

DEPARTMENT: Software Engineering

COURSE: SEN 4821 - Robotics Project

LEVEL: 400 Level

SEN 481 ROBOTICS PROJECT DOCUMENTATION

AUTONOMOUS WASTE COLLECTION ANT ROBOT

A Smart Hexapod Walking Robot with AI Vision

PROJECT TYPE:

Hexapod Walking Robot with AI Vision

APPLICATION:

Autonomous Waste Identification, Collection & Disposal

2025/2026 ACADEMIC SESSION

October 2025

**PROJECT TEAM MEMBERS**

1. Izuafa Abdulrafiu Braimah - VUG/SEN/22/7708

2. Oke Emmanuel Olamide- VUG/SEN/22/7035

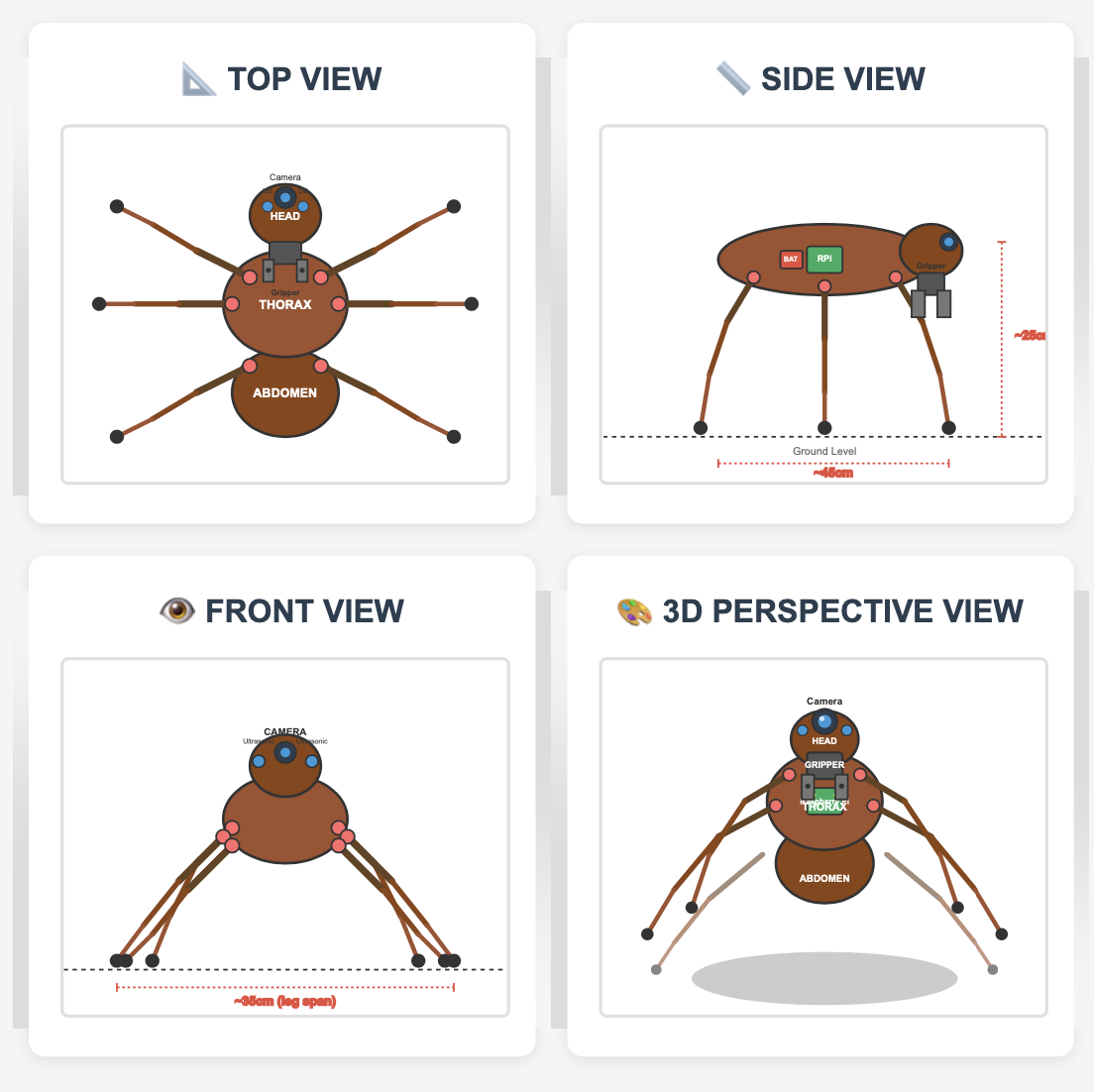
3. Onoja Hillary Leonard- VUG/SEN/22/7403

4. Ikenna Iheanaetu - VUG/SEN/22/7788

5. Ibrahim Rahmat - VUG/SEN/22/8245

## 

**Robot Visual Design**



## **TABLE OF CONTENTS**

1. Project Overview
2. System Architecture
3. Hardware Components Specification
4. Electronics & Wiring
5. Mechanical Design
6. AI/ML Vision System
7. Assembly Instructions
8. Testing & Calibration
9. Budget & Cost Analysis
10. Expected Outcomes
11. Safety Considerations
12. References & Datasheets

## 

## **1. PROJECT OVERVIEW**

### **1.1 Project Description**

This project involves designing and building a hexapod (6-legged) ant-inspired robot capable of:

* **Autonomous navigation** using leg-based locomotion (no wheels)
* **Waste detection** using computer vision and AI
* **Object manipulation** through a gripper mechanism
* **Waste classification** (recyclable vs. non-recyclable)
* **Autonomous disposal** at designated collection points

### **1.2 Biomimetic Inspiration**

The design mimics ant morphology and behavior:

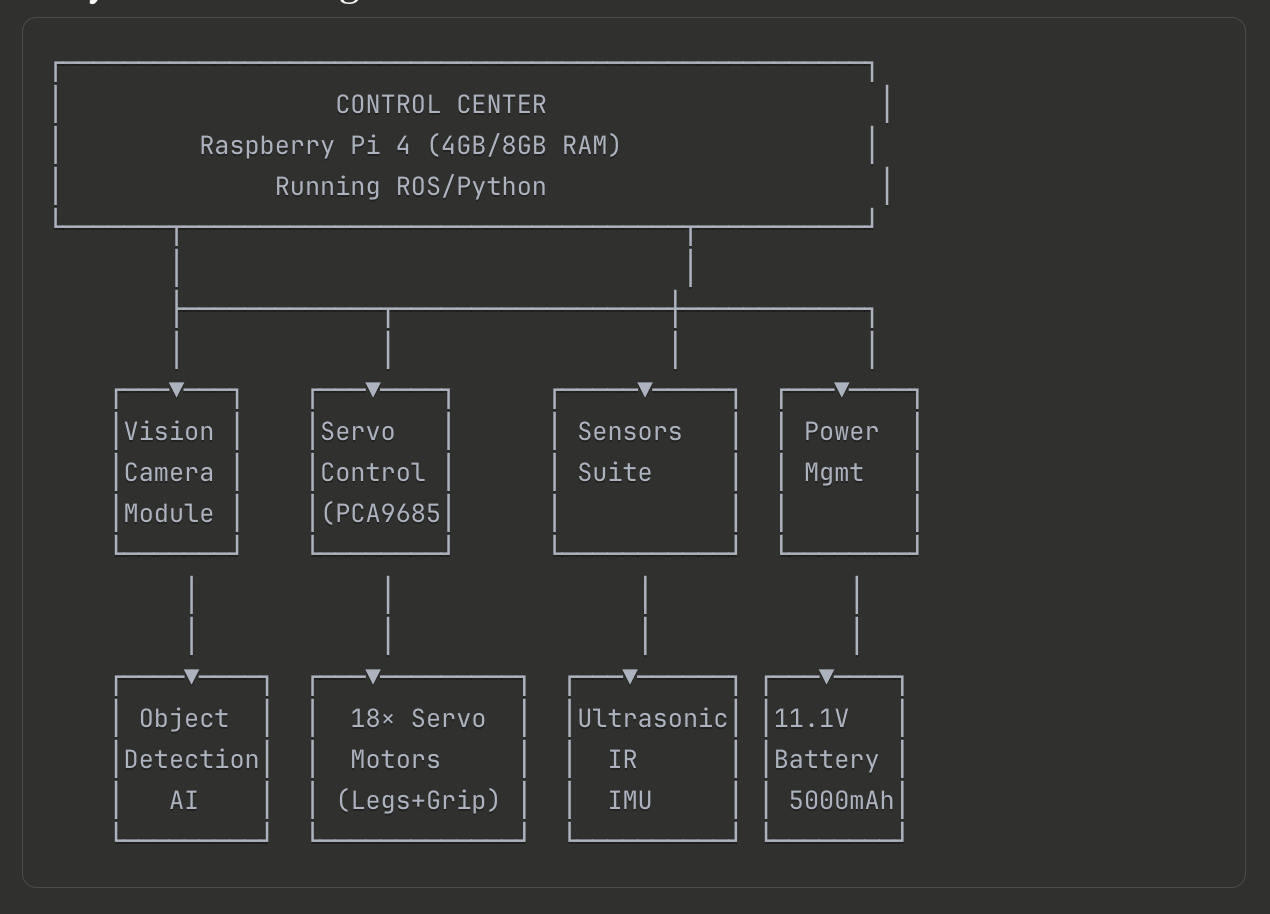
* **Six-leg configuration** for stable terrain traversal
* **Segmented body** (head, thorax, abdomen)
* **Mandible-inspired gripper** for object manipulation
* **Antenna sensors** for obstacle detection

### **1.3 Key Specifications**

* **Payload Capacity:** 0.5 - 1.0 kg
* **Walking Speed:** 0.1 - 0.3 m/s
* **Operating Time:** 45-60 minutes per charge
* **Detection Range:** 0.5 - 3 meters
* **Classification Accuracy:** >85% (target)
* **Dimensions:** 45cm (L) × 35cm (W) × 25cm (H)
* **Weight:** 2.5 - 3.5 kg
* **Degrees of Freedom:** 18 DOF (3 per leg × 6 legs)

## **2. SYSTEM ARCHITECTURE**

### **2.1 System Block Diagram**



### **2.2 Subsystem Overview**

#### **2.2.1 Locomotion System**

* 18 servo motors (3 per leg × 6 legs)
* Tripod gait controller for stable walking
* Inverse kinematics for smooth leg movements

#### **2.2.2 Vision & AI System**

* Camera module for image acquisition
* CNN-based waste classification using YOLOv5
* Real-time object detection and distance estimation

#### **2.2.3 Manipulation System**

* 2-DOF gripper mechanism (base rotation + jaw open/close)
* Adjustable gripping force
* Object size adaptation (50mm - 150mm opening range)

#### **2.2.4 Sensor Suite**

* Obstacle detection using ultrasonic sensors
* Ground proximity sensors (IR)
* Body orientation tracking (IMU)

## **3. HARDWARE COMPONENTS SPECIFICATION**

### **3.1 MAIN PROCESSING UNIT**

#### **Raspberry Pi 4 Model B**

* **Model:** 4GB or 8GB RAM variant
* **Processor:** Broadcom BCM2711, Quad-core Cortex-A72 @ 1.5GHz
* **Storage:** 64GB MicroSD Card (Class 10, UHS-I)
* **Operating System:** Raspberry Pi OS (64-bit)
* **Purpose:** Main controller, AI inference, sensor fusion
* **Quantity:** 1
* **Datasheet:** [raspberrypi.com/documentation](https://www.raspberrypi.com/documentation/)

**Alternative:** NVIDIA Jetson Nano 4GB (better GPU for AI processing)

### **3.2 SERVO MOTORS (LOCOMOTION)**

#### **MG996R Metal Gear Servo**

* **Torque:** 11 kg·cm @ 4.8V, 13 kg·cm @ 6V
* **Speed:** 0.17 sec/60° @ 4.8V
* **Operating Voltage:** 4.8V - 7.2V
* **Weight:** 55g
* **Gear Type:** Metal gears (high durability)
* **Control:** PWM (50Hz, 1000-2000μs pulse width)
* **Rotation:** 180° (120° usable range)
* **Current Draw:** 500mA typical, 2.5A stall
* **Quantity:** 18 (3 per leg × 6 legs)
* **Purpose:**
  + Coxa joint (hip): Body-leg connection, lateral movement
  + Femur joint (knee): Up-down leg motion
  + Tibia joint (ankle): Foot positioning

### **3.3 SERVO MOTORS (GRIPPER & CAMERA)**

#### **MG90S Micro Servo**

* **Torque:** 2.2 kg·cm @ 4.8V
* **Speed:** 0.10 sec/60° @ 4.8V
* **Weight:** 13.4g
* **Quantity:** 2-4 units
* **Purpose:**
  + Gripper base rotation (1 unit)
  + Gripper open/close mechanism (1 unit)
  + Camera pan/tilt servos (2 units)

### **3.4 SERVO CONTROLLER**

#### **PCA9685 16-Channel 12-bit PWM Servo Driver**

* **Channels:** 16 per board (expandable to 62 via I2C addressing)
* **Resolution:** 12-bit (4096 steps)
* **Communication:** I2C (address: 0x40-0x7F)
* **Supply Voltage:** 2.3V-5.5V (logic), 5V-10V (servo power)
* **PWM Frequency:** 40Hz - 1000Hz (typically 50Hz)
* **Current Capability:** 25mA per channel
* **Quantity:** 2 boards
  + Board 1 (0x40): Leg servos
  + Board 2 (0x41): Gripper and camera servos
* **Datasheet:** [NXP PCA9685](https://www.nxp.com/docs/en/data-sheet/PCA9685.pdf)

