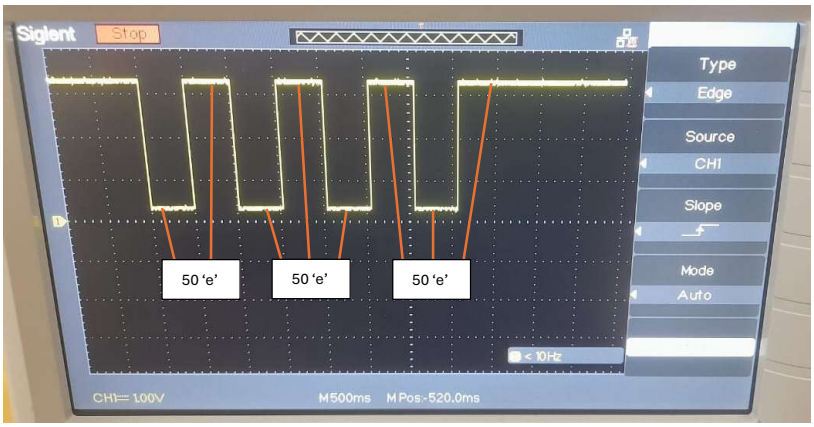
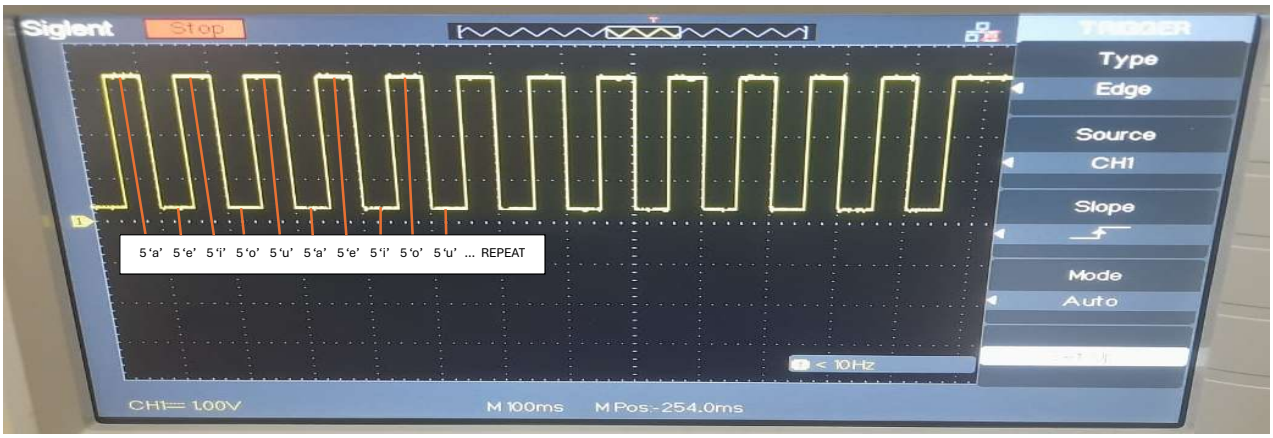
Test 1 – Repeated Character Debounce Validation:



To verify that the FSM correctly ignores repeated characters received within the 500ms debounce window, I used the keyboard paste function to input 400 consecutive 'e' characters into the serial terminal. This simulated a high-frequency input stream over approximately 4000ms. The expected behavior was that only the first 'e' would trigger a state transition, while subsequent 'e' characters arriving within the debounce interval would be ignored. Due to the 10ms ticks, the 500ms would elapse within 50 inputs of the same character.

Test 2 – Character Change Detection Within Debounce Window:



To test the FSM's ability to detect and respond to character changes occurring within the 500ms window, I pasted a mixed sequence of characters into the terminal: "aaaaaeeeeeiiiiiooooouuuuu" x 5
This input stream included frequent transitions between different characters, simulating rapid user input. The goal was to confirm that the FSM correctly resets the debounce timer on character change and allows valid transitions when a new character is received, even if the previous one was within the debounce period. Due to the 10ms tick, each character

group (eg 5 ‘a’) would take 50ms before the state transitions due to different character input.