Solar Hydroponics Monitoring Vehicles
Final Research Project Report

SAC Undergraduate Research Project

Design and Drive Team:

Tepher Ward

Juli Williams

Michelle Mata

Submitted: August 19, 2016

TABLE OF CONTENTS

ABSTRACT

The ReEnergize Solar Hydroponics project at the SAC Eco Centro needs tools to enable monitoring of growing plants while minimizing entry and exit from the outside (Lewis). A mobile robot that can stay within the container while allowing remote control and transmission of visual and sensor data would have substantial benefit. Two remote monitoring vehicle prototypes were built to test the feasibility of an affordable, expandable and rugged mobile robot to be used within a hydroponic shipping container. The first vehicle was built from Tetrix parts and uses an Arduino microcontroller. The second vehicle, built from PVC and other common materials, is controlled by a Raspberry Pi single board computer. Both vehicles can be controlled remotely and stream video from an on-board camera, and both are large enough to accommodate additional equipment to extend functionality. The first vehicle is more rugged, but the second vehicle offers more flexibility at half the cost of the first. The project has met its primary objectives of having remote control drivability and camera visibility via WiFi. Future enhancements include the line sensors for the vehicles to follow pre-programmed paths for sensor readings and the swivel-mount for the camera to view objects from left to right or up and down without repositioning the vehicle. Adding a camera hoisting mechanism to view higher plant racks will require retesting the vehicles' balance and weight distribution to ensure consistent traction. As sensor equipment is added, the WiFi will need to transmit the additional data readings. The final enhancement recommended is to make all systems moisture-proof.

PARTICIPANTS AND ACKNOWLEDGEMENTS

Team Participants

Stephanie (Tepher) Ward – Team Leader; V1 & V2 Design & Construction

Juli Williams – V2 Lead, V2 programmer

Michelle Mata – V1 Lead, V1 programmer

Team Advisors and Supporters

Mr. Steven Lewis – Director of Eco Centro, Project Advisor

Mr. Klaus Bartels – Adjunct Faculty, Physics, Engineering, and Architecture Department, Project Advisor

Mr. Ben Uresti – Academic Lab Technician for the MESA Center, Technical Advisor

Ms. Bly Korseau – Engineering Administrative Assistant

Ms. Barbara Knotts – Adelante Tejas Project Grant Director

Ms. Patty Medina – Exitos Grant Director

Ms. Dee Dixon – MESA Center Coordinator

Ms. Sylvia San Miguel – Administrative Assistant

Ms. Susan Paddock – LSAMP/CIMA Grant Co-PI

Ms. Susan Espinoza – Director of College Grants and Development

Dr. Robert Vela – SAC President

Sponsors

Exitos grant (Award No. 031S140099) – Grantor and Supporter

Adelante Tejas (Award No. PO31C110039) – Grantor and Supporter

San Antonio College – Contributor and Supporter

INTRODUCTION

San Antonio College works with Texas State University through the ReEnergize program, which aims to research renewable energy alternatives and to share green technology solutions (ReEnergize - Overview). San Antonio College's Eco Centro is the home for one of the ReEnergize projects focused on developing a low-cost, solar-powered hydroponic shipping container that could grow and supply food for local food deserts, global climate-challenged areas, or urban areas where food supplies are limited.

In working with the ReEnergize Solar Hydroponics team, they discovered that the hydroponics shipping container at Eco Centro should only be accessed on a minimal basis to reduce exposure to pests and maintain optimal air, light, and water conditions for growing plants (Lewis). As cost-containment and long-term adaptability are essential for the hydroponic shipping container program, these concerns were incorporated into the goals of this project. Additionally, since the hydroponics teams would have limited access to the shipping container for on-going monitoring and periodic surveillance, the solution needed to be controlled remotely and to be programmable for scheduled sensor reading in various locations in the shipping container. While the shipping container will have a ventilation system, the atmospheric conditions will range from 70° - 90° F and from 60% - 85% humidity.

Based on the climatic extremes of the environment, the vehicle needs to be corrosion resistant in the humid environment, energy efficient to minimize charging and down time, provide a platform for attaching and running various sensor equipment and monitors, and provide wireless video capability for visual checks. The base of the vehicle needs to be large enough to accommodate future enhancements that would include a hoisting camera mechanism to view higher plant racks or to raise interchangeable attachments for harvesting or other functions. These enhancements must accommodate reaching up to 8 ft., which is the interior height of the shipping container.

After assessing the requirements, it was determined that a remote-controlled monitoring vehicle not only met the needs of the ReEnergize Solar Hydroponics team but also addressed the environmental constraints and challenges. While there were robotic kits available, the concern was that they would not meet the program requirements of being cost efficient and adaptable. The Hydroponics Monitoring Vehicles team set out to compare two vehicles to determine which would be better suited to meet all needs, including end-user, environmental, and program requirements.

Given time constraints and very limited team expertise in robotics, some features were deferred as not key in determining the viability of the goals of the project. These included the elimination of line sensors for the vehicles to follow pre-programmed paths for sensor readings, and the elimination of swivel-mounted cameras for viewing objects from left to right or up and down without repositioning the vehicle.

