THE ILLINI WAGON

By

Ian Watson

Neha Joseph

Ramya Reddy

Final Report for ECE 445, Senior Design, Spring 2025

TA: John Li

May 2025

Project No. 39

Abstract

Transporting groceries and other items can be a significant challenge for individuals without access to a personal vehicle or reliable public transportation. Additionally, activities such as tailgating or picnicking often involve carrying heavy or bulky items over long distances. The Illini Wagon addresses these issues by providing a low-cost, autonomous transport solution. Utilizing Ultra-Wideband (UWB) transceivers, ESP32 microcontrollers, and supporting hardware, the Illini Wagon is capable of accurately tracking a user's location and following them with minimal user intervention. This system offers an affordable and efficient means of load carrying, enhancing convenience in everyday tasks.

Though our individual subsystems were all functional, we faced challenges when attempting to weave these systems together. We elaborate on the challenges we faced and potential solutions that may be considered if this project was to be continued further.

Contents

1. Introduction

Problem

Bringing in groceries can be challenging for the elderly, and hauling a cooler, chairs, and other gear to the beach can be a hassle. Carrying heavy packages from the car, transporting picnic or camping supplies, and lugging gear for outdoor events can quickly become exhausting. Parents juggling strollers, diaper bags, and toys during outings could use an extra hand, while festival-goers and tailgaters often struggle with chairs, food, and drinks. Hikers and campers could travel farther without the strain of carrying supplies, and warehouse or delivery workers would benefit from extra support moving boxes and equipment. Whether for daily tasks, outdoor adventures, or work-related needs, the Illini Wagon can take the load off, making life more convenient.

While similar products are available on the market today, they are often either too expensive, lack true autonomy, or both, making them less practical for everyday use. Many require manual operation or remote control, which still places a burden on the user. Others rely on computer vision which drives up costs and limits their accessibility. For people who need a simple, affordable, and truly hands-free solution, there are practically zero options available. The Illini Wagon will be both cost-effective and fully autonomous in order to fill this gap, providing a practical tool for a wide range of users.

Solution

Our proposed solution is an autonomous, self-following wagon designed to carry loads for users in both indoor and outdoor environments. The wagon will employ an Ultra-Wideband (UWB) tracking system to accurately follow the user without requiring direct manual control. This system will consist of a UWB Tag and Anchor setup, where two UWB tags will be mounted on the wagon, and the user will carry a remote with an embedded UWB anchor. By utilizing the Two-Way Ranging (TWR) method, the system will determine the user's real-time position and distance through Time of Flight (ToF) calculations.

To enhance navigation stability, the wagon will incorporate additional sensors such as IMUs (Inertial Measurement Units) for motion estimation and ultrasonic or LiDAR sensors to detect obstacles and prevent collisions. A combination of onboard processing and real-time feedback loops will ensure responsive and reliable tracking, even in dynamic environments. This solution provides a hands-free alternative to traditional rolling carts and backpacks, reducing physical strain while seamlessly adapting to urban and campus landscapes.

