ROBOTIC HEAD CONNECTED TO CHAT GPT

A PROJECT REPORT

Submitted by

BEEMA DEWAN T D (310821106016)

DEECHANA J (310821106017)

INDHUMATHI V (310821106031)

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

In

ELECTRONICS AND COMMUNICATION ENGINEERING

JEPPIAAR ENGINEERING COLLEGE, CHENNAI.

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2025

ANNA UNIVERSITY: CHENNAI-600 025

BONAFIDE CERTIFICATE

Certified that this project report **"ROBOTIC HEAD CONNECTED TO CHAT GPT"** is a bonafide work of **BEEMA DEWAN T D (310821106016), DEECHANA J (310821106017), INDHUMATHI V (310821106031)** who carried out the project work under my supervision.

SIGNATURE	**SIGNATURE**
Dr. J. JEBASTINE, M.E, Ph.D.	Mrs. R. GRACELIN SHEEBA, M.E
HEAD OF THE DEPARTMENT	**SUPERVISOR**
	ASSISTANT PROFESSOR
DEPARTMENT OF ELECTRONICS	DEPARTMENT OF ELECTRONICS
AND COMMUNICATION	AND COMMUNICATION
ENGINEERING,	ENGINEERING,
JEPPIAAR ENGINEERING	JEPPIAAR ENGINEERING
COLLEGE,	COLLEGE,
CHENNAI – 600 119.	CHENNAI – 600 119.

CERTIFICATE OF EVALUATION

S.NO	NAME OF THE STUDENTS WHO HAVE DONE THE PROJECT	TITLE OF THE PROJECT	NAME OF THE SUPERVISOR WITH DESIGNATION
01. 02. 03.	BEEMA DEWAN T D (310821106016) DEECHANA J (310821106017) INDHUMATHI V (310821106031)	ROBOTIC HEAD CONNECTED TO CHAT GPT	Mrs. R. GRACELIN SHEEBA M.E. ASSISTANT PROFESSOR

The Reports of the project work submitted by the above students in partial fulfilment for the award of Bachelor of Engineering Degree in Electronics & Communication Engineering of Anna University, Chennai were confirmed to be reports of the work by the above students and then evaluated.

Submitted for the examination held on: ________________

INTERNAL EXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We are very truly indebted to **Late Col. Dr. Jeppiaar, M.A., B.L., Ph.D.,** Founder, Jeppiaar Educational Trust, **Dr. M. Regeena J Murali, B.Tech, M.B.A., Ph.D.,** our Chairman and Managing Director and our Principal **Dr. K. Senthil Kumar, M.E., Ph.D.,** for their blessings and the facilities to carry out the project.

We would like to express our gratitude to **Dr. J. Jebastine M.E., Ph.D.**, Professor and Head of the Department. We thank our project coordinator **Dr. J. Jebastine M.E., Ph.D.**, We are thankful to our supervisor **Mrs. R. Gracelin Sheeba M.E.,** for giving valuable suggestions and guiding us in all stages of the project. We finally thank all the faculty members of the **Department of Electronics and Communication Engineering** for their constant support.

We would also like to thank our parents and friends for all their support and we thank the Almighty God for helping us complete this project, successfully.

ABSTRACT

This project presents the development of an animatronic robotic head capable of exhibiting human-like facial expressions and engaging in real-time AI-powered conversations using ChatGPT. The primary goal is to enhance human-robot interaction by integrating natural language processing with physical expressiveness, allowing the robot to communicate in a more lifelike and emotionally intelligent manner. The robotic head is designed with multiple servo motors to simulate facial movements such as eye moment and jaw motion. A microphone captures the user's voice input, which is then converted to text using speech-to-text (STT) algorithms. This input is processed by ChatGPT to generate context-aware, coherent responses. The output is converted back into speech using text-to-speech (TTS) technology, synchronized with appropriate facial expressions based on the tone and sentiment of the conversation. The system is implemented using components such as Arduino, servo motors, and AI APIs, ensuring cost-effectiveness and scalability. This work demonstrates a step toward emotionally intelligent and accessible robotic systems that can interact with humans in a more natural and engaging way.

CHAPTER NO	TITLE	PAGE NO
	ABSTRACT	**v**
	LIST OF TABLES	**vi**
	LIST OF FIGURES	**ix**
	LIST OF ABBRIEVATION	**x**
1	**INTRODUCTION**	**1**
	1.1 BRIEF OVERVIEW OF THE PROJECT	**1**
	1.2 INTRODUCTION TO HUMAN ROBOT INTERACTION	**1**
	1.3 CONVERSATIONAL AI AND CHATGPT	**2**
	1.3.1 Overview of Conversational AI	**3**
	1.4 SPEECH RECOGNITION AND TEXT-TO-SPEECH CONVERSION	**3**
	1.5 ANIMATRONIC MECHANISM AND FACIAL MOVEMENTS CONTROL	**4**
2	**LITERATURE SURVEY**	**5**
	2.1 FACIAL ROBOTICS AND ANIMATRONICS	**5**
	2.2 SPEECH INTERACTION IN ROBOTICS	**5**
	2.3 CONVERSATIONAL AI AND REAL TIME INTEGRATION	**6**
3	**PROPOSED SYSTEM**	**8**
	3.1 DISADVANTAGES OF EXISTING SYSTEMS	**10**
	3.2 ADVANTAGES OF THE PROPOSED SYSTEM	**12**
	3.2.1 System Block Diagram	**14**
	3.2.2 Hardware Tools	**14**

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI – ARTIFICIAL INTELLIGENCE

API – APPLICATION PROGRAMMING INTERFACE

CAD – COMPUTER-AIDED DESIGN

GPIO – GENERAL PURPOSE INPUT/OUTPUT

GPT – GENERATIVE PRETRAINED TRANSFORMER

GUI – GRAPHICAL USER INTERFACE

HCI – HUMAN-COMPUTER INTERACTION

HMI – HUMAN-MACHINE INTERFACE

IOT – INTERNET OF THINGS

JSON – JAVASCRIPT OBJECT NOTATION

MCU – MICROCONTROLLER UNIT

ML – MACHINE LEARNING

NLP – NATURAL LANGUAGE PROCESSING

SDK – SOFTWARE DEVELOPMENT KIT

STT – SPEECH-TO-TEXT

TTS – TEXT-TO-SPEECH

UART – UNIVERSAL ASYNCHRONOUS RECEIVER-TRANSMITTER

CHAPTER-1

INTRODUCTION

1.1 BRIEF OVERVIEW OF THE PROJECT

This project presents the design and implementation of an animatronic robotic head that combines basic facial expressions with real time AI-based conversational capabilities. The robot is capable of mimicking human like interaction by integrating jaw and eye movements enabled through servo motors with intelligent dialogue generation powered by OpenAI’s ChatGPT. The structure is built using 3D-printed plastic for lightweight, customizable facial features, and is controlled by an Arduino microcontroller. This fusion of mechanical expression and conversational AI provides a new direction for enhancing human-robot interaction in fields such as education, customer service, and therapy.

1.2 INTRODUCTION TO HUMAN-ROBOT INTERACTION:

Human-Robot Interaction (HRI) is a multidisciplinary research domain that focuses on the communication and collaboration between humans and robotic systems. As robots become increasingly integrated into everyday environments ranging from homes and hospitals to educational institutions the ability of these systems to engage with humans in a natural and meaningful way becomes critically important.

