Reflection: RSLK - Finite State Machines: Arnav Krishnan

1. I gained a greater appreciation for how each FSM state holds not just its outputs but also an array of pointers to the next states, clearly showing how the robot transitions between its different possibilities. This enables the main loop to stay much simpler.

2. One thing that worked well was using #define aliases like Center, LeftOff1, etc., which made the FSM table human-readable. There was a point where my only issue was a FSM table mismatch; debugging would have been significantly worse if I hadn't used aliases.

3. For the elevator, we used more conditional logic for FSM transitions, instead of an FSM table. I wonder when a table-based FSM is more useful/harder to use than explicit conditional logic, and whether one implementation has favorable performance.