# Table of Contents

## Overview
- Parts
- Optional

## Build the FunHouse Parking Assistant
- Cable Prep
- NeoPixel Strip

## Code the FunHouse Parking Assistant
- Install CircuitPython
- Text Editor
- Shhhh... Secrets
- Download the Project Bundle
- How it Works
- Variables
- FunHouse and NeoPixels
- Sonar
- Main Loop

## Use the Parking Assistant
Overview

Use the FunHouse and an ultrasonic distance sensor to assist you when parking your vehicle in your garage or other permanent parking space.

This project uses the FunHouse to read a distance sensor and then guide the driver with colored LEDs and sounds!

Parts

Adafruit FunHouse - WiFi Home Automation Development Board
Home is where the heart is...it's also where we keep all our electronic bits. So why not wire it up with sensors and actuators to turn our house into an electronic wonderland....
https://www.adafruit.com/product/4985
Adafruit NeoPixel LED Strip with 3-pin JST PH 2mm Connector
Plug in and glow, this Adafruit NeoPixel LED Strip with JST PH Connector has 30 total LEDs in a "60 LED per meter" spacing,...
https://www.adafruit.com/product/3919

Ultrasonic Distance Sensor - 3V or 5V - HC-SR04 compatible
If you're like me, you've dreamed of being a dolphin - smoothly gliding through the water. Using your echo-location abilities to detect tasty fish treats. Until genetic...
https://www.adafruit.com/product/4007

STEMMA JST PH 2mm 3-Pin to Female Socket Cable - 200mm
This cable will let you turn a JST PH 3-pin cable port into 3 individual wires with high-quality 0.1" female header sockets on the end. We're carrying these to match up with...
https://www.adafruit.com/product/3894

Official Raspberry Pi Power Supply 5.1V 3A with USB C
The official Raspberry Pi USB-C power supply is here! And of course, we have 'em in classic Adafruit black! Superfast with just the right amount of cable length to get your Pi 4...
https://www.adafruit.com/product/4298
Optional

**Premium Female/Female Jumper Wires - 20 x 6" (150mm)**
These female-female jumper wires are handy for making wire harnesses or jumpering between headers on PCB's. These premium jumper wires are 6" (150mm) long and come in a...
https://www.adafruit.com/product/1950

**Mini Magnet Feet for RGB LED Matrices (Pack of 4)**
Got a glorious RGB Matrix project you want to mount and display in your workspace or home? If you have one of the matrix panels listed below, you'll need a pack of these...
https://www.adafruit.com/product/4631

---

**Build the FunHouse Parking Assistant**
You'll use the three JST PH 3-Pin ports on the FunHouse to control the ultrasonic distance sensor and the NeoPixel strip.

Cable Prep
The distance sensor uses power, ground, a digital output (to control pings), and a digital input (to read the signal). You can wire these four connections by using two of the JST ports.

Prep two of the cables by removing the ground from one and the power from the other as shown. Pry up the plastic tab holding the crimp connector and then slide the wire out of the housing.

Alternately, you can simply leave the extra two wires in place and simple insulate their ends to prevent accidental shorting.

You don’t *have* to prep the cables, it just neatens the wiring a bit! If you skip this step, just wrap and tie the extra wires out of the way.
Plug the two cables into the A0 and A1 ports of the FunHouse.

Optional mini magnetic feet can be used for mounting the FunHouse to some metal surfaces.
You can plug the sensor into the JST cables directly, or use extension cables to connect the JST wires to the ultrasonic sensor as shown here:

- 3V to VCC
- A0 to Trig
- A1 to Echo
- Gnd to Gnd
NeoPixel Strip
Plug in the NeoPixel strip's JST plug to the A2 port on the FunHouse.

You're now ready to code the FunHouse Parking Assistant using CircuitPython.
Code the FunHouse Parking Assistant

Install CircuitPython

The first thing to do is install CircuitPython on your Funhouse. Follow this guide() to get it set up with the latest release version.

Text Editor

Adafruit recommends using the Mu editor for using your CircuitPython code with the Funhouse. You can get more info in this guide().

Shhh... Secrets

In order for the Funhouse to connect to the internet, you'll need to include a secrets.py file on the board that contains your WiFi access point ssid and password, as well as your AIO key.

Follow this guide page() to get your secrets.py file set up.

Download the Project Bundle

Your project will use a specific set of CircuitPython libraries and the code.py file. In order to get the libraries you need, click on the Download Project Bundle link below, and uncompress the .zip file.

```python
# SPDX-FileCopyrightText: Copyright (c) 2021 John Park for Adafruit
#
# SPDX-License-Identifier: MIT
# FunHouse Parking Assistant

import time
import board
import adafruit_hcsr04
import neopixel
from adafruit_funhouse import FunHouse

SLOW_DISTANCE = 30  # distance (in centimeters) when you should slow
STOP_DISTANCE = 8   # distance when you should hit those brakes

GREEN = 0x00FF00
AMBER = 0xF0D000
RED = 0xFF0000

funhouse = FunHouse(default_bg=None, scale=3)
funhouse.peripherals.dotstars.brightness = 0.05
funhouse.peripherals.dotstars.fill(GREEN)
```
pixel_pin = board.A2
temp_pixels = 30
pixels = neopixel.NeoPixel(pixel_pin, num_pixels, brightness=0.3, auto_write=False)

