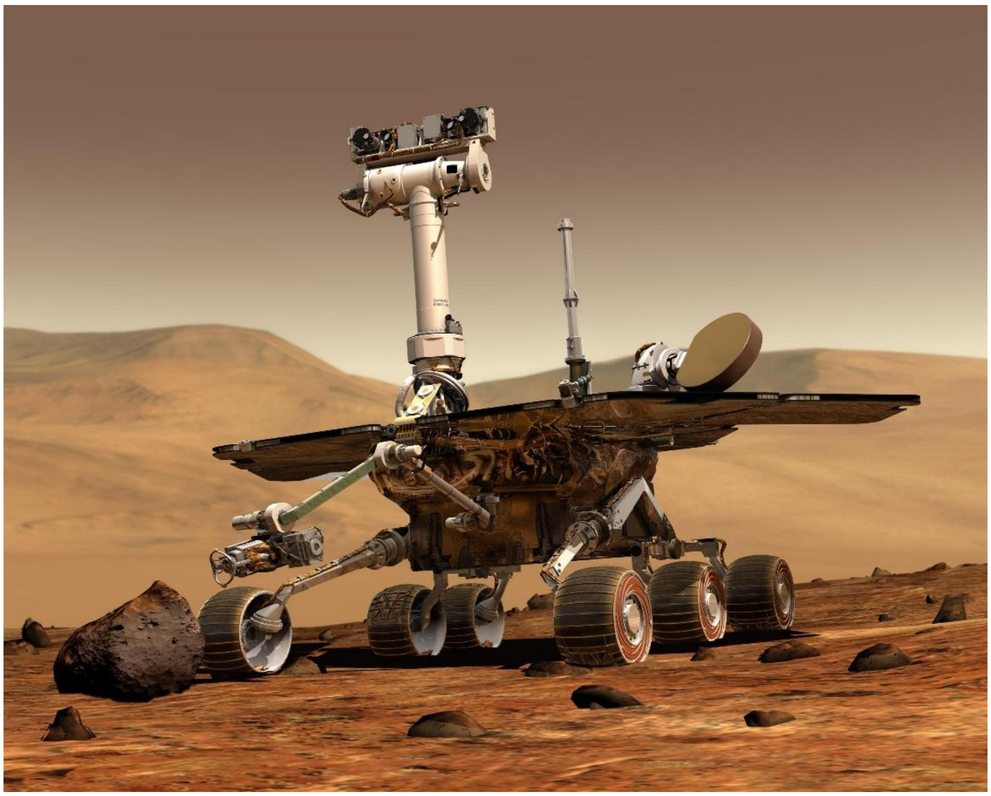
SoilSpirit

Autonomous No-dig Farming Rover



Appendix A

Creator:

Damon Rocha, Robotics and Embedded Systems

Last Updated: 2/18/2025

Technical Field

This project focuses on the following technical fields: robotics, embedded systems, autonomous systems, agriculture, and computer vision.

Background Information

No dig farming is a method of growing food where the soil is undisturbed. Instead of digging into the ground, compost is spread evenly on top of the soil and the seeds are directly put into this layer of material. It leads to healthy plants and soil, reduces weeds, and allows for better water conditions for the plants (Dowding, 2024). SoilSpirit plans to reduce the physical strain on small farms and gardeners through the automation of compost and seed spreading. While also having a small enough footprint that it does not compact the soil or cause unwanted tilling/digging of the compost. It takes inspiration from simple autonomous vehicles for obstacle avoidance and the Mars rovers (Spirit, Curiosity, Perseverance) for its rocker-bogie suspension system.

Prior Art (legal term)

The design of the product was heavily influenced by the Mars rovers and the 3d models of the initial prototype come from the Instructables tutorial in **AppendixC**. The suspension system of this rover comes from a direct iteration of the Rocker Bogie Mechanism popularized by the Mars rovers Curiosity and Perseverance. As for similar products, the closest to my idea is an autonomous spreader built by Raven Industries. It is an industrial vehicle that is autonomous. It is great for automation for large scale farms but would not benefit gardeners, homesteaders, and small-scale farms wanting to use no dig gardening. Another similar product that is of the same form factor/user is FarmBot. A CNC based garden bed that automates the process of growing food entirely. This company will be a direct competitor with the main difference being our modes of distribution. Their machine is directly attached to a garden bed whereas mine can drive around and be used for different types of gardens, farms, and homesteads.

