ECE 445

Senior Design Laboratory

Final Report

A World-Model based Infant Interaction Robot

Team #44

Zishuo Feng (zishuo.21)
Artha Amanda Sidauruk
(arthaamanda.22) Sylvia Chung
Yan Shan (sylviachung.22) Ruijin
Xu (ruijin.22)

May 17, 2026

Abstract

This paper presents the design and implementation of A World-Model based Infant Interaction Robot, a small differential-drive vehicle developed to engage infants in playful chasing interactions while ensuring physical safety. The robot employs two forward-facing infrared proximity detectors to sense an approaching infant. Unlike purely reactive systems that initiate a simple reverse motion upon detection, our approach integrates a world model—an internal probabilistic representation that estimates the infant's relative position and motion dynamics. By maintaining and continuously updating this model through sensor fusion, the robot is able to predict the infant's near-future trajectory and make anticipatory avoidance decisions. A model-based controller computes smooth, directional escape maneuvers that guide the robot away from the infant toward open space, thereby sustaining a safe and enticing distance. This predictive strategy produces more natural, less jerky behavior compared to reactive thresholds, which is crucial for keeping the infant engaged without causing frustration or collision risk. The hardware prototype was built on a low-cost two-wheeled chassis, and its real-time performance was evaluated in a series of indoor play scenarios. Experimental results demonstrate that the robot reliably detects approaching infants, correctly estimates their motion intent within its world model, and executes fluid avoidance actions that successfully maintain a safe interactive perimeter. The proposed system illustrates the benefit of incorporating an internal world model into infant-directed mobile robots, paving the way for more intelligent and adaptive child-robot playmates.

Contents

1 Introduction

Infancy is a critical period for motor development, cognitive exploration, and the emergence of social engagement. Activities such as crawling, chasing, and interactive play not only strengthen physical coordination but also stimulate curiosity and early causal reasoning. A mobile robot that can move autonomously and "play" with an infant has the potential to serve as a developmental companion. However, introducing a motorized platform into an infant's play space presents a fundamental challenge: ensuring absolute safety while maintaining the playfulness and continuity of the interaction. Simple remote-controlled toys lack autonomous interactivity, while purely reactive obstacle-avoidance robots often disrupt the flow of play with abrupt, jerky motions. To address these issues, this project designs and implements a multimodal infant interaction robot vehicle that integrates motion control, human sensing, voice interaction, and world-model-based intelligent behavior, thereby providing a safe, smooth, and engaging chase-and-play experience for infants.

1.1 Project Purpose

The primary objective of this project is to develop a mobile interactive robot platform specifically designed for infant-age children. The design goals can be decomposed into three layers. At the safety layer, the robot must actively avoid any collision with the infant under all circumstances, eliminating the risk of physical contact or startling caused by sudden maneuvers. At the interaction layer, the robot should perceive the infant's approach and movement tendencies and respond with natural, predictable motions that encourage the infant to chase, thereby prolonging effective playtime and stimulating gross motor activities such as crawling and turning. At the developmental layer, the robot incorporates a lightweight voice chat function, enabling it to provide auditory stimulation and vocal responses beyond physical movement, laying a foundation for future integration of language initiation and social interaction. Ultimately, through this prototype, we aim to demonstrate that a safe and entertaining infant-robot interaction paradigm can be realized under low-cost, resource-constrained hardware conditions.

1.2 Project Functionality

To achieve the above objectives, the robot vehicle integrates four core functions. These functions jointly support infant-directed interactive behaviors from the perspectives of mobility, perceptual input, interaction channels, and decision-making strategies.

1.2.1 Mobile Motion Control

The robot employs a two-wheel differential-drive chassis, possessing omnidirectional maneuverability including forward and backward movement, in-place turning, and smooth curved trajectories. The low-level motion controller precisely regulates the left and right wheel speeds, ensuring the robot can respond to high-level commands with smooth acceleration and deceleration, avoiding sudden start-stop jerks. This function serves as the foundation for all mobile interactions, providing motion execution support for actions such as approaching, retreating, or circumventing.

