# The Marauder Map

## EE 249 Fall 2014

LongXiang Cui, Ying Ou, Jordan Makansi, Chenyang Xu

Dec 18th 2014

# Project motivation

Information of occupancy of room is important

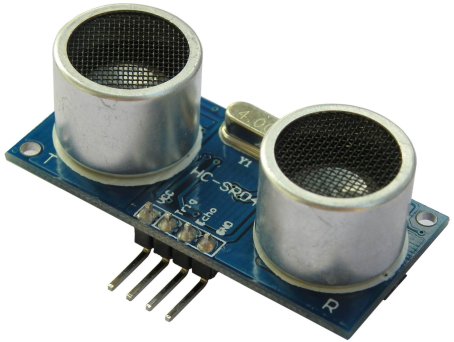
- Energy saving
- Convenience reasons

# Project Overview

Two main components:

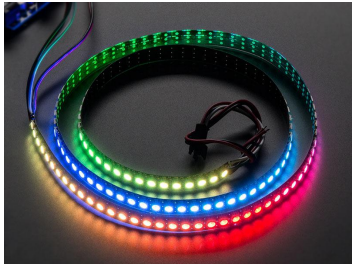
- Counter recording occupancy of the room
- Transfer the results to a remote server page

# Hardware



Human detection

Additional Feature

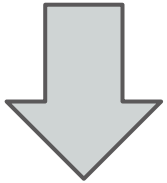


Server and PC

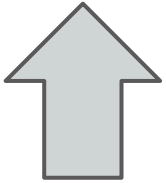
WIFI module

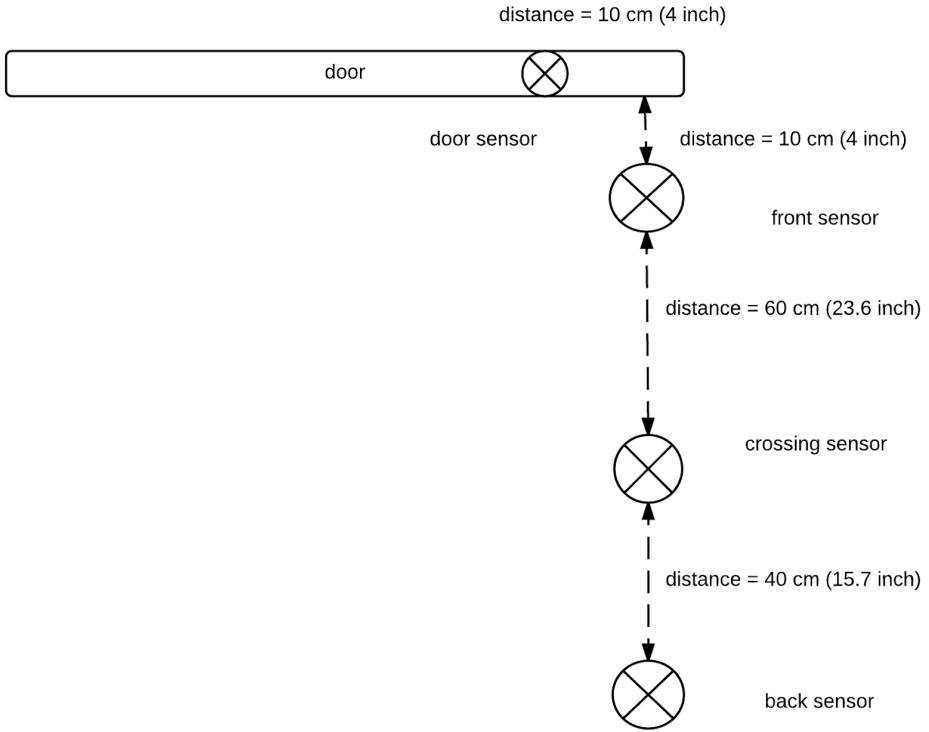
# Detection method

Going Out

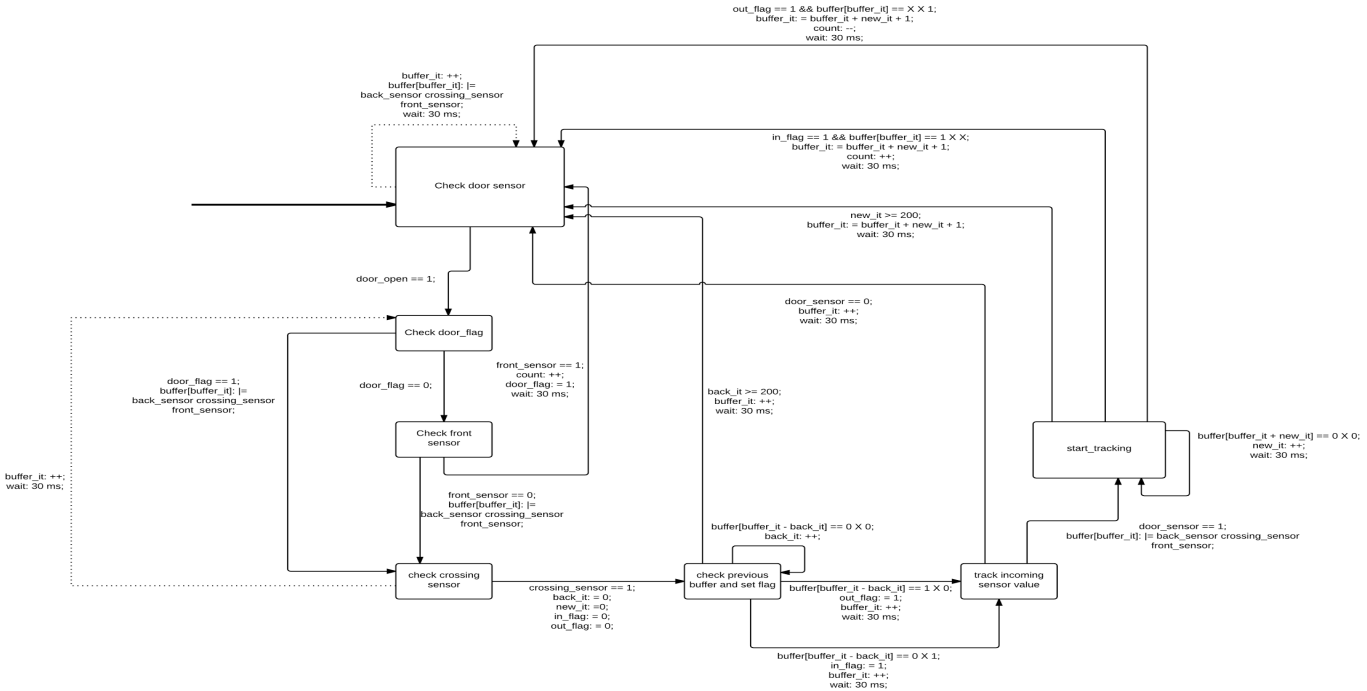


Coming In

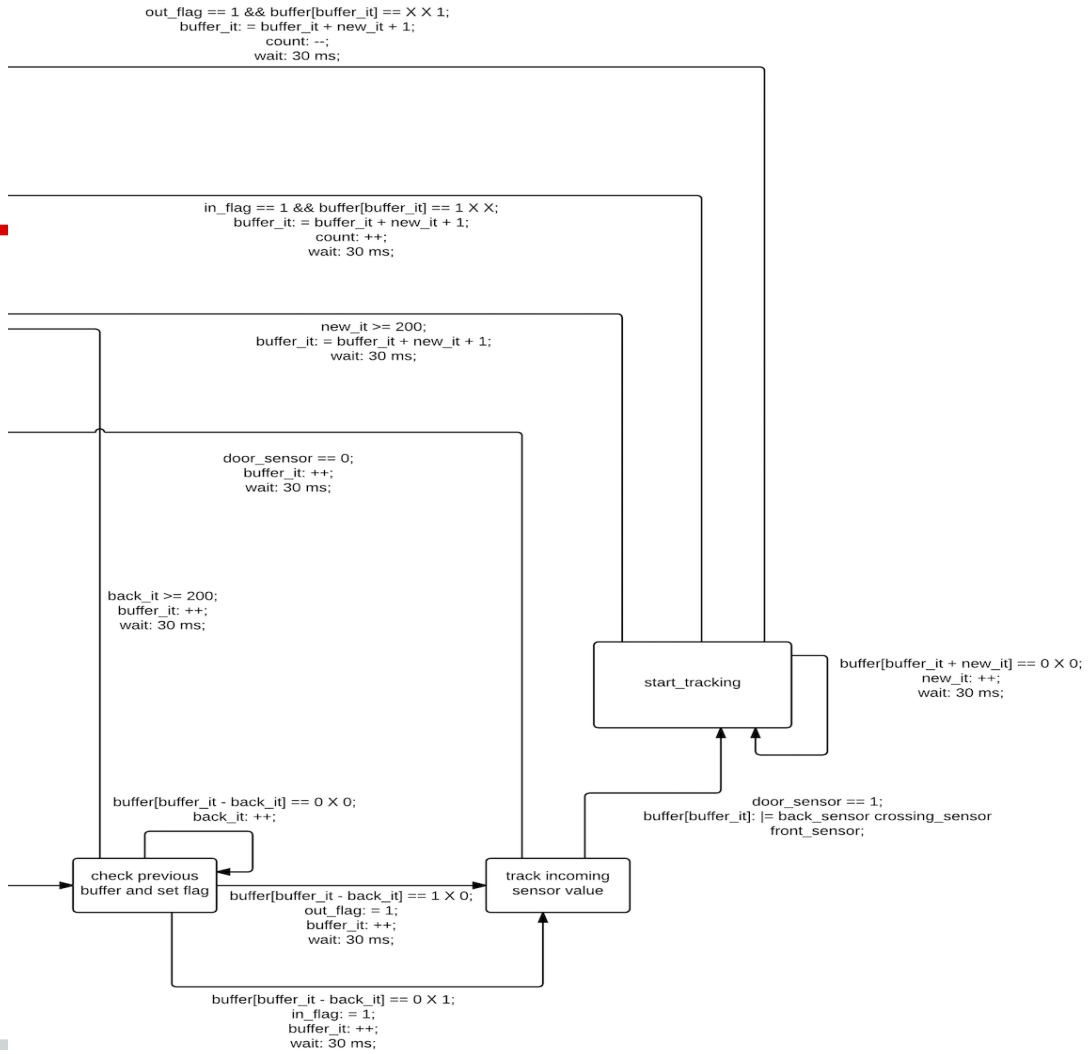




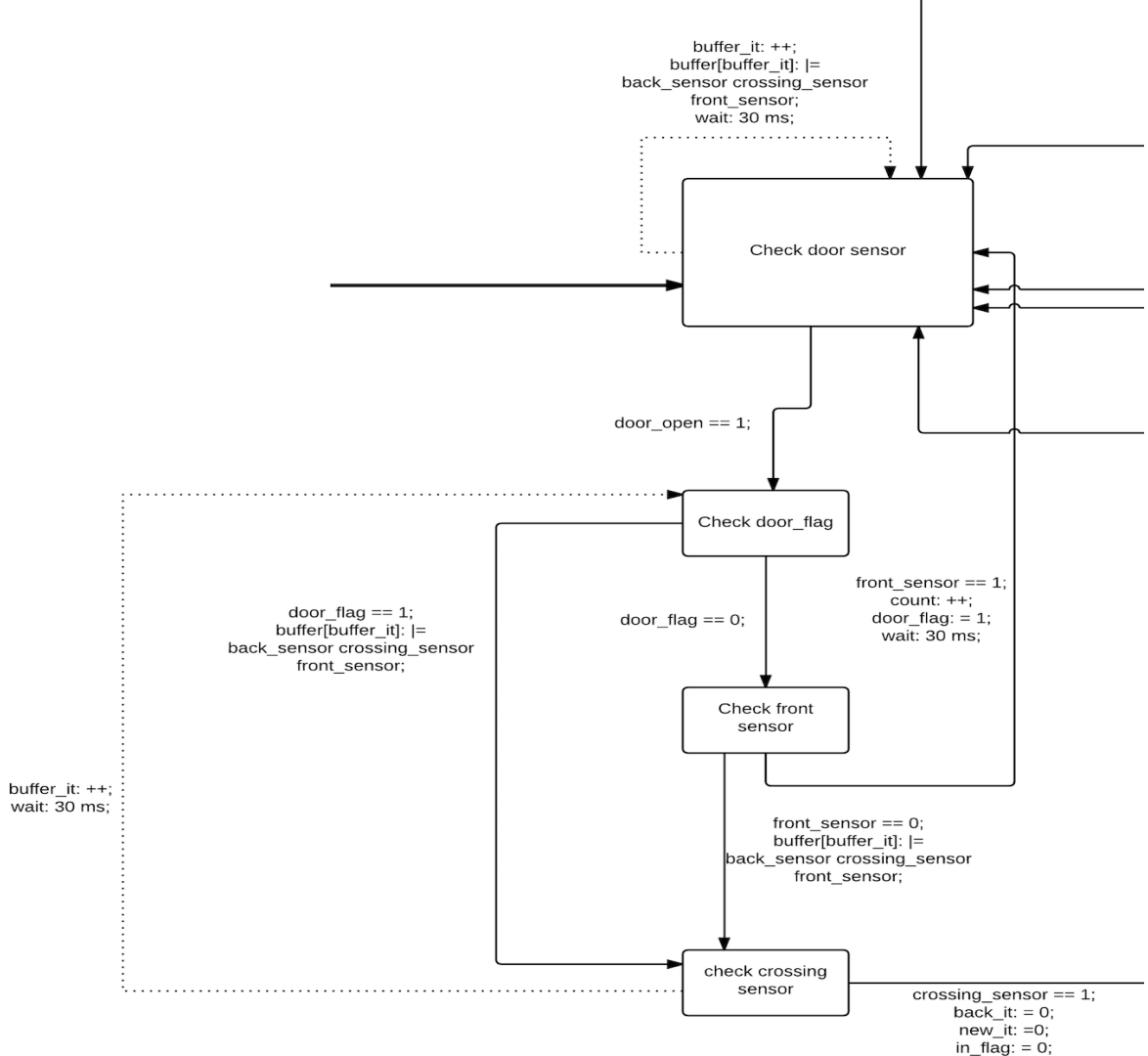
# Finite State Machine



# Hole Part


out_flag == 1 && buffer[buffer_it] == X X 1;
buffer_it: = buffer_it + new_it + 1;
count: --;
wait: 30 ms;
in_flag == 1 && buffer[buffer_it] == 1 X X;
buffer_it: = buffer_it + new_it + 1;
count: ++;
wait: 30 ms;
new_it >= 200;
buffer_it: = buffer_it + new_it + 1;
wait: 30 ms;
door_sensor == 0;
buffer_it: ++;
wait: 30 ms;
back_it >= 200;
buffer_it: ++;
wait: 30 ms;
start_tracking
buffer[buffer_it + new_it] == 0 X 0;
new_it: ++;
wait: 30 ms;
buffer[buffer_it - back_it] == 0 X 0;
back_it: ++;
door_sensor == 1;
buffer[buffer_it]: |= back_sensor crossing_sensor
front_sensor;
check previous
buffer and set flag
buffer[buffer_it - back_it] == 1 X 0;
out_flag: = 1;
buffer_it: ++;
wait: 30 ms;
track incoming
sensor value
buffer[buffer_it - back_it] == 0 X 1;
in_flag: = 1;
buffer_it: ++;
wait: 30 ms;

