AC 2010-2314: APPLICATION OF RAPID PROTOTYPING FOR DESIGN OF A WALKING ROBOT

Richard Chiou, Drexel University

Eric Carr, Drexel University

Robin Kizirian , Drexel University

Yueh-Ting Yang , Drexel University

Brittany Killen, Drexel University

Yongjin Kwon, Ajou University

Application of Rapid Prototyping for Design of a Walking Robot

Abstract

The desired set of skills required of modern engineers and technologists has been steadily expanding. In addition to familiarity with robotics and CAD/CAM techniques, rapid prototyping is increasingly becoming an essential tool in the design and manufacturing of complex systems. In this paper, the application of rapid prototyping in fabricating a walking robotic system and mechanism is presented. Using a Dimension uPrint Personal 3D Printer, prototypes of a robot body and legs are fabricated. These components are then used to fabricate the articulated structure of an experimental prototype for a quadruped robot. The necessary information about methods of control, power, sensors, batteries, electronics, and more is presented. Materials, methods, and tools are outlined, including the use of servomotors and microcontroller-based control systems. Students in the Applied Engineering Technology program are required to work with this robotic project as part of a laboratory experiment in the "MET 205 Robotics and Mechatronics" class. Providing students with such a hands-on approach enables them to improve their robotic skills by using rapid prototyping and microcontrollers for performing different robotic applications.

Background

In Drexel University's School of Technology and Professional Studies, many courses related to robotics, design, and materials are offered to the students in the Bachelor of Science in Applied Engineering Technology program. Courses such as Robotics and Mechatronics, Quality Control, Manufacturing Materials, Microcontrollers, and Applied Mechanics can benefit from the laboratory experience in applications of mechatronics, robotics, and rapid prototyping. As well as helping in the teaching of various courses, such experience benefits students who are pursuing degrees in the engineering field. Students in the Mechanical, Electrical, and Industrial fields along with many others can learn many new skills from multi-disciplinary projects such as the rapid prototype design of a walking robot. Such projects show students how to use different types of technology, and demonstrate how advanced technology can be used in an actual application. This project teaches future engineers and technologists various advanced skills that can be used in their careers. Overall, many different fields of engineering can benefit from this application, enabling the development of skill and knowledge in many different engineering aspects and processes.

Students in the Applied Engineering Technology programs are required to complete a capstone course MET 205 Robotics and Mechatronics that increases the student's mechatronics design-for-manufacturability knowledge. A novel rapid product manufacturing technique that reduces cost and compresses time for delivery of products produced in batches. The technique is composed of three stages – digital prototype, physical prototype and rapid tooling robotic system. The technique effectively integrates the contemporary advanced manufacturing technologies such as solid modeling, CAE, rapid prototyping, and rapid tooling in improving the product design, analysis, prototype, and production process.

This application allows students to develop their knowledge of engineering and familiarity with all the technical components that go into it based on the fabrication of this robot. It can benefit students in fields such as Mechanical, Electrical, and Industrial Engineering. Providing students with a hands-on approach when teaching robotics classes in the ET curriculum enables students to become aware of how their mechatronics design, rapid prototyping, and computer control can drastically influence the downstream design and manufacturing processes. This is especially helpful for students in the mechanical, electrical, and industrial concentrations as they have a high probability of designing parts that will require machining processes during their manufacture.

This project takes a simple idea of designing a robot and and improves its technology so that it can not only be run semi-autonomously, but also can walk and balance itself. Starting with the basic components of manufacturing materials, prototyping, programming, and building basic systems, this project takes engineering to the next level. It incorporates these components, but uses more complex systems in the final product. Instead of "hand-making" materials, students use CAD software (SolidWorks, ProE, Google SketchUp, etc…) to design the materials[1]. Then, an uPrint 3D Printer is able to print out these materials based on what is designed in the CAD software. This approach works well, showing students how software can be used to create the different components of the final product.

Moreover, students have to learn electronic applications such as circuit design and breadboarding along with the development of microcontroller firmware in order to be able to prototype the robot[2-5]. For example, microcode (written in PIC assembly in this case) to control the robot's servomotors, is necessary to accomplish the dynamic movement. The legs give the robot freedom to walk and maneuver more accurately in comparison to a more traditional wheeled robot. Also, students have to learn how to program the microcontrollers in order to make the robot run semi-autonomously. All of these components incorporate new technology into the educational process in order to teach the students how complex tasks can be accomplished through the use of technology such as rapid prototyping and microcontroller programming [6-9].