1.2.2 PIR-based Human Motion Detection

The robot is equipped with a passive infrared (PIR) sensor to detect changes in infrared radiation emitted by human bodies in the environment. The PIR sensor is highly sensitive to moving human subjects and can quickly trigger a "person approaching" perception event when an infant enters the robot's activity area. Unlike pure distance measurement, the PIR signal directly reflects the presence and movement magnitude of a dynamic human body, effectively filtering out interference from static obstacles. It provides a reliable infant activity detection wake-up signal for the robot and assists in judging the infant's activity level.

1.2.3 Voice Chatbot Interaction

To enrich the interaction with playfulness and multimodal stimulation, the robot integrates a lightweight voice dialogue module. This module can play preset child-friendly spoken phrases, imitate animal sounds, or sing simple nursery rhymes. Under specific conditions—such as the infant maintaining a safe distance for a while or after a successful dodge—it can proactively issue vocal invitations to encourage the infant to continue chasing. The voice interaction also endows the robot with a degree of anthropomorphic character, helping to attract the infant's attention and establish an emotional connection, thereby elevating the purely physical chase into a more socially engaging game.

1.2.4 Safe State-based / World-Model Behavior

This constitutes the core of the robot's decision-making. The system internally maintains a dynamic estimation of the infant's relative position and motion state—referred to as the world model. By fusing PIR trigger patterns and, where available, auxiliary infrared distance sensors, the robot not only knows "how close" the infant is, but also infers "which direction" the infant is coming from and "at what speed" the infant is approaching. On this basis, a safety state machine switches behavior modes according to the estimated collision risk level. Outside the warning zone, the robot can wander moderately or issue voice invitations. When the infant rapidly approaches and enters the alert zone, the robot activates predictive avoidance based on the world model, selecting a smooth escape path that heads towards open space and away from the infant's predicted trajectory. This anticipatory, state-driven behavior pattern avoids the abrupt backward motions and deadlock situations commonly seen in purely reactive systems, making the robot's evasive actions natural and coherent, continuously maintaining a safe interactive distance so that the game can proceed without interruption.

1.3 Subsystem Overview

Include a top-level block diagram showing: Infant / Human Motion → PIR Sensors → ESP32 Robot Controller → Motor Driver → Motors and User Voice → Microphone → XiaoZhi AI ESP32 → Cloud AI Platform → Speaker / OLED Response.

2 Design

2.1 Overall System Architecture

The Infant Interaction Robot is designed as a modular embedded platform, integrating two high-level subsystems: motion control and interaction. The architecture is centered around the **ESP32 microcontroller**, which orchestrates all sensory inputs, actuator commands, and AI processing in a coordinated framework.

- **Motion Subsystem:** Implements differential drive control of two DC motors using a TB6612FNG motor driver. Motion decisions are derived from reactive input signals captured by HC-SR505 PIR sensors, providing collision avoidance and basic path adaptation.
- **Interaction Subsystem:** Hosts a personalized AI chatbot designed for infant engagement. The module processes voice input from an I2S microphone, generates context-aware audio responses through a speaker, and optionally provides visual feedback via LEDs. The AI behavior was customized to adapt to infant-directed interaction, supporting multi-turn dialogue and expressive feedback.

The ESP32 executes periodic task loops at distinct frequencies for each subsystem (motion: 50 Hz; interaction: 20 Hz; safety monitoring: 40 Hz), ensuring smooth coordination between sensor inputs, actuator outputs, and AI responses.