MATERIALS AND METHODS

Two vehicles with similar functionality were designed and constructed to move forwards and backwards as well as to turn on command. The Tetrix vehicle (V1) was made with Tetrix parts with an Arduino controller (Figure 1). The hybrid vehicle (V2) was made of PVC and other off-the-shelf

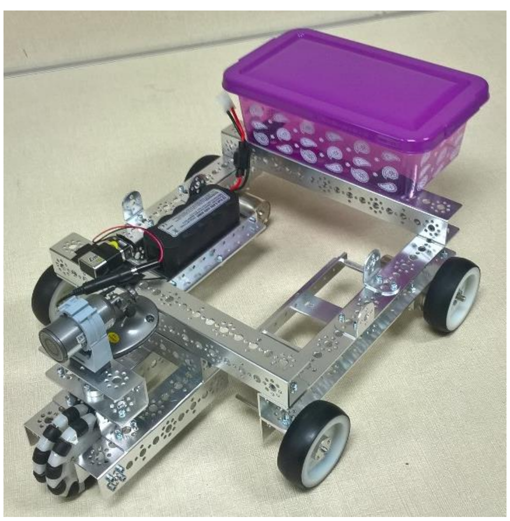

Figure 1: Tetrix Vehicle Design for V1

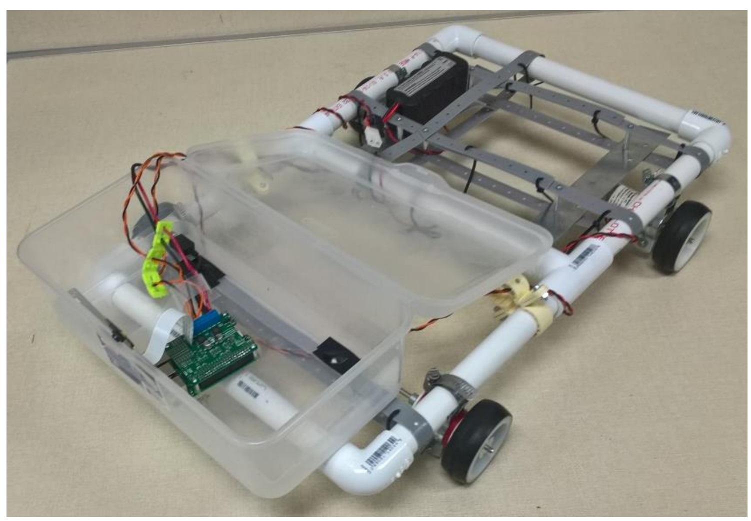

Figure 2: Hybrid Vehicle Design for V2

components available at hardware stores (Figure 2) and is controlled by a Raspberry Pi single board computer. The centers of both vehicles remained open for the addition of sensors or other components. Since the PVC pipe could be cut to any size, V2 was designed a bit larger to allow for better balance for a future enhancement arm that would extend to the full height in the shipping container of 8 feet (i.e., a camera hoisting or harvesting mechanism harvesting).

The total cost to build the two vehicles was $1,298.64. Some of these costs were deferred as San Antonio College provided two existing Tetrix Prime Kits at no cost as an alternative to the Tetrix Urban Challenge Set. Total cost of parts and materials for V1 was $879.17 (Appendix A, Table A1), and for V2 was $419.47 (Appendix A, Table A2).

V1 – Tetrix Vehicle

Construction (Appendix B, Images B1 – B3)

The frame of the main body of V1 was constructed in a square using 288 mm beams and extended from the center of the front to mount the camera. A plastic container was attached at the back to house the controllers and future sensors. The total length is 22.5 in. from front to back. Wheels extend out on the sides for a total width of 14 in. The height including the plastic container was 8.25 in. The camera was attached on a fixed mount 7 in. from the floor, which provided enough visible range to view the floor at approximately 12 in. in front of the vehicle.

The motors were connected to the rear wheels with the battery pack for the motors in the center to distribute weight evenly. The weight of V1 including its components was 6 lbs. 15.2 oz. The 9V battery for the camera was secured to the frame by electrical tape.

Electronics

V1 uses an Arduino UNO microcontroller, an Arduino WiFi Shield 101 that creates an interface to control the

Arduino UNO, a Pololu motor shield that has the capability to control two 12V motors, and an attachable wireless camera that has built-in IR LEDs for night vision.

The Arduino UNO is a board that includes an ATmega328P microcontroller with a USB connection, a power jack, an ICSP header, a reset button, and a 16MHz quartz crystal. The Arduino UNO contains pins which are attached to the microcontroller which include 14 digital input/output pins for which 6 of those can be used as a pulse-width modulation (PWM) pins, and 6 analog inputs. The Arduino UNO is a small board of 68.6mm in length, 53.4mm in width and only weighs 25g. Technical specs of the Arduino UNO include a 5V operating voltage, 7-12V recommended input voltage, 6-20V input voltage limit, 20mA DC current per I/O pin, 50mA DC current for 3.3V pin, 2KB SRAM, 1KB EEPROM, a Clock Speed of 16MHz and a Flash Memory of 32KB which is stored in the microcontroller and of which 0.5KB is used by the bootloader (Arduino - Overview).

The WiFi Shield 101 allows the Arduino UNO board to connect to a local network using IEEE 802.11 wireless specifications. The shield is mounted on top of the Arduino UNO and contains an AMTEL SmartConnect WINC1500 module that is a network controller capable of TCP and UDP protocols, features a hardware encryption/decryption security protocol included in the ATECC508A CryptoAuthentication chip, and includes its own downloadable library of the Arduino software. The shield uses digital pins 5, 7, and 10. Pin 5 is used as a reset pin between the shield and Arduino, pin 7 is used as a handshake between the shield and Arduino, and pin 10 is used as an SPI to the Arduino (Arduino WiFi Shield 101).

The Pololu Motor Shield included a MC33926 Motor Driver that can control up to two motors. The motor shield was stacked on top of the WiFi Shield 101 and was separately powered from the Arduino UNO and WiFi shield. The motor shield has a 5V minimum operating voltage, 28V maximum operating voltage, 3A of continuous output current per motor, 0.525 V/A current sense, 20kHz of maximum PWM frequency, 2.5V of minimum logic voltage, 5.5V maximum logic voltage and includes reverse voltage protection included in the MOSFET. The shield uses many pins which include digital pins 4, 7, 8, 9, 10, 12, and analog pins A0 and A1. Pins 7-10 are used to control the motor’s power and speed while pin 4 is the "if fault" command that stops the motors if a fault occurs and pin 12 disables motor output inputs when low. Below is the truth table for the Motor shield (Table 1).