**Connections:**

* V+ (Servo Power): 6V regulated supply
* VCC (Logic): 5V from Raspberry Pi
* GND: Common ground
* SDA/SCL: I2C communication (GPIO2/GPIO3)

### **3.5 CAMERA MODULE**

#### **Raspberry Pi Camera Module V2**

* **Sensor:** Sony IMX219 8-megapixel
* **Resolution:** 3280 × 2464 pixels (still), 1920×1080@30fps (video)
* **Field of View:** 62.2° horizontal × 48.8° vertical
* **Connection:** CSI 15-pin ribbon cable
* **Focus:** Fixed (optimized for 50cm - infinity)
* **Quantity:** 1
* **Mount:** Pan-tilt mechanism using 2 servos
* **Datasheet:** [Raspberry Pi Camera Docs](https://www.raspberrypi.com/documentation/accessories/camera.html)

### **3.6 ULTRASONIC SENSORS**

#### **HC-SR04 Ultrasonic Distance Sensor**

* **Range:** 2cm - 400cm
* **Accuracy:** ±3mm
* **Operating Voltage:** 5V DC
* **Current Draw:** 15mA
* **Frequency:** 40kHz ultrasonic
* **Trigger Input:** 10μs TTL pulse
* **Echo Output:** 5V pulse (requires voltage divider for Pi)
* **Quantity:** 3-4 sensors
  + Front: 2 sensors (left/right obstacle detection)
  + Sides: 2 sensors (lateral awareness)
* **Purpose:** Obstacle detection, cliff detection

**Important:** Echo pin outputs 5V - must use voltage divider (1kΩ + 2kΩ resistors) to protect Raspberry Pi 3.3V GPIO pins.

### **3.7 INFRARED SENSORS**

#### **IR Proximity Sensor (E18-D80NK)**

* **Detection Range:** 3cm - 80cm (adjustable via potentiometer)
* **Operating Voltage:** 5V DC
* **Output:** Digital (high/low)
* **Current Draw:** <25mA
* **Quantity:** 2-4 units
* **Purpose:** Ground detection (prevent falls), close-range obstacles
* **Mounting:** Downward-facing at leg/body junction

### **3.8 INERTIAL MEASUREMENT UNIT (IMU)**

#### **MPU6050 6-Axis Gyroscope & Accelerometer**

* **DOF:** 6 (3-axis gyro + 3-axis accelerometer)
* **Communication:** I2C (address: 0x68 or 0x69)
* **Gyroscope Range:** ±250, ±500, ±1000, ±2000 °/s
* **Accelerometer Range:** ±2g, ±4g, ±8g, ±16g
* **Operating Voltage:** 3.3V - 5V
* **Quantity:** 1
* **Purpose:** Body orientation, tilt detection, balance control
* **Datasheet:** [InvenSense MPU6050](https://invensense.tdk.com/products/motion-tracking/6-axis/mpu-6050/)

### **3.9 POWER SYSTEM**

#### **LiPo Battery 3S**

* **Type:** Lithium Polymer (LiPo)
* **Configuration:** 3S1P (3 cells in series)
* **Voltage:** 11.1V nominal (12.6V full, 9.0V cutoff)
* **Capacity:** 5000mAh - 6000mAh
* **Discharge Rate:** 30C continuous
* **Connector:** XT60 or Deans T-plug
* **Weight:** ~400-500g
* **Quantity:** 1
* **Runtime:** 45-60 minutes estimated

**Safety Equipment Required:**

* LiPo fireproof storage bag
* Battery voltage alarm (<3.3V per cell)
* Balance charger (IMAX B6 or equivalent)

#### **Voltage Regulators**

**5V Buck Converter (for Raspberry Pi)**

* **Model:** LM2596 DC-DC Buck Converter
* **Input:** 11.1V LiPo
* **Output:** 5V @ 3A (adjust to 5.1V under load)
* **Efficiency:** ~85-90%
* **Quantity:** 1

**6V Buck Converter (for Servos)**

* **Model:** D24V50F6 or equivalent
* **Input:** 11.1V LiPo
* **Output:** 6V @ 5A continuous (10A peak recommended)
* **Quantity:** 1
* **Purpose:** High-current servo power rail

**3.3V Regulator (for Sensors)**

* **Model:** AMS1117-3.3 or LD1117V33
* **Input:** 5V
* **Output:** 3.3V @ 1A
* **Quantity:** 1

#### **Power Protection**

* **XT60 Connectors:** 60A continuous rating
* **10A Blade Fuse:** Main battery protection
* **5A Fuse:** Servo rail protection
* **2A Fuse:** Logic rail protection
* **Emergency Stop Switch:** Manual cutoff
* **4700µF Capacitor:** Servo power rail filtering (reduces voltage spikes)

### **3.10 MECHANICAL COMPONENTS**

#### **Structural Materials**

**Robot Chassis & Frame**

* **Material Options:**
  + 3mm-5mm acrylic sheets (laser-cut, budget-friendly)
  + Aluminum plates (stronger, more expensive)
  + 3D printed PLA/PETG (custom shapes)
  + Carbon fiber sheets (lightweight, expensive)
* **Base Plate:** 250mm × 200mm
* **Body Segments:** Custom cut to design
* **Quantity:** 5-8 plates

**Leg Segments**

* **Material:** Aluminum rods (6mm diameter) or 3D printed
* **Dimensions per leg:**
  + Coxa (hip segment): 60mm
  + Femur (thigh segment): 80mm
  + Tibia (shin segment): 100mm
* **Quantity:** 6 complete leg sets (18 segments total)

**Foot Pads**

* **Material:** Rubber or silicone
* **Diameter:** 15-20mm
* **Purpose:** Traction and shock absorption
* **Quantity:** 6

#### **Fasteners & Hardware**

* **M3×6mm screws:** 50 pieces (servo mounting)
* **M3×10mm screws:** 30 pieces (brackets to body)
* **M3×16mm screws:** 20 pieces (through-body connections)
* **M3 nuts:** 80 pieces
* **M3 lock washers:** 40 pieces
* **M2.5×6mm screws:** 10 pieces (Raspberry Pi mounting)
* **M3 standoffs:** 20-30 pieces (15mm, 20mm lengths)

#### **Servo Brackets**

* **Type:** Aluminum L-brackets or 3D printed
* **Quantity:** 18-24 pieces
* **Purpose:** Mount servos at correct angles for leg joints

