ECE 445 Senior Design Laboratory

Final Report

Magnetic-Wheeled Pipe-Climbing Robot for Power-Tool Rust Removal

Team 20

Huanyu Feng
Xuhao Yang
Xiaocheng Zhang
Junxiang Qin

Sponsor: Jiahuan Cui
TA: Yanbing Yang

May 16, 2026

Abstract

This project developed a magnetic-wheeled pipe-climbing robot for robotic rust removal on external carbon steel pipe surfaces. The system integrates a magnetic-wheel chassis, a power-tool rust-removal end effector, a pipe test platform, and a Raspberry Pi and ESP32 based control system. The final prototype uses magnetic wheel adhesion for pipe-surface locomotion, a lead-screw-driven end-effector mechanism for contact adjustment, HX711/load-cell feedback for pressure measurement, and an electrical architecture that separates chassis control from local end-effector control. Final functional verification showed normal operation of the adhesion, locomotion, rust-removal, contact-force regulation, and safety-control functions.

Contents

1 Introduction

1.1 Problem Statement and Motivation

Corrosion on external steel pipelines remains a persistent maintenance problem in industrial environments [1]. Manual rust removal with handheld tools can be labor-intensive, inconsistent, and hazardous, especially on curved, elevated, or confined pipe surfaces. In addition to exposing workers to dust, vibration, and debris, manual cleaning may lead to nonuniform surface preparation and reduced coating quality.

This project addresses these limitations by developing a magnetic-wheeled robotic platform capable of carrying an abrasive rust-removal end effector along the outside of a ferromagnetic pipe. The engineering problem is not only to remove rust, but to do so while maintaining adhesion, traction, controlled tool contact, and repeatable measurement of the cleaned surface, which are common concerns in magnetic climbing and robotic grinding systems [4, 5, 6].

1.2 Project Function and System Overview

The robot is divided into four major blocks: the magnetic-wheel chassis, the rust-removal end effector, the pipe test platform, and the control system. Figure 1 shows the top-level system block diagram.

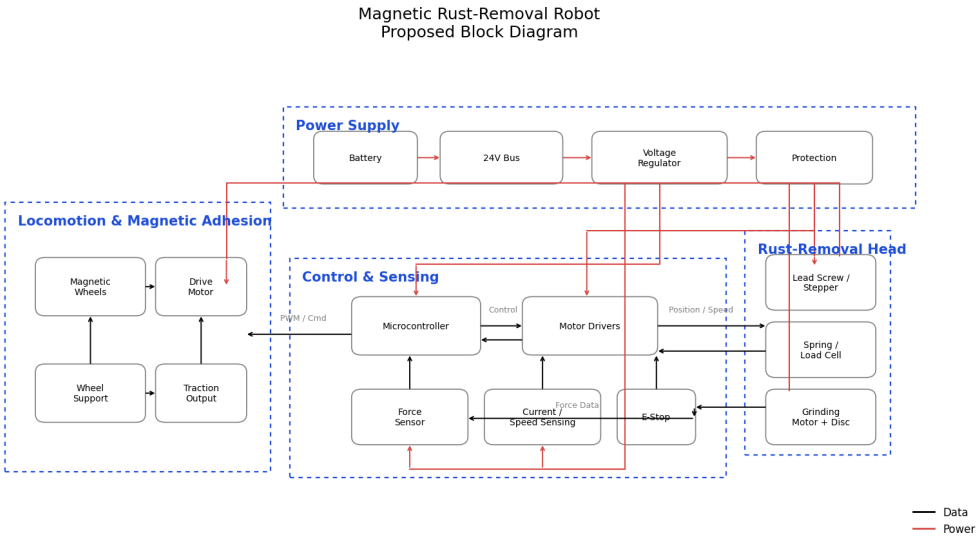


Figure 1: System block diagram of the magnetic-wheeled pipe-climbing rust-removal robot.

The chassis provides adhesion and locomotion, the end effector performs mechanical rust removal, the test platform supports repeatable pipe-surface experiments, and the control system coordinates motion, pressure sensing, and local actuator behavior.

1.3 Performance Requirements

Table 1 summarizes the system-level requirements used to evaluate the final prototype.

Table 1: System-level performance requirements.

Requirement	Target or success criterion	Verification location
Adhesion stability	Robot supports the 15 kg design mass on a ferromagnetic steel pipe during static, powered, and tool-on tests	Section 3.2
Locomotion stability	Robot completes a 1.0 m vertical climb with cumulative slip not exceeding 10 mm and repeatability within ±20 mm over repeated 1.0 m travel tests	Section 3.2
Combined tangential load	Robot resists a design load of 307 N from weight and grinding reaction	Section 2.1
End-effector contact	Contact force is controlled or maintained near the selected setpoint; project target was within ±5 N	Section 3.3
Overload shutdown	Tool shutdown within 0.5 s when force exceeds 120 N	Section 3.4
Rust removal performance	Abrasive end effector completes at least a 300 mm cleaning path and targets a visual cleanliness grade consistent with ISO 8501-1 St 3 or SSPC-SP 11 [2, 3]	Section 3.3
Safety	Emergency stop, guarded testing, and controlled laboratory operation	Section 5.3

1.4 Report Organization

Chapter 2 presents the design procedure and details for the mechanical, electrical, and control blocks. Chapter 3 summarizes the final verification results. Chapter 4 reports the project cost estimate. Chapter 5 summarizes accomplishments, limitations, safety considerations, and future work.

2 Design

2.1 Mechanical Design

2.1.1 End Effector

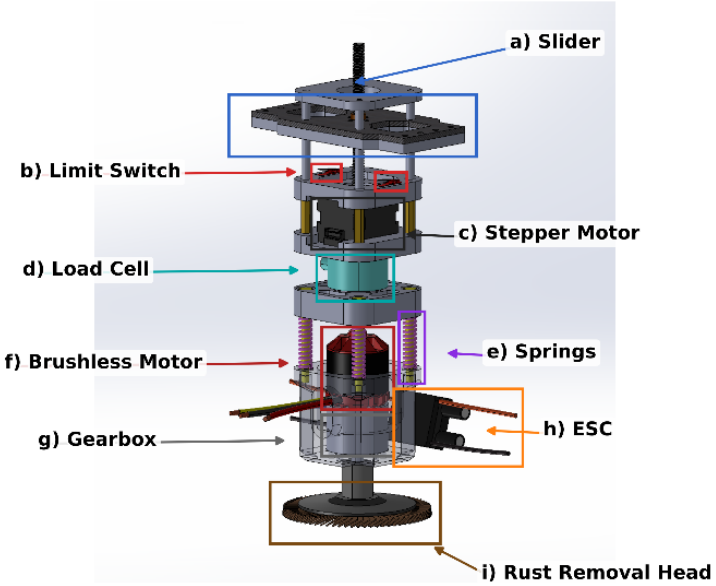
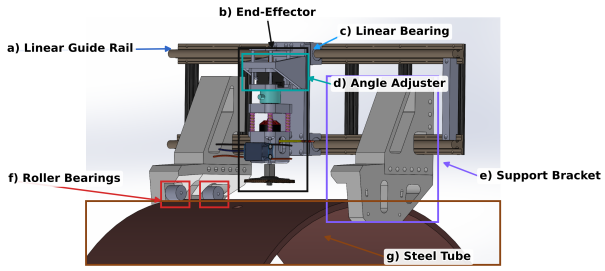