MODE	NotD2	MxPWM	MxDIR	MxA	MxB	NotSF
Forward	H	H	L	H	L	H
Reverse	H	H	H	L	H	H
Brake	H	L	X	L	L	H
Coast	L	X	X	Z	Z	L

Table 1: Logic Controls for Arduino Motor Shield Programming

The wireless camera used on V1 is the SecurityMan CUcam1 that features colored video and audio, a night vision range of 23 ft., video transmission up to 330 ft. from the receiver, and a 3.3 mm fish eye lens. It is weatherproof so it can easily be used indoors or outdoors. The website indicates it is a WiFi interference-free camera indicating that the device will not interfere with other wireless devices (SecurityMan).

Programming

The Arduino software was installed and run using Windows 10 OS. The Arduino software can be found on the official Arduino site and many downloadable libraries are within the software (Arduino-Software). For this project it was necessary to download the WiFi Shield 101 library and the Pololu Motor Shield library, which can only be obtained in the motor shield's GitHub page (Pololu / Dual MC33926 Motor Shield).

After mounting the WiFi shield on the Arduino, the first programming done for the project was to connect it to a local network. The WiFi shield library contained an example of connecting the shield to the network and in using this it was successfully connected. The WiFi shield library also contained an example to control an LED over a web server which was also a success. The Pololu motor shield was initially tested directly mounted to an Arduino without the WiFi Shield 101, and it ran successfully. The Pololu motor shield had only one sample code called "DEMO" that moved both motor channels forward and in reverse. The motor shield was then tested mounted on the WiFi shield, but then it only ran one motor successfully with the WiFi Shield due to a pin conflict.

To simulate a remote environment, we attached a phone charger to provide battery power to the Arduino as opposed to a direct connection of power through a PC. The Arduino powered up and executed the code successfully using the phone charger.

V2 – Hybrid Vehicle

Construction (Appendix B, Images B4 – B6)

V2's frame was constructed from a 10 ft., 0.75 in. Schedule 40 PVC pipe. Four pieces were cut to 9 in. for the sides and three pieces were cut to 12 in. for the cross beams. The four 9 in. pieces make up the side beams and were attached between ¾-inch 90-degree fittings at the ends and to a ¾-inch Tee fitting at the center. The side beams were then attached with the 12 in. PVC pipes to complete the frame (Figure 2). The total length was 26.25 in. with a width of 17.5 in. and height of 8 in. to the top of the closed flip-top box.

There are two motor-driven wheels in the rear of the vehicle held by a #2 conduit hanger with a bolt tightened to prevent the motor from slipping. The top of the conduit hanger is mounted to the chassis by a 1.25 in. hose repair clamp. For the wheels without motors, thread spools of a similar size to the motors were used with the wheel axle run through the center spool keeping the vehicle balanced from front to back.

The battery pack was placed at the rear to add weight for keeping the tires in contact with the floor. Plastic hanger straps were wound around the back frame and attached by 5 in. zip ties (Figure 3). Then, a 1.5 in. by 9 in. steel plate was affixed to the lower strap with the upper strap used to compartmentalize the battery pack. Standoffs were used to add stability between the straps and steel plate. The weight of V2 including all its components was 7 lbs. 2.7 oz.

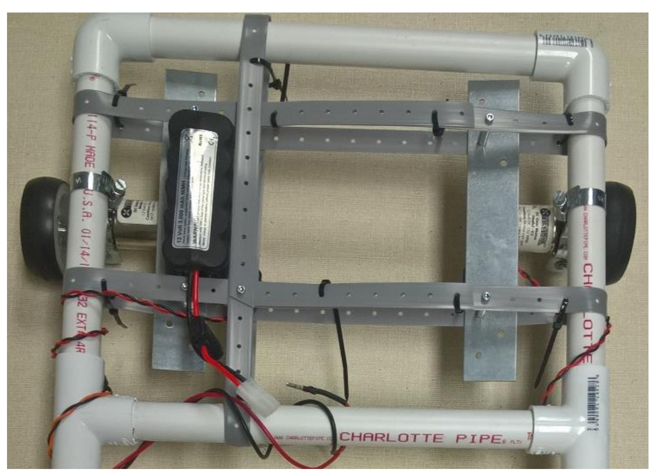

Figure 3: V2 battery holder

Electronics

Vehicle 2 uses a Raspberry Pi 3 single board computer, a Pololu dual motor driver add-on board controlling two 12V DC motors for movement, and a Pi NoIR (no infrared) camera module providing a video feed.

The Raspberry Pi 3 (RPi) features a Broadcom BCM2837 system-on-chip, combining a quad-core 64-bit ARM Cortex processor at 1.2GHz clock speed, 1GB RAM, and 400MHz VideoCore IV GPU. The board also includes a microSD slot, built-in Bluetooth 4.0 and 802.11n WiFi connectivity, 4 USB ports, 10/100 Ethernet port, HDMI connector and 3.5mm audio/video jack. Its small size, low cost, 40-pin general-purpose input/output (GPIO) header, and large user community contributing code, project ideas, and compatible hardware, make it ideal for inexpensive robotics projects with plenty of expansion opportunities. It is powered by a standard 5V micro USB supply, such as a cellphone charger, with a total current draw typically under 1A (Raspberry Pi FAQs).