### **3.11 GRIPPER MECHANISM**

#### **Design Specifications**

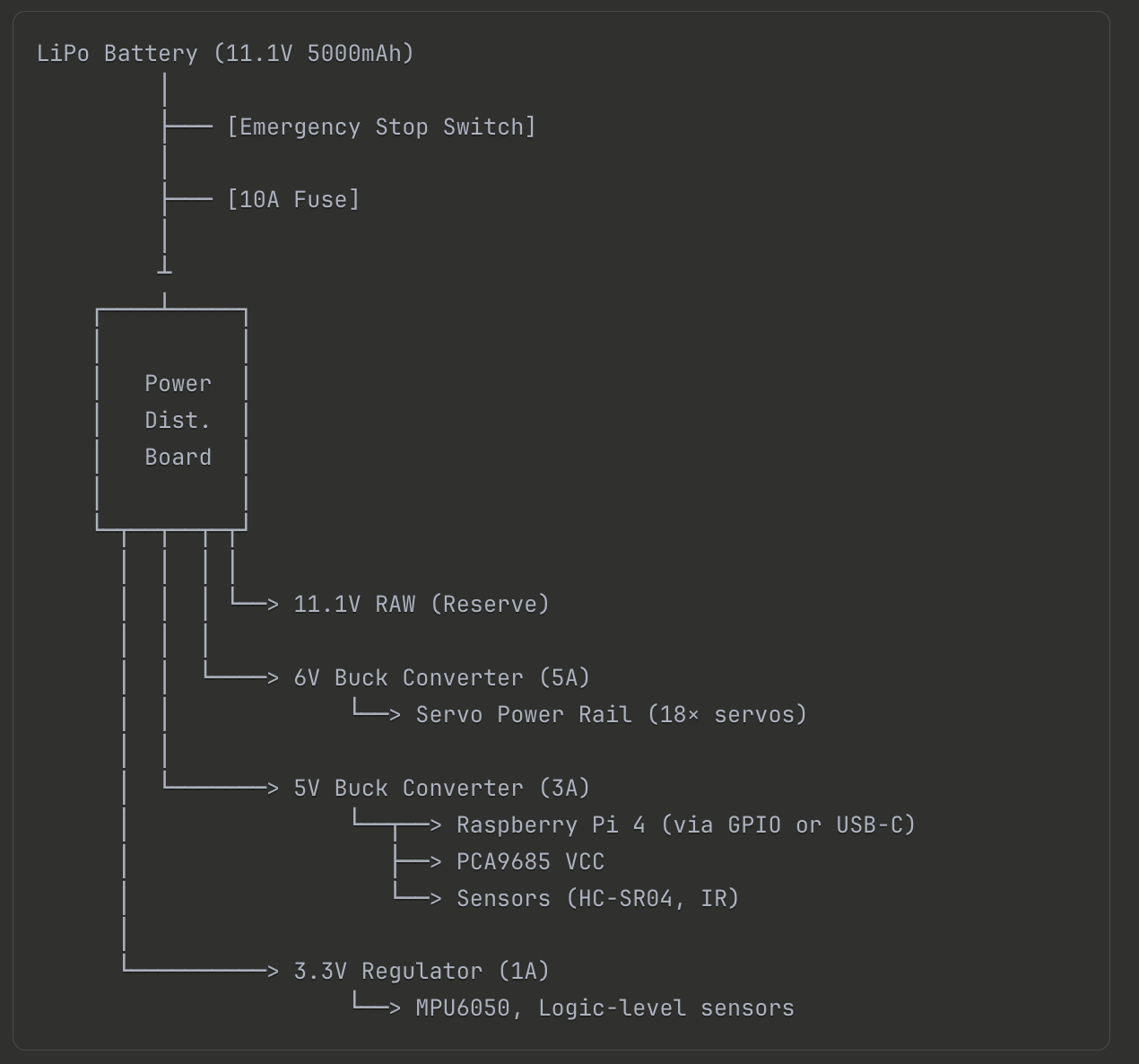
* **Type:** Parallel jaw gripper
* **Material:** 3D printed PLA/ABS or aluminum
* **Opening Width:** 50mm (closed) - 150mm (open)
* **Gripping Force:** 1-3 kg
* **Actuation:** MG90S servo with mechanical linkage
* **Force Sensor (optional):** FSR 0.2kg-20kg (2 units, one per jaw)

### **3.12 ADDITIONAL ELECTRONICS**

* **5V Cooling Fan:** 40mm×40mm×10mm (Raspberry Pi cooling)
* **LED Indicators:** 5mm LEDs - Red, Green, Blue (3-5 units)
* **220Ω Resistors:** Current limiting for LEDs
* **Emergency Stop Button:** Red mushroom-style, Normally Closed
* **Active Buzzer:** 5V piezo (audio feedback)
* **Wires:** 22-26 AWG stranded (signal), 16 AWG (power)
* **Cable Ties:** Various sizes for wire management
* **Heat Shrink Tubing:** Wire protection and insulation

## **4. ELECTRONICS & WIRING**

### **4.1 POWER DISTRIBUTION DIAGRAM**



### **4.2 RASPBERRY PI GPIO PIN ASSIGNMENT**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **GPIO Pin** | **Function** | **Connected To** | **Voltage** | **Notes** |
| GPIO2 (SDA) | I2C Data | PCA9685, MPU6050 | 3.3V | Pull-up resistors |
| GPIO3 (SCL) | I2C Clock | PCA9685, MPU6050 | 3.3V | Pull-up resistors |
| GPIO4 | Output | Buzzer | 3.3V | PWM capable |
| GPIO17 | Output | LED Green | 3.3V | Status indicator |
| GPIO18 | Output | LED Red | 3.3V | Error indicator |
| GPIO22 | Output | Ultrasonic Trig (Front) | 3.3V |  |
| GPIO23 | Input | Ultrasonic Echo (Front) | 3.3V | Via voltage divider |
| GPIO24 | Output | Ultrasonic Trig (Side L) | 3.3V |  |
| GPIO25 | Input | Ultrasonic Echo (Side L) | 3.3V | Via voltage divider |
| GPIO5 | Output | Ultrasonic Trig (Side R) | 3.3V |  |
| GPIO6 | Input | Ultrasonic Echo (Side R) | 3.3V | Via voltage divider |
| GPIO12 | Input | IR Sensor 1 | 3.3V | Ground detection |
| GPIO13 | Input | IR Sensor 2 | 3.3V | Ground detection |
| GPIO16 | Input | Emergency Stop | 3.3V | Internal pull-up |
| 5V Power | Output | Sensor VCC | 5V | Max 1A shared |
| 3.3V Power | Output | Sensor VCC | 3.3V | Max 50mA |
| GND | Ground | Common ground | 0V | Multiple connections |

**I2C Addresses:**

* PCA9685 Board 1 (Legs): 0x40
* PCA9685 Board 2 (Gripper): 0x41
* MPU6050 IMU: 0x68