Traditional robotic systems were primarily designed for industrial automation and followed rigid task specific programming. These systems lacked social presence and emotional responsiveness, which are key elements for user engagement. Recent developments in artificial intelligence, natural language processing, and actuated robotic mechanisms have enabled a new generation of robots capable of interacting socially, adapting to user needs, and simulating human-like behaviours.

1.3 CONVERSATIONAL AI AND CHATGPT:

- **OVERVIEW OF CONVERSATIONAL AI:**

Conversational AI refers to the technology that enables machines to understand, process, and respond to human language in a natural and meaningful way. It encompasses natural language processing (NLP), automatic speech recognition (ASR), dialogue management, and text-to-speech (TTS) systems. Over the last decade, conversational AI has evolved from rule-based chatbots to advanced generative models capable of understanding context and generating human-like responses.

This evolution has made it feasible for robots and embedded systems to engage in realistic, intelligent dialogue with users. The integration of conversational AI into physical systems, such as animatronics, allows for human-robot interaction that is both expressive and semantically rich.

- **CHATGPT: A GENERATIVE LANGUAGE MODEL**

ChatGPT, developed by OpenAI, is based on the Generative Pre-trained Transformer (GPT) architecture, a deep learning model trained on extensive textual data. Unlike traditional chatbots, ChatGPT does not rely on pre-programmed responses but instead generates replies dynamically, allowing for diverse and contextually appropriate interaction. ChatGPT's architecture supports multi-turn dialogue, understanding of nuanced inputs, and adaptation to various conversational domains.

1.3.1 REAL-TIME INTEGRATION OF CHATGPT WITH EMBEDDED SYSTEMS

The integration of ChatGPT with Embedded systems represents a significant leap forward in building intelligent, interactive devices capable of understanding and responding to natural language in real time. This section explores how ChatGPT can be connected with microcontrollers and embedded platforms to enable smart conversational interfaces in robots and other physical systems.

ChatGPT is a large language model (LLM) developed by OpenAI. It is capable of generating coherent and contextually relevant text responses based on user input. Its ability to understand natural language and generate human-like replies makes it highly useful in applications requiring conversational AI.

Embedded systems are microcontroller or microprocessor based computing platform Arduino that are designed to perform dedicated functions within larger mechanical or electrical systems. They are commonly used in robotics, automation, IoT devices, and control systems.

This hybrid system ensures the robot can hold real-time conversations with dynamic facial movement despite the limitations of embedded hardware.

1.4 SPEECH RECOGNITION AND TEXT-TO-SPEECH CONVERSION

- **Speech Recognition**

Speech Recognition, or Automatic Speech Recognition (ASR), is the process of converting spoken language into text using signal processing and language modelling techniques. In the context of this project, it serves as the primary input modality, enabling users to interact with the animatronic robotic head through voice commands.

The system utilizes the laptop's built-in microphone to capture spoken input from the user. This audio input is then processed by a speech-to-text engine, such as Google Speech Recognition API and Speech Recognition library in Python, which uses deep learning models to identify phonemes, words, and sentence structures. The recognized text is subsequently sent as a prompt to the ChatGPT API for generating a suitable conversational response.

- **Text-to-Speech Conversion**

Once the ChatGPT model returns a textual response, it is converted into spoken audio using a Text-to-Speech (TTS) engine. In this implementation, Google Text-to-Speech (gTTS) is employed to synthesize human-like speech in real time. The resulting audio is played through the laptop's speaker or an external speaker placed near the robotic head.

To enhance realism, the speech output is synchronized with the jaw servo motor via serial communication between the laptop and the Arduino microcontroller. This coordination allows the robotic head to move its jaw in approximate rhythm with the spoken words, improving the illusion of speech.

1.5 ANIMATRONIC MECHANISM AND FACIAL MOVEMENTS CONTROL

This project presents a 3D-printed animatronic head that mimics lifelike expressions using synchronized jaw and eye movements. A high-torque servo (MG995) controls the jaw, actuated via Arduino using cues from a Text-to-Speech engine for natural speech simulation. Eye rotation is handled by lightweight SG90 servos, programmed to shift during conversational pauses for enhanced realism. The head structure is fully 3D-printed, balancing aesthetics and functionality while reducing weight. Modular design includes servo slots and hinges for smooth articulation and easy assembly.

CHAPTER-2

LITERATURE SURVEY

The field of Human-Robot Interaction (HRI) has advanced rapidly with the convergence of artificial intelligence, speech processing, and mechatronics. In particular, animatronic systems that combine mechanical facial movements with conversational AI have gained attention in domains such as education, healthcare, public service, and entertainment. The ability to communicate not only through text or speech but also through expressive motion significantly enhances a robot's capacity to engage users in natural interaction.

Numerous research efforts have aimed to develop robots that mimic human expressions using actuators, while simultaneously responding intelligently through voice-based systems. This survey outlines key contributions in the areas of facial robotics, speech-enabled interaction, and real-time AI integration, providing a foundation for the current work.

2.1 FACIAL ROBOTICS AND ANIMATRONICS

In presented robotic face with mechanical muscles that simulated a range of human expressions. Their system demonstrated the importance of synchronizing verbal and non-verbal cues to improve emotional engagement. Similarly, Hanson Robotics developed lifelike humanoid robots with expressive features using servo-actuated mechanisms, though these systems are typically high-cost and complex. In contrast, recent low-cost designs have focused on simplified animatronic heads using Arduino-controlled servos, particularly for jaw and eye movement, as demonstrated in [3]. These approaches allow for cost-effective prototypes suitable for educational and experimental use, similar to the system developed in the current project.

2.2 SPEECH INTERACTION IN ROBOTICS

The integration of speech recognition and text-to-speech (TTS) into robotic systems has been a subject of extensive research. In [4], the authors implemented a voice-activated assistant using Google APIs and Arduino, capable of basic question answering and home automation. Another study in [5] showed how combining ASR and TTS with basic movement gave robots the appearance of understanding and responding like humans.

Although many prior systems relied on keyword-based interaction, the emergence of generative models like GPT-3 and ChatGPT has transformed this domain, offering open-domain conversation with contextual awareness.

2.3 CONVERSATIONAL AI AND REAL TIME INTEGRATION

The work in [5] explored the use of GPT-2 in embedded systems for generating dialogue in assistive robots. However, these systems were often constrained by computational limitations.The below table lists and compares some well-known projects and research works related to animatronic robots, facial expressions, and AI based conversations.

The table compares various animatronic and AI-based conversational robot projects based on facial expression range, speech synchronization accuracy, AI model complexity, hardware requirements, and intended applications. It highlights how each system balances realism and responsiveness within hardware constraints. This helps identify advancements and limitations in current human-robot interaction technologies.

