



PERMUTES

GRABHI

The Advanced 4x4 Robotic Car with Grab-and-Go Capability

Robotic Car That Grabs and Delivers, Redefining Convenience and Innovation.

PRESENTED BY

Avni Jaiswal, Nimish Gupta, Tanishq Som

Table of Contents



- 03** About Grabhi Bot
- 04** System Architecture
- 05** Software Components
- 06** WorkFlow
- 07** Production Budget
- 08** Future Work
- 09** Further Upgrades & Potential Applications
- 10** Conclusion



About Grabhi Bot

The Grabhi robot represents a groundbreaking innovation in the realm of robotics, featuring a 4x4 robotic car with intricate mechanisms designed for precise object manipulation and movement. This report details the components, design, assembly, and operational features of Grabhi.

Our STEM team worked hard to achieve certain perks of Grabhi. The seamless radio-based Communication between the controller and bot reduces the latency and make the controls swift and dynamic. Rich software allows our bot to enable Halt/Semi-Automatic mode to Autonomously control itself and pick stuff up for a short period of time. Keeping the future aspects and further modifications in mind, Grabhi is capable of broadcasting various Sensors' data over Wi-Fi which will improve overall maneuverability of the bot. It will also expand the wide field of applications for Grabhi.

Features	
Payload Potential	~ 150 gm
Speed	~ 0.7 m/s
Battery Life	~ 2 Hrs
Assembly Time	~ 35 min
Gripper Length	~ 13 cm

Contributors and Visionaries



Tanishq Som
Electricals & Software



Nimish
Advisor



Avni Jaiswal
Media & Outreach



Lakshya Vats
System Architecture



Vatsal Sikarwar
Design & Mechanics



Shrikant Vidhya
Faculty Mentor

System Architecture



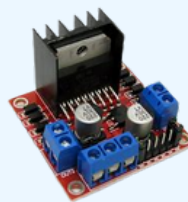
Hardware Components:
The hardware components used in Grabhi includes :



Raspberry PI



Servo Motor

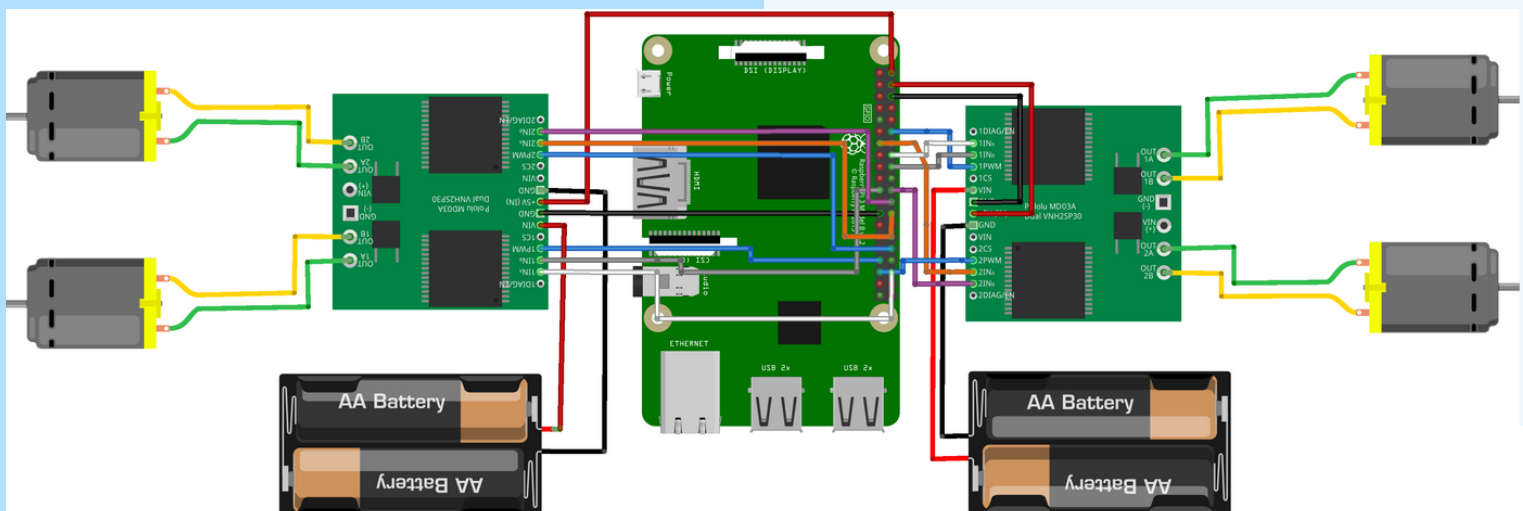


L298N



Micro-Controller

Schematics



fritzing

Software Components

Programming Languages Used:

- The ATMEGA328P microcontroller code is written in the Arduino programming language, while the Raspberry Pi code is written in Python, leveraging Pygame for game controller interaction and serial communication with the ATMEGA328P.
- Libraries and Frameworks:
 - The ATMEGA328P code utilizes the Arduino Servo library. The Python code employs Pygame for game controller interaction and Serial library for communication with the ATMEGA328P microcontroller.