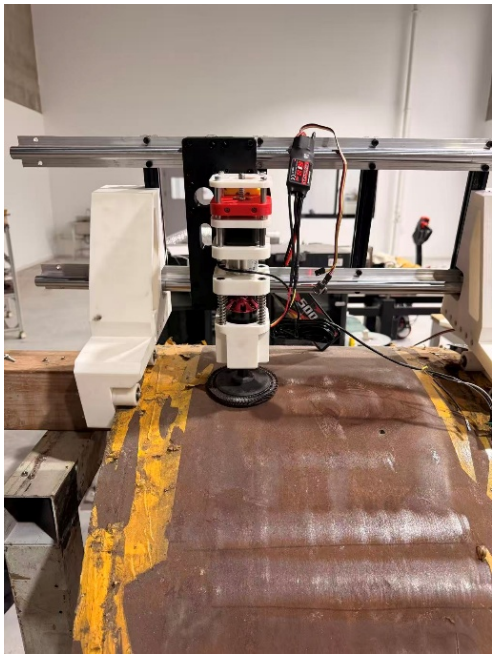
Figure 2: End Effector

In Figure 2, the rust removal end-effector consists of 9 main components. The slider and stepper motor are used to adjust the vertical position of the tool head, making sure the rust removal head maintains contact with the pipe surface. The load cell measures the normal contact force on the tool head, and the springs absorb vibration and help stabilize the contact between the tool and the curved surface. The brushless motor drives the rust-removal head through the gearbox, and the ESC controls the motor speed. Limit switches are included to define the zero position every time the end-effector reaches its highest position.

2.1.2 Test Platform



(a) CAD model

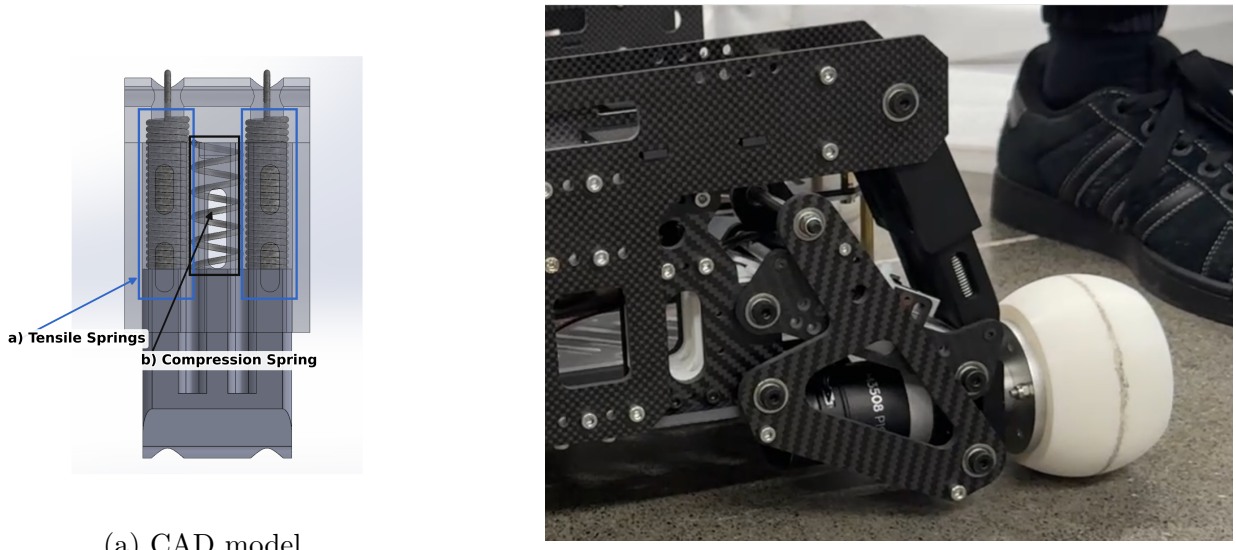


(b) Test Platform

Figure 3: Test Platform

Before the end-effector was integrated into the robot chassis, we designed a test platform to evaluate its rust removal performance under controlled conditions. As shown in Figure 3, the platform provides three degrees of freedom to reproduce the main motion and adjustment requirements of the pipe-climbing robot. The roller bearings allow the platform to rotate around the steel tube, while the horizontal linear guide rail enables axial movement along the pipe direction. In addition, the angle adjuster allows the end-effector to rotate in the vertical plane, thereby changing the contact angle between the tool head and the tube surface. The support brackets provide structural rigidity, and the linear bearings guide the sliding motion of the end-effector assembly.

2.1.3 Suspension System



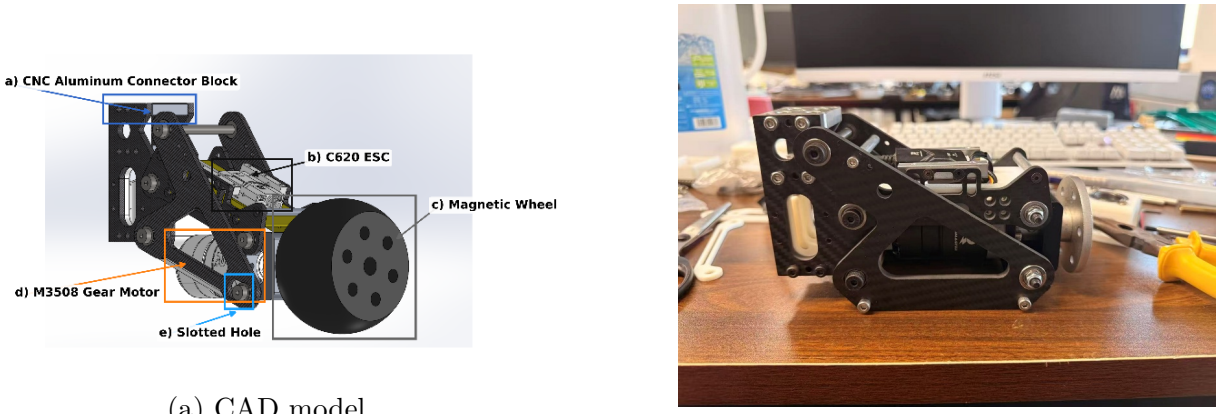
(a) CAD model

(b) Suspension System

Figure 4: Suspension System

Figure 4 shows the suspension system designed for the pipe-climbing robot. The mechanism adopts a hybrid spring configuration consisting of two tensile springs and one compression spring. Through spring selection and mechanical design, the suspension operates in tension when the robot is at the pipe's lower side and in compression when it moves to the pipe's upper side. This arrangement allows the wheel module to maintain adaptive contact with the curved steel surface under different climbing positions. In addition, the suspension system helps absorb vibration and reduce impact during motion, thereby improving contact stability, surface adaptability, and obstacle-crossing capability on uneven curved surfaces.

