Dylan Lozon

ECE 101-02 MATLAB and C Programming

Mr. Watchorn

May 16, 2023

Mr. Watchorn,

Executive Summary

I have created a finite state machine to control our new model of fan. It should be all but ready to distribute to distribute to our partners at the circuit printing company. I’ve coded an interactive demonstration of its functionality in MATLAB, attached to this email with the name FanStateMachine.mlx.

Discussion

See figure 1 on the next page for a diagram of the finite state machine intended for our new fans.

Note 1: The INIT and END state are essentially the same state when deployed to a real fan. This is because a real fan can be turned back on after being turned off unlike the demonstration. I chose to display them as distinctly separate start and end states largely for easy readability and understandability. This also means that while the script allows a user to jump straight from INIT to END, this is not on the diagram because you never really started the fan to begin with.

Note 2: The demonstration allows for the user to input invalid instructions. This is handled by not changing state, but I chose not to include this on the diagram because it is impossible on a real fan, leading to the diagram including irrelevant information.

Note 3: Because this is already a deterministic FSM, every possible action has a clearly defined result, and major optimizations will not be necessary.

Note 4: FanState.m is a class containing a static enumeration of each possible fan state and FanFSMInputParser.m is a class containing several utility functions used to parse user input.

Outcomes

The script’s execution perfectly matches the diagram in figure 1, with the caveat mentioned in note 2 above. The MATLAB project required 1 hour, 15 minutes, and 22 seconds of development time.

Conclusions

Hopefully this will be useful for demonstrating the unit’s function. Please contact me if you have any questions or concerns.

Best,

Dylan

FIGURE 1:

