Journal of open hardware

# OpenScout: Open Source Hardware Mobile Robot

HARDWARE METAPAPER

ubiquity press

SAMUEL J. CARTER
NIKOLAOS C. TSAGKOPOULOS
GARRY CLAWSON
CHARLES FOX

*Author affiliations can be found in the back matter of this article

## ABSTRACT

OpenScout is a low-cost open source hardware and software mobile robot that can be used for both indoor and outdoor tasks, transporting up to 15 kg of payload. It is designed to be easily and cheaply (350 USD) buildable and modifiable by non-specialists, and to function as a new standard physical platform for robotics research and real-world tasks, replacing current proprietary options. It uses four-wheel differential drive steering, and a hinged body which enables the wheels to drive over small obstacles without the need for suspension. Example applications include last mile and factory floor delivery, site survey and site monitoring.

CORRESPONDING AUTHORS:
**Samuel J. Carter**
University of Lincoln, GB
25735727@students.lincoln.ac.uk

KEYWORDS:
Autonomous vehicle; automation; mobile; robot; open source platform

TO CITE THIS ARTICLE:
Carter, SJ, Tsagkopoulos, NC, Clawson, G and Fox, C. 2023. OpenScout: Open Source Hardware Mobile Robot. *Journal of Open Hardware*, 7(1): 9, pp. 1–11. DOI: https://doi.org/10.5334/joh.54

## METADATA OVERVIEW

Main design files: https://github.com/cbedio/OpenScout.

Target group: researchers and hobbyists interested in autonomous vehicle research and robotics.

Skills required: 3D printing – easy; Mechanical assembly – easy; Electrical assembly – easy; Software – easy.

Replication: OpenScout has been successfully rebuilt from its build instructions only, by an MSc robotics student unfamiliar with the original design work.

## (1) OVERVIEW

### INTRODUCTION

The robotics community needs to easily reproduce whole research systems. This would allow multiple research groups to build large systems by extending, modifying, and integrating existing components. The community has made great progress using open source software, via the ROS ecosystem [8], which enables software components to be shared and to grow in this way. But it lacks standard open hardware platforms on which to run this software.

A few manufacturers have invested in creating open source software models of their proprietary hardware robots which currently function as *de facto* standards. However many researchers, especially in developing countries, cannot afford these commercial products. Like all proprietary products, they pose a lock-in risk to systems building upon them that the companies or products may vanish or increase their prices at any time. They may also be both technically and legally difficult to modify for new research needs.

Much of mobile robotics research and real-world deployment requires vehicles around 0.5 m in length and carrying around 15 kg of load, which have become *de facto* standards in many cases. 15 kg is roughly what a human can comfortably carry in a backpack or in their arms [11]. For research, such vehicles are small enough to cause minimal damage if colliding with a person or obstacle, while being able to carry batteries, sensors, and computers needed by research algorithms. For deployment, they can carry packages such as last-mile parcel and supermarket deliveries or restaurant food orders. Examples of proprietary vehicles in this class include the Clearpath Husky and Jackal, Pioneer-3 DX, Summit XL, AgileX, Bulldog, Leo Rover, and Husarion robots.

*Open Source Hardware (OSH)* is a recent movement [6] modelled on previous developments in Open Source Software [18] to enable cumulative collaboration in hardware designs. Presenting a first research design as OSH enables it to be extended gradually by the community and develop into a robust and deployable solution. OSH does *not* simply mean publishing CAD files of a design, rather it has a legal definition which requires the complete design to be easily available, buildable, and modifiable by anyone from commodity components. In addition to CAD files, this means provision of clear and well-tested step-by-step build instructions in the style of Lego or Ikea. OSH designs may only make use of components which are easily available on the open market. The emerging convention [6] in OSH is that reviewers check not just a paper but also a repository containing the design, build instructions, and licence, for OSH compliance.

### RELATED SYSTEMS

Several small, RC-scale ('toy'), cars have been completed and built as OSH including F1Tenth [1], AutoRally [9], BARC [10], MIT Racecar [2], MuSHR [3], [12], [17], and [16]. However these are unable to carry 15 kg payloads.

Larger, human-carrying OSH vehicles have also been designed and built, including the PixBot [15] and Tabby EVO [13] cars and the OSE LifeTrac tractor [14]. These are very large projects requiring thousands or tens of thousands of dollars of components and weeks or months of build time.