2 Design

2.1 Block Diagram

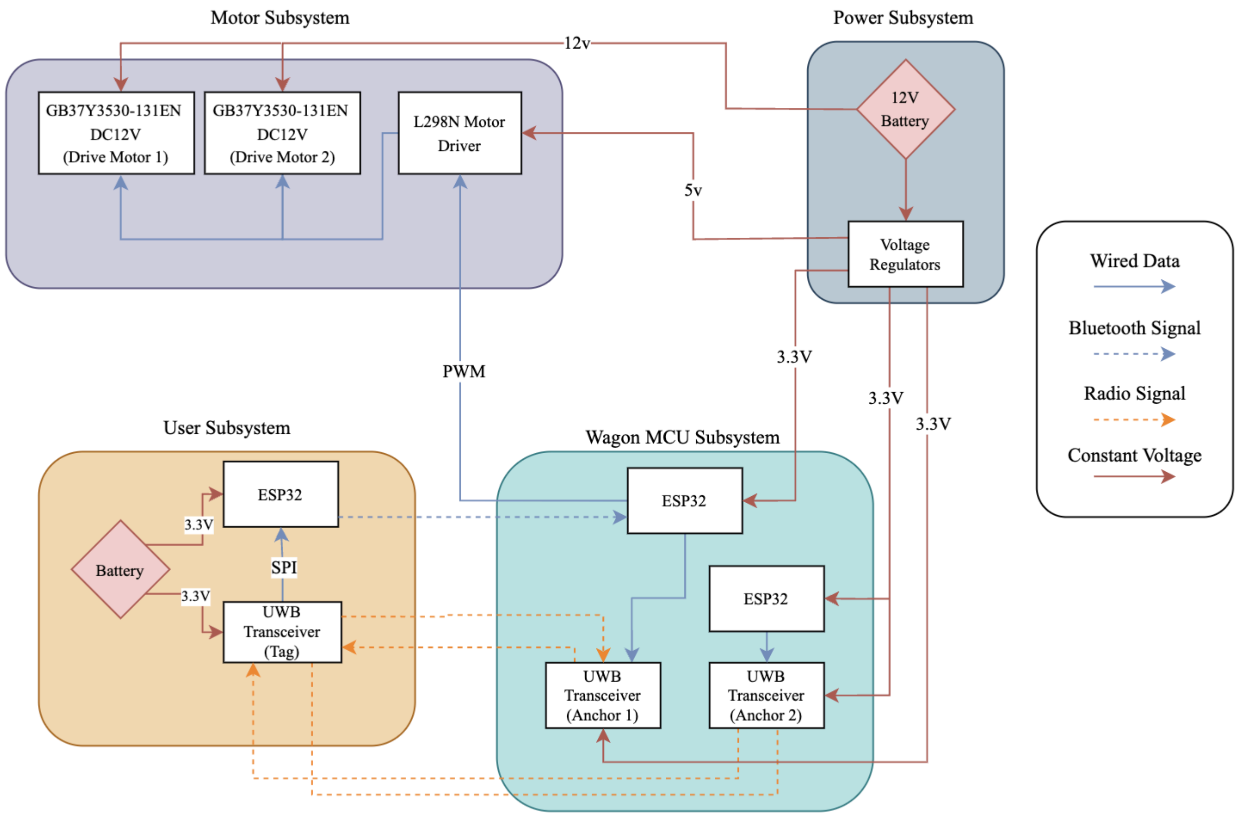


Figure 1: High Level Block Diagram

Our design has three main subsystems, each of these having its own power subsystem: motor subsystem, user subsystem, and wagon subsystem. The motor subsystem consists of two 12V DC motors, a motor driver, and an ESP32. The wagon subsystem consists of two UWB transceivers, both acting as anchors, 2 ESP32s, and respective power systems. Each UWB module has its own ESP32 module since they will be on separate PCBs.

The user subsystem consists of a UWB transceiver, which acts as the tag, an ESP32, and a 6 V power source, which is converted down to 3.3 V to supply power to each component. This system will be carried by the user in order for the wagon to follow the user. The tag essentially sends signals to the anchors on the robot via radio waves to calculate the respective distance to each anchor. This data is then sent to the ESP32, which will translate the timestamp information into distance and angle calculations. This information is then sent to the wagon subsystem ESP32 to undergo further processing. Here it is converted into PWM signals which are sent to the motor drivers, which then control the speed of each motor.

2.2 Power Subsystem

2.2.1 High Level Purpose and Description

Our design has three main power systems, each corresponding to its respective subsystem. The motor subsystem will be powered by a 12 V rechargeable battery. The wagon subsystem will be powered by four 1.5 V AA batteries connected in series, outputting 6 V in total. The user subsystem will also be powered by four 1.5 V AA batteries. We chose to have three separate power systems because each subsystem has its own PCB.

The motor power subsystem is responsible for supplying voltage to one ESP32, one UWB transceiver (DWM1000), the motor driver, two motors, and the USB connection module. To supply the correct voltage to each of these components, we have two systems for stepping down the 12 V source. We first have a LM2596SX-5.0/NOPB buck converter, that steps down 12 V to 5 V. We then have a linear voltage regulator, that steps down the 5 V power supply from the buck converter to 3.3 V. This will power the UWB transceiver, the ESP32, and the TB6612FNG motor driver.

The user and wagon power subsystems are identical. They are each responsible for supplying voltage to one ESP32 and one UWB transceiver. It will step down the 6V power source to 3.3V utilizing an AP2112K-3.3 fixed voltage converter.

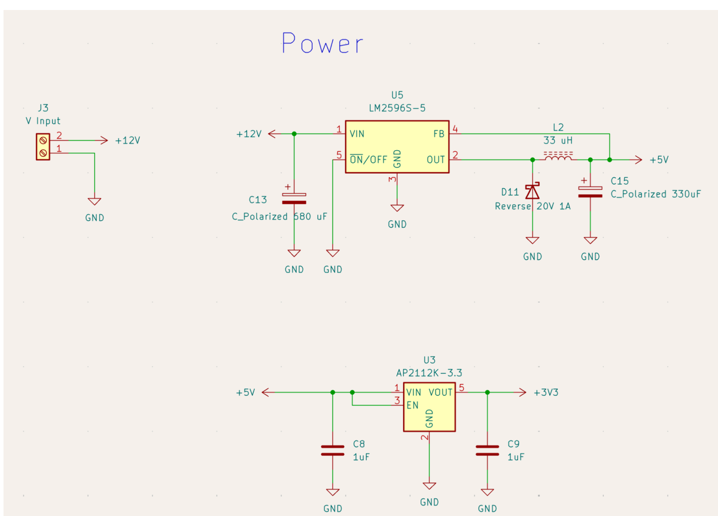


Figure 2: Motor Power Subsystem Schematic

2.2.2 Design Procedure and Details

Motor Subsystem Power Source:

For the motors in the motor subsystem, we see a minimum start voltage of 1 V, and an ideal running voltage of 12 V. Due to this operational range, we have decided to use a 12 V battery. We have specifically chosen the KBT 12 V 5200mAh battery (1C), as it is able to handle the max current draw of our components. It also has a max continuous discharge rate of 15.6 A (3C), which means it can handle the worst case stall current of 14A total for both motors. These calculations are detailed below.

Through documentation, we know the rated current of our motors is 0.5A. Based on documentation we have also found a max current draw of 40mA from the motor driver module, 352mA for the ESP32, and 160mA for the UWB transceiver. This results in a total current draw of $1.552A$.

$$I_{max} = 0.5A + 0.5A + 0.16A + 0.04A + 0.352A += 1.552A$$

Since the motors have a max stall current of 7A per motor, this results in a total stall current of 14A. Since we are not straining the motors with a hefty load, the motors may only experience a surge in current requirement upon start up. Assuming the motors experience a surge in current usage for a maximum time of 1 minute, we have calculated the max current draw upon start up to be:

$$Time = \frac{1\,min}{60\,min} = 0.0167\ hours$$
$$I_{drawn} = 14A * 0.0167\ hours = 0.2333Ah$$
$$5.2Ah - 0.2333Ah = 4.96667Ah\ remaining\ battery\ life$$

Based on this information and the max current draw calculated previously, we will have a battery runtime of:

$$Runtime = \frac{Battery\ Capacity\ (Ah)}{Current\ Draw\ (A)} = \frac{4.96667Ah}{1.552\ A} \approx 3.2\ hours \approx 192\ minutes$$

Since this wagon is a prototype model, a runtime between 45-60 minutes should suffice, and this battery would provide the system with enough power to fulfill that requirement.