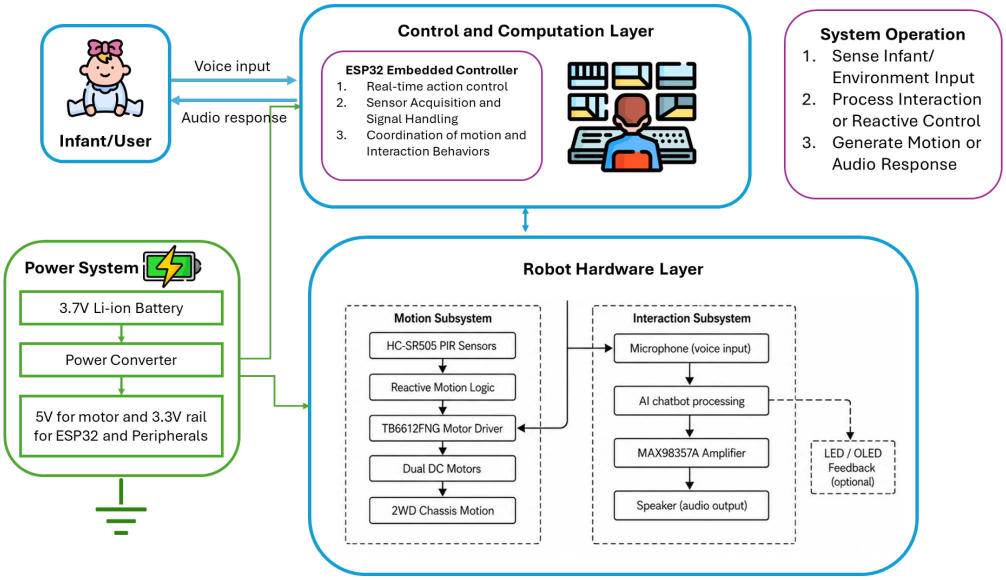

Figure 1: Block Diagram

2.2 Control System

The control system leverages the ESP32's real-time processing capabilities:

- Configurable PWM channels generate precise motor control signals.

- GPIO and timer-based loops synchronize interaction tasks and motion control.
- Reactive algorithms enable real-time response to sensor events.
- All control logic is encapsulated in modular firmware tasks, allowing extensibility for future AI or sensor integration.

2.3 Power and Electronics

The system is powered by dual 3.7V Li-ion batteries supplemented by a regulated 5V output for motor and amplifier operation. Key power considerations include:

- ESP32 operates at 3.3V, while motors and audio amplifiers receive 5V.
- Peripheral devices such as microphones and displays draw regulated 3.3V from the ESP32.
- A common ground architecture ensures stable operation across motors, sensors, and interaction devices.

2.4 Robot Hardware Overview

Motion Subsystem:

- Two DC motors driven by TB6612FNG motor driver.
- PIR sensors for reactive motion and obstacle detection.
- Differential drive chassis enabling basic locomotion.

Interaction Subsystem:

- Voice input captured via I2S microphone.
- AI chatbot firmware flashed onto ESP32, with infant-specific response customization.
- Audio output delivered through a small speaker; optional visual cues provided via LEDs.

2.4.1 Pinout Assignment

2.5 Interaction Module Overview

The interaction module provides an abstracted AI-driven engagement capability for infants:

- The AI chatbot firmware is pre-flashed onto the ESP32 and customized for infant-directed interaction.
- Audio responses are generated in real-time via a speaker, while the microphone captures infant vocalizations and commands.

Device / Function	**ESP32 Pin**	**Notes**
Status LED	2	Onboard indicator
Left Motor PWM	25	TB6612FNG PWMA
Right Motor PWM	26	TB6612FNG PWMB
Left Motor IN1	27	TB6612FNG AIN1
Left Motor IN2	14	TB6612FNG AIN2
Right Motor IN1	32	TB6612FNG BIN1
Right Motor IN2	33	TB6612FNG BIN2
Motor Standby	13	TB6612FNG STBY
PIR Sensor Left	34	Digital input
PIR Sensor Right	35	Digital input
I2C SDA	21	Optional
I2C SCL	22	Optional
Buzzer	-1	Disabled / optional
Servo	-1	Disabled / optional

Table 1: ESP32 Pin Assignment for Infant Interaction Robot

- The module is designed to be modular and extensible, allowing future integration of additional sensors, actuators, or cloud-assisted AI capabilities.

2.6 System Integration

The ESP32 integrates the motion and interaction subsystems, coordinating sensor input, actuator commands, and AI tasks. PWM channels control motor speed and direction based on PIR sensor feedback, while interaction tasks operate concurrently in scheduled loops. This design ensures low-latency responses, reliable motion control, and consistent infant engagement, forming a cohesive and modular embedded robotic platform.

