

The background is white and features a collection of decorative elements. There are several orange keys scattered across the page. Additionally, there are various colored circles: a large cyan circle in the top-left, a large green circle in the top-right, a large orange circle in the bottom-right, and a large green circle in the bottom-left. Some of these circles have smaller circles inside them, and some are outlined with dashed lines. A thin, light blue dashed line curves through the background, passing behind the text.