Designing and building a 0.5 m scale robot is a common exercise for robotics students and hobbyists – many thousands may be created each year around the world – but strangely there does not yet appear to be any design which has been fully and legally open sourced in sufficient detail to act as a standard platform for other researchers to continue to use. This may be due to the considerable additional effort needed to create reproducable build instructions, or to the recency of OSH as a legal concept.

OpenScout is intended to fill this gap (Figure 1). A complete, permanent, OSH design that is buildable by anyone with basic mechatronics skills for under 350 USD, and includes detailed step-by-step build instructions, a legal CERN-OSH-W licence, and OSHWA certification.

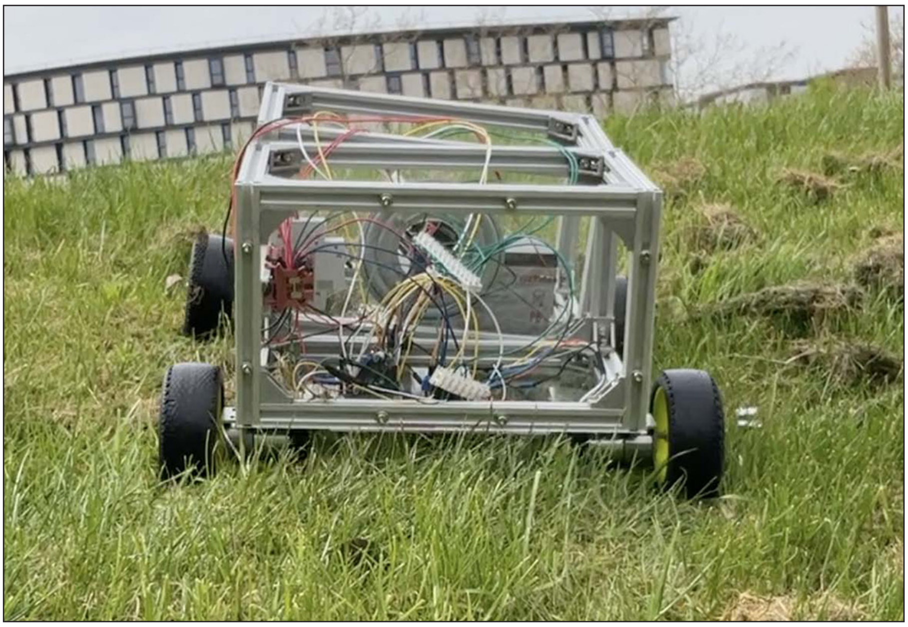


**Figure 1** Photo of the fully assembled OpenScout platform traversing a grassy incline demonstrating the hinged body mechanism.

## (2) OVERALL IMPLEMENTATION AND DESIGN

### SPECIFICATION

To fill the above use cases, the following specification is used for the design and is summarised in Table 1.

#### Application environment

The robot can operate indoor and outdoors on smooth floors, pavements, and uneven off-road terrains. This means it is capable of traversing 38 mm ground variation, as found in urban and grassy areas.

| | SPECIFICATIONS |
|---|---|
| Dimension | Max: L515 W450 H275 mm |
| Cost | 350 USD |
| Environment | Indoor, outdoor, uneven terrain |
| Ground Clearance | >38 mm |
| Weight | Max 10 kg |
| Payload | Max 15 kg |
| Mobility | Max speed: 500 mm/s |
| Gradient | Max: 1 : 2 |
| Turning Circle | <762 mm |
| Battery | 3 hours running time, 180 minutes recharge |
| Extensibility | Modular |
| Interface | wireless, RF controller |

**Table 1** Formal Specifications.

### Cost

The total components cost is below 350 USD, which is a typical small project budget accessible by researchers, educators, and hobbyists, including in developing countries. Where costs need to be further reduced the platform is able to incorporate a variety of alternative components.

### Maximum payload

The platform is able to carry a load of 15 kg and come to a safe stop in less than 1 second. A load of 15 kg is the average load a human is safely able to pick or place down at mid to lower leg height without assistance.

### Mobility

The robot is able to turn within a 765 mm turning circle to allow control in tight spaces such as indoor spaces, factory environments and site survey areas. The minimum speed of the platform is no less than 0.5 m/s and the maximum speed is no faster than 1.5 m/s. This is to allow for a comfortable, safe, walking speed alongside the robot. The robot is able to climb a gradient of up to 1 : 2 gradient unaided with no load. This enables it to access most human-accessible areas as well as formal access ramps.