Teaching Engineering using Robotics

In order to be able to successfully complete a project such as this, one must have a basic background in Robotics and Mechatronics. Creating prototypes is a complex and advanced idea and thus requires the designer to establish some basic skills prior to developing a project. First, basic techniques used in assembling mechanical parts should be known. This will help in the assembly of the robot. Additionally, one must know how to program the microcontrollers. They are programmed in PIC assembly using Microchip's MPLAB IDE v8.40[10]. The PIC microcontroller firmware allows the user to control and program the functionality of the different servomotors. This is a good topic for students to understand – how servomotors operate and what their mechanical and electrical requirements are. Last of all, the manufacturing of materials is a skill that can come in handy for this project. Students will learn how to design materials and assemble them, making the prototyping process a lot smoother. The skills need to develop this prototype can be learned utilizing a few different components. In this paper, different components are described to help students with the design process. First, when utilizing tools such as Google SketchUp, they students create a CAD model, which is then fabricated on the 3D printer. The students then assemble a circuit board, based on schematics provided to the students. Basic knowledge of engineering and engineering design can benefit students in the application of prototyping this walking robot[11-13]. Also, many of these basic skills can be learned in engineering classes. From these classes the students should be able to utilize their skills and knowledge to complete this project.

Based on the use of Google SketchUp, MPLAB IDE, the uPrint 3D Printer, along with many different types of components and materials, it is clear that the use of Rapid Prototyping in Engineering Education can be very beneficial. Google SketchUp provides users with an easy way to design and manufacture components that can be used for this (or nearly any other) application. It provides the students with an easy way of designing parts with little knowledge of the use of CAD based software. Additionally, the uPrint 3D printer provides an efficient way to manufacture these pieces from the designed CAD file.

With little knowledge of how to build parts, students are nevertheless able to make these parts using this method. Also, the MPLAB IDE allows the user an easy way to program the microchips which control the walking robot. This will allow for the functioning of the walking robot itself based on semi-autonomous commands. This application is an easy way to educate students on how to build robots from scratch. Even with just basic knowledge of the many different parts that go into it, students are able to easily assemble a functional walking robot.

Walking Robot Design

The robot itself, even though might seem very complex, is actually taking the simple concept of robotics and incorporating multiple technology into it. This application can benefit many people who are in or who are going into an engineering field. It provides a way for students to get a "hands on" perspective of robotics. Not only does it teach one how to build a robot, but it shows them how many different components go into the assembling of a system such as this one. It gives students hands-on experience with a futuristic type of technology, taking a simple robotic prototype, manufacturing it using an advanced 3D printer, and designing firmware to make it walk.

As shown in Figure 1, the main chassis holds the circuit board, the battery case, and the four "shoulder" servomotors. Each of these servomotors is connected to a leg bracket. The four "knee" servomotors are inserted in the upper part of the legs, one to a leg. These servomotors connect to the leg brackets, providing the robot with two-degrees-of-freedom articulation of each leg. By varying the duty cycle of the timing pulses (therefore the angle of the servos), each leg can be moved independently to produce a variety of gaits. Two views of the chassis, along with those legs and shoulder brackets, are shown in Figure 2. The uPrint Personal 3D Printer as shown in Figure 3 is used to print out the major mechanical pieces used in the prototype.