### **4.3 PCA9685 SERVO CHANNEL ASSIGNMENT**

#### **Board 1 - Legs (Address 0x40)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Channel** | **Leg** | **Joint** | **Servo** |
| 0 | Right Front | Coxa | MG996R |
| 1 | Right Front | Femur | MG996R |
| 2 | Right Front | Tibia | MG996R |
| 3 | Right Middle | Coxa | MG996R |
| 4 | Right Middle | Femur | MG996R |
| 5 | Right Middle | Tibia | MG996R |
| 6 | Right Rear | Coxa | MG996R |
| 7 | Right Rear | Femur | MG996R |
| 8 | Right Rear | Tibia | MG996R |
| 9 | Left Front | Coxa | MG996R |
| 10 | Left Front | Femur | MG996R |
| 11 | Left Front | Tibia | MG996R |
| 12 | Left Middle | Coxa | MG996R |
| 13 | Left Middle | Femur | MG996R |
| 14 | Left Middle | Tibia | MG996R |
| 15 | Left Rear | Coxa | MG996R |

#### **Board 2 - Gripper & Extras (Address 0x41)**

|  |  |  |
| --- | --- | --- |
| **Channel** | **Function** | **Servo** |
| 0 | Left Rear Femur | MG996R |
| 1 | Left Rear Tibia | MG996R |
| 2 | Gripper Base Rotation | MG90S |
| 3 | Gripper Jaw Open/Close | MG90S |
| 4 | Camera Pan | SG90 |
| 5 | Camera Tilt | SG90 |
| 6-15 | Reserved | - |

### **4.4 WIRING SPECIFICATIONS**

#### **Wire Gauge Selection**

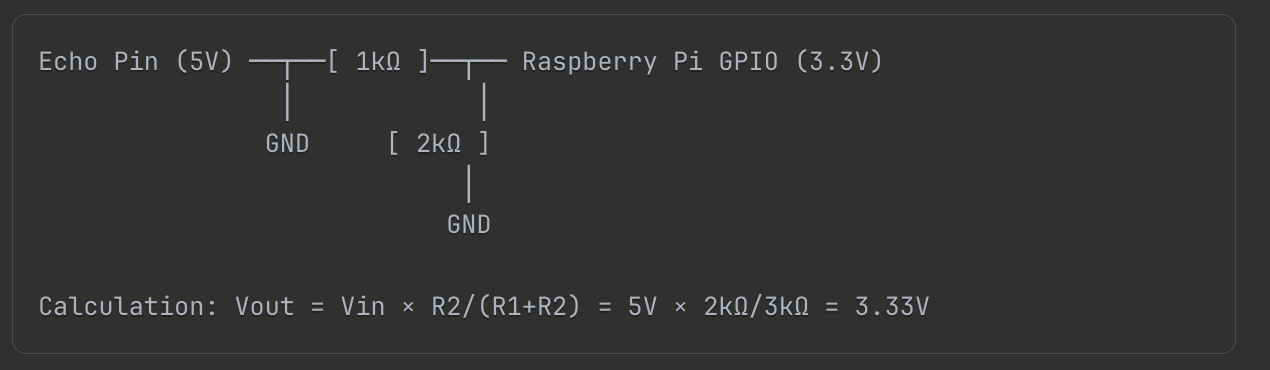
|  |  |  |  |
| --- | --- | --- | --- |
| **Purpose** | **AWG** | **Current** | **Color** |
| Battery to Main | 14 | 15A | Red/Black |
| Power Distribution | 16 | 10A | Red/Black |
| Servo Power Rails | 18 | 5A | Red/Black |
| Logic Power | 22 | 1A | Red/Black |
| GPIO Signals | 24 | 0.5A | Various |
| Sensor Lines | 26 | 0.1A | Various |
| I2C Communication | 26 | - | Yellow/Orange |

#### **Wire Color Standards**

* **Red:** Positive voltage (V+)
* **Black:** Ground (GND)
* **Yellow/Orange:** I2C communication
* **Green:** GPIO outputs
* **Blue:** GPIO inputs

### **4.5 VOLTAGE DIVIDER CIRCUIT (HC-SR04 Echo Pin)**

**Circuit Design:**



**Calculation:** Vout = 5V × 2kΩ/(1kΩ+2kΩ) = 3.33V (safe for Pi)

**Required Components:**

* 1kΩ resistor (3 units, one per ultrasonic sensor)
* 2kΩ resistor (3 units, one per ultrasonic sensor)

### **4.6 CRITICAL WIRING NOTES**

1. **Common Ground:** ALL components must share common ground
2. **Servo Power Capacitor:** 4700µF across 6V rail (reduces spikes)
3. **I2C Pull-ups:** 4.7kΩ resistors on SDA/SCL (if not on boards)
4. **Fuse Protection:** Install fuses before voltage regulators
5. **Star Grounding:** Single ground point to reduce noise

## **5. MECHANICAL DESIGN**

### **5.1 LEG KINEMATICS**

#### **Leg Configuration**

Coxa (60mm)

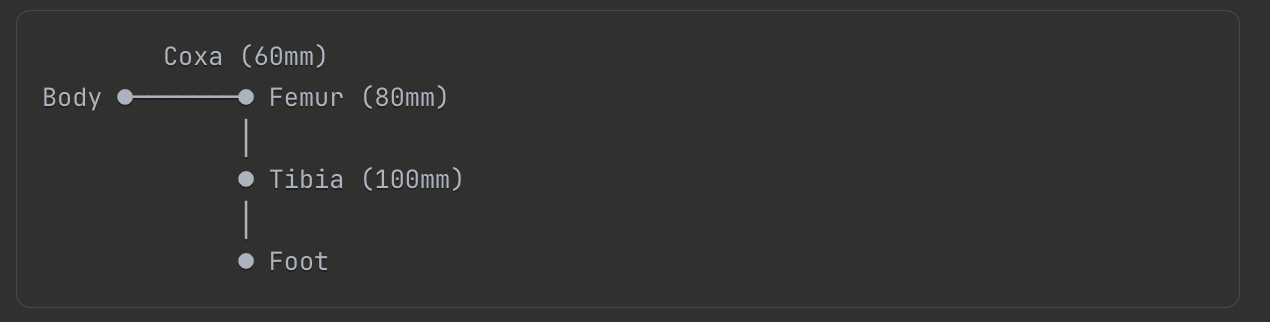
Body ●───────●─────── Femur (80mm)

│

●──── Tibia (100mm)

│

● Foot



**Joint Specifications:**

* **Coxa Joint:** Horizontal rotation, ±45° range
* **Femur Joint:** Vertical rotation, 0-120° range
* **Tibia Joint:** Vertical rotation, 0-150° range

**Total DOF:** 3 per leg × 6 legs = 18 DOF

#### **Leg Positioning on Body**

* **Front legs (RF, LF):** ±30° from center axis
* **Middle legs (RM, LM):** ±90° from center axis
* **Rear legs (RR, LR):** ±150° from center axis

### **5.2 BODY STRUCTURE**

#### **Overall Dimensions**

* **Length:** 450mm (including head)
* **Width:** 350mm (maximum leg span)
* **Height:** 250mm (standing position)
* **Ground Clearance:** 80mm minimum

