**All-Terrain Sampling Rover Project**

1. **Define mission requirements**
   1. What is the goal?

The goal of this project is to create a rover-like robot that has similar functionality to real-life rovers used today, albeit on a much simpler scale. I plan to add to this project over time, learning about advanced features related to computer vision and robot navigation eventually.

* 1. What are the operational constraints?
     1. Mechanical:
        1. Collect 10mL sample and store
        2. Traversable across rocky/uneven terrain
     2. Electrical:
        1. Minimum runtime of 15 minutes
        2. Wireless operation
        3. Solar-rechargeable battery
     3. Software:
        1. Autonomous control – Finite State Machine
        2. Sensor feedback (IMU, Camera, Ultrasonic Sensor, Wheel Encoders)
        3. Decision-making based off I/O
        4. GPS-Tracking

1. **Subsystem breakdown**
   1. Chassis
      1. Purpose: Houses majority of electronics and hardware, provides protection against outside elements
      2. Function: Shield electronics, Maintain upright position, Hold inner contents securely.
      3. Components: Frame, housings, internal/external mounts
   2. Drivetrain
      1. Purpose: High-torque system that allows for movement on steep inclines.
      2. Function: Forwards/backwards movement, differential drive, steering capability
      3. Components: 4x Wheel motors (BLDC), High-grip wheels, leg assemblies, wire housing
   3. Power System
      1. Purpose: Providing power to all components that require it
      2. Function: Provides stable voltage, current protection, battery management, stepping down voltage for logic devices, turning robot operations on and off
      3. Components: Battery pack, On/off switch, 12V?-5V buck converter, Fuse/current limiter, Ground Bus, Microcontroller Breakout Board, PCB Board
   4. Control
      1. Purpose: Managing IO and coordinating robot functions
      2. Function: Generating PWM signals, processing remote control inputs, managing sensor data, using logic to make decisions.
      3. Components: Main Controller (Raspberry Pi 3 – Arduino Mega doesn’t have camera integration for vision), Drive ESCs, Articulated Arm Controller (Pi Pico)
   5. Sensing
      1. Purpose: Using sensing devices that measure and record data that the robot will use to make decisions
      2. Function: Track robot position/orientation, detect obstacles, track state of sample collector
      3. Components: Ultrasonic Sensor, IMU, Camera Module, Wheel Encoders
   6. Actuation
      1. Purpose: Controlling the device that is used to gather geological samples from the terrain, allowing the rover to physically interact with its environment.
      2. Function: Utilizing a 2 (maybe 3) DOF articulated arm for scooping actions
      3. Components: 1 main arm link, 3 servo motors (shoulder base rotation, shoulder hinging, wrist hinging – HIGH TORQUE), end effector (scooper)
   7. Communications
      1. Purpose: Data transfer between onboard subsystems and controllers.
      2. Function: Sending sensor data to controller, outputting controller commands, and uploading new instructions to master controller.
      3. Components: UART/I2C/SPI buses, USB/Ethernet Cable (for uploading code to controller)
2. **Performance specs**
   1. Chassis
   2. Drivetrain
   3. Power System
   4. Control
   5. Sensing
   6. Actuation (Sampler Device)
   7. Communications
   8. Software
3. **Interfacing**
   1. **I2C**
      1. Benefits? Only uses two wires to connect many components
      2. Where will I use it? For connecting all the sensors to the Pi
      3. Limitations? Can only be used in short wires, susceptible to noise, sometimes devices can have the same I2C address, slow speed (not suitable for camera)
   2. **CSI**
      1. Benefits? High data throughput, Pi is optimized for this, easy connection, power-efficient
      2. Where will I use it? Connection between camera module
      3. Limitations? Limited to 1-2 cameras (depending on # of pi ports), requires CSI-compatible camera
   3. **Wifi** (for streaming camera feed/sensor data to laptop from Pi)
   4. **PWM** (controller output signal to ESCs and motors)
   5. **UART**
      1. Benefits? Only 2 lines, compatible with most GPS systems, asynchronous streaming makes it simpler.
      2. Where will I use it? Communicating with a GPS module
      3. Limitations? Can only hook up between two devices (controller and GPS), both devices must agree on the data transfer speed (baud rate – 9600 bauds/s), code needs to be able to read the data somewhat quickly
4. **Documentation**
   1. [BOM Link](https://docs.google.com/spreadsheets/d/1kCPw5noG4wiGUhccGp7jNWA5f-jaydbZb4b8pzQ2dAw/edit?usp=sharing)
   2. [GitHub](https://github.com/kwag1227/Autonomous-Rover-Project)
   3. YT Video Progress Reports:

**Week 1 – Project Planning & Scoping**

* Finalize the project overview document (mission, constraints, subsystems)
* Create early sketches of subsystems (especially chassis and drivetrain)
* Begin organizing the documentation system (e.g., GitHub, folders)

**Week 2 – Conceptual Design**

* Create 2D sketches/layouts of major mechanical components
* Define basic control flow between sensors, logic, and motors
* Start brainstorming PCB needs (what you want the custom board to handle)

**Week 3 – Detailed Design**

* Finalize mechanical and control system architecture
* Create wiring diagrams (power and signal paths)
* Select core electrical components (ESCs, motors, sensors, regulators)
* Begin PCB schematic in KiCad or EasyEDA

**Week 4 – Component & 3D Printer Orders**

* Order primary components (drive motors, ESCs, microcontroller, sensors)
* Order 3D printer (if committing to the project at this point)
* Refine PCB schematic based on chosen components

**Week 5 – Mechanical CAD + PCB Design**

* Begin CAD design of chassis and drivetrain
* Finalize schematic for PCB
* Begin PCB layout (trace routing, board shape, connectors)

**Week 6 – Finalize Design & Order PCB**

* Finalize 3D models and overall system layout
* Submit PCB for fabrication (budget 1-week delivery)
* Order any additional electrical components (connectors, headers, etc.)

**Week 7 – Parts Arrival & Mechanical Printing Begins**

* Begin 3D printing mechanical parts
* Receive and organize electronic components
* Receive custom PCB and inspect for quality

**Week 8 – PCB Assembly + Mechanical Assembly**

* Solder components onto the PCB
* Test power, signal, and microcontroller connections on the PCB
* Begin chassis/mechanical assembly

**Week 9 – Wiring & Integration**

* Mount PCB and electronics into chassis
* Route and connect power/signal wiring to all devices
* Begin system-level electrical checks

**Week 10 – Initial System Bring-Up**

* Power-on system and test motors via ESCs
* Confirm communication between Pi/Arduino and peripherals
* Begin testing remote control or manual interface

**Week 11 – Sensor & Control Logic**

* Test ultrasonic sensor, IMU, and any camera modules
* Begin implementing logic for obstacle avoidance
* Prototype and test basic sample collection mechanism (if applicable)

**Week 12 – Software & Field Testing**

* Refine control code
* Conduct outdoor testing over varied terrain
* Debug electrical, mechanical, or software issues as needed

**Week 13 – Wrap-Up & Documentation**

* Final testing and polish
* Record project demo and final video log
* Finalize GitHub repo, BOM, Gantt chart, and summary report