**White Box**

**mpu6050\_read\_raw**

18 – 17 + 2 = 3

Path 1: 1-2-3-4-5-6-7-6-8-9-10-11-12-11-13-14-15-16-17

Path 2: 1-2-3-4-5-6-7-6-8-9-10-11-13-14-15-16-17

Path 3: 1-2-3-4-5-6-8-9-10-11-13-14-15-16-17

**getAcceloDetection**

13 – 14 + 2 = 1

**getHump**

23 – 21 + 2 = 4

Path 1: 1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-16-21

Path 2: 1-2-3-4-5-6-7-8-9-10-11-17-18-19-20-21

Path 3: 1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-21

Path 4: 1-2-3-4-5-6-7-8-9-11-12-13-14-15-21

|  |  |  |  |
| --- | --- | --- | --- |
| Test Case | Height | Count | Outcome |
| 1 | x >= 0.7 | x > 15 | Hump detected |
| 2 | x >= 0.7 | x <= 15 | No hump |
| 3 | x < 0.7 | x > 15 | - |
| 4 | x < 0.7 | x <= 15 | Reset values |

**setupUltrasonicPins**

6 – 7 + 2 = 1

**getPulse**

21 – 20 + 2 = 3

Path 1: 1-2-3-4-5-6-7-8-9-10-11-12-9-10-11-12-13-14-15-16-17-14-15-16-17-18-19-20

Path 2: 1-2-3-4-5-6-7-8-9-10-11-12-9-10-11-12-13-14-15-16-17-18-19-20

Path 3: 1-2-3-4-5-6-7-8-9-10-11-12-13-14-15-16-17-18-19-20

**getCm**

3 – 4 + 1 = 1

**getUSDetection**

15 – 15 + 2 = 2

Path 1: 1-2-3-4-5-6-7-8-5-9-10-11-12-13-14-15

Path 2: 1-2-3-4-5-9-10-11-12-13-14-15

**Black Box**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Cause |  | value | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| C1 | Avg\_front < 5.5 | Y/N | N | N | N | N | Y | Y | Y | Y |
| C2 | Avg\_left < 5.5 | Y/N | N | N | Y | Y | N | N | Y | Y |
| C3 | Avg\_right  <5.5 | Y/N | N | Y | N | Y | N | Y | N | Y |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Effect |  |  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| E1 | Move forward |  | x | x | x | x |  |  |  |  |
| E2 | Move left |  |  |  |  |  | x | x |  |  |
| E3 | Move right |  |  |  |  |  |  |  | x |  |
| E4 | Reverse |  |  |  |  |  |  |  |  | x |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Cause |  | value | 1-4 | 5-6 | 7 | 8 |
| C1 | Avg\_front < 5.5 | Y/N | N | Y | Y | Y |
| C2 | Avg\_left < 5.5 | Y/N | - | N | Y | Y |
| C3 | Avg\_right  <5.5 | Y/N | - | - | N | Y |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Effect |  |  | 1-4 | 5-6 | 7 | 8 |
| E1 | Move forward |  | x |  |  |  |
| E2 | Turn left |  |  | x |  |  |
| E3 | Turn right |  |  |  | X |  |
| E4 | Reverse |  |  |  |  | X |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario** | **Test case** | **Pre-condition** | **Test Steps** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| Check block base on distance | Check Response for front, left, right > 5.5 | Car must have started | 1. Car moving 2. Ultrasonic sends pulse 3. Ultrasonic gets back pulse 4. Code calculates distance | front: 5.6  left: 5.6  right: 5.6 | Move forward | Move forward | Pass |
| Check block base on distance | Check Response for front, left, right < 5.5 | Car must have started | 1. Car moving 2. Ultrasonic sends pulse 3. Ultrasonic gets back pulse 4. Code calculates distance | front: 5.4  left: 5.4  right: 5.4 | reverse | reverse | Pass |
| Check block base on distance | Check Response for front < 5.5, left, right > 5.5 | Car must have started | 1. Car moving 2. Ultrasonic sends pulse 3. Ultrasonic gets back pulse   Code calculates distance | front: 5.4  left: 5.6  right: 5.6 | Turn left | Turn left | Pass |
| Check block base on distance | Check Response for front, left < 5.5, right > 5.5 | Car must have started | 1. Car moving 2. Ultrasonic sends pulse 3. Ultrasonic gets back pulse 4. Code calculates distance | front: 5.4  left: 5.4  right: 5.6 | Turn right | Turn right | Pass |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Test Scenario** | **Test case** | **Pre-condition** | **Test Steps** | **Test Data** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| Check if there is hump | Check Response from accelerometer where height > 0.7 | Car must be moving | 1. Car moving 2. Accelerometer gets x, y value 3. Code calculates height 4. Filter off any height below 0.7 5. Get height > 0.7 more than 3 times | Height: 0.7,0.8,0.7 | Display hump detected  & height of hump | Display hump detected, height = 0.73 | Pass |
| Check if there is hump | Check Response from accelerometer where height < 0.7 | Car must be moving | 1. Car moving 2. Accelerometer gets x, y value 3. Code calculates height 4. Filter off any height below 0.7 5. Get height < 0.7 | Height: 0.5 | Display no hump detected & height = 0 | Display no hump detected & height = 0 | Pass |