3 Requirements and Verification

3.1 Verification Strategy

The verification process was organized around the robot's main functional blocks: motion, sensing, AI interaction, power, and full-system integration. Each block was tested individually before being tested again as part of the complete robot platform.

Verification focused on confirming both independent subsystem operation and safe integration of motion and audio interaction. The motor-control ESP32 was verified using serial commands, PIR sensor readings, motor-driver response, and status feedback. The chatbot ESP32 was verified using the XiaoZhi AI workflow, including firmware flashing, Wi-Fi setup, device pairing, personality configuration, voice input, and audio output through the INMP441 microphone and MAX98357A amplifier/speaker path. ESP Web Tools logs were also used to confirm firmware boot, Wi-Fi connection, cloud/MQTT activation, audio initialization, and idle-state readiness, as well as to debug temporary brownout, OLED I2C, and display-task issues.

Verification was divided into four levels:

1. Component-level verification: Each sensor, motor driver, motor, microphone, amplifier, speaker, OLED, and power rail was checked individually.
2. Firmware-level verification: The ESP32 motor-control firmware was tested using serial commands such as F, B, L, R, S, REACTIVE_ON, REACTIVE_OFF, ESTOP, RESET_ESTOP, and STATUS?.
3. Subsystem-level verification: The motion subsystem and interaction subsystem were tested separately to confirm reliable operation before full integration.
4. System-level verification: The complete robot was tested as an infant-interaction robot that can detect motion, react with safe movement, and provide voice-based interaction.

Since the final design uses HC-SR505 PIR sensors, the sensing requirement was defined as motion detection rather than distance measurement, because the sensors only provide digital motion/no-motion outputs.

3.2 Functional Requirements

Table 2 summarizes the final functional requirements and verification procedures for the completed robot.

Table 2: Requirements and Verification Summary

Requirement	**Verification Procedure**	**Expected / Measured Result**	**Status**
The robot shall power on safely from the battery power system and distribute power to the ESP32 controllers and peripherals.	Connect the 2S Li-ion battery pack through the power switch and regulation/distribution circuit. Measure that the robot powers on without reset loops or overheating.	The robot powers on successfully, the ESP32 boards boot, and all connected peripherals receive stable power.	Pass
The motion-control ESP32 shall initialize successfully and report readiness after boot.	Upload the firmware to the ESP32 DevKit V1 / ESP32-WROOM-32 and monitor the serial terminal at 115200 baud.	The firmware prints the ready message and enters the idle state after startup.	Pass
The robot shall support basic differential-drive motion commands.	Send serial commands F, B, L, R, and S through the serial monitor. Observe motor direction and stopping behavior.	The robot moves forward, backward, turns left, turns right, and stops correctly.	Pass
The TB6612FNG motor driver shall correctly control two DC motors using PWM and direction signals.	Verify motor-driver wiring and send motion commands while observing both motors. Confirm that PWM speed changes and direction pins respond correctly.	Both motors respond correctly to the ESP32 control signals, and the robot produces 2WD chassis motion.	Pass
The robot shall read two HC-SR505 PIR sensors for human/infant motion detection.	Place a moving person or hand in front of the left and right PIR sensors separately and monitor serial debug/status output.	The left and right PIR sensors report motion/no-motion correctly.	Pass

