ECE 492
Motor Design History

Overview

The motion system for the BAD-2024 autonomous wheelchair consists of two MP24 Brushless DC (BLDC) motors, powered by a 24V 60Ah $LiFePO_4$ battery and managed by EZ Drive motor controllers. These components provide reliable propulsion, while integrated electronics ensure precise control and responsiveness in both manual and autonomous modes.

Version 1 – Conceptual Selection and Performance Criteria (Sept 2024)

In the early phase of the project, we outlined the performance requirements of the drive motors based on load and speed goals:

- Must support a load of up to 400 lbs
- Target speed: 4 mph
- Estimated wheel diameter: 12 inches, requiring 112-140 RPM
- Desired torque: 100–150 Nm peak, 8–15 Nm continuous

Initial brainstorming considered several DC motor models. We prioritized brushless gear motors due to their high torque, reliability, and efficiency in robotics platforms. At this point, we were still uncertain about exact compatibility with controllers and frame geometry.After evaluating multiple options, we selected MP24 Right/Left Hand Brushless DC (BLDC) gear motors rated for:

- 24V operation,
- Approx. 140 RPM with a 21.33:1 gear ratio, and
- Continuous torque of ~10–15 Nm, with peaks exceeding 100 Nm.

Version 2 – Integration Planning and First Wiring Attempt (Oct–Dec 2024)

By mid-fall, we selected the MP24 Right/Left Hand BLDC motors, rated at:

- 24V operation
- 140 RPM output with a 21.33:1 gear ratio
- Integrated Hall effect sensors for real-time feedback

We also chose the EZ Drive motor controllers, which matched the motor specifications and provided built-in safety features. These controllers were ideal for our needs, but integration posed a challenge. Despite proper wiring and controller setup, the motors initially refused to spin. We had to go through multiple troubleshooting sessions over Zoom with the ElectroCraft Support team to explore possible solutions.

After several attempts, we finally understood the necessary process to get them running. By the end of the Fall semester, we were able to operate the motors without using a reference speed. After returning from Winter Break, we focused on integrating the system with a microcontroller to send the reference speed and successfully drive the motor.

Version 3 – Signal Conditioning and Controller Interface (Jan 2025)

After troubleshooting with ElectroCraft support over Zoom and analyzing datasheets, we verified that to get the motor spinning:

- A proper analog voltage signal must be fed into the speed control input.
- Hall sensors (U, V, W) and motor phase wires must be correctly connected.
- Brake signal needs to be powered to release the lock.

By Mid February, we were able to run the motors without reference speed, using default controller parameters.

