Morse Code Encoder and Decoder

Team Members:
Sricharan Vinoth Kumar, Saikiran S

December 1, 2024

Contents

1 Problem Statement

To Implement a morse code encoder and decoder using Arduino.

2 Aim of the Project

To use Arduino to implement a Morse code encoder and decoder obtaining inputs in various forms and displaying output in various forms.

3 Components Used

- Arduino(with USB cable)
- Push Button
- Breadboard
- 2x16 LCD display
- Buzzer
- LED
- Jumper Wires
- Some Resistors
- ESP 8266
- Dip switch
- Potentiometer

4 Circuit Schematics and Process Diagrams

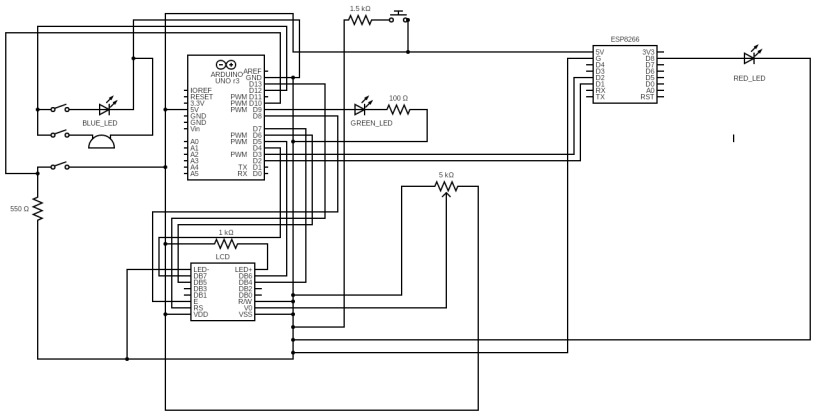

Figure 1: Circuit Schematic

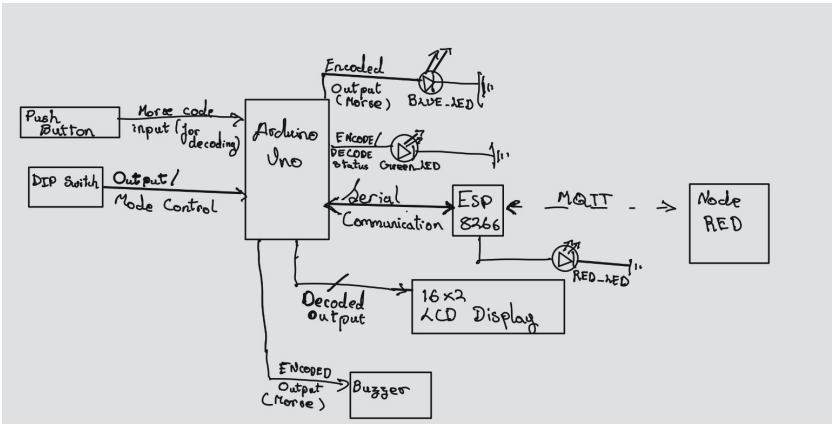

Figure 2: Process Diagram

5 Flowchart

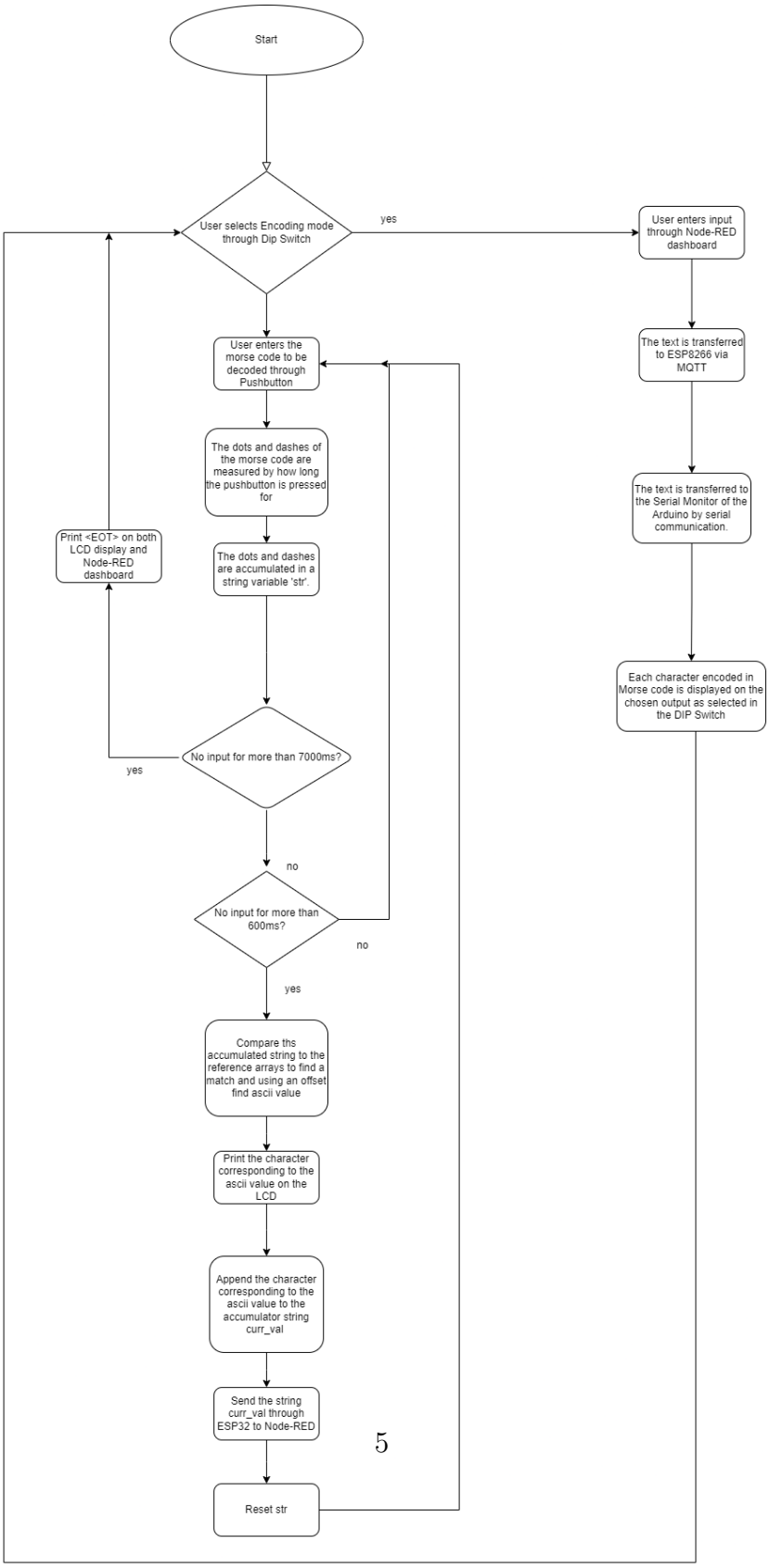

Figure 3: Flowchart

6 Description of the Project

Encoding/Decoding Process

For Encoding

1. The ESP takes input through Node-Red wirelessly.
2. The ESP writes the data to the Arduino via Serial port.
3. The Arduino takes the input and divides the input into characters.
4. The Arduino looks up the Morse Code equivalent of each character, outputs it, and moves to the next character, until it reaches the end.
5. The output is displayed by 2 methods:
   (a) **Visual Output:** Flashing an LED in a pattern closely matching the standards of Morse Code, i.e.,
      - ON for 200ms for a dot,
      - ON for 600ms for a dash,
      - OFF for 600ms for character separation,
      - OFF for 1200ms for word separation.

   (b) **Audio Output:** Toggling a buzzer in the same pattern mentioned above.
6. The user can select which output to enable or disable using switches on the PCB, which gate the output signal sent by the Arduino to these components.

For Decoding

1. The Arduino takes input through a push-button embedded in the PCB.
2. The input is processed into Morse Code using logic similar to the encoding process, with adjustments to account for human reaction times.
   - The press of a push-button is defined as HIGH, and the state of no press is defined as LOW.
   - The duration for which the button is HIGH or LOW determines the Morse Code.

3. When a character separation is detected and the input of the next letter begins, the decoded English letter is outputted.
4. When there is no input for 7 seconds, the decoded string is reset.
5. The output letters are written to the ESP via Serial port.
6. The ESP sends this data to Node-Red wirelessly.
7. The output is displayed in the Node-Red dashboard in textual format.
8. The user can switch between the two modes using a switch embedded in the PCB.

Data Transfer Mechanism

- The data transfer between the Arduino and the ESP module takes place through a Virtual Serial port.
- The data transfer between the ESP and the Node-Red server takes place through Wi-Fi using the MQTT Protocol, with the help of a Mosquitto Broker.

