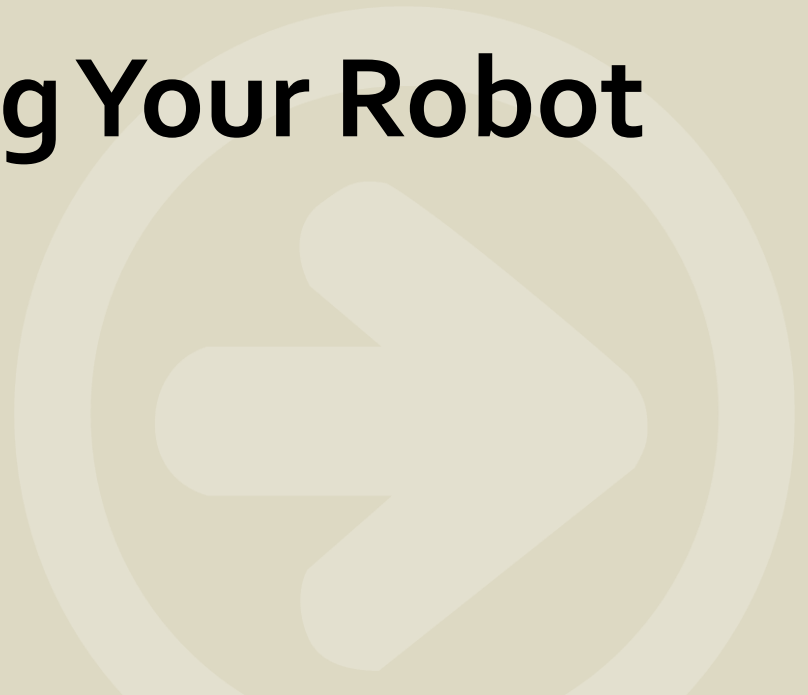




Lab 3. Navigating Your Robot





We have learned ...