The Pololu expansion board is capable of driving two DC motors between 5-28V and providing up to 3A continuous, 5A peak current per channel, with current limiting and over-temperature protection. In initial testing, the 12V Tetrix DC motors each drew up to 0.15A under no load, 0.38A with a light to moderate load, and 3.9A to start the motor. The board is plugged into the pins directly on top of the RPi, held securely with screws, and includes terminal blocks for connecting the motors and separate power supply.

The connection between the motor driver board and the RPi uses 8 GPIO pins, with each motor having a status flag output, enable pin (EN), direction pin (DIR), and speed input using pulse-width modulation (PWM). For basic operation, EN must be set to High to turn the motor on while polarity, which controls the direction the shaft turns, is set by DIR. Setting this pin to High will enable counterclockwise rotation, while Low switches the direction to clockwise rotation (Table 2).

Direction	M1EN: pin 22	M1DIR: pin 24	M2EN: pin 23	M2DIR: pin 25	State
Forward	High	High	High	Low	M1 counterclockwise, M2 clockwise
Reverse	High	Low	High	High	M1 clockwise, M2 counterclockwise
Left	High	Low	High	Low	M1 and M2 counterclockwise
Right	High	High	High	High	M1 and M2 clockwise
Stop					Speed, PWM set to zero

Table 2: DC Motor controls

The Pi NoIR camera module uses a Sony IMX219 8M sensor and lacks an infrared filter, making it suitable for measuring plant health when used with a blue filter (What's That Blue Thing). It connects to the RPi through a camera serial interface (CSI) port, providing a direct connection to the GPU, and provides video at resolutions of 1080p30, 720p60 and 640x480p90, or stills up to 3280x2464.

Programming

The OS used is the March 2016 release of MINIBIAN, a minimal implementation of Raspbian "Jessie" which fits on a 2GB SD card along with a selection of applications required for web-based controls and streaming video. Raspbian is an operating system tailored for Raspberry Pi and derived from the Debian Linux operating system.

The RPi uses Apache2, a Linux-based web server software, with the mod_wsgi plugin to serve an HTML file that includes a video stream from the NoIR camera with buttons to control the direction of movement and a text input field for setting the movement time in seconds. The webpage can be accessed by a computer on the same local area network by opening a browser and pointing it to the static address given for the RPi.

Video streaming is managed by Userspace Video4Linux (UV4L), which provides a web-based streaming server that can be accessed through port 8080 at the RPi's static address (e.g., 172.16.8.1:8080) and allows direct access to the configuration file for the video feed with appropriate credentials. The live stream can be viewed by selecting MJPEG/Stills stream.

Vehicle movement uses python scripts adapted from a tutorial with code provided on the *Linux User & Developer Magazine* website by Liam Fraser. The basis of this code is open-source under the Apache2 license, including the Bootstrap web framework which integrates a plugin allowing user interaction without disturbing the video feed. The scripts used to manipulate the GPIO pins associated with the motors work by setting up a Python Remote Objects (Pyro) server that allows remote access. The "movement server" script defines the GPIO pins to be used, the functions called to manipulate them, and enables the Pyro server to wait for these commands. The application script takes the arguments passed to it from the web page and calls the assigned function to execute the command.

RESULTS AND DISCUSSION

V1 – Tetrix Vehicle

Construction

The design was intended to provide adaptability for the addition of future sensors along with an extended camera hoisting mechanism in the center while still remaining balanced. No holes were added to the controller container, so wires were draped over the edge with electrical tape as an anchor. While the design was original, its simplicity in structure resulted in no changes to the initial build design.

Electronics

The initial design had included the use of Pitsco's Tetrix PRIME Arduino Hardware Kit, which included an Arduino UNO and Tetrix Motor shield at a cost of $160. This shield could only support servo motors, which are not powerful enough for moving vehicles. The final version used the Pololu motor shield, which did overcome the limitations and only cost $29.95.

Programming

Following the successful programming controls for the motor shield, the WiFi shield was then added. A problem was encountered when the motor shield and WiFi shield were both mounted on the Arduino. Both shields were interrupting the program to work properly because they used some similar pins. Adding shields on top of shields seems to be a very common problem in the Arduino community.

With the current problem there were different options to proceed with the project. One was to rearrange the mapping of the motor shield pins to not make an interference; but, it would also not allow the use of the motor shield library. Another option was to only power one motor, meaning the design of V1 would need to change. Michelle looked for help on online forums to get any advice on which option would be a better fit.

Writing up the code to control the motors over a web server was the most difficult part of the project. Having a beginner programmer do this kind of programming took hours of research and many hours of self-taught advanced programming. The challenge was incorporating HTML code into the Arduino programming. Even if the program compiled fine the web server would not appear because of a small mistake. Once the web server was up and buttons and dialogue appeared on screen the programming moved faster. The code that was created was meant to move both motors forward, reverse and to stop both motors. With a deadline approaching we were able to only make one motor be controlled over WiFi.

V2 – Hybrid Vehicle

Construction

The initial design matched fully with a single exception in attaching of the motors. The first build included 0.75 in. plastic tube clamps. While they held the motors firmly, they would rotate around the PVC pipe frame. Two options were assessed: adding a screw through the tube clamp and the PVC pipe or changing the part to a stronger clamp. Based on durability and relatively comparable prices, the final design used a galvanized steel conduit hanger with a bolt that adjusts to provide a secure hold on the motor. That was then attached to the PVC pipe using the hose repair clamp. If a smaller motor needs to be used in the future, the conduit hanger can be easily and inexpensively replaced with a smaller size.

Like V1, no holes were added to the controller box, so wires were draped over the edge with electrical tape as an anchor. Since the camera is a chip about the size of a quarter, it is taped to the front of the box with the wires preventing the box from closing on it.