ATMEGA328P IC Code (Arduino):

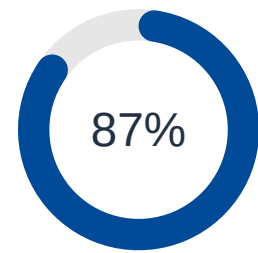
- The code designed for the ATMEGA328P microcontroller is written in the Arduino programming language. This code manages the servo motors connected to the ATMEGA328P IC. It sets up the serial communication interface, receives instructions from the Raspberry Pi, and controls the movement of five servo motors based on the commands received. The code structure includes initialization, servo attachments to specific pins, and a loop to continuously read instructions via serial communication, adjusting the servo angles accordingly. This code is integral to Grabhi's arm mechanism control.

Python Code for Raspberry Pi:

- The Python script operates on the Raspberry Pi and is primarily responsible for controlling Grabhi's overall movements, utilizing the Pygame library to interface with the wireless game controller and managing serial communication with the ATMEGA328P microcontroller. It initializes the GPIO pins, sets up PWM (Pulse Width Modulation), and continuously monitors the wireless game controller's inputs. Based on the joystick inputs, this script controls Grabhi's movements (forward, backward, left, and right) along with adjusting servo angles and controlling arm mechanisms based on specific button and joystick inputs.

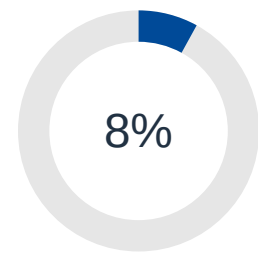
Both code segments are integral to Grabhi's functionality. The ATMEGA328P code specifically manages the arm mechanism, whereas the Python code for the Raspberry Pi handles the overall movements and interactions with the wireless game controller. These codes combined enable Grabhi's comprehensive operation, allowing precise control over its movements and arm manipulations, making it a versatile and functional robotic system.

PROJECT REPORT



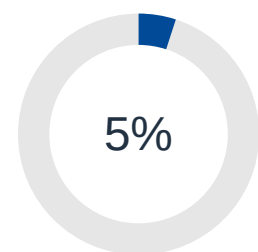
PYTHON

Script to read signals and control the Bot.



C++

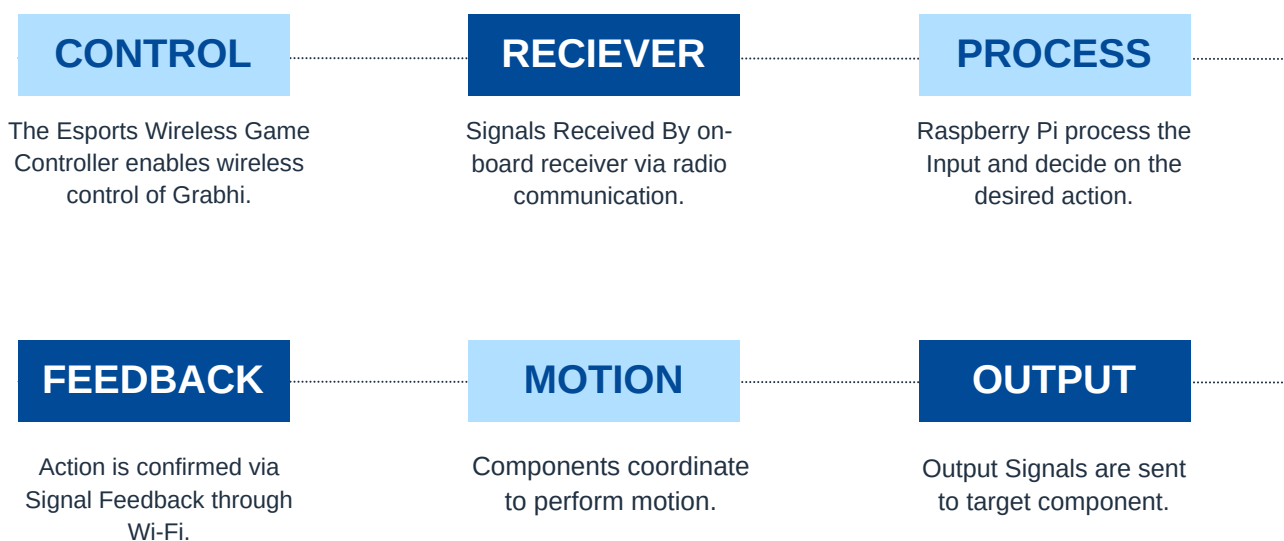
Serial Communication b/w Raspberry and IC for Arm.