Project Description

The aim of the project is to develop an autonomous rover capable of spreading fertilizer and seeds over a garden while avoiding obstacles. It should be able to:

- Drive on uneven surfaces
- Pull ~1lb of fertilizer
- Operate on 6V
- Evenly spread materials
- Autonomously avoid obstacles using camera and ultrasonic sensor
- Move ~ 2mph while not hauling spreader

It will consist of the following components:

1. Chassis to house electronics
2. 2 L298N motor driver
3. 4 SG90 Servo Motors
4. 6 DC Motors
5. Teensy4.1 Microcontroller
6. ESP32 XAIO S3 camera
7. 3.7V Lip rechargeable battery
8. Object detection model
9. Hopper trailer with spring-based trap door to spread fertilizer/seeds

To demonstrate the system's capabilities, I plan on first running this on a mock test area in my garage that will consist of fake grass and various obstacles. Then I want to move on and try it with freshly cut grass in my backyard and get it to spread fertilizer until I have issued a stop command.

Innovation Claim

SoilSpirit will be the first adaptation of Mars rover rocker-bogie suspension for agriculture and will make automation accessible to more people. On top of that it promotes a sustainable agricultural practice in making non-dig farming easier. Lastly, the incorporation of computer vision in a non-experimental realm will lead to this device being more robust and intelligent than most small-scale autonomous systems.

Usage Scenario

The primary demographic of this product encompasses gardeners, homesteaders, and small farm owners who are interested in no-dig farming or already practice it. However, this system could easily be extended including applications across industry. In aerospace, this may very well become integrated with future iterations of rovers and space vehicles. Technology inspired by SoilSpirit could possibly be used for farming on other planets, if we ever get there, or as small-scale versions of the Mars rovers that are more accessible for smaller companies. SoilSpirit could also extend into disaster recovery and reconnaissance. A scaled-up version of this rover could be used for search and rescue, area mapping, and debris clearing. Additionally, this project would be valued by the education and start-up industry due to it being a starting point for various projects.

Evaluation Criteria

The following questions will identify the successful completion of the project:

1. Core System Functionality:
    a. Can the rover smoothly drive without user intervention?
    b. Can the rover move at least 2MPH while not hauling the hopper?
    c. Does the hopper start/stop spreading based on movement of the rover?
    d. Does the rover successfully detect obstacles and either avoid or stop before hitting them?
    e. Can the rover drive on uneven surfaces?
2. Motor control and turning:
    a. Can the motors drive the rover forward and backwards?
    b. Can the servos cause the rover to turn left/right?
    c. Is there enough charge to run the motors and the servos to allow for spreading to finish?
3. Object Detection:
    a. Does the camera and model detect objects in front of the rover?
    b. Does the camera and model detect objects to the left/right peripherals of the rover?
    c. Do the emergency stop ultrasonic sensors detect obstacles missed by the camera?
    d. Does a detection lead to the rover stopping?
    e. Can the rover go around the object?
4. Documentation:
    a. Is the system and setup documented for ease of use and quickly getting started?
    b. Are limitations correctly represented by the documents?
    c. Is there a troubleshooting guide?

Objectives and Tasks Associated with the Project

High-level Objectives

1. Development of Motor control and rocker-bogie suspension system
    a. Create 2 rocker-bogie suspension systems
    b. Connect to microcontroller to test control
    c. Test DC motors, servos, and full suspensions systems individually then as a system