The initial plan included 6 in. square grates located in the center of each of the sections to serve as the base for the camera hoisting mechanism. This was not included in the project and can be added as a future enhancement. By excluding this, the weight was reduced significantly and likely resulted in the wheels not having sufficient traction with the floor to move the vehicle properly. To add weight near the motors, a second steel plate was added at the holding bin.

Electronics

The initial motor driver did not have enough capacity to support the Tetrix DC motors being used. After researching alternative motor drivers, it was determined that Pololu provided a RPi compatible motor driver board. This did successfully power the motors. As Pololu had an Arduino-compatible shield, both vehicles used this motor driver.

Programming

The vehicle does not require a full desktop environment with graphical user interface (GUI) and standard user applications; so, these programs are not installed. Instead, the basic system provides a command line interface with a Secure Shell (SSH) server enabled to allow remote login. Additional services that need to be installed and configured include a web server, video camera streamer, WiFi access point and Python 2.7 programming language. Other useful, optional services include an email system and scheduler. All software used is available under some variant of free open source software (FOSS) license, with the most restrictive being the GNU General Public License (GPL), which requires that derivative works carry the same license (Morin, Urban and Sliz). However, when dealing with FOSS distributions it is important to note that some packages may not be backwards-compatible with GPL code and should be installed separately.

In order to have a video streaming connection without requiring an external network, the RPi has its own wireless access point (WAP) using Hostapd and Dhcpd. These daemons, or non-interactive background processes, control authentication and IP address assignment for the connecting client machines. The default configuration for this WAP uses the reserved private address range of 172.16.8.1-

22. This range was chosen to minimize potential conflicts with the more commonly used 10.0.0.0/8 and 192.168.0.0/16 spaces.

The web controls present live video streamed from the attached camera and buttons for selecting commands to move the vehicle: forward, back, left, right, and stop. A text box is provided to input the number of seconds to run the motor.

Vehicle movement was originally attempted using code provided by Pololu Robotics and Electronics for use with their motor driver board. However, this code used tools requiring root (administrator) privileges to access within what should be web-accessible controls, so another approach was sought. To get the controls working as quickly as possible, the framework previously described was used, with modifications to the scripts to access the correct pins and define the functions appropriately. More complex control will require different methods for manipulating the pins.

To determine the speed of the vehicle in testing, distance traveled was measured at intervals of 2, 4 and 6 seconds (Appendix C, Table C1). These measurements were taken at different speed constants declared in the movement python script: 50, 60, and 80%. At the default speed of 50%, the vehicle averaged 0.29 m/s. At 60% the average speed was 0.35 m/s, and at 80% the speed was 0.48 m/s. Rotation angle was planned to be determined by noting the number of seconds required to turn in a complete circle. Left and right turning were not the same, and required a weight to be added on the rear left to balance the battery pack which had been placed at the rear right. A complete rotation at the default speed took approximately 9.5 seconds leftward, but failed to make a full rotation to the right at test values below 80% speed.

A mail system, Postfix, and a scheduler, Cron, are included to aid future task automation. Predefined movements for routine sensor readings can be defined in a script which is called by the scheduler, with reports generated and sent using the mail system. Many maintenance tasks can be automated with the scheduler while logs, reports and summaries are routed to an external network.

Pigpio is also installed to prepare for more complex control of the motors using PWM to vary the speed and facilitate the development of advanced scripts for autonomous movement.

CONCLUSIONS

As expected, both vehicles proved to be capable of meeting the end-user needs for remote control and WiFi transmission of sensor readings. While these were not programmed or tested specifically, the WiFi connections to run motors were. Both Arduino UNO and Raspberry Pi R3 are capable of additional functionality and/or added controller boards. As for the environmental requirements of handling somewhat extreme climatic conditions with high temperatures and humidity, both Arduino and Raspberry Pi have sensors and components that are waterproof or water resistant. As alternatives, there are many moisture-resistant, fitted cases available for either. Another alternative includes heat shrink tubing to ensuring exposed wires are tightly sealed at connection points. As all of these are more permanent or challenging to reconfigure easily, it is recommended that using these options be deferred until more functionality is included and tested.

Meeting the program requirements was the final area of assessment. This is where differences in the practicability of the vehicles was more noticeable. While the capabilities of Arduino UNO as a controller was relatively comparable to the Raspberry Pi in terms of function and price, the Tetrix parts and materials for V1 cost approximately $420 compared to those used in the hybrid V2 at approximately $170. That differential increases further when non-Tetrix motors and rechargers are substituted for V2. As for long-term adaptability, the Tetrix materials are limited in size and much heavier. With our simplistic designs, the weights were almost equal, but V2 had a much larger base. When comparing the area that can be used for component additions, V2 had a workable area of 445 sq. in. compared to V1 with 127 sq. in. In addition to space for components to be attached, the increased area improved balance when adding components that extend higher like the hoisting mechanism to reach the full height of the shipping container for sensor readings.

In the end, V2 was the better option for meeting the needs of end-user, environmental, and program requirements and would prove to be a beneficial tool for assisting the ReEnergize team in developing a low-cost hydroponic shipping container that could grow and supply food for local food deserts, global climate-challenged areas, or urban areas where food supplies are limited.

Before full deployment, there are several enhancements and improvements that need to be added. These include adding line sensors for the vehicles to follow pre-programmed paths for sensor readings, and the swivel-mount for the camera to view objects from left to right or up and down without repositioning the vehicle. Adding a camera hoisting mechanism to view higher plant racks will require retesting the vehicles' balance and weight distribution to ensure consistent traction. As sensor equipment is added, the WiFi will need to transmit the additional data readings. The final enhancement recommended is to make all systems moisture-proof.

REFERENCES

"Arduino - Introduction." *What Is Arduino?* Arduino. N.d. Web. May 2016. <https://www.arduino.cc/en/guide/introduction>.