2.1.4 Wheel-Assembly



(a) CAD model

(b) Wheel-Assembly

Figure 5: Wheel-Assembly

In Figure 5, the wheel assembly consists of 5 main parts. The magnetic wheel provides both adhesion and traction on the steel pipe surface, while the gear motor drives the wheel to generate locomotion. The ESC is used to regulate motor operation. The CNC aluminum connector block connects the wheel module to the main chassis and improves the structural rigidity of the assembly. In addition, a slotted hole is introduced in the mounting structure to allow adjustment of the wheel camber angle. This enables the wheel to tilt inward or outward, so that it can better accommodate uneven curved surfaces and maintain stable contact with the pipe [4, 5].

2.1.5 Magnetic Wheel Part

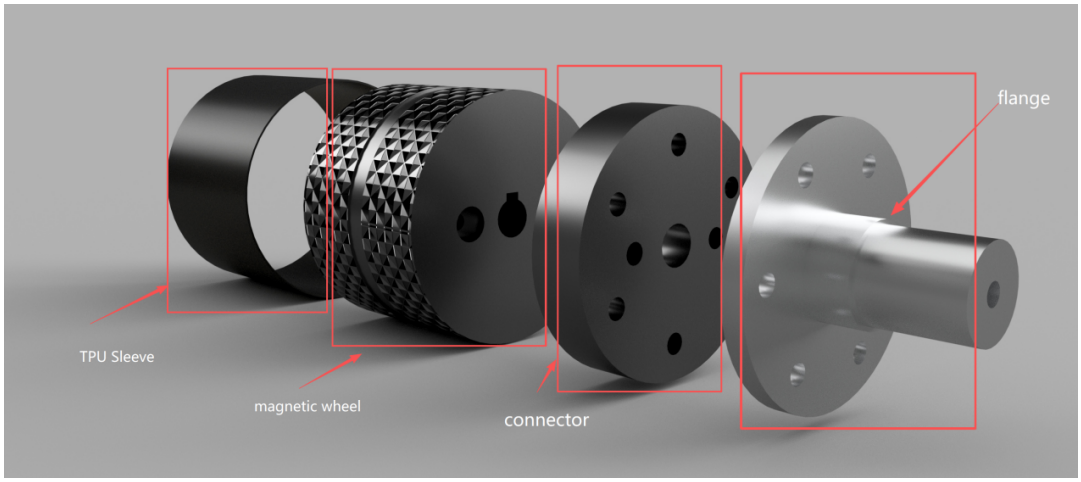


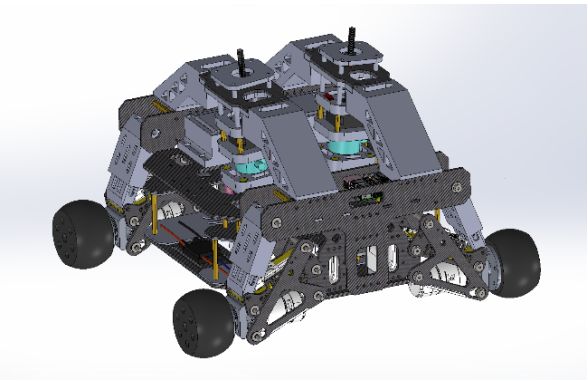
Figure 6: Magnetic wheel part

Figure 6 shows the magnetic wheel stack used in the locomotion module. Each magnetic wheel has a mass of 0.97 kg. While the raw permanent magnet provides a theoretical maximum pull-off force of 58 kg on a flat, ideal ferromagnetic surface, practical application on rusted pipes requires a protective interface. To prevent corrosion of the magnetic core and to increase the coefficient of friction, a 0.6 mm thick Thermoplastic Polyurethane (TPU) sleeve was designed and fitted over each wheel.

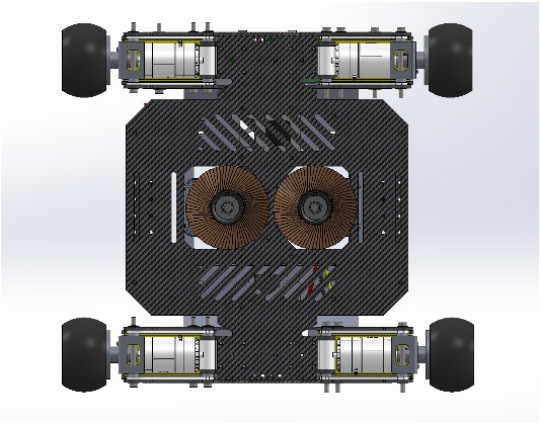
The introduction of the 0.6 mm TPU sleeve creates a non-magnetic gap, which significantly reduces the effective magnetic force due to flux leakage [5]. However, experimental verification confirmed that the effective adhesive force remains at least 69.1 N per wheel. The system maintains a sufficient safety factor to withstand the gravitational component during vertical climbing and the vibrations induced by the abrasive tool.

The magnetic wheel is integrated into the chassis via a robust mechanical stack. A standardized flange serves as the primary load-bearing interface between the motor shaft and the wheel. Custom-printed structural components were developed to bridge the flange to the magnetic core, allowing for precise axial alignment and easy maintenance or replacement of individual wheel units. These custom parts were optimized for infill density to ensure high rigidity while minimizing the overall mass of the locomotion block.

