. The robot had IR sensors, NiCad battery, power supply, power subsystem, and microcontroller. This robot was required to operate autonomously in an unknown maze to detect walls as it moved from the start to point to an exit.

Fang et al. designed a low cost modular robot, NEUrobot, that could be used for research and to teach controls, mechatronics, and robotics [16]. The robot was modular and multi-functional which encouraged creativity and imagination based upon the different types of robots. The students, typically graduate students, built their own control plant for the robot. The authors designed the robot so that the students could learn about all the parts of the whole system, which is particularly important for mechatronics. There was also simulation software in the course to enable students to understand all aspects of the educational platform.

McLurkin et al. design an advanced low cost system of robotics for engineering education [17]. This course involved problem based learning with multi-robots for collaborative teamwork.

The robots were used to teach graduate and undergraduate students and for outreach activities. The Rice r-one robot is a great example of an engineered system with power sensors and abilities. The students completed hands-on homework and design challenges as well as multi-student and multi-robot assignments. Typically, robotics education courses do not include multiple robot applications and have very limited hardware and sensors. The benefits of the multi-robot curriculum are that students can examine applications such as exploration, mapping, search and rescue, surveillance, manipulation, and construction. This platform sought to solve the gaps between theory, simulation and reality for large robot populations. The robot was designed to be low cost, advanced for multi robot sensors communication and localization, and usability for basic operations, programming, charging, and debugging. The robot has differential drive with 2 wheel encoders, light sensors, 8 bump sensors, 3-axis gyro, and 3-axis accelerometer. There is also infrared infrared inter-robot communication. This sensor measures bearing and orientation of the neighboring robots and has a 2-Mb/s radio for centralized command and control. There are also cameras and interfaces with GumStix Linux computing module. The robot can be programmed in C/C++ or embedded Python with embedded debugging through a JTAG programmer. In advanced courses the robot is used to teach topics such as sensing, localization, mapping, motion planning, and state estimation.

In the course described by Meuth et al., students constructed and then programmed a robot kit, LabRat, that was designed by the course instructors [18]. The robot project assignments were programmed by using MATLAB, embedded C programming and LOGO, and open source player stage framework. The course was online with open source lectures, readings, references, robot kit hardware and software. Juniors in CS, CpE, EE, ME and math took the course. The broad goals of the course were to design and build robots, program robots for practical real world applications, and to make robots act as a human. The authors believed that robotics education should start with practice first and then transition into robotics theory. The robot had batteries, motors, 2 bumper whisker sensors, 3 IR proximity sensors, and a channel for inter robot communication. The robot also has an optical mouse sensor on the bottom in order to allow it to acquire position information at a high rate with high accuracy. The assignments were motion control, sensor acquisition, maze solving, path planning, way point navigation, swarm programming, image processing, and multi-robot interaction.

Rahnavard et al. design an educational robot that was used in a principles of electrical engineering course [19]. The course was designed based upon modules or blocks that presented a few engineering ideas to the students at a time. This method will help the students to understand the overall system as well as the divisions in the system and to encourage creative thinking. Robots are ideal to teach system thinking because it involves an integration of several engineering disciplines. Topics in this course included control, actuation, wireless, signal transmission and analog to digital conversion.

Rahnavard's robot was different from other educational robots because it was possible to access all of the components through the low-level integration, which is different from most robots, which have a high-level integration of components and software control. Students would have access to signals on the robot that could be measured with standard equipment. The robot parts included discrete electrical, electronics, and integrated circuit elements. The robot has two analog commands for velocity and direction, six digital commands, and analog data that can be used with an ADC.  There is an LED display for digital data, which can be converted from parallel to serial for PWM modulation. The robot can also receive communication via a serial data on a radio.

