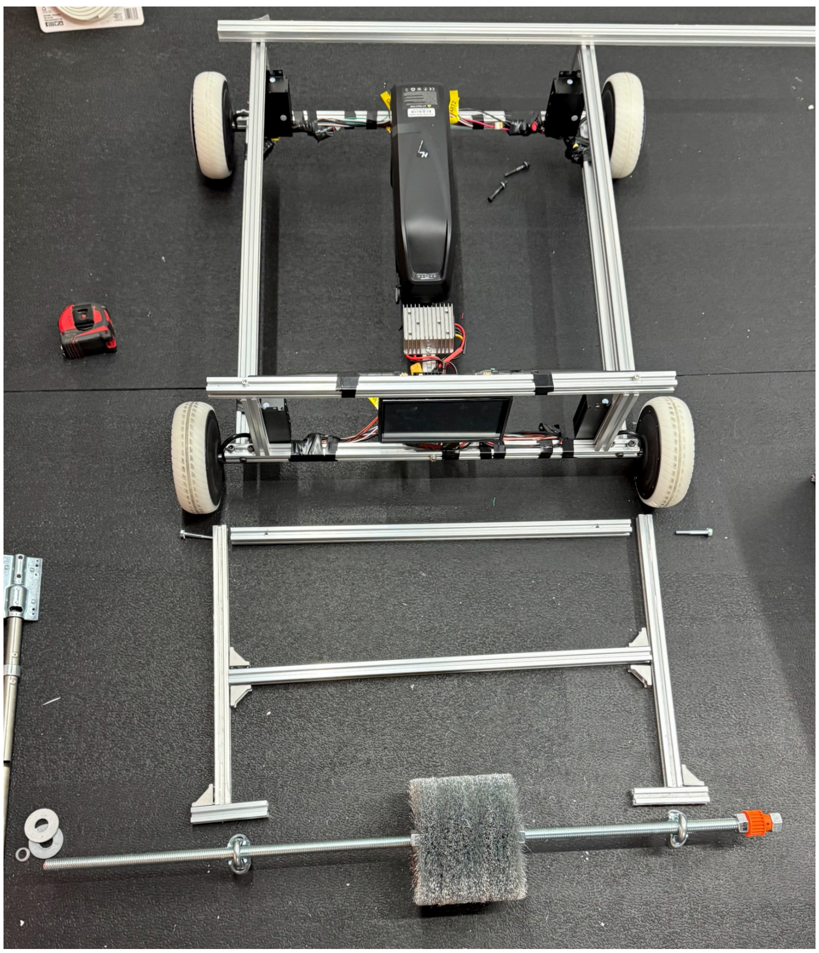
TECHNICAL MANUAL

BVR0

Base Vectoring Rover



Revision 0.2   January 2026

Municipal Robotics
Cleveland, Ohio
muni.works

Contents

About This Manual

This manual is written by the people who built and operate BVR rovers. It's not a marketing document or a wish list: it's the real procedures we use, with the mistakes we've made and the lessons we've learned.

If something is unclear, wrong, or missing, let us know. This is a living document that improves with every build.

Revision History

Rev	Date	Changes
0.1	December 2025	Initial release
0.2	January 2026	Added time estimates, lessons learned, BOM, glossary

Document Conventions

Throughout this manual you'll see:

"We learned…"	Lessons from real build experience
⚡ COMMON MISTAKE	Errors we've seen (and made)
●●○	Difficulty rating (1-3)
15 min	Estimated time for procedure

ℹ **NOTE** This is a living document. Report errors or suggestions at github.com/muni-works/muni.

BVR0 Cheat Sheet

Print this page. Laminate it. Tape it to your workbench.

Emergency Stop

1. Red button on rover
2. Spacebar on controller
3. Guide button (gamepad)
4. Connection loss (auto)

CAN Bus IDs

0	VESC Front Left
1	VESC Front Right
2	VESC Rear Left
3	VESC Rear Right
10+	Tool Attachments
0x0B00	LED Controller

Battery Voltages

Full charge	**54.6V**
Nominal	48V
Low warning	42V
Cutoff	**39V**
12V rail	11.5-12.5V

Torque Specs

Frame (M5)	**4 Nm**
Motor mounts (M5)	**4 Nm**
Electronics (M3)	**0.5 Nm**
Wheel axle	Hand tight

Wire Colors

- 48V Power (+)
- Ground (-)
- 12V Power
- Motor Phase A
- Motor Phase B
- Motor Phase C
- CAN High
- CAN Low

LED Status Codes

- Solid Ready
- Pulse Teleop
- Pulse Autonomous
- Blink Low Battery
- Solid E-Stop
- Fast Fault
- Off No Power

CLI Commands

`muni status`	Health
`muni can scan`	Devices
`muni motors test`	Spin
`muni leds set idle`	Reset
`muni logs -f`	Live logs
`systemctl restart bvrd`	Restart

Network Ports

Hostname	bvr-XX
SSH	22
WebSocket	8080
Video	5600
Metrics	8086

Pre-Flight (2 min)

- [?] Battery > 42V
- [?] E-Stop released
- [?] Wheels spin free
- [?] Wheel bolts tight
- [?] Connectors secure
- [?] Camera/LiDAR clean
- [?] Controller paired

Quick Troubleshooting

No power
→ Check breaker, battery

Motors not responding
→ Release E-Stop, `muni can scan`

Erratic movement
→ Check motor IDs, phase order

Video lag
→ Check WiFi, reduce resolution

GPS no fix
→ Open sky, check antenna

Dimensions

Frame L × W	60 × 50 cm
Height	45 cm
Wheelbase	45 cm
Track width	55 cm
Clearance	8 cm
Weight (empty)	15 kg

Support

Web	muni.works
Docs	muni.works/docs
GitHub	github.com/muni-works

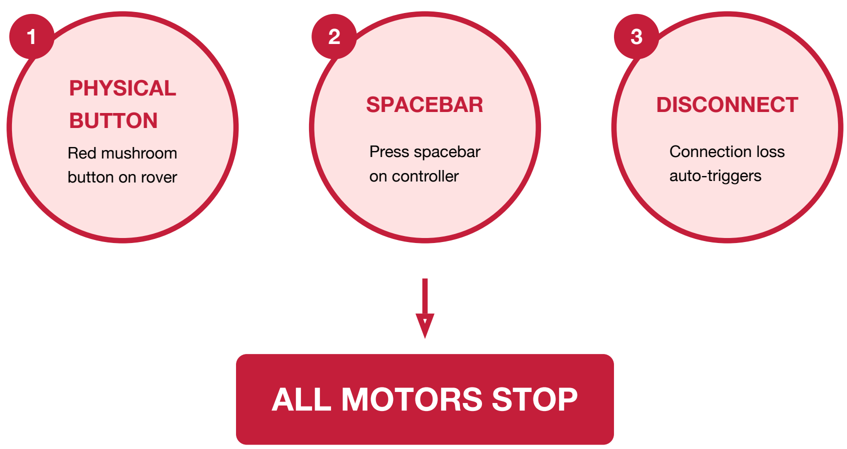
Serial: _____ Build: _____

Quick Reference

These three pages are the ones you'll use daily. Memorize the e-stop methods. Run the pre-flight checklist every time. Keep the controls layout in your head. Everything else in this manual is for building and fixing. This section is for operating.

Emergency Stop

⚠ **DANGER** **Know this page.** If anything goes wrong, use one of these three methods immediately.



When to E-Stop:

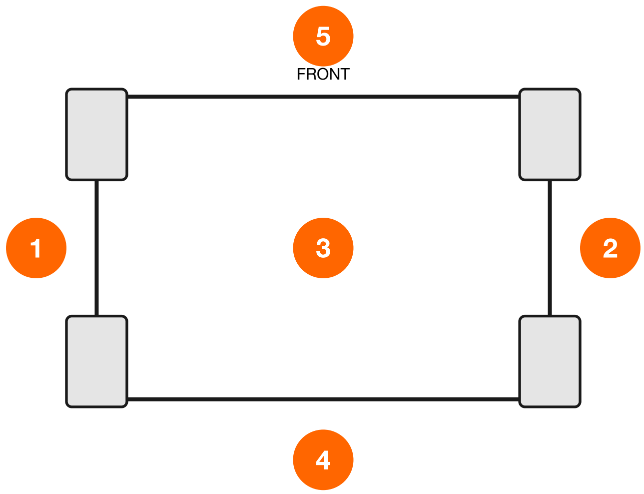
- Person in path of rover
- Unexpected movement
- Smoke, sparks, or fire
- Loss of control
- Any doubt about safety

To resume after E-Stop:

1. Resolve the cause
2. Release physical button (if used)
3. Reconnect controller
4. Verify telemetry on dashboard
5. Resume operation

Pre-Flight Checklist

Daily inspection before operation 2 min



- [ ] **1** Wheels spin freely, no debris
- [ ] **2** All wheel bolts tight
- [ ] **3** E-Stop button not stuck
- [ ] **4** Battery voltage > 40V
- [ ] **5** Camera and LiDAR clean
- [ ] **6** All connectors secure

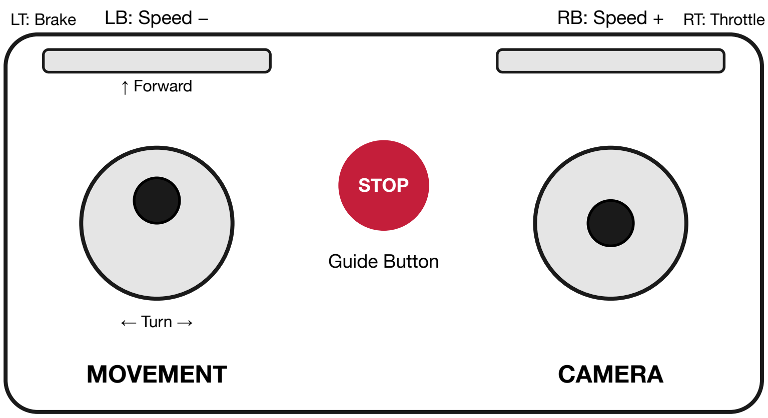
> i **NOTE** If any check fails, do not operate. Resolve the issue first.

> *“We learned this the hard way:” We once operated with a loose wheel bolt. 10 minutes in, the wheel nearly came off mid-turn. The 2-minute checklist beats a 2-hour field repair.*

Controls

Controller mapping reference

1 min read



Input	Action
Left Stick Up/Down	Forward / Reverse
Left Stick Left/Right	Turn left / right
Right Stick	Pan camera view
Left Bumper (LB)	Decrease max speed
Right Bumper (RB)	Increase max speed
Left Trigger (LT)	Brake / slow down
Right Trigger (RT)	Throttle (overrides stick)
Guide Button (center)	**Emergency Stop**
Spacebar (keyboard)	**Emergency Stop**

Before You Begin

This section explains what you're building, what you'll need, and how long it takes.

What You're Building

The BVR0 is a four-wheeled skid-steer rover with hub motors, a 48V power system, and onboard compute. It's designed for outdoor municipal work (snow clearing, mapping, patrol) but the base platform is general-purpose.

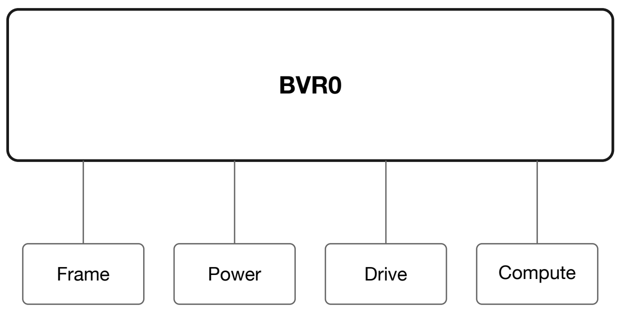


Figure 4: Four subsystems: mechanical frame, power distribution, drivetrain, and compute/sensors.

Prerequisites

Skills needed:

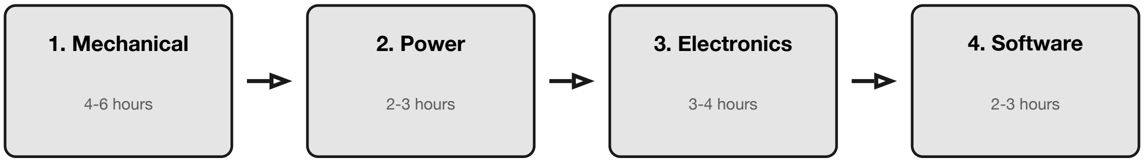
- Basic hand tools (hex keys, screwdrivers, wrenches)
- Wire stripping and crimping
- Soldering (through-hole level)
- Comfort with Linux command line
- Ability to read wiring diagrams

You do NOT need:

- CNC or machining (parts are outsourced or hand-cut)
- PCB design (all boards are off-the-shelf)
- Deep embedded programming (firmware is pre-built)

Build Phases

The build has four phases. Complete each before starting the next.



Total: 12-16 hours (split across 2-3 days recommended)

Recommended Order

Day	Tasks	Time
1	Cut extrusions, assemble frame, mount motor brackets	4-5 hr
2	Install motors, wire power system, mount electronics plate	4-5 hr
3	Wire VESCs and CAN bus, flash Jetson, configure and test	4-5 hr

✓ **TIP** Don't rush. A clean build with good cable management saves hours of debugging later.