MQTT Protocol Topics

- Input given in the Node-Red dashboard is sent under the topic `encoding/inp`.
- Data sent to the Node-Red dashboard during encoding (the word being currently encoded) is sent under the topic `arduino/serial`, with the identifier `1_`.
- Data sent to the Node-Red dashboard during decoding (the decoded string) is sent under the topic `decoding/out`, with the identifier `0_`.

7 Results

Thus the circuit for a morse code encoder and decoder has successfully been implemented and the conversion from and to morse code as per the table has been achieved.

Letter/Number	Morse Code		Letter/Number	Morse Code	
A	.-		N	-.	
B	-...		O	---	
C	-.-.		P	.--.	
D	-..		Q	--.-	
E	.		R	.-.	
F	..-.		S	...	
G	--.		T	-	
H	....		U	..-	
I	..		V	...-	
J	.---		W	.--	
K	-.-		X	-..-	
L	.-..		Y	-.--	
M	--		Z	--..	
1	.----		6	-....	
2	..---		7	--...	
3	...--		8	---..	
4	....-		9	----.	
5	.....		0	-----	

Table 1: Morse Code for Letters and Numbers

8 Short Video Demonstration

- Demonstration Video: `https://bit.ly/4icYdnE`

9 Photos

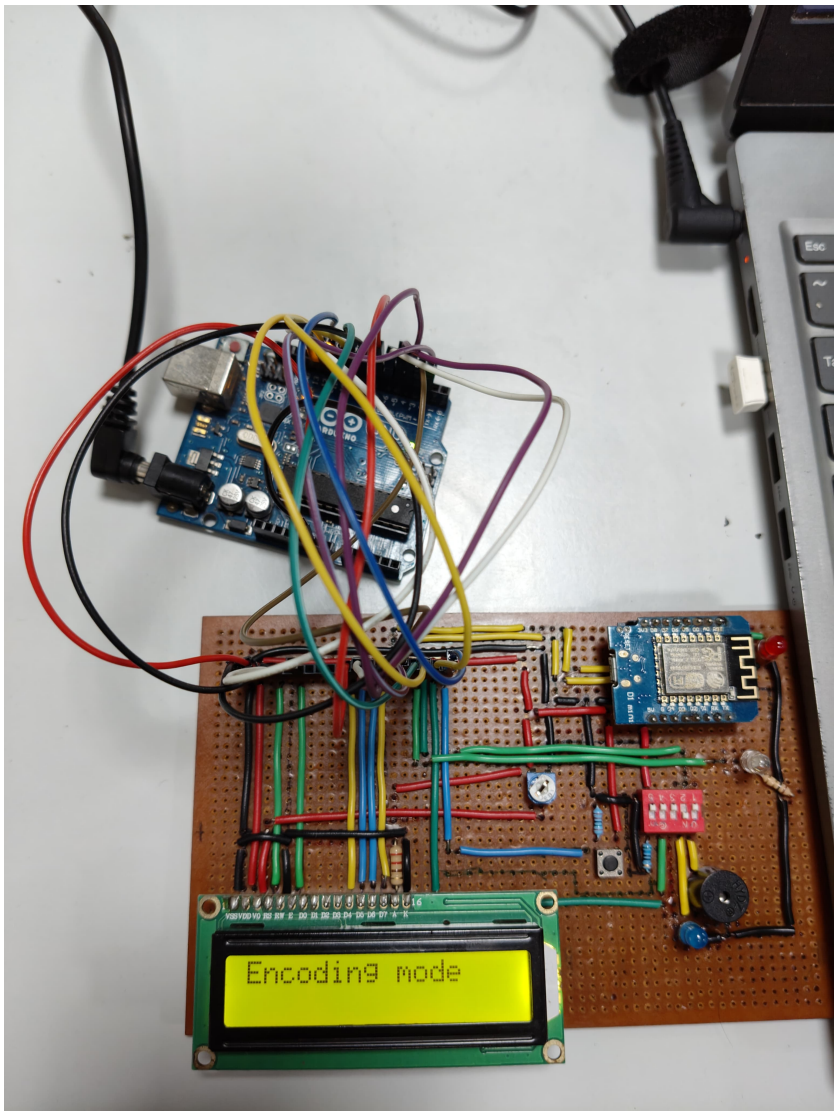

Figure 4: Project Photo

10 Bibliography

References

[1] http://www.steves-internet-guide.com/mqtt/

[2] https://nodered.org/docs/getting-started/

[3] https://science.howstuffworks.com/innovation/inventions/morse-code.htm

[4] https://forum.arduino.cc/t/serial-communication-between-esp8266-and-arduino/590972

[5] https://www.javatpoint.com/what-is-a-serial-port