MORSOLINGO

Team Members Details

1.) Yashvi Mehta (BT2024136)
2.) Dhyan Patel (BT2024075)
3.) Hardh Kava (BT2024041)
4.) Aditi Pandey (BT2024132)
5.) Hemakshi Jadeja (BT2024002)
6.) Raj Gandhi (BT2024172)

PROJECT WILL ONLY WORK WITH TYPESCRIPT CODE, CONVERTING IT TO .hex FILE.

Github Repository Link : https://github.com/yashvi-27/Morsolingo.git

Demonstration link :

https://drive.google.com/drive/folders/1-_08m_GYKpbuucokPn31rE68nyytWaG4

Objective of the Project

The primary objective of this project was to create an interactive **Morse code training tool** using the Micro: bit. The tool helps users learn Morse code by:

- Randomly generating letters and playing their Morse code pattern.
- Allowing users to send their guesses through a mobile app connected via Bluetooth.
- Providing real-time feedback (tick/cross) based on the user's input.

Additionally, the project supports different operating modes for text-to Morse conversion, Morse-to-text conversion, and two quiz modes (letter quiz and Morse quiz).

Implementation Details

- Modes of Operation:
  - T1 (Text to Morse): User sends English text through mobile app, and Micro: bit plays the Morse code.
  - Q1 (Letter Quiz): Micro: bit randomly selects a letter, plays its Morse code, and user types the guessed letter.
  - T2 (Morse to Text): User sends Morse code, and Micro: bit decodes and displays the corresponding English letter.
  - Q2 (Morse Quiz): Micro: bit displays a letter; user sends its Morse code.
- Controls:
  - Button A: Switch between training and quiz modes within the current group (T1<->Q1, T2<->Q2).
  - Button B: Switches between T1↔T2 and Q1↔Q2.
- Morse Code Playback:
  - Dots and dashes are played with corresponding LED patterns and sound tones.
  - Timing rules strictly follow Morse code standards.
- Answer Checking:

- User inputs are compared against the correct letter (in letter quiz) or Morse code (in Morse quiz).
- Correct inputs are rewarded with a tick icon and a tone.
- Incorrect inputs result in a cross icon, error tone, and display of the correct answer.

Challenges Faced and How They Were Addressed

- Searching for Project Ideas:
  It was challenging to find an engaging and technically feasible idea that combined Morse code, Bluetooth, and mobile interaction.
- Connecting Bluetooth to Device:
  Setting up Bluetooth communication between the Micro: bit and the mobile app required careful understanding of UART services and troubleshooting connectivity issues.
- Generating ARM Code from Scratch:
  Writing the ARM-based MakeCode program in typescript without any prior experience in ARM microcontroller programming was difficult and required significant research and trial-and-error.
- Working Simultaneously with Typescript and ARM Hardware:
  Managing javascript/typescript, Bluetooth communication, and ARM hardware programming at the same time made the project complex and demanded multitasking across different technologies.

Results:

- The project successfully works as intended.
- Users can connect via the Micro: bit mobile app, receive Morse code quizzes, and get instant feedback.
- The system reliably interprets both English-to-Morse and Morse-to English conversions.
- Training and quiz modes offer a comprehensive way to learn Morse code interactively.