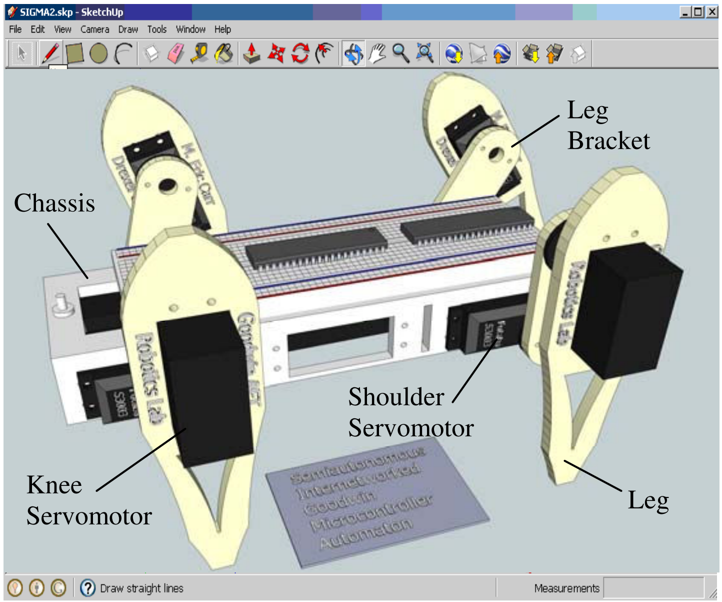
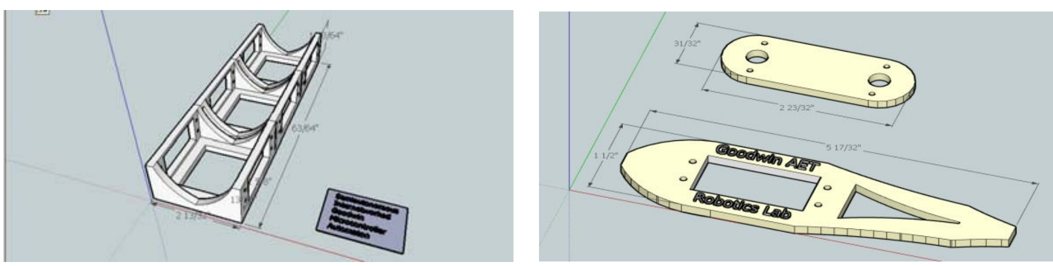


Figure 1: The walking robot designed in Google SketchUp.



(a) (b)

Figure 2: (a) The chassis and (b) the shoulder and leg in Google SketchUp.

Figure 3: The uPrint 3D Rapid Prototyping machine used for this project.

In the current design, the chassis is used as the base to which the other elements of the robot are attached, either directly (for example, the circuit board, shoulder servos, battery pack, and etc.) or indirectly (for example, the knee servos and legs). The breadboard, circuit board, and the microcontrollers themselves are mounted on the upper side of the chassis. The circuit board and breadboard are powered by the “AAA” battery pack, which is attached to the underside of the chassis.

Robotic Control System Design

The non-expensive robotic control system has been designed for instruction purpose in MET 205 Robotics and Mechatronics, which contains a variety of pieces such as rapid prototyped lags, servomotors, and sensors to build and control vehicles, robots, and systems. It contains a programmable PIC16F887-I/P microcontroller that can be programmed using PIC Assembly language. The detailed cost of the parts for the robotics system is shown in Table 1.

Table 1. Detailed cost of the parts for the robotics system

Part description	**Manufacturer**	**Unit cost**	**# required**	**Total cost**
Solderless breadboard	(Generic)	$4.95	1	$1.00
PIC16F887-I/P microcontroller	Microchip	$2.20	2	$4.40
1/8-watt 1kOhm resistors	(Generic)	$0.09	2	$0.18
S3003 servo	Futaba	$9.99	8	$79.92
Power switch	(Generic)	$5.28	1	$5.28
4xAAA battery holder	Radio Shack	$1.91	1	$1.91
4xAAA batteries	(Generic)	$2.75	4	$11.00
Solid hookup wire	(Generic)	$3.95	2	$7.90
PICKit2 programmer/debugger	Microchip	$34.99	1	$34.99
Robot chassis	(prototyped)		1	
Robot upper leg bracket	(prototyped)		4	
Robot leg	(prototyped)		4	
Total				**$ 146.58**

The eight servomotors are controlled by a PIC16F887 microcontroller, allowing the robot to perform complex movements. Several canned routines (walk forwards, walk backwards, turn right, turn left, spin right, spin left, stand-low, stand-mid, stand-high, sit/stay, lie down, and stop) are implemented by the on-board microcontroller. Seven-bit command words are passed to the servo microcontroller via a dedicated 8-bit bus (the 8th bit is used for synchronization). At this stage, the sequence of movements is choreographed by a second microcontroller on a solderless breadboard, although the control-bus design is intended to accommodate wireless control and/or an onboard control computer, allowing semiautonomous or possibly fully autonomous operation of the robot. The robot is programmed at runtime to simply follow a pre-planned sequence of commands.

