CARSS - Controlled Autonomous Robot by Saideep and Sankalp

PURDUE UNIVERSITY FORT WAYNE

Presenter: Saideep K, Sankalp S
Advisor: Dr. Chao Chen

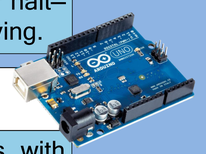
Abstract

We have entered the Automated Era–where everything from robots to file cleanup runs on automation. “Zanda Claus”, the first-generation autonomous car from ECE 30300, is a prime example. Using an embedded system to dodge obstacles within 20 cm, it’s a simplified take on what modern EVs represent. Now, in ECE 49600, it's evolving–the “ARCKS-Bot” will include camera vision, voice control, and advanced navigation. These upgrades push the limits of what automated cars can really do!

In-Progress Journey: 303 to 496

The "Zanda Claus" Object Avoiding Car embodied this philosophy by integrating robotics, embedded systems, and real-time control mechanisms to create an autonomous vehicle capable of navigating its surroundings without human intervention. Originally inspired by modern cars' ability to detect obstacles and apply emergency brakes, Zanda Claus took automation a step further as an honors-option course project in ECE30300 during fall 2024. Instead of just stopping, it actively scans for obstacles, checks the left side for a clear path, and intelligently reroutes itself to maintain motion.

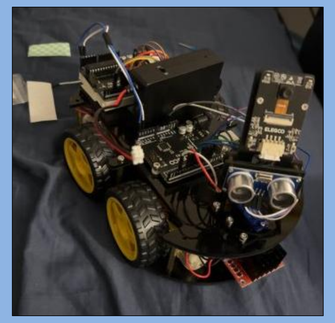
This project is now continuing as an independent study honors project under ECE 49600 in spring 2025. The next-generation model “CARSS” under development is set to achieve even more. Equipped with AI-powered voice control, it will recognize commands like go, stop, and turn left through a trained microphone. Additionally, with color recognition, it will detect specific colors that dictate whether it should proceed or halt–bringing advanced automation closer to real-world autonomous driving.



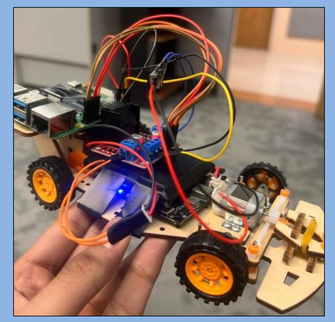
Project Goals

Automated and Controlled - Implement self-driving capabilities with precise motor and obstacle avoidance.
Color Recognition - Utilize computer vision to detect and respond to different colors for navigation and decision-making.
Voice Control - Enable hands-free operation through voice commands for remote and local control.

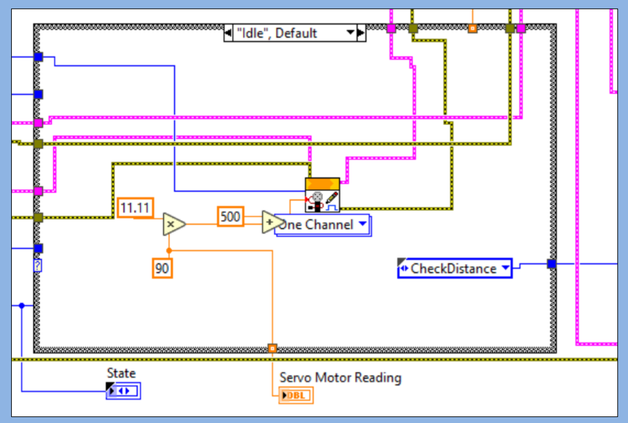
Comparison



Zanda Claus



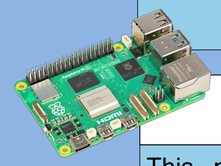
Prototype Car



Servo Motor Case Structure on LabVIEW

```python
def set_servo_angle(angle):
    duty = (angle / 18) + 2  # Convert angle to duty cycle
    GPIO.output(SERVO_PIN, True)
    servo.ChangeDutyCycle(duty)
    time.sleep(0.5)
    GPIO.output(SERVO_PIN, False)
    servo.ChangeDutyCycle(0)
```

Servo Motor Function on Python



Challenges and Changes

- **Power constraint**: Zanda Claus initially ran on a single Arduino board, but an unexpected issue arose–it couldn’t provide enough power for all connected sensors. To overcome this, an additional Arduino board had to be integrated.
- **Software limitation**: We initially used LabVIEW to run the control algorithm. However, the code had to be to run on a computer instead of directly uploaded to the Arduino board. This means that the car had to remain constantly connected to the source code, limiting its independence and ability to move around.
- **System reliability**: Several loose wires compromised the stability of the connections. This led to intermittent sensor failures and unreliable performance, requiring extra effort to secure and manage the wiring effectively.

We have made the following changes to address the challenges:

Hardware Changes -
• **Arduino UNO → Raspberry Pi 5**:
Enhanced processing, communication, and flexibility for advanced autonomous car projects.
• **Standard Chassis → 3D-Printed Chassis**:
Improved reliability, easier modifications, and better expandability.

Software Changes -
•**LabVIEW (GUI) → Python (CLI)**:
Improves automation, scripting, and performance by removing graphical overhead and enabling direct command execution.

Acknowledgement

This project is supported by the Honors Research and Creative Endeavors Scholarships in spring 2025. We would also like to thank the ECE Department for providing with the equipment and support necessary to start this project.