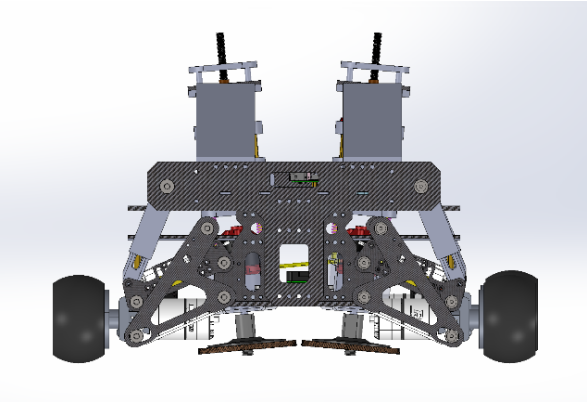
2.1.6 Overall Robot Assembly



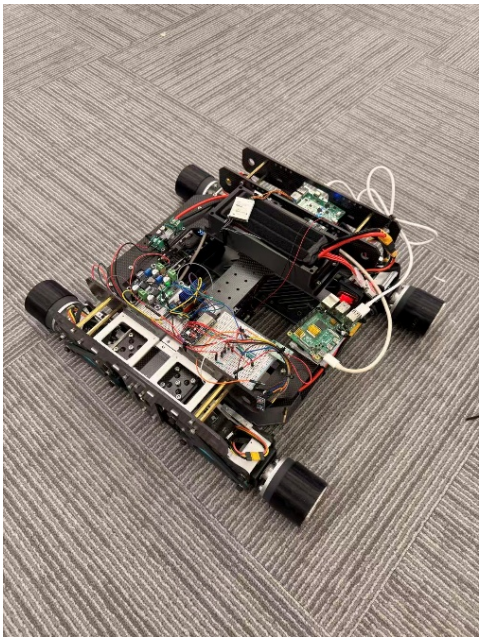
(a) Isometric View



(b) Bottom View



(c) Front View



(d) Final-Assembly

Figure 7: Overall Robot Assembly

Figure 7 presents the final assembly of the pipe-climbing rust removal robot. The platform consists of two rust removal end-effectors and four magnetic wheel assemblies mounted on a rigid chassis. The magnetic wheels provide adhesion and driving force on the steel pipe surface, while the end-effectors perform the rust removal process through rotary abrasive contact. The arrangement of the four wheel modules improves the robot's stability during movement, and the dual end-effector layout increases the system's effective working area.

2.1.7 Block Design

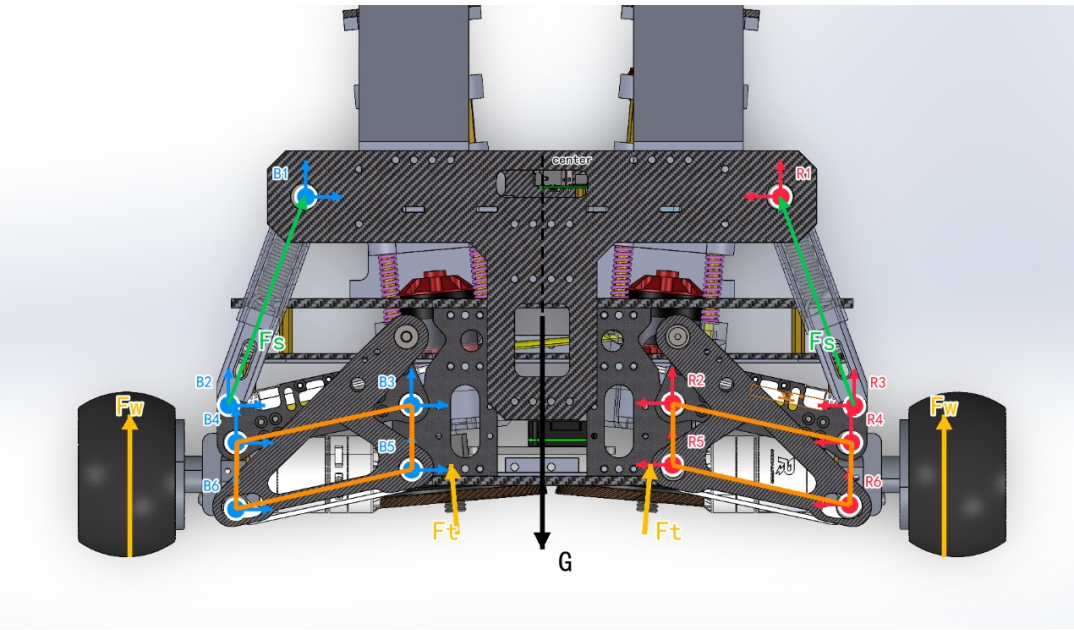


Figure 8: Annotated Force Diagram

2.1.8 Calculations and Force Analysis

2.1.8.1 Static Force Analysis of the Compound Suspension and Wheel-Linkage System

A simplified static force analysis was used to describe the load distribution of the robot suspension system. In Figure 8, the robot's weight is denoted as G, which acts downward at the center of the chassis. The wheel reaction force is represented by F_w, and the tool contact force is represented by F_t. Under static conditions, the total upward force should balance the robot's weight:

$$\sum F_y = 0, \qquad 2F_w + 2F_t - G = 0. \tag{1}$$

Therefore,

$$2F_w + 2F_t = G. \tag{2}$$

For one side of the robot, the vertical force balance can be written as:

$$F_w + F_t = \frac{G}{2}. \tag{3}$$

The spring force is denoted as F_s. Since springs are installed at an inclined angle, the spring force can be divided into a horizontal component and a vertical component:

$$F_{s,x} = F_s \cos\theta, \qquad F_{s,y} = F_s \sin\theta. \tag{4}$$

The spring force is calculated using Hooke's law:

$$F_s = k\Delta x + F_{s0}, \tag{5}$$

where k is the spring stiffness, Δx is the spring deformation, and F_{s0} is the initial spring preload.

For one suspension module, the simplified vertical equilibrium can be written as:

$$F_w + F_t + F_s \sin\theta = \frac{G}{2}. \tag{6}$$

This equation shows that the spring force helps support the suspension module and improves the contact between the wheels, the tool head, and the pipe surface. The pin forces at B_1 and B_6 are internal reaction forces generated by the linkage mechanism. These forces transfer the load through the suspension arms and allow the wheel assembly to adapt to the curved pipe surface.

2.1.8.2 Spring Stiffness Estimation

The suspension system consists of two tensile springs and one compression spring in each wheel module. This hybrid arrangement allows the robot to maintain stable contact with the external surface of the steel pipe as it moves through different angular positions. Since the direction of gravity relative to the pipe surface varies continuously, the suspension must provide different support forces at the pipe's highest and lowest positions.

The stiffness of both the compression spring and tensile spring was estimated using the standard helical spring stiffness equation:

$$k = \frac{Gd^4}{8D^3 n}, \qquad D = D_{\text{outer}} - d, \tag{7}$$

where k is the spring stiffness, G is the shear modulus, d is the wire diameter, D is the mean coil diameter, and n is the number of active coils.

For the compression spring, the parameters were $d = 2.0$ mm, $D_{\text{outer}} = 20.0$ mm, $L = 100.0$ mm, and $G = 80000$ N/mm^2. The number of active coils was estimated as:

$$n \approx \frac{L}{d} = 50. \tag{8}$$

Therefore, the estimated compression spring stiffness is:

$$k_c \approx 0.549 \text{ N/mm}. \tag{9}$$

For the tensile spring, the parameters were $d = 1.8$ mm, $D_{\text{outer}} = 18.0$ mm, $L_{\text{body}} = 60.0$ mm, and $G = 80000$ N/mm^2. The number of active coils was estimated as:

$$n \approx \frac{L_{\text{body}}}{d} = 33.33. \tag{10}$$

Thus, the estimated tensile spring stiffness is:

$$k_t \approx 0.741 \text{ N/mm}. \tag{11}$$

For an extension spring, the tensile force is calculated by:

$$F = F_0 + kx, \tag{12}$$

where F_0 is the initial tension and x is the extension displacement. With $F_0 = 2.0$ N and $x = 10.0$ mm, the tensile force of one spring is:

$$F \approx 9.407 \text{ N}. \tag{13}$$

2.1.8.3 Force Analysis Around the Pipe

When the robot moves around the pipe, the highest and lowest positions are the most critical cases. At the highest point, the robot is pushed toward the pipe surface, so the compression spring helps prevent excessive inward displacement. At the lowest point, the robot tends to move away from the pipe surface, so the tensile springs provide pulling force to maintain wheel contact.

