Design Document: Autonomous Ammunition Loading and Firing Robotic System

Group 35

Team Members:

Yidong Zhu, Yuxuan Nai, Xinchen Yao, Xiaoman Li

Course: ME470 Senior Design Project

Date: March 24, 2026

Contents

1 Introduction

1.1 Problem Statement

In competitive robotics environments such as the RoboMaster competition and other competitive robotics projectile systems, rapidly and dynamically managing ammunition is a decisive tactical advantage. Current standard loading mechanisms primarily rely on predefined, hard-coded spatial coordinates. These systems are highly susceptible to mechanical jamming and operational failure when projectiles are displaced by robot movement, collisions, or unstructured staging environments. Consequently, there is a critical engineering requirement for an autonomous manipulation system capable of visually identifying, grasping, and loading unstructured ammunition into a launching mechanism in real-time, thereby eliminating the need for human intervention or perfectly aligned staging setups.

1.2 Solution Overview & Visual Aid

Our proposed solution is an integrated, fully autonomous robotic arm and launching system. The system is initiated via a user-friendly trigger (physical button or remote command). Once activated, the system leverages an onboard RGB-D camera feed or vision setup to observe the staging area behind it and determine the spatial coordinates of ammunition. To achieve robust manipulation, an adaptable, intelligent decision-making policy (e.g., a lightweight object detection algorithm like YOLOv5 [1] integrated with a classical control pipeline) processes the data. Guided by this policy, a Linux-based compute unit solves the necessary inverse kinematics and trajectory planning, while an STM32-based controller orchestrates the low-level movement of a multi-axis robotic arm via a CAN bus network [2]. The arm dynamically adjusts its trajectory to retrieve the target projectile, aligns it with a custom loading port, secures it, and executes the launch sequence.

Operational Workflow (Conceptual Visual Aid)

(1) Operator presses start button or sends remote command.

(2) Rear-facing RGB-D camera scans unstructured dart staging area.

(3) Vision policy estimates dart position and confidence score.

(4) Linux computer solves inverse kinematics and plans arm motion; STM32 executes low-level control.

(5) Robotic arm grasps dart and aligns with loading port.

(6) Launcher receives fire-enable signal and executes discharge.

(7) Cycle status is logged; system returns to search state for next dart.

Figure 1: Conceptual usage flow of the autonomous ammunition retrieval, loading, and firing process.

Figure 2: Physical prototypes of the operational system: the multi-axis robotic arm with custom gripper (left) and the friction-wheel launching mechanism (right), demonstrating the hardware utilized for the autonomous staging and firing sequence.

1.3 High-Level Requirements

1. Autonomous Retrieval Rate: The perception and manipulation subsystems must successfully detect, grasp, and load a randomly placed projectile in the staging area with a success rate of $\geq 80\%$, completing each individual cycle in under 5 seconds.
2. Launch Execution Reliability: Once the projectile is successfully grasped and secured into the loading port, the firing mechanism must reliably activate and propel the dart in a general forward direction. The system must achieve an $\geq 80\%$ mechanical discharge success rate without stalling or jamming, independent of ballistic targeting accuracy.
3. System Endurance: The combined electromechanical system (including the CAN communication network and motor actuations) must operate continuously for 100 consecutive reload-and-fire cycles without critical software crashes or hardware failures.

2 Design

2.1 Block Diagram

The system architecture is partitioned into four primary modules: Perception, Control, Actuation, and Power. The Perception Module (RGB-D Camera and computation unit) outputs projectile coordinates via UART to the Control Module (Linux computer). The Control Module processes trajectory planning and transmits high-frequency torque/position commands via a dual CAN bus to the Actuation Module (robotic arm and launcher motors). The Power Module distributes a regulated 24V DC to the motors and a stepped-down 5V DC to the logic controllers and sensors.