The robot shall perform reactive motion when PIR motion is detected and reactive mode is enabled.	Enable reactive mode using REACTIVE_ON, trigger the left PIR, right PIR, and both PIR sensors, then observe robot behavior.	Motion on the left causes the robot to turn away to the right; motion on the right causes the robot to turn away to the left; motion on both sides causes the robot to back up straight.	Pass
The AI interaction module shall accept voice input through the microphone.	Flash and configure the XiaoZhi AI voice firmware, activate listening mode, and speak test commands or greetings.	The AI module detects voice input and begins processing the spoken interaction.	Pass
The AI interaction module shall provide an audio response through the amplifier and speaker.	Ask the chatbot a simple question, observe the speaker output, and monitor serial logs for audio codec/audio service initialization.	The system initialized the audio service and produced an audible AI response through the MAX98357A amplifier and speaker.	Pass
The AI chatbot shall connect to Wi-Fi and pair with the XiaoZhi cloud console.	Configure 2.4 GHz Wi-Fi through the temporary XiaoZhi hotspot, read the OLED pairing code, add the device in the XiaoZhi console, and monitor ESP Web Tools serial logs for Wi-Fi connection, IP assignment, cloud/MQTT activation, and idle-state transition.	The device connected to Wi-Fi, obtained an IP address, completed activation, entered the idle state, and appeared in the XiaoZhi console.	Pass
The robot shall support infant-oriented interaction behavior through the AI personality configuration.	Configure the XiaoZhi role introduction/personality to use safe, simple, friendly, infant-oriented dialogue. Test several short interactions.	The chatbot responds with short, friendly, age-appropriate dialogue behavior.	Pass

The complete system shall operate with a shared ground between power, control, sensors, actuators, and audio devices.	Verify continuity between subsystem grounds and test the full robot under integrated operation.	After power and wiring corrections, the motion and interaction subsystems operated together without persistent reset loops or unstable sensor readings.	Pass

3.3 Experimental Results

The final prototype satisfied all required functions. The motion-control ESP32 booted correctly, produced the expected ready message, and reliably responded to basic serial commands for forward, backward, left turn, right turn, and stop. The TB6612FNG motor driver correctly converted the ESP32 PWM and GPIO signals into differential-drive motor motion.

The two HC-SR505 PIR sensors were verified as digital motion sensors. Left-side detection caused the robot to turn away from the left, right-side detection caused it to turn away from the right, and detection from both sensors caused the robot to back up straight. This matched the intended safe reactive motion behavior.

The safety functions also passed verification. When command communication timed out, the robot stopped and entered the communication-lost state. The emergency stop command immediately stopped the robot, and the reset command returned the system to normal operation.

The AI interaction subsystem was verified after flashing and configuring the XiaoZhi firmware. Serial logs confirmed Wi-Fi connection, IP assignment, firmware version checking, MQTT connection, activation completion, idle-state transition, and audio service startup. After setup, the microphone accepted voice input, and the MAX98357A amplifier and speaker produced audible responses, confirming successful voice-based interaction.

Table 3: Experimental Results

Test Case	**Result**	**Status**
ESP32 motor firmware boot test	Firmware booted and entered idle state correctly.	Pass
Forward motion test	Robot moved forward when F command was sent.	Pass
Backward motion test	Robot moved backward when B command was sent.	Pass

Left/right turning test	Robot turned left and right using L and R commands.	Pass
Stop command test	Robot stopped when S command was sent.	Pass
PIR left detection test	Left PIR detection was reported correctly.	Pass
PIR right detection test	Right PIR detection was reported correctly.	Pass
Reactive behavior test	Robot turned or backed away according to PIR detection pattern.	Pass
Status feedback test	STATUS? returned state, motion, PIR, reactive, communication, and E-stop information.	Pass
Communication timeout test	Robot entered safe stop after command loss.	Pass
XiaoZhi Wi-Fi setup test	Chatbot ESP32 connected to the configured 2.4 GHz Wi-Fi network.	Pass
AI chatbot serial-log debugging	ESP Web Tools logs confirmed firmware boot, Wi-Fi configuration, Wi-Fi connection, cloud/MQTT activation, idle-state transition, and audio startup. Earlier logs also helped identify temporary brownout, OLED I2C, and display-task issues during debugging.	Pass
XiaoZhi pairing test	Device paired successfully using the OLED pairing code.	Pass
Voice input test	Microphone captured spoken input.	Pass
Audio output test	Speaker produced chatbot response audio.	Pass
Full system integration test	Motion subsystem, sensing subsystem, AI interaction subsystem, and power system worked together.	Pass

Overall, the completed prototype met the required functionality by integrating two major behaviors: reactive physical motion through embedded sensing and motor control, and voice-based AI interaction through the XiaoZhi ESP32 chatbot system.

