Infrared Line Follower Kit (#28034)

The Infrared Line Follower Kit from Parallax provides eight infrared emitter and receiver pairs for high-precision line-following applications. Mounting hardware for both the Boe-Bot® and Stingray™ robots are also included for easy startup.

All IR frequency generation is done onboard using an ICM7555 chip, requiring no external signaling. This allows for simple interfacing to the Infrared Line Follower, since the user only needs to read the state of each IR receiver to detect white or black surfaces. An "Enable" pin is also included which when pulled low puts the Infrared Line Follower into low-power mode.

Features

- Compatible with both the Boe-Bot and Stingray robots from Parallax
- Onboard 3 V regulator for easy use with Propeller-powered robots
- Easily adjustable 38–43 kHz IR frequency using an onboard potentiometer (See Calibration, page 8)
- Onboard LEDs for instant line detection communication
- High immunity to ambient lighting conditions
- Mounting hardware included
- No external signaling required

Key Specifications

- Power Requirements: 3.3 to 5.5 VDC; maximum 35 mA working, 30 µA when disabled
- Communication: Single bit high/low output for each IR emitter/receiver pair
- Operating temperature: 32 to 140 °F (0 to 50 °C)
- Dimensions: 3.03 x 1.33 in (7.7 x 3.4 cm)

Application Ideas

- Line following
- Maze navigation
- Robotic contests
- Detect line stripe width
- Edge detection

Bill of Materials

Part #	Quantity	Description
-	1	Infrared Line Follower PCB
-	2	10” 4-pin female/female cables
-	1	10” 3-pin female/female cable
-	2	4-pin male/male headers
-	1	3-pin male/male header
710-00007	2	7/8" 4-40 pan-head screws
700-00028	2	1/4" 4-40 pan-head screws
700-00060	2	1" round standoffs
713-00007	2	1/2" round spacers
700-00003	2	4-40 x 3/8" nuts

Additional Items Required

- #2 Philips-head screwdriver
- Black 3/4" electrical tape
- White poster board

Resources and Downloads

Check for the latest version of this document, free software, and example programs from the Infrared Line Follower Kit product page. Go to www.parallax.com and search 28034.

Theory of Operation

Upon connecting power, the onboard ICM7555 chip begins sending a 38–43 kHz signal through all 8 IR LEDs. If the IR LED is over a white surface, light is reflected to the IR receiver, and its output is low. When the IR LED is over a black surface, no light is reflected to the IR receiver, and its output is high.

The voltage regulator included on the PCB limits the power to the Infrared Line Follower’s circuitry to 3 V. This means that a 5 V supply can safely be used with microcontrollers operating at 3.3 V.

Red LEDs located on the top of the board are wired to the output of each IR receiver, and the anode of each LED is connected to power. When the IR LED is over a white surface, the low signal completes the LED circuit and turns the LED on. Conversely, when the IR LED is over a black surface, the LED receives dual high signals and the LED is off. This allows for easy visual feedback of the Infrared Line Follower’s output states.

An onboard potentiometer also allows for the easy adjustment of the infrared frequency between 38 and 43 kHz. This allows the sensor to detect different colored lines, and also allows for the easy adjustment of the sensor to different lighting conditions or mounting positions. For more information on how to calibrate the sensor, see the Calibration section on page 8.

To isolate each IR senor and prevent false triggering, headers are placed between each IR emitter and receiver pair.

Quick-Start Guide

Before using the Infrared Line Follower, it's a good idea to test the sensor's responsiveness in the area intended for use. This section includes instructions for mounting and testing the Infrared Line Follower on your Boe-Bot or Stingray robot, as well as for calibrating it in your lighting conditions. In order to use the example code below, you will need a strip of black electrical tape on white paper.