The suspension stroke was assumed to be 30 mm, and the middle working position was $L_{\text{mid}} = 90$ mm. Therefore:

$$L_{\text{top}} = 75 \text{ mm}, \qquad L_{\text{mid}} = 90 \text{ mm}, \qquad L_{\text{bottom}} = 105 \text{ mm}. \tag{14}$$

The total required forces at the highest, middle, and lowest positions were assumed to be -5 kgf, 25 kgf, and 55 kgf, respectively. Since the robot has four wheel modules, the force required for each wheel module is:

$$F_{\text{top}} = -1.25 \text{ kgf}, \qquad F_{\text{mid}} = 6.25 \text{ kgf}, \qquad F_{\text{bottom}} = 13.75 \text{ kgf}. \tag{15}$$

The compression spring was assumed to be unloaded at the lowest position, while the tensile springs start stretching from the highest position:

$$L_{c0} = L_{\text{bottom}} = 105 \text{ mm}, \qquad L_{t0} = L_{\text{top}} = 75 \text{ mm}. \tag{16}$$

The required compression spring stiffness and equivalent suspension stiffness are calculated as:

$$k_c = \frac{-F_{\text{top}}}{L_{c0} - L_{\text{top}}}, \qquad k_{\text{eq}} = \frac{F_{\text{bottom}} - F_{\text{top}}}{L_{\text{bottom}} - L_{\text{top}}}. \tag{17}$$

Substituting the design values gives:

$$k_c = 0.0417 \text{ kgf/mm} = 0.409 \text{ N/mm}, \qquad k_{\text{eq}} = 0.500 \text{ kgf/mm} = 4.903 \text{ N/mm}. \tag{18}$$

Since each wheel module contains two tensile springs and one compression spring, the equivalent stiffness is:

$$k_{\text{eq}} = 2k_t + k_c. \tag{19}$$

Therefore, the required stiffness of each tensile spring is:

$$k_t = \frac{k_{\text{eq}} - k_c}{2} = 0.229 \text{ kgf/mm} = 2.247 \text{ N/mm}. \tag{20}$$

2.1.8.4 Force Check

The force generated by the combined spring system can be expressed as:

$$F(L) = 2k_t \max(L - L_{t0}, 0) - k_c \max(L_{c0} - L, 0). \tag{21}$$

Using this equation, the suspension force at the three key positions is:

$$F(75) = -1.25 \text{ kgf}, \qquad F(90) = 6.25 \text{ kgf}, \qquad F(105) = 13.75 \text{ kgf}. \tag{22}$$

These values match the target force requirements, which confirms that the spring combination can provide compression support at the highest point and tensile support at the lowest point.

2.2 Electrical and Control System Design

2.2.1 Design Procedure

The electrical control subsystem was designed around two main requirements. First, the robot needed a reliable operator-command path for switching the chassis between a locked safety state and an enabled driving state. Second, the rust-removal end effector needed a local controller that could switch between reset, closed-loop pressure control, and manual adjustment without depending on the full Raspberry Pi chassis software stack. The original alternative was to implement both functions on the Raspberry Pi, but that would make end-effector testing depend on ROS nodes, CAN setup, and complete vehicle bring-up.

The chosen design duplicates the same DBUS receiver signal into two independent control paths. The Raspberry Pi path controls chassis enable/disable behavior and publishes chassis motion commands when the chassis is enabled. The ESP32 path controls the end effector only: reset motion, pressure closed-loop control, manual screw motion, and derusting motor enable. This split was selected for testability and timing isolation. Step pulse generation, HX711 sampling, DBUS mode decoding, and ESC output remain local to the ESP32, while the Raspberry Pi path handles chassis-level safety locking and ROS motion output.

2.2.2 Signal-Flow and Operating Overview

Figure 9 shows the electrical control signal flow. The DBUS receiver signal is not used as a communication link between the Raspberry Pi and the ESP32. Instead, one receiver output is duplicated by a signal splitter so the same operator command source can be interpreted independently by the chassis controller and the end-effector controller.

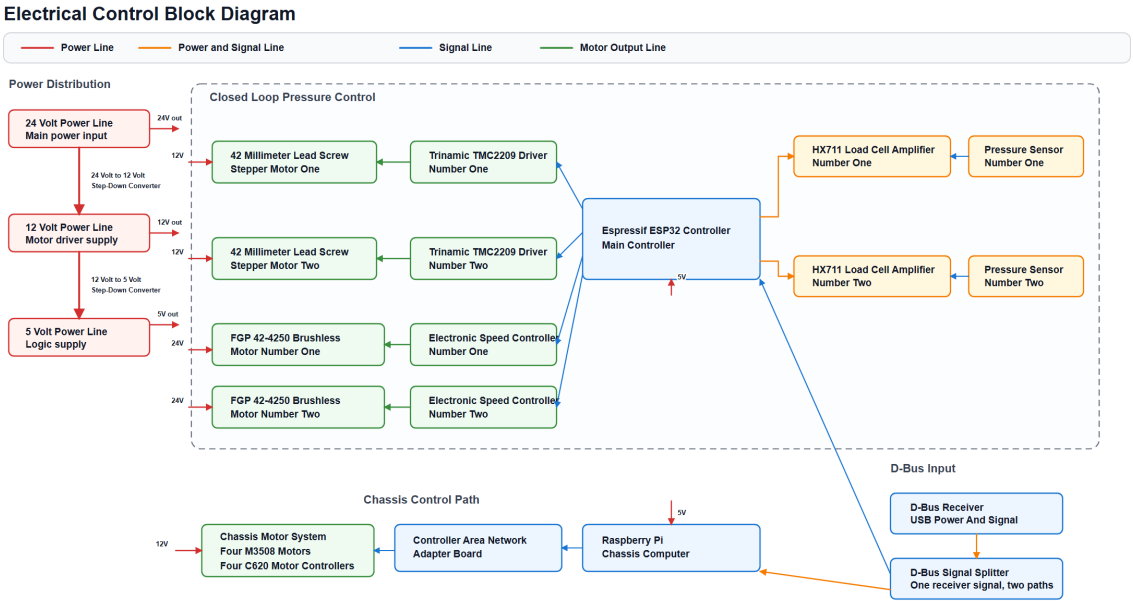


Figure 9: Electrical control block diagram for the chassis and end-effector control paths.