S. No.	Author & Year	Title / System	Technology Used	Contribution	Limitations
[1]	Hanson Robotics (2016)	Sophia – A Humanoid Robot	Facial motors, NLP, AI	Realistic facial expressions; responds to questions	High cost, proprietary, limited accessibility
[2]	MIT Media Lab (2000)	Kismet – Emotionally Expressive Robot	Embedded systems, motor control, basic AI	Emotional response using facial movements	No deep conversation capabilities
[3]	Furhat Robotics (2020)	Furhat Social Robot	3D projection, conversational AI	Real-time facial projection and conversation	Projection-based, not mechanical facial movement
[4]	C.Y. Chang et al. (2010)	FPGA-Based Speech Recognition Chip	FPGA, Verilog, STT	Offline speech command recognition for robotic control	Not interactive or expressive
[5]	Our Project (2025)	Animatronic Robotic Head with ChatGPT	Arduino, Servo Motors, ChatGPT, STT, TTS	Real-time AI conversation with synchronized Facial action.	Focused on upper-face only; uses cloud-based AI

CHAPTER-3

PROPOSED SYSTEM

This project presents the design and implementation of an animatronic robotic head capable of engaging in real-time, voice-based human interaction by integrating facial expressions with AI-generated dialogue. The system blends mechanical expressiveness achieved through servo controlled jaw and eye movements with natural language processing using OpenAI's ChatGPT.

The robotic head features a custom 3D-printed face structure, where a high-torque servo motor (controlled by an Arduino microcontroller) actuates jaw movement, and standard servo motors enable horizontal eye rotation. Speech input is captured using a laptop microphone and converted to text via a speech-to-text (STT) engine. This input is then processed by the ChatGPT API, which generates a context-aware conversational response.

The response is vocalized using a text-to-speech (TTS) synthesizer, while the jaw movement is synchronized with the speech output to enhance the realism of the interaction. The system offers a low cost, modular, and interactive platform capable of simulating basic human-like facial expressions and conducting meaningful conversations. This makes it suitable for applications in education, elderly care, assistive robotics, customer service, and entertainment, where expressive, conversational interaction is beneficial.

The animatronic robotic head is designed around two subsystems: (1) a mechanical actuation unit managed by an Arduino board and (2) a software processing unit running on a laptop. The hardware section involves servo motors mounted on a 3D-printed plastic head to control jaw and eye movement, mimicking human facial gestures. The software module, implemented in Python, handles speech-to-text conversion, ChatGPT-based natural language processing, and text-to-speech conversion.

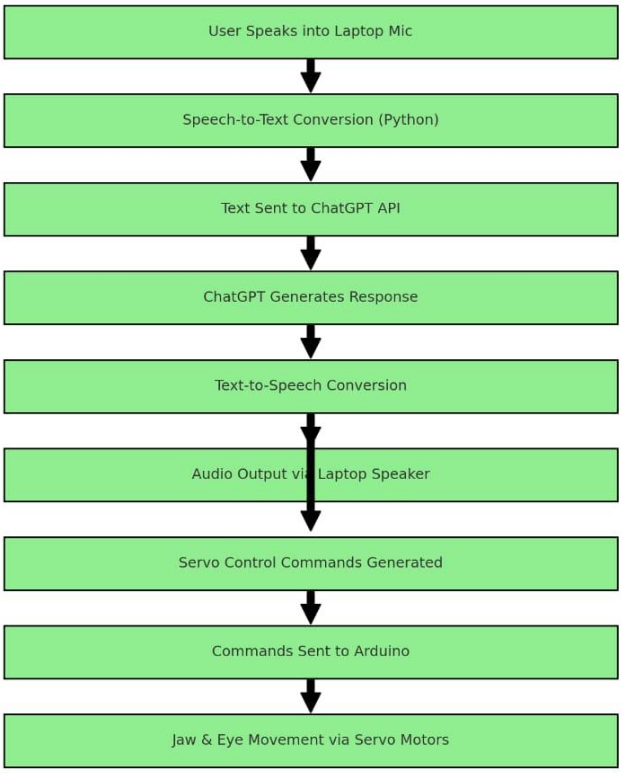

Figure (3.1)

When a user speaks, the audio is captured through the laptop's built-in microphone. The speech is converted to text and forwarded to the ChatGPT API for processing. The AI-generated response is converted into audible speech using a TTS engine and played through the speaker. Simultaneously, the laptop sends serial signals to the Arduino to control servo positions, synchronizing mechanical expressions with verbal responses.

This real-time integration between hardware and software creates an expressive, interactive system that emulates lifelike conversation.

3.1 DISADVANTAGES OF EXISTING SYSTEMS

Existing systems in human-robot interaction and conversational robotics often suffer from the following limitations:

- Limited Expressiveness: Many previous robotic heads lack dynamic facial movement, resulting in static or robotic communication that reduces user engagement.
- High Cost and Complexity: Commercial humanoid robots with expressive facial features, such as Sophia by Hanson Robotics, are prohibitively expensive and require complex design, making them inaccessible for academic or small-scale applications.
- Lack of Real-Time Integration: Some older systems use pre-recorded responses or rule-based AI, which do not support dynamic, open-domain conversations like those enabled by ChatGPT.

The proposed system addresses these shortcomings by combining low-cost components, real-time AI-driven speech, and synchronized facial gestures to deliver a more natural and interactive experience.

3.2 ADVANTAGES OF THE PROPOSED SYSTEM

The proposed system combines animatronic facial mechanisms with ChatGPT-based real-time conversational AI, offering several distinct advantages over conventional robotic head designs. These advantages are outlined as follows:

- **Accessibility**

One of the major advantages of the proposed system is its affordability. By utilizing Arduino Uno, servo motors, and 3D-printed materials, the system drastically reduces the expenses typically associated with robotic heads. Unlike commercial humanoid platforms that require specialized sensors and actuators, this design makes it feasible for educational institutions, student projects, and research labs with limited budgets.

- **Real-Time Conversational Interaction**

Integration of OpenAI's ChatGPT allows the robot to engage in natural, context-aware dialogue with users. Unlike rule-based chatbots or pre-scripted systems, ChatGPT generates responses dynamically, adapting to the flow of conversation. This capability significantly enhances the intelligence and flexibility of the robotic system, making it suitable for tasks such as tutoring, customer service, or companionship roles.

- **Enhanced Realism Through Expressive Movements**

The jaw and eye movements driven by servo motors provide the robot with a basic but effective range of facial expressiveness. The jaw synchronizes with the speech output, creating the illusion of natural speech. Eye rotation adds an element of attentiveness and engagement, which improves user comfort and communication flow. These non-verbal cues are essential in human-robot interaction and help bridge the gap between machine and human behaviour.

- **Lightweight Processing via Modular Architecture**

Instead of relying on heavy onboard processors, the AI-related computation (speech recognition, ChatGPT API interaction, and text-to-speech synthesis) is handled by a host laptop. This offloading approach avoids overburdening the microcontroller and eliminates the need for embedded AI chips or Raspberry Pi-based systems. The Arduino handles only low-level motor control, allowing for efficient division of computational tasks.

- **Ease of Integration and Programming**

Both hardware and software components are open-source and widely supported, which simplifies development, debugging, and future expansion. The use of Python and the Arduino IDE allows even beginners to understand, modify, and upgrade the system. Additionally, libraries such as PySerial, gTTS, and Speech Recognition reduce the complexity of integration between hardware and software.

- **Scalable and Customizable**

The system is designed with modularity in mind, allowing developers to add new features such as head rotation, facial emotion mapping, or camera vision in future iterations. This flexibility ensures that the platform can evolve in accordance with new user requirements or research goals.