#### **Body Segments**

1. **Head:** 80mm × 80mm × 60mm
   * Houses camera with pan-tilt mount
   * Front ultrasonic sensors
2. **Thorax:** 200mm × 150mm × 80mm
   * All 6 legs attach here
   * Raspberry Pi mounting
   * PCA9685 boards mounting
3. **Abdomen:** 150mm × 120mm × 100mm
   * Battery compartment
   * Gripper mechanism base
   * Power distribution board

### **5.3 GRIPPER DESIGN**

#### **Specifications**

* **Type:** 2-finger parallel jaw
* **Opening Range:** 50mm (closed) to 150mm (open)
* **Gripping Force:** 1-3 kg adjustable
* **Weight Capacity:** 0.5 kg typical waste items
* **Actuation:** Single MG90S servo with linkage
* **Linkage Ratio:** 1:2 (servo angle to jaw separation)

### **5.4 3D PRINTED COMPONENTS**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Component** | **Quantity** | **Material** | **Infill** | **Est. Print Time** |
| Body segments | 3 | PLA/PETG | 30% | 8 hours |
| Servo brackets | 24 | PLA | 50% | 12 hours |
| Leg segments | 18 | PETG | 40% | 10 hours |
| Foot pad holders | 6 | TPU | 100% | 2 hours |
| Gripper jaws | 2 | ABS | 50% | 3 hours |
| Camera mount | 1 | PLA | 30% | 2 hours |
| Sensor housings | 6 | PLA | 20% | 3 hours |
| Battery tray | 1 | PETG | 30% | 4 hours |

**Total Print Time:** 40-50 hours

## **6. AI/ML VISION SYSTEM**

### **6.1 OBJECT DETECTION MODEL**

#### **YOLOv5 (You Only Look Once v5)**

* **Framework:** PyTorch
* **Model Variant:** YOLOv5s (small) or YOLOv5n (nano)
* **Input Resolution:** 320×320 or 416×416 pixels
* **Inference Speed:** 30-60 FPS on Raspberry Pi 4
* **Model Size:** 7-14 MB after quantization

**Waste Categories to Detect:**

1. Plastic bottle
2. Aluminum can
3. Paper/cardboard
4. Glass bottle
5. Organic waste
6. General trash

**Alternative Models:**

* MobileNet SSD (TensorFlow Lite)
* EfficientDet-Lite

### **6.2 DATASET PREPARATION**

#### **Data Collection Requirements**

* **Total Images:** 1000-2000 minimum
* **Per Class:** 150-300 images
* **Sources:**
  + Custom photography (recommended)
  + Public datasets: TACO, TrashNet
  + Data augmentation for variety

#### **Annotation**

* **Tool:** LabelImg or Roboflow
* **Format:** YOLO format (class x\_center y\_center width height)
* **Data Split:**
  + Training: 70% (700-1400 images)
  + Validation: 20% (200-400 images)
  + Testing: 10% (100-200 images)

### **6.3 MODEL TRAINING**

#### **Training Environment Options**

1. Google Colab (free GPU access)
2. Local machine with NVIDIA GPU
3. Cloud platforms (Azure ML, AWS SageMaker)

#### **Training Parameters**

* **Epochs:** 100-200
* **Batch Size:** 16 (adjust for GPU memory)
* **Learning Rate:** 0.01 initial
* **Image Size:** 416×416 pixels
* **Augmentation:** Flip, rotate, scale, color jitter

**Training Duration:**

* 2-6 hours on GPU (NVIDIA GTX 1660 or better)
* 24-48 hours on CPU (not recommended)

### **6.4 MODEL OPTIMIZATION**

#### **Optimization Steps**

1. **Export to ONNX format** for compatibility
2. **INT8 Quantization:** Reduces model size by 75%
3. **Model Pruning:** Remove redundant weights (optional)

**Expected Performance:**

* Model Size: 7-14 MB (quantized)
* Inference Time: 50-150ms per frame
* Accuracy: 85-92% mAP (mean Average Precision)

### **6.5 DISTANCE ESTIMATION**

#### **Monocular Camera Method**

Uses known object size and focal length calculation:

**Formula:** Distance (m) = (Actual Height × Focal Length) / Bbox Height (pixels)

**Calibration Process:**

1. Place known objects (e.g., plastic bottle 0.20m tall) at measured distances
2. Record bounding box heights at 0.5m, 1.0m, 2.0m
3. Calculate focal length constant
4. Test accuracy across operating range

**Example Calculation:**

* Bottle height: 0.20m
* Bbox height at 1m: 150 pixels
* Focal length = (150 × 1.0) / 0.20 = 750 pixels

## **7. ASSEMBLY INSTRUCTIONS**

### **7.1 PHASE 1: COMPONENT PREPARATION (Week 1-2)**

#### **Step 1: Inventory Check**

* Verify all components against Bill of Materials
* Test battery voltage (should be 11.1V ± 0.3V)
* Check servo horns and mounting screws
* Inspect 3D printed parts for defects

#### **Step 2: Servo Testing & Calibration**

1. Connect each servo individually to PCA9685
2. Power with 6V supply
3. Test sweep from 0-180°
4. Mark 90° center position on servo horn with marker
5. Record any faulty servos for replacement

#### **Step 3: Raspberry Pi Setup**

1. Flash Raspberry Pi OS (64-bit) to SD card
2. Enable SSH and set hostname
3. Boot and update system
4. Install required Python libraries
5. Enable I2C and Camera interfaces via raspi-config

### **7.2 PHASE 2: MECHANICAL ASSEMBLY (Week 3-5)**

#### **Step 1: Body Construction**

1. Cut acrylic/aluminum plates (laser cutter or jigsaw)
2. Drill mounting holes:
   * Servo brackets: 3mm diameter
   * Standoffs: 2.5mm diameter
3. Sand all edges smooth (220 grit sandpaper)
4. Test-fit components before final assembly

#### **Step 2: Leg Assembly (Per Leg - 20-30 minutes each)**

1. Attach servo bracket to coxa servo
2. Mount coxa servo to body at correct angle
3. Secure with M3×10mm screws (apply Loctite threadlocker)
4. Connect coxa link (60mm) to servo horn
5. Mount femur servo to coxa link
6. Connect femur link (80mm)
7. Attach tibia servo to femur link
8. Connect tibia link (100mm) with foot pad
9. Install rubber foot pad
10. Route and secure wiring along leg

**Repeat for all 6 legs**

#### **Step 3: Gripper Assembly**

1. Assemble 3D printed gripper linkage
2. Mount base rotation servo (MG90S)
3. Attach jaw open/close servo
4. Test opening range: 50mm to 150mm
5. Optional: Install force sensors on jaws

#### **Step 4: Camera Mount**

1. Assemble pan-tilt mechanism with 2× SG90 servos
2. Attach camera module to mount
3. Secure to head section
4. Route camera ribbon cable carefully