# Door Part


buffer_it: ++;
buffer[buffer_it]: |=
back_sensor crossing_sensor
front_sensor;
wait: 30 ms;
Check door sensor
door_open == 1;
Check door_flag
door_flag == 1;
buffer[buffer_it]: |=
back_sensor crossing_sensor
front_sensor;
door_flag == 0;
front_sensor == 1;
count: ++;
door_flag: = 1;
wait: 30 ms;
Check front
sensor
buffer_it: ++;
wait: 30 ms;
front_sensor == 0;
buffer[buffer_it]: |=
back_sensor crossing_sensor
front_sensor;
check crossing
sensor
crossing_sensor == 1;
back_it: = 0;
new_it: =0;
in_flag: = 0;

# Demo Video

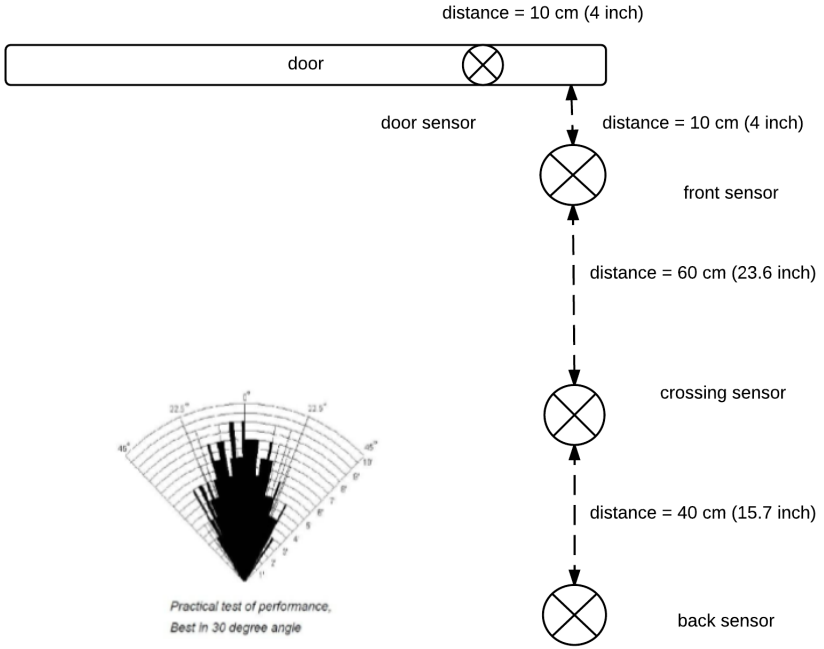
https://www.youtube.com/watch?v=ZzLX6iWxS8M

# Design Considerations

Sensor distance:

- At least 45.5 cm apart to not interference with each other for 92cm door

- As close as possible to allow counting people following each other



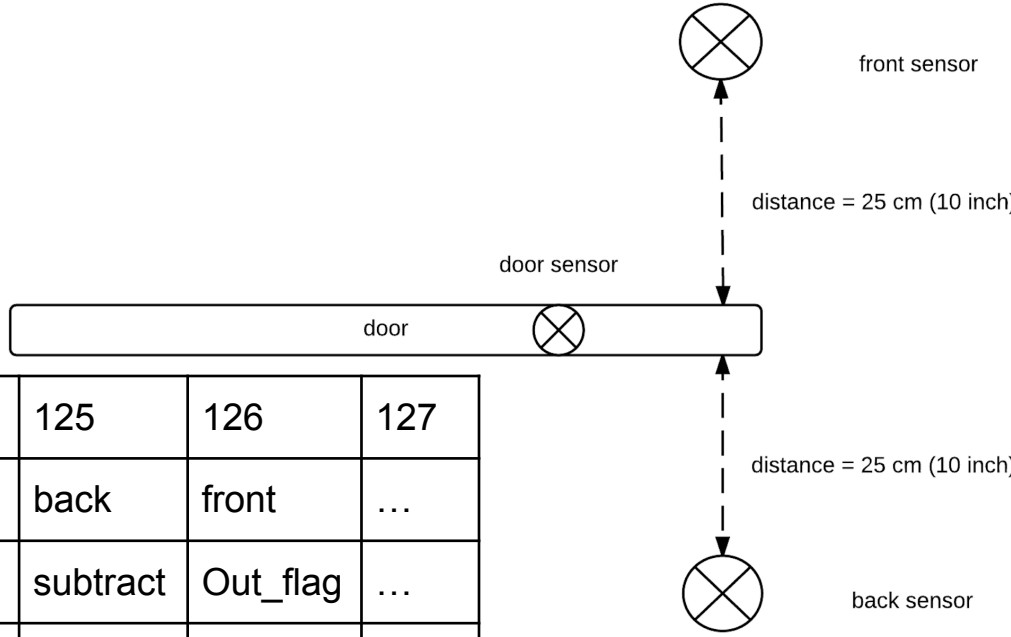



*Practical test of performance, Best in 30 degree angle*

# First Design

Two-sensor:

- Pro:
  - Simple
  - Fast for detection
- Con:
  - Not stable
    - 2 people coming in

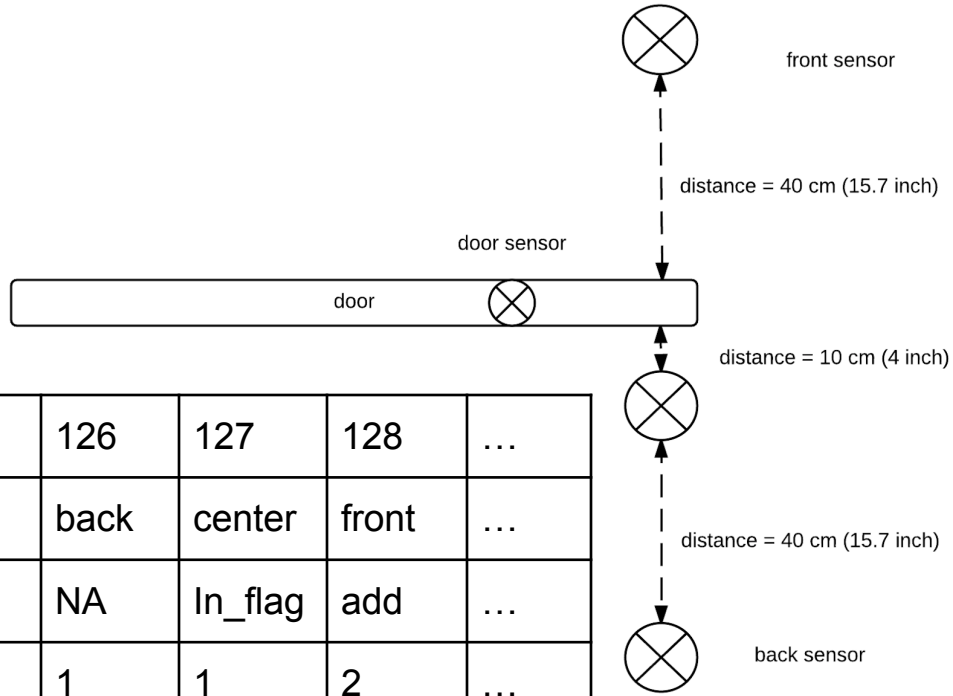



| Buf # | ... | 121 | 122 | 123 | 124 | 125 | 126 | 127 |
|---|---|---|---|---|---|---|---|---|
| Buf val | … | NA | back | front | front | back | front | … |
| behavior | … | NA | In_flag | add | Out_flag | subtract | Out_flag | … |
| count | … | 0 | 0 | 1 | 1 | 0 | 0 | … |

# Second Design

Three-sensor:

- Pro:
  - More accurate
  - Save power
- Con:
  - Confliction with door
    - 2 people coming in



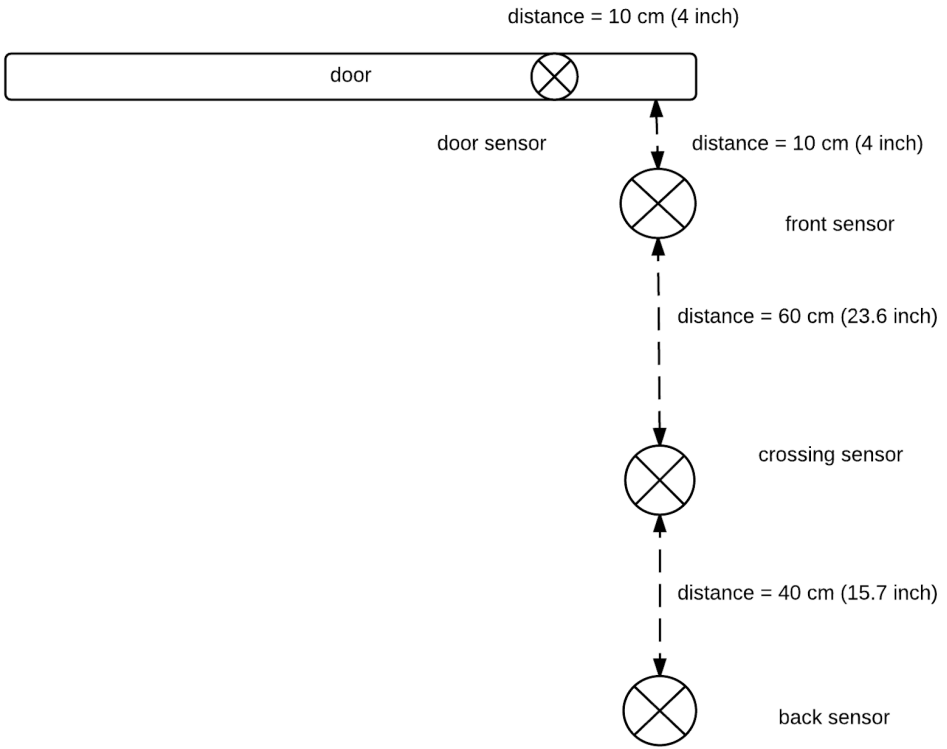

| Buf # | ... | 121 | 122 | 123 | 124 | 125 | 126 | 127 | 128 | … |
|---|---|---|---|---|---|---|---|---|---|---|
| Buf val | … | NA | back | center | front | front | back | center | front | … |
| behavior | … | NA | NA | In_flag | add | NA | NA | In_flag | add | … |
| count | … | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 2 | … |

# Third Design

Three-sensor one side:

- Pro:
  - Accurate
  - Could detect the door

- Con:
  - Complicate
  - Power consumption issue



# Limitation

Wide angle sensor

- Intrusive setting
- People cannot following each other too closely
- Setting need tweaking for each different environment

# Future Developments

Accuracy

- More accurate ultrasonic sensors or different type of sensors
- More stable algorithm to detect edge cases

System Integration

- Light and temperature control
- Not prone to hysteresis

Scale

- Fast and easy system setting up

# Thank you!

Special thanks to:

Ben Zhang

Antonio Ianopollo

Prof. Edward A. Lee