3.4 Failed Requirements and Analysis

No functional requirements failed in the final verification stage. During development, the AI chatbot logs showed temporary issues, including brownout resets during Wi-Fi startup, OLED I2C/display communication errors, and display/LVGL task watchdog errors. These were treated as debugging findings, not final failures, because the corrected chatbot subsystem successfully connected to Wi-Fi, completed cloud/MQTT activation, entered the idle state, and produced audio output.

However, several limitations remain:

1. The PIR sensors do not measure distance.

   The HC-SR505 sensors only detect motion, so the robot cannot determine the exact distance to the infant/user. This is acceptable because the final sensing requirement was motion-based interaction, not distance-based navigation.

2. The system is a proof-of-concept, not a certified infant-care device.

   The robot demonstrates safe low-level behavior such as stopping, backing away, and emergency-stop control, but real infant-care deployment would require enclosure improvements, long-duration testing, and safety certification.

In conclusion, the final prototype passed all defined functional requirements. The remaining issues are practical limitations and future improvement opportunities, not failed requirements.

4 Cost and Schedule

4.1 Cost Analysis

Table 4: Cost Analysis

No.	Item	Quantity / Description	Cost (RMB)
1	Arduino UNO R3 7.4V battery holder	1 piece	5.50
2	MB-102 breadboard power supply module	3 pieces	16.92
3	0.96-inch OLED display module	2 orders	21.73
4	HC-SR501 PIR motion sensor module	About 6 pieces	36.21
5	TB6612FNG / DRV8833 motor driver module	Multiple orders	34.46
6	GP2Y0A21YK infrared distance sensor	2 pieces	52.00
7	GP2Y0A02YK0F infrared distance sensor	1 piece	30.25
8	ESP32 development board, 30-pin CP2102 Type-C	About 4 pieces	78.10
9	ESP32-CAM development board	1 piece	51.50
10	ESP32-C3 SuperMini development board	1 piece	12.06
11	Infrared obstacle avoidance / tracking sensor module	2 pieces	10.76
12	Dupont jumper wires	1 set	2.87
13	Active buzzer module	High-level and low-level trigger versions	5.64
14	Breadboard / MB-102 breadboard	1 order	15.56
15	18650 lithium batteries with charger	1 set	33.80
16	Smart car chassis kit	1 set	17.38

17	INMP441 omnidirectional microphone module	1 piece	16.65
18	MAX98357 I2S amplifier module	1 piece	8.49
19	Type-C data cable	1 piece	3.87
20	Smart car shell design	1 design	39.00
21	First PCB design and fabrication	About RMB 170	170.00
22	Second PCB design and fabrication	About RMB 170	170.00

The estimated engineering labor cost is calculated using the standard formula: labor cost = hourly rate × working hours × 2.5. Assuming four team members, each contributing 120 hours at an engineering rate of \$50/h, the labor cost for each member is \$15,000. Therefore, the total estimated engineering labor cost is \$60,000.

4.2 Project Schedule

Table 5: Project Schedule

Stage	**Main Tasks**	**Outcome**
System planning and requirement definition	Defined the robot's main functions, including motion control, infant motion detection, voice interaction, and safe behavior	Overall system architecture and project requirements were established
Component selection	Selected ESP32 boards, PIR sensors, motor drivers, motors, OLED display, microphone, amplifier, speaker, battery components, and chassis	Major hardware components were purchased
Initial hardware assembly	Connected the ESP32, TB6612FNG motor driver, DC motors, PIR sensors, and power system on a prototype platform	Basic motion-control hardware was completed