“Arduino – Overview.” Overview. Arduino. N.d. Web. https://www.arduino.cc/en/Main/ArduinoBoardUno.

“Arduino - Software.” *Download the Arduino Software*. Arduino. N.d. Web. July 2016. https://www.arduino.cc/en/Main/Software.

“Arduino WiFi Shield 101.” *Arduino WiFi Shield 101 Overview.* N.d. Web. July 2016. https://www.arduino.cc/en/Main/ArduinoWiFiShield101.

“SecurityMan.” *Cucam1*. N.d. Web. July 2016. http://www.securitymaninc.com/?page_id=3683.

"Hostapd." *Gentoo Wiki*. N.p., n.d. Web. 10 Aug. 2016. https://wiki.gentoo.org/wiki/Hostapd.

Lewis, Amanda. "Eco Centro's Hydroponics Container." Telephone interview. 08 Apr. 2016.

"Pololu / Dual MC33926 Motor Shield." *GitHub*. 7 June 2012. Web. July 2016. https://github.com/pololu/dual-mc33926-motor-shield.

"Pololu Dual MC33926 Motor Driver for Raspberry Pi (Assembled)." *Pololu Robotics and Electronics*. n.d. Web. 15 Aug. 2016. https://www.pololu.com/product/2756/resources/.

"Raspberry Pi 3 is out Now! Specs, Benchmarks & More The MagPi Magazine." The MagPi Magazine, Mar. 2016. Web. 11 Aug. 2016. https://www.raspberrypi.org/magpi/raspberry-pi-3-specs-benchmarks/

"Raspberry Pi FAQs - Frequently Asked Questions." Raspberry Pi Foundation, Web. 12 Aug. 2016. https://www.raspberrypi.org/help/faqs/.

"Raspberry-Pi NoIR Camera Module v2 Technical Datasheet." Farnell Element14 | Electronic Component Distributors. 2016. Web. 11 Aug. 2016. http://www.farnell.com/datasheets/2056180.pdf/.

“ReEnergize – Overview.” *Texas State University Ingram School of Engineering ReEnergize Overview*. n.d. Web. Aug. 2016. http://reenergize.engineering.txstate.edu/about/overview.html.

Soltoggio, Luca. "MINIBIAN 2016-03-12 is Out! | MINIBIAN – Raspberry Pi." *MINIBIAN - Raspberry Pi*. N.p., 12 Mar. 2016. Web. 10 Aug. 2016. https://minibianpi.wordpress.com/2016/03/12/minibian-2016-03-12-is-out/.

"Tetrix Prime Arduino Hardware Kit." *Pitsco Tetrix*. N.d. Web. May 2016. <http://www.tetrixrobotics.com/TETRIX_PRIME_Arduino_Hardware_kit>.

"Tutorial — Pyro 4.46 Documentation." *Documentation of Various Python Packages*. N.p., n.d. Web. 12 Aug. 2016. https://pythonhosted.org/Pyro4/tutorials.html/.

"UV4L – Projects." *Projects – Home*. N.p., n.d. Web. 10 Aug. 2016. http://www.linux-projects.org/uv4l/.

"What's That Blue Thing Doing Here?" *Raspberry Pi*. N.p., n.d. Web. 11 Aug. 2016. https://www.raspberrypi.org/blog/whats-that-blue-thing-doing-here/.

APPENDICIES

Appendix A – Parts and Materials Lists by Vehicle

V1 Parts and Materials								
	Item Name	**Purpose**	**Vendor**	**Model / Part #**	**Qty Used**	**Price / item**	**Qty Purc**	**Total Cost**
1	288 mm Square Beams (2/pkg)	Frame	Pitsco	W39068	4	16.95	2	33.90
2	160 mm Square Beams (2/pkg)	Frame section for camera	Pitsco	W39067	2	14.95	1	14.95
3	96 mm Square Beams (2/pkg)	Frame section for camera	Pitsco	W39066	2	11.95	1	11.95
4	32 mm Square Beams (2/pkg)	Frame section for camera; wheel mounts	Pitsco	W39065	2	9.95	1	9.95
5	Max L Bracket	Wire channels	Pitsco	W39062	2	4.95	2	9.90
6	144 mm MAX Angles (2/pkg)	Battery holder	Pitsco	W39072	2	10.95	1	10.95
7	MAX Flat Bracket	Battery holder	Pitsco	W39061	2	5.95	2	11.90
8	2” Stand-off Posts (12/pkg)	Battery holder rail	Pitsco	W39103	4	7.95	1	7.95
9	3” MAX Wheels	Wheels	Pitsco	W39025	4	15.95	4	63.80
10	3” MAX Omni Wheels (2/pkg)	Wheels supporting camera frame	Pitsco	W31132	2	24.95	1	24.95
11	MAX Axle Hub	Axles	Pitsco	W39172	6	6.95	6	41.70
12	11 mm PRIME Bronze Bushing	Axles	Pitsco	W40227	6	9.95	6	59.70
13	1/8” PRIME Axle Spacers (12/pkg)	Axles	Pitsco	W39100	6	1.95	1	1.95
14 4	80 mm PRIME Steel Axles	Axles	Pitsco	W40225	3	12.95	3	38.85
15	MAX Axle Set Collar	Axles	Pitsco	W39092	2	3.95	2	7.90
16	MAX Motor Mount	Motor mount	Pitsco	W39089	2	7.95	2	15.90
17	MAX DC Motor	Motor	Pitsco	W39530	2	24.95	2	49.90
18	MAX Motor Power Cable	Motor	Pitsco	W31903	2	1.95	2	3.90
19	Additional wires	Motor wires	Harbor Freight		4	1.99	1	1.99
20	MAX 12V 3000mAh NiMH Battery	Motor battery	Pitsco	W39057	1	49.95	1	49.95
21	NiMH Battery Pack Charger	Motor battery recharger	Pitsco	W39830	1	24.95	1	24.95

22	Tetrix MAX Wireless Camera Kit	Tetrix vehicle camera; includes 9V battery	Pitsco	W39683	1	140.00	1	140.00
23	½" Socket Head Cap Screws (100/pkg)	Screws	Pitsco	W39097	67	9.95	0.67	6.67
24	Kep Nut (100/pkg)	Nuts	Pitsco	W39094	67	2.95	0.67	1.98
25	Plastic Container	Houses controllers	Dollar Tree	N/A		1.00	1	1.00
26	Arduino UNO	Controller	Intertex	A000066	2	23.95	2	
27	Arduino WiFi Shield 101	WiFi shield	Adafruit	2891	1	49.95	1	49.95
28	Dual MC33926 Motor Driver Shield	Motor driver shield	Pololu	2503	1	29.95	1	29.95
29	Stacking Headers	Shield stacking headers	Intertex	OSE-LS000008	1	1.75	1	1.75
30	Micro USB Battery	Controller battery			1	~5.00		5.00
							V1 TOTAL	**$ 879.17**

Table A1: V1 Parts and Materials

V2 Parts and Materials								
	Item Name	**Purpose**	**Vendor**	**Model / Part #**	**Qty Used**	**Price / item**	**Qty Purc**	**Total Cost**
1	10' PVC ¾" Pipe	Frame	Home Depot	57471	1	2.67	1	2.67
2	¾" PVC 90-Degree Elbow	Frame	Home Depot	C406-007	4	0.36	4	1.44
3	¾" PVC Tee	Frame	Home Depot	C401-007	2	0.46	2	0.92
4	¾" Tube Clamps (5/pkg)	Wire channels	Home Depot	HD559-3PK2	4	1.38	1	1.38
5	1 ½" x 9" Steel Safety Plate	Battery holder	Home Depot	HD538-09PK	2	0.48	2	0.96
6	1" Stand-off Posts (12/pkg)	Battery holder	Pitsco	W39102	5	3.95	1	3.95
7	¾" x 10' Plastic Hanger Strap	Battery holder	Home Depot	339232	1	2.97	1	2.97
8	5" Zip Ties (100/pkg)	Battery holder anchors	Harbor Freight	60254	13	1.99	1	1.99
9	3" MAX Wheels	Wheels	Pitsco	W39025	4	15.95	4	63.80
10	MAX Axle Hub	Axles	Pitsco	W39172	4	6.95	4	27.80
11	11 mm PRIME Bronze Bushing	Axles	Pitsco	W40227	2	9.95	2	19.90
12	1/8" PRIME Axle Spacers (12/pkg)	Axles	Pitsco	W39100	2	1.95	1	1.95

13	80 mm PRIME Steel Axles	Axles	Pitsco	W40225	2	12.95	2	25.90
14	MAX Axle Set Collar	Axles	Pitsco	W39092	2	3.95	2	7.90
15	#2 Conduit Hanger with Bolt	Wheel motor mounts	Home Depot	67820	4	0.72	4	2.88
16	1 ¼" Hose Repair Clamp	Wheel mount clamps	Home Depot	6712595	4	0.98	4	3.92
17	1 ¼" Plastic Thread spools	Non-motor wheel axle	Joanne's		2	0.99	2	1.98
18	MAX DC Motor	Motor	Pitsco	W39530	2	24.95	2	49.90
19	MAX Motor Power Cable	Motor	Pitsco	W31903	2	1.95	2	3.90
20	Additional wires	Motor wires			2	6.99	1	6.99
21	MAX 12V 3000 mAh NiMH Battery	Motor battery	Pitsco	W39057	1	49.95	1	49.95
22	NiMH Battery Pack Charger	Motor battery recharger	Pitsco	W39830	1	24.94	1	24.95
23	½" Socket Head Cap Screws (100/pkg)	Screws	Pitsco	W39097	32	9.95	0.33	3.28
24	Kep Nut (100/pkg)	Nuts	Pitsco	W39094	16	2.95	0.33	0.97
25	¼" washers	Washers	Ace		8	0.05	8	0.40
26	Large Flip-top Box	Container for controllers	Home Depot	1805860 6	1	2.97	1	2.97
27	Raspberry Pi 3 - Model B - ARMv8 with 1G RAM	Raspberry Pi with built-in WiFi for V2	Adafruit	3055	1	39.95	1	39.95
28	Dual 33926 Motor Driver for RPi	RPi motor hat	Pololu	2756	1	34.95	1	34.95
29	5MP camera module	Camera for V2	Adafruit	1367	1	19.95	1	19.95
30	GPIO Stacking Header for Pi 3 - Extra-long 2x20 Pins	Camera rotator gear for V2	Adafruit	2223		2.50	1	2.50
31	Brass M2.5 Standoffs for Pi HAT	Standoffs for RPi Hat	Adafruit	2336		0.75	2	1.50
32	Micro USB battery	Controller battery			1	~5.00	1	5.00
							V2 TOTAL	**$ 419.47**