Piperidis et al. designed a low cost modular robot appropriate for research as well as for education [20]. The authors state that some of the disadvantages with commercially available robots are inaccurate locomotion, limited motion and energy, limited memory, unreliable sensing and no wireless communication. Although these issues can be resolved with a higher price point, this is not the ideal solution. In addition, these commercially available robots may not be compatible with other off the shelf solutions or parts. One other disadvantage is that it may not be possible to program or customize this platform without advanced software and electronics knowledge. The goal of this work was to minimize the cost of the platform but also not to compromise on the capabilities of the system. The goal is for novice users to be able to seamlessly interact and program the robot. The locomotion design was appropriate for indoor and outdoor environments. The chassis was a Rogue Blue robotics vehicle with an OOPic microcontroller. The OOPic was programmed using an object oriented high level language and had a modular design to afford the use of plug and play devices. The OOPic was used for non-recursive algorithms so it communicated with a personal computer via a serial port to send data in real time. Some of the sensors and devices were infrared, ultrasonic, electronic compass and odometer for each drive wheel. The next versions of the robot after the HELOT were the ALE and ALLE II. The ALE II included a CMU CAM2 for image processing and pattern recognition. It was possible to implement inter-robot communication by using a wireless RS232 serial link to a computer by using Bluetooth. The remote computer used a MATLAB program to serve as the master for the robot slave or slaves. As part of the research, the platform was used to create a fuzzy logic controller that evolved with a genetic algorithm with different fitness functions. The robot was also used in an undergraduate course to perform localization and navigation. One possible upgrade for the low computational onboard power would be to add a small PC.

Sahin et al. [21] designed microrobots to teach undergraduate and graduate students about multidisciplinary engineering projects. These same robots were also used for graduate and undergraduate research, clubs, and organizations. This worked used microbots, which were a swarm of small, inexpensive, autonomous agents. These robots were able to quickly and cheaply cover more ground and were good for reconnaissance, search and rescue, and wildfire detection. The swarm of micro agents were referred to as MEMScouts and these were categorized as SensScouts, GroundScouts, and AirScouts. For example, the GroundScouts had a modular architecture that included locomotion, communication, control and sensors on various layers. Each layer had the same pin connection so that they could be switched out and attached in any order. In general, each robot had seven layers including a base, motor, drives, power, controller, communication, and sensors. The most important characteristic of these robots was the ability to swap the base while keeping the rest of the robot the same such as swapping between wheels, legs, etc.

Shvartsman et al. [22] described a modular mobile robot platform designed for computer science and engineering education. The mobile robot platform included a laptop as the controller, a microcontroller for motion control, vision processing, sensing and navigation modules. This platform affords multiples programming languages such as C++, JAVA, and Visual Basic. The platform also uses standard I/O computer interfaces such as RS232 serial port, USB, networking, audio and parallel port devices. The cost of the platform is kept low by using off the shelf components. These robots were used to introduce adaptable software systems such as artificial neural networks, evolutionary computing, fuzzy, and hybrid software architectures to students. Other topics taught in this course include motion control, electronics, motor system design, embedded real-time programming, and communication protocol between multiple processors. The sensors include a rotating polaroid sonar sensor and 2 Sony IR sensors. The Handy Board microcontroller is used to process the sensor data. The computer vision uses webcam and a JAVA SPI and open CV library to process image data. The robot also has a GPS to navigate an outdoor course.

Tur and Pfeiffer [23] designed a modular low-cost three-wheeled autonomous robotic platform to teach robotics and help students develop communication and teamwork skills. This course used multidisciplinary teamwork with project oriented and collaborative learning activities. The course included students from mechanical, mechatronics, electronics, and communication engineering. Students apply their theoretical framework of knowledge from microprocessors, sensor, actuators, wireless communication, computer vision, artificial intelligence, and electronics to a real world problem. The modular robot platform was designed by the professors and was used for motion control, trajectory planning, teleoperation as well as a testbed for the student teams' software applications of their projects. The platform was designed to have hardware and software modularity, module independence, robustness, precise motion capability, safe operation in human environments, high-energy autonomy and the ability to complete autonomous and teleoperated missions. The differential drive system had two electric dc motors, high-resolution optical encoders, distance sensors, and a distributed control architecture. The robot was comprised of interconnected subsystems that had some level of independence so that the low-level primitives of trajectory planning, obstacle avoidance, and object tracking were not required by the high-level modules. This meant that students did not have to deal with how the behaviors had to compete on the robot platform because those modules were already resident.