User and Wagon Subsystem Power Source:

According to the Duracell datasheet, their Coppertop AA batteries have a voltage of 1.5 V and a total service hours time of approximately 3 hours when current is being drawn at a constant rate of 500 mA. This can be seen from the image taken from the data sheet below. Since both the Wagon and User systems require approximately 3.3 volts and max 0.515 A of current, the AA batteries will provide more than sufficient power for our goal runtime between 45-60 minutes.

Stepping Down Voltage:

In order to provide a stable 3.3 V power source to all the components on the motor subsystem, we had to consider how we wanted to step down 12 V to 3.3 V. Our first thought was to use a fixed linear voltage regulator to directly step down the 12 V to 3.3 V. After further research, we determined that the heat dissipation caused by this high of a voltage drop may be unsafe for other sensitive components on the PCB. To properly handle this high voltage drop, we decided to utilize a step-down buck converter, which generates minimal heat. Since buck converters are generally known to produce noise and provide a somewhat unstable voltage output, we decided against dropping directly from 12 V to 3.3 V using this method. Instead, we converted 12 V to 5 V, and used a fixed 3.3 V linear voltage regulator to drop the 5 V to 3.3 V.

The linear voltage regulator is able to provide a clean and stable 3.3 V output, which is what we need for components like the ESP32 and DWM1000 which require around 3.3 V continuously to operate in a stable manner.

For the user and wagon subsystems, we were able to directly step-down the 6 V power source to 3.3 V using the fixed 3.3 V voltage regulator. We determined that the heat dissipation from a 2.7 V drop would be acceptable for the components on these subsystems.

2.2.3 Design Alternatives

After running and testing our completed project, we have determined that it is unnecessary for the motor subsystem and wagon subsystem to have separate power subsystems. Since the subsystems would only be 18 in apart on the wagon, we could easily have a clean wired connection from the 3.3 V output from the motor subsystem to the wagon subsystem. The main thing we would have to change in the 3.3 V linear voltage regulator we use, since it only has a max current output of 600mA, which may be too small to supply current to two ESP32s, two UWB modules, and one motor driver. In a future design, we could easily find a voltage regulator with a higher output current rating. This way we limit our power sources to one 12 V battery, and one quad AA battery pack.

2.3 Motor Subsystem

2.3.1 High Level Purpose and Description

The Illini Wagon's motor subsystem will be responsible for controlling its speed and direction of movement. It features two motors that drive the front wheels. These motors are powered by a motor driver (TB6612), which receives signals from the microcontroller (ESP32). The microcontroller processes input from the wagon and user subsystems in order to update, in real time, the values of the PWM signals sent to each motor.

2.3.2 Design Procedure and Details

To determine the location of the user, the motor subsystem receives the distance between the user and each of the anchors. Using properties of side-side-side triangles, we can find the distance between the middle of the wagon to the user and each of the angles.

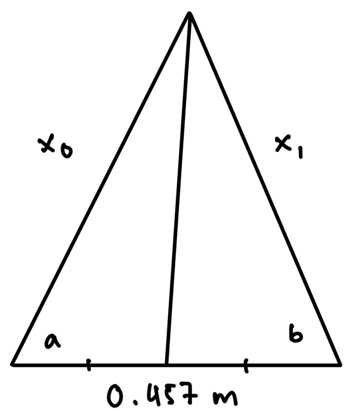


Figure 3: Motor Subsystem Illustration

To determine the rate at which motor spins, we first see if the distance between the wagon and the user is less than 1.5 meters. If the user is farther than 1.5 meters from the user, the duty cycle logic continues, otherwise the wagon stops. We then compare the values of angle a and angle b. If they are within .2 radians, the duty cycle of each motor will be identical and linearly scaled from 0 to 150 rpm. If angle a exceeds angle b by more than .2 radians, the value of the duty cycle sent to motor a will be reduced by 25%. The same logic applies to angle b.

2.3.3 Design Alternatives

<u>Smooth Motor Control:</u>

During the design process, we did not realize that two identical motors could still have differences in their rotation speeds, even at the same RPM and voltage settings. This was a problem because it was difficult to keep the motor movements consistent, and we resorted to manually testing different RPM value combinations to make the wheel movement as consistent as possible. However, the results were still not ideal. Additionally, when implementing turns and steering, we realized that we were turning the motors by a fixed amount, resulting in the wagon oscillating in its movements.

A better design choice would have been to incorporate motors with encoders and use the rotational position output for a PID controller to smooth out the motors' uneven movements. This would also help filter out noise in the movements and allow for better straight-line movement and turning performance.