Firmware development	Developed and tested the motor-control firmware, including movement commands, PWM motor control, PIR input reading, reactive motion, status feedback, and emergency-stop behavior	Motion subsystem became functional
AI interaction setup	Flashed and configured the XiaoZhi AI firmware, connected the ESP32 to Wi-Fi, paired the device with the cloud console, and tested microphone and speaker output	Voice-based chatbot interaction was verified
PCB design and revision	Designed the first PCB version, tested the layout, identified issues, and produced a second version with improvements	Two PCB versions were completed
Mechanical integration	Installed the electronics, sensors, battery, chassis, and shell structure into the smart car platform	The robot became a complete physical prototype
System integration	Combined motion control, sensing, power, and AI interaction subsystems	The complete robot operated as an integrated infant-interaction platform
Verification and debugging	Tested motor motion, PIR detection, reactive avoidance, voice input, audio output, Wi-Fi connection, power stability, and full-system operation	The final prototype passed the defined functional requirements
Final documentation	Prepared the final report, cost analysis, verification results, conclusion, ethics, and safety discussion	Final project report was completed

5 Conclusion

5.1 Project Accomplishments

This project successfully developed a prototype infant interaction robot based on a small differential-drive smart car platform. The final system integrated motion control, human motion sensing, voice-based AI interaction, power distribution, and basic safe reactive behavior into a single embedded robotic platform.

For the motion subsystem, the ESP32 controller was able to control the two DC motors through the TB6612FNG motor driver. The robot successfully supported basic motion commands, including forward movement, backward movement, left turning, right turning, and stopping. PWM-based control allowed the robot to produce stable differential-drive motion.

For the sensing subsystem, the robot used PIR motion sensors to detect human or infant movement. When motion was detected on the left side, the robot turned away to the right. When motion was detected on the right side, the robot turned away to the left. When both sides detected motion, the robot moved backward. This behavior allowed the robot to respond to an approaching user and reduce the chance of direct collision.

For the interaction subsystem, the robot integrated a XiaoZhi AI-based voice interaction module. The system accepted voice input through the INMP441 microphone and produced audio responses through the MAX98357 I2S amplifier and speaker. This allowed the robot to provide simple voice-based interaction in addition to physical motion.

The final prototype demonstrated the feasibility of combining embedded motion control and AI voice interaction in a low-cost infant-directed robotic platform. Although the current design is still a proof-of-concept prototype, it achieved the main project goal of creating a mobile robot that can sense human motion, react safely, and provide interactive feedback.

5.2 Project Uncertainties and Limitations

Although the final prototype satisfied the main functional requirements, several limitations remain.

First, the PIR sensors used in the final system can only detect motion. They cannot measure the exact distance, direction, or speed of the infant. Therefore, the robot's perception ability is limited compared with a system using distance sensors, cameras, or depth sensors. The current system can determine whether motion is present on the left or right side, but it cannot build a precise spatial map of the surrounding environment.

Second, the AI voice interaction subsystem depends on Wi-Fi and cloud service availability. If the Wi-Fi connection is unstable or the cloud service is unavailable, the chatbot function may not operate correctly. This limits the robot's reliability in environments without stable network access.

Third, the robot is a senior design prototype rather than a certified infant-care product. Although the robot includes safety-oriented behaviors such as stopping, backing away, and turning away from detected motion, it has not undergone professional child-safety certification. In real infant-use scenarios, the robot would require a safer enclosure, softer external materials, stronger wiring protection, battery protection, long-duration reliability testing, and stricter motion-speed limits.

Finally, the current world-model behavior is limited by the available sensors and embedded computing resources. The robot demonstrates basic state-based and reactive behavior, but its prediction ability is still simple. A more advanced world model would require richer sensor input, better estimation algorithms, and more powerful onboard computation.

5.3 Future Work and Alternatives

Several improvements can be made in future versions of the robot.

First, the sensing system can be upgraded by adding more accurate distance sensors, such as time-of-flight sensors, ultrasonic sensors, or infrared distance sensors. These sensors would allow the robot to estimate the infant's distance more accurately and make smoother avoidance decisions.

Second, a camera or depth sensor could be added to improve environmental perception. With visual input, the robot could better identify the infant's position, motion direction, and surrounding obstacles. However, camera-based sensing would also increase privacy concerns and computational requirements.