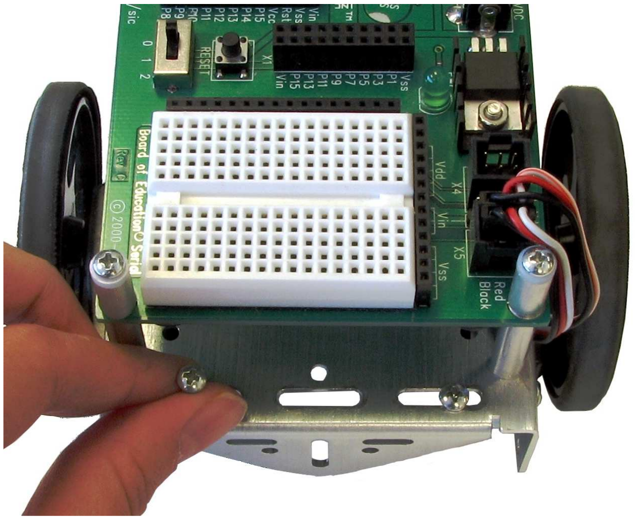
Boe-Bot Assembly Instructions

To mount the Infrared Line Follower on your Boe-Bot robot, you will need the following materials:

- (2) 7/8" 4-40 pan-head screws
- (2) 1/4" 4-40 pan-head screws
- (2) 1" round standoffs
- (2) 1/2" round spacers

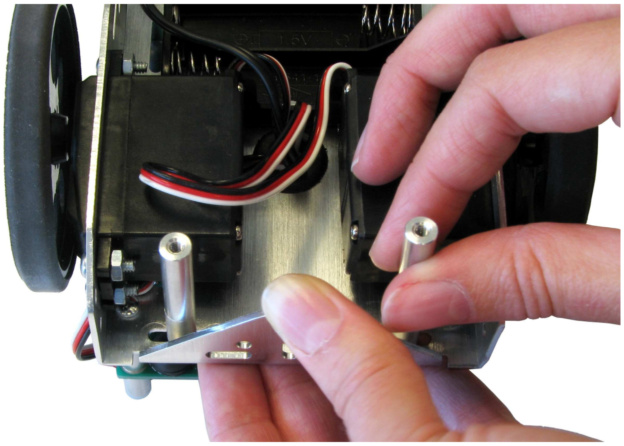
Step 1

Insert the 7/8" screws through the top of the Boe-Bot chassis through the left and right slots at the front.



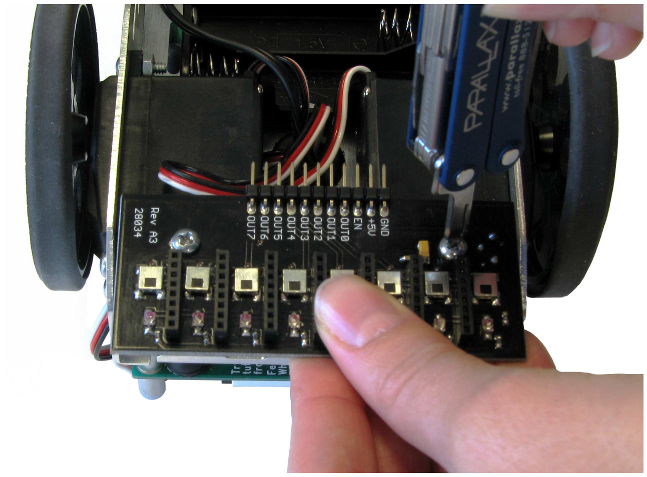
Step 2

On the underside of the chassis, slip on the 1/2" unthreaded spacer on each screw, followed by the 1" standoff. Don't tighten the standoff completely to allow for easier alignment of the Infrared Line Follower's mounting holes in the next step.



Step 3

Align the threaded standoffs with the mounting holes on the Infrared Line Follower and use the 1/4" screws to secure the sensor to the screws. The Infrared Line Follower should be facing downward, with the headers pointing toward the tail wheel.