### **7.3 PHASE 3: ELECTRONICS INTEGRATION (Week 6-7)**

#### **Step 1: Power Distribution Assembly**

1. Solder components on perfboard:
   * XT60 female connector
   * Emergency stop switch terminals
   * 10A fuse holder
   * Screw terminals for regulators
2. Install buck converters (adjust voltages)
3. Add 4700µF capacitor across 6V servo rail
4. Install LED indicators

#### **Step 2: Component Mounting**

1. Mount Raspberry Pi on thorax using M2.5 standoffs
2. Secure PCA9685 boards near servo clusters
3. Install power distribution board centrally
4. Place battery in abdomen (Velcro straps)
5. Mount sensors: ultrasonic (front/sides), IR (downward), IMU (center)
6. Install emergency stop button (accessible location)

#### **Step 3: Wiring**

1. Connect battery to power board via XT60
2. Wire emergencies stop in series with battery positive
3. Connect voltage regulators to distribution board
4. Wire all servos to PCA9685 channels (see Section 4.3)
5. Connect sensors to GPIO pins (see Section 4.2)
6. Build voltage dividers for ultrasonic echo pins
7. Connect I2C devices (PCA9685, MPU6050)
8. Verify common ground connections

#### **Step 4: Cable Management**

1. Route power cables along body underside
2. Use cable ties every 5cm
3. Keep signal wires away from high-current wires
4. Leave slack at all joint locations
5. Hot glue cables at fixed attachment points
6. Label all connections

## **8. TESTING & CALIBRATION**

### **8.1 UNIT TESTING CHECKLIST**

|  |  |  |  |
| --- | --- | --- | --- |
| **Test ID** | **Component** | **Test Procedure** | **Pass Criteria** |
| T-001 | Power System | Measure voltage rails | 11.1V, 6V, 5V, 3.3V (±5%) |
| T-002 | Servos | Sweep all servos 0-180° | Smooth motion, no noise |
| T-003 | Ultrasonic | Test at known distances | ±3cm accuracy |
| T-004 | Camera | Capture test image | Clear, focused image |
| T-005 | Object Detection | Test on waste samples | >80% confidence |
| T-006 | Locomotion | Walk 1 meter forward | Stable gait, straight |
| T-007 | Gripper | Pick up bottle | Successful grip |
| T-008 | Obstacle Avoid | Place object in path | Robot stops/avoids |
| T-009 | Battery Life | Full operation | >45 minutes |
| T-010 | Emergency Stop | Press during motion | All motors halt |

### **8.2 CALIBRATION PROCEDURES**

#### **Servo Calibration**

1. Position each servo to physical 90° angle
2. Record PWM value at this position
3. Calculate offset for each servo
4. Save calibration data to configuration file

#### **Camera Calibration**

1. Print checkerboard pattern (9×6, 25mm squares)
2. Capture 20+ images from different angles
3. Calculate camera intrinsic matrix
4. Calculate lens distortion coefficients
5. Save calibration parameters

#### **Distance Estimation Calibration**

1. Place known objects at measured distances
2. Record bounding box dimensions
3. Calculate focal length constant
4. Verify accuracy across 0.5m - 3m range

### **8.3 INTEGRATION TESTING**

#### **Scenario 1: Single Object Collection**

1. Place plastic bottle 1 meter from robot
2. Start autonomous operation
3. Observe complete cycle: Detect → Approach → Grip → Dispose
4. Measure completion time (target: 60-90 seconds)

#### **Scenario 2: Multiple Objects**

1. Place 3 different waste items in view
2. Verify robot prioritizes closest object
3. Collect all items sequentially
4. Measure overall success rate (target: >80%)

#### **Scenario 3: Obstacle Course**

1. Create path with obstacles between robot and waste
2. Verify ultrasonic sensors detect obstacles
3. Observe path replanning behavior
4. Successful navigation to target

### **8.4 PERFORMANCE METRICS**

|  |  |  |
| --- | --- | --- |
| **Metric** | **Target** | **Measurement Method** |
| Detection Accuracy | >85% | Test on 50 objects |
| Approach Precision | ±10cm | Measure final position |
| Grip Success Rate | >90% | 20 attempts |
| Walking Speed | 0.15 m/s | Time over 1 meter |
| Battery Life | 45-60 min | Continuous operation |
| Obstacle Detection | 10-200cm | Sensor range test |

## **9. BUDGET & COST ANALYSIS**

### **9.1 DETAILED COST BREAKDOWN (USD)**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Category** | **Component** | **Qty** | **Unit Price** | **Total** |
| **Processing** | Raspberry Pi 4 (4GB) | 1 | $55 | $55 |
|  | MicroSD Card 64GB | 1 | $12 | $12 |
| **Servos** | MG996R (legs) | 18 | $8 | $144 |
|  | MG90S (gripper) | 4 | $5 | $20 |
| **Controllers** | PCA9685 Driver | 2 | $12 | $24 |
| **Vision** | Pi Camera V2 | 1 | $25 | $25 |
| **Sensors** | HC-SR04 Ultrasonic | 3 | $3 | $9 |
|  | IR Proximity | 2 | $4 | $8 |
|  | MPU6050 IMU | 1 | $6 | $6 |
| **Power** | LiPo 3S 5000mAh | 1 | $40 | $40 |
|  | Buck Converters | 2 | $7 | $14 |
|  | LiPo Charger | 1 | $25 | $25 |
|  | XT60, Fuses, etc | - | $10 | $10 |
| **Mechanical** | Acrylic/Al Plates | 1 set | $30 | $30 |
|  | 3D Printing (1kg) | 1 | $20 | $20 |
|  | Brackets, Hardware | 1 set | $30 | $30 |
| **Electronics** | Wires, Resistors, etc | 1 set | $30 | $30 |
| **Misc** | Adhesives, Ties, LEDs | 1 set | $15 | $15 |
| **Tools** | Soldering Kit (opt) | 1 | $30 | $30 |
|  | Multimeter (opt) | 1 | $20 | $20 |
|  | **TOTAL** |  |  | **$567** |

### **9.2 COST OPTIMIZATION**

**Potential Savings:**

* Use lab equipment (tools, multimeter): Save $50
* 3D print servo brackets instead of metal: Save $15
* Buy servo bulk packs (10-20% discount): Save $25-30
* Use recycled materials for chassis: Save $20

**Optimized Budget:** $450-500

## **10. EXPECTED OUTCOMES**

### **10.1 TECHNICAL ACHIEVEMENTS**

#### **Hardware Deliverables**

1. **Functional Hexapod Robot**
   * 18-DOF locomotion system
   * Stable tripod gait walking
   * 0.5-1.0 kg payload capacity
   * 45-60 minute battery life
2. **AI Vision System**
   * Real-time object detection (30-60 FPS)
   * 6 waste categories classified
   * 85% classification accuracy
   * Distance estimation ±10cm
3. **Autonomous Operation**
   * Obstacle detection and avoidance
   * Self-navigation to targets
   * Adaptive gripping mechanism
   * Error recovery capabilities