The programming for both the servo and command microcontrollers is developed in PIC assembly (for a PIC16F887 microcontroller). Microchip MPLAB IDE v8.40 is used to develop and debug the firmware as well as to download the program to robot. The microcontroller on the breadboard is the (temporary) command controller. It issues a predetermined sequence commands that are sent to the servo microcontroller to tell it what tasks to perform. The pre-programmed command sequence is easily changed using MPLAB to recompile and download modified code onto the command microcontroller. Simple commands are issued to the servo microcontroller; implementation at the servo signal level is not handled on the controller. This hierarchy allows easy and efficient changes in the implementation of the command scheme without requiring a redesign of the entire servo firmware package.

The servos are controlled by PWM (Pulse Width Modulated) signals. One control line is used for each of the eight servos. The slave microcontroller produces a 50Hz sequence of pulses, the duty cycle of which controls the commanded angle of each servo. For example, a 600us pulse could command the servo to move to its full counter-clockwise position; a 1320us pulse could command it to the center, or neutral, position, and a 2040us pulse could command it to move to the full-clockwise position. By varying the time that each control line is driven high by the microcontroller, the servo angles, and therefore the robot gait, can be controlled.

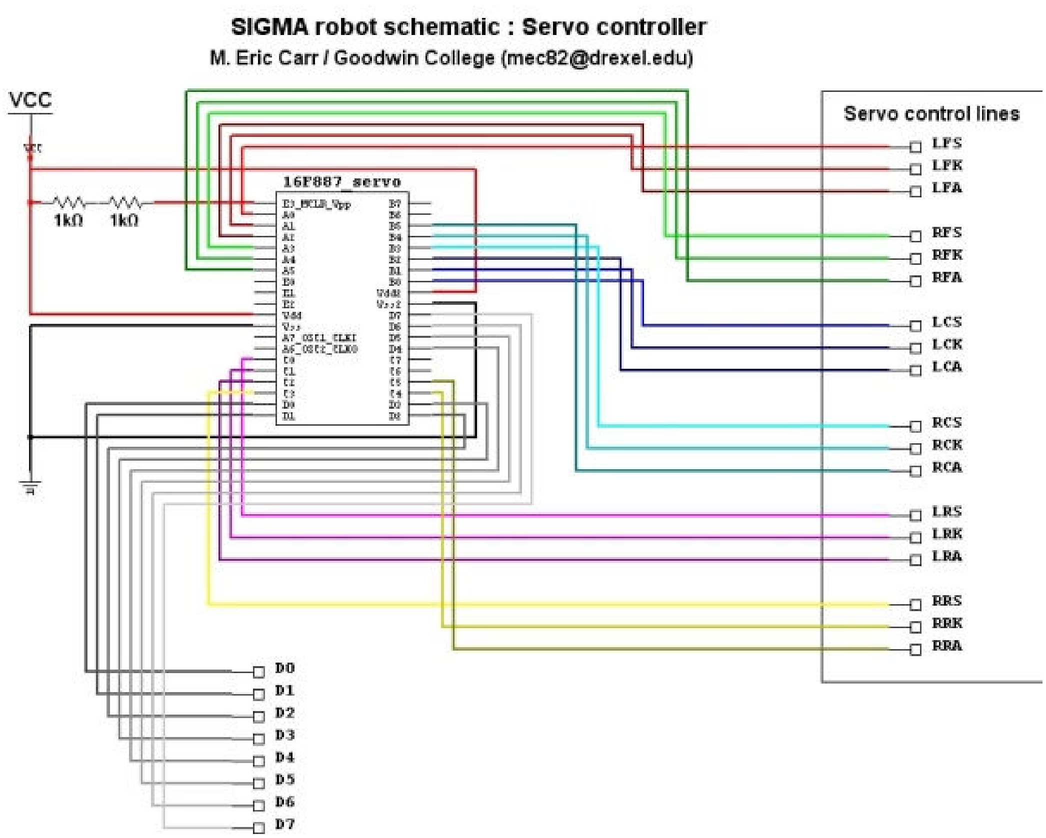


Figure 4: The schematic for the servo (slave) microcontroller.