Tools Required

Essential:

- Hex key set (2, 2.5, 3, 4, 5 mm)
- Phillips screwdriver
- Wire strippers (10-22 AWG)
- Crimping tool (for ferrules)
- Soldering iron + solder
- Multimeter
- Heat gun or lighter (heat shrink)

Helpful:

- Torque wrench (4 Nm range)
- Miter saw or hacksaw (for extrusions)
- Deburring tool
- Cable tie gun
- Label maker
- Helping hands (for soldering)

Materials Checklist

Before starting, verify you have:

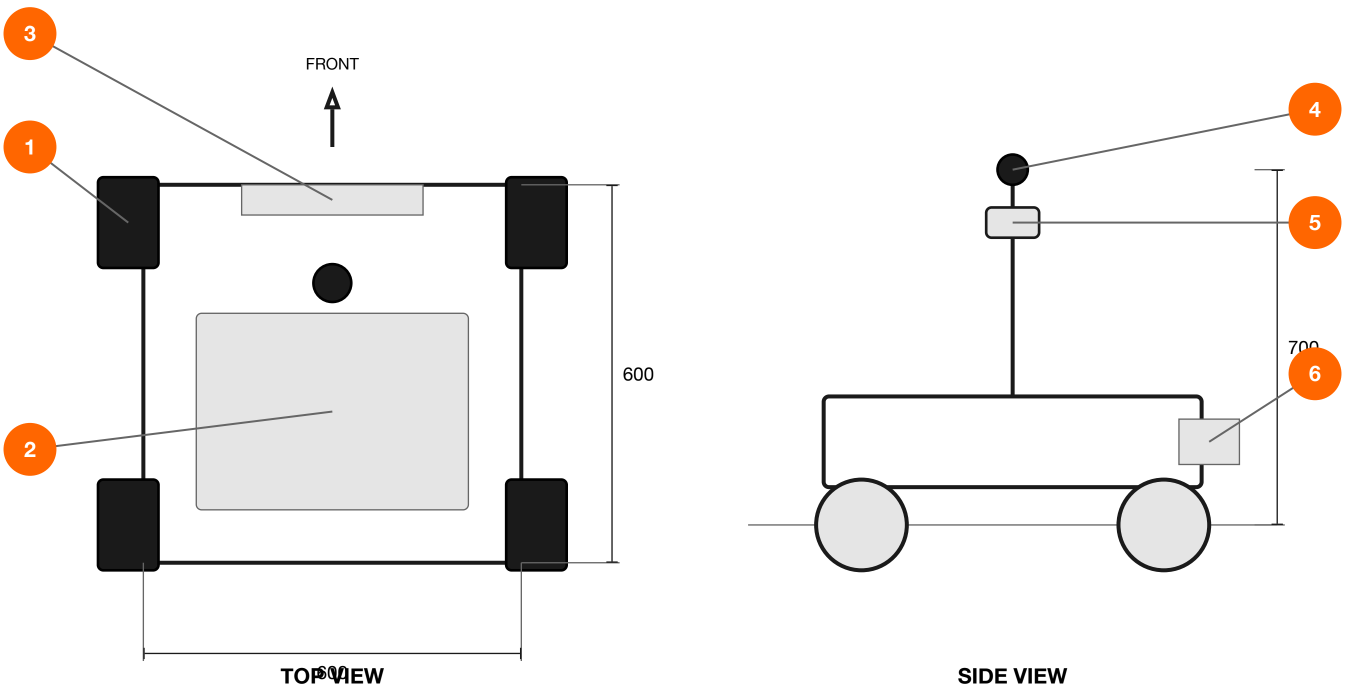
- [ ] All BOM items received and inspected
- [ ] Extrusions cut to length (or stock to cut)
- [ ] Motor brackets fabricated (or ordered)
- [ ] Electronics plate fabricated (or ordered)
- [ ] Battery charged to 50%
- [ ] Jetson flashed with JetPack

See **Appendix A: Bill of Materials** for the complete parts list with vendors.

Overview

The BVR0 is the first rover in the Muni fleet. It's intentionally simple: a rigid aluminum frame, four hub motors, and enough compute to handle autonomy. No suspension, no steering linkages, no complex mechanisms. Everything that can break has been removed.

The design philosophy is "municipal-grade": it needs to survive Cleveland winters, sidewalk salt, and the occasional collision with a park bench. The 2020 aluminum extrusion frame can be rebuilt with hardware store parts. The hub motors are the same units used in hoverboards and e-scooters (proven, cheap, replaceable). The electronics are mounted on a single plate that slides out for service.



Components

1	Hub motor wheels (×4)
2	Electronics bay
3	Tool mount
4	360° camera
5	LiDAR sensor
6	Tool attachment

Key Specifications

Dimensions	600 × 600 × 700 mm
Weight	30 kg with battery
Speed	1.0-2.5 m/s
Runtime	4 hours
Temp range	-20°C to +40°C

Specifications

These are the target specifications for a standard BVR0 build. Your rover may vary slightly depending on component sourcing and local modifications.

Mechanical

Footprint	600 × 600 mm
Height	700 mm (with mast)
Weight	30 kg
Ground clearance	50 mm
Wheel diameter	160 mm
Frame	2020 aluminum extrusion

Electrical

Main battery	48V 20Ah (960 Wh)
Chemistry	13S LiPo
Voltage range	39-54.6V
Accessory rail	12V 10A
Main fuse	100A

Drivetrain

Motors	4× 350W hub motors
Controllers	4× VESC 6.7
Drive type	Skid-steer
Max speed	2.5 m/s
Cruise speed	1.0 m/s

Perception

LiDAR	Livox Mid-360
Camera	Insta360 X4 (360°)
GPS	RTK-capable (optional)

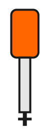
Compute

Main computer	Jetson Orin NX 16GB
Connectivity	LTE + WiFi
CAN bus	500K baud

Required Tools

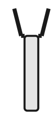
Hex Keys

2.5, 3, 4, 5 mm



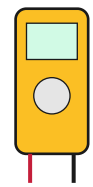
Screwdriver

Phillips 2



Wrenches

8, 10, 13 mm



Multimeter

V / Ω / Continuity

Required

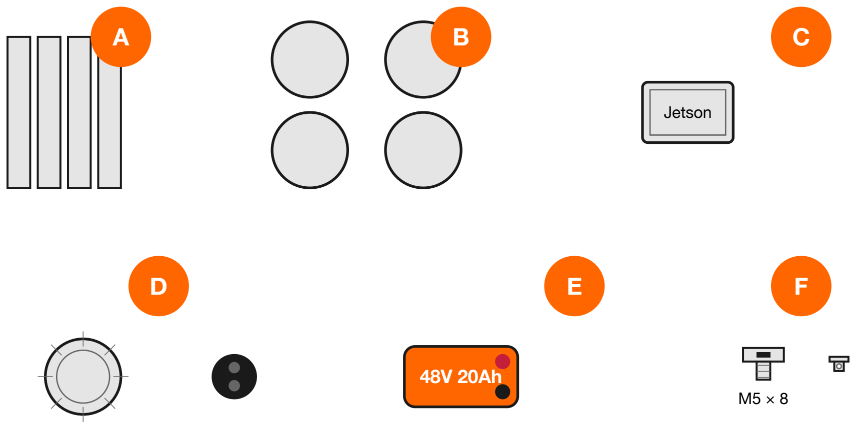
- Hex key set (metric: 2.5, 3, 4, 5 mm)
- Phillips screwdriver (2)
- Adjustable wrench or socket set (8, 10, 13 mm)
- Multimeter (voltage, resistance, continuity)
- Wire strippers (20-12 AWG)
- Soldering iron (40W+) and solder
- Heat shrink assortment
- Miter saw or hacksaw (for extrusions)

Recommended

- Torque wrench (4 Nm for M5)
- Drill and drill bits (3.2, 4.2, 5 mm)
- Tap set (M4×0.7, M5×0.8)
- Deburring tool
- Cable ties (assorted sizes)
- Label maker
- Work mat
- Helping hands / PCB holder

> i **NOTE** All M5 bolts should be torqued to 4 Nm. Over-tightening can strip aluminum threads.

Parts List



Parts

A	Chassis: extrusions, brackets, plate	$150
B	Drivetrain: motors, VESCs	$830
C	Electronics: Jetson, CAN, LTE	$900
D	Perception: LiDAR, camera, pole	$1,800
E	Power: battery, DC-DC, E-stop	$400
F	Hardware: bolts, T-nuts, wire	$100

Cost Summary

Chassis	$150
Drivetrain	$830
Electronics	$900
Perception	$1,800
Power	$400
Hardware	$100
Total	**$4,180**

All parts commercially available.

Full BOM with links:
`docs/hardware/bom.md`

Where to Buy:

Category	**Primary Source**
Notes	Extrusions
Amazon, Misumi	2020 V-slot or T-slot
Motors	AliExpress
Search “hoverboard hub motor 350W”	VESCs
Flipsky	VESC 6.6 or 6.7
Jetson	NVIDIA, Arrow, Seeed
Orin NX 16GB + carrier	LiDAR
Livox / DJI Store	Mid-360, 1 week ship
Camera	Amazon, B&H
Insta360 X4	Custom cuts
SendCutSend	Upload DXF, 3-5 day turnaround

Custom Fabricated Parts

BVR0 requires **zero** custom fabricated parts.

No custom parts needed

How BVR0 avoids custom fabrication:

Component	BVR0 Approach
Alternative (BVR1)	Motor mounting
Direct bolt to 2020 T-slot	Custom brackets
Battery	Downtube e-bike battery
Custom pack + tray	Electronics
Tape/zip-tie to chassis	Custom plate

> i **NOTE** BVR0 is intentionally scrappy. The goal is to get a working rover with parts you can order today and assemble this weekend. No waiting for laser cutting, no CAD required.
>
> Once you've proven the concept, BVR1 introduces custom parts for a cleaner, more serviceable build.

Hardware Reference

Standard fasteners and hardware used throughout the build.

Bolts

Size	Use
M3×8	Electronics mounting
M5×8	T-nut, light duty
M5×10	T-nut, standard
M5×12	Motor to frame
M5×16	T-nut, through plate

T-Nuts

Type	Use
M5 drop-in	Post-assembly insertion
M5 slide-in	Pre-assembly (easier)
M6 drop-in	Heavy-duty mounts

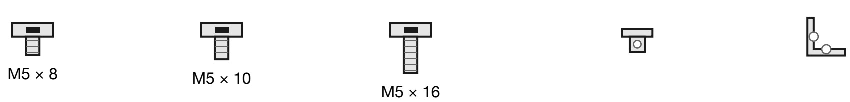
Connectors

Type	Rating
Use	XT90
90A	Battery main
XT60	60A
Motor phase	XT30
30A	12V power
JST-PH	3A
CAN bus, signals	DT 4-pin
25A	Tool connector

Wire Gauge

AWG	Use
8 AWG	Battery to bus
10 AWG	Bus to VESCs
14 AWG	12V power
22 AWG	CAN bus, signals

Common Hardware (actual size)



M5×8 M5×10 M5×16 M5 T-Nut Corner

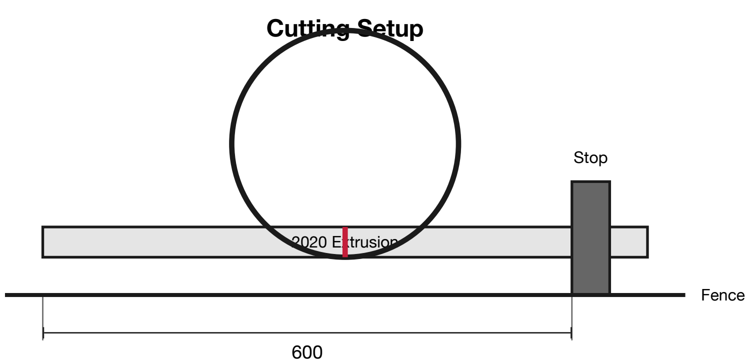
Chassis Assembly

The chassis is the skeleton of the rover. It's built from 2020 aluminum extrusion, the same stuff used in 3D printer frames and CNC machines. The T-slot design means you can mount anything anywhere, and if you mess up a hole, just slide the T-nut to a new position.

The frame goes together like adult LEGO. No welding, no precision machining. If you can use a saw and a hex key, you can build this chassis.

Cutting Extrusions

Cut aluminum stock to length 20 min



Procedure:

1. Clamp stop block at 600mm from blade
2. Place extrusion against fence and stop
3. Cut slowly to prevent burrs
4. Rotate 90° and re-cut if needed for square ends
5. Deburr all cut edges with file or deburring tool

Cut List (BVR0 standard):

Qty	Length
Purpose	4
600mm	Base frame
4	600mm
Top frame	4
250mm	Vertical posts

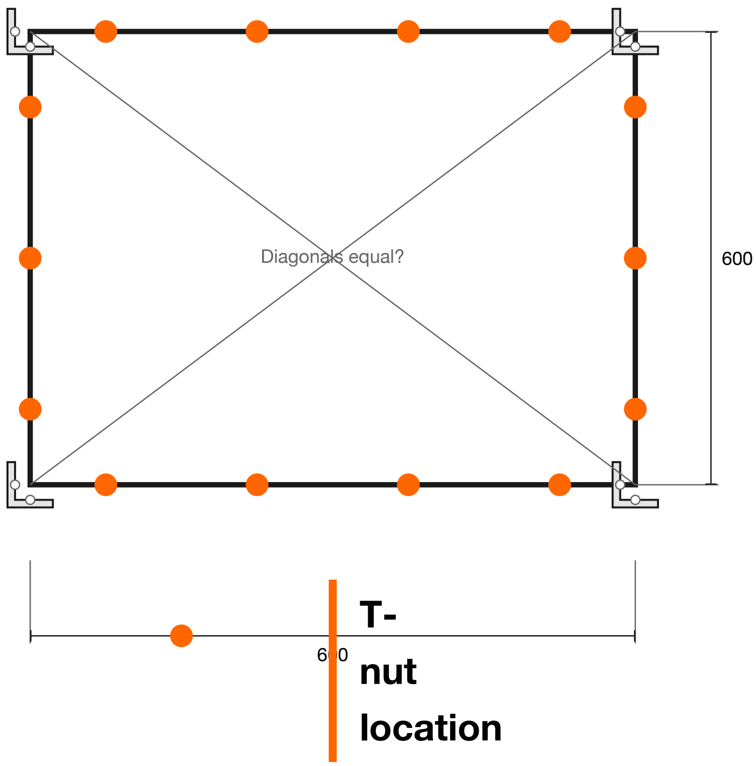
Total: 5.8m of 2020 extrusion needed.

⚠️ **WARNING** Aluminum chips are sharp. Wear safety glasses. Clean chips from T-slots before assembly.

⚡ **COMMON MISTAKE** Cutting too fast causes burrs that jam T-nuts. Slow cuts with a fine-tooth blade save deburring time.