2.4 User and Wagon Subsystem

2.4.1 High Level Purpose and Description

The user and wagon subsystems are responsible for communicating with each other to determine and calculate the user's absolute location. The user subsystem consists of a remote carried in the user's pocket, which includes a UWB transceiver and an ESP32 microcontroller. The wagon subsystem consists of two UWB transceivers mounted on either side of the wagon, along with two ESP32 microcontrollers that act as hosts for the UWB transceivers (one of the ESP32 microcontrollers is shared with the motor subsystem). The user and wagon subsystems use the UWB transceivers to implement the TWR (Two-Way Ranging) methodology to calculate the distance between the wagon and the user. TWR is performed using the ToF (Time of Flight) measurements from the UWB transceiver—acting as the tag—on the user subsystem to the UWB transceivers on the wagon subsystem, acting as the anchors.

The user subsystem has three main components: a DWM1000 UWB transceiver, an ESP32 module, and a mini power system supplying 3.3V as described in the power system above. The wagon subsystem comprises : two UWB anchors that communicate with the user's UWB tag, two ESP32 microcontrollers, and power sources that will supply 3.3V of power to each component as described in the power subsystem description.

The user UWB tag will communicate with both the UWB anchors mounted on the motors and the system will produce two distance measurements: the distance from anchor one to the tag and the distance from anchor two to the tag. The distance measured from anchor two will be sent to the ESP32 of anchor one. The two distances will then be used in a triangulation method to determine the motor movements through duty cycle calculations.

2.4.2 Design Procedure and Details

ESP32 As Host Device For DWM1000:

The ESP32 serve as a host device for the UWB transceivers, each DWM1000 module is connected to their respective ESP32 module via SPI communication protocols. The distance measured by the UWB transceivers are accessed by the ESP32 modules. On the wagon, anchor 2's ESP32 will then communicate with anchor 1's ESP32 in order to send the distance measurements to allow for the motor movement calculations.

UWB Distance Measuring Implementation:

The UWB transceivers we chose to use are the DWM1000 module, which can act as both the anchor or the tag in the communication system. The transceivers use ToF calculations and TWR to measure the distance between the tag and anchor. It then provides the two distances measured by each of the anchors to the motor subsystem needed to calculate the correct duty cycle.

TWR is a method to measure the ToF without needing clock synchronization between the two devices. The TWR process involves an initiator sending a poll message to the responder and the responder sending a response message back to the initiator.

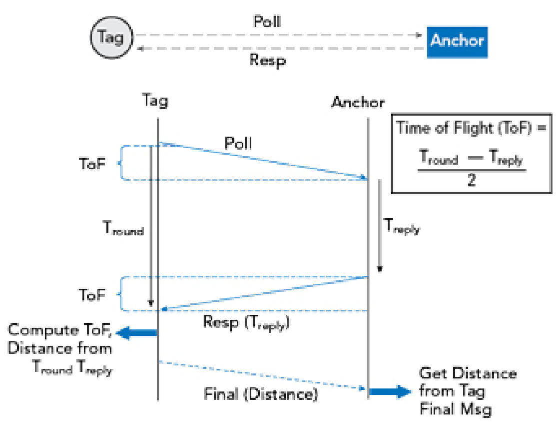


Figure 4: Single-Sided Two Way Ranging

The time stamp of the initial transmission and the response are recorded. The ToF can then be calculated as:

$$TOF = \frac{t_2 - t_1 - t_{reply}}{2}$$

Equation 1: Time of Flight Calculation

t_1: $time\ of\ initial\ transmission$
t_2: $timestamp\ of\ response\ receival$
t_{reply}: $delay\ from\ anchor\ transmitting\ response$

The ToF calculated from the TWR method is the time it takes for a signal to travel from the transmitter to the receiver. Since the signals are radio waves which travel at the speed of lights, the ToF can be converted into distance with the following formula:

$$distance\ = C * TOF$$

Equation 2: Distance Calculation

TOF: $Calculated\ time\ of\ flight\ value$
C: $Speed\ of\ light$

However, we soon realized that the distance measured by the UWB transceivers can be slightly inaccurate with an offset. This is a result of the propagation delay throughout the DWM1000 devices, they are also known as antenna delays. The antenna delays are internal to the chip and are included in the transmission timestamp and the receiving timestamp.

$$tMeasured\ = tADTX\ +\ ToF\ +\ tADRX$$

Equation 3: Distance Calculation

ToF: $Calculated\ time\ of\ flight\ value$
$tMeasured$: $The\ measured\ time\ from\ the\ transmit\ timestamp\ to\ the\ receive\ timestamp$
$tADTX$: $Transmit\ antenna\ delay$
$tADRX$: $Receive\ antenna\ delay$

To fix the error in the distance measurement, we included an antenna delay calibration each time the UWB anchors are powered on. They should be placed at a distance of 1 meter from the UWB tag, the UWB anchors will then try different antenna delay values until it measures a distance as close to 1 meter as possible. This significantly reduces the error in distance measurement in our project.

Another design choice to further reduce the error in distance measurement is the use of Kalman filtering. Because the UWB transceivers are susceptible to noise, we decided to use Kalman filtering to better improve our distance measurements. Kalman filtering uses the measurements observed, including noise and other inaccuracies, to produce estimates of what the results should be. After filtering the distance data, we took the average of 5 data points.

2.4.3 Design Alternatives

Communication Protocol Between Anchors:

In our original design, we had the wagon subsystem communicating its distance information to the motor subsystem using BLE (bluetooth low energy) communication. We originally chose this since our differing ESP32 dev boards both supported this communication and it had low power consumption due to the radio being turned off when not in use. It also supported the frequency of data transmission that we required between the anchors.