LINUX

To Execute Python and Cpp file in Raspbian Linux OS

WorkFlow



Operating Instructions

- **Powering Up the Robot:**
 - To power Grabhi, ensure the necessary power cells are connected to the motor drivers, and the Raspberry Pi is functional. The wireless game controller activates the robot for maneuvering.
- **Communication Protocols:**
 - Grabhi operates through serial communication, enabling commands from the Python script on the Raspberry Pi to control the ATMEGA328P microcontroller and its connected servo motors.
- **Control Interface:**
 - The Esports Wireless Game Controller enables wireless control of Grabhi, allowing users to steer, manipulate the arm, and grab objects.

Results and Performance

- **Metrics:**
 - Grabhi demonstrates exceptional maneuverability, precise object manipulation, and reliable wireless control, showcasing its versatility in various applications.

Challenges Faced

- Rigorous testing made it difficult to maintain integrity of Bot.
- Power Overloads.
- Latency in Arm Actions.
- Very few internet resource and community support available.

Improvements Made

- Used Metal to improve Durability.
- Improved Power Source.
- Separate Microcontroller for Arm.

Production Budget

FOR GRABHI

Component	Cost (INR)	Quantity	₹ /-
Raspberry Pi 3B+	₹ 6,500	x1	₹ 6,500
Raspberry Pi 3	₹ 4,300	x1	₹ 4,300
32 GB Memory Card (Amazon)	₹ 400	x1	₹ 400
128 GB Memory card (Samsung)	₹ 1,100	x1	₹ 1,100
L298N Motor Driver	₹ 200	x2	₹ 400
USB To Serial Converter	₹ 140	x1	₹ 140
ATMega 328P	₹ 410	x1	₹ 410
16 MHz Crystal Oscillator	₹ 10	x5	₹ 50
22pf, 0.1uf Capacitor	₹ 250	x1	₹ 250
MG996 Servo Motor	₹ 380	x4	₹ 1,520
9G Servo Motor	₹ 140	x6	₹ 840
BO Motor	₹ 90	x6	₹ 540
Mecanum Wheel	₹ 205	x4	₹ 820
Esports Wireless controller	₹ 1,300	x1	₹ 1,300
18650 Lithium ion Battery	₹ 80	x8	₹ 640
Miscellaneous	₹ 800	--	₹ 800
TOTAL	--	43x	₹ 20,010

Future Work

IT'S THE WAY YOU
HOLD...

Further improvements like integrating different sensors to sense the environment and improve autonomous motion of Grabhi can be made.

These includes sensor like Camera to enable Computer Vision, Ultrasonic to avoid obstacles and navigate autonomously. And Algorithms like Image segmentation, Classification, Detection etc...



Further Upgrades

COMPONENT	COST (INR)	QUANTITY	₹ /~
<u>Raspberry Pi Camera Module</u>	₹ 1,200	x1	₹ 1,200
<u>Ultrasonic Sensors</u>	₹ 90	x6	₹ 540
<u>GPS Module</u>	₹ 2,500	x1	₹ 2,500
Chassis	₹ 400	x6	₹ 2,400