The onboard servo (slave) microcontroller located on the circuit board directly controls the eight servomotors on the robot. It interprets the commands issued by the command microcontroller, and produces the proper sequence of timed pulses necessary to move the eight servos to the desired positions. Using MPLAB greatly simplifies the student's task of programming the robot. The microcontroller on the breadboard produces a sequence of timed 7-bit command words (such as walk-forward, stop, stand up, walk backwards, sit, etc). The microcontroller on the circuit board receives these commands and implements them as canned timing signal sequences to be sent to the eight servomotors – one shoulder servo and one knee servo per leg. The shoulder servos sit inside and are attached directly to the underside of the chassis. The knee servos fit inside each leg of the robot. For the time being, rubber bands are used to secure the solderless breadboard, circuit board, and battery pack to the chassis. In general, the design can be prototyped as seen in Figure 5.

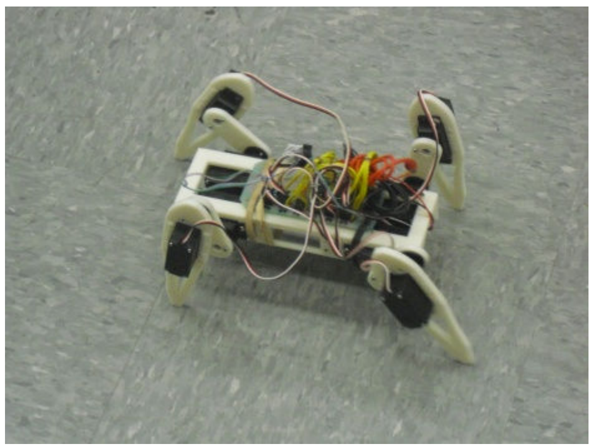


Figure 5: The walking robot in action

As currently implemented, the robot is a two-degree-of-freedom quadruped (using eight servos) with the circuit itself upgradeable to hexapod configuration with three degrees of freedom (which would use a total of eighteen servos) if the MCU speed is increased to 20MHz or more. Our robotic laboratory experiment emphasizes those aspects of microcontroller-based control systems and robotics that are most closely related to computer science. These aspects include the following:

- Architectures and instruction sets of microcontrollers
- Interfacing a microcontroller with memory and I/O
- I/O techniques (serial, parallel, interrupt-driven, digital to analog conversion, analog to digital conversion)
- Microcontroller programming languages and techniques
- The use of timers in responding to and controlling the gait system
- Walking efficiency of the robot based

The current walking gait is hand coded and is diagrammed in Microsoft Excel as an X-Y plot. An initial estimate at a footfall sequence and foot path is made by slow-motion video observation of a walking draft horse. Specifically, the gait sequence used is: LR → LF → RR → RF etc, with a 90 degree phase angle between each. Unlike a horse, however, the SIGMA robot uses reverse symmetry: the front legs of the robot “pull” while the rear legs “push.” (This allows closer spacing of the onboard servos, allowing a smaller physical size for the chassis.) The footfall sequence and the foot paths are reversed for the backwards gait. (Essentially, the backwards gait is the same as the forwards gait reversed in time.) The “spinning-in-place” and turning gaits are loosely based on the walking gait. The gate sequence timing and walking efficiency are measured in the laboratory experiment as shown in Figure 6. The students in MET 205 Robotics and Mechatronics, working in groups, collected data for gait cycles of the walking robot. Data are then entered into Microsoft Excel for analysis.

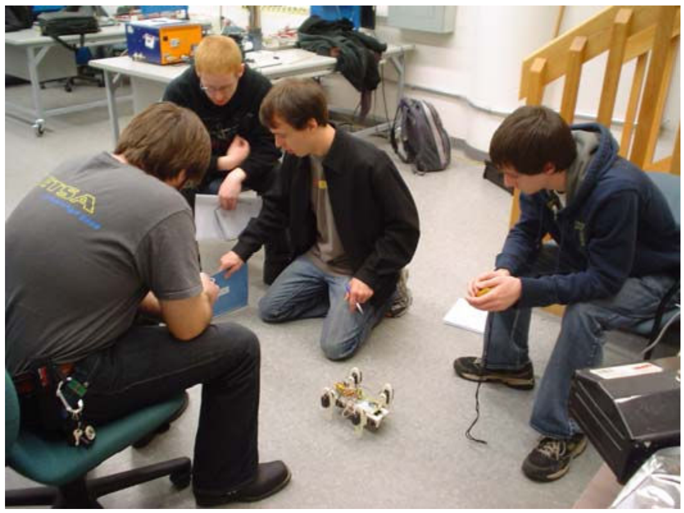


Figure 6: Students measuring walking gait and efficiency of the walking robot in MET 205 Robotics and Mechatronics.