### Extensibility

The mechanical design leaves space for additional equipment. The chassis design is modular and created from parts which are either easy to manufacture from raw materials or easily accessible from online suppliers.

### Assembly

The design is buildable with basic mechatronics skills (e.g. a STEM undergraduate student), from open market components and basic tools (e.g. soldering, screwing; no metalwork) in one day.

### Control interface

The user is able to control the platform without a tethered link to improve safety of operation as well as enable operation at a distance.

## MECHANICAL MODIFICATION FOR STEERING

To reduce cost and assembly difficulty, and increase maneuverability, a differential drive design is used. (This comes at the expense of driving accuracy, which would be higher for an Ackermann steered design.)

As the robot is required to travel over uneven terrain, a passive revolute joint is included which has its rotational axis in parallel with the robots centerline. This ensures that the wheels always makes four points of contact with the surface it is travelling over, making differential drive possible on these terrains, without the need for more complex suspension.

To reduce weight for safety, ease of lifting and placing the robot, and easy mounting of payloads and extensibility, the body is based on a aluminium profile frame no taller than mid lower leg height. Max speed is set to 0.5 m/s which is fast enough for the use cases but slow enough to minimise collision damage.

To enable simple manual control, an RF interface is included. This can be used by both by a remote manual controller or by onboard automated control systems.

## STRUCTURE

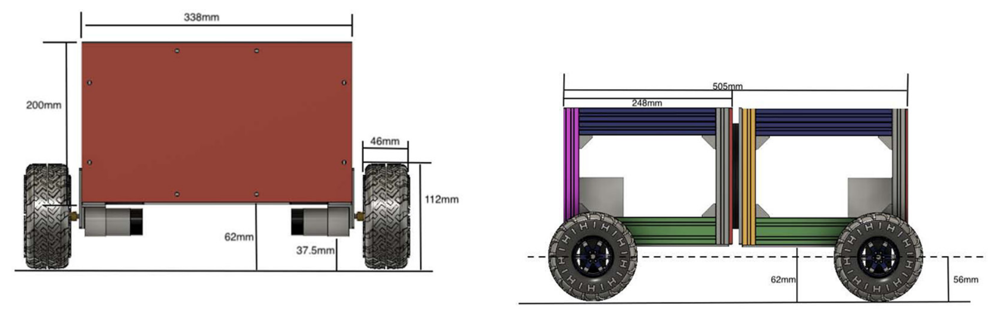
The robot structure profile is shown in Figure 2. The robot has a rectangular footprint, as is expected by and performs best with most robotic path planners.[1] This is particularly important for navigating through narrow gaps or doors.

The mechanical chassis consists of aluminium extrusions from the non-patented, open hardware, V-Slot profile system. This design makes it easy to vary dimensions, to mount

1 http://wiki.ros.org/dwa_local_planner.

*Journal of Open Hardware*
DOI: 10.5334/joh.54

payloads, and to modify the design. Acrylic panels are fitted to the aluminium chassis sides. The bottom panel is used to mount sensors, motor drivers, power supplies and other electrical components. The top panel is left empty, available for any additional sensors or components.




(a) Rear robot view. (b) Side robot view.

**Figure 2** Rear and side CAD views.

Dimensions of the aluminium extrusions are 20 × 20 mm profile (a standard size used in most desktop CNC machines) cut in lengths of 200 mm and 300 mm. This thickness is suitable to take the loads that the robot is expecting. The lengths are chosen to be simple multiples of 100 mm to enable reuse of cut profiles. The frame is coupled with M5 bolts connected to T-nuts through 90 degree brackets, in three places. To prevent bolts and screws from loosening upon vibrations while driving, sealant can be applied to their threads when attaching the nuts.

The robot has four wheels, one at each corner of the base footprint, each with radius 56 mm and offset from the base frame by 10 mm to avoid possible contact when skid steering when carrying a payload.

To aid navigational planning software, the centre of rotation is in the robot's $z$ axis positioned at the centre of the robot. This is done by aligning the cross section of the construction lines between diagonally opposite wheels with the centre of the robot in CAD software.

The robot is fitted with a lazy Susan turntable between the front half and the back half of the robot. This creates a strong revolute joint on the robot, though which electronics and wires can pass. These joints work well under compression and tension, however if the inner ring and outer ring are twisted in opposite directions, friction in the ball bearings within the turntable will create a large amount of resistance in the joint. On the robot the lazy Susan is fitted between two acrylic panels. This creates the desired hinge effect (passive compliant revolute joint).