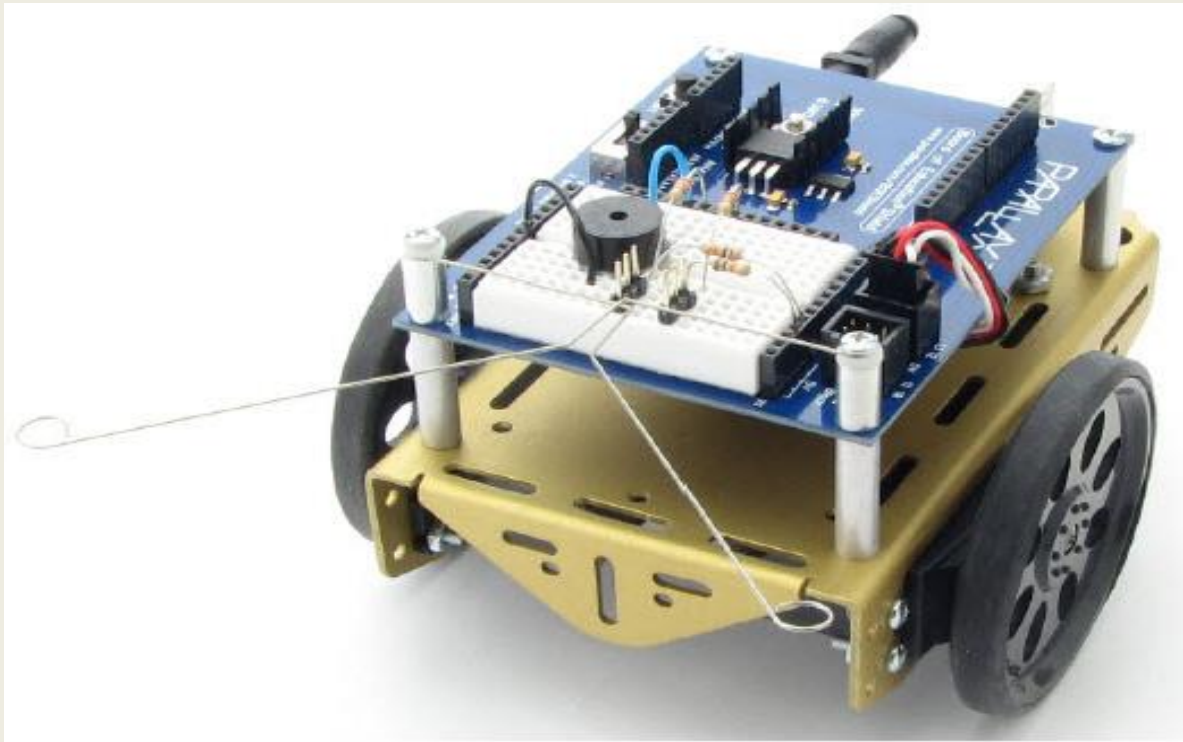


- ❑ Parallax Boe-bot
 - ❑ Attach servos
 - ❑ Move forward or backward
 - ❑ Turn left or right
 - ❑ Stop



In this Lab, you will learn ...

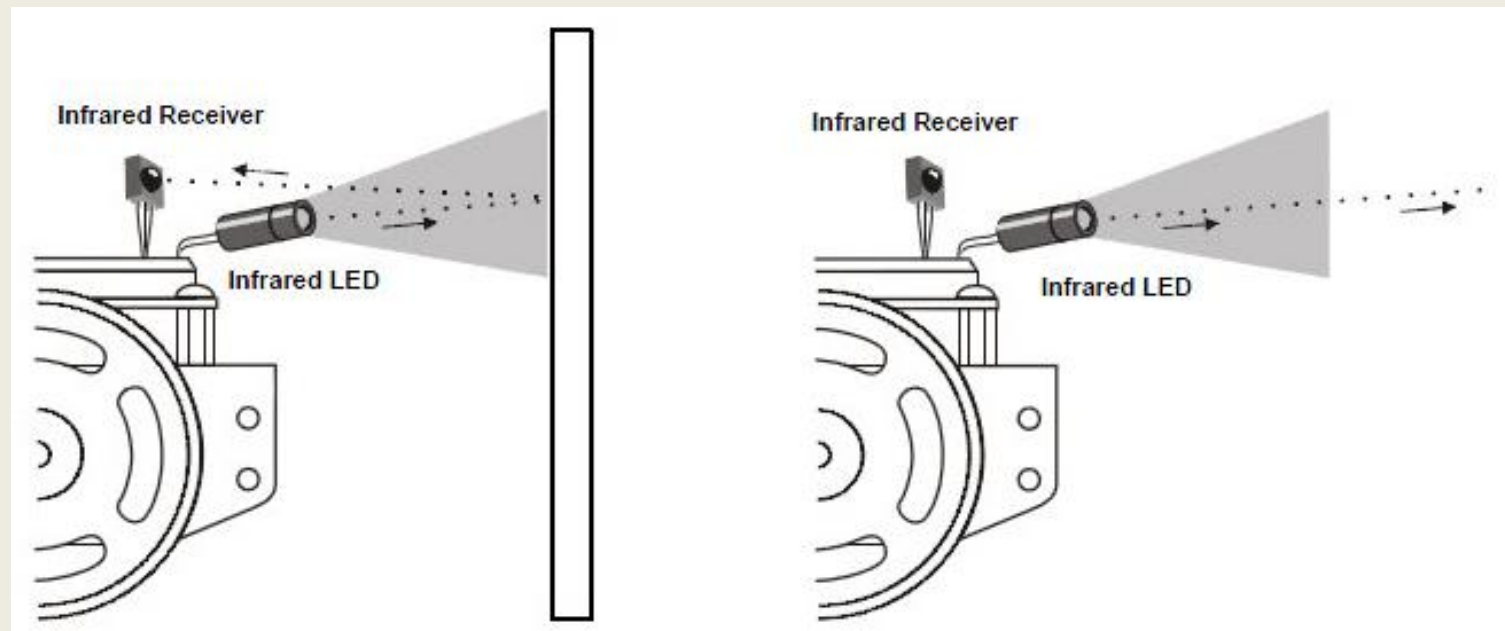
- ❑ Navigating with whisker switches





In this Lab, you will learn ...