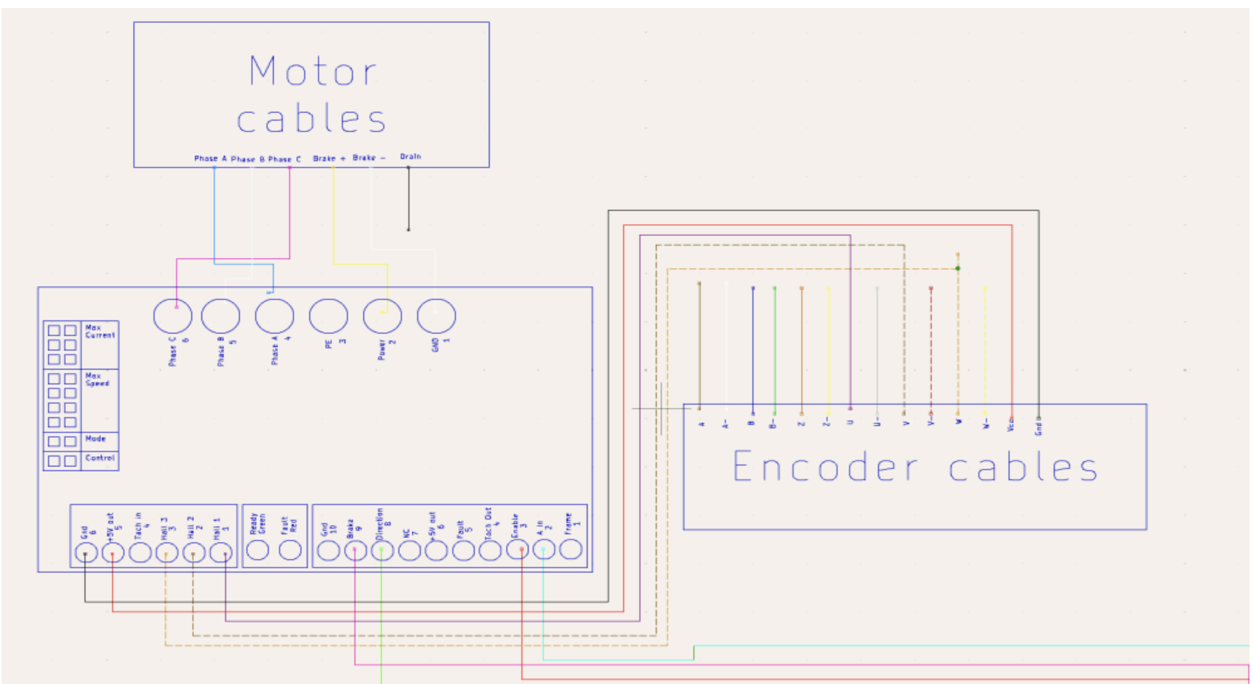

Fig 1 : Motor design wiring diagram for the motor and controller.

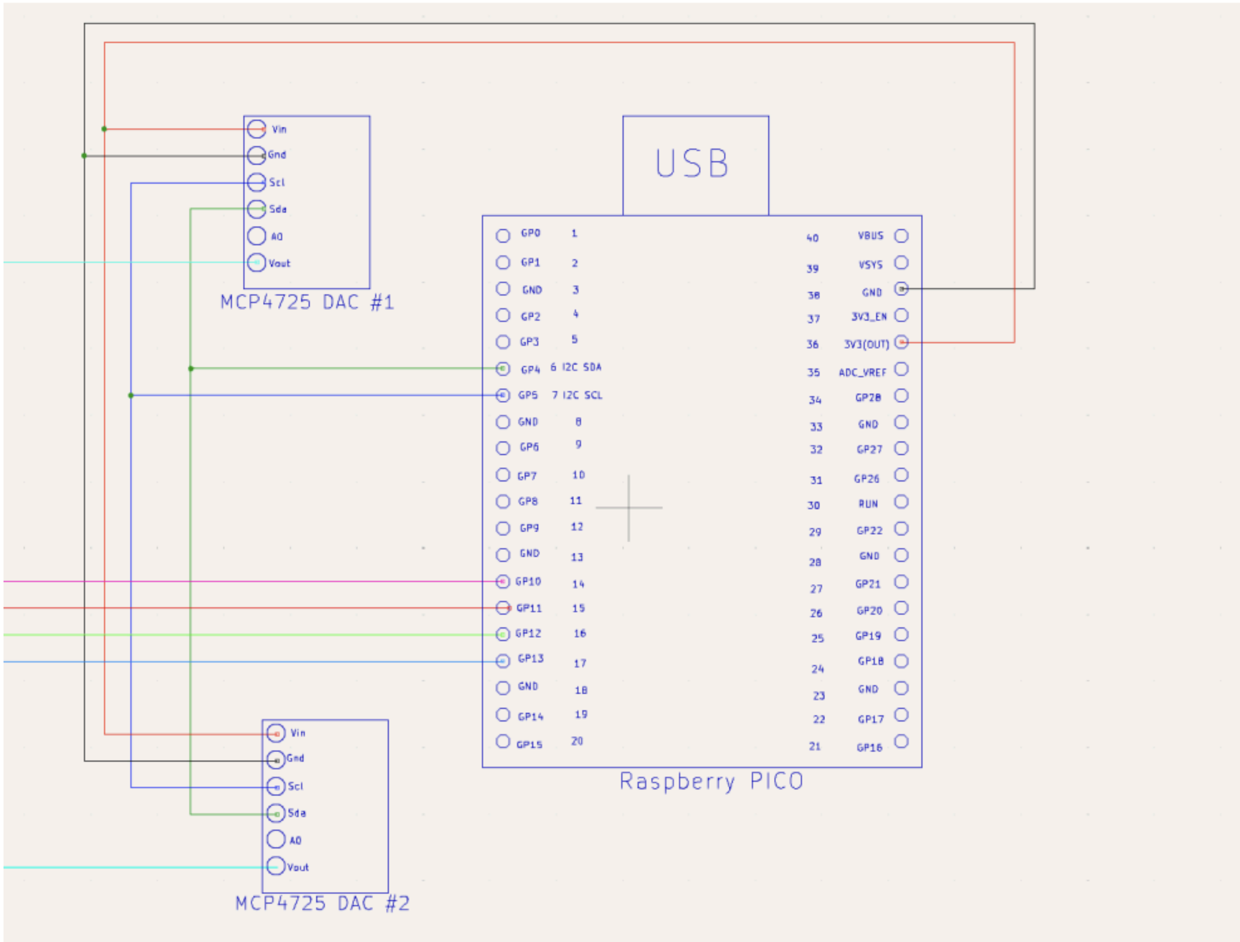

Fig 2 : Wiring diagram for using the motor controller via a microcontroller

Final Version – Full Integration and Reference Speed via Microcontroller (March 2025)

After running the motor we focused on integrating the motors with our control system. The solution involved building a custom signal chain:

1. Jetson Nano computes the desired speed based on LIDAR and Sensor data.
2. Speed is transmitted via USB serial to the Raspberry Pi Pico.
3. The Pico sends this speed to a DAC, converting it to an analog voltage.
4. A 3.3× gain amplifier scales the voltage to the required 0–10V range.
5. This signal is fed to the EZ Drive controller, initiating motor movement.