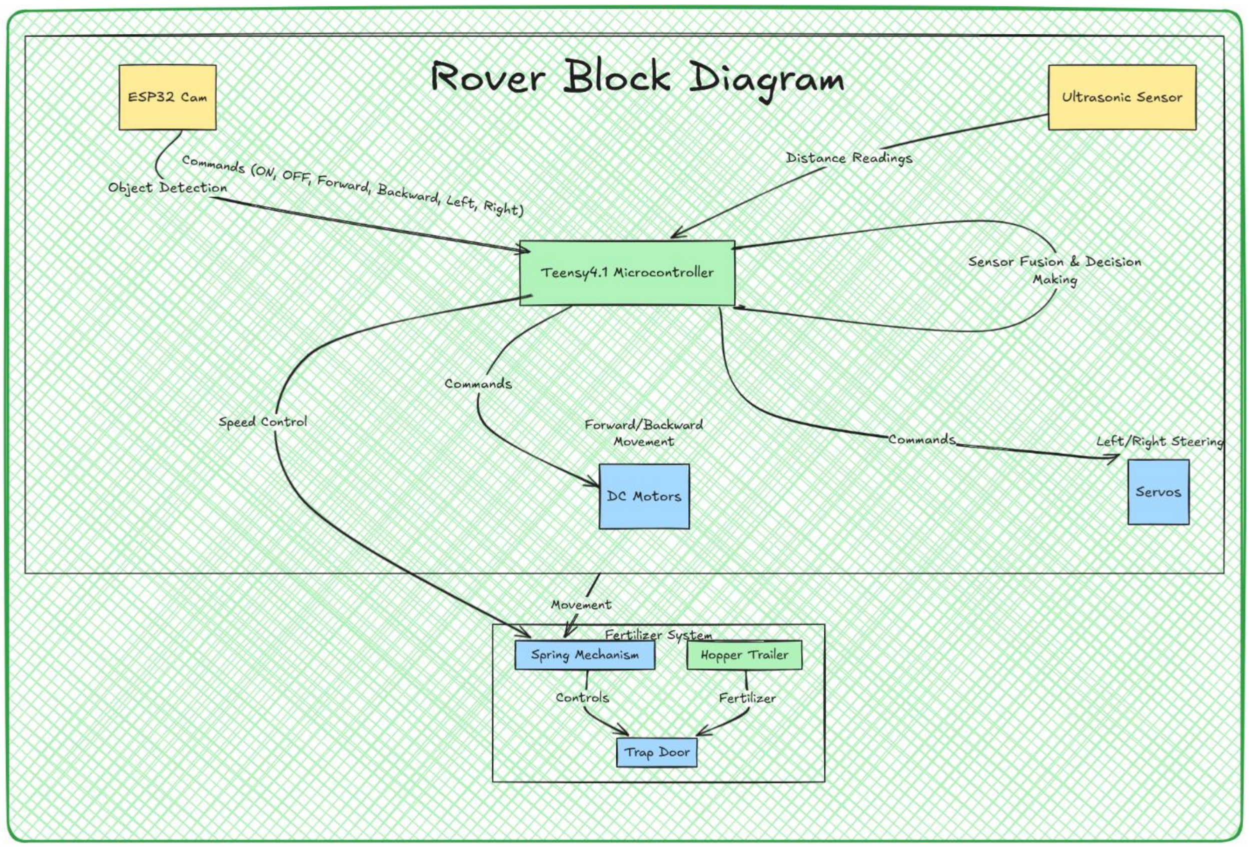
    d. Setup interface to drive motors based on shared commands from ESP32 S3 Camera
2. Obstacle detection model created and integrated with ESP32 S3 Camera
    a. Develop object detection with ESP32 S3 Camera
    b. Send detection results from ESP32 to Teensy4.1 micro controller over USART comms
    c. Test detection of different objects, conditions, and distances
3. Rover control and drive system
    a. Integrate object detection into motor control system
        i. ON/OFF/start/stop/left/right commands from motor control system interface
    b. Setup start/stop commands from ESP32 camera that a user can send to start autonomous control
    c. Setup ultrasonic sensors as an emergency stop
4. Sensor fusion
    a. Create sensor fusion between ESP32 S3 Camera output and ultrasonic sensors.
5. Spread of material
    a. 3D Model Hopper Trailer
    b. Slice and print hopper trailer parts
    c. Setup trap door for material spreading
    d. Connect to rover
6. Test full system integration