sonar = adafruit_hcsr04.HCSR04(trigger_pin=board.A0, echo_pin=board.A1)

while True:
    try:
        print((sonar.distance,))
        if sonar.distance > SLOW_DISTANCE:
            funhouse.peripherals.dotstars.fill(GREEN)
pixels.fill(GREEN)
pixels.show()
        elif sonar.distance < SLOW_DISTANCE and sonar.distance > STOP_DISTANCE:
            funhouse.peripherals.dotstars.fill(AMBER)
pixels.fill(AMBER)
pixels.show()
funhouse.peripherals.play_tone(1000, 0.3)
        elif sonar.distance < STOP_DISTANCE:
            funhouse.peripherals.dotstars.fill(RED)
pixels.fill(RED)
pixels.show()
funhouse.peripherals.play_tone(2600, 0.3)
    except RuntimeError:
        print("Retrying!")
time.sleep(0.01)

Next, drag the contents of the uncompressed bundle directory onto you microcontroller board CIRCUITPY drive, replacing any existing files or directories with the same names, and adding any new ones that are necessary.
How it Works

First, you'll load the necessary libraries, including `adafruit_hcsr04` for using the ultrasonic distance sensor.

```python
import time
import board
import adafruit_hcsr04
import neopixel
from adafruit_funhouse import FunHouse
```

Variables

Next, you'll set variables for the distance thresholds, in centimeters. You can adjust these as necessary for your particular garage/parking space and vehicle.

```python
SLOW_DISTANCE = 30  # distance (in centimeters) when you should slow
STOP_DISTANCE = 8   # distance when you should hit those brakes
```

FunHouse and NeoPixels

Now, you'll set up the FunHouse and the NeoPixels.

```python
GREEN = 0x00FF00
AMBER = 0xF0D000
RED = 0xFF0000

funhouse = FunHouse(default_bg=None, scale=3)
```
Sonar

The ultrasonic distance sensor is set up on the FunHouse A0 pin for triggering a ping, and A1 for receiving the echo.

```python
sonar = adafruit_hcsr04.HCSR04(trigger_pin=board.A0, echo_pin=board.A1)
```

Main Loop

Here's what happens in the main loop of the program. Inside of a try-except loop, the `sonar.distance` value is checked.

If it is greater than the `SLOW_DISTANCE` the NeoPixels are set to green.

If it is less than the `SLOW DISTANCE` and greater than the `STOP DISTANCE`, the NeoPixels are set to yellow and the FunHouse speaker plays a medium tone.

If the distance is less than the `STOP_DISTANCE`, the NeoPixels are set to red and the tone plays at a higher pitch.

```python
while True:
    try:
        print((sonar.distance,))

        if sonar.distance > SLOW_DISTANCE:
            funhouse.peripherals.dotstars.fill(GREEN)
            pixels.fill(GREEN)
            pixels.show()

        elif sonar.distance < SLOW_DISTANCE and sonar.distance > STOP_DISTANCE:
            funhouse.peripherals.dotstars.fill(AMBER)
            pixels.fill(AMBER)
            pixels.show()
            funhouse.peripherals.play_tone(1000, 0.3)

        elif sonar.distance < STOP_DISTANCE:
            funhouse.peripherals.dotstars.fill(RED)
            pixels.fill(RED)
            pixels.show()
            funhouse.peripherals.play_tone(2600, 0.3)

    except RuntimeError:
        print("Retrying!")
        time.sleep(0.01)
```
Use the Parking Assistant

Depending on your parking space or garage, your options will vary for mounting the FunHouse, ultrasonic distance sensor, and NeoPixel strip.

In the example shown here, there is a post that provides a convenient mounting point, and isn't something we want to hit with the bumper!

It's good to start with a temporary mounting method, such as the gaffer's tape seen here. This allows you to test out the mounting positions to make sure you get a good distance read and that the NeoPixel strip is visible to the driver.
After testing and getting the sensor height to match your vehicle's bumper, you may want to use something a little more permanent, such as double stick, high-strength tape.

Now, you can pull in as close as you need with ease by listening to the beeps and watching the NeoPixels change from green to yellow to red!

This demonstration was door at an outside parking space -- this project works most easily inside a garage -- if you need to set it up in the outdoors, you'll want to use a water resistant enclosure, such as this one, to protect from the elements.
Large Plastic Project Enclosure - Weatherproof with Clear Top
Store your project safe and sound in this nice weatherproof box with a clear top. We picked up this box because we like the machinable ABS plastic body and tough clear polycarbonate...
https://www.adafruit.com/product/905

Cable Gland PG-9 size - 0.158" to 0.252" Cable Diameter
We have some great waterproof and weather-proof items in the adafruit shop, but once you have a project built, you'll want...
https://www.adafruit.com/product/761