Autonomous Search and Rescue Robot

**MTE 380 – University of Waterloo | Winter 2025**  
*131g | 13.0s mission time | 1st place (68× score of 2nd place)*

# My Role

In this 5-person design challenge, I led the development of the fan and suction system, mechanical-electrical integration, and co-designed the PCB. I also took a systems integration and project management perspective to ensure the electrical, mechanical, and software subsystems worked together reliably under tight deadlines. Key contributions included:

* Designing and testing custom impellers and suction skirt geometries to increase theoretical cornering speed
* Casting custom high-μ silicone wheels for optimized grip
* Co-designing a 4-layer PCB chassis, including low-latency routing for encoders and signal lines
* Validating and tuning control loops and high-speed sensor-actuator timing
* Engineering a servo gripper system with foam dampening to reduce vibration error and a sloped LEGO backboard to prevent figure snagging
* Creating cutouts in the PCB for both the LEGO figure and suction fan to reduce weight and support subsystem integration

📸 **Insert: team photo or hero shot of final robot (add the videos, pics, and other bs here)**

# Performance Index (PI)

The competition used the following metric to determine success:

PI=CR⋅m⋅t2PI = \frac{C}{R \cdot m \cdot t^2}

Where:

* R = number of failed runs (0.75 for 3 successful runs, 0.9 for 2 successful runs, 1 for 1 successful run)
* m = robot mass (g)
* t = time to complete task (s)

This incentivized designs that were light, fast, and reliable. Our 131g robot completed the full challenge in 13.0 seconds, succeeded in all 3 runs, and scored 68× higher than the next closest team.

📈I**nsert: PI score comparison graph (including/excluding our team)**

Suction Downforce System (Theoretical)

To increase cornering performance beyond the friction-limited cap, I led the development of a suction fan system with custom impellers and skirts. While ultimately cut from the final run due to power instability, the system was fully functional and validated through structured testing.

## Why Downforce? Derivation from First Principles

We start with Newton’s Second Law:

F = ma

In circular motion, the required centripetal force is:

F\_c = \frac{mv^2}{r}

This must be provided by friction, which depends on normal force:

F\_f = \mu F\_N \quad \text{and} \quad F\_N = mg + F\_d

Setting them equal:

\mu (mg + F\_d) = \frac{mv^2}{r} \Rightarrow v = \sqrt{\frac{r \mu (mg + F\_d)}{m}}

Given:

* μ=1.2\mu = 1.2
* r = 0.15 \text{ m}
* g = 9.81 \text{ m/s}^2
* m = 0.131 \text{ kg}
* F\_d = 0.148 \text{ N} (measured from incline testing)

v\_{\text{max, actual}} = \sqrt{ \frac{0.15 \cdot 1.2 \cdot (0.131 \cdot 9.81 + 0.148)}{0.131} } \approx \boxed{1.40\, \text{m/s}}

| **Downforce Level** | **Normal Force (N)** | **Max Speed (m/s)** |
| --- | --- | --- |
| None (gravity only) | 1.285 | 1.33 |
| +131g (1x mass) | 2.570 | 1.88 |
| +262g (2x mass) | 3.855 | 2.30 |
| +393g (3x mass) | 5.140 | 2.67 |
| **Measured: 0.148 N** | 1.433 | **1.40** |

📸 **Insert: derivation diagram or equations screenshot**

## Impeller Design and Parameter Testing

I fabricated and tested **over 70 impeller designs**, varying:

* Blade count (2–14)
* Blade pitch (15-35mm)
* Blade curvature (45°–60°)
* Diameter (22–30 mm)
* Height (5–15 mm)
* PCB cutout hole diameter (10 – 20mm)
* Revolved cut dimensions
* Style: radial vs. turbo axial

Each impeller was tested on a 3D printed robot-shaped test chassis over five timed runs down a 25° 1’ ramp using a fixed 5V, 0.5A power supply on the motor.

Below compares the main takeaways, mainly a 5 bladed radial impeller design is much better than any compression style. They both shared the similarity that performance was clearly peaked at 5 blades and fell off extremely once there was an increase to 7 or a decrease to 2. The most surprising thing was 2 blades had similar results to 7 blades and that 3 blades was better than 6 blades, but 5 was a clear winner with 4 not trailing too far behind.

| **Impeller Type** | **Avg. Ramp Time (s)** | **Relative Traction** |
| --- | --- | --- |
| No Fan | 0.84 | 1× |
| Compression (5 blade) | 21.13 | 25× |
| Final Radial (5-blade) | 55.12 | **65×** |

## Skirt Design and Testing

The skirt system needed to create a seal against the course while being low friction and not catching on the lips between the tiles on the course. I iterated a few designs listed below and shown:

Skirt systems tested:

* Flexible foam
* PLA skirts
* Mylar sealing tape
* PLA side rails + sealing tape front/back

**SKIRT PICS**

The final version used:

* 3D printed side rails for lateral sealing
* Tape on the front and back for airflow containment
* Angled PLA ramp to prevent snagging on floor transitions

The final version also was able to move up and down while still maintaining a good enough seal to create some sort of suctioning effect thanks to the side rails.