## MECHATRONICS

A full electrical schematic of the platform is shown in Figure 3. The main mechatronic components of OpenScout consist of motors, motor drivers, encoders, mirco-controller, power supply and FR control.

### Motors

The motors need to carry at least 25 kg on asphalt and grass, and be affordable and easy to build with. Thus DC brushed motors are used. (Rather than brushless hub motors, which would have better form factor and reliability but increase cost and build difficulty.)

The robot uses four 12V brushed motors (supplier: CQRobot model number: CQR37D12V64EN-I) with integrated encoders and 90:1 inline transmission. Each has torque $\tau$ = 3.13 $Nm$ and maximum frequency $f$ = 2 $Hz$. With the current specified wheels, the maximum ground speed is $u_{max}$ = f × l = 0.7 $m/s$.

The total force due to weight is $F_w$ = 25 $kg$ × 9.8 $m/s^2$ = 245 $N$. Friction $\mu$ between rubber on asphalt is 0.9. Therefore the traction needed to move it is $F_\tau = F_w \times \mu$ = 220 $N$. The minimum required torque at max load is $\tau = F_\tau \times R$ = 12.3 $Nm$. The ground traction from each motor is $F_\tau = \frac{\tau}{R} = 55.8N$, hence four motors have ground traction of $F_\tau$ = 223.5 $N$, just over the required traction of 220 $N$.

DOI: 10.5334/joh.54

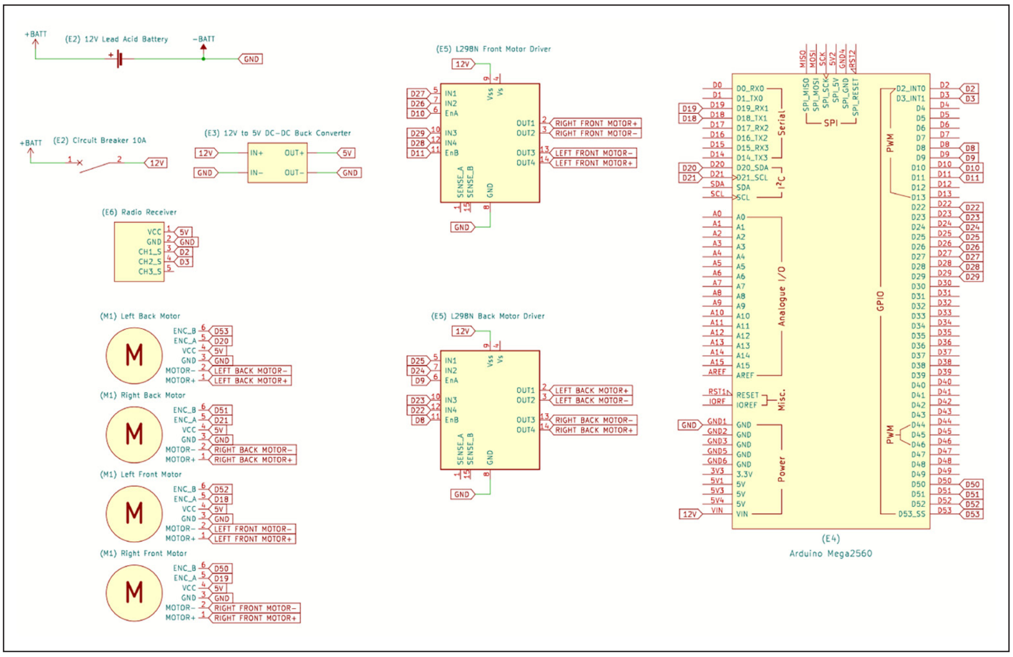



Figure 3 Electrical schematic.

## Motor drivers

The robot uses two L298N motor drivers for the front and back respectively. These each include two H-bridge circuits, which can drive two motors at the same time and set the direction of rotation.

## Encoders

Each brushed motor has two Hall effect encoders fitted to the motor shaft. The output voltage from an encoder works as an incremental encoder. Using the values of the incremental encoder, the orientation and velocity of the wheels can be determined.

## Microcontroller

The Arduino Mega 2560 is used as the platform's embedded microcontroller. This is due to its large (256 KB) on-board flash memory and external interrupt registers. The Arduino Mega can thus form the backbone for not only controlling all four motors through PID, but also interface with the RF receiver or other interfaces which could be added to it in the future.

