CS 350

Jose Munoz

1. **Why do you have a sleep command in your loop?**  
   The sleep call in the loop acts as a simple timing control: it pauses the program briefly between writes so that the LCD isn’t bombarded with updates too quickly. Without that delay, the display would flicker or update so fast that you couldn’t read the characters. By putting in a short pause often something like half a second you ensure each message stays on screen long enough to be legible before the next update runs.
2. **What is the purpose of having a text display on an embedded device?**  
   Adding a 16×2 character display transforms a “dumb” LED circuit into a simple user interface. Instead of just blinking or changing color, you can show real information: sensor readings, menu options, status messages, or even simple prompts. Text feedback helps both developers and end-users understand what the device is doing at any moment especially useful during debugging or when you want to convey multiple data points without hooking up a laptop.
3. **How can you think of the display device as something that could relate to a state machine?**  
   In many embedded designs, each screen or message corresponds to a particular system state like “Waiting for input” or “Processing data,” or “Error.” Transitioning from one message to other mirrors moving from one state to the next in a state machine. For example, when a button is pressed your code might transition from the “Idle” state (showing a welcome message) into a “Measure” state (showing sensor data), then back again. The LCD simply becomes the external window into which the state machine “prints” its current state.

**Brief Reflection**

Setting up the 16×2 LCD was smoother than I expected once all the jumper wires were in place, but I did hit a snag at first with the contrast the screen just showed solid blocks until I remembered to tweak the potentiometer. Adjusting that little dial until the characters popped into view took a minute, and I had to power-cycle the Pi after making the wiring changes to get everything recognized. Overall, the biggest takeaway was how important it is to double-check pin mappings misplacing a single wire meant I spent extra time debugging a “dead” display. Now that I’ve got it working, I feel more confident adding more complex UI elements in future modules.