Table A2: V2 Parts and Materials

Appendix B – Image Views of Vehicles

Images of V1

Figure B1: Front view of V1

Figure B2: Side view of V1

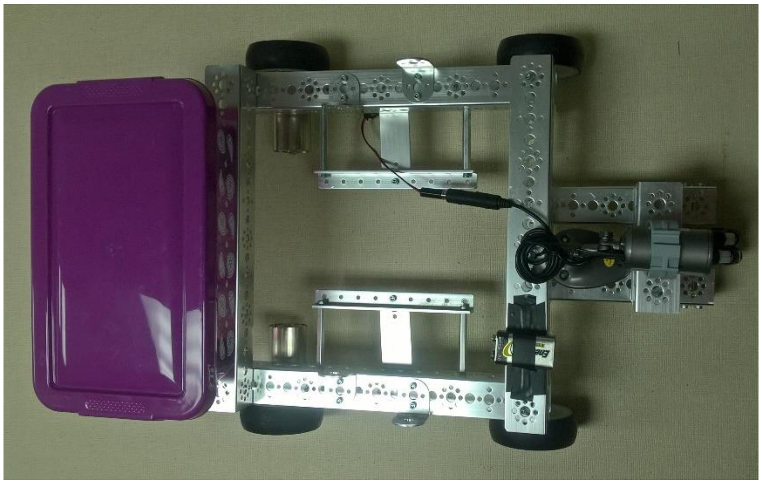

Figure B3: Top view of V1

Images of V2:

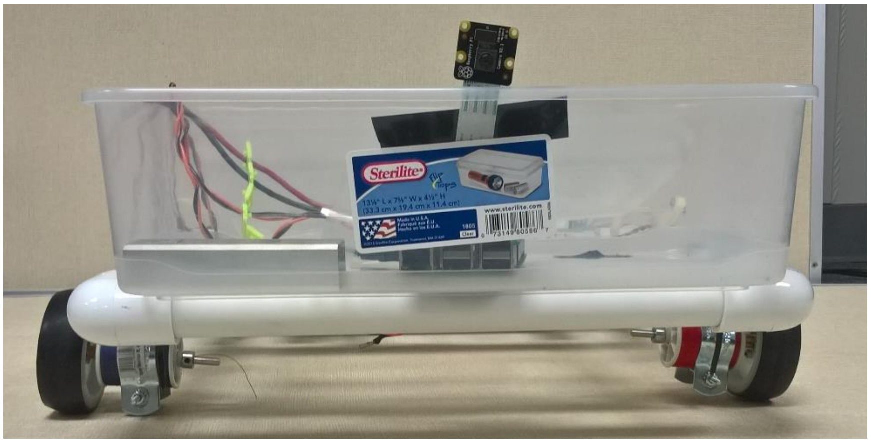

Figure B4: Front view of V2 with flip-top down

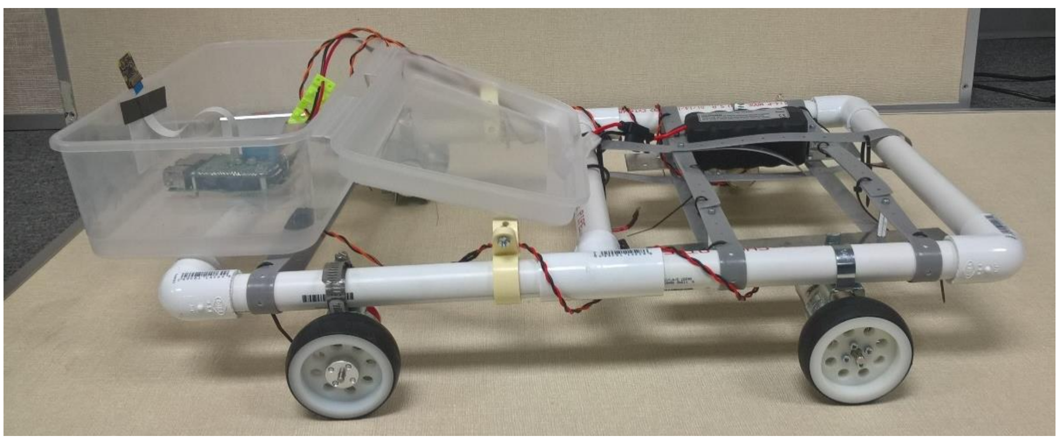

Figure B5: Side view of V2

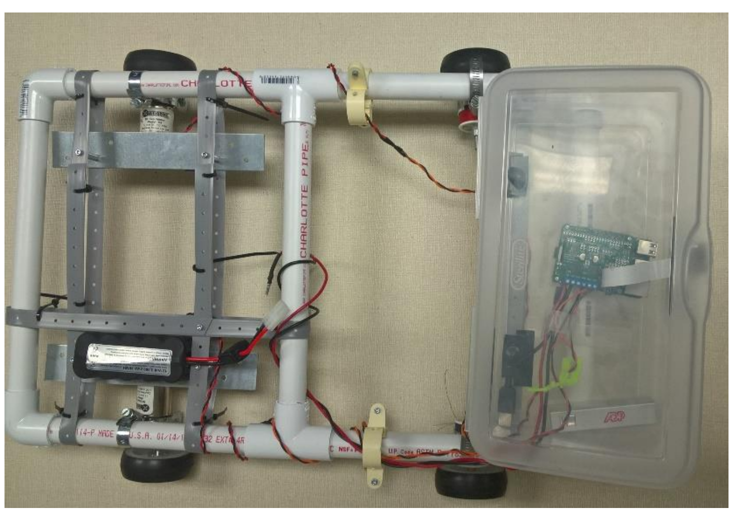

Figure B6: Top view of V2

Appendix C – V2 Speed Tests

V2 Speed Tests					
% speed, σ, SE	**time (s)**	**forward (cm)**	**reverse (cm)**	**forward 2 (cm)**	**average (cm/s)**
50%	2	61	59	62	30.3
σ = 1.126	4	131	117	128	31.3
SEM=0.375	6	189	182	185	30.9
60%	2	73.5	70	73.5	36.2
σ = 0.932	4	145.5	142	142	35.8
SEM=0.311	6	213	203	211	34.8
80%	2	96.5	94	96.5	47.8
σ = 1.892	4	194	187	195	48
SEM=0.631	6	310	300	313	51.3

Table C1: V2 Speed Tests