During our testing phase though, we realized that the ADC2 pins on the ESP32 cannot be used when BLE is in use. This severely limited how we were able to utilize the ESP32, since we needed a good number of GPIO pins for our motor driver and DWM1000 module. We also had issues with bad connectivity due to issues with our ESP32 and breadboard. It also utilizes a complex server/client implementation.

In future work, we would utilize I2C communication protocol between the two subsystems. This would still be supported by both development boards, and it only utilizes two ports on the ESP32 allowing for increased design flexibility. It is also quite simple to implement master and slave device communication. Since our subsystems are only 18 inches apart, having a wired communication protocol would be acceptable.

Microcontroller Limitations:

As mentioned above, the ESP32 boards have pin limitations when BLE is in use. We were not able to use one ESP32 module for Bluetooth communication, motor driver host, and UWB transceiver host. This is because the ESP32 microcontroller is a slightly older microcontroller with many shared stacks and less I/O flexibility. One change we would make in the design is to use a different microcontroller. A better example would be the ESP32-S3 microcontroller; it is newer, with more peripheral flexibility, faster CPU processing, more memory, and more ports.

3. Design Verification

3.1 Power Subsystem

Requirements	Verification
1. Provide a stable voltage of 3.3 V to the ESP32 and DWM1000.	● Measure output voltage from 5 V to 3.3 V voltage regulator using an oscilloscope to ensure it is providing a steady voltage of 3.3 V$\pm$.165 V.
2. Ensure the system can handle the current and voltage requirements of all components for between 45-60 minutes.	● Using an alternate power supply (non-battery), we will measure the current draw using a multimeter for each large component. ● We will then ensure that the total current draw from the system fits within 2A, to verify the validity of the chosen 12 V 5200mAh battery supply.

In Appendix A.1, we can see that requirement 1 is satisfied. We were able to get a stable and continuous reading of approximately 3.14 V, which falls within the 3.3 V $\pm$.165 V range. For requirement 2, we ensured that the system was functional by allowing our machine to run for 45 minutes during testing. The battery was able to sustain this usage, and did not need to be recharged. We also measured the current drawn using a benchtop voltage source, and found it to be within our calculated expectations (as listed in the power subsystem section).

3.2 Motor Subsystem

Requirement	Verification
RPM of drive motors is proportional to distance from user and falls within 0-200 RPM	● Turn on wagon, then walk away from wagon ● Measure RPM while the user walks away and ensure it increases with distance, and caps out at 200 RPM.
Wagon points to user within 0.15 radians within .5 seconds	● Ensure that each when the user is to the right of the wagon, the left wheel has a higher duty cycle, and vice versa. ● Measure this using an oscilloscope. ● Test the duty cycle values by having wagon operate in controlled environment

Wagon is able to maintain constant distance of 1.3 to 1.7 meters, assuming walking speed	● Bring the wagon to a controlled area, turn on the wagon, and have the user stand close to the wagon to ensure it does not move. ● Have the user walk away slowly and ensure that the wagon is able to track the user within 1.3 to 1.7 meters.
Current supplied to the motor will be between .4A to 7A.	● Have the user within stop distance of the wagon and ensure that the duty cycle is 0, and confirm the wheels are not spinning. ● Measure current with current sensor and ensure that it is within 300-400 mA. ● Have user stand far away from wagon, and ensure that the duty cycle is at its maximum ● Measure current again and ensure the value is less than 7A.

In Appendix B, we provide a picture of the first two requirements being fulfilled. As the user moves away, the RPM of the motors increases. When the user moves to the left and right, the RPMs of each motor correspond to the direction of the user. It is difficult to show these verifications in images, but we provide an image of the wheels spinning to demonstrate that the behavior is correct.

3.3 User and Wagon Subsystem

Requirements	Verification
1. Transmit data back to the wagon MCU unit on the robot	● We will ensure that we write our code in a way that enables us to see the data that is transmitted as packets back to the wagon system via on screen/on terminal display.
2. Calculate TOF and distance information through data stored on UWB tag	● We should be able to see the printed values for time stamps/distance on a pop up window or on the terminal via source code. ● We will measure the accuracy of these calculations by comparing the physically measured distance to the calculated distance.
3. The two UWB anchors should be able to send	● Since we are able to display the distance values on the screen, we can also see what information is

and receive signals from the user's UWB tag	being transmitted between the UWB modules. ● We can determine the accuracy after computing the distance.

As seen in Appendix C.1, you can visually see the transmitted and received data via the serial monitor. The left screen shows data produced by anchor 1 and received by anchor 2. The right screen shows data produced by anchor 2 and transmitted to anchor 1. This satisfies the verification criteria for requirements 1-3.

As seen in Appendix C.2, we have compiled distance readings between one UWB anchor and one UWB tag. These distance readings were gathered after calibrating the anchor delay, and doing data filtering using the Kalman filter. For requirement 2, we wanted to see the accuracy of our UWB distance calculations compared to the physically measured distance. We were able to verify that a distance of 1.475 m had a measurement error of 0.255%, a distance of 1 m had a measurement error of 2.43%, and a distance of .75 m had an error of 20.97%. Since our wagon is meant to stop once it reaches a distance of 1.5 m, we are able to neglect the large margin of error that comes with reading distances less than 1 m. The other margins of error fell within the expected range of error, and we were satisfied with the results.

4. Costs

4.1 Parts

From the table listed in Appendix D.1, our total comes out to roughly $199.56.

Many of the smaller PCB components such as resistors and capacitors have not been listed in the final cost, as they are minimal cost components that have been provided to us through the ECE department. Some parts such as the motors, ESP32s, etc. have also been provided to us through the ECE department, but the cost has been listed to give a more accurate total price estimate.