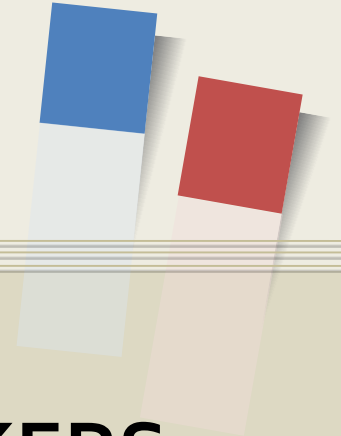
- ❑ Obstacle avoidance with InfraRed (IR) sensor



Infrared reflected, obstacle detected

Infrared not reflected, no obstacle detected

CS310 Embedded Computer System
Computer Architecture Lab.



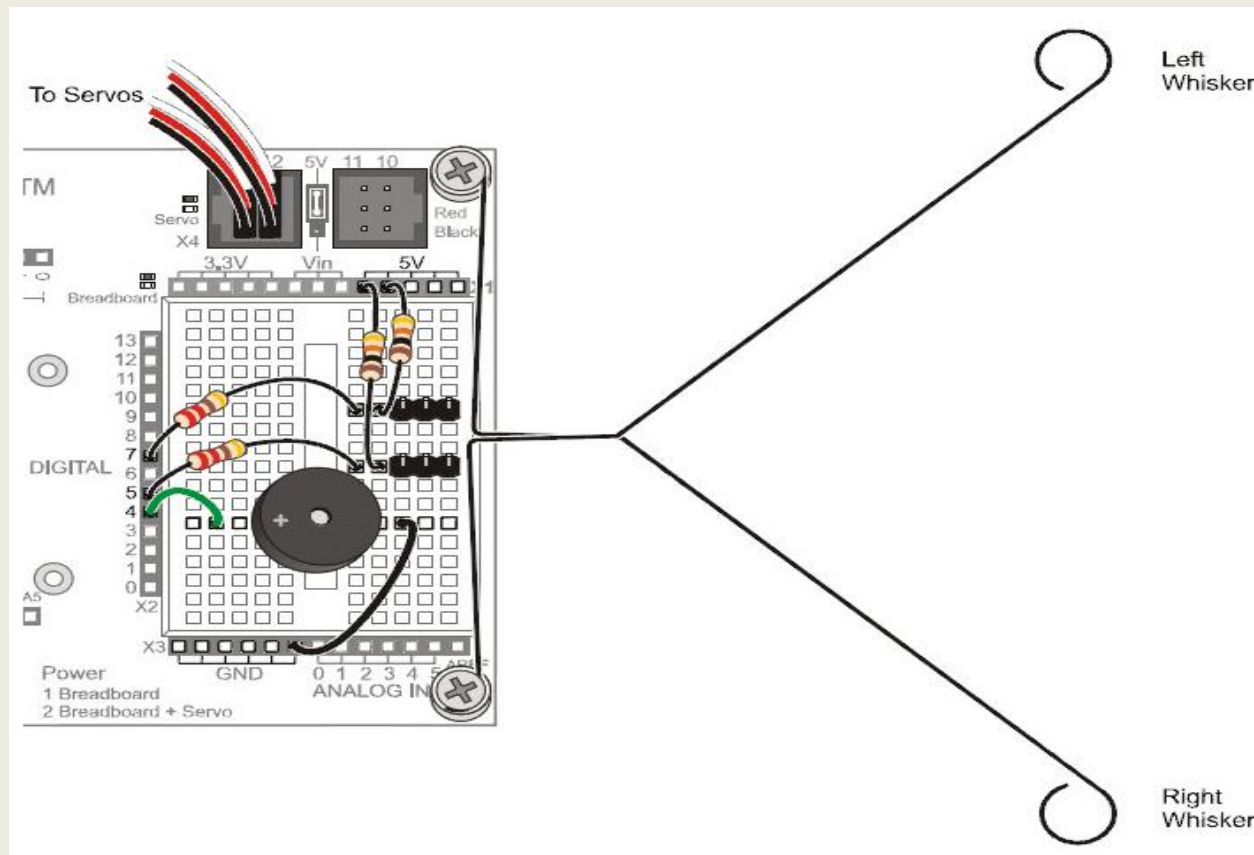
BUILD AND TEST THE WHISKERS



Build the Whiskers

❑ Refer below URL

❑ <http://learn.parallax.com/node/236>





Test Whiskers [1/2]



```
❑ void setup()                // Built-in initialization block
❑ {
❑   tone(4, 3000, 1000);      // Play tone for 1 second
❑   delay(1000);              // Delay to finish tone