Base Frame Assembly

Assemble the base frame square 15 min



Assembly Steps:

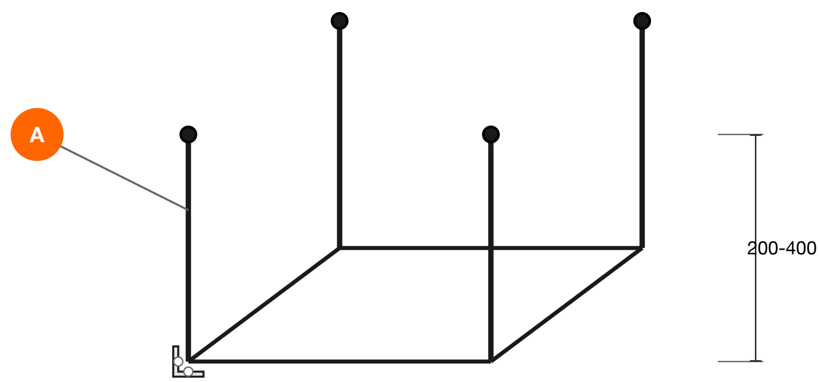
1. **Pre-insert T-nuts** into all extrusion channels (8 per extrusion, 32 total for base)
2. **Dry-fit** all four extrusions in a square, corners aligned
3. **Attach corner brackets** loosely (finger-tight M5×10 bolts)
4. **Check squareness**: measure both diagonals. They must be equal (±1mm).
5. **If not square**: tap the long diagonal corner with a mallet to adjust
6. **Tighten all bolts** to 4 Nm in a star pattern

> i **NOTE** Leave extra T-nuts in channels for later mounting. Easier now than adding drop-in nuts later.

Vertical Posts

Install corner vertical posts

10 min



Mounting Method A: Corner Bracket

Use 90° corner brackets at each post base.

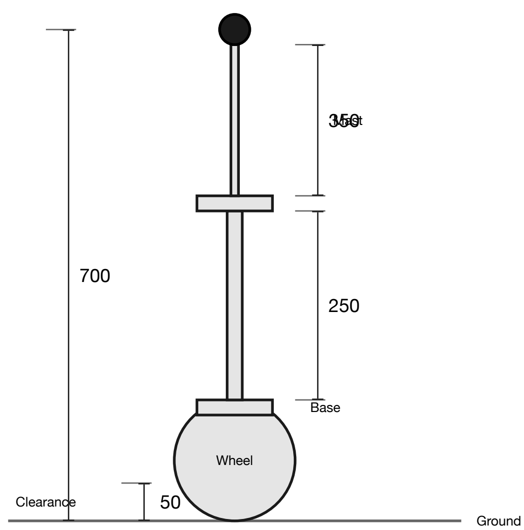
- 2× M5×10 bolts per bracket
- Insert T-nuts in both base and post
- Tighten to 4 Nm

Mounting Method B: Blind Joint

Use blind joint connectors for cleaner look.

- Drill 5mm access hole in base extrusion
- Thread M5×25 bolt through into post
- Hidden hardware, harder to adjust

Height Calculation:

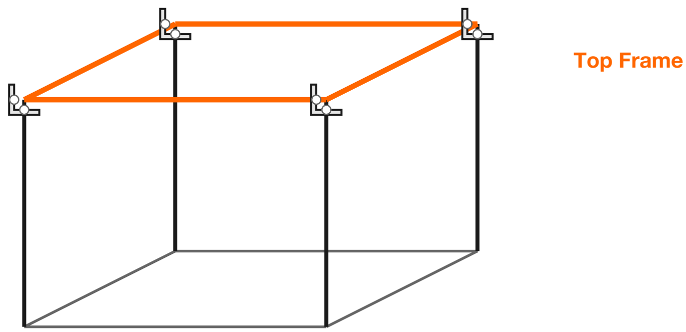


Component	Height
Cumulative	Wheel radius
80mm	80mm

Ground clearance	50mm
-	Base to top frame
250mm posts + 40mm	370mm
Sensor mast	330mm
700mm	

Top Frame

Complete the box frame 15 min



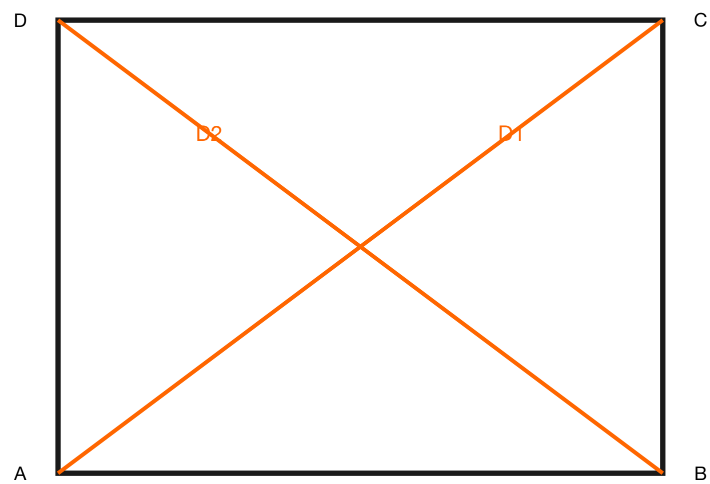
Assembly:

1. Attach corner brackets to top of each vertical post (loosely)
2. Place top frame extrusions onto brackets
3. Align extrusions flush with vertical posts
4. Check that top frame is level (use spirit level)
5. Tighten all connections to 4 Nm

> i **NOTE** The top frame provides mounting points for the electronics plate, sensor mast, and protective covers.

Squareness Check

Verify frame geometry 5 min



✓ D1 = D2 ± 1mm

Squareness Test:

1. Measure diagonal A→C (D1)
2. Measure diagonal B→D (D2)
3. Compare: D1 should equal D2 within 1mm
4. If not equal: loosen corners, tap long diagonal, re-tighten

Rigidity Test:

1. Grip opposite corners
2. Try to twist the frame
3. Frame should not flex or rack
4. If loose: check all bolt torque, add corner braces if needed

Final Checklist:

- [ ] All corners have brackets installed
- [ ] All bolts torqued to 4 Nm
- [ ] Diagonals equal within 1mm
- [ ] Frame does not rack or twist
- [ ] All T-slots clear of debris
- [ ] Extra T-nuts in channels for later use

✓ **TIP** Take a photo of the diagonal measurements. Useful reference if the frame gets knocked out of square later.

Electronics Mounting

BVR0 takes the simplest possible approach: mount electronics directly to the chassis using zip ties, electrical tape, and the T-slot channels. No custom plate, no drilling, no fabrication.

This isn't pretty, but it works. The goal of BVR0 is to get a rover running with zero custom parts. You can always upgrade to a proper electronics plate later (see BVR1 manual).

Direct Mounting Strategy

Mount electronics to chassis 45 min

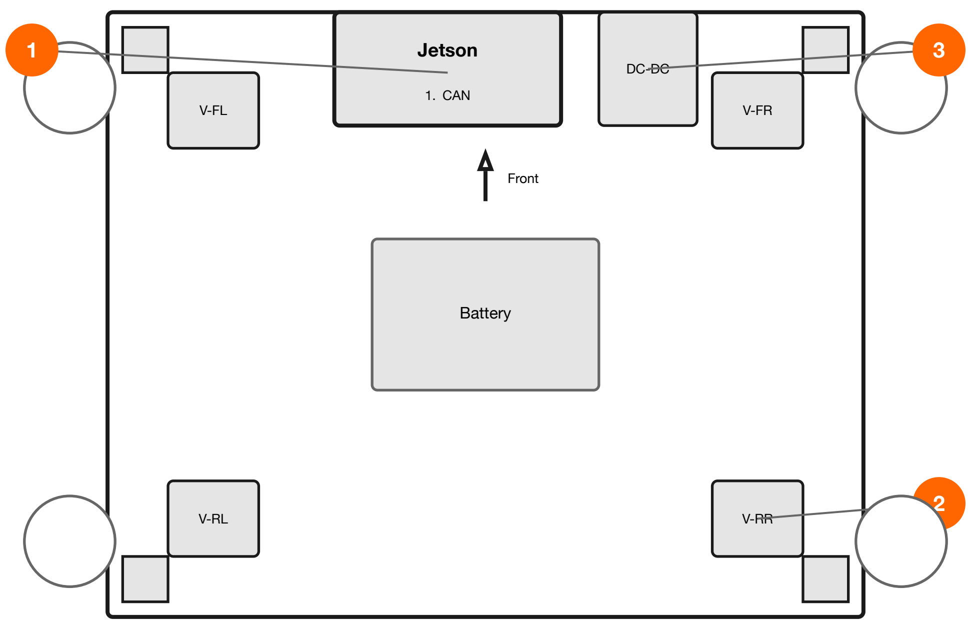


Figure 17: VESCs at corners (on vertical posts), Jetson and DC-DC on top rail.

Components:

1 Jetson Orin NX + CAN board
2 VESC 6.7 ×4 (one per corner)
3 DC-DC 48V→12V

Mounting Methods:

- Electrical tape (quick, repositionable)
- Zip ties through T-slot channels
- Velcro strips (for DC-DC)
- Double-sided foam tape (vibration dampening)

Jetson + CAN Board

Mount Jetson compute module 15 min

The Jetson Orin NX sits on a carrier board with an integrated CAN interface. This eliminates the need for a separate USB-CAN adapter.

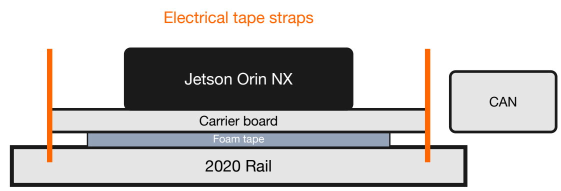


Figure 18: Jetson mounted with foam tape and electrical tape straps.

Mounting Steps:

1. Clean the extrusion surface with isopropyl alcohol
2. Apply double-sided foam tape to carrier board bottom
3. Press carrier board onto top of frame rail
4. Wrap electrical tape around rail and carrier (2-3 wraps)
5. Connect CAN board to carrier via ribbon cable or headers
6. Route power and data cables away from tape

Carrier Board with CAN:

- Waveshare or Seeed carrier with CAN
- Or: separate CAN HAT/board
- CAN-H, CAN-L, GND to VESC bus
- 120Ω termination at end of bus

Power:

- 12V from DC-DC converter
- Barrel jack or screw terminal
- 3A average, 5A peak

> *"We learned this the hard way:" Electrical tape sounds janky, but it's actually great for prototyping. It's repositionable, leaves no residue, and you can see exactly where everything is. Once the layout is proven, upgrade to proper mounts.*

VESC Mounting

Mount motor controllers 15 min

Each VESC mounts directly to the vertical post next to its wheel. This keeps phase wires as short as possible: the motor is right there.

CORNER DETAIL

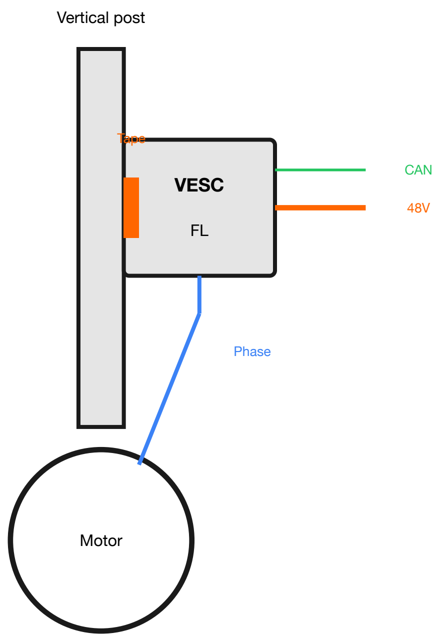


Figure 19: VESC on vertical post, directly adjacent to its motor. Minimal phase wire length.

Mounting:

- Electrical tape or Velcro to vertical post
- VESC flat against post, heatsink facing out
- Position at comfortable height for wiring
- One VESC per corner (4 total)

Wiring:

- Phase wires: direct to motor (< 15cm ideal)
- 48V power: runs from central bus to each corner
- CAN: daisy-chain around frame perimeter
- Termination: 120Ω at first and last VESC

> *"We learned this the hard way:" Mounting VESCs at the corners means longer power runs but shorter phase wires. Phase wires carry high-frequency switching currents: keeping them short reduces EMI and heat. The 48V DC bus doesn't care about a few extra centimeters.*

> ℹ **NOTE** Label each VESC with its motor position (FL, FR, RL, RR). You'll thank yourself during debugging.

DC-DC Converter

Mount voltage regulator 10 min

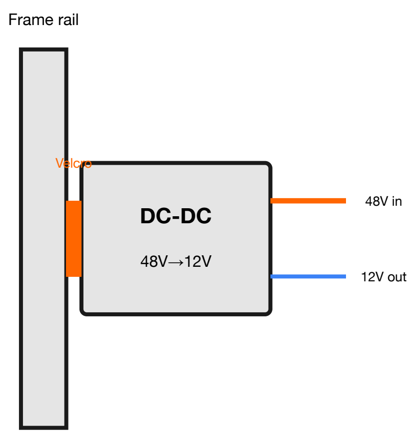


Figure 20: DC-DC converter attached to frame rail with Velcro.

Mounting Tips:

- Velcro strips work well for DC-DC (easy removal)
- Route high-current wires away from signal wires
- Leave slack for service access
- Position near Jetson to minimize 12V wire runs

⚠ **WARNING** Don't mount the DC-DC upside down. The heatsink needs to face up or outward for convection cooling.

i **NOTE** BVR0 has no inline fuse. The battery's internal BMS provides overcurrent protection. This is acceptable for a prototype but not recommended for production. BVR1 adds proper fusing.