In Figure 9, the left and right branches refer to the two duplicated electrical signal paths, not to two processors communicating with each other. The Raspberry Pi branch reads only the chassis-related receiver inputs. In this report, the chassis enable switch is written as S_{L} because it is the left switch in the Pi-side software field `lsw`. The Pi also reads the left-stick horizontal command, `ls_x`, and the wheel command, `wheel`. If S_{L} is not active, the Pi locks the chassis command at zero. If S_{L} is active, `ls_x` becomes the forward/back chassis velocity command and `wheel` becomes the yaw command.

The ESP32 branch reads the end-effector-related receiver inputs. The right-side mode switch is written as S_{R} in this report and appears as `S2` in the firmware. This switch selects reset, closed-loop pressure control, or manual control. In manual mode, the ESP32 also reads DBUS channel 1, centered at 1024, as the right-stick up/down command for the screw actuator. This separation means the Pi decides whether the chassis is locked or allowed to drive, while the ESP32 decides whether the rust-removal head should reset upward, regulate contact pressure, or move manually. The DBUS signal is therefore a shared operator-command source, not a data bus between the two controllers. Table 2 summarizes the responsibility of each duplicated DBUS path.

Table 2: Control responsibility of each duplicated DBUS path.

Path	DBUS role	Controlled hardware
Raspberry Pi chassis path	Reads S_{L}, `ls_x`, and `wheel`; selects chassis lock or chassis driving state	ROS chassis controller and omni chassis motor system
ESP32 end-effector path	Reads S_{R}/`S2` and, in manual mode, DBUS channel 1; selects reset, closed-loop pressure control, or manual screw control	Stepper screw actuator, HX711/load cell, ESC, and brushless derusting motor

2.2.3 Raspberry Pi Chassis Control Design

The Raspberry Pi branch is responsible for chassis-level behavior. The receiver stream reaches the Pi software stack as a DBUS control message. The `vehicle_clean` node subscribes to the `dbus_control` topic, interprets the operator input, and publishes a `geometry_msgs/Twist` command to `omni_chassis_controller/reference`. The publish period is 15 ms, so the chassis reference is updated at approximately 66.7 Hz.

The key safety behavior is implemented in the Pi-side DBUS interpreter. The left switch state, S_{L}, is the chassis enable condition. In the current code this is the `lsw` field, and the active state is `DOWN`. When S_{L} is not active, the interpreter sets the linear and angular velocity commands to zero. This is the chassis locked or safety mode. When S_{L} is active, the interpreter maps the left-stick horizontal input to translational velocity and the wheel input to angular velocity. Table 3 lists the Pi-side DBUS fields used by this logic. The current chassis configuration uses a translational command limit of 2.5 m/s, a rotational command limit of 3.0 rad/s, and a 0.3 normalized dead zone; the main chassis-control parameters are summarized in Table 4.

Table 3: Raspberry Pi DBUS inputs used for chassis control.

Report notation	Software field	Use in chassis controller
S_{L}	`lsw`	Chassis enable switch; `DOWN` enables driving, other states force zero velocity
Left-stick horizontal	`ls_x`	Forward/back translational velocity command after dead-zone filtering
Wheel	`wheel`	Yaw angular velocity command after dead-zone filtering

Table 4: Raspberry Pi chassis-control parameters.

Parameter	Value	Function
DBUS device path	`tty_DBUS`	Receiver input device on the Pi
Controller-manager update rate	200 Hz	ROS 2 control update frequency
Chassis enable switch	$S_{\mathrm{L}} =$ `DOWN`	Selects safety lock or chassis driving
Command publish period	15 ms	Chassis reference publish period
Maximum translational command	2.5 m/s	Linear chassis velocity limit
Maximum rotational command	3.0 rad/s	Angular chassis velocity limit
Dead zone	0.3 normalized command	Receiver input noise rejection
Wheel radius	0.076 m	Omni-wheel kinematic parameter
Drive motors and controllers	Four M3508 motors with C620 controllers	Chassis wheel actuation hardware
Wheel transmission reduction	20.0	Motor-to-wheel mechanical reduction in the chassis description
Wheel PID gains	$K_p = 0.02$, $K_i = 0.0001$, $K_d = 0$	Wheel velocity control gains

2.2.4 ESP32 End-Effector Control Design

The ESP32 branch is responsible for the rust-removal head rather than the chassis. The DBUS receiver signal enters the ESP32 on GPIO8 as an inverted UART signal. After a valid DBUS frame is decoded, the S_{R} switch, implemented as `S2` in the firmware, is used as the end-effector mode selector. The ESP32 firmware also reads channel 1 when `S2`=3 so the operator can command proportional manual up/down screw motion with the right stick. The selected `S2` mode determines both screw-actuator behavior and ESC behavior.

In reset mode, the ESP32 commands the screw actuator toward the upward reset direction and commands the ESC stop pulse, so the derusting motor remains off. Table 5 summarizes the end-effector mode mapping.

Table 5: ESP32 DBUS inputs and end-effector mode mapping.

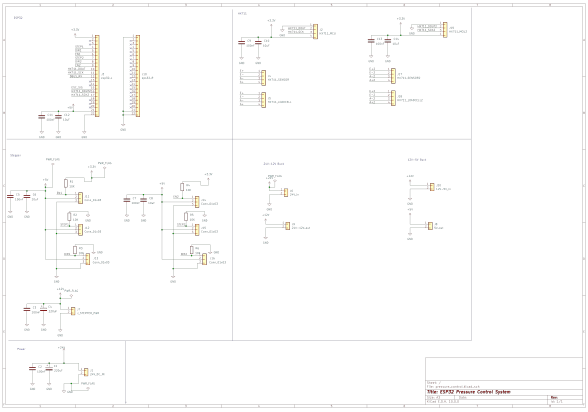
DBUS input	System mode	Screw/pressure behavior	ESC derusting motor behavior
S_R/S2 = 1	Reset	Closed loop stops; screw actuator moves upward or toward the reset direction until reset is complete	Stop pulse; derusting motor off
S_R/S2 = 2	Closed-loop pressure control	HX711 feedback controls the stepper screw actuator inside the target pressure window	Run pulse; derusting motor on
S_R/S2 = 3, CH1	Manual control	Channel 1, centered at 1024, directly commands right-stick up/down screw motion	Run pulse; derusting motor on
Invalid/no frame	Safe stop	Automatic DBUS-controlled motion is not serviced	Stop pulse

After this top-level mode selection, the ESP32 performs the low-level signal handling. The DBUS input is decoded through UART1 at 100 kbps using 8 data bits, even parity, and one stop bit. Each valid DBUS frame contains four 11-bit analog channels and two switch states. The firmware rejects frames whose channel values fall outside the expected DBUS range of 364 to 1684 or whose switch states are outside the valid range of 1 to 3. In manual mode, channel 1 is centered around 1024 and converted into signed stepper bursts after a dead band is removed. Table 6 lists the ESP32 wiring used for the DBUS, stepper, ESC, and HX711 signals.