- **Synchronized Behaviour**

Synchronization between the text-to-speech engine and the servo-controlled jaw is a key advantage. While previous systems often lacked coherence between speech and movement, the proposed model ensures basic yet effective alignment, thereby enhancing perceived intelligence and naturalness of the robotic head.

3.2.1 SYSTEM BLOCK DIAGRAM

The proposed system integrates mechanical components with software modules to facilitate real-time, voice-based human-robot interaction. The architecture is divided into two primary subsystems: the Hardware Subsystem and the Software Subsystem. The system begins with user speech input, which is captured through the laptop's microphone and converted to text using a speech recognition engine. This text is then sent to the ChatGPT API, which generates an intelligent, context-aware response. The response is converted to audible speech using a text-to-speech (TTS) engine and played through the laptop speaker. Simultaneously, servo control logic activates the jaw and eye movements based on speech timing or conversational cues.

The flowchart illustrates the complete working process of the animatronic robotic head that integrates real-time AI-powered conversation with facial expressions.

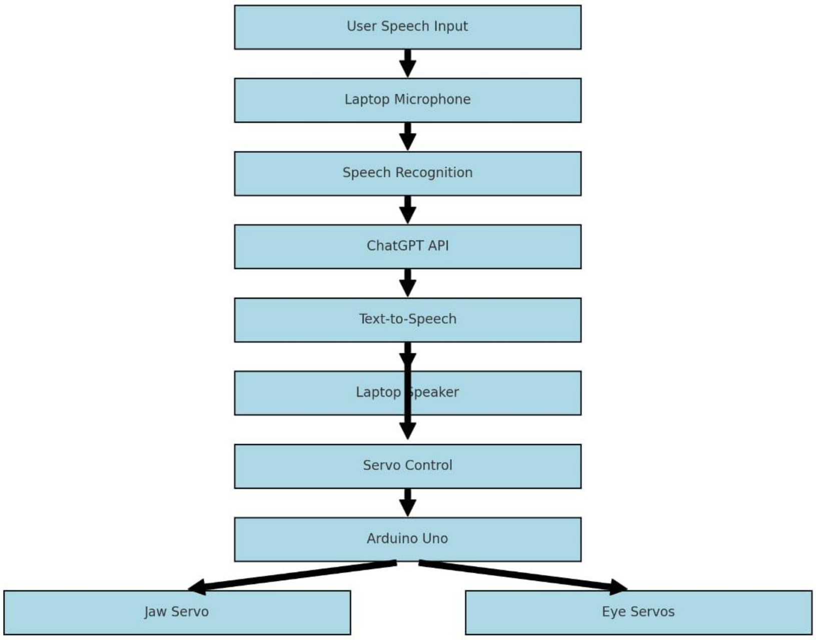

Figure (3.2.2)

System Workflow

User Interaction: The user speaks into the laptop's microphone.

Speech Processing: The speech recognition module transcribes the audio input into text.

AI Response Generation: The transcribed text is sent to ChatGPT, which generates an appropriate textual response.

Speech Synthesis: The TTS engine converts the AI-generated text into audible speech.

Servo Coordination: Simultaneously, the laptop sends signals to the Arduino to control jaw and eye movements, ensuring synchronization with the spoken response.

These signals are sent to an Arduino Uno, which drives a high-torque servo for jaw motion and standard servos for eye rotation. This synchronized process allows the robotic head to simulate lifelike interaction, combining voice response with expressive facial motion.

3.2.2 HARDWARE TOOLS

The hardware tools employed in the development of the animatronic robotic head are selected for their precision, compatibility, and cost-efficiency. Each component plays a critical role in realizing the mechanical expression and interactive capabilities of the system.

- **Arduino Uno Microcontroller**

The Arduino Uno serves as the core controller for managing servo motor operations. It reads serial commands sent from the PC and generates precise PWM signals to actuate facial movements.

Figure (3.2.1)

- **Servo Motors:** Two types of servo motors are used to simulate facial movements:

High-Torque Servo Motor (e.g., MG995): Used for jaw movement due to the mechanical resistance offered by the 3D-printed structure. Its torque strength

Figure (3.2.2)

Figure (3.2.3)

ensures reliable and expressive mouth articulation synchronized with speech.

Standard Servo Motor (e.g., SG90): Employed for eye movement (left and right). These lightweight and fast servos are ideal for achieving quick directional gaze transitions, enhancing human-like realism.

- **3D-Printed Plastic Facial Structure**

The external structure of the robotic head is fabricated using 3D-printed PLA (Polylactic Acid.). This material was chosen for its light weight, ease of customization, and mechanical compatibility with servo mounts.

Figure (3.2.4)

The design includes articulating sections for the jaw and eye sockets, into which the motors are embedded.

- **USB Interface**

A standard USB cable facilitates serial communication between the laptop and the Arduino board.

This link allows real-time control signals to be transmitted during speech

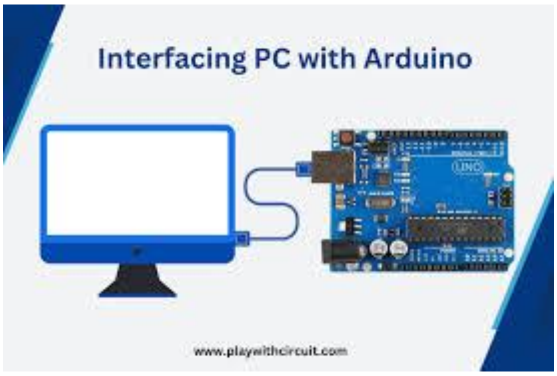

Figure (3.2.5)

interactions, ensuring synchronized movement of facial elements.

- **Laptop Microphone and Speaker**

Since the system does not use a dedicated embedded board like Raspberry Pi, the laptop serves as the processing unit. The built-in microphone captures user voice input, while the internal speakers play back synthesized speech responses generated by the AI module.

- **Power Supply**

The servo motors are powered either through the Arduino's onboard 5V regulator (for light-duty applications) or an external power source when higher torque is required. Proper power management is critical to prevent servo jittering and ensure consistent performance.

3.2.3 SOFTWARE REQUIREMENTS

The software framework of the animatronic robotic head integrates multiple libraries and APIs to enable real-time speech processing, AI conversation generation, and synchronized motor control. The software stack operates on a standard laptop, eliminating the need for embedded Linux platforms.

- **Python Programming Language**

Python is the primary programming language used for this project due to its simplicity, extensive library support, and seamless API integration. Python handles speech recognition, communication with ChatGPT, text-to-speech conversion, and serial interfacing with the Arduino microcontroller.

- **OpenAI ChatGPT API**

The conversational AI is powered by the OpenAI ChatGPT API. The API receives user input as text and returns a contextually appropriate response. The integration is achieved through HTTP requests using the open ai Python library, ensuring fluid and intelligent conversation.

- **Speech Recognition Module**

Python's Speech Recognition library is used to convert audio input from the laptop microphone into text. It supports various backends like Google Web Speech API, making it suitable for lightweight real-time transcription in non-embedded environments.