## Mobile Robotics Course

The author created the course and has taught Mobile Robotics at Rose-Hulman Institute of Technology for the past eleven years [1], [2], [24]. This course is an upper-level 4-credit hour elective course that is also required for the robotics minor. Juniors, seniors and graduate students in Computer Science, Computer, Electrical, Mechanical, and Software Engineering typically take Mobile Robotics. The pre-requisites for the course are programming proficiency and a controls course. The course has been taught in a traditional classroom as well as in a flipped classroom with 2 hours per week of lecture and 4 hours per week of labs. The course has lectures, literature review, weekly quizzes, labs, and a comprehensive final project and competition. The labs include obstacle avoidance, wall following, Braitenberg Vehicles, homing, docking, and path planning. The final project includes localization, map making, search and rescue.

This course has been taught with a variety of robots including the CEENBot, Traxster II, and Arduino in an attempt to try to meet the needs of all users. However, some of the deficiencies with these robots is the cumbersome, limited programming language, an IDE that is not user friendly or limited I/O to support numerous sensors and peripherals. This limitation also means that students cannot develop their own custom functionality, which is sometimes a benefit on the final project. The course sensors and peripherals typically includes LEDs, LCD, pushbuttons, sonar, IR, photoresistor, temperature sensor, motor, encoders and a wireless module.

## Method

The modular platform was designed using the framework of a central controller and various modules for sensors and peripherals. Since most microcontrollers will have limited IO ports, the central controller for the robot platform was designed by using an SPI bus communication technology. In this framework, the central controller is the "master" and the modules are the "slave". The central controller will coordinate turning the modules on and off to insure appropriate bidirectional communication. The controller had a 32-bit embedded Linux system, which allows it to be programmed using various languages. This also allows users to remotely login and control or program the robot through the SSH protocol. This remote access affords the use of different devices for programming such as Android phones or tables, Apple devices, Window PCs or Chromebooks. The central controller was designed based upon the Raspberry Pi 2, which is a 32-bit embedded Linux platform. All of the sensor and peripheral modules were designed based upon the MSP430G2553 microcontroller. The baseboard was installed in the bottom of the robot base frame and includes the power regulators and modules.

*Hardware*

Due to its simplicity and popularity, wheeled differential drive was selected for the educational mobile robot platform. The robot had two front wheels driven independently on a common axis with one caster wheel in the back for stability. This platform was designed on top of an old Traxster robot chassis. It was decided to convert from DC motors to stepper motors because DC motors without encoders were less accurate. The robot also had four infrared sensors mounted around the periphery.

*Modules*

As a proof of concept, the platform was built with three modules including motor control, IR sensor, and custom. However, as is stated in the premise for this paper this can be easily expanded with the creation of additional modules. In fact, the custom module is designed so that users can create their own circuits and add them to the platform. This module allows users to access all of the free pins on the MSP430G2553. The motor control module is used to send data to the two Adafruit 4-wire bipolar stepper motors. The IR sensor module is used to acquire data from the four Sharp 0A41SK0F infrared sensors.

*Power*

The power supply is built with a power regulator, an on/off switch and a 12V Lithium-Ion battery. The power regulator supports 5V and 3.3V to handle the various peripherals and sensors. The power system also includes a micro USB connector used to power the Raspberry Pi.

*Firmware*

Each module was designed to have its own firmware to control the module and communicate with the central controller. Each of the modules is an SPI slave with a four-wire SPI bus to select when the communication channel is open. The firmware uses the SPI independent interrupt capability to wake up the module from the lower power mode when the robot gets or send a message from the central controller. This method of control enables the robot to save power.