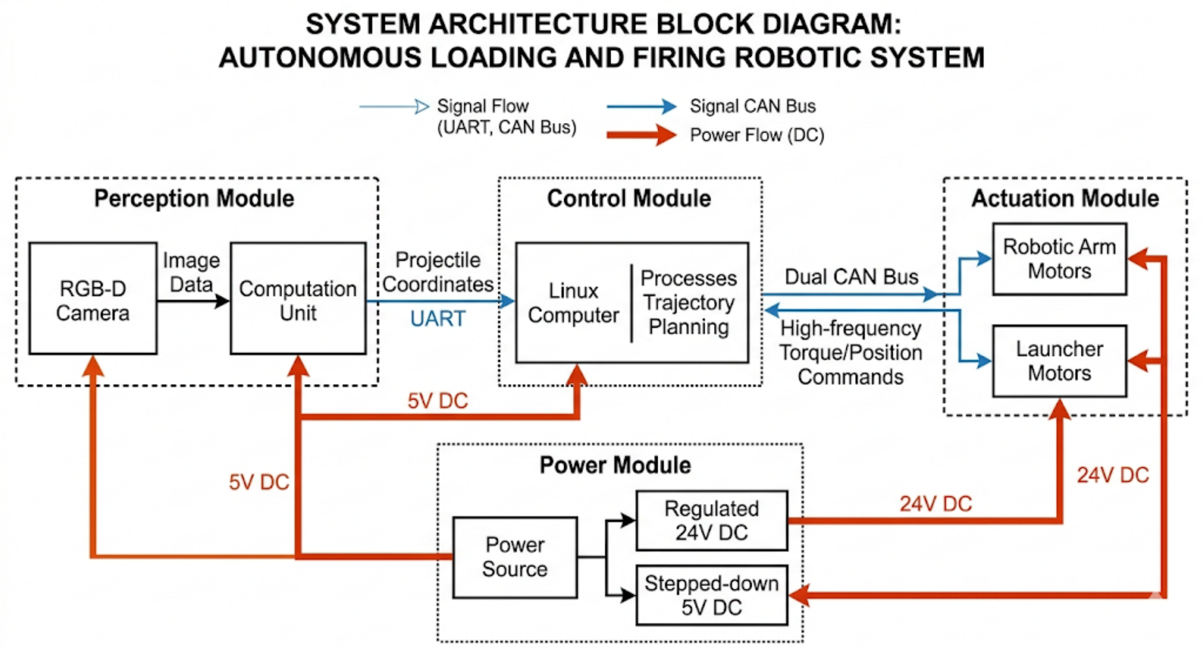

Figure 3: System Architecture Block Diagram: Visualizing the Perception, Control, Actuation, and Power subsystems and their detailed interconnections via UART, Dual CAN Bus, 24V DC, and 5V DC.

2.2 Perception and Decision Subsystem

This subsystem consists of the onboard camera(s) and the central compute unit running the overarching intelligent policy (e.g., YOLO-based neural network) to perceive the dart's location (bounding boxes and relative depths) and generate the corresponding target coordinates.

Requirements	Verification
1. The detection algorithm must identify darts within a 1.5 m unstructured staging radius under standard indoor lighting. 2. The vision processing latency must not exceed 50 ms per frame to ensure real-time tracking.	1. Randomly scatter 50 darts within the radius; verify that the software successfully outputs coordinates for ≥ 40 of them (80% accuracy). 2. Record the system timestamp before and after inference over 1000 continuous frames; verify the maximum (worst-case) latency $\Delta t \leq 50$ ms for every frame.

2.3 Control and Planning Subsystem

Translates the high-level perception outputs into physical actions. The Linux compute unit parses visual data into joint angles using inverse kinematics and generates smooth, collision-free trajectories. The STM32 microcontroller acts as the low-level execution core, receiving these trajectories to compute high-frequency PID loops and manage CAN bus communication protocols with the motor drivers.

Requirements	Verification
1. The CAN bus transmission delay between the STM32 controller and motor drivers must be ≤ 1 ms.	1. Connect a logic analyzer to the CAN TX/RX lines and measure the time delta between sent and acknowledged packets over 100 samples; verify delay ≤ 1 ms.