## Power

A 12V 7.Ah leisure battery is connected – via a circuit breaker to protect the electrical components from current surges – to a 12V to 5V buck (step-down) converter, to maintain a regulated 5V voltage which is used for digital components such as the encoders on the backs of the motors and the onboard RF receiver. The design includes extra internal space for modification to larger, longer-lasting batteries if needed. Lead acid battery was preferred over Lithium ion for lower price and safety. It is charged using a standard Lead battery charger with alligator clips.

## RF control

An RF receiver (2.4GHz FS-A3) is included as the control interface, which is interfaced to the Arduino for processing.

## Software

Microcontroller source code is provided and can be compiled and transferred into the onboard Arduino Mega to provide basic teleoperation. It reads the RF receiver and the motor encoders, and sends target speeds to the motors.

## Interrupt pins

As the encoders and RF receiver output signals in unknown time, we used the external interrupts pins. That way, in each signal, the running program will halt at the location of the current instruction and will call the associated service routine with the interrupt. When the interrupt has ended, the program will continue at the location of the current instruction.

There are only 6 normal interrupts but 8 encoder signals making it impossible to connect both ENC_A and ENC_B for every motor to their own interrupt pin. To overcome this, only ENC_A is attached to interrupt pins in rising edge mode, while ENC_B is connected to normal digital input pins. In every ENC_A rising edge, the wheel has rotated $\Delta\theta = 360/N_t$, where $N_t$ is encoder's resolution. The direction of the robot is determined by the value of ENC_B.

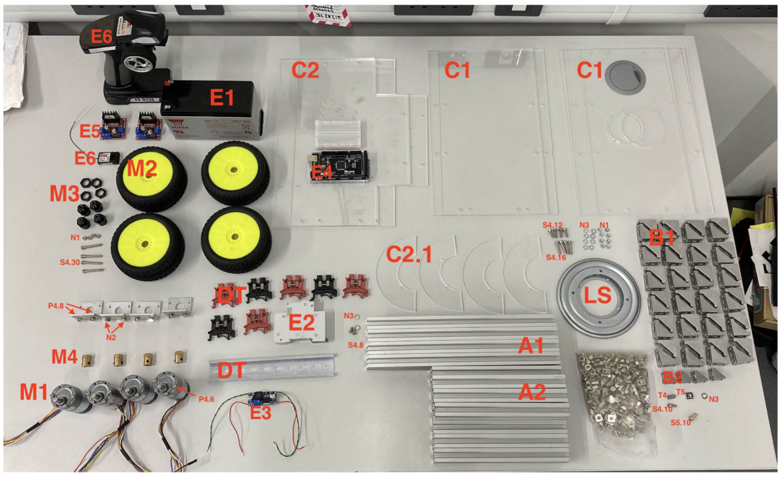
However, by connecting only ENC_A in interrupt pins, the encoder resolution is divided over 2, and as the interrupt routine is triggered only for rising edges, the overall encoder resolution is divided by 4. For a 90:1 geared motor, with 64CPR encoder resolution, the smallest measured angle is $\Delta\theta = \frac{360}{90\times64} = \frac{1}{5760} = 0.06°$ and for reduced resolution over 4, $\Delta\theta = 0.25°$ which is still less than one degree.

### Timer Interrupts

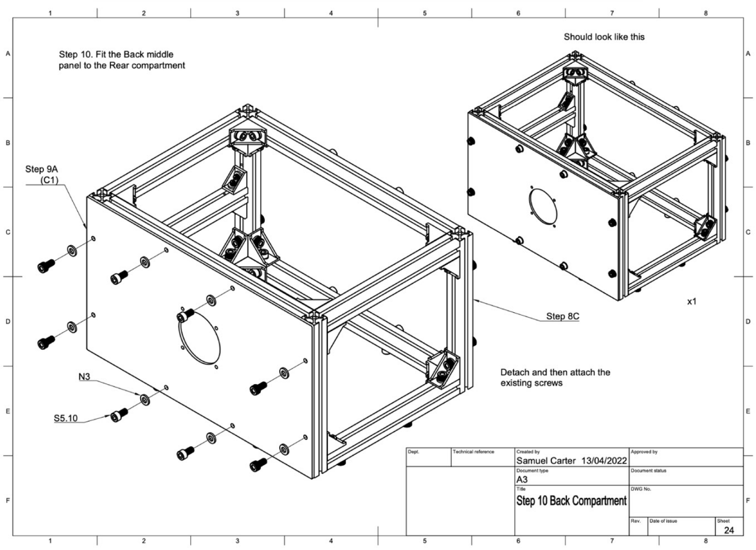
To save processing resources spent on computing the PID velocity control for each wheel, a timer interrupt is used to reduce the frequency velocity commands, calculated to 20Hz. On the Arduino Mega, timer 5 is usually reserved for servos. However as no servos are to be used with the base controller, this timer is altered to manage the control loop frequency.