Wiring Overview

Route and secure cables 30 min

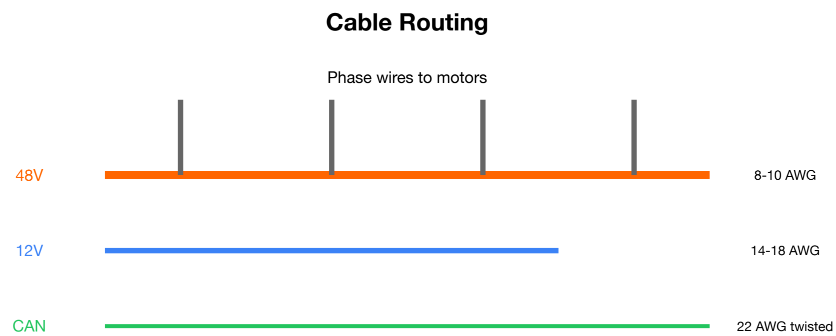


Figure 21: Keep power, 12V, and signal cables separated.

Cable Management:

- Bundle power cables together (red/black)
- Bundle CAN cables separately (twisted pair)
- Use split loom or spiral wrap for protection
- Zip tie to frame at regular intervals
- Leave service loops near connectors

Cable Separation:

- Keep CAN bus away from motor phase wires (EMI)
- Cross power and signal cables at 90° angles
- Don't run cables over hot components

> i **NOTE** Messy wiring works for BVR0 prototyping. But label everything. Future you will appreciate it.

Drivetrain

The BVR0 uses skid-steer drive: four independent hub motors, one in each wheel. To turn, the left and right sides spin at different speeds (or opposite directions). It's the same principle as a tank or a Roomba.

Hub motors eliminate chains, belts, gearboxes, and axles. The motor **is** the wheel. This means fewer parts, less maintenance, and no drivetrain to align. The tradeoff is that hub motors are heavier than outrunner motors with belt drive, but for a utility rover that's not a problem.

The motors we use are 350W hoverboard motors. They're mass-produced, cheap (around $85 each), and rated for exactly the kind of abuse a sidewalk rover will see.

Direct Frame Mounting

Mount hub motors directly to frame 30 min

The BVR0 keeps things simple: the hub motors bolt directly to the 2020 aluminum extrusion frame. No brackets, no adapters. The motor's flat mounting face sits against the extrusion, and M5 bolts pass through into T-nuts in the channel.

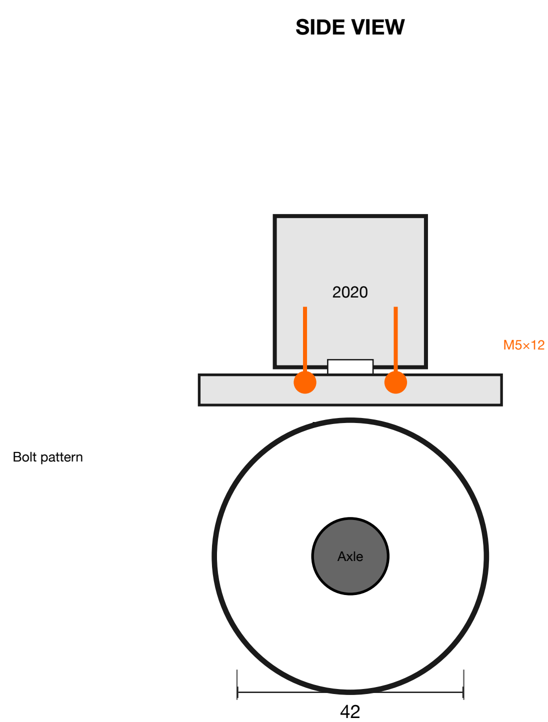


Figure 22: Hub motor mounts directly to extrusion. No bracket required.

Mounting Procedure:

1. Slide 2× drop-in T-nuts into the extrusion channel at each wheel position
2. Position hub motor with mounting face flat against extrusion
3. Align motor's 42mm bolt pattern with T-nuts
4. Insert M5×12 bolts through motor mounting holes into T-nuts
5. Hand-tighten, then torque to 4 Nm

Motor Positions:

Mount motors at frame corners, centered on each side rail. The exact position isn't critical: the T-slot system allows adjustment.

Location	Rail
Front Left	Left side rail
Front Right	Right side rail
Rear Left	Left side rail
Rear Right	Right side rail

Hardware Per Motor:

- 2× M5×12 button head bolt
- 2× M5 drop-in T-nut

Standard 6.5″ hoverboard motors have a 42mm square bolt pattern with M5 threads.

> i **NOTE** The 42mm hole pattern fits standard 6.5″ hoverboard hub motors. Some motors may have different patterns: verify before ordering.

> *"We learned this the hard way:" We originally planned custom brackets. Then we realized the motor face sits flat against the extrusion perfectly. Sometimes the simplest solution is no solution at all.*

Hub Motor Installation

Install and wire hub motors 20 min

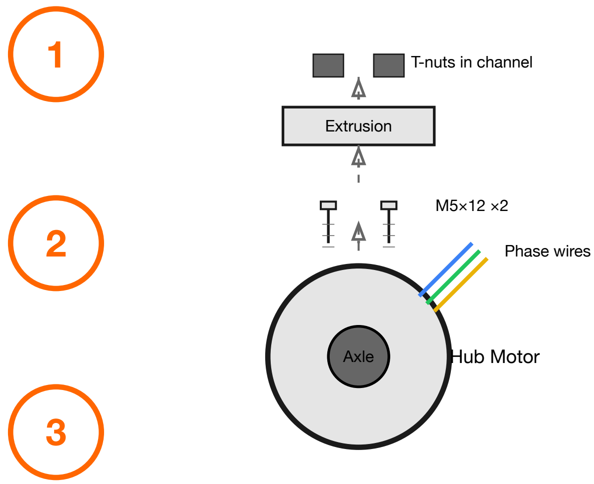


Figure 23: Motor installation sequence. Wheels come pre-mounted on hub motors.

Installation Steps:

1. Position T-nuts in extrusion channel at wheel location
2. Align motor mounting holes with T-nut positions
3. Insert M5×12 bolts through motor into T-nuts
4. Hand-thread first, then tighten to 4 Nm
5. Route phase wires toward electronics bay
6. Secure wires with cable ties (leave slack for service)

⚠️ **WARNING** Do not pinch phase wires between motor and frame. Route wires away from mounting surface before tightening.

"We learned this the hard way:" Hand-thread every bolt first before using a driver. Cross-threading an M5 into aluminum is easy to do and hard to fix.

Wheel Alignment

Check and adjust wheel alignment 15 min

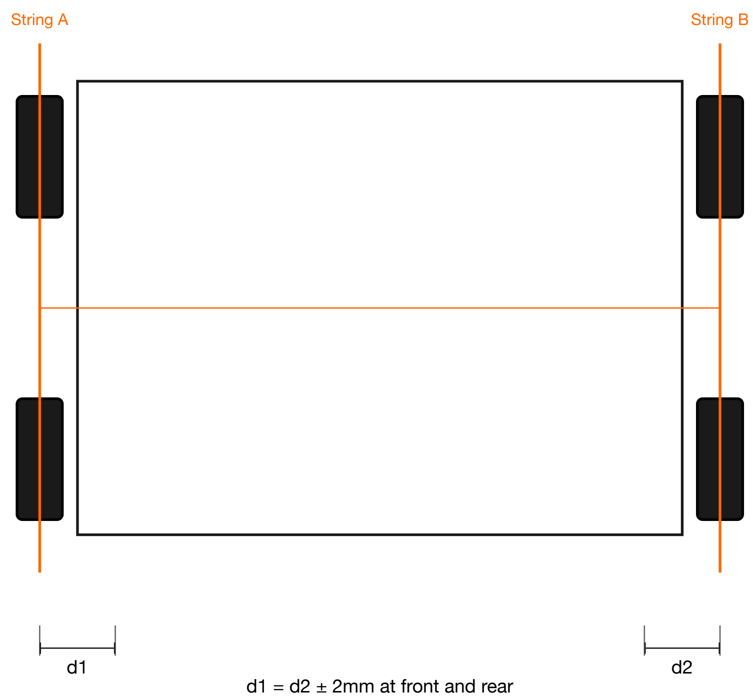


Figure 24: String alignment method. Stretch strings parallel to frame sides.

Alignment Procedure:

1. Stretch two parallel strings along frame sides
2. Measure gap from string to front wheel edge
3. Measure gap from string to rear wheel edge
4. Gaps should be equal (±2mm) on each side
5. If not equal: loosen motor bolts, slide motor in T-slot, re-tighten

Common Issues:

Symptom	Cause
Fix	Rover pulls left
Right wheels toe-in	Slide right motors outward
Rover pulls right	Left wheels toe-in
Slide left motors outward	Excessive tire wear
Wheels not parallel	Realign all motors
Vibration at speed	Wheel out of round
Replace tire or motor	

> ℹ **NOTE** The T-slot mounting makes alignment adjustable. This is one advantage of the direct-mount approach: loosen two bolts, slide, retighten.

Power System

The rover runs on 48V nominal (13S lithium). This voltage is high enough to be efficient (less current means thinner wires and less heat) but low enough to avoid the regulatory complexity of “high voltage” systems.

BVR0 keeps the power system as simple as possible: battery → distribution bus → loads. The battery’s integrated BMS handles overcurrent and undervoltage protection. No separate inline fuses.

Respect the battery. A 48V 20Ah pack stores nearly 1 kWh of energy. That’s enough to weld metal if shorted, or start a fire if punctured. The safety section covers handling in detail.

> i **NOTE** BVR0 has an E-stop on the sensor mast but no inline fuse. The E-stop cuts power to all motors. BVR1 adds fuses and a relay-based e-stop with watchdog.

Battery Mounting

Mount downtube battery to frame — 15 min

BVR0 uses an off-the-shelf 48V downtube-style e-bike battery. These batteries have an integrated mounting system: a bracket bolts to the frame, and the battery slides in and locks. No custom fabrication required.

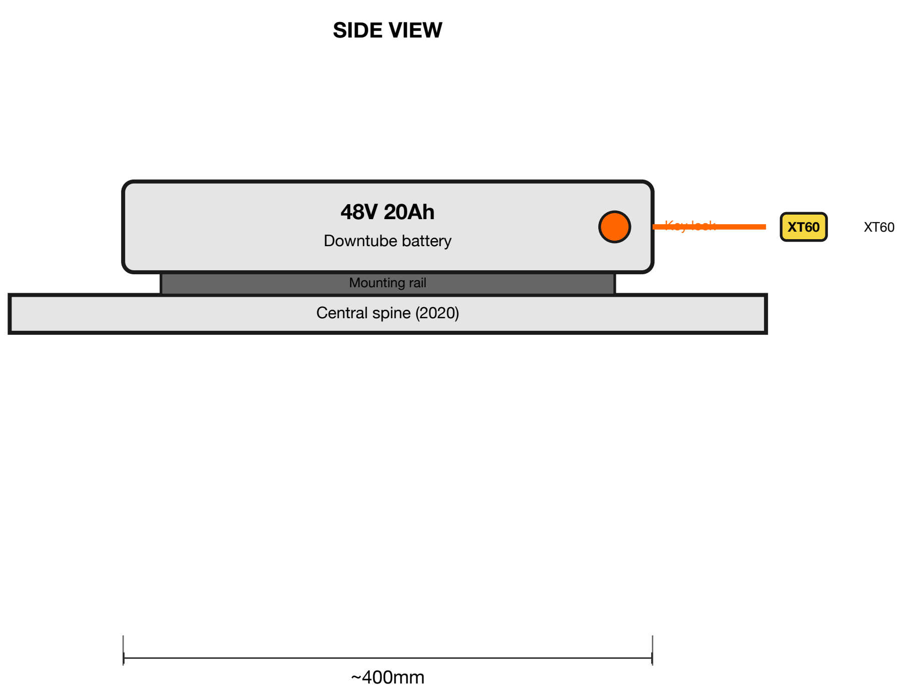


Figure 25: Downtube battery slides onto rail mounted to central 2020 spine.

Battery Requirements:

- 48V nominal (13S lithium)
- 15-20Ah capacity
- Downtube/bottle mount style
- Integrated BMS
- XT60 or similar output connector

Common sources:

- Unit Pack Power (AliExpress)
- Luna Cycle
- EM3ev

Mounting:

- Battery rail bolts directly to 2020 T-slot
- Use M5 bolts + T-nuts (2-4 per rail)
- Rail position: center of frame, lengthwise
- Battery slides in from one end, locks with key

Why downtube? Off-the-shelf, replaceable, keyed lock, integrated BMS, weather-resistant housing.

> i **NOTE** No custom battery tray needed. The downtube battery's integrated mounting system is purpose-built for this. One less custom part.

Safety Considerations

BVR0 has a physical E-stop button on the sensor mast and relies on the battery's BMS for overcurrent protection. This is sufficient for prototype testing and supervised operation.

What BVR0 has:

- E-stop button (physical kill on mast)
- BMS overcurrent protection (60-100A)
- BMS undervoltage cutoff
- BMS short circuit protection
- Cell balancing

What BVR1 adds:

- Inline fuse (wire protection)
- E-stop relay (software-controlled kill)
- Watchdog timer (auto-stop on software crash)
- Headlights and tail lights

Emergency Shutdown (BVR0):

1. Press E-stop button on sensor mast (cuts power to motors)
2. Or: remove battery key and slide battery off rail
3. Or: disconnect XT60 at battery output

> i **NOTE** The E-stop button is wired directly in the motor power path. Pressing it immediately cuts power to all VESCs. No software required.

DC-DC Converter

Install voltage regulator 10 min

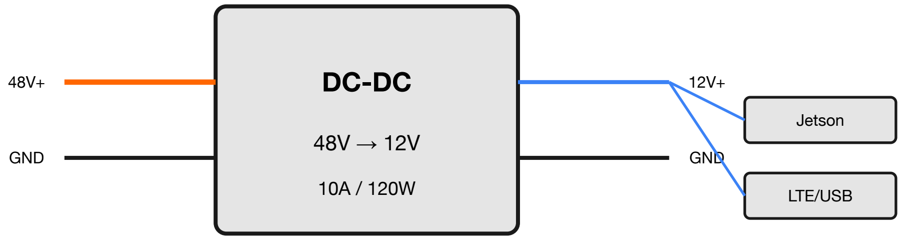


Figure 26: DC-DC powers all 12V devices from the 48V main bus.