4.2 Labor

We have taken the average starting salary of both Electrical and Computer Engineers at the University of Illinois at Urbana-Champaign.

$$Hourly\ Wage\ =\ \$98472.5/52\ weeks/5\ days/8\ hours\ =\ \$47.3425481$$

Assuming our wage is $47.34/hr and that we each spend 200 hours on this project, the total compensation is calculated as $\$47.34\ *\ 3\ employees\ *\ 200\ hours\ =\ \$28404$

Given the labor costs and the part costs listed in the previous section, we have a total project cost of $28603.56.

5. Conclusion

Overall, this project gave us valuable hands-on experience in PCB design, strengthened our research skills, and deepened our understanding of how various hardware components integrate and function as a cohesive system.

5.1 Accomplishments

One of our main accomplishments is our absolute distance measurement system as it works within 0.26-2.43% measurement error. Additionally, when provided artificial distances, the motor subsystem was able to steer in the proper direction by changing the speed of one or both the wheels. Lastly, our power subsystem was able to supply 12V and 3.3V of continuous power to the different subsystems.

5.2 Uncertainties

One of the biggest uncertainties and shortcomings of this project was that we were using faulty breadboards throughout the entire development process. Our design consistently failed, and after extensive testing of each component in each subsystem, we determined that the issue was caused by the mini breadboards we were using. The suspected cause was a lack of bridging between vertical rows due to faulty dividers, as well as poor continuity within rows due to defective connections.

This issue resulted in damage to several of our ESP32 development boards, as well as two DWM1000 UWB transceivers. We overcame this challenge by moving our systems onto larger breadboards, after which the functionalities worked as expected. However, because we were left with only two functional pairs of ESP32 and DWM1000 modules, we were only able to simulate and demonstrate the right anchor and right wheel functioning correctly.

5.3 Ethical considerations

Our primary demographic for this project is college students, hence the name 'Illini Wagon.' However, this product can also benefit other groups, including the elderly, individuals with mobility impairments, and families. Although our intention for this project is to create a product that makes carrying things easier, we must acknowledge the safety risks that come with autonomous movement, such as potential collisions or malfunctions.

Ensuring that this product is safe and ethical is one of our top priorities, as it will likely operate autonomously in public spaces. The ethical concerns of this project include ensuring the safety of the user, as the self-driving wagon could pose a hazard to both users and bystanders. Additionally, poor obstacle detection or control loop failures could result in collisions or injuries.

The Illini Wagon's navigation must be highly reliable, using sensors and fail-safe mechanisms to prevent collisions. Safety considerations include redundant braking systems, obstacle detection, and emergency stop features to prevent accidents. We will consult the US Department of Transportation's Automated Vehicles Comprehensive Plan[4] in order to make sure that our vehicle complies with this handbook's standards.

5.4 Future Work

As mentioned above, future work on this project includes replacing the current motors with encoder-equipped motors and implementing a PID control loop to improve movement precision. Additionally, the damaged ESP32 and DWM1000 modules will need to be replaced. Potential feature enhancements include integrating a computer vision component for more robust obstacle detection and adding a braking system to enable emergency stops and prevent accidents.

References

[1] Toshiba Corporation, "TB6612FNG – Driver IC for Dual DC Motor," Datasheet, Oct. 2014. [Online]. Available: https://toshiba.semicon-storage.com/us/semiconductor/product/motor-driver-ics/brushed-dc-motor-driver-ics/detail.TB6612FNG.html [Accessed: Feb 26, 2025]

[2] Qorvo, "Getting Back to Basics with Ultra-Wideband (UWB)," Available: https://www.qorvo.com/resources/d/qorvo-getting-back-to-basics-with-ultra-wideband-uwb-white-paper [Accessed: Jan. 30, 2025]

[3] Qorvo, "DWM100 Datasheet," Available: https://www.qorvo.com/products/d/da007948 [Accessed Feb. 28, 2025]

[4] U.S. Department of Transportation, "USDOT Automated Vehicles Activities," US Department of Transportation. Available: https://www.transportation.gov/AV [Accessed Feb. 13, 2025]

[5] Espressif Systems, "ESP32-S3-WROOM-1 WROOM-1U Datasheet," Available: https://www.espressif.com/sites/default/files/documentation/esp32-s3-wroom-1_wroom-1u_datasheet_en.pdf [Accessed Feb. 10th, 2025]

[6] Qorvo, "APS013 APPLICATION NOTE The implementation of two-way ranging with the DW1000, " Available: https://www.qorvo.com/products/d/da008448 [Accessed Feb. 10th, 2025]

[9] DFRobot. (n.d.). *Metal DC geared motor w/encoder - 12V 251rpm 18kg.CM Wiki*. DFRobot. Available: https://wiki.dfrobot.com/12V_DC_Motor_251rpm_w_Encoder__SKU__FIT0186_ [Accessed: Feb. 12, 2025]

[11] Diodes Incorporated, "600mA CMOS LDO REGULATOR WITH ENABLE," AP2112 Data Sheet, [Online]. Available: https://www.mouser.com/datasheet/2/115/AP2112-271550.pdf?srsltid=AfmBOoozxiRdWX9KGtM1-j-q3ANoxcZ6SNJQxErg-MNQbqkQ-12IHqVc [Accessed: Feb. 27, 2025]

[12] Duracell, "MN1500 Size: AA Alkaline-Manganese Dioxide Battery," Data Sheet, [Online]. Available: https://www.duracell.com/wp-content/uploads/2016/03/MN15USCT0122-A.pdf [Accessed: Mar. 6, 2025]

