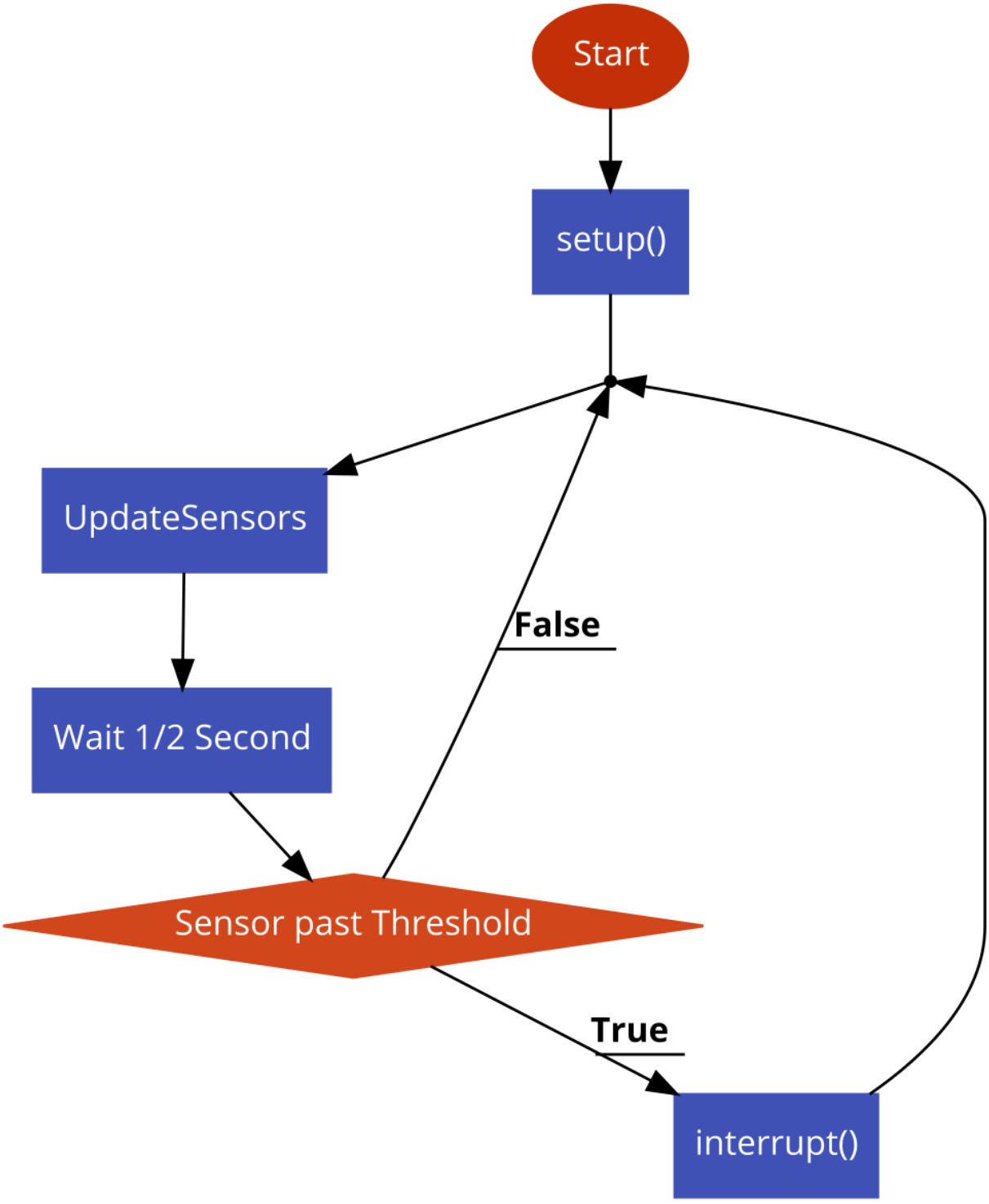
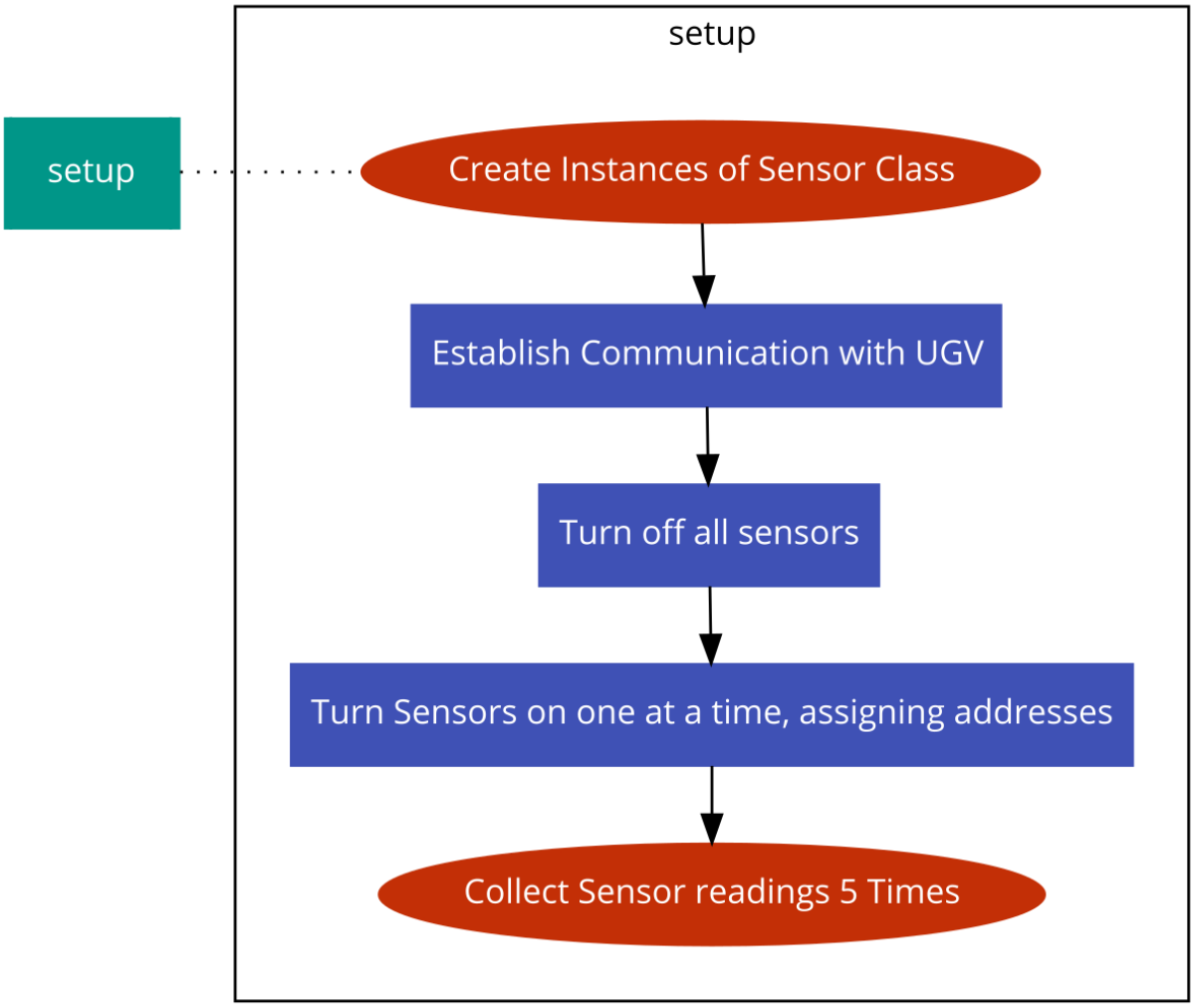
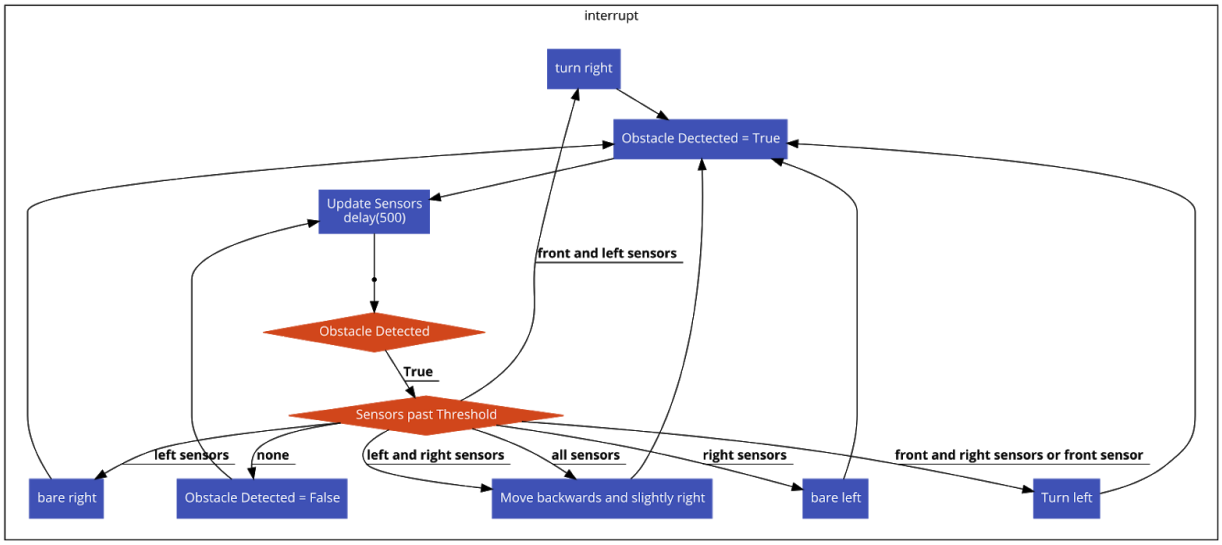
Criterion B: Design

Diagrams




setup
setup
Create Instances of Sensor Class
Establish Communication with UGV
Turn off all sensors
Turn Sensors on one at a time, assigning addresses
Collect Sensor readings 5 Times


interrupt
turn right
Obstacle Dectected = True
Update Sensors
delay(500)
front and left sensors
Obstacle Detected
True
Sensors past Threshold
left sensors
none
left and right sensors
all sensors
right sensors
front and right sensors or front sensor
bare right
Obstacle Detected = False
Move backwards and slightly right
bare left
Turn left

Success Criteria

Test Type	Nature of Test	Example
The system can read distance from the sensors	Check that change in distance is measured.	Placing hand in front of sensor alerts a change in the console.
An algorithm can determine whether it is more likely that the sensor reading is a false-positive or an actual obstacle to avoid.	Simulate a false-positive obstacle quickly, or have the algorithm operate properly on the UGV	Moving UGV with the system active, and then towards an object. The system should not alert the UGV only when there is an object
The system can determine approximately where the obstacle is located with respect to the robot.	Algorithm knows where each sensor is located on the robot to roughly determine where the obstacle it is detecting is.	Upon sensing an obstacle to the right of the robot, the Arduino alerts the UGV that the obstacle is on the right.
The system can communicate to the main computer about what actions to take if an obstacle is detected.	The Arduino sends messages to the computer on the UGV.	The main computer will move based on the established protocol with Arduino.