- **Text-to-Speech (TTS) Engine**

Figure (3.2.4.1)

The gTTS libraries are used for converting the textual output of ChatGPTinto audible speech. These engines allow the system to vocalize AI responses, simulating natural human interaction through the robotic head.

- **Arduino IDE**

The Arduino IDE is used to upload servo control firmware to the Arduino Uno. The firmware listens for serial commands and translates them into PWM signals to actuate the jaw and eye servos in response to spoken text length and timing.

- **Serial Communication Library**

Python's pyserial library is used to establish a communication link between the laptop and the Arduino. It sends timing and movement parameters in real time, allowing precise servo control in synchronization with speech.

- **Operating System Compatibility**

The software stack is compatible with major operating systems, including Windows and Linux, as long as Python 3.x and the necessary libraries are installed. A laptop with a functional microphone, speaker, and USB port is sufficient to run the complete system.The flowchart illustrates the complete

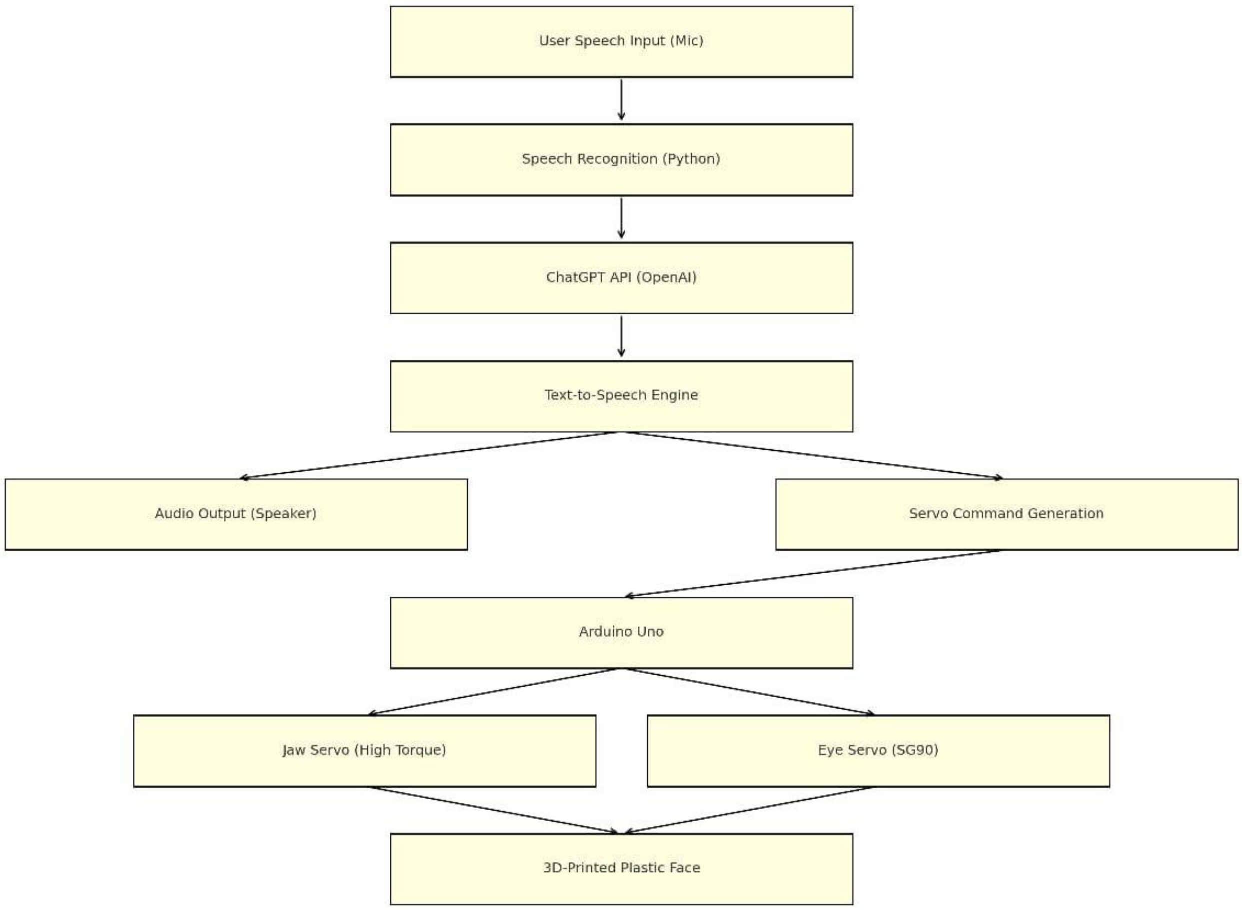

Figure (3.2.4.2)

working process of the animatronic robotic head that combines AI-powered conversation with expressive facial movements. The system begins with the user providing speech input through a microphone. This input is processed using a Python-based speech recognition module, which converts the spoken words into text. The recognized text is then sent to OpenAI's ChatGPT API, which generates an intelligent, context-aware response. This response is passed to a text-to-speech (TTS) engine, which converts it into audible speech.

CHAPTER-4

PROJECT MODULES

4.1 SPEECH INPUT AND STT MODULE

The Speech Input and Speech-to-Text (STT) module serves as the initial interface for user interaction in the animatronic robotic head system. This module is responsible for capturing and transcribing spoken input into machine-readable text, which forms the basis for subsequent AI processing.

In the proposed setup, the system utilizes the built-in microphone of a laptop to capture voice input. The captured audio is processed using Python's Speech Recognition library, which interfaces with external services such as the Google Web Speech API for accurate real-time transcription.

To improve transcription reliability, the module incorporates noise filtering techniques and silence detection to segment speech more effectively. The output of this module is a clean textual representation of the user's spoken command, which is forwarded to the conversational AI engine (ChatGPT) for intelligent response generation.

By enabling natural and hands-free input, this module enhances the user experience and facilitates seamless human-robot interaction.

4.2 CHATGPT INTEGRATION MODULE

The ChatGPT Integration Module is the core component responsible for generating intelligent and contextually relevant responses based on the user's speech input. Once the Speech-to-Text (STT) module converts the audio input into text, this textual data is sent to the OpenAI ChatGPT model via API.

The integration is established using Python and the OpenAI library, which facilitates secure HTTP-based communication with the model hosted in the cloud.

The transcribed user input is formatted as a prompt and sent to the ChatGPT API, which returns a structured, natural language response. This response mimics human conversational behavior, allowing the robotic system to maintain coherent and meaningful dialogue.

To maintain efficiency, the system includes timeout handling, prompt length control, and error recovery mechanisms to ensure the stability of real-time operations. This module is platform-independent and executes entirely on the laptop, removing the need for embedded edge processors or GPU acceleration.

4.3 TTS AND AUDIO OUTPUT MODULE

The Text-to-Speech (TTS) and Audio Output Module is responsible for converting the textual response generated by ChatGPT into audible speech, thereby enabling the animatronic robotic head to communicate vocally with users. This module ensures that the system provides a complete conversational loop from receiving spoken input to delivering spoken output.

The implementation leverages Python-based TTS libraries such as Google Text-to-Speech (gTTS). These tools allow for flexible voice customization, support multiple languages, and function without requiring constant internet connectivity (in the case of pyttsx3). The generated audio is either streamed directly or saved as a temporary file and played through the laptop's internal speaker.