Specifications:

Parameter	Value
Input voltage	36-60V (fits 13S LiPo range)
Output voltage	12V regulated
Output current	10A continuous
Efficiency	>90%
Mounting	M3 holes, heatsink on bottom

12V Load Budget:

Device	Current
Jetson Orin NX	5A peak, 3A average
LTE modem	1A
USB hub	0.5A
Accessories	1A reserve
Total	**6A typical, 10A max**

Power Distribution

Wire power bus 30 min

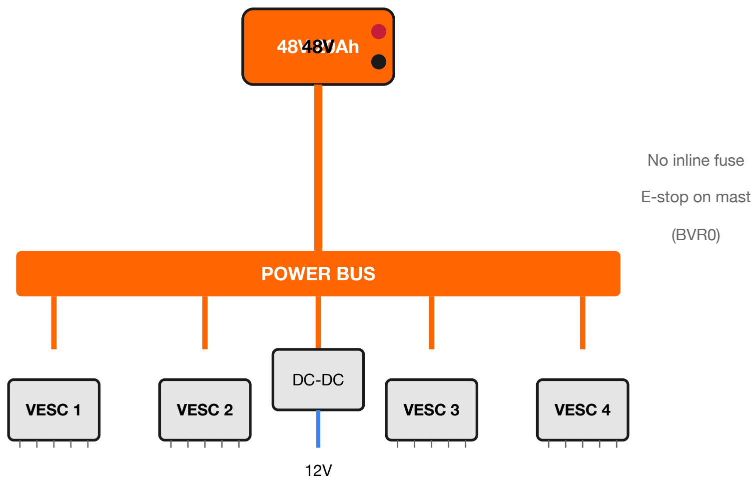


Figure 27: BVR0 power topology: battery direct to bus. Simple but minimal protection.

Bus Options:

Splitter Cable (BVR0):
- XT60 to 5× XT60 (or solder joints)
- Simple, no fabrication
- Higher resistance, less tidy

Bus Bar (BVR1):
- Solid copper bar with tapped holes
- Clean, low resistance
- Integrates fuse and e-stop relay

Electronics

The electronics are the nervous system: motor controllers that translate commands into wheel motion, a compute module that runs the autonomy stack, and a CAN bus that ties everything together.

We use VESC motor controllers because they're open-source, powerful, and have a decade of real-world use in electric skateboards and robotics. The Jetson Orin NX handles perception and planning. It's overkill for teleoperation, but essential for autonomous operation.

VESC Mounting

Mount motor controllers 20 min

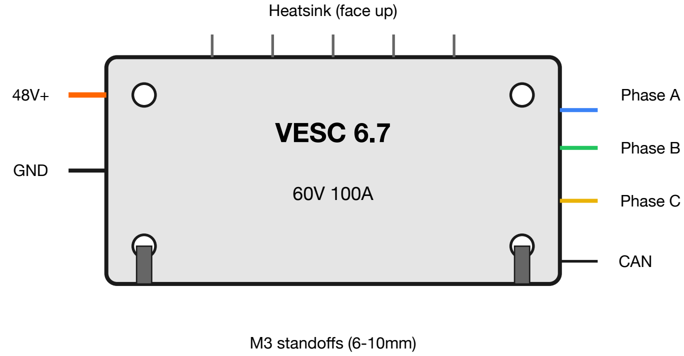


Figure 28: VESC mounted on standoffs for airflow. Heatsink faces up.

Mounting:
- M3×6 standoffs at all 4 corners
- M3×8 bolts through plate into standoffs
- Thermal pad between VESC and plate (optional, for heat transfer)

Power Connections:
- 10 AWG wire for 48V input
- XT60 connectors recommended
- Keep power wires short

⚡ **COMMON MISTAKE** VESCs generate serious heat under load. Without standoffs for airflow, thermal throttling kicks in after 3 minutes of hard driving.

VESC Configuration

Set CAN IDs and motor parameters

10 min per VESC

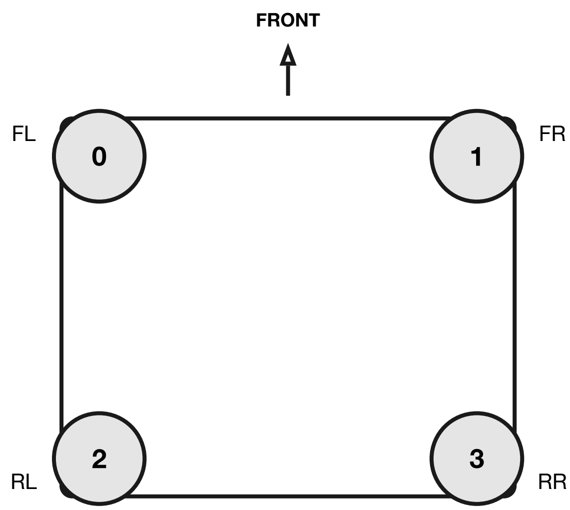


Figure 29: CAN ID assignment. ID 0-3 for wheels, ID 10+ for tools.

VESC Tool Configuration:

Parameter	Value
Controller ID	0, 1, 2, 3 (unique per VESC)
CAN Mode	VESC
CAN Baud Rate	CAN_500K
Send CAN Status	Enabled
CAN Status Rate	50 Hz
Motor Type	BLDC or FOC (depends on motor)
Current Limit	30A (per motor)

Motor Detection:

1. Connect VESC to computer via USB
2. Open VESC Tool
3. Run Motor Detection wizard
4. Save configuration to VESC
5. Disconnect USB, connect CAN

Jetson Mounting

Install compute module 15 min

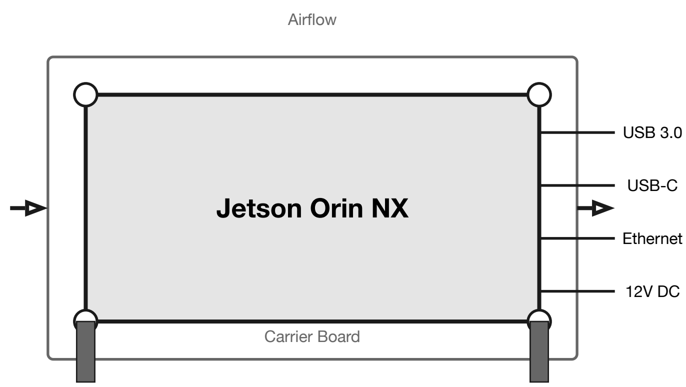


Figure 30: Jetson mounted on standoffs. Ensure airflow around heatsink.

Connections:

Port	Connection
CAN pins	CAN transceiver module
USB 3.0	USB hub (camera, LTE)
12V DC	From DC-DC converter
GPIO	Not used (e-stop is hardwired)

Software:
- JetPack 6.0 or later
- bvrd daemon (auto-start on boot)
- Insta360 SDK for camera

Jetson Setup:

```
# Flash JetPack 6.0 using SDK Manager on host PC
# After boot, install dependencies:
sudo apt update && sudo apt install -y can-utils
pip install pyserial

# Clone firmware repo
git clone https://github.com/muni-works/muni
cd muni/bvr/firmware

# Build and install bvrd
cargo build --release
sudo cp target/release/bvrd /usr/local/bin/
```

```
sudo cp config/bvrd.service /etc/systemd/system/
sudo systemctl enable bvrd
```

Full setup instructions at github.com/muni-works/muni.

GPIO Pinout

Reference: GPIO connections 5 min

BVR0 uses minimal GPIO: just the CAN interface on the carrier board. The E-stop is wired directly in the power path (not relay-controlled). BVR1 adds GPIO-controlled e-stop relay and watchdog.

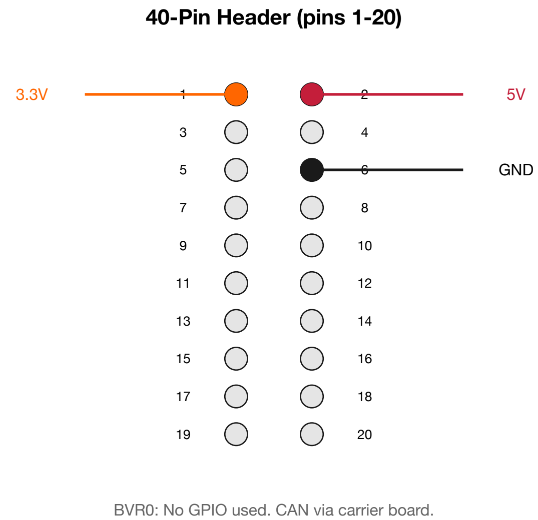


Figure 31: GPIO header reference. BVR0 uses carrier board CAN, not GPIO.

BVR0 GPIO Usage:

BVR0 doesn't use any GPIO pins directly. The CAN bus is provided by the carrier board's dedicated CAN controller (not bit-banged GPIO).

Pin	Function
BVR0	**BVR1**
1	3.3V
Unused	Status LED
2	5V
Unused	Unused
6	GND
Unused	E-Stop ground
32	GPIO12
Unused	E-Stop relay

i **NOTE** BVR1 adds e-stop relay on GPIO12. See BVR1 manual for wiring details. The relay is normally-open: GPIO LOW = motors disabled (fail-safe).

CAN Transceiver

Connect CAN bus interface 10 min

The Jetson carrier board (Seeed reComputer J401 or similar) has CAN controller pins exposed. A CAN transceiver module converts these logic-level signals to the differential CAN bus.

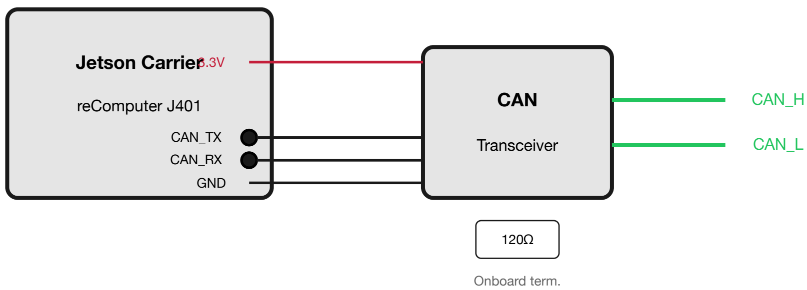


Figure 32: CAN transceiver connects carrier board CAN pins to differential bus.

Transceiver Modules:

- Waveshare SN65HVD230 module
- Any 3.3V CAN transceiver breakout
- Often included on carrier board (check your model)

Wiring:

Carrier Pin	Transceiver
Notes	CAN_TX
TXD	Logic level out
CAN_RX	RXD
Logic level in	3.3V
VCC	Power
GND	GND
Common ground	

Configuration:

```
# Enable CAN interface (may vary by carrier board)
sudo modprobe mttcan
sudo ip link set can0 type can bitrate 500000
sudo ip link set can0 up

# Test with candump
candump can0
```

Termination:

- Most transceiver modules have onboard 120Ω termination
- CAN bus needs exactly 2 termination resistors (one at each end)
- If transceiver is at bus end: enable termination
- Verify: measure 60Ω across CAN_H/CAN_L (two 120Ω in parallel)

> ⚡ **COMMON MISTAKE** The reComputer carrier uses the Jetson's native CAN controller (mttcan driver), not a USB adapter. Check your carrier's pinout for CAN_TX/CAN_RX locations.

Sensor Mast

The sensor mast is the rover's eyes. It elevates the LiDAR and 360° camera above the chassis to get an unobstructed view of the world.

Height matters. Too low and the sensors see mostly wheels. Too high and the mast becomes a sail in the wind and a lever arm for tip-overs. We settled on 700mm total height (from ground to camera) as a good compromise for sidewalk-scale operation.

The mast is intentionally simple: a tube, a clamp, and some brackets. If it gets bent (it will), you can straighten it or replace it in minutes.

Sensor Mast Assembly

Build sensor pole 20 min

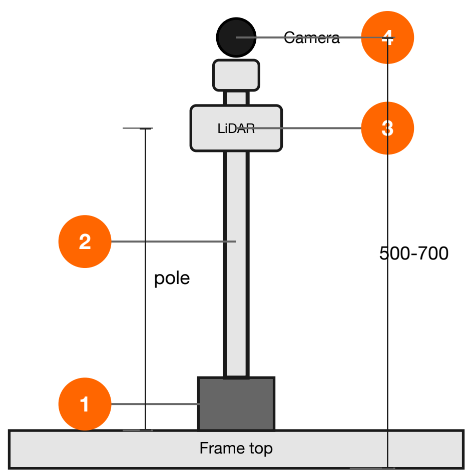


Figure 33: Sensor mast with LiDAR below camera for unobstructed 360° view.

Components:

1. Pole mount bracket
2. Carbon fiber or aluminum tube
3. LiDAR mount plate
4. Camera mount (1/4-20)

Pole Specifications:

- Diameter: 25-30mm OD
- Material: Carbon fiber (light) or 6061-T6 aluminum
- Length: 400-600mm depending on design
- Wall thickness: 2mm minimum

LiDAR Mounting

Mount LiDAR sensor 15 min

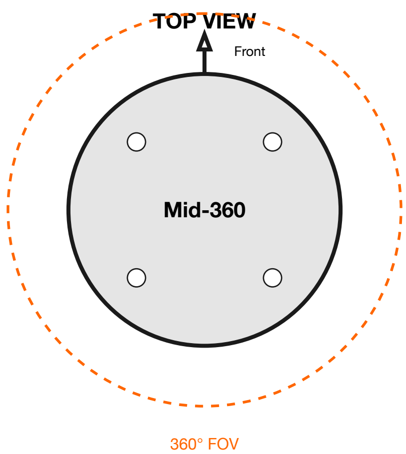


Figure 34: Top view: full 360° sensing

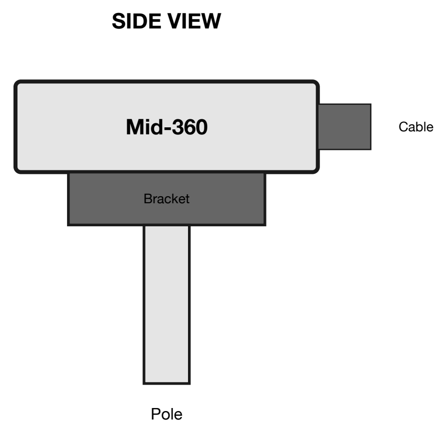


Figure 35: Side view: cable routes down

Installation:

1. Attach LiDAR to mount plate with M3 bolts
2. Level the mount plate (use spirit level)
3. Secure mount plate to pole with hose clamps or bolts
4. Route cable inside pole or along outside with ties
5. Connect to Jetson via Ethernet

Orientation:

- LiDAR "front" should face rover front
- Ensure level within ±1°
- No obstructions in 360° view

Camera Mounting

Mount 360° camera 10 min

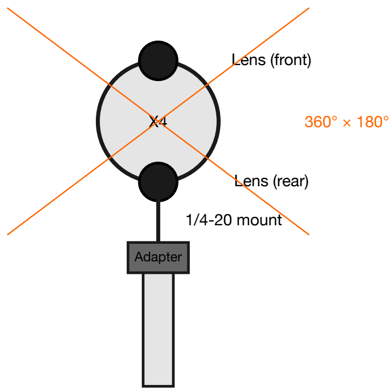


Figure 36: Camera at mast top. Dual lenses capture full spherical view.