Third, the AI interaction subsystem could be improved by using a local AI model instead of relying on cloud services. A local model would reduce network dependence and improve privacy, although it would require a more powerful embedded computing platform.

Fourth, the mechanical structure could be improved. A future version should use a safer and more durable enclosure with rounded edges, covered wires, protected batteries, and soft external materials. The shell should also be designed to prevent infants from touching internal electronics or small detachable parts.

Fifth, the power system could be upgraded with better battery management, charging protection, and voltage monitoring. This would improve system reliability and reduce the risk of brownout resets or unsafe battery operation.

Finally, the world-model algorithm could be strengthened. Future work could include sensor fusion, motion prediction, path planning, and adaptive behavior selection. These improvements would allow the robot to move more naturally and maintain a safer interactive distance from the infant.

6 Ethic and Safety

6.1 Ethical Considerations

Because this project is designed for infant interaction, ethical considerations are especially important. Infants are a vulnerable user group and cannot fully understand or control the robot's behavior. Therefore, the system must prioritize safety, transparency, privacy, and responsible use.

The robot should not be treated as a replacement for human supervision or caregiving. It is only intended to demonstrate an interactive play concept under controlled conditions. Any real use with infants should occur only under adult supervision. The robot's purpose is to support playful interaction and motor engagement, not to make independent caregiving decisions.

Privacy is another important ethical concern. The voice interaction module uses a microphone and cloud-based AI service. If voice data is transmitted to an online service, users should be informed about the data flow and potential privacy risks. For a future product version, local processing or stronger data-protection methods should be considered to reduce privacy concerns.

The robot's behavior should also be designed to avoid emotional or physical harm. It should not produce sudden, loud, frightening, or unpredictable actions. Its speech output should be simple, friendly, age-appropriate, and non-manipulative. The robot should encourage interaction without overstimulating or frustrating the infant.

6.2 Broader Impacts

This project demonstrates how low-cost embedded systems, sensing modules, and AI interaction can be combined to create a small interactive robot. Such systems may have positive broader impacts in early childhood play, education, and assistive interaction research. A safe and engaging mobile robot could encourage infants to crawl, turn, follow motion, and respond to simple audio feedback.

The project also shows the educational value of integrating multiple engineering domains, including embedded systems, motor control, PCB design, power electronics, sensing, audio processing, AI interaction, mechanical design, and system-level testing. This makes the project a useful example of interdisciplinary engineering design.

However, broader impacts also include potential risks. If similar systems are poorly designed, they could create safety hazards, privacy issues, or unrealistic expectations about robotic caregiving. Infant-directed robots should always be designed with conservative safety limits and should not be marketed as substitutes for human care. Future versions should follow relevant safety standards and include clear user instructions.

6.3 Safety Considerations

Safety was a central concern throughout the project because the robot is intended for infant-directed interaction. The robot uses low-speed movement and simple reactive behaviors to reduce the risk of collision. When PIR motion is detected, the robot turns away or backs up rather than moving toward the detected subject. The system also supports stop and emergency-stop behavior during testing.

Electrical safety was considered by separating voltage levels and using regulated power for the ESP32 controllers and peripheral modules. The ESP32 operates at 3.3 V, while the motors and audio amplifier use appropriate power rails. A shared ground was used to ensure stable communication and reduce unstable sensor readings. During testing, attention was given to avoiding overheating, unstable wiring, and reset loops.

Mechanical safety is also important. The current prototype should be used only in a controlled test environment. For real infant interaction, the robot would need a fully enclosed structure, rounded edges, protected wires, secured batteries, and no exposed small parts. The wheels and moving parts should also be shielded to prevent pinching or entanglement.

In addition, acoustic safety must be considered because the robot includes a speaker. The audio output should remain at a low volume to avoid startling or harming the infant. The voice responses should be calm, short, and friendly.

Overall, the current prototype demonstrates the basic safety-oriented behavior required for a senior design project, but it should not be considered a finished infant-care product. Further enclosure design, long-term testing, battery protection, child-safety certification, and adult-supervised evaluation would be required before any real-world infant use.

A Appendix