❑   pinMode(7, INPUT);        // Set right whisker pin to input
❑   pinMode(5, INPUT);        // Set left whisker pin to input

❑   Serial.begin(9600);       // Set data rate to 9600 bps
❑ }
❑
❑ void loop()                 // Main loop auto-repeats
❑ {
❑   byte wLeft = digitalRead(5); // Copy left result to wLeft
❑   byte wRight = digitalRead(7); // Copy right result to wRight

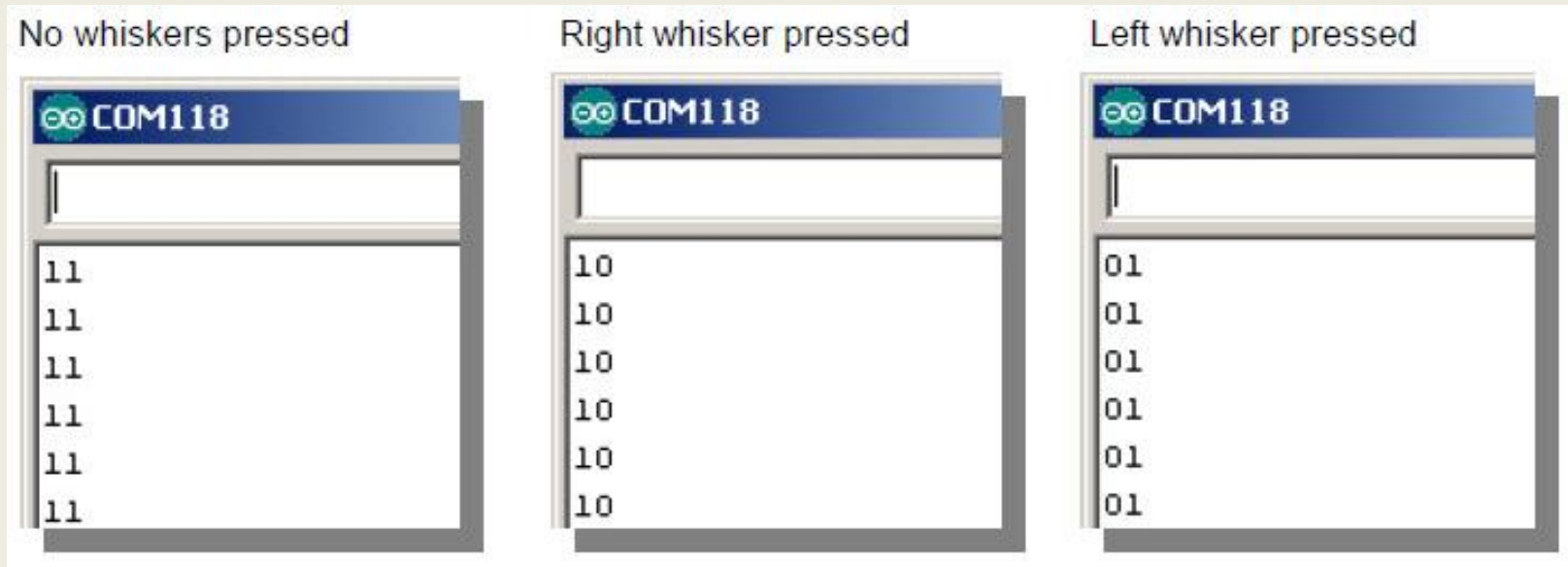
❑   Serial.print(wLeft);      // Display left whisker state
❑   Serial.println(wRight);   // Display right whisker state

❑   delay(50);                // Pause for 50 ms
❑ }
```