2.4 Actuation and Launch Subsystem

This subsystem comprises the physical hardware, including the joints of the RoboMaster-inspired arm (driven by brushless DC motors like the GM6020) for handling the dart, and the integrated friction-wheel launching mechanism responsible for launching the projectile.

Requirements	Verification
1. The base yaw joint of the robotic arm must be capable of rotating 120° and settling within 1 second.	1. Send a 120° step command via the controller and utilize motor encoder feedback to plot the response, confirming settling time ≤ 1 s.

2.5 Mechanical Design and CAD Assembly

To provide a comprehensive overview of the physical architecture, Figure 4 illustrates the complete mechanical CAD assembly of the autonomous loading and firing system. The design integrates the multi-axis robotic arm directly with the inclined launching track. This compact configuration minimizes the required operational envelope and ensures that the dart can be seamlessly transferred from the gripper to the loading port without singular configurations. The structural frame is constructed from extruded aluminum profiles to maintain rigidity under the dynamic recoil loads of launching, while custom brackets and motor mounts are designed for precision fabrication to ensure consistent alignment between the perception and actuation modules.

2.6 Power and Interface Subsystem

Includes the power management circuits that supply stable voltage (regulated 24V DC to the motors and stepped-down 5V DC to the microcontroller and sensors), along with the physical/remote initiation interfaces.

2.7 Tolerance Analysis

Since the system's primary objective prioritizes retrieval and loading reliability over ballistic targeting, a critical failure point is the positional misalignment of the robotic arm's end-effector during the grasping phase. If the gripper's center deviates from the projectile's center beyond the physical clearance margin,

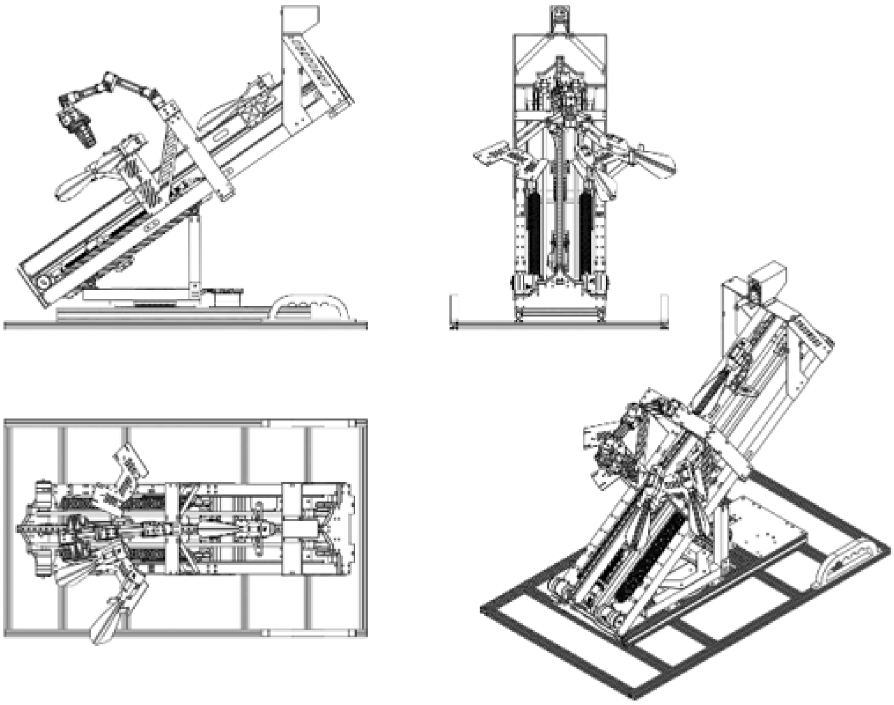

Figure 4: Complete mechanical CAD assembly from multiple viewing angles, detailing the integration of the robotic arm, friction-wheel launcher, and the underlying aluminum extrusion chassis.