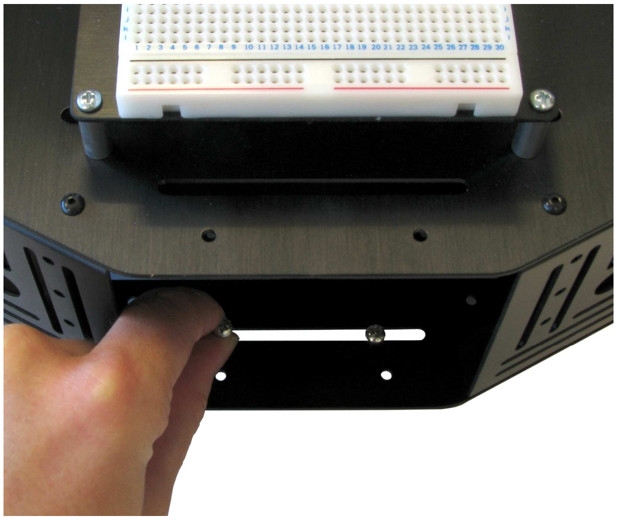
Stingray Assembly Instructions

To mount the Infrared Line Follower on your Stingray, you will need the following materials:

- (2) 7/8" 4-40 pan-head screws
- (2) 1/2" round spacers
- (2) 4-40 x 3/8" nuts

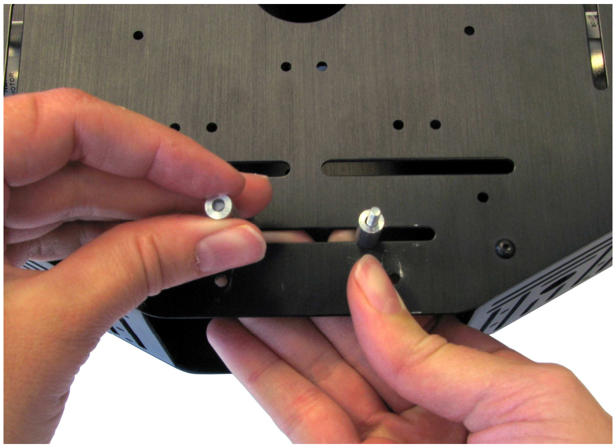
Step 4

Remove the front Stingray plate and insert the 7/8" screws through the top of the chassis through the front slot.



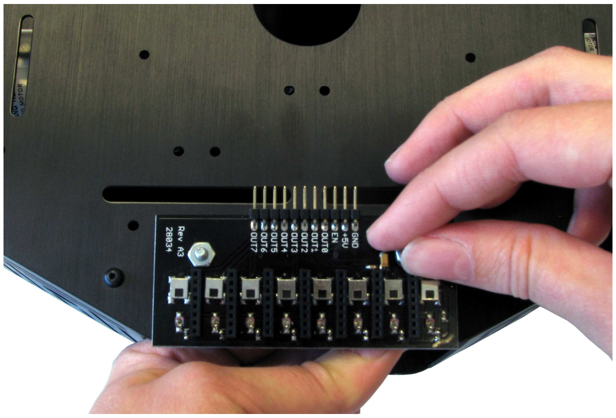
Step 5

On the underside of the chassis, slip on the 1/2" unthreaded spacer on each screw.



Step 6

Align the screws with the mounting holes on the Infrared Line Follower and use the nuts to secure the sensor to the screws. The Infrared Line Follower should be facing downward, with the headers pointing toward the tail wheel.



Pin Definitions

Pin	Name	Type	Function
1	GND	G	Ground -> 0V
2	Vdd	P	Supply Voltage
3	EN	I	Enable pin, pull low for low-power mode
4	OUT0	O	Output state of IR receiver 0
5	OUT1	O	Output state of IR receiver 1
6	OUT2	O	Output state of IR receiver 2
7	OUT3	O	Output state of IR receiver 3
8	OUT4	O	Output state of IR receiver 4
9	OUT5	O	Output state of IR receiver 5
10	OUT6	O	Output state of IR receiver 6
11	OUT7	O	Output state of IR receiver 7

Pin Type: P = Power, G = Ground, I = Input, O = Output

Quick-Start Circuit

The circuit pictured can be used with the example code below, as well as for the line following code included at the end of this document.

Note: In order to use this sensor with the Stingray, you will need to bypass the voltage translators on the Propeller Robot Control Board. The Infrared Line Follower has a 3 V onboard regulator, and the Propeller Robot Control Board's translators will interfere with signals coming from the sensor.

For instructions on how to disable the translators, see the documentation available on the Propeller Robot Control Board's product page. Go to www.parallax.com and search 28230.