**Results**

In order to confirm that the modular platform is a viable option to teach the mobile robotics course, several demonstration applications were created. The three demo projects created were: basic control, obstacle avoidance, and wall following. The basic control performs a simple test on each of the robot modules (motors, sensors). The obstacle avoidance demo is a basic behavior where the robot used to illustrate that the robot can use IR sensor feedback to create a basic behavior where the robot avoids objects in the environment as described in Figure 1. Wall following is very similar to obstacle avoidance except the robot detects objects with the IR sensor but uses an algorithm to move tangential to the object. These are the basic AI algorithms required for any mobile robot that can then be built upon to create behaviors such as path planning, localization, mapping, line following, homing, docking, light sensing, and object tracking.

*Basic Control Demo*

In this demo, a computer is used to control the robot platform by sending various commands. The main program gets commands as input through the command line to call the functions in the stepper motor and IR sensor API. A summary of the basic control commands are shown in Table 1.

Table 1: Basic Control Demo Commands

| Commands | Description |
|---|---|
| ./robot go <speed right><direction right><speed left><direction left> | This command will move the robot wheel's at a given speed where direction is 1 or 0 for forward or reverse. |
| ./robot go | This command uses default settings to move the robot forward at 50 rpm. |
| ./robot stop | This command stops the robot motion. |
| ./robot get | This command will retrieve the total steps travelled after the robot stops motion. |
| ./robot cget | This command will retrieve the total steps while the robot is running. |
| ./robot irget | This command will retrieve the data from all four IR sensors. |
| ./robot irget<sensor name> | This command will retrieve the sensor data from one of the four IR sensors. |

*Obstacle Avoidance Demo*

There are three parts of the demonstration of the obstacle avoidance behavior: collide, feel force, and avoid obstacle using potential fields. These are based upon a subsumption architecture where the task-achieving layers can be inhibited or subsumed at the input or output to make the overall behavior. The input is the IR sensor data and the output is the motor commands based upon the robot primitives, sense-plan-act. Figure 1 shows an example of the obstacle avoidance behavior implemented for the demo. In the collide behavior the robot moves forward or backward and stops or halts if an obstacle is detected. If the object is removed, the robot will continue to move forward or in reverse. The avoid behavior, the robot is stationary and moves based upon the force field detected from the four IR sensors. In the complete obstacle avoidance behavior, these two behaviors are integrated so that the robot may move randomly and as sensors are triggered, the robot uses a potential field to move away from or around the obstacle.

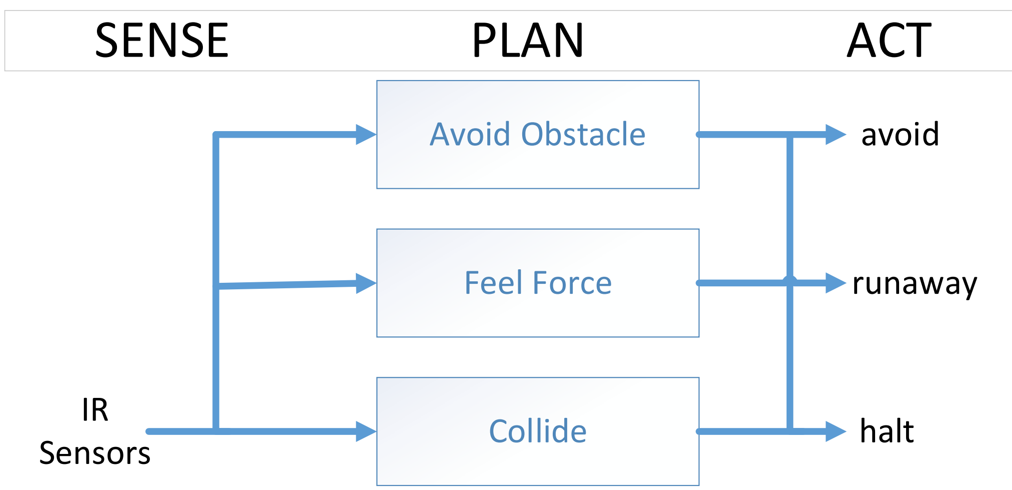


Figure 1: Obstacle Avoidance Subsumption Architecture

*Wall Following Demo*

The wall following demo uses feedback control with a P, PI, or PID controller to determine how to maintain a given distance from a wall and how to handle when the robot encounters a door or corners. If the robot detects an inner corner which means there is a wall in front it will turn toward the opposite wall and continues following the wall. If the robot detects an outer corner, which could be a wall or a door exit, the robot should turn one direction until another wall is detected or a door and move forward to go through the door or continue falling the wall.

The demo videos can be found in the following YouTube playlist: bit.ly/ModEdMobot.

*Comparison*

The final part of the results compared the final hardware platform to some of the other robots used in the course. One of the key advantages of this platform is that the user can add any type of hardware including GPIO, ADC, DAC, SPI, I2C, UART in order to design their own custom module. The robot has the potential to have 20 extra custom modules based upon the current design of the SPI bus. There are some pre-existing features of current robots that would have to be customized on the new platform including an LCD, speaker, pushbuttons, compass, and line following sensors. One other key difference is that there is no IDE and the programming is typically done in text editors. In the current iteration, the user has to compile the program and write their own makefile but this would be a necessary improvement with the creation of an IDE. In order to create a custom module, the users will have to understand embedded firmware and how to program a microcontroller. In general, the robot hardware platform is more powerful but may not be as user-friendly for the novice user.

*Course Deployment*

This platform has not been deployed in the course yet because it must be mass-produced for 40 students and 20 teams. In order for this to happen, there is some necessary future work, that will be described in the following section. However, the assignments with learning objectives and grading rubric will be similar to those currently used in the course and they are available at the following links: https://bit.ly/2v5HbV3 and https://bit.ly/2JxsBsu.

## Conclusions and Future Work

This paper has presented the design and implementation of a modular educational robotics platform. The goal was to design a platform that was conducive for multidisciplinary education involving students with various interests and prerequisite skills. This hardware will solve the challenges associated with traditional educational robots including limited pins, single programming language, and single programming devices. There were also three demonstrations created to exploit the current capabilities of the platform including obstacle avoidance and wall following. The plug and play feature of the design makes it easy to customize the robot and also to keep the interest of the more advanced student. This system was tested by an undergraduate student volunteer who stated that the robot was “powerful and simple to use”.

Future work includes the creation of an IDE which can support users with different level of experience with programming and embedded systems. There can also be several levels of IDEs from graphical such as for LEGO Mindstorms, LabVIEW , Arduino, Visual C++, Eclipse, or Linux. Another improvement would be the addition of a battery monitoring system which detects when the power is low and either turns the robot off or moves it to a charging station. There should also be the addition of more sensors and modules to make the robot more powerful and provide the ability to create more AI control algorithms. After these changes are made, the robot must be mass produced in order to be used for 40 students (including 20 teams) in the classroom.

**Acknowledgment**

The authors would like to thank the ECE department at Rose-Hulman Institute of Technology for providing resources supporting this work.

**References**

[1] Berry, Carlotta A. "Mobile robotics for multidisciplinary study." Synthesis Lectures on Control and Mechatronics. Morgan & Claypool, 3.1 (2012): 1-95.

[2] Berry, Carlotta A. "Robotics Education Online: Flipping a Traditional Mobile Robotics Classroom", Proceedings of ASEE/IEEE Frontiers in Education Conference, Indianapolis, IN, October 18-21, 2017.

[3] Hassan, Houcine, et al. "A Multidisciplinary PBL Robot Control Project in Automation and Electronic Engineering." IEEE Transactions on Education 58.3 (2015): 167-172.

[4] Heller, Mahlon D. "Mobile Robots and Interdisciplinary Design-Mobots." Proceedings for the 1999 Annual Conference and Exposition, 1999.

[5] Pack, Daniel J., and Alan R. Klayton. "Engineering Education Through Robotics at the United States Air Force Academy." Intelligent Automation & Soft Computing 13.1 (2007): 45-56.

[6] Wikander, Jan, Martin Torngren, and Mats Hanson. "The science and education of mechatronics engineering." IEEE Robotics & Automation Magazine 8.2 (2001): 20-26.

[7] Wiedenman, Nathan, and Barry Shoop. "A Collaborative and Interdisciplinary Approach to Mechatronics." age 10 (2005): 1.

[8] Hwang, Deborah J., and Dick K. Blandford. "A multidisciplinary team project for electrical engineering computer engineering and computer science majors." age 5 (2000): 1.

[9] Maxwell, Bruce A., and Lisa A. Meeden. "Integrating robotics research with undergraduate education." IEEE Intelligent systems and their applications 15.6 (2000): 22-27.

[10] Jung, Seul. "Experiences in developing an experimental robotics course program for undergraduate education." IEEE Transactions on Education 56.1 (2013): 129-136.

[11] Spencer, D., and N. Jaksic. "A multidisciplinary robotics learning environment: what Mindstorms and DARPA urban challenge have in common.", *ASEE Computers in Education Journal 1.3* (2010): 32-39.

[12] Weinberg, Jerry B., et al. "A multidisciplinary model for using robotics in engineering education." *Proceedings of the 2001 American Society of Engineering Education Annual Conference and Exposition,* Albuquerque, New Mexico, June 24-27, 2001.

[13] Weinberg, Jerry B., and Xudong Yu. "Robotics in education: Low-cost platforms for teaching integrated systems." IEEE Robotics & Automation Magazine 10.2 (2003): 4-6.

[14] White, W., et al. "Assessing an interdisciplinary robotics course." *Proceedings of the 2005 American Society of Engineering Education Annual Conference and Exposition,* Portland, OR., June 12-15, 2005.

[15] Barrett, S. F., et al. "Using Robots to Teach Complex Real Time Embedded Systems Concepts." Computers in Education Journal 16.4 (2006): 58-58.

[16] Fang, Zheng, Yanhua Fu, and Tianyou Chai. "A low-cost modular robot for research and education of control systems, mechatronics and robotics." Industrial Electronics and Applications, 2009. ICIEA 2009. 4th IEEE Conference on. IEEE, 2009.

[17] McLurkin, James, et al. "Using multi-robot systems for engineering education: Teaching and outreach with large numbers of an advanced, low-cost robot." IEEE Transactions on Education 56.1 (2013): 24-33.

[18] Meuth, Ryan, Paul Robinette, and Donald Wunsch. "Introducing Robots." Proceedings of the 2009 American Society of Engineering Education Annual Conference and Exposition, Austin, TX. 2009.

[19] Rahnavard, M., et al. "Educational Robot for Principles of Electrical Engineering."

[20] Piperidis, S., et al. "A low cost modular robot vehicle design for research and education." Control & Automation, 2007. MED'07. Mediterranean Conference on. IEEE, 2007.

[21] Sahin, Ferat, and Wayne Walter. "Multidisciplinary microrobotics teaching activities in engineering education.", *Proceedings of the 2003 American Society of Engineering Education Annual Conference and Exposition,* Nashville, TN, June 22-25, 2003.

[22] Shvartsman, Andrey, Maurice Tedder, and Chan-Jin Chung. "A Modular Mobile Robotic Platform As An Educational Tool In Computer Science And Engineering." Proc. Int. Conf. Comp., Commun., Control Technol.(CCCT). Vol. 5.

[23] Tur, JM Mirats, and Carlos F. Pfeiffer. "Mobile robot design in education." IEEE Robotics & Automation Magazine 13.1 (2006): 69-75.

[24] Berry, C.A., “Mobile Robotics: A tool for application-based integration of multidisciplinary undergraduate concepts and research”, Computers in Education Journal, Vol. 1, No. 3, pp. 102-111, 2010.