JSTI 2021 Virtual Robotics

Shon Barthell, Olivia Chen, Elena De Santo, Fay England, Darryl France, Yale Kim, Kayla Magruder, Adam Tedesco, and Vincent Vest

Joint Science and Technology Institute, Virtual 2021

Introduction

Robotics is building upon itself and changing everyday, from phone applications to national security, and everything in between. The robotics group used a robot called a Boe-Bot, which is a controllable robot that uses touch and infrared sensors to navigate its environment. The Boe-Bot is controlled using the programing language PBASIC. Python, a language that is commonly used in computer science, was used to control the quadcopter as well as solving coding challenges. Robotics is applicable and beneficial in many ways and is constantly adapting to new needs.

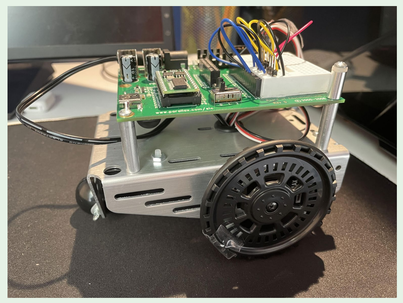
Background

Many of the machines that helped in everyday life worked manually, but as time went on people needed to find a more efficient way of using tools which has led to four main aspects to robotics:

- **Mechanical Aspect**
  - Physical frame of the robot
  - Supports the rest of the robot
- **Computer Programming Aspect**
  - Considered the brain of the robot.
  - How a robot knows to do something and how it decides when it does it
- **Electrical Aspect**
  - Gives the robot power
  - Controls various different machinery
  - Used for movement, sensing, and operation
- **Artificial Intelligence Aspect**
  - Type of coding that allows the robot to learn on its own (without human influence)

```
DO
  IF (IN5 = 0) AND (IN7 = 0) THEN
    GOSUB Back_Up
    GOSUB Turn_Left
    GOSUB Turn_Left
  ELSEIF (IN5 = 0) THEN
    GOSUB Back_Up
    GOSUB Turn_Right
  ELSEIF (IN7 = 0) THEN
    GOSUB Back_Up
    GOSUB Turn_Left
  ELSE
    GOSUB Forward_Pulse
  ENDIF
LOOP
```

A piece of code used to program the robots in PBasic



The base Boe-Bot put together with wires installed.

Materials and Methods

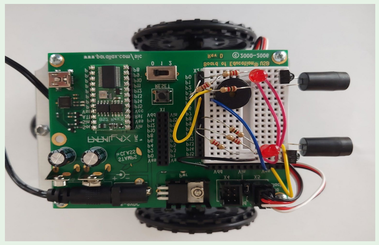
- BASIC Stamp - This software supports the programming language PBASIC that gives the chance to communicate, program and control the Boe-Bot in various ways.
- Boe-Bot - The Boe-Bot included a guide of instructions about how to modify and change the robots in many ways. It showed the different behaviors and what those changes do to the Boe-Bot.
- PyCharm - This software helps to learn and understand python, which is a more commonly used language of programming, often used in modern software applications
- Codrone - This small drone, along with pre-written code, helped to visualize what a complex and fully written code base could do with the device that is being controlled.

Results

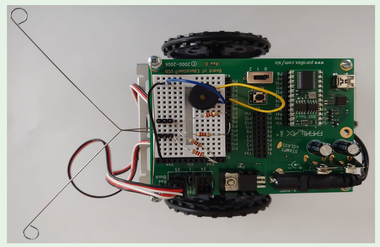
The Robotics group research consisted of 2 main parts:

- Constructing the Boe-Bot and coding it with PBASIC
- Programming in the python language in various projects

	Characteristics and Abilities	Findings and code changes
Functions and design	Basic Boe-Bot - Which wheels served as the front or tail - Efficient location of parts Servo Motors - Experience in physical construction - Servos calibrated to ensure synchronization	- How multiple systems can work together, i.e car assembly - Understanding the commands for rotation of a servo in many everyday uses - Pulse duration controls servo speed
Circuitry and interaction	LED - Intensity varied with resistance and current - Worked in tandem with wires, speakers, and sensors Speaker - Signals program restart/re-run - Ability to control frequency or pitch through	- Modifications to show when things are functional. Error messages, warning beeps - How current, voltage, and resistance all affect robotic systems and circuitry - HIGH/LOW command is intertwined with location of speaker - FREQOUT command controls pitch/frequency and duration
Autonomy	Whiskers - Adaptation based on physical interaction input IR Sensors - Adaptation based on sight and distance detection - Viewing distance changed resistors and color of objects	- Distinction between autonomous and semi-autonomous systems - Navigation applications like self driving cars and automatic braking - Various methods of robots interacting with environment - Signalling when there is input of computational data



Boe-Bot with Infrared Sensors and Red LEDs



Boe-Bot with Whisker Sensors and Piezo Speaker

Programming Results

Students used Python to program four main projects: Farm Legs, Dice Roll, Guess the Number, and Rock Paper Scissors. In these projects many aspects of coding were implemented:

- **Importing packages:** In most coding languages, packages have been created so that you do not need to reinvent the wheel every time you want to use it. The packages are then called into programs and used.
  `import random`
- **Variables:** Defining variables is a way to set a letter or a string of letters to a value to be used frequently in the code. For example: A = 9, were the variable A is given the value 9.
- **User Input:** A way for the program to get values necessary for completion from a user. `user_input = input()`
- **While loops:** A line of code that runs the code until the defined conditional statement is false.
- **Functions:** A way to group specific parts of code will need to be executed multiple times.
- **If, Else statements:** Statements that say if a conditional is met follow one path of code and if the conditional is false then follow a different path.

Conclusions

Robotics was a great learning experience in programming and engineering. Machines and robots help us everyday. Without robotics, think about how much harder some things would be, such as how construction efficiency would decrease and the industry as a whole would suffer. The skills necessary for robotics, such as programming and problem solving, are so versatile that they are applicable to almost any career, within or outside of STEM. Some projects we completed that led to the acquisition of these skills included programming a drone's flight patterns using Python, and building & programming our own robot.

Building the robot was the catalyst of learning to troubleshoot due to the wide array of conditions that needed to be met in order for it to work properly. The process of building and coding the robot was far from flawless. There would be issues ranging from errors in our code, circuits in incorrect places, or things just malfunctioning, but we would always find the problem, solve the problem, and get things back on track. Working in Python taught us important lessons in programming which will most definitely be useful in almost any job involving computers. The world of robotics is incredibly beneficial, and it was great for our group to gain some experience in that world thanks to the help of JSTI.

Acknowledgements

Thank you to Dr. Ron Hann, Carl Brown, Jennifer Tyrell, Kayla Canario, Karen Brummett, Gary Cipinko, Harrison Hughes and Edward Rychwalski. We appreciate the opportunity provided to us by the Defense Threat Reduction Agency's Joint Science and Technology Office for Chemical and Biological Defense, the Oak Ridge Institute for Science and Education, and the Georgia Tech Research Institute.