[13] Qorvo, “DW1000 Datasheet,” Data Sheet, [Online]. Available: https://www.qorvo.com/products/d/da007946 [Accessed: Jan. 28, 2025]

[14] Qorvo, “DW1000 User Manual,” User Manual, [Online]. Available: https://www.qorvo.com/products/d/da007967 [Accessed: Jan. 30, 2025]

[16] Markerfabs, “ESP32 UWB/UWB Pro,” [Online]. Available: https://wiki.makerfabs.com/ESP32_UWB.html [Accessed: Jan. 30, 2025]

[17] Makerfabs, (2023, October 12), “Makerfabs-ESP32-UWB/hardware,” ESP32 UWB .sch file, [Online]. Available: https://github.com/Makerfabs/Makerfabs-ESP32-UWB/tree/main/hardware [Accessed: Feb 23rd, 2025]

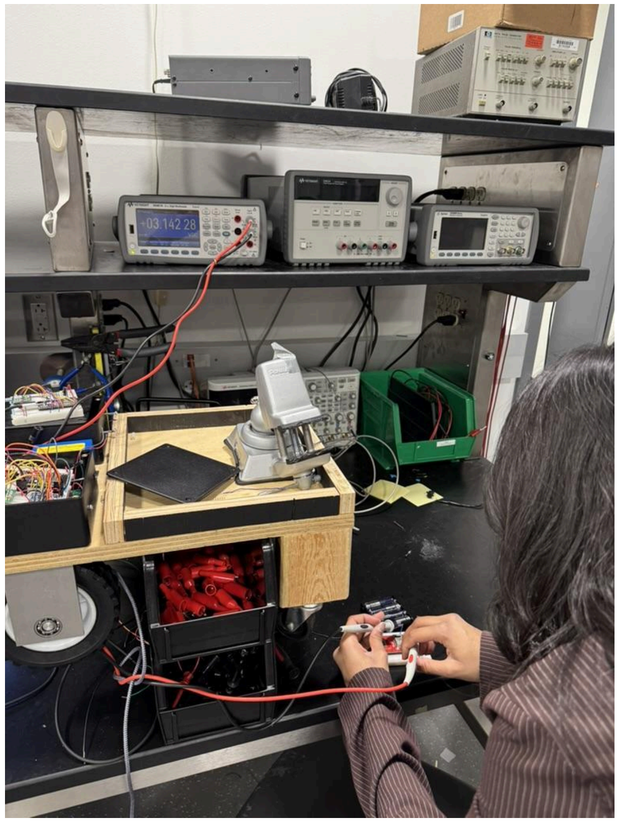
[18] IEEE. ”IEEE Code of Ethics.” (2024), [Online]. Available: https://www.ieee.org/%20about/corporate/governance/p7-8.html [Accessed: Feb. 28th, 2025]

[19] Afaneh, (2022, September 6), “Bluetooth Low Energy (BLE): A Complete Guide,” [Online]. Available: https://novelbits.io/bluetooth-low-energy-ble-complete-guide/

[20]Admin, “Getting Started with ESP32 UWB (Ultra Wideband) Module,” *How To Electronics*, May 31, 2022. [Accessed: March 10th, 2025] https://how2electronics.com/getting-started-with-esp32-dw1000-uwb-ultra-wideband-module/