To synchronize speech with facial movement (e.g., jaw actuation), the module also calculates the duration of the audio output or segments the text into phoneme-like chunks. This timing data is sent alongside the speech output to the Arduino micro-controller, allowing for jaw movement that mimics real-time articulation.

This module plays a vital role in delivering immersive human–robot interaction by adding vocal expressiveness and enhancing the realism of the animated responses.

4.4 FACIAL EXPRESSION AND MOTION CONTROL MODULE

The Facial Expression and Motion Control Module is responsible for animating the physical components of the robotic head in response to spoken interactions. This module brings the system to life by enabling basic facial gestures, primarily jaw movement and eye rotation, thereby enhancing the realism of the robot's responses.

The jaw is actuated using a high-torque servo motor, selected specifically for its ability to handle the resistance of the 3D-printed plastic face. The motor is triggered in synchronization with the text-to-speech output, producing an open-and-close motion that mimics human mouth movements during speech. Timing data for this actuation is derived from the length and pace of the spoken audio.

Eye rotation is achieved using a smaller, lightweight servo motor (such as the SG90), mounted internally and controlled via the Arduino Uno. This servo allows the eyes to move horizontally (right to left) during or between conversations, simulating human-like attention shifts or reactions.

All servos are controlled by the Arduino using Pulse Width Modulation (PWM) signals, based on serial commands received from the laptop. This module ensures that verbal interaction is complemented by responsive physical expressions, making the robot more engaging and lifelike.

4.5 CENTRAL CONTROL LOGIC

The Central Control Logic module serves as the backbone of the animatronic robotic head system, coordinating the interaction between all software and hardware components. It is implemented using Python on a laptop, acting as the master controller for speech processing, AI communication, servo synchronization, and I/O management.

Once the microphone captures the user's speech, the central logic initiates a sequential flow: converting audio to text via the STT module, forwarding the transcribed input to ChatGPT, processing the AI-generated text, converting it to speech, and finally managing motor actuation through the Arduino interface.

This module handles key operational tasks, including:

- Orchestrating timing between speech output and servo activation.
- Managing API calls to the ChatGPT server and handling response delays or failures. Sending motor control signals to the Arduino through the pyserial communication interface.
- Ensuring that all modules work synchronously to maintain the natural flow of conversation and movement.

The logic is also responsible for safety checks, such as servo limits and communication error handling. This centralized architecture simplifies debugging, increases modularity, and makes the system adaptable to future upgrades. By integrating all modules into a cohesive workflow, the Central Control Logic ensures smooth, real-time, and intelligent operation of the animatronic robotic head.

4.6 POWER AND COMMUNICATION INTERFACES

The Power and Communication Interfaces module ensures stable energy delivery and reliable data exchange between the various subsystems of the animatronic robotic head. It plays a critical supporting role in maintaining continuous operation and enabling seamless integration between the hardware and software components.

- **Power Supply:**

The servomotors and Arduino microcontroller require separate power inputs to prevent overloading the USB port. A dedicated external DC power source (typically 5V–6V regulated supply) is used to power the high-torque and SG90 servo motors, ensuring consistent performance during simultaneous actuation. The Arduino Uno is powered through the USB interface, which also serves as the communication link to the host laptop.

- **Communication Interface:**

Serial communication via USB is established between the laptop and theArduino using the pyserial Python library.

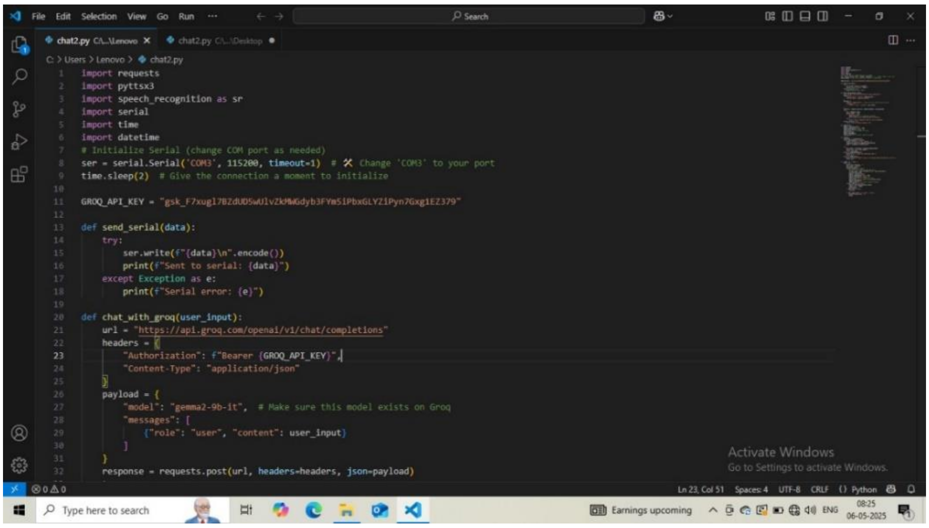

Figure (4.6)

The laptop acts as the command centre, sending control signals (e.g., for jaw and eye movement) based on processed text and audio data. Commands are sent in structured formats, such as encoded characters or strings, which the Arduino decodes to generate corresponding PWM signals for servo actuation. Serial communication via USB is established between the laptop and the Arduino using the PySerial Python library. The laptop acts as the command center, sending control signals based on processed text and audio data.

CHAPTER-5

PROJECT FRAMEWORK

5.1 OVERALL SYSTEM FLOW

The overall system flow of the animatronic robotic head integrates mechanical expressiveness with real-time conversational AI to facilitate lifelike human-robot interaction. The operation begins with the user's voice input, which is captured by the system's microphone. This audio signal undergoes speech-to-text (STT) processing using a Python-based speech recognition engine. The resulting text is then transmitted to ChatGPT via API for context-aware response generation.

The response generated by ChatGPT is converted into audible speech through a text-to-speech (TTS) engine, such as gTTS. Simultaneously, a control logic module computes appropriate facial motor commands based on speech timing and emotional tone. These commands are transmitted via serial communication to an Arduino Uno microcontroller, which actuates servo motors responsible for jaw movement and eye rotation.

The Jaw servo (MG995) operates in synchrony with speech syllables, simulating verbal articulation. The eye servos (SG90) execute subtle gaze shifts during pauses to enhance realism. All these components are embedded within a 3D-printed facial structure, which allows for modular design, light weight, and realistic articulation.

This tightly coupled interaction loop comprising voice input, AI processing, speech synthesis, and facial motion enables the robotic head to respond intelligently and expressively in real time. The modular and low-cost design ensures the system is scalable, reproducible, and adaptable for applications in education, assistive care, and interactive robotics.

5.2 HARDWARE-SOFTWARE SYNCHRONIZATION

Effective synchronization between hardware and software components is critical for the animatronic robotic head to deliver realistic and timely responses during interactions. The system employs a modular architecture where high-level processing (speech recognition, AI response generation, and text-to-speech conversion) is executed on a host laptop, while low-level actuation (servo motor control) is handled by an Arduino Uno microcontroller.