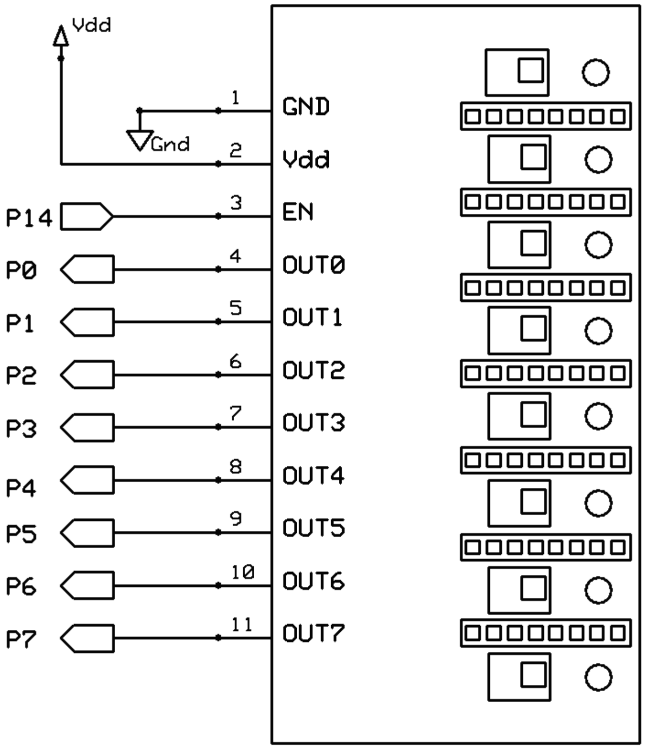


Figure 1: Infrared Line Follower Quick-Start Circuit

BASIC Stamp 2 Example Code

The program InfraredLineFollower_Simple.bs2 will test the Infrared Line Follower sensors before they're used in a line-following application. Follow the instructions below to test that all 8 sensors are working correctly.

- ✓ Affix a 3/4" piece of electrical tape on a white piece of paper.
- ✓ Enter & run InfraredLineFollower_Simple.bs2 (below).
- ✓ Place sensor S0 over the electrical tape, and the other sensors over the white paper.
- ✓ The Debug Terminal should read %00000001 and the S0 LED should be off.
- ✓ Repeat this process until all eight sensors are tested.

If your sensor doesn't display a logical 1 when over a black surface and a logical 0 when over a white surface, check your connections and/or try adjusting the frequency in the Calibration section on page 8.

```
' InfraredLineFollower_Simple.bs2
' Displays output states of the Infrared Line Follower using the Debug Terminal.

' {$STAMP BS2}
' {$PBASIC 2.5}

state   VAR    Byte

DO
  state = INL                                    ' Read input state of pins 0-7
  DEBUG HOME, "state = %", BIN8 state            ' Display states on Debug Terminal
LOOP
```

Propeller P8X32A Example Code

The program below will test the Infrared Line Follower sensors before they're used in a line-following application. Follow the instructions below to test that all 8 sensors are working correctly.

- ✓ Affix a 3/4" piece of electrical tape on a white piece of paper.
- ✓ Enter & run InfraredLineFollower_Simple.spin.
- ✓ Place S0 over the electrical tape, and the other sensors over the white paper.
- ✓ The Parallax Serial Terminal should read %00000001 and the S0 LED should be off.
- ✓ Repeat this process until all eight sensors are tested.

If your sensor doesn't display a logical 1 when over a black line and logical 0 when over a white line, check your connections and/or try adjusting the frequency in the Calibration section on page 8.

```
'' InfraredLineFollower_Simple.spin
'' Displays output states of the Infrared Line Follower using the Parallax Serial Terminal.

  _clkmode = xtal1 + pll16x
  _xinfreq = 5_000_000

VAR

  byte  state

OBJ

  pst : "Parallax Serial Terminal"

PUB Main

  outa[0..7]~                             ' Sets pins 0-7 as input
  pst.start(115_200)                      ' Initialize Parallax Serial Terminal
  waitcnt(clkfreq + cnt)

  repeat
    state := ina[7..0]                    ' Read the states of pins 0-7
    pst.home                              ' Set cursor to upper left corner
    pst.bin(state, 8)                     ' Display states in PST
    waitcnt(clkfreq/200 + cnt)
```

Calibration

Different lighting conditions, mounting positions or changes in temperature may require different frequency settings. For this reason, the Infrared Line Follower has a user selectable infrared frequency range of 38-43 kHz using an onboard potentiometer. If you are having problems reading a line when running the test code, place all IR detectors over a black surface and gently adjust the onboard potentiometer until all eight red LEDs are off.