Detailed Task Breakdown

To start I plan on building out the motor control and object detection in tandem while I am waiting for a few final parts for the drive system. I plan on building up the object detection of the ESP32 cam using the FOMO(Faster Objects, More Objects) Object detection from Edge Impulse. If I run into issue with the AI model or run into walls with performance I will revert to classic computer vision algorithms and do some feature extractions for less precise but still working object detection. I want to first test these systems by themselves and integrate them together with the Teensy4.1. Once I have this to a working state, I plan on moving to communication between the boards using USART while modeling the hopper trailer. I am going to build a basic communication interface that consists of basic commands dependent on object detection results and where they are in the image. I will do various system tests and integration tests to ensure everything is working individually and together. Afterwards, I plan on integrating the ultra-sonic sensors for emergency stops and finishing up the design/build of the rover chassis. Once I finish these parts I will test out the rover on its own and attach the 3 tow hitches to connect to the hopper trailer. Then finally finish building the hopper and test out full system integration.

Description of Design Prototype

The main devices of this project will consist of the ARM Teensy4.1 microcontroller connected through USART to a ESP32 S3 Camera module. The Teensy4.1 microcontroller will use 2 L298N Motor Drivers to power the DC motors that will drive the wheels. For turning 4 servo motors are mounted on the front and back wheels which will be connected to the Teensy4.1 powered using pulse width modulation on Left/Right commands. The DC motors will be powered ON/OFF/Forward/Backwards depending on commands from the ESP32 Camera. As a backup object detection system there will also be an ultrasonic sensor used for emergency object detection in case that the camera goes offline or misses something. As stated earlier the mechanical design of the rovers rocker-bogie suspension system was predesigned but I only plan on using the suspension and will be using a custom chassis and slightly different electronics.
As for the Hopper trailer, this will be 3D printed and contain no electronics. It will consist of a hopper module that feeds the fertilizer/seed to a trap door which is opened/closed based on a spring which will be connected to the rover so that upon it moving the trap door will open/close.



Appendix B

Evaluation Plan

This project will go through testing and evaluation across many different levels. Each system will be broken up into modular parts and tested independently (unit tests) and in junction with other systems (integration tests). Each section of the evaluation criteria represents one system and will be tested as such. So for example, the motor control of each wheel module will be tested independently and then together, then as a whole for the rover, then also integrated as part of the full autonomous system. On top of this the requirements I stated at the beginning of this brief either need to be met or show areas that need improvement. For clarity, here those are again:

- Drive on uneven surfaces
- Pull ~1lb of fertilizer
- Operate on 6V
- Evenly spread materials
- Autonomously avoid obstacles using camera and ultrasonic sensor
- Move ~ 2mph while not hauling spreader

Project Completion Assessment

Note: This section must completed prior to SIP403/409.

Provide an in-depth description of the completion assessment of your project. Describe how well the completed components function and highlight the innovative facets of your design. This is sometimes known as a "Post-Mortem" or "Lessons-Learned Report". A good approach for this section is to answer the following 4 questions: "What went right? What went wrong? What was learned throughout the process? What would be done differently if you had to do it again?

Appendices

Appendix A: Picture of Spirit Mars Rover – SpiritRover.jpeg

Appendix B: Rover Block diagram – RoverBlockDiagram.png

Appendix C: References

Dowding, C. (n.d.). *Beginner's guide to no dig gardening*. Charles Dowding. Retrieved February 18, 2025, from https://www.charlesdowding.co.uk/resources/beginners-guide

FarmBot. (n.d.). FarmBot: Open-source CNC farming. Retrieved February 18, 2025, from http://farm.bot/?srsltid=AfmBOopHmAScrAOu6x2bxmTIahYDvzZ45t3wZjTq5M9lsXgofhvaoTWa

Instructables. (n.d.). *Arduino-based Perseverance rover model (Fusion 360)*. Instructables. Retrieved [date], from https://www.instructables.com/Arduino-Based-Perserverance-Rover-Model-Fusion-360/

Raven Industries. (n.d.). Case IH Trident 5550 applicator with Raven autonomy. Retrieved February 18, 2025, from https://www.ravenind.com/case-ih-trident-5550-applicator-with-raven-autonomy

Shekhar, C., & Singh, S. (2017). Design of rocker-bogie mechanism. International Journal of Innovative Science and Research Technology. Retrieved from https://ijisrt.com/wp-content/uploads/2017/05/Design-of-Rocker-Bogie-Mechanism-1.pdf