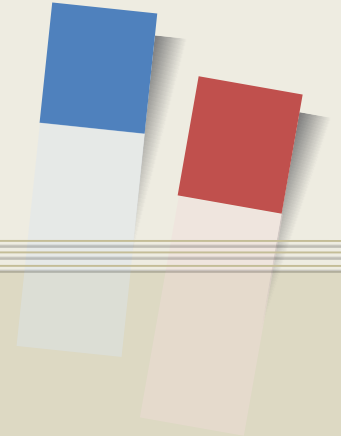
Test Whiskers [2/2]

- ❑ On the serial monitor ...



- ❑ Return 1, when not pressed
- ❑ Return 0, when pressed

CS310 Embedded Computer System
Computer Architecture Lab.



EXAMPLE #1



It's time to move!

- ❑ We know that
 - ❑ `wLeft == 0`, when left whisker touched
 - ❑ `wRight == 0`, when right whisker touched

- ❑ Make movements as followings:
 - ❑ `wLeft == 0`, Backward for 1 second, Turn right
 - ❑ `wRight == 0`, Backward for 1 second, Turn left
 - ❑ Both `== 0`, Backward for 1 second, Turn left or right



Demo





Code Snippet

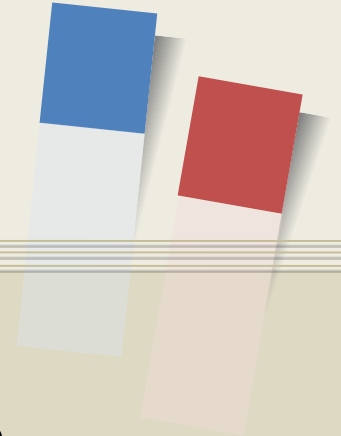


```
❑ void setup() {  
❑   // attach servos and set pinMode  
❑ }  
❑  
❑ void loop() {  
❑   byte wLeft = digitalRead(5);  
❑   byte wRight = digitalRead(7);  
  
❑   if (wLeft == 0) {  
❑       // Backward for 1 second and turn right  
❑   } else if (wRight == 0) {  
❑       // Backward for 1 second and turn left  
❑   } else if ((wLeft == 0) && (wRight == 0)) {  
❑       // Backward for 1 second and turn left or right  
❑   } else {  
❑       // Forward for a short period of time (e.g. 20ms)  
❑   }  
❑ }
```

CS310 Embedded Computer System
Computer Architecture Lab.