Note: A small flathead screwdriver is required to adjust the frequency range. The Parallax screwdriver included in the Boe-Bot and Stingray robot kits is usually sufficient.

Boe-Bot Robot Line Following Code

The code below can be used for simple Boe-Bot line following applications, and is compatible with the circuit shown in Figure 1. Keep in mind that this is not the most robust line following code, which can and should be expanded to better fit your needs.

```
' InfraredLineFollower_BoeBot.bs2
' Simple line following Boe-Bot code using the Infrared Line Follower.

' {$STAMP BS2}
' {$PBASIC 2.5}

DO
  SELECT INL                                  ' Select line follower states
    CASE %11000000, %11100000                 ' Pivot right
      PULSOUT 13, 650
      PULSOUT 12, 650
    CASE %01100000, %01110000                 ' Curve right
      PULSOUT 13, 700
      PULSOUT 12, 650
    CASE %00110000                            ' Slight right
      PULSOUT 13, 750
      PULSOUT 12, 650
    CASE %00111000                            ' Adjust right
      PULSOUT 13, 800
      PULSOUT 12, 650
    CASE %00011000                            ' Straight
      PULSOUT 13, 850
      PULSOUT 12, 650
    CASE %00011100                            ' Adjust left
      PULSOUT 13, 850
      PULSOUT 12, 700
    CASE %00001100                            ' Slight left
      PULSOUT 13, 850
      PULSOUT 12, 750
    CASE %00000110, %00001110                 ' Curve left
      PULSOUT 13, 850
      PULSOUT 12, 800
    CASE %00000011, %00000111                 ' Pivot left
      PULSOUT 13, 850
      PULSOUT 12, 850
  ENDSELECT
LOOP
```

Stingray Robot Line Following Code

The line following code included below is ideal for longer courses with wide-radius turns, and is compatible with the circuit shown in Figure 1.

This code was written to be used as a skeleton for higher-precision line following applications using PID and/or other control algorithms.

```
'' InfraredLineFollower_Stingray.spin
'' Simple line following code compatible with the Parallax Stingray Robot, best used
'' on longer courses with wide-radius turns

CON

  _clkmode = xtal1 + pll16x
  _xinfreq = 5_000_000

  MotorA = 25                      ' Motor A connected to P25
  MotorB = 26                      ' Motor B connected to P26

VAR

  byte DutyCycleA, DutyCycleB, state

OBJ

  PWM : "PWM_32_V2.spin"

PUB Main

  PWM.Start                                 ' Start PWM object
  dira[7..0]~                               ' Set pins 7..0 as input

  repeat
    state := ina[7..0]                      ' Set input states to variable state
    case state
      %00000011, %00000001, %00000010:      ' Pivot left
        DutyCycleA := 65
        DutyCycleB := 0
      %00000111, %00000110, %00000100:      ' Slight left
        DutyCycleA := 60
        DutyCycleB := 0
      %00001110, %00001100:                 ' Adjust left
        DutyCycleA := 55
        DutyCycleB := 0
      %00011100, %00011000, %00111000:      ' Straight
        DutyCycleA := DutyCycleB := 50
      %01110000, %00110000:                 ' Adjust right
        DutyCycleA := 0
        DutyCycleB := 55
      %11100000, %01100000, %0010000:       ' Slight right
        DutyCycleA := 0
        DutyCycleB := 60
      %11000000, %01000000, %10000000:      ' Pivot right
        DutyCycleA := 0
        DutyCycleB := 65
      other:                                ' Anything else, slowly move forward
        DutyCycleA := DutyCycleB := 20

    PWM.Duty(MotorA, DutyCycleA, 50)        ' Adjust motor speeds
    PWM.Duty(MotorB, DutyCycleB, 50)
```