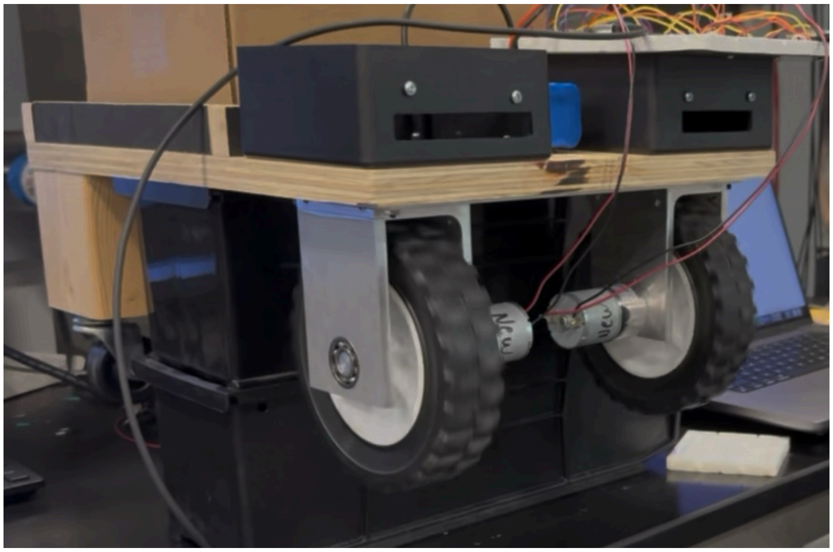
Appendix A: Power Subsystem

A.1 Image Demonstrating Stable Voltage of 3.3 ± 0.16



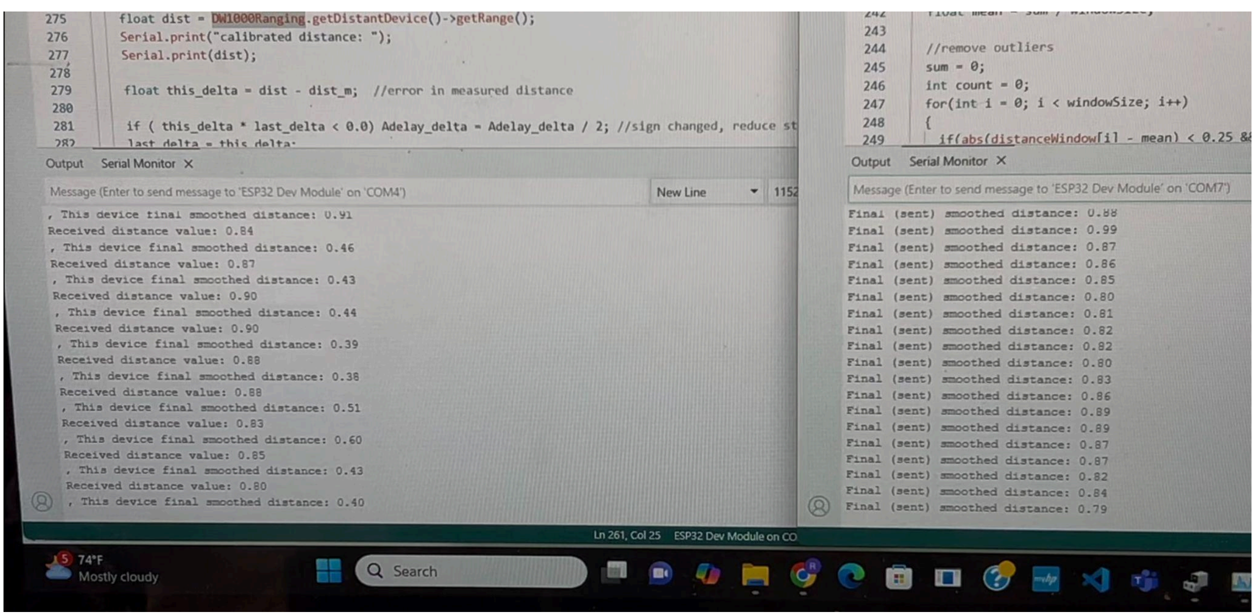
Appendix B: Motor Subsystem

In this photo, the user is placed about 2 meters away from the wagon. The wheels spin proportionally to the distance from the user.



Appendix C: User and Wagon Subsystem

C.1 Visual Display of Data Produced and Received by UWB Anchors



C.2 UWB Distance Readings and Error Calculations

Actual Distance 1.475 m	UWB Measurement	Actual Distance 1 m	UWB Measurement	Actual Distance 0.75 m	UWB Measurement
	1.47		1.01		0.62
	1.49		0.98		0.61
	1.49		0.95		0.6
	1.49		0.98		0.65
	1.48		0.99		0.68
	1.47		0.93		0.54
	1.45		0.99		0.64
	1.44		0.93		0.56
	1.45		0.99		0.63
	1.48		0.98		0.63
	1.5		0.97		0.62
	1.52		1		0.68
	1.5		0.97		0.67
	1.46		0.97		0.61
	1.47		0.99		0.63
	1.49		0.99		0.6
	1.45		0.96		0.63
	1.47		0.94		0.61
	1.51		1		0.63
	1.45		1.01		0.61
	1.49		0.98		0.68
	1.37		0.91		0.64
	1.51		0.99		0.61
	1.47		1.01		0.61
	1.48		0.97		0.61
	1.47		0.99		0.58
	1.47		0.99		0.6
	1.38		0.96		0.57
	1.45		0.97		0.59

	1.5		0.97		0.69
	1.49		1		0.67
	1.47		0.97		0.54
Average(s):	1.47125		0.97625		0.62
Real Distance in m	1.475		1		0.75
ERROR %:	0.2549%		2.4328%		20.9677%
ERROR (measured - real):	-0.00375		-0.02375		-0.13

Appendix D: Other

D.1 Parts List and Costs

Part Number/Description	Manufacturer	Quantity	Unit Price	Total Price	Link
DWM1000	Qorvo	3	$25.00	$70.86	Link
ESP32-S3-Wroom-1	Espressif	3	$6.56	$19.68	Link
12V Metal DC Geared Motor	Greartisan	2	$14.99	$29.98	Link
12V 5200mAh Rechargable Battery	KBT	1	$32.99	$32.99	Link
4x AA batteries	Duracell	2	$4.99	$9.98	Link
AA battery holder	Jameco Electronics	2	$3.04	$6.08	Link
AP2112K-3.3 (voltage regulator)	Diodes Incorporated	3	$0.52	$1.56	Link
LM2596SX-5.0/NOPB (buck converter)	Texas Instruments	1	$5.76	$5.76	Link
680 uF electrolytic capacitor 667-EEE-FT1V681UP	Panasonic	1	$1.53	$1.53	Link
330 uF electrolytic capacitor 594-MAL215099003E3	Vishay/BC Components	1	$2.62	$2.62	Link
33 uH Inductor 652-SRP1050WA-330M	Bourns	1	$1.19	$1.19	Link
22.1 kOhm resistor 755-SFR01MZPF2212	ROHM Semiconductor	3	$0.15	$0.45	Link
47.5 kOhm resistor 660-RK73H2BTTDD4752F	KOA Speer	3	$0.13	$0.39	Link
USB to UART chip CP2102N-A02-GQFN28	Silicon Labs	3	$3.53	$10.59	Link
ESD protection diodes SP0503BAHTG	Littlefuse Inc.	3	$0.63	$1.89	Link

BJT NPN ASS8050-L-HF	Comchip Technology	6	$0.39	$2.34	Link
TB6612FNG Motor Driver	Toshiba	1	$1.67	$1.67	Link