## (3) QUALITY CONTROL

Detailed Lego/IKEA style step-by-step build instructions (Figure 4) are provided in the repository, including 38 visual build steps. CAD and software files showing a complete build are also




(a) Component knolling.




(b) Example of one of the 38 visual build steps from the build instructions.

**Figure 4** Building: components and step-by-step instructions.

provided, and can be modified to extend the design. The repository also contains instructions to calibrate the speed PID controller.

To validate the build instructions, a second build has been successfully reproduced by a student unfamiliar with the original design work.

## SAFETY

The total platform weight is no more than 10 kg. This is to ensure that the platform can be picked up from lower leg height in a safe manner. This also ensures that the platform is light enough to not cause significant damage to a human, wall, or office/lab obstacle on collision at maximum speed. The platform is also be robust enough so that no nuts, screws, joints or connections come loose from day to day operation between charging periods.

## GENERAL TESTING

Validation was performed through empirical testing of the OpenScout robot across each specification criterion over a variety of terrains to ensure suitability.

Maximum linear speed of the unloaded platform was found to be 0.5 m/s with maximum acceleration of 1 $m/s^2$. With an 8 kg load applied to the top surface of the robot the top linear speed and acceleration profile remained the same. With a load between 9 kg and 15 kg, the minimum speed was found to be 0.5 m/s and maximum acceleration found to be 0.5 $m/s^2$. A demonstration video: *OpenScout stopping and starting while carrying an 8 kg and 13 kg load respectively* is available in the repository.

OpenScout was found capable of moving across tarmac, gravel, short and medium height grass as well as undulated soiled ground with divots of 40 mm in height. The revolute mechanism shown in Figure 5 demonstrates an example of OpenScout's lazy Susan hinge enabling all four wheels to maintain ground contact while driving over an obstacle. A demonstration video: *OpenScout traversing uneven ground with 40 mm peaks and troughs* showing this is available in the repository.

Using its lazy Susan hinge, the platform can incrementally traverse objects up to a maximum height of 150 mm while still maintaining full ground contact across each of the 4 wheels. To achieve the 150 mm height, incremental steps were required at no greater than 50 mm each. A video showing this, *Demonstrating the Lazy Susan mechanism over obstacles of 50 mm in height* is provided in the repository.

The platform incorporates a skid steering system achieving a turning circle of 700 mm. The rotational speed achieved was measured to be 0.62 rads/sec giving a full rotation in 10 seconds. A demonstration video: *OpenScout skid steer turning within a 350 mm turning radious* is provided in the project repository.