BUILD AND TEST IR SENSOR





-



Test your IR Sensor

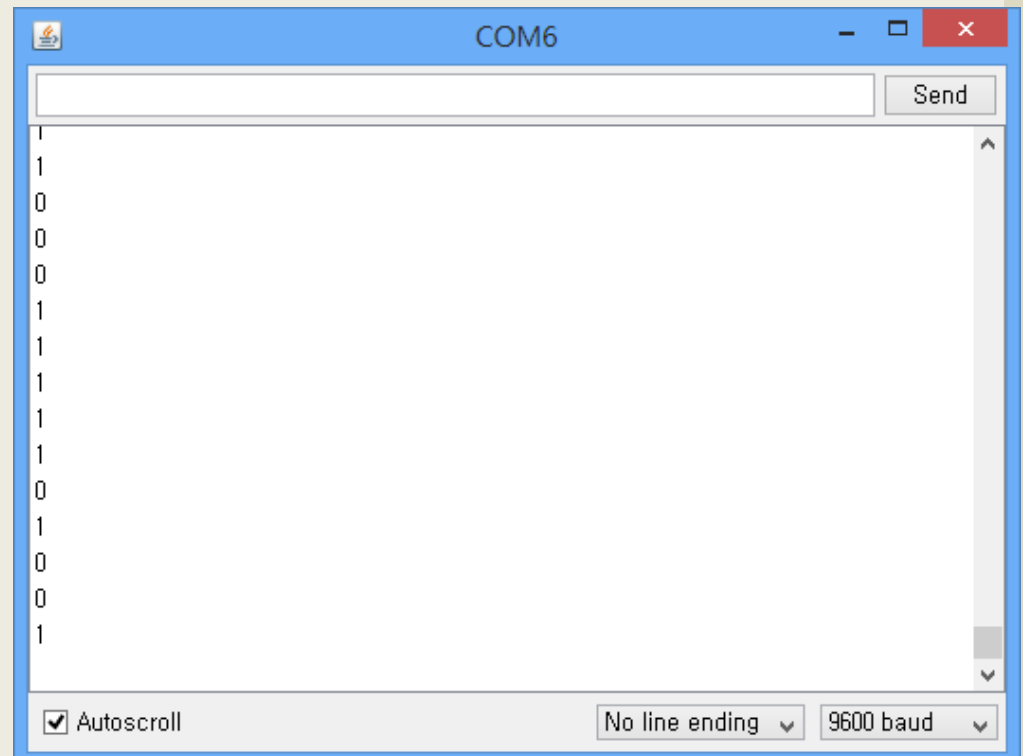
```
❑ void setup() {                                // Built-in initialization block
❑   pinMode(10, INPUT);                          // Left IR Receiver
❑   pinMode(9, OUTPUT);                         // Left IR LED
❑ }
❑ void loop() {
❑   int irLeft = irDetect (9, 10, 38000); // Check for object
❑   // Do something
❑ }

❑ // IR Object Detection Function
❑ int irDetect (int irLedPin, int irReceiverPin, long frequency) {
❑   tone(irLedPin, frequency, 8);    // IRLED 38 kHz for at least 1 ms
❑   delay(1);                        // Wait 1 ms
❑   int ir = digitalRead(irReceiverPin); // IR receiver -> ir variable
❑   delay(1);                        // Down time before recheck
❑   return ir;                       // Return 1 not detected, or 0 detected
❑ }
```



Test your IR Sensor [2/2]

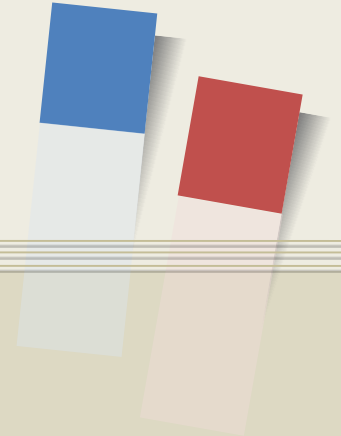
- ☐ When objects detected
 - ☐ Return 0 (zero)
- ☐ When not detected
 - ☐ Return 1 (one)
- ☐ If it prints all 1s,
 - ☐ Turn on the switch to '1'



CS310 Embedded Computer System
Computer Architecture Lab.



EXAMPLE #2





It's time to move!