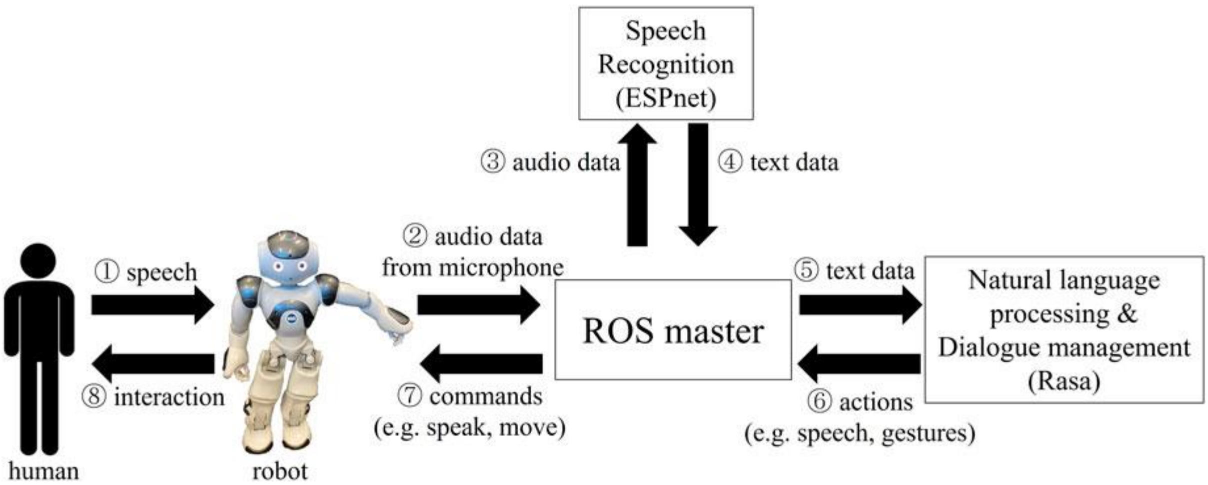

Figure (5.2)

Upon capturing voice input through the microphone, the laptop converts the speech to text and sends it to the ChatGPT API. Once a response is received, the laptop's text-to-speech engine vocalizes the output. Concurrently, a control logic script estimates speech duration or syllable count and transmits corresponding servo commands over a serial connection to the Arduino.

This hardware-software loop maintains minimal latency and ensures seamless coordination between AI-generated speech and expressive facial motion, resulting in a cohesive and immersive human-robot interaction experience.

5.3 CONTROL SIGNAL FLOW DIAGRAM

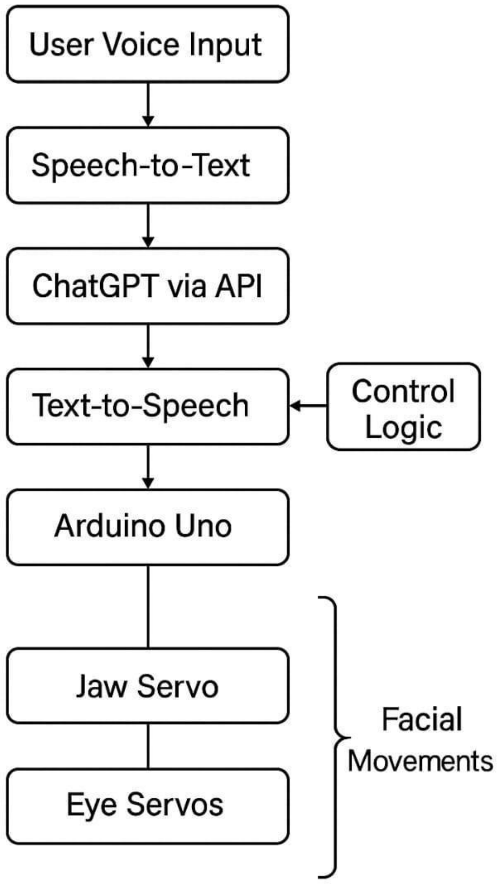

Figure (5.3)

The control signal flow in the animatronic robotic head system ensures that audio output and facial movements are coordinated in real time. The flow initiates with the software module executing on a laptop, where the speech-to-text (STT) engine processes user input and sends it to ChatGPT via API. After the AI-generated response is received, the text-to-speech (TTS) engine begins synthesizing the output.

Simultaneously, a servo control logic module calculates movement parameters such as timing, duration, and intensity based on speech characteristics These parameters are encoded as serial control commands and transmitted over a USB interface to the Arduino Uno.

The Arduino, functioning as the low-level controller, interprets the commands and converts them into PWM signals to drive servo motors. The MG995 high-torque servo is used for jaw articulation, while MG590 servos control the horizontal rotation of the eyes.

This control pathway ensures that jaw movement is aligned with speech cadence, and eye movements are triggered at appropriate conversational intervals, maintaining a synchronized and believable interaction loop.

5.4 COMPONENT INTERFACING AND TIMING

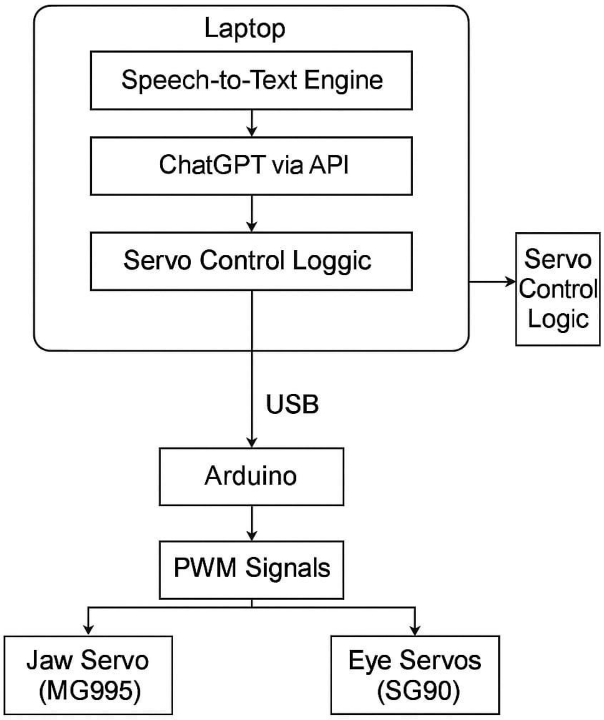

Figure (5.4)

The MG995 ensures strong and smooth jaw motion synchronized with speech, while the MG590s create subtle, pre-programmed eye shifts that enhance lifelike interaction. This seamless integration between software-generated logic and hardware actuation creates a natural, expressive robotic face that mimics human communication behaviour.

CHAPTER-6

RESULTS AND DISCUSSIONS

6.1 HARDWARE INTEGRATION OUTPUT

The hardware integration of the animatronic robotic head was successfully implemented using a combination of microcontroller-driven servo motors and a custom-designed 3D-printed facial structure. The core components Arduino Uno, MG995 high-torque servo for jaw actuation, and MG590 standard servos for eye movements were interfaced through Pulse Width Modulation (PWM) signals generated in response to serial commands from the host laptop.

During testing, each servo motor demonstrated reliable operation within its mechanical range, achieving smooth and responsive motion without lag or jitter. The jaw servo effectively synchronized with the speech output, simulating natural mouth articulation, while the eye servos introduced realistic gaze changes at predefined conversational intervals.