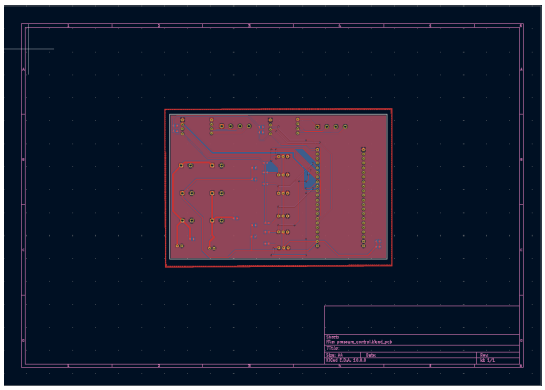
Table 6: ESP32 end-effector wiring and signal functions.

External module	Signal	ESP32 pin
Motor 1 stepper driver	STEP	GPIO4
Motor 1 stepper driver	DIR	GPIO5
Motor 1 stepper driver	EN	GPIO6
Motor 2 stepper driver	STEP	GPIO7
Motor 2 stepper driver	DIR	GPIO15
Motor 2 stepper driver	EN	GPIO16
ESC	Throttle signal	GPIO9
DBUS receiver	RX input	GPIO8
HX711	DT/DOUT	GPIO17
HX711	SCK	GPIO18

Figure 10 shows the pressure-control electronics for the ESP32 end-effector controller. The schematic defines the electrical connections for the ESP32, pressure-sensing interface, actuator-control signals, and supporting components. The PCB layout implements those schematic connections in a compact board so the HX711/load-cell path, stepper driver signals, DBUS input, and ESC output can be wired more reliably than with loose prototype wiring.



(a) ESP32 pressure-control schematic



(b) Pressure-control PCB layout

Figure 10: ESP32 pressure-control schematic and PCB implementation for the end-effector electronics.

The end-effector actuator hardware is built around a TMC2209 stepper driver, a 42 mm class 1.8 degree stepper motor, and a T8 lead screw with a 2 mm lead. With 1/16 microstepping, this gives 3200 microsteps per motor revolution and a theoretical screw displacement of 0.000625 mm per microstep. The rust-removal disc is driven by an FGP 42-4250 class brushless motor through an ESC. In the current ESP32 firmware, the ESC signal is a 50 Hz PWM-style pulse: reset mode and invalid DBUS frames command a 1000 μs stop pulse, while closed-loop and manual modes command a 1940 μs run pulse. Manual screw motion uses proportional step bursts from 30 to 300 steps based on the magnitude of DBUS channel 1.

The pressure-control target is a window rather than a single set point. The current test window is 12000 g to 13000 g, with a midpoint of 12500 g and a dead band of 500 g on either side. If the measured pressure is inside this window, the screw actuator holds position. If the pressure is below the window, the screw advances. If the pressure is above the window, the screw retracts. Table 7 gives the implemented pressure-loop parameters.

Table 7: ESP32 pressure closed-loop parameters.

Parameter	Value	Function
Low pressure threshold	12000 g	Lower edge of target window
High pressure threshold	13000 g	Upper edge of target window
Controller midpoint	12500 g	Nominal pressure target
Maximum correction	400 steps/cycle	Limits aggressive screw motion
Minimum active correction	60 steps/cycle	Avoids ineffective tiny moves
Maximum step change	160 steps/cycle	Smooths correction changes
Control interval	5 ms	Pressure-loop service period
Step pulse width	175 us	Stepper pulse timing
K_p	0.05 step/g	Proportional correction gain
K_i, K_d	0	Integral and derivative disabled

Using the microstep displacement above, the minimum active correction of 60 steps corresponds to approximately 0.0375 mm, and the maximum correction of 400 steps corresponds to approximately 0.25 mm. These limits prevent the controller from issuing corrections that are too small to overcome mechanical friction or too large to be safe during contact.

The ESC is controlled by a 50 Hz PWM-style throttle signal generated by the ESP32. The HX711 and stepper motor regulate contact force, while the ESC starts and drives the brushless motor used for rotating rust removal. In the current runtime configuration, reset mode and invalid DBUS frames command a 1000 μs stop pulse, while closed-loop and manual modes command a 1940 μs run pulse. Manual serial commands such as `esc 1500` and `esc 1940` are kept for bench testing the waveform before tying the behavior to the DBUS mode table.

2.3 Integrated System Design

The final system integrates the magnetic-wheel chassis, abrasive end effector, test platform, and control hardware into a single verification platform. During operation, the chassis moves the robot across the pipe surface, the end effector removes rust through abrasive contact, and the Raspberry Pi and ESP32 control paths coordinate chassis and local actuator behavior.

3 Verification

3.1 Verification Overview

Verification connects each result back to the requirements in Table 1. The final tests showed normal functional performance for the chassis, end effector, sensing/control path, and safety behavior. The result tables report the numeric requirement thresholds used during verification and the observed pass/fail outcome.

3.2 Chassis Locomotion and Adhesion Verification

Figure 11 shows the assembled robot used for chassis adhesion and locomotion verification. Table 8 summarizes the observed chassis verification results.

Figure 11: Assembled magnetic-wheeled robot used for adhesion and locomotion verification.

Table 8: Chassis locomotion and adhesion verification results.

Test	Metric	Result	Status
Static adhesion	Attached/detached; holding time	Robot remained attached for the required holding period with no detachment observed	Pass
Unloaded locomotion	Average speed and path stability	Robot moved along the pipe surface without obvious slip, detachment, or loss of direction control	Pass
Tool-on stability	Attachment and slip during rust removal	Robot remained attached while the rust-removal tool operated; no unsafe vibration or tool-induced sliding was observed	Pass
Inclined/vertical partial test	Stability under increased gravity component	Robot remained stable during inclined/vertical testing, with no unsafe detachment or uncontrolled sliding	Pass

3.3 End-Effector Rust-Removal Verification

The rust-removal test compares the pipe surface before and after abrasive contact under controlled tool settings. Figure 12 shows the rust-removal test surface used for end-effector verification. The target is a continuous cleaning path of at least 300 mm, with the cleaned region evaluated against ISO 8501-1 St 3 or SSPC-SP 11 [2, 3].

Figure 12: Rust-removal test surface used for end-effector verification.

The image shows a visibly cleaned track on the pipe surface after the end effector passed over the rusted region. The cleaned area is lighter and more uniform than the surrounding oxidized surface, which verifies that the abrasive head was able to remove surface rust during operation. During this test, the end effector completed the cleaning-path target, maintained contact within the intended $100N \pm 5$ N contact-force band, and showed no abnormal stall or loss of contact. Based on this visual verification, the rust-removal function is considered to have normal functional performance.