Requirements	**Verification**
1. The 24V motor rail must remain within 24 ± 1.2 V under dynamic load, and the 5V logic rail must remain within 5.0 ± 0.25 V.	1. Apply representative peak load (simultaneous arm motion + launcher spin-up) and record rail voltages using an oscilloscope for 10 minutes; verify all samples remain within limits.
2. Emergency stop and user trigger interfaces must be deterministic: E-stop shall cut actuator power in ≤ 100 ms, and remote/manual start command latency shall be ≤ 200 ms.	2. Perform 30 repeated trials for each interface. For E-stop, measure delay from button press to motor current drop. For start command, measure delay from command issue to first actuator motion. Verify mean and worst-case latency constraints.

the arm will knock the ammunition away rather than securing it. The maximum allowable positional error (E_{max}) at the end-effector is dictated by the gripper's maximum opening width (w_g) and the projectile's body diameter (d_p):

$$E_{max} = \frac{w_g - d_p}{2} \tag{1}$$

Assuming a larger dart projectile with a body diameter of $d_p = 40$ mm and a gripper maximum opening of $w_g = 100$ mm, the allowable translational error is $E_{max} = 30 \text{ mm} = 0.030$ m. The most significant contributor to this end-effector error is the angular tolerance ($\Delta\theta$) of the base yaw motor. The translational deviation (Δs) at a maximum arm extension length (L) can be approximated by the arc length:

$$\Delta s = L \cdot \Delta\theta \tag{2}$$

Assuming a maximum operational reach of $L = 0.5$ m, maintaining $\Delta s \leq E_{max}$ requires the angular error to be bounded by:

$$\Delta\theta \leq \frac{0.030 \text{ m}}{0.5 \text{ m}} = 0.06 \text{ rad} \approx 3.44^\circ \tag{3}$$

While this simplified single-factor model isolates the base yaw motor, a comprehensive tolerance allocation must also account for multi-joint error propagation, gripper compliance, and camera calibration offsets. To guarantee the $\geq 80\%$ autonomous retrieval success rate specified in our High-Level Requirements while accommodating these additional error sources, the closed-loop PID control system must utilize high-resolution encoder feedback to strictly constrain the yaw joint's mechanical angular tolerance to a conservative $\pm 1.5^\circ$.

3 Cost

Table 1 gives a cost-controlled bill of materials (BOM) in RMB for one prototype. To keep the budget under 1500 CNY, existing lab assets (RGB-D camera and vision compute unit) are reused and only incremental hardware purchases are included.

Item	**Estimated Cost (RMB)**
STM32 controller board and CAN transceiver modules	120
Arm/launcher motors and basic drivers (used or student discount)	380
24V battery pack and power protection components	220
DC-DC converters (24V to 5V) and interface power distribution	80
Mechanical structure, gripper parts, bearings, and fasteners	220
3D-printed safety shroud and launcher fixtures	110
Sensors, switches, and emergency stop hardware	90
Cables, connectors, and wiring consumables	70
Contingency (15%)	180
Total	**1470**

Table 1: Cost-controlled prototype budget (RMB), capped below 1500 CNY.

4 Schedule

Table 2 summarizes a 12-week development timeline for design, integration, and validation, with explicit task delegations for all team members.

5 Ethics and Safety

5.1 Ethics

As this system involves autonomous ammunition loading and launching, it is imperative to adhere to the IEEE Ethically Aligned Design standards [3]. We commit to prioritizing human well-being by ensuring our system is strictly constrained to educational and competitive robotics contexts. The perception algorithm will be explicitly trained to detect inanimate targets (e.g., armor plates, standardized darts) and will feature hard-coded logic to prevent the targeting or tracking of human profiles, avoiding any dual-use applications that could promote violence.

5.2 Safety

The physical safety of operators and bystanders is our primary concern, and our design methodology aligns with the risk assessment and reduction principles outlined in ISO 12100 [4]. The friction-wheel launcher will be enclosed within a custom 3D-printed shroud to prevent accidental finger contact with high-speed rotating components. Furthermore, software-level safety limits will be programmed into the STM32 controller to cap the maximum RPM of the firing wheels, ensuring the kinetic energy of the launched darts remains strictly within the safe limits prescribed by RoboMaster competition regulations.

To ensure immediate hazard mitigation, an easily accessible hardware emergency stop (E-stop) circuit is implemented in strict compliance with ISO 13850 standards [5]. This E-stop mechanism physically

Week	Milestone and Deliverables / Task Assignments
Weeks 1–2	**Yidong & Xinchen:** Define system architecture, subsystem control interfaces, and kinematic constraints. **Yuxuan & Xiaoman:** Formalize vision perception requirements, perform risk assessment, and evaluate mechanical gripper concepts.
Weeks 3–4	**Yidong & Xiaoman:** Finalize mechanical CAD, fabricate gripper/launcher prototypes, and assemble hardware structure. **Xinchen & Yuxuan:** Conduct bench-level CAN motor tests and set up the overarching compute unit environment.
Weeks 5–6	**Yuxuan & Xinchen:** Implement vision pipeline (dataset collection, model selection) and develop STM32 bootstrapping. **Yidong & Xiaoman:** Set up physical camera positioning, optimize mounting, and complete robotic arm mechanical assembly.
Weeks 7–8	**Yidong & Xinchen:** Solve inverse kinematics on the Linux unit and integrate low-level CAN bus PID control on the STM32 firmware. **Yuxuan & Xiaoman:** Optimize vision processing latency, align coordinate systems (camera-to-arm), and verify physical dynamic clearances.
Weeks 9–10	**Yidong & Xinchen:** Tune PID controllers and integrate state-machine logic for autonomous pickup-loading trials. **Yuxuan & Xiaoman:** Link vision UART outputs to control inputs, assist in troubleshooting, and coordinate full closed-loop testing.
Week 11	**Yidong & Xiaoman:** Conduct 100-cycle stress test, inspect mechanical wear, and perform hardware failure analysis. **Xinchen & Yuxuan:** Monitor software stability/latency, and execute safety verification (E-stop deterministic latency checks).
Week 12	**Yidong & Xinchen:** Final dynamic hardware tuning, prepare system for demonstration, and write control/circuit documentation. **Yuxuan & Xiaoman:** Finalize perception and mechanical CAD documentation, format the final report, and prepare presentation materials.

Table 2: Planned 12-week execution schedule, detailing milestone progressions and specific member responsibilities to ensure continuous parallel development.

interrupts the 24V power line to all actuators instantly without relying on software logic, ensuring a fail-safe state. Additionally, the 24V power subsystem utilizes battery management and protective circuitry conforming to IEC 62133 safety standards [6] to prevent over-current, short-circuit, and thermal runaway conditions during high-load operations.

References

[1] J. Qin and K. Xu, "Design and implementation of automatic assisted aiming system for robomaster ep based on yolov5," *arXiv preprint arXiv:2312.05055*, 2023.

[2] Z. Lin, T. Wang, Q. Gao, and Y. Liu, "Design of robot platform based on can bus," in *2011 International Conference on Electrical and Control Engineering.* IEEE, 2011, pp. 645–648.

[3] IEEE Global Initiative on Ethics of Autonomous and Intelligent Systems, *Ethically Aligned Design: A Vision for Prioritizing Human Well-being with Autonomous and Intelligent Systems.* IEEE, 2019.

[4] "Safety of machinery — general principles for design — risk assessment and risk reduction," International Organization for Standardization, Geneva, CH, Standard, Nov. 2010.

[5] "Safety of machinery — emergency stop function — principles for design," International Organization for Standardization, Geneva, CH, Standard, Dec. 2015.

[6] "Secondary cells and batteries containing alkaline or other non-acid electrolytes - safety requirements for portable sealed secondary cells, and for batteries made from them, for use in portable applications - part 2: Lithium systems," International Electrotechnical Commission, Geneva, CH, Standard, Feb. 2017.