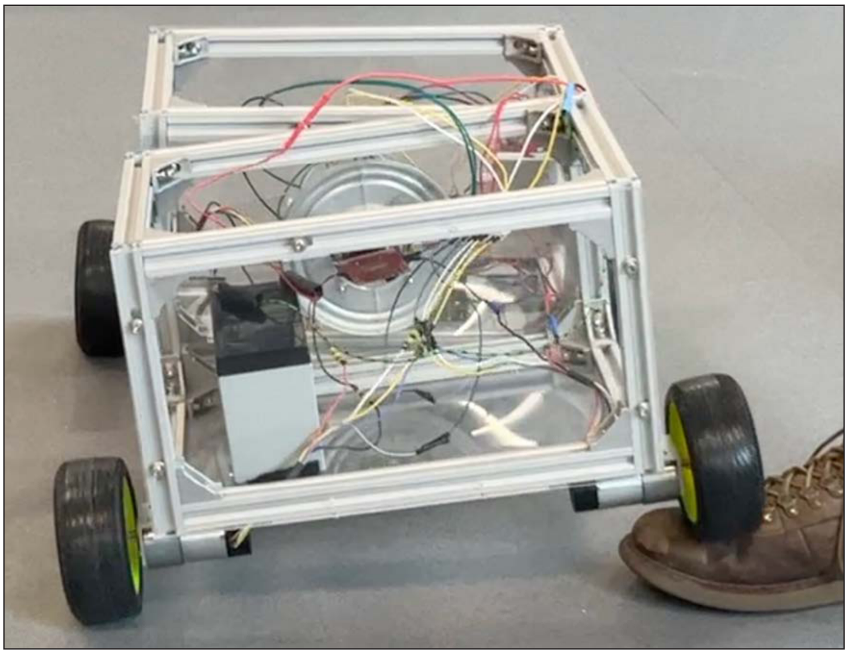


**Figure 5** OpenScout with revolute hinge driving over an obstacle.

OpenScout was found to be capable of climbing and descending embankments with short to medium grass height that had an uneven undulating surface with a gradient 1 : 2. A demonstration video *OpenScout climbing and descending grassy slope of 1 : 2 gradient* showing this is provided in the repository.

Typical platform run time is over 3 hours with constant use. Battery replacement takes 15 seconds, with typical battery recharge time taking 180 minutes.

During field validation the OpenScout was tested within pedestrian areas and grassed space and inspected on a periodic basis to ensure all bolts and connections were still taught. No components required refastening at any inspection.

Total platform assembly time was found to be 6.5 hours including all electronics and component assembly by a non-specialist performing a successful rebuild from the build instructions, with no additional contact with the designers.

## (5) LICENCES

### LICENCING

The CERN-OSH-W licence [7] is a recent (2020) legal definition of OSH and is used here. Designed as a hardware analogy of the GNU LGPL software licence, it ensures that any modifications made to the design are contributed back to the community, but also allows non-open products to use it unmodified as a sub-component. The licence ensures that all CAD, video, and build instruction files are in open formats, and that they remain open for ever. It ensures that all components are open market available. (It does not however require components to be OSH themselves, and is sometimes criticized for this. We use OSH components where possible, such as the V-Slot profiles, but it is currently hard or impossible to be OSH 'all the way down' to OSH microcontrollers and motors.)

### OSH CERTIFICATION

OpenScout has been certified[2] by the Open Source Hardware Association (OSHWA) which is a peer reviewed process for compliance across software, hardware and documentation. Compliance demonstrates that the project follows the community definition of open source hardware maintained by OSHWA. Hardware projects that display the certification logo are licensed and documented in a way that makes it easy for users to use and build upon them.

### HARDWARE DOCUMENTATION AND FILES LOCATION:

Name: GitHub

Project repository: https://github.com/cbedio/OpenScout

(Publication project archive DOI: 10.5281/zenodo.10263675.)

Licence: CERN-OHL-W for hardware design and build instructions; GPL for software source code. – The hardware is structured as two separate formal OSH designs, each licenced as CERN-OSH-W. The first covers all components which are easily transferable to other vehicles without modification. The second contains all components which are specific to the mobility scooter donor vehicle. This structure enables the first design to be used as sub-component of closed products while also preventing closed modifications of it.

Date published: 04/04/2022

## (6) DISCUSSION

### CONCLUSIONS

The validation shows that the spcifications are met so that OpenScout is an affordable, open hardware modular differential drive robotic platform which can carry a load up to 15 kg. Its passive revolute joint in the middle enables it to drive over small objects while keeping contact

2 https://certification.oshwa.org/uk000042.html.

with all four wheels, making it suitable for outdoor as well as indoor environments. OpenScout's load could be used to carry sensors and computers, and/or physical deliveries such as parcels or food. The design is trivially modifiable to different shapes and sizes simply by replacing the aluminium profiles with different lengths.

We invite interested members of the OSH community to contribute, extend, fork and build on the project, to deeper OSH, higher accuracies and additional applications. Specific future work could include: The motor drivers used are not OSH can could be replaced with OSH ones such as OSMC [5]. The microcontroller is not OSH and could be replaced by an open RISC-V. We are not aware of any OSH motors or RF receivers, though if developed they could be swapped in. Deep OSH wheels and tyres are in development [4] and could be swapped in. The RF interface could be replaced or complemented by a serial protocol (or CAN bus, Bluetooth, Infrared, or USB). A higher level ROS interface could be created to interface to the RF or serial interface. A physical Gazebo simulation implementing the same ROS interface could be created.

OpenScout is a simple mobile robot and similar in design to many hobbyist projects. However, this is the first 0.5 m scale mobile robot to be fully and correctly OSH licenced, which unlike those projects enables it to function and grow as a new and solid – both physically and legally – standard platform for mobile robotics research and deployment. Please join us and see if together we can do for mobile robot hardware what ROS has done for software.