Gait calculations and experimental results

As shown in Figure 7, various calculations are necessary in order to come up with a suitable walking gait for the robot. A rough sketch of the desired foot movement pattern is designed, along with a rough idea of the desired foot velocity at each point in the cycle. This sketch is translated into a positional X/Y graph in Excel, in order to determine the corresponding horizontal and vertical positions for each point in Figure 7. To perform an exhaustive search of possible shoulder and knew servomotor angles, a program is written in FreeBasic to find the best fit for each point in the cycle. This program simply runs through the 180x180 possible combinations of knee and shoulder joint angles for each position, and selects the combination which produces the best fit to the desired leg position for that phase.

The points in Figure 7 are equidistant in time, showing much faster movement at the top, where the foot is drawn forward in preparation for the next step. Footfall occurs at position ~(330,0). Distances are in millimeters, measured from the rearmost extent of foot travel.

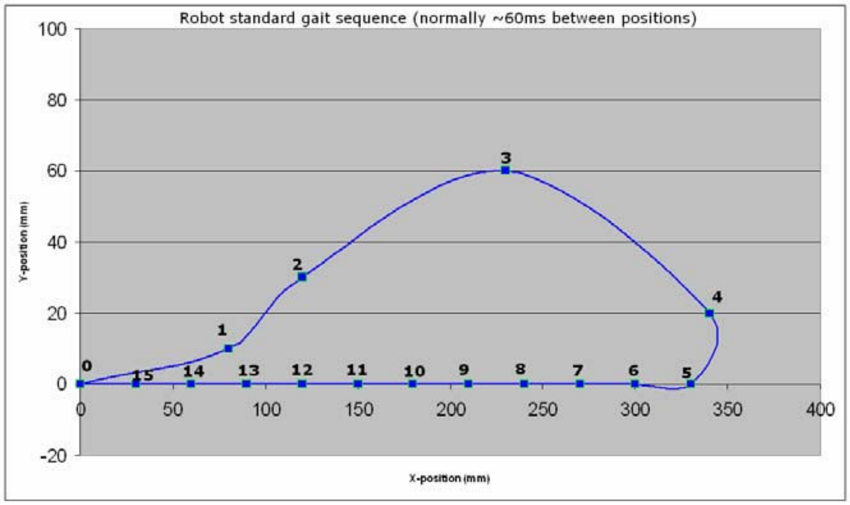


Figure 7. An X/Y positional plot of foot position during one cycle of the walking gait.

As the emphasis at this phase of the project is on the development of the walking robot application as a module suitable for inclusion in an Engineering or Engineering Technology program, data has not yet been gathered on robot performance. However, a qualitative observation does indeed provide enough proof that the robot is capable of semiautonomous forward, backward, and turning motions. Judging from student interest and participation in the project thus far, we can definitely conclude that this is a project which not only provides a suitable vehicle for teaching many Engineering-related concepts, but one which also captures students' interest and can foster student enthusiasm for Engineering as a career.

Future work

Several additions and refinements are planned for this project. First, although the current handcoded walking gait works reasonably well, we are confident that a Genetic Algorithm-based approach, using a physics simulation of the robot and a GA to evolve a suitable gait, will yield a more efficient, smoother, and more "natural" gait, even with the current two-degree-of-freedom design. Along with such software improvements, an upgraded power solution incorporating a Li-Ion or NiMH battery pack is planned, allowing for both greater runtime as well as increased peak current capacity (allowing greater torque from the servos, and therefore a wider range of feasible gaits). Eventually, we plan to investigate the possibility of upgrading the robot design to incorporate three degrees of freedom per leg. This will allow much more natural, versatile (and hopefully more efficient) gaits. The circuit board on the robot is actually capable of controlling up to eighteen servos as it is. Adding these servos would simply be a matter of redesigning the shoulder bracket component to include the third servo, as well as modifying the onboard servo microcontroller firmware to drive the third set of servos. Improvements and additions to the control side of the project are also planned, including implementation of wireless control; a degree of autonomy via the inclusion of onboard sensors as well as various firmware upgrades.