**FINAL SKIRT PIC**

Paired with the best impeller, this configuration yielded the best traction test results and met my requirements while staying very light since I printed with almost no infill and the actual skirt was printed at 0.5mm layer height.

## Why It Was Cut

We originally specked a more efficient motor, but a vendor mix-up forced us to use a higher-power fan motor, which disrupted our original power budget calculations significantly. While our 220 mAh 25C drone LiPos (2× 6 g) should have supplied 5.5 A, we saw instability in the Pixy2.1, gripper, and motor systems.

Suspecting false discharge ratings or power ratings for certain components and lacking time to verify, we dropped the suction system 3 days before competition to ensure final system reliability for game day so we could focus on completing the mission instead of optimizing.

The suction system added ~10g. While traction improvements were clear, we didn’t complete full-course testing with the system, so performance impact on full complete runs remains unknown therefore it is unclear whether it would have even been worth the extra weight but comparing the PI scores on average between runs with the suction system on and off would have clearly shown which is better.

# Electrical System: PCB & Power

I co-designed a 4-layer PCB that acted as the electrical core and structural chassis.

The overall design is summarized by the following block diagram:  
**BLOCK DIAGRAM PCB**

**Features:**

* Teensy 4.0 MCU (SPI to Pixy2.1)
* DRV8871 motor drivers
* LM22670 buck converter (5V @ 5A)
* Simple FET drivers for fan and gripper motors
* Separate layers for encoder and communication traces
* Cutouts for LEGO and suction impeller

**Power:**

* 2× 3.7V 220 mAh LiPo cells (6g each)
* 80% peak current draw estimate: ~2.58 A
* Runtime need: ~43 mAh (1-minute full mission which has safety factor for peak conditions)

📸 **Insert: PCB top/bottom image or board schematic photo**

There was lots of debugging for our initial PCB, mainly there was a servo issue which ended up being a ground pin that was floating and causing unpredictable behaviour. We figured this out through systematically tracing each pin starting from power and known working pins and working from the parts that worked to the issue. We then figured out our problem after systematic troubleshooting and grounded that pin to a nearby ground pad. There were a few other issues like backwards headers and IC’s that needed a new spec due to the fan motor changing, so for PCB 2 these updates were made and there were zero issues.

# Software Architecture & PD Tuning

We used a deterministic state machine:  
Each robot behavior (line-follow, object detect, drop-off, return) was handled by structured conditions and transitions.  
**STATE MACHINE PIC**

We used PD control with the vector output from the pixy2.1 as our target and the center of the frame as our current heading then corrected the difference between them. We opted for PD and ignored I because we believed the steady state error was negligible and would only slow down our response compared to a PD controller only. The following notes about our PD control should be noted:

**PD Control:**

* Started with Ziegler-Nichols tuning method
* Tuned further through live tests
* Ambient lighting affected Pixy2.1 performance
* Built indoor vs outdoor controller configs for Pixy 2.1

# Motor Selection

We used **Pololu 6V Micro Metal Gearmotors (10:1)**:

* No-load: 3100 RPM
* Torque: 0.02157 Nm
* Paired with robot mass and wheel sizing, this gave us a projected top speed of ~**3.0 m/s** well-matched to our goal time.

Looking back, we could only utilize PWM values above 60/255 to move our robot and anything below it did not move it. We also only utilized up to 105/255 for our 13s run so it is clear we did not come near our peak performance capabilities, but the suction system could have helped push it further.

# Final Competition Performance

| **Metric** | **Value** |
| --- | --- |
| Weight | 131g |
| Mission Time | 13.0 seconds |
| Reliability | 3/3 successful runs |
| Score vs. 2nd Place | 68× higher |

A key optimization was that we were the only team to stop at the closest green box — which is when the clock stopped. Other teams returned home. This small rule-based insight saved critical time.

Also, considering our potential velocity increase would have been from 1 m/s (7.5s to the bullseye at approximately 7.5m long of a course) to 1.4 m/s would mean that our score could have been way faster. Also, if we could improve the pickup system that would be ideal. We found a loophole that allowed us to configure the LEGO figure with a paperclip and use an electromagnet for acquisition, but it was denied by the teaching staff unfortunately. Also, due to previous years, they also banned system that launched the LEGO figure, so physically grabbing it and moving it was the only option.

🎥 **Insert: final competition run video or animation**

# Reflections

This was one of the most demanding and satisfying projects I’ve worked on.

Key Takeaways:

* Shared team goals matter, we all built toward the same vision and committed equally
* Testing > calculations on short timelines
* Cutting scope is smart when system reliability matters and timelines are short
* Prior art matters, the micromouse projects and competition accelerated our overall design as well as other academic research papers on line following robots
* Simple systems are more tunable, debuggable, and scalable

If you want to read the full technical report, please download here…it still feels like we could have written double what we did in the report, but we were limited to our page count, which we already exceeded with this submission. So please, if you have any questions reach out and I would love to explain our thinking or answer questions! If you are doing a similar project and you came across this goodluck! Reach out and tell me about it, I would love to hear how you are approaching it and what kind of optimizations you made!

**Insert: full report PDF link**  
**Insert: final run video demo link**