Potential Applications

WAREHOUSE MANAGEMENT

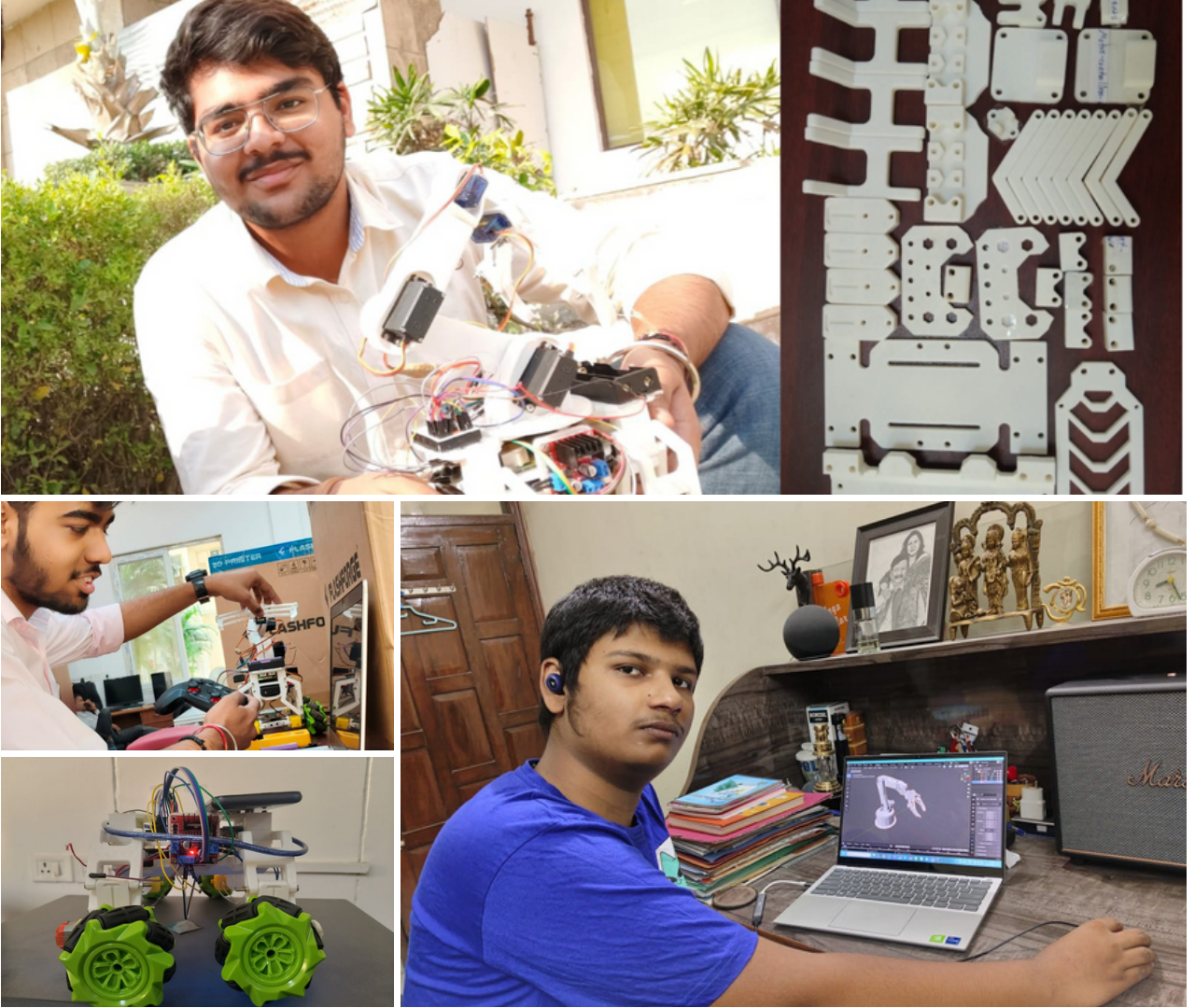
Grabhi offers potential applications in optimizing inventory management, aiding in picking and placing tasks, and streamlining warehouse operations.

MATERIAL HANDLING

It demonstrates adaptability in transporting and organizing materials, facilitating efficient logistics and handling tasks across various industries.

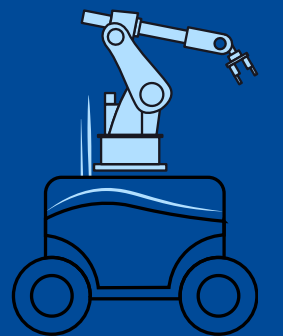
EDUCATIONAL SETTINGS

Its multifunctional design serves as an educational tool, fostering learning in robotics, automation, and control systems for students and enthusiasts.



रोबोटिक्से उत्कृष्टता च सृष्टिर्नवीनत्वं
च यत्र स्वप्नानां सत्यत्वं समर्थं
करोति।

"Excellence and innovation in robotics are the
messages that transform our dreams into reality."





PERMUTES

By amalgamating cutting-edge hardware, sophisticated software, and versatile functionalities, Grabhi stands as an emblem of innovation, promising transformative applications across industries and educational domains, pushing the boundaries of robotics and automation.



For Updates, Code files &
3D files... Check out



www.permutes.github.io/Grabhi