3.4 Control System Verification

Table 9 summarizes the observed control and safety verification results.

Table 9: Control system verification results.

Control function	Key requirement	Result	Status
Chassis command	Normal motion and stop response	Chassis responded to operator motion commands and returned to zero command when disabled	Pass
DBUS decoding	100 kbps 8E1; $S2 = 1/2/3$ mode selection	Receiver decoding was normal, and the reset, closed-loop, and manual modes were selected correctly	Pass
Pressure feedback	12000–13000 g operating window	HX711/load-cell feedback was received normally and supported force-feedback operation	Pass
Pressure loop	± 5 N contact-force target	Closed-loop screw adjustment maintained contact during nominal cleaning [6]	Pass
ESC output	1000 μs stop; 1940 μs run	ESC output followed the selected reset, closed-loop, and manual operating modes	Pass
Safety stop	120 N threshold; 0.5 s shutdown	Overload and safe-stop behavior disabled the actuator path within the required response time	Pass

3.5 System-Level Verification Summary

Table 10 summarizes the final functional verification status.

Table 10: System-level verification summary.

Requirement	Verification method	Result	Status
Magnetic adhesion	Static and tool-on tests	5-minute hold passed; 307 N design tangential-load case remained stable	Verified
Locomotion	Axial/circumferential or vertical motion tests	1.0 m travel target passed with slip below 10 mm and repeatability within $\pm$20 mm	Verified
Rust removal	Before-after surface comparison and tool tests	300 mm cleaning path completed with visible rust removal and contact maintained within $\pm$5 N target band	Verified
Safety	Emergency stop and guarded test procedure	120 N overload threshold and 0.5 s shutdown requirement were satisfied	Verified

4 Costs

Table 11 reports the documented prototype cost estimate used for this project.

Table 11: Prototype cost estimate.

Item	**Estimated Cost**	**Notes**
Magnetic wheel assemblies and drive motors	\$400	Four wheel modules
Rust-removal motors, ESCs, and grinding heads	\$200	Two end effectors
Stepper motor, lead screw, and force-sensing components	\$40	Force-control subsystem
Controller, wiring, low-voltage electronics, and sensors	\$80	MCU, ADC, protections
Battery or power-supply components	\$10	24 V bus and regulated rails
Carbon-fiber or aluminum structural parts and machining	\$200	Custom fabricated parts
Fasteners, connectors, and assembly materials	\$20	Miscellaneous hardware
Total Parts Cost	**\$960**	Documented parts total
Estimated labor (course prototype)	\$240	Documented labor estimate
Total Estimated Cost	**\$1200**	Parts plus estimated labor

5 Conclusion

5.1 Summary of Accomplishments

This project developed and functionally verified a magnetic-wheeled pipe-climbing rust-removal robot prototype. The final system combines a magnetic-wheel chassis, dual rust-removal heads, a normal-force adjustment mechanism, a Raspberry Pi chassis controller, and an ESP32 end-effector controller. Verification showed normal adhesion, locomotion, end-effector rust-removal, force-control, ESC, DBUS, and safety-stop behavior.

5.2 Limitations

The prototype met the functional verification targets reported in Chapter 3, but it remains a laboratory-scale system rather than an industrial field product. Remaining limitations include dependence on a controlled pipe fixture, limited environmental protection for dust and debris, limited onboard power integration, and a control interface that still relies on supervised operation.

5.3 Ethical and Safety Considerations

The main ethical motivation is reducing human exposure to hazardous rust-removal work [7]. However, it would be unethical to claim field readiness unless adhesion margin, traction margin, contact-force stability, and abnormal-event handling are robust under representative conditions. The prototype should be treated as a supervised laboratory system until industrial-grade safeguards and operating limits are implemented [8].

The most important safety concerns are the rotating abrasive tool, falling-object risk from possible adhesion loss, dust and debris exposure, noise, vibration, and electrical faults. Testing should use guarding, personal protective equipment, ventilation or debris collection, low-height pipe fixtures, an accessible emergency stop, secure wiring, and conservative shutdown logic [9, 10]. Failed trials, slip events, tool wear, and measurement uncertainty should be reported honestly [8].

5.4 Future Work

To make the system practical for real pipe-maintenance use, future versions should improve environmental robustness, autonomy, and deployment safety. The chassis could use interchangeable magnetic wheel sleeves or adjustable wheel modules to adapt to different pipe diameters, coatings, and surface roughness. The end effector could be upgraded with a sealed dust-collection shroud, replaceable abrasive cartridges, and more robust contact-force control for long cleaning passes [6]. At the system level, onboard battery packaging, weather-resistant electronics, automatic helical path planning, remote monitoring, and redundant safety interlocks would make the robot more suitable for supervised industrial deployment.

References

[1] Association for Materials Protection and Performance, "From Rust to Risk: Global Campaign Urges Action on Corrosion Crisis," Apr. 22, 2025. [Online]. Available: https://www.ampp.org/. Accessed: May 15, 2026.

[2] International Organization for Standardization, *Preparation of steel substrates before application of paints and related products – Visual assessment of surface cleanliness – Part 1*, ISO 8501-1.

[3] SSPC/NACE, *Power Tool Cleaning to Bare Metal*, SSPC-SP 11/NACE No. 3.

[4] W. Song, H. Jiang, T. Wang, and S. Zhu, "Design of permanent magnetic wheel-type adhesion-locomotion system for water-jetting wall-climbing robot," *Advances in Mechanical Engineering*, vol. 10, no. 7, 2018, doi: 10.1177/1687814018787378.

[5] A. Sahbel, A. Abbas, and T. Sattar, "Experimental and numerical optimization of magnetic adhesion mechanism," *International Journal of Mechanical Engineering and Robotics Research*, 2019. [Online]. Available: https://www.ijmerr.com/uploadfile/2018/1228/20181228035802817.pdf. Accessed: May 15, 2026.

[6] T. Li, Q. Sun, C. Wang, X. Yuan, and K. Wang, "A robot floating grinding and rust removal approach based on composite force-position fuzzy control," *Sensors*, vol. 25, no. 7, p. 2204, 2025, doi: 10.3390/s25072204.

[7] K. H. Koh *et al.*, "Teleoperated service robotic system for on-site surface rust removal and protection of high-rise exterior gas pipes," *Automation in Construction*, vol. 125, p. 103609, 2021, doi: 10.1016/j.autcon.2021.103609.

[8] IEEE, "IEEE Code of Ethics," 2016. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. Accessed: May 15, 2026.

[9] Occupational Safety and Health Administration, "Hand and Power Tools." [Online]. Available: https://www.osha.gov/hand-power-tools. Accessed: May 15, 2026.

[10] Occupational Safety and Health Administration, "Personal Protective Equipment." [Online]. Available: https://www.osha.gov/personal-protective-equipment. Accessed: May 15, 2026.