At this stage, we also finalized the microcontroller control of brake, direction, and enable/disable signals, allowing full digital control over the motor.We successfully completed all integration and verified motor functionality under load using both joystick and autonomous inputs. The motors now reliably power the wheelchair in all control modes.

Battery

From the beginning of the project, we identified that a 24V battery would be required to meet the operating voltage of our motors, controllers, and electronic systems. To ensure the wheelchair could run continuously for a full workday, we aimed for a minimum capacity of 50Ah, with an ideal target of 60Ah to allow for buffer and inefficiencies.

We selected a 24V 60Ah $LiFePO_4$ battery, which met all of our core specifications:

- High energy density (1440Wh)
- Stable voltage output under load
- Built-in protection and long cycle life
- Lightweight (27.5 lbs), making it manageable for the robot frame

This battery powers:

- Both BLDC motors
- All sensors and compute modules
- The frontend display and user interface.
- All electronic components including LIDAR and Jetson Nano

It was installed securely in the bottom electronics box using custom 3D-printed holders, one of which also supports the integrated charger adapter. The battery is flush-mounted against the 80/20 frame to prevent shifting during movement.

Battery Monitoring Circuit

To maintain a safe and reliable power system for our wheelchair, we needed a way to monitor the battery's voltage and estimate its state of charge. This information had to be accessible to the microcontroller and viewable on the onboard screen, while adding minimal cost and power overhead.

Design Iterations and Options Considered

We evaluated four main options during our design process, weighing each based on cost, data visibility, and integration feasibility:

Option 1: LED Battery Indicator Module

- Pros: Simple, plug-and-play, extremely cheap
- Cons: No data output; could only visually indicate levels
- Rejected due to lack of digital communication—couldn't be logged or displayed.

Option 2: Prebuilt LCD Voltage Display

- Pros: Affordable and accurate voltage display
- Cons: Stand-alone display, no integration with Jetson Nano or Pico
- Rejected because it was visual-only and not useful for system integration.

Option 3: Smart Bluetooth Battery Monitor

- Pros: Included mobile app to show SoC, voltage, and current
- Cons: Too expensive, and not screen-integratable
- Rejected due to cost and limited customization options.

Final Selection – Option 4: Custom Voltage Divider + Op-Amp + Microcontroller

- Components used:
    - Resistors for a voltage divider
    - LM324 op-amp to condition and buffer the analog signal
    - Raspberry Pi Pico to read and log values

- Pros: Digital data access, real-time logging, fully integratable, no added cost

- Cons: Required coding and soldering, but allowed full system integration

This was selected as the final implementation because it met all design goals - accurate, inexpensive, integratable, and robust.With this we were able to send battery status over ROS and display percentage on screen.

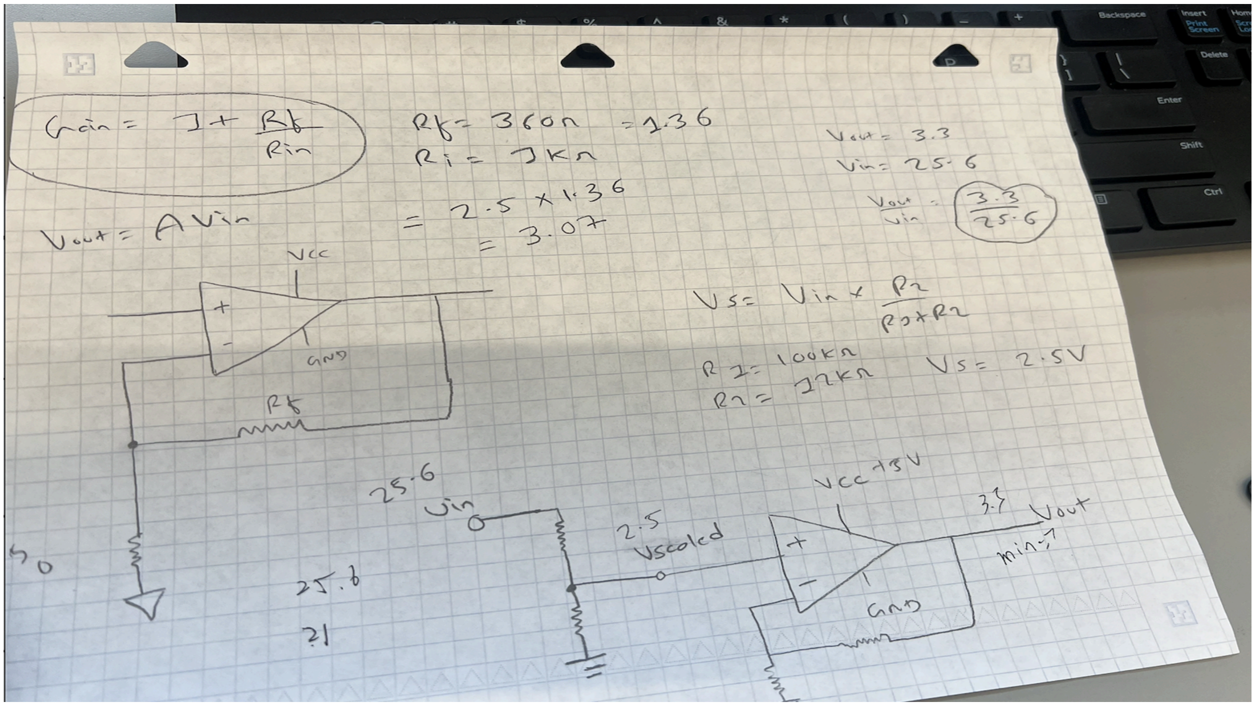

Fig 1 : First Version Design for Battery Monitoring Circuit

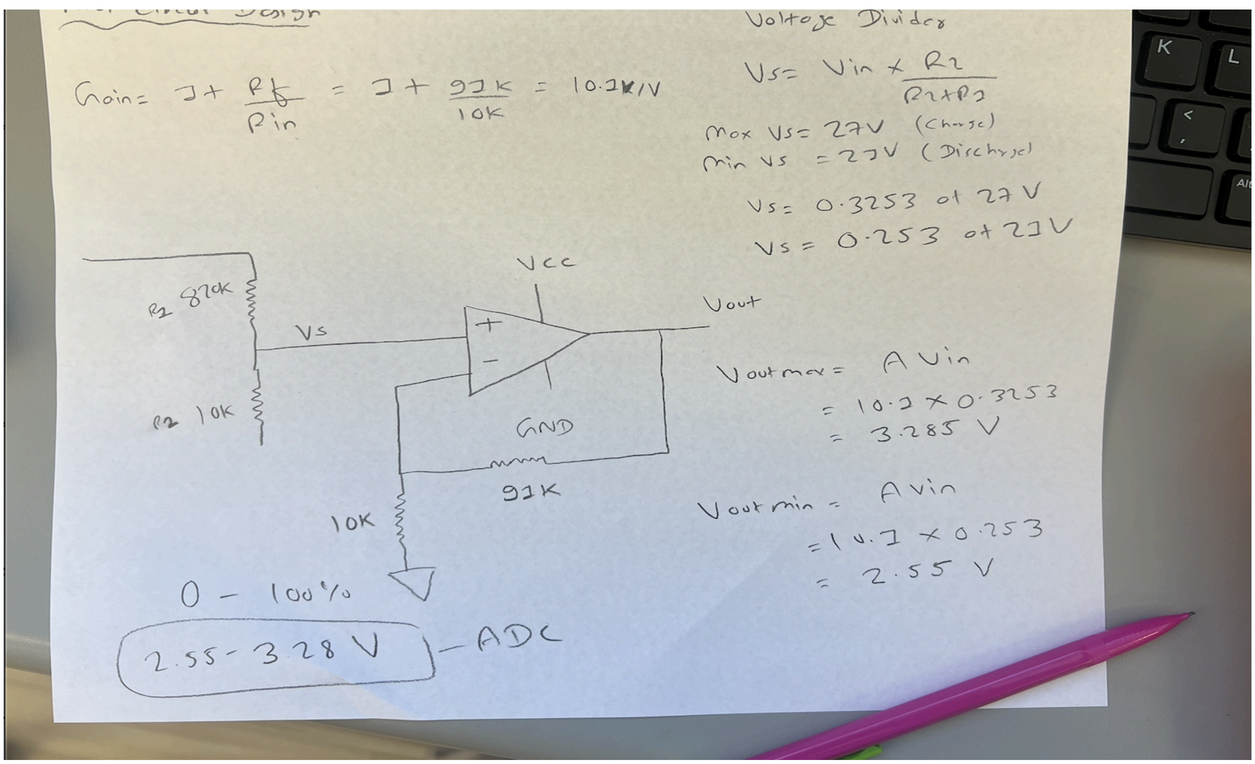

Fig 1 : Final Design for Battery Monitoring Circuit