Mount Options:

- 1/4-20 threaded insert in pole top cap
- GoPro-style mount adapter
- Custom 3D-printed adapter

Cable Routing:

- USB-C cable to Jetson
- Route inside pole if possible
- Secure with cable ties
- Leave strain relief loop at camera

Camera Settings:

Setting	Value
Mode	Live streaming (H.265)
Resolution	4K or 5.7K
Frame rate	30 fps
Stabilization	FlowState (on)

Wiring

Wiring is where builds go wrong. A loose connection causes intermittent failures. A reversed polarity destroys components. A pinched wire works fine until it doesn't.

Take your time here. Label every wire. Double-check polarity before applying power. Use the correct gauge wire for the current it carries. The extra hour spent on clean wiring saves days of debugging later.

The schematic on the next page shows the complete system. Study it before you start, and refer back to it often.

System Wiring Schematic

Reference: complete wiring diagram reference

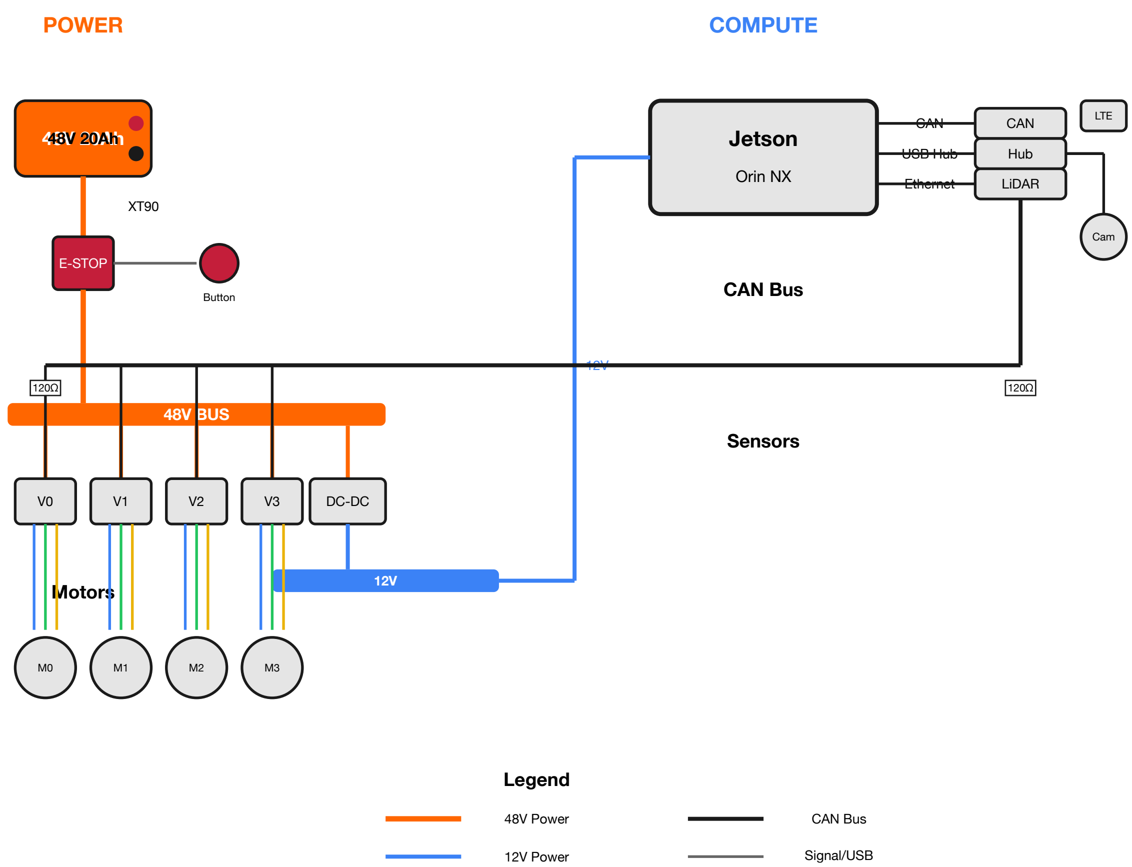


Figure 37: Complete system wiring. Power flows left, compute/sensors right.

CAN Bus Wiring

Wire CAN bus network 30 min

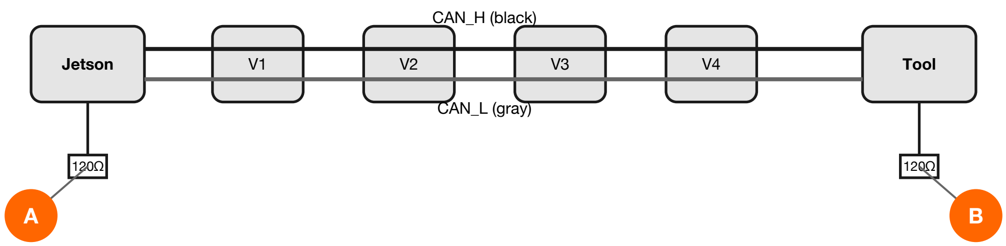


Figure 38: CAN bus with 120Ω termination at each end (A and B).

Wiring Rules:

- Use twisted pair wire (22 AWG)
- CAN_H to CAN_H, CAN_L to CAN_L at each device
- Maximum total bus length: 40m at 500K baud
- Exactly 2 termination resistors (one at each end)
- Keep CAN wires away from motor phase wires

JST Connector Pinout:

Pin	Signal
Color (typical)	1
GND	Black
2	CAN_L
Gray or White	3
CAN_H	Orange or Yellow
4	+5V (optional)
Red	

Motor Phase Wiring

Connect motor phase wires 15 min per motor

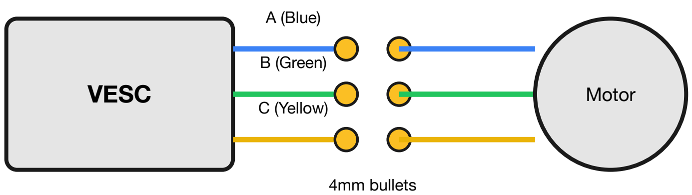


Figure 39: Phase wires connect VESC to motor via bullet connectors.

Connection Notes:

- Motor wire colors may not match VESC colors
- If motor spins wrong direction: swap any two phase wires
- Use 4mm gold bullet connectors (60A rated)
- Solder connections, use heat shrink
- Keep phase wires away from signal wires (EMI)

Wire Lengths:

Motor Position	Approx. Length
Front Left	400mm
Front Right	500mm
Rear Left	300mm
Rear Right	400mm

i **NOTE** Add 50mm extra for service loops. Too tight = strain on connectors.

⚡ **COMMON MISTAKE** Phase wire order doesn't matter for direction (any two can be swapped). But inconsistent colors across all 4 motors makes troubleshooting a nightmare. Pick a convention and stick to it.

Signal Wiring

Route USB and signal cables 20 min

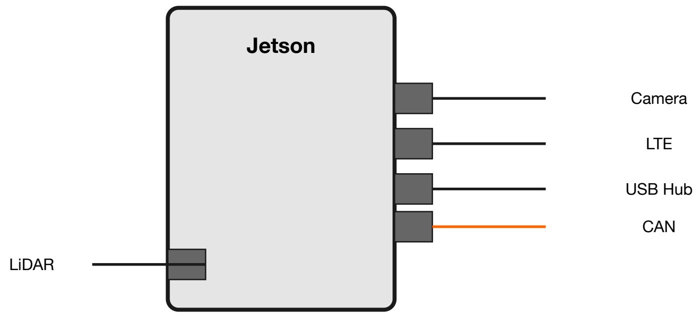


Figure 40: Jetson connections. USB for peripherals, CAN for VESCs, Ethernet for LiDAR.

Port Allocation:

Port	Device
Cable	CAN pins
CAN transceiver → VESCs	JST → bus
USB 3.0 1	USB Hub
USB-A to hub	Hub Port 1
Insta360 X4	USB-C
Hub Port 2	LTE modem
USB-A	Ethernet
Livox LiDAR	RJ45

E-Stop (BVR0):

- Hardwired switch on sensor mast
- Wired in series with motor power
- No GPIO control (direct disconnect)

Cable Management

Secure and organize all wiring

30 min

Frame interior (top view)

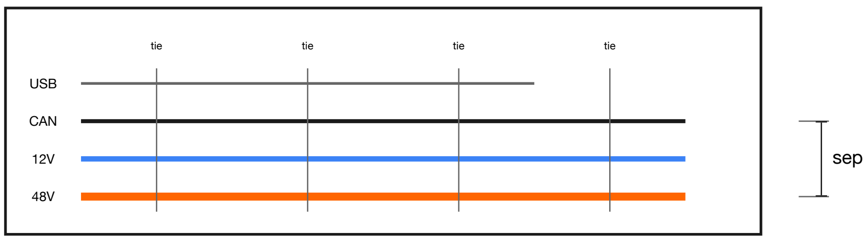


Figure 41: Route power and signal cables separately. Secure every 150mm.

Routing Rules:

- Separate power (48V) from signals by at least 25mm
- CAN bus twisted pair reduces interference
- Use cable ties every 100-150mm
- Leave service loops at connectors
- Label both ends of each cable

Cable Tie Points:

- Frame corners
- Near each connector
- Before/after bends
- At entry to electronics bay

- [ ] No cables in wheel path
- [ ] No cables near hot components (VESCs)
- [ ] All connectors accessible
- [ ] Service loops at key points
- [ ] Labels on power cables

> *"We learned this the hard way:" Beautiful cable management is great until you need to replace a VESC. Leave enough slack to pull components 10cm out for service without disconnecting everything upstream.*

Testing

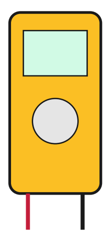
This is the moment of truth. You've built the mechanical system, wired the electronics, and configured the software. Now you find out if it all works together.

Go slow. The pre-power checks exist because we've made every mistake in this section at least once. A reversed connector, a missed termination resistor, a VESC with the wrong ID, these are easy to fix **before** you apply power and much harder after.

The first power-up should be anticlimactic. If you smell burning, hear crackling, or see smoke: power off immediately. Something is wrong, and continuing will make it worse.

Pre-Power Checks

Verify connections before first power-on 10 min



Multimeter Tests

Continuity Tests (power OFF):

Test	Probe Points
Expected	48V+ to GND
Battery connector pins	Open (no beep)
12V+ to GND	DC-DC output
Open (no beep)	CAN_H to CAN_L
CAN connector	60Ω (two 120Ω in parallel)
Phase A to B	Motor connector
Low resistance (motor windings)	

Visual Inspection:

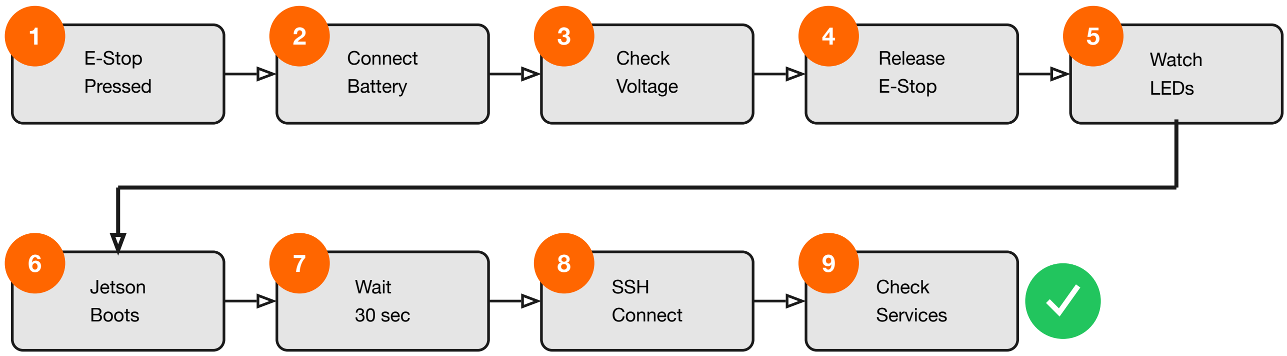
- [ ] No exposed wire or bare conductors

- [ ] All connectors fully seated
- [ ] Polarity correct (red to +, black to -)
- [ ] No pinched wires
- [ ] E-Stop button in pressed (safe) position
- [ ] Battery key removed

> ⚠️ **DANGER** If any continuity test shows a short (beep) between power and ground, DO NOT APPLY POWER. Find and fix the short first.

First Power-Up

Initial power-on sequence 5 min



What to Watch:

Indicator	Normal
Problem	VESC LEDs
Solid green	Red = fault, none = no power
Jetson LED	Solid then blinking
None = power issue	DC-DC LED
Green (if equipped)	None = input voltage issue
Smell	None
Burning = immediate power off	Sound
Quiet hum	Buzzing = loose connection

VESC Configuration

Configure motor controllers 15 min per VESC ●●●

Configure motor controllers using VESC Tool.