### **10.2 DEMONSTRATION SCENARIOS**

#### **Live Demo (5-7 minutes)**

* Place 5 different waste items on floor
* Robot autonomously detects, approaches, grips, and disposes
* Success rate target: 4/5 objects (80%)
* Showcase obstacle avoidance
* Display waste classification on screen

### **10.3 PROJECT DELIVERABLES**

**Documentation:**

1. Technical report (this document)
2. Circuit diagrams (Fritzing/KiCad)
3. 3D CAD models (STL files)
4. Source code repository (GitHub)
5. User operation manual
6. Video demonstration (5-10 minutes)

**Presentation Materials:**

1. PowerPoint slides (15-20 slides)
2. Live demonstration
3. Project poster (A1 size)

### **10.4 LEARNING OUTCOMES**

**Technical Skills:**

* Embedded systems programming
* Computer vision and AI/ML
* Robotics kinematics
* Power electronics design
* Mechanical design and fabrication
* System integration

**Soft Skills:**

* Project management
* Technical documentation
* Teamwork and collaboration
* Problem-solving
* Presentation skills

## **11. SAFETY CONSIDERATIONS**

### **11.1 ELECTRICAL SAFETY**

#### **LiPo Battery Safety Rules**

⚠️ **CRITICAL:**

1. Never discharge below 3.0V per cell (9.0V total)
2. Never charge unattended
3. Always use balance charger
4. Store in LiPo safety bag
5. Never puncture or damage battery
6. If battery swells/smokes: place in metal container outside

#### **Working with Electronics**

* Disconnect power before wiring
* Use insulated tools
* Verify polarity before connections
* Install fuses for overcurrent protection
* Wear anti-static wrist strap

### **11.2 MECHANICAL SAFETY**

#### **During Assembly & Testing**

* Servos generate significant torque under load
* Keep fingers away from moving parts
* Use emergency stop during all tests
* File down sharp edges on cut materials
* Wear safety glasses when cutting/drilling

#### **Testing Procedures**

1. Test servos individually before assembly
2. Test walking with robot elevated initially
3. Clear testing area of obstacles and people
4. Have emergency stop accessible
5. Monitor for overheating components

### **11.3 OPERATIONAL SAFETY**

**Pre-Operation Checklist:**

* Battery voltage >11.0V
* Emergency stop tested and functional
* All connections secure
* Testing area clear
* Fire extinguisher accessible

## **12. REFERENCES & DATASHEETS**

### **12.1 COMPONENT DATASHEETS**

**Microcontrollers:**

1. Raspberry Pi 4 Documentation: raspberrypi.com/documentation
2. Raspberry Pi Camera V2: raspberrypi.com/documentation/accessories/camera.html

**Motor Controllers:** 3. PCA9685 Datasheet: www.nxp.com/docs/en/data-sheet/PCA9685.pdf 4. MG996R Servo: towerpro.com.tw/product/mg996r

**Sensors:** 5. HC-SR04 Ultrasonic: cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf 6. MPU6050 IMU: invensense.tdk.com/wp-content/uploads/2015/02/MPU-6000-Datasheet1.pdf

**Power:** 7. LM2596 Buck Converter: www.ti.com/lit/ds/symlink/lm2596.pdf 8. LiPo Battery Safety: batteryuniversity.com/article/bu-409

### **12.2 ACADEMIC REFERENCES**

**Hexapod Robotics:** 9. Tedeschi, F., & Carbone, G. (2014). "Design of Hexapod Walking Robots" 10. Erden, M. S. (2008). "Free gait generation with reinforcement learning"

**Computer Vision:** 11. Redmon, J., & Farhadi, A. (2018). "YOLOv3: An Incremental Improvement" 12. Jocher, G. (2021). "YOLOv5: State-of-the-Art Object Detection"

**Waste Detection:** 13. Aral, R. A. (2018). "Classification of TrashNet Dataset" 14. Hong, J. (2020). "TrashCan: Semantically-Segmented Dataset"

### **12.3 SOFTWARE & TOOLS**

**Development:** 15. PyTorch: pytorch.org/docs 16. OpenCV: docs.opencv.org 17. ROS: wiki.ros.org

**Design Tools:** 18. Fusion 360: autodesk.com/products/fusion-360 19. Fritzing: fritzing.org 20. Roboflow: roboflow.com

## **APPENDICES**

### **APPENDIX A: QUICK REFERENCE TABLES**

#### **Power Requirements Summary**

|  |  |  |  |
| --- | --- | --- | --- |
| **Component** | **Voltage** | **Current** | **Power** |
| Raspberry Pi 4 | 5V | 3A | 15W |
| 18× MG996R Servos | 6V | 9A peak | 54W |
| Camera Module | 5V | 250mA | 1.25W |
| Sensors (all) | 3.3-5V | 200mA | 1W |
| **Total Peak** | - | - | **~70W** |

#### **I2C Device Addresses**

|  |  |  |
| --- | --- | --- |
| **Device** | **Address** | **Configurable** |
| PCA9685 Board 1 | 0x40 | Yes (A0-A5) |
| PCA9685 Board 2 | 0x41 | Yes (A0-A5) |
| MPU6050 IMU | 0x68 | Yes (AD0 pin) |

### **APPENDIX B: TROUBLESHOOTING GUIDE**

|  |  |  |
| --- | --- | --- |
| **Problem** | **Cause** | **Solution** |
| Servos jitter | Low current supply | Add capacitor, upgrade converter |
| No camera feed | Ribbon cable loose | Reseat cable, check enable |
| I2C not detected | Wiring error | Check connections, verify address |
| Robot tips | CoG too high | Lower battery position |
| Gripper fails | Binding linkage | Lubricate, check angles |
| Slow detection | Large model | Reduce input size, use YOLOv5n |

### **APPENDIX C: PROJECT TIMELINE**

|  |  |  |
| --- | --- | --- |
| **Week** | **Phase** | **Tasks** |
| 1-2 | Preparation | Component procurement, 3D printing |
| 3-5 | Mechanical | Body and leg assembly |
| 6-7 | Electronics | Wiring and integration |
| 8-9 | Software | AI training, programming |
| 10-11 | Testing | Integration and debugging |
| 12-13 | Final | Documentation, video, presentation |
| 14 | Defense | Project demonstration and exhibition |

## **CONCLUSION**

This documentation provides comprehensive specifications for building an ant-inspired hexapod robot with autonomous waste collection capabilities. The project demonstrates integration of mechanical design, electronics, computer vision, and artificial intelligence.

**Key Features:**

* 18-DOF walking locomotion
* AI-powered waste detection (>85% accuracy)
* Autonomous navigation and manipulation
* Complete technical specifications and datasheets
* Detailed assembly procedures
* Budget: ~$550-600

**Success Criteria:** ✓ Stable hexapod walking  
 ✓ Real-time object detection  
 ✓ Autonomous waste collection  
 ✓ 45+ minute operation time  
 ✓ Complete documentation