## FUNDING INFORMATION

This work was partially supported by the Engineering and Physical Sciences Research Council [EP/S023917/1].

## COMPETING INTERESTS

The authors have no competing interests to declare.

## AUTHOR AFFILIATIONS

**Samuel J. Carter** orcid.org/0000-0002-8643-4816
School of Computer Science, University of Lincoln, UK

**Nikolaos C. Tsagkopoulos** orcid.org/0009-0006-0385-998X
School of Computer Science, University of Lincoln, UK

**Garry Clawson** orcid.org/0000-0003-0315-4147
School of Computer Science, University of Lincoln, UK; Department of Engineering, University of Cambridge, UK

**Charles Fox** orcid.org/0000-0002-6695-8081
School of Computer Science, University of Lincoln, UK; Institute for Transport Studies, University of Leeds, UK

## REFERENCES

1. **F1tenth.** https://f1tenth.org/.
2. **MIT racecar.** https://mit-racecar.github.io/.
3. **MuSHR.** https://mushr.io/.
4. **Openwheels.** hackaday.io/project/16024-openwheel-parametric-osh-wheelstyrestracks.
5. **OSMC motor driver.** http://www.robotpower.com/products/osmc_info.html.
6. **Jérémy Bonvoisin, Jenny Molloy, Martin Häuer,** and **Tobias Wenzel.** Standardisation of practices in open source hardware. *arXiv preprint arXiv: 2004.07143*, 2020. DOI: https://doi.org/10.5334/joh.22
7. **CERN.** CERN-OSH-W licence. https://cern-ohl.web.cern.ch/home.
8. **Steve Cousins, Brian Gerkey, Ken Conley,** and **Willow Garage.** Sharing software with ROS. *IEEE Robotics & Automation Magazine*, 17(2): 12–14, 2010. DOI: https://doi.org/10.1109/MRA.2010.936956
9. **Brian Goldfain, Paul Drews, Changxi You, Matthew Barulic, Orlin Velev, Panagiotis Tsiotras,** and **James M. Rehg.** Autorally: An open platform for aggressive autonomous driving. *IEEE Control Systems Magazine*, 39(1): 26–55, 2019. DOI: https://doi.org/10.1109/MCS.2018.2876958
10. **Jon Gonzales.** *Planning and Control of Drift Maneuvers with the Berkeley Autonomous Race Car*. PhD thesis, University of California at Berkeley, 2018.

11. **MF Haisman.** Determinants of load carrying ability. *Applied Ergonomics*, 19(2):111–121, 1988. DOI: https://doi.org/10.1016/0003-6870(88)90004-X
12. **Naohiro Nakamoto** and **Hiroyuki Kobayashi.** Development of an opensource educational and research platform for autonomous cars. In *IECON 2019 – 45th Annual Conference of the IEEE Industrial Electronics Society*, volume 1, pages 6871–6876, 2019. DOI: https://doi.org/10.1109/IECON.2019.8926794
13. **Open Motors.** Tabby EVO. https://www.openmotors.co/evplatform/.
14. **Open Source Ecology.** Lifetrac. https://wiki.opensourceecology.org/wiki/LifeTrac.
15. **PixMoving.** Pixbot. https://gitlab.com/pixmoving/pixbot.
16. **H. Rogers,** and **C. Fox.** An open source seeding agri-robot. In *Proc. 3rd UK-RAS Conference*, 2020. DOI: https://doi.org/10.31256/Or6Mf2T
17. **Bastien Vincke, Sergio Rodriguez Florez,** and **Pascal Aubert.** An open-source scale model platform for teaching autonomous vehicle technologies. *Sensors*, 21(11), 2021. https://github.com/BastienV-SATIE/AutonomousCar/. DOI: https://doi.org/10.3390/s21113850
18. **Sam Williams.** *Free as in Freedom: Richard Stallman's Crusade for Free Software*. O'Reilly Media, 2011.



**TO CITE THIS ARTICLE:**
Carter, SJ, Tsagkopoulos, NC, Clawson, G and Fox, C. 2023. OpenScout: Open Source Hardware Mobile Robot. *Journal of Open Hardware*, 7(1): 9, pp. 1–11. DOI: https://doi.org/10.5334/joh.54

**Submitted:** 03 November 2022
**Accepted:** 08 December 2023
**Published:** 27 December 2023