Conclusion

The application of rapid prototyping of a walking robot has excellent potential as an educational project in Engineering and Engineering Technology programs. Through the use of Google SketchUp 7.1, Microchip MPLAB IDE v8.40, and a uPrint Personal 3D Printer, students are able to accurately design and prototype a walking robot. The walking robot can be used for many different types of applications. With this robot, students can learn many interdisciplinary skills in robotics and mechatronics, microcontrollers, servomotor control, gait geometry, remote control/ telepresence, surveillance, and inspection of inaccessible areas. The walking robot itself is a simple design that allows a rapid prototype model of it to be easily made. It is certainly a useful device that can be used for many different applications. Additionally, in the future, a wireless system is planned as an upgrade to the walking robot. This system will allow the user to control and program the robot via a wireless connection. In addition, the course MET 205 Robotics and Mechatronics has successfully provided the students with skills in rapid prototyping and microcontrollers for robotic control applications.

Acknowledgement

This work is supported by the National Science Foundation (Award # DUE-0410719 and DUE-0618665, CCLI Phase I and II Projects). The authors wish to express sincere gratitude for the Foundation's financial support of this project.

Bibliography

1. Google Sketchup Software: http://sketchup.google.com/
2. Bhömer, M. t., Bartneck, C., Hu, J., Ahn, R., Tuyls, K., Delbressine, F., et al., "Developing Novel Extensions to Support," Prototyping for Interactive Social Robots, Proceedings of the 21st Benelux Conference on Artificial Intelligence (BNAIC 2009), Eindhoven, pp. 11-17, 2009.
3. C. Bartneck and J. Hu. "Rapid prototyping for interactive robots," the 8th Conference on Intelligent Autonomous Systems, pp 136-145. 2004.
4. Asiabanpour, B., Wasik, F., Cano, R., Jayapal, V., VanWagner, L., McCormick, T., "New waste-saving heater design for the SIS Rapid Prototyping Process," Industrial Engineering Research Conference (IERC), Atlanta, GA, 2005.
5. Chua Chee Kai, Leong Kah Fai, Lim Chu-Sing, "Rapid Prototyping: Principles and Applications," World Scientific Publishing Company, Singapore, 2003.
6. Laliberté T., Gosselin C. and Côté G., 2000, "Rapid Prototyping of Lower-Pair, Geared-Pair and Cam Mechanisms," Proceedings of the ASME Mechanisms and Robotics Conference, Baltimore MD, September 10-13, 2000, Paper DETC2000/MECH-14202.
7. Won J., DeLaurentis K. and Mavroidis C., "Rapid Prototyping of Robotic Systems," Proceedings of the 2000 IEEE International Conference on Robotics and Automation,
8. "Bailey, S. A., Cham J., G., Cutkosky, M. R., Full, R. J., "Biomimetic Robotic Mechanisms via Shape Deposition Manufacturing," Robotics Research: the Ninth International Symposium, John Hollerbach and Dan Koditschek (Eds), Springer-Verlag, London, 2000.

9. Goel, P., Rajagopalan, S., and Cutkosky, M.R., "Error Analysis for the In-Situ Fabrication of Mechanisms," Proceedings of the ASME DETC/DFM Conference, Baltimore, MD, Sept. 13-16, 2000, Paper DETC2000/DFM-14019.
10. http://www.microchip.com
11. Howell, A., Laramee, C., McGrann, R., and Way, E. "Autonomous Robots as a Generic Teaching Tool," Proceedings of 36th ASEE/IEEE Frontiers in Education Conference (FIE)*, San Diego, CA,* 28-31 October 2006.
12. C. Bartneck and J. Forlizzi, "A design-centred framework for social human-robot interaction," 13th IEEE International Workshop on Robot and Human Interactive Communication, pp. 591-594, Roman, 2004.
13. Peatman, J. B., Embedded Design with the PIC18F452 Microcontroller. Pearson Education Inc, New Jersey, 2003.