# Event Driven State Machine Lab

**BME554L - Spring 2026 - Palmeri**

Dr. Mark Palmeri, M.D., Ph.D.

2026-01-05

## Table of contents



> **Tip**
>
> I would recommend doing this with pencil and paper first, not immediately jumping into a tool. Err on the side of more detail than not.

### Digital Kitchen Timer

Generate a state diagram for a simple digital kitchen timer that has the following buttons:

- `Start/Stop` Button
- `Reset` Button
- `Pause` Button
- `Up/Down/Left/Right` Arrow Buttons (4)

The timer outputs time intervals in the format: `HH:MM:SS`

### Functional States

- The kitchen timer has an "Idle" state when it isn't actively counting down time. In this state:
  - The `Up/Down` buttons increment the time for the `HH`, `MM` or `SS` values.

  - The `Left/Right` buttons let you move between `HH`, `MM` and `SS`.
  - Time intervals will wrap around to `00` when a reasonable max value is exceeded.
  - The `Reset` button sets all of the time values to `00`.
  - The `Pause` button is inactive.

- `Start/Stop` will start the timer countdown when in `Idle`.
- In the `Countdown` state:
  - The `Start/Stop` button will stop and reset the countdown interval.
  - The `Pause` button will pause, but not reset, the countdown time.
- When the countdown is done, the timer enters an `Alarm` state, where:
  - The `Up/Down/Right/Left` arrow buttons, `Reset` and `Pause` buttons are inactive.
  - `Start/Stop` will stop the alarm and return the timer to `Idle` with the previously set countdown time.

> **Tip**
>
> If any functional state expectation isn't explicitly described above, please implement something reasonable.

Make sure that your state diagram:

- Has initialization (i.e., batteries put in) and termination (i.e., completely powered off) dots.
- Includes all functional states, including (but not limited to):
  - `Idle`
  - `Countdown`
  - `Paused`
  - `Alarm`
- Consider if states should have entry and exit actions.
- Make sure all transitions between states are annotated with associated triggering events.

### Generate the State Diagram

Formally render your state diagram using one of the tools outlined on the State Diagram Tools page.

> **Tip**
>
> For PlantUML-generated digrams, if the diagram's layout is difficult to follow using the automated layout, consider manually specifying relative positions and

> length of arrows using the following annotation: https://crashedmind.github.io/PlantUMLHitchhikersGuide/layout/layout.html#:~:text=6.,Tip.

## Gradescope

Upload your rendered state diagram to the associated Gradescope assignment.