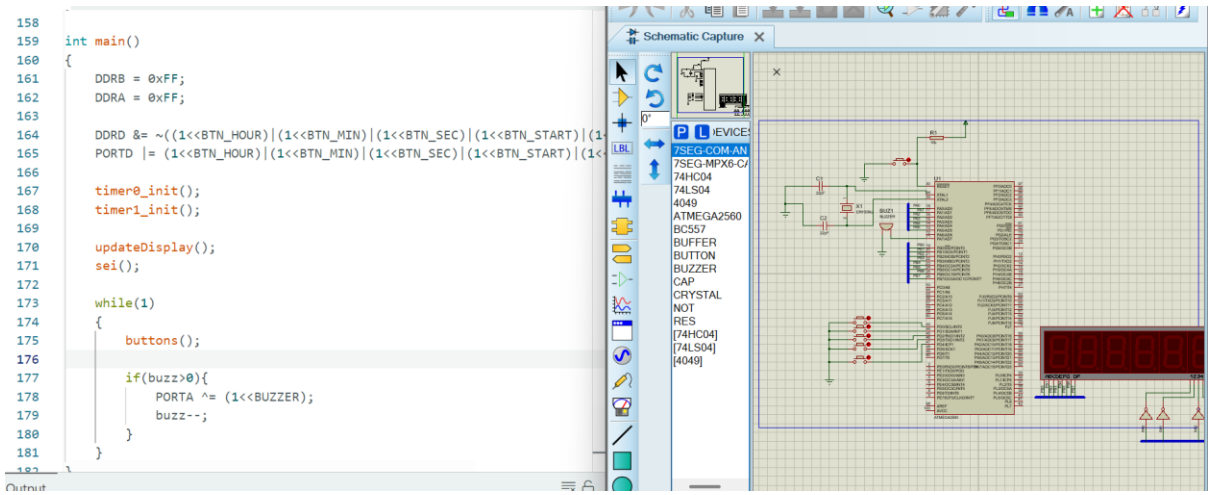
My Portfolio: Embedded Systems & Documentation

Project Overview: Dual-Timer System with LCD & Keypad

This project demonstrates a complex embedded solution developed for Arduino Mega 2560. The system features a custom user interface using a 4x4 Matrix Keypad and a 16x2 LCD Display, fully simulated and verified in Proteus.



Key Technical Features:

- Hardware: Arduino Mega 2560, LCD (I2C/Parallel), 4x4 Keypad, Piezo Buzzer.
- Interrupt Management: Utilizes TIMER5 hardware interrupts for precise timing and background tasks.
- Scanning Logic: High-efficiency row-column scanning for keypad input handling.
- UI/UX: Real-time countdown visualization and status updates on the LCD.

Code Translation & Documentation Sample

Below is an example of how I document and translate technical logic for international developers:

// --- Global Timer Configuration ---

// Setting up TIMER5 to trigger an interrupt every 100ms

```
// This ensures precise timing for the countdown without blocking the main loop.
void setupTimer5() {
  TCCR5A = 0;             // Set entire TCCR5A register to 0
  TCCR5B = 0;             // Same for TCCR5B
  TCNT5  = 0;             // Initialize counter value to 0
  OCR5A = 15624;          // Set compare match register (16MHz / 1024 / 1Hz - 1)
  TCCR5B |= (1 << WGM52); // Turn on CTC mode
  TCCR5B |= (1 << CS52) | (1 << CS50); // Set 1024 prescaler
  TIMSK5 |= (1 << OCIE5A); // Enable timer compare interrupt
}

// --- Keypad Scanning Logic ---
// Efficiently scans the 4x4 matrix to detect user input for T1/T2 settings.
char scanKeypad() {
  for (int r = 0; r < 4; r++) {
    digitalWrite(rowPins[r], LOW); // Activate current row
    for (int c = 0; c < 4; c++) {
      if (digitalRead(colPins[c]) == LOW) { // Check column state
        delay(20); // Debounce delay
        return keys[r][c];
      }
```

```
        }
        digitalWrite(rowPins[r], HIGH); // Deactivate row
    }
    return 0;
}
```

Why this matters for your project:

I provide clear, industry-standard documentation that allows other engineers to understand, maintain, and scale your code. Whether it's a README for GitHub or inline comments, I ensure your technical intent is never "lost in translation."