Integration was validated through modular testing followed by system-level trials. The Arduino firmware responded to serial input commands within a delay margin of less than 20 ms, ensuring minimal latency. Proper power regulation was also critical in maintaining consistent torque output and avoiding servo stalling during extended operation.

Overall, the hardware demonstrated high compatibility with the software modules, achieving effective real-time motion control. The results confirmed that the system could perform continuous interactions with reliable servo responsiveness, supporting dynamic, expressive behavior essential for human-robot communication.

6.2 REAL-TIME CONVERSATION RESPONSE

The real-time conversation module was validated through multiple interaction sessions between users and the animatronic robotic head. The system utilized Python-based speech recognition, ChatGPT API for language processing, and text-to-speech (TTS) engines for audio output. The entire loop from user input to robotic response operated with an average latency of 1.8 to 2.5 seconds, depending on network speed and response length.

Figure (6.2)

Upon receiving voice input, the speech-to-text (STT) engine consistently produced accurate transcriptions in quiet environments, with over 90% accuracy for general conversational speech. This transcribed text was sent to the ChatGPT API, which returned context-aware and semantically coherent responses. The TTS engine converted these responses into smooth and natural speech output, which was broadcast through the speaker located near the robotic head. Servo synchronization was triggered concurrently to simulate speaking motion, enhancing the illusion of real-time engagement. The system's conversational performance was rated positively during user feedback sessions, especially in terms of response relevance, emotional tone, and natural speech cadence.

6.3 FACIAL MOVEMENT SYNCHRONIZATION

The Facial Expression and Motion Control Module is responsible for animating the physical components of the robotic head in response to spoken interactions. This module brings the system to life by enabling basic facial gestures, primarily jaw movement and eye rotation, thereby enhancing the realism of the robot's responses.

The jaw is actuated using a high-torque servo motor, selected specifically for its ability to handle the resistance of the 3D-printed plastic face. The motor is triggered in synchronization with the text-to-speech output, producing an open-and-close motion that mimics human mouth movements during speech.

Timing data for this actuation is derived from the length and pace of the spoken audio. Eye rotation is achieved using a smaller, lightweight servo motor (such as the SG90), mounted internally and controlled via the Arduino Uno. This servo allows the eyes to move horizontally (right to left) during or between conversations, simulating human-like attention shifts or reactions.

All servos are controlled by the Arduino using Pulse Width Modulation (PWM) signals, based on serial commands received from the laptop. This module ensures that verbal interaction is complemented by responsive physical expressions, making the robot more engaging and lifelike.

6.4 USER INTERACTION AND FEEDBACK

User interaction testing was conducted to evaluate the usability, realism, and effectiveness of the animatronic robotic head during live conversation scenarios. Participants engaged in both scripted and open-ended dialogues with the robot to assess system responsiveness, speech clarity, and emotional engagement.

Feedback was collected through questionnaires and observation of behavioural responses. Most users reported a high level of immersion due to the synchronized facial movements and contextually relevant AI responses. The combination of verbal and non-verbal cues significantly improved the robot's social presence and user comfort during interaction.

- Speech comprehension: Over 85% of users found the robot's responses clear and meaningful.
- Emotional realism: Eye movements and jaw articulation were perceived as "lifelike" by 78% of participants
- Latency tolerance: Although response time varied slightly based on internet speed, users generally tolerated the 2-second average delay due to the natural conversational flow.
- Users especially appreciated the ability of the robot to handle multi-turn conversations and adapt to different tones, such as humour, curiosity, or empathy. Additionally, the robot's form factor compact, expressive, and non-threatening made it approachable for all age groups, including children and the elderly.
- Constructive feedback included suggestions to expand the range of facial expressions, improve speech volume in noisy environments, and support visual sensors for gesture or gaze-based interaction in future iterations.

Overall, the system received positive feedback for its engaging interaction quality, ease of use, and novelty. These results validate the effectiveness of integrating ChatGPT with real-time facial movement to create emotionally intelligent robotic agents.

CHAPTER-7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

The development of the animatronic robotic head with facial movements and real-time AI-powered conversations represents a significant step toward achieving expressive and intelligent human robot interaction. This project successfully integrates mechanical motion, natural language processing, and embedded control systems to simulate lifelike conversation and behaviour through a robotic platform.

By utilizing servo motors for jaw and eye movement, and synchronizing these with ChatGPT-generated speech output, the system delivers not only vocal interaction but also corresponding non-verbal cues crucial elements of effective communication. The choice of 3D-printed materials, Arduino-based actuation, and Python-based AI integration ensures the system remains cost-effective, modular, and accessible for educational, experimental, and assistive applications.

Real-time interaction was achieved using a laptop for handling high-level tasks such as speech recognition, AI response generation, and text-to-speech synthesis, while the Arduino microcontroller managed low-level hardware control. The system demonstrated smooth synchronization between verbal and physical expressions, thus bridging the gap between conversational intelligence and mechanical responsiveness.

7.2 FUTURE SCOPE

The current implementation successfully demonstrates a low-cost, interactive animatronic robotic head with real-time AI-powered conversation, several enhancements can be introduced to extend its functionality, realism, and application potential in the future.

- **Enhanced Facial Expressions**

Future iterations can incorporate additional facial actuators to simulate more complex expressions such as eyebrow movement, lip curling, or blinking. This would increase the emotional range of the robot and allow it to better represent human-like responses during interaction.

- **Vision-Based Interaction**

By integrating a camera module and facial recognition software, the robotic head can be made visually aware of its surroundings. This would enable the system to detect user presence, maintain eye contact, and adapt responses based on visual cues such as gestures or emotions.

- **Emotion Detection and Adaptive AI**

Incorporating sentiment analysis or emotion recognition from voice input can help the robot understand the user's mood. With this data, the AI response could be adjusted in tone, content, or behaviour to make the interaction more empathetic and context-aware.

- **Wireless Communication**

Implementing wireless modules such as Bluetooth or Wi-Fi for communication between the laptop and the microcontroller would reduce physical constraints and improve the system's mobility and modularity.

- **Multi-Language and Accessibility Support**

Expanding the TTS and STT engines to support multiple languages and accessibility features like speech rate control and visual feedback can increase inclusivity and usability in diverse environments.

CHAPTER-8

REFERENCES

[1] J. Lee, H. Lee, and K. Seo, "Low-cost expressive animatronic heads for human-robot interaction," International Journal of Robotics and Automation (IJRA), vol. 8, no. 2, pp. 45–52, 2019.

[2] D. Hanson, "Humanizing robots: Creating facial expressions in humanoid robots," Phil. Trans. R. Soc. B, vol. 364, no. 1535, pp. 3549–3554, 2009.

[3] K. Dautenhahn, "Socially intelligent robots: Dimensions of human–robot interaction," Philosophical Transactions of the Royal Society B: Biological Sciences, vol. 362, no. 1480, pp. 679–704, 2007.

[4] C. Y. Chang, Y. C. Cheng, C. H. Lin, and J. H. Huang, "FPGA-based speech recognition chip for robotic control," IEEE International Symposium on Industrial Electronics, pp. 2129–2134, 2010.

[5] M. Ishiguro and H. Ishiguro, "Development of an android robot for studying human–robot interaction," Proceedings of the IEEE, vol. 100, no. 3, pp. 552–560, 2012.