▸ VESC Setup Walkthrough

Connection:

1. Connect laptop to VESC via USB
2. Open VESC Tool
3. Select serial port, click Connect

Motor Wizard:

1. Navigate to Motor → Motor Wizard
2. Select motor type (usually "Large outrunner")
3. Run detection: VESC will spin motor briefly
4. Review detected parameters
5. Write configuration to VESC

CAN Configuration:

1. Navigate to App → CAN Status
2. Set unique Controller ID (0, 1, 2, 3)
3. Set CAN Baud to 500K
4. Enable "Send CAN Status"
5. Write configuration

Per-VESC Settings:

VESC	ID
Motor Direction	Front Left
0	Forward = CCW
Front Right	1
Forward = CW	Rear Left
2	Forward = CCW
Rear Right	3
Forward = CW	

ℹ **NOTE** Left and right motors spin opposite directions for forward motion in skid-steer.

⚡ **COMMON MISTAKE** Swapping VESC IDs 0↔1 or 2↔3 makes the rover spin in circles instead of driving straight. The CAN ID diagram on the Quick Reference Card saves debugging time.

Motor Testing

Verify motor response

10 min ●●●

⚠ **WARNING** Elevate rover so all wheels are off the ground before motor testing.

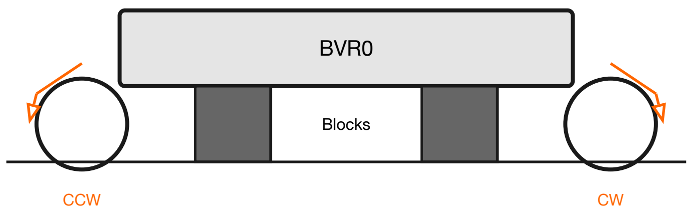


Figure 44: Test with wheels elevated. Verify each motor spins correct direction.

Test Procedure:

1. Elevate rover on blocks (all wheels free)
2. Power on, release E-Stop
3. Connect controller
4. Command forward slowly: all wheels should spin “forward”
5. Command reverse: all wheels should spin “backward”
6. Command left turn: right wheels forward, left wheels backward
7. Test E-Stop: press button, verify immediate stop

Direction Fix: If a motor spins wrong direction, swap any two phase wires on that motor.

- [ ] All 4 motors respond to commands
- [ ] Direction correct for each motor
- [ ] E-Stop stops all motors immediately
- [ ] No unusual sounds or vibration
- [ ] VESCs not overheating

“We learned this the hard way:” The first road test had the rover on flat ground. It took off at 2 m/s and hit a wall. Always test on blocks first, with throttle at minimum.

Operation

Operating the rover is straightforward once it's built and tested. The startup sequence takes about 3 minutes. Shutdown takes 1 minute. Most of that time is waiting for the Jetson to boot.

The key habit is consistency. Use the same startup sequence every time. Check the same indicators. Park in the same spot. Consistent routines catch problems early, when they're small and easy to fix.

Startup Procedure

Power on and connect 3 min

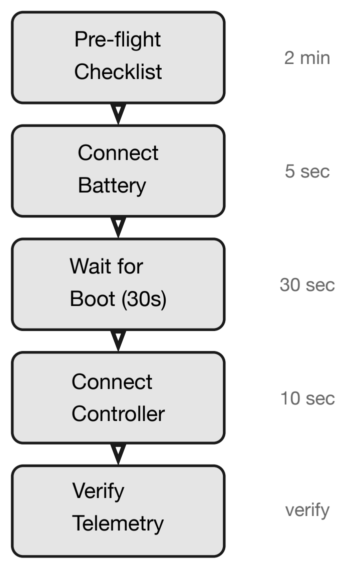


Figure 45: Startup takes approximately 3 minutes.

Detailed Steps:

1. **Pre-flight:** Complete checklist on page 2
2. **Battery:** Connect XT90 (hear click). E-Stop should be pressed.
3. **Boot:** Release E-Stop. Wait for Jetson to boot (30s). VESC LEDs turn green.
4. **Controller:** Power on controller. Connect to operator station.
5. **Telemetry:** Verify video feed, voltage reading, and mode indicator.

> i **NOTE** Do not operate if telemetry shows errors or video feed is absent.

Shutdown Procedure

Safe power-off sequence 1 min

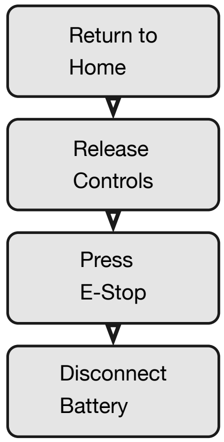


Figure 46: Always press E-Stop before disconnecting battery.

Shutdown Checklist:

- [ ] Rover parked in designated area
- [ ] Controller set down / powered off
- [ ] E-Stop button pressed (red button down)
- [ ] Wait 5 seconds for Jetson to save state
- [ ] Disconnect battery (pull XT90)
- [ ] Store battery properly (50-60% charge for long storage)

⚠ **WARNING** Never disconnect battery while Jetson is running. This can corrupt the filesystem.

Tool Attachment

Attach tools to chassis 15 min

BVR0 uses direct bolt mounting for tools: no quick-release, no modular interface. Tools attach to the 2020 extrusion frame using T-nuts and bolts, just like everything else on the rover.

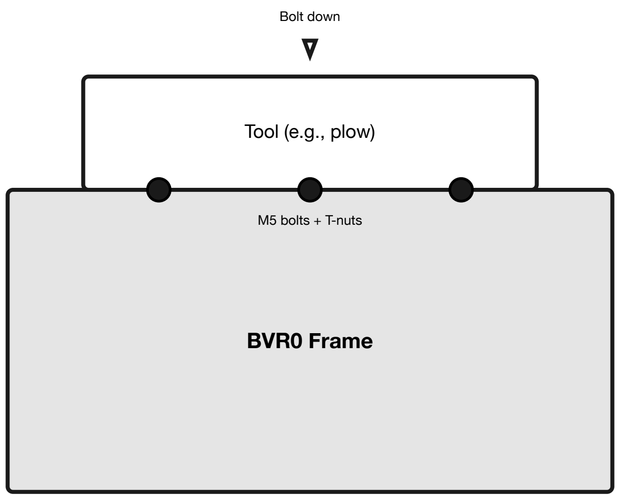


Figure 47: Tool bolts directly to frame using T-slot hardware.

Attachment Procedure:

1. Power OFF rover (E-Stop pressed)
2. Position tool on frame, align with T-slots
3. Insert M5 bolts through tool mounting holes
4. Thread into T-nuts in frame extrusion
5. Tighten with 4mm hex key (hand-tight + 1/4 turn)
6. Connect power cable (if tool is powered)
7. Power ON rover

Removal:

1. Power OFF rover
2. Disconnect power cable first
3. Loosen M5 bolts
4. Lift tool off frame

> i **NOTE** BVR1 adds a quick-release rail system with electrical pass-through. BVR0 keeps it simple: bolts work.

Safety

The BVR0 is a machine, and machines can hurt you. The hazards are real: spinning wheels that don't care about fingers, a battery that can catch fire, a 30kg robot that can pin you against a wall.

None of this is meant to scare you. These hazards are manageable with basic awareness and respect for the machine. The safety protocols in this section come from experience (some of it painful). Follow them.

Personal Protective Equipment

PPE requirements by task — reference

Task	Required PPE
Cutting extrusions	Safety glasses, work gloves
Soldering / wiring	Safety glasses, fume extraction
Battery handling	Insulated gloves, safety glasses
Motor testing	Safety glasses, hearing protection
Operation	None required (stay clear of rover)

Safe Operating Distance:

- Operator: minimum 2m from rover during teleop
- Bystanders: minimum 5m from operating rover
- During charging: check every 15 min, do not leave unattended

Hazard Zones

Know the danger zones reference

⚠ **DANGER** Stay clear of marked zones during operation. Serious injury possible.

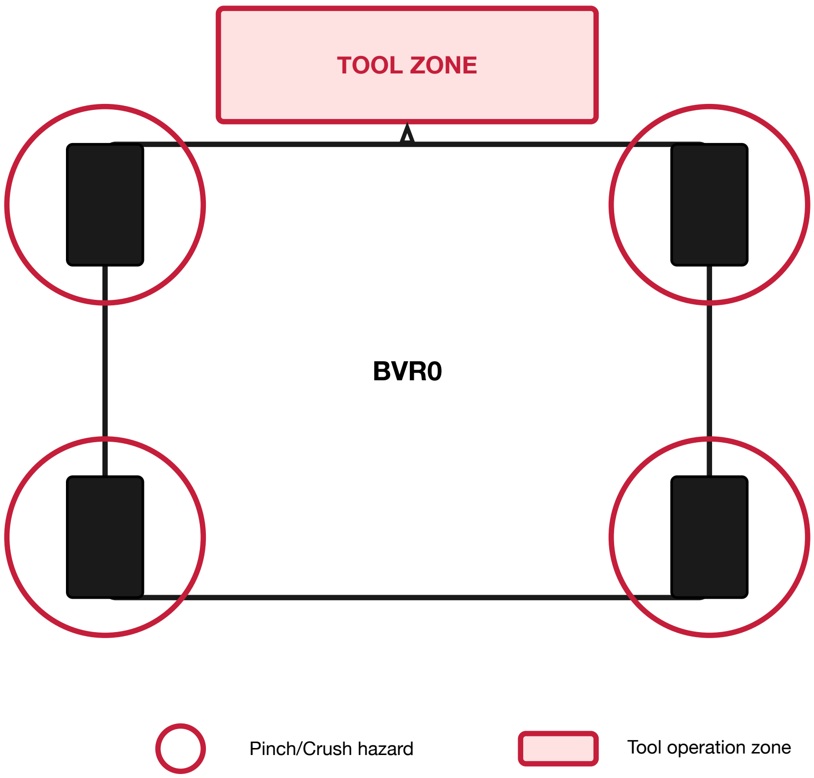


Figure 48: Keep hands, feet, and loose clothing clear of marked zones.

Hazard Types:

Zone	Hazard
Injury Type	Wheel areas
Rotating wheels, motor torque	Crush, pinch, friction burn
Tool zone	Rotating auger/blade
Laceration, amputation	Underside
50mm ground clearance	Crush if rover tips
Battery area	Electrical, thermal
Shock, burns	

Battery Safety

Handle lithium batteries safely reference

⚠️ **WARNING** Li-ion batteries can catch fire if damaged, punctured, or short-circuited.

DO		DON'T	
15-25°C	Store temp	No water	Keep dry
50-60%	Storage charge	No puncture	Protect case
Inspect	Before each use	No fire	Never burn

Signs of Battery Damage:

- Swelling or bulging
- Unusual heat
- Hissing or venting
- Visible damage to case
- Reduced capacity

In Case of Battery Fire:

1. **Evacuate** the immediate area (minimum 10m / 30ft)
2. Call fire department: **911**
3. If small and contained: use **CO2** or **ABC dry chemical** extinguisher
4. If large or spreading: **do not attempt to extinguish** (let professionals handle)
5. Ventilate area (toxic fluoride fumes)
6. Water in **large quantities** can cool adjacent cells and prevent spread, but small amounts can make it worse

⚠️ **WARNING** Li-ion fires re-ignite. Monitor for at least 1 hour after fire appears out. Do not move battery until cool.

⚠️ **DANGER** Never attempt to charge a damaged battery. Dispose at authorized battery recycling facility (Call2Recycle, Best Buy, etc.).

Firmware

The rover runs two firmware stacks: `bvrd` on the Jetson (the main brain), and embedded firmware on tool attachments (ESP32-based).

Most operators never need to touch firmware. It comes pre-flashed on shipped units, and updates are pushed over-the-air. This section is for those building from scratch or doing development work.

If you're comfortable with Linux command lines and embedded toolchains, this will be familiar. If not, follow the steps exactly, and don't skip the verification steps.

Initial Jetson Setup

Flash and configure Jetson 45 min ●●●

Full setup guide: github.com/muni-works/muni

1. Flash JetPack OS:

Download JetPack 6.0+ from NVIDIA. Flash using SDK Manager on Ubuntu host:

```
# On Ubuntu 20.04/22.04 host machine
sudo apt install nvidia-sdk-manager
sdkmanager  # GUI will launch
```

Select "Jetson Orin NX" and JetPack 6.0. Follow prompts to flash.

2. First Boot Configuration:

```
# Set hostname
sudo hostnamectl set-hostname bvr0

# Create muni user (if not done during setup)
sudo adduser muni
sudo usermod -aG sudo,dialout,video muni

# Enable SSH
sudo systemctl enable ssh
```

3. Install Dependencies:

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y can-utils build-essential \
    libclang-dev pkg-config libssl-dev

# Install Rust
curl --proto '=https' --tlsv1.2 -sSf \
    https://sh.rustup.rs | sh
source ~/.cargo/env
```

CAN Bus Setup

Configure CAN interface 10 min

1. Load CAN Modules:

```
# Add to /etc/modules-load.d/can.conf
echo "can" | sudo tee /etc/modules-load.d/can.conf
echo "can_raw" | sudo tee -a /etc/modules-load.d/can.conf
echo "slcan" | sudo tee -a /etc/modules-load.d/can.conf
```

2. Create Startup Service:

Create `/etc/systemd/system/can.service`:

```
[Unit]
Description=CAN Bus Interface
After=network.target

[Service]
Type=oneshot
RemainAfterExit=yes
ExecStart=/sbin/ip link set can0 type can bitrate 500000
ExecStart=/sbin/ip link set can0 up
ExecStop=/sbin/ip link set can0 down

[Install]
WantedBy=multi-user.target
```

```
sudo systemctl enable can.service
sudo systemctl start can.service
```

3. Verify CAN:

```
# Should show can0 interface
ip link show can0

# Monitor CAN traffic (VESCs should send status)
candump can0
```

LiDAR Setup

Configure LiDAR network 15 min

1. Network Configuration:

The Mid-360 uses a static IP. Configure the Jetson Ethernet:

```
# Add to /etc/netplan/01-lidar.yaml
network:
  version: 2
  ethernets:
    eth0:
      addresses:
        - 192.168.1.50/24
      routes:
        - to: 192.168.1.0/24
          via: 192.168.1.1
```

```
sudo netplan apply
```

2. LiDAR Default Settings:

Parameter	Value
LiDAR IP	192.168.1.1xx (xx = last 2 of serial)
Host IP	192.168.1.50
Data Port	56000
Command Port	56001

3. Test Connection:

```
# Ping LiDAR (replace with your unit's IP)
ping 192.168.1.100

# Install Livox SDK2 for testing
git clone https://github.com/Livox-SDK/Livox-SDK2
cd Livox-SDK2 && mkdir build && cd build
cmake .. && make -j4
```

bvrd Installation

Install rover daemon 15 min

1. Clone Repository:

```
cd /opt
sudo mkdir muni && sudo chown muni:muni muni
git clone https://github.com/muni-works/bvr.git
cd bvr/firmware
```

2. Build:

```
cargo build --release
sudo cp target/release/bvrd /opt/muni/bin/
```

3. Install Service:

```
sudo cp config/bvrd.service /etc/systemd/system/
sudo systemctl daemon-reload
sudo systemctl enable bvrd
sudo systemctl start bvrd
```

4. Verify:

```
sudo systemctl status bvrd
journalctl -u bvrd -f
```

Firmware Overview

The rover uses two main firmware components.

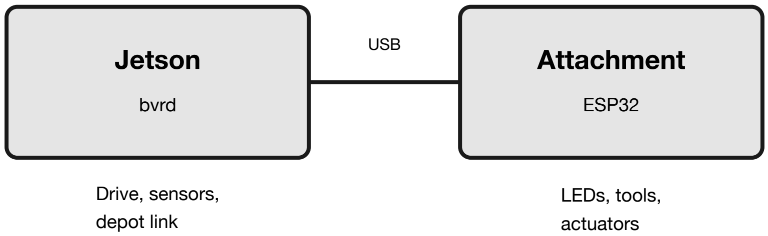


Figure 50: Firmware architecture: Jetson runs bvrd, attachments run on ESP32.

bvrd (Jetson)

- Main rover daemon
- Motor control via CAN
- Sensor fusion
- Depot communication
- Attachment discovery

Attachment (ESP32)

- Tool-specific firmware
- LED control
- Local sensors
- SLCAN protocol
- Status heartbeat

Updating bvrd

Deploy firmware updates 5 min

Prerequisites:

- SSH access to Jetson
- Rust toolchain with aarch64 target

Build and Deploy:

```
# On development machine
cd bvr/firmware
cargo build --release --target aarch64-unknown-linux-gnu

# Copy to Jetson
scp target/aarch64-unknown-linux-gnu/release/bvrd \
    muni@<jetson-ip>:/opt/muni/bin/

# On Jetson - restart service
ssh muni@<jetson-ip>
sudo systemctl restart bvrd
```

Verify:

```
# Check service status
sudo systemctl status bvrd

# View logs
journalctl -u bvrd -f
```

Updating Attachment Firmware

Flash ESP32 attachments 10 min

Prerequisites:

- ESP32 Rust toolchain (`espup install`)
- `espflash` tool installed
- Attachment connected via USB

Build:

```
cd mcu/bins/esp32s3
source ~/export-esp.sh
cargo build --release
```

Flash:

```
espflash flash \
    --ignore-app-descriptor \
    --partition-table partitions.csv \
    --bootloader bootloader.bin \
    --min-chip-rev 0.0 \
    target/xtensa-esp32s3-none-elf/release/mcu-esp32s3
```

Monitor Serial Output:

```
espflash monitor
# Or: screen /dev/cu.usbserial-0001 115200
```

If flash fails:

1. Unplug USB
2. Hold BOOT button
3. Plug in USB (keep holding)
4. Release after 2 seconds
5. Retry flash command

Attachment Protocol

Attachments communicate via SLCAN (CAN-over-serial).

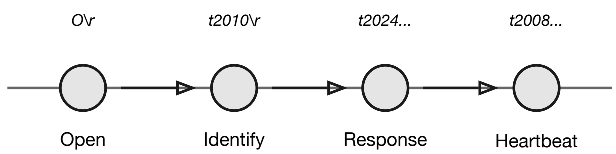


Figure 51: Discovery flow: open channel, identify, then heartbeat begins.

CAN Message IDs (Attachment Slot 0):

ID	Direction
Purpose	0x200
Attach → Host	Heartbeat (1Hz)
0x201	Host → Attach
Identify request	0x202
Attach → Host	Identity response
0x203	Host → Attach
Command	0x204
Attach → Host	Acknowledgment

Text Commands (for debugging):

```
led 255,0,0    # Set LED red
cycle          # Rainbow cycle mode
state running  # Set state
help           # Show commands
```

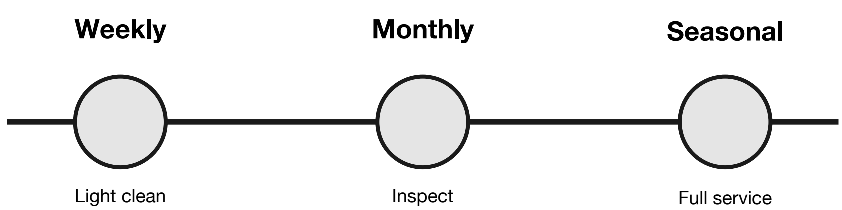
Maintenance

A well-maintained rover is a reliable rover. The maintenance tasks are simple: clean it, inspect it, keep the bolts tight and the battery healthy.

The schedule below is based on real-world operation in Cleveland conditions (salt, snow, mud, temperature swings). If you operate in a milder environment, you can extend the intervals. If you're running daily in harsh conditions, shorten them.

Maintenance Schedule

Preventive maintenance overview varies



Weekly

- Clean wheels and chassis
- Wipe camera lens
- Wipe LiDAR lens
- Check connector seating
- Verify wheel spin
- Test E-Stop function

Monthly

- Inspect all wiring
- Check bolt torque
- Clean electrical contacts
- Check battery health
- Update firmware
- Review error logs

Seasonal

- Full electrical inspection
- Check wheel bearings
- Replace worn tires
- Deep clean chassis
- Calibrate sensors
- Battery capacity test

Troubleshooting

Diagnose common issues varies

Symptom	Likely Cause
Solution	Won't power on
Battery disconnect	Check XT90 connection, verify fuse
No video feed	Camera USB
Reconnect camera, check USB hub power	Motor not responding
CAN wiring	Check CAN connections, verify VESC ID
Erratic movement	VESC ID mismatch
Verify IDs match wheel positions	E-Stop won't release
Button stuck	Check relay wiring, verify mechanism
Overheating	Ventilation blocked
Clean vents, reduce load	Poor LTE signal
Antenna position	Reposition antenna, check SIM
Battery dies quickly	Battery age
Check cell balance, replace if needed	Jerky motion
Motor calibration	Re-run VESC motor detection
Drift to one side	Wheel alignment
Re-align motor brackets	

Diagnostic Commands:

```
# Check system status
bvr status

# List CAN devices
bvr can scan

# Test individual motor
bvr motor test <id>

# View recent logs
journalctl -u bvrd -n 100
```

▸ Troubleshooting Walkthrough

Storage

Store rover properly 5 min

50-60%	Disconnect	15-25°C
Battery charge	Unplug battery	Temperature

Short-Term Storage (< 1 week):

- [ ] Press E-Stop
- [ ] Disconnect battery
- [ ] Cover if stored outdoors

Long-Term Storage (> 1 week):

- [ ] Charge battery to 50-60%
- [ ] Disconnect battery completely
- [ ] Clean chassis and wheels
- [ ] Cover camera and LiDAR lenses
- [ ] Store in dry location (15-25°C)
- [ ] Check battery monthly (recharge if < 40%)

Returning from Storage:

1. Inspect for moisture, corrosion, pest damage
2. Charge battery fully
3. Run pre-flight checklist
4. Test all functions before field use

Bill of Materials

This is everything you need to build one BVR0 from scratch. Prices are approximate as of late 2025 and vary by region and vendor.

A few notes on sourcing: the expensive items (Jetson, LiDAR, camera) are worth buying from authorized distributors for warranty support. The commodity items (extrusions, fasteners, wire) can come from anywhere. The hub motors are sourced from AliExpress because that's where they're cheapest; allow 2-3 weeks for shipping.

The total cost ($4,200) is roughly split: one-third mechanical, one-third power/drive, one-third compute/sensors. If you're building multiple units, the mechanical and power costs drop significantly with bulk ordering.

Structural

Part	Qty	Unit	Total	Source
2020 Aluminum Extrusion, 1m	6	$8	$48	Amazon / 8020.net
90° Corner Bracket	16	$1.50	$24	Amazon
M5×10 Button Head Bolt	64	$0.15	$10	McMaster
M5 T-Nut (drop-in)	64	$0.20	$13	Amazon
M5 T-Nut (pre-load)	32	$0.15	$5	Amazon
25mm Aluminum Tube (mast)	1	$15	$15	Local metal supplier

Drivetrain

Part	Qty	Unit	Total	Source
Hub Motor 350W 48V	4	$85	$340	AliExpress / ODrive
VESC 6.7 (Flipsky)	4	$120	$480	Flipsky
M5×12 Button Head Bolt	8	$0.15	$2	McMaster
4mm Bullet Connectors	24	$0.50	$12	Amazon

Power

Part	Qty	Unit	Total	Source

48V 20Ah Downtube Battery	1	$400	$400	Unit Pack Power / Luna
E-Stop Mushroom Button	1	$15	$15	Amazon
XT60 Connector Pair	8	$2	$16	Amazon
DC-DC 48V→12V 10A	1	$35	$35	Amazon
10 AWG Silicone Wire (red)	3m	$2/m	$6	Amazon
10 AWG Silicone Wire (black)	3m	$2/m	$6	Amazon

Electronics

Part	Qty	Unit	Total	Source
Jetson Orin NX 16GB	1	$600	$600	Seeed / Arrow
Carrier Board w/ CAN	1	$130	$130	Seeed / Waveshare
USB 3.0 Hub (powered)	1	$25	$25	Amazon
LTE Modem (USB)	1	$50	$50	Amazon
22 AWG Wire (CAN, assorted)	10m	$0.50/m	$5	Amazon
JST-XH 4-pin Connector	10	$0.50	$5	Amazon
Electrical Tape	2 rolls	$3	$6	Amazon
Velcro Strips	1 pack	$8	$8	Amazon

Sensors

Part	Qty	Unit	Total	Source
Livox Mid-360 LiDAR	1	$1,000	$1,000	Livox
Insta360 X4 Camera	1	$500	$500	Insta360
RTK GPS Module (optional)	1	$200	$200	SparkFun

Miscellaneous

Part	Qty	Unit	Total	Source
Zip Ties (assorted)	1 pack	$8	$8	Amazon
Heat Shrink Tubing	1 kit	$12	$12	Amazon
Loctite 243 (blue)	1	$8	$8	Amazon
Dielectric Grease	1	$6	$6	Amazon
Cable Sleeve (split loom)	5m	$1/m	$5	Amazon

Subtotals

Structural	$115
Drivetrain	$834
Power	$478
Electronics	$829
Sensors	$1,700

Total: $3,995

Excludes tools, shipping, and taxes.
Prices vary by region and vendor.

Misc	$39

Hardware Reference

1:1 SCALE — Print at 100%

Print this page at 100% scale. Use to verify hardware sizes.

Bolt Sizes (Socket Head Cap Screw)

M3 × 8	M4 × 10	M5 × 12	M5 × 16
Electronics	Motors	Frame	Standoffs

Washer and Nut Sizes

M3 washer | M4 washer | M5 washer | M5 nut

Wire Gauges (Cross-Section)

10 AWG	18 AWG	22 AWG
48V Power	12V	CAN/Signal

Connector Sizes

> ℹ **NOTE** If hardware doesn't match these silhouettes at 100% print scale, verify your print settings. Some PDF viewers default to "Fit to Page" which scales incorrectly.

Glossary

This manual uses a lot of acronyms and technical terms. If you encounter an unfamiliar term, check here first.

AWG American Wire Gauge. Lower numbers = thicker wire. 10 AWG for power, 22 AWG for signals.

BLDC Brushless DC motor. Uses electronic commutation instead of brushes.

BVR Base Vectoring Rover. Muni's first rover morphology.

CAN Controller Area Network. Industrial communication bus used for motor control.

CAN_H / CAN_L CAN High and CAN Low. Differential pair signals.

CCW / CW Counter-clockwise / Clockwise. Motor rotation direction.

DC-DC DC-DC converter. Steps voltage down (48V → 12V).

DT Connector Deutsch DT connector. Weatherproof automotive connector.

E-Stop Emergency Stop. Cuts power to motors immediately.

FOC Field-Oriented Control. Advanced motor control algorithm for smooth, efficient operation.

GPIO General Purpose Input/Output. Digital pins on the Jetson.

Hub Motor Motor integrated into the wheel hub. No external gears or chains.

Jetson NVIDIA Jetson. Embedded AI computer. BVR0 uses Orin NX.

LiDAR Light Detection and Ranging. Laser-based 3D sensor.

LiPo Lithium Polymer battery. High energy density, requires careful handling.

LTE Long-Term Evolution. Cellular data connection.

Nm Newton-meter. Unit of torque.

RTK Real-Time Kinematic. GPS correction for centimeter accuracy.

Skid-Steer Steering method where left and right sides drive at different speeds.

T-Nut Threaded nut that slides into aluminum extrusion T-slots.

Teleop Teleoperation. Remote control of the rover by a human operator.

Termination 120Ω resistor at CAN bus endpoints to prevent signal reflection.

VESC Vedder Electronic Speed Controller. Open-source motor controller.

XT Connector Amass XT series. Yellow power connectors (XT90, XT60, XT30).

Index

Page numbers reference the first significant mention of each topic. See the Table of Contents for section-level navigation.

Municipal Robotics
Cleveland, Ohio
muni.works