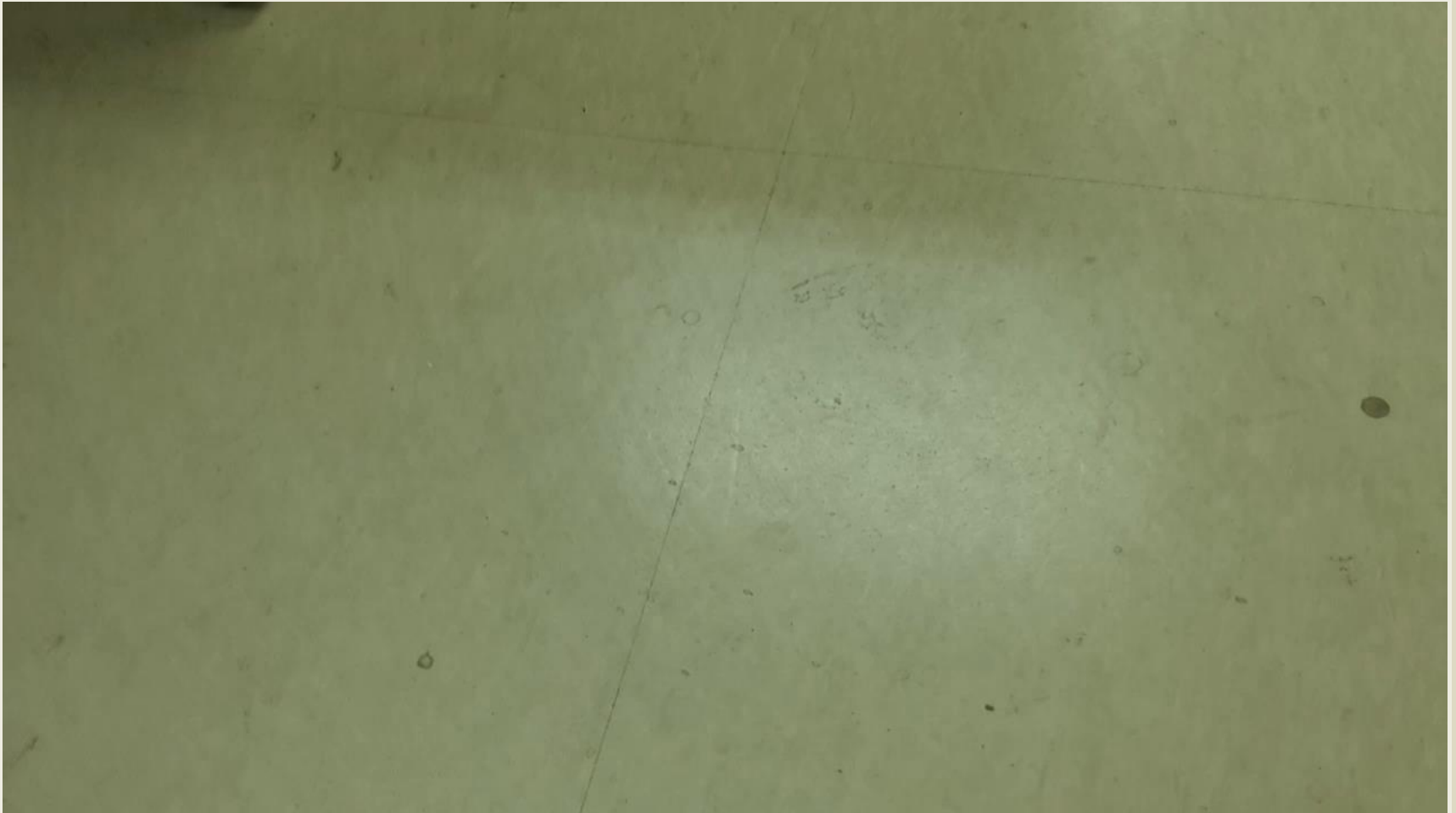


- ❑ When IR sensor detects an object,
 - ❑ Return 0
 - ❑ Otherwise, return 1

- ❑ Make movements:
 - ❑ `irLeft == 0`, backup 1 second & turn right about 60 degree
 - ❑ `irRight == 0`, backup 1 second & turn left about 60 degree
 - ❑ Both `== 0`, backup 1 second and turn about 120 degree



Demo





Code Snippet



```
❑ void setup() {                                // Built-in initialization block
❑   pinMode(10, INPUT); pinMode(9, OUTPUT);      // Left IR LED & Receiver
❑   pinMode(3, INPUT); pinMode(2, OUTPUT);       // Right IR LED & Receiver
❑   ...
❑ }
❑
❑ void loop() {                                  // Main loop auto-repeats
❑   int irLeft = irDetect(9, 10, 38000);         // Check for object on left
❑   int irRight = irDetect(2, 3, 38000);         // Check for object on right
❑
❑   if((irLeft == 0) && (irRight == 0)) {        // If both sides detect
❑     // Back up 1 second and turn left about 120 degrees
❑   } else if(irLeft == 0) {                     // If only left side detects
❑     // Back up 1 second and turn right about 60 degrees
❑   } else if(irRight == 0) {                   // If only right side detects
❑     // Back up 1 second and turn left about 60 degrees
❑   } else {                                     // Otherwise, no whisker contact
❑     // Forward 1/50 of a second
❑   }
❑ }
```



Caution

- ❑ If it cannot detect anything,
 - ❑ Make sure the power switch is on 1 or 2

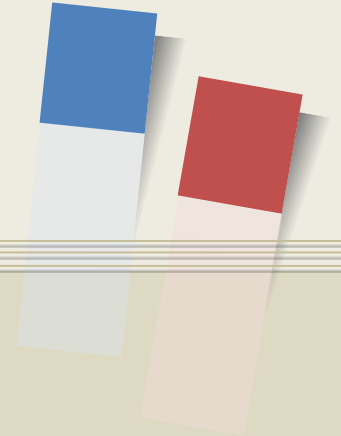
- ❑ If it detects objects that are too far,
 - ❑ Try with different resistors
 - ❑ Ex) brown-black-orange

 - ❑ Adjust the directions of your IR LEDs are facing

CS310 Embedded Computer System
Computer Architecture Lab.



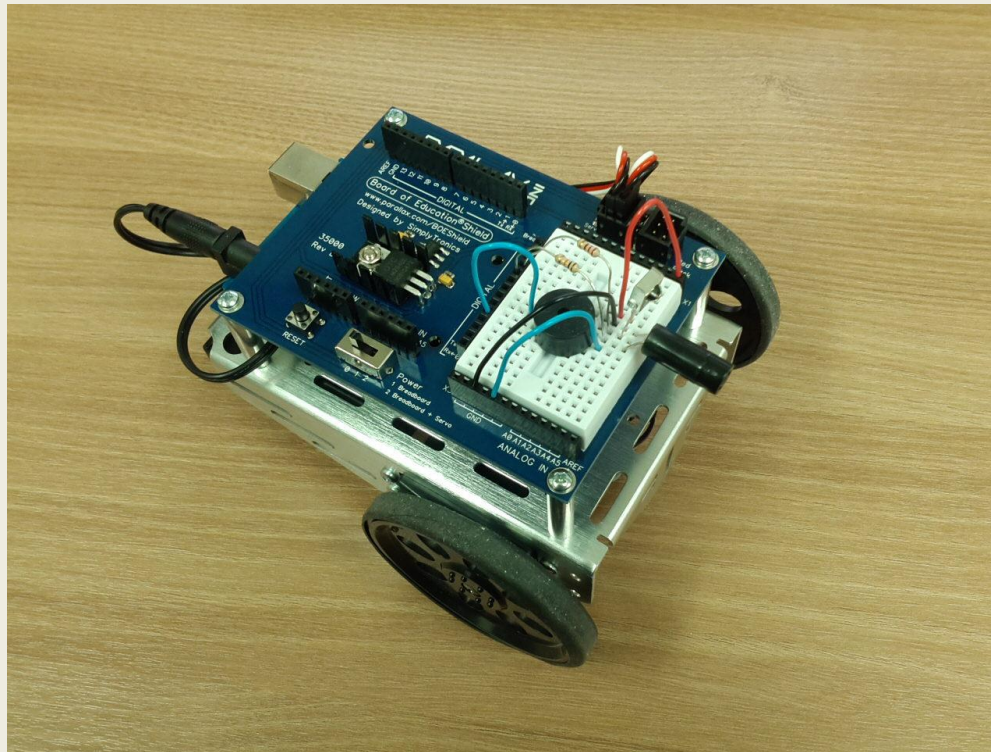
ASSIGNMENT





Description

- ❑ Given a terrain, escape it with only one IR Sensor





Probe both directions!



- ❑ When encounter an obstacle,
 - ▣ Probe both left and right side and choose empty path
 - ▣ If obstacles are on both sides, turn 180 degree

- ❑ There is no strict answer!
 - ▣ 'Play around with the robot' is the purpose of this lab

CS310 Embedded Computer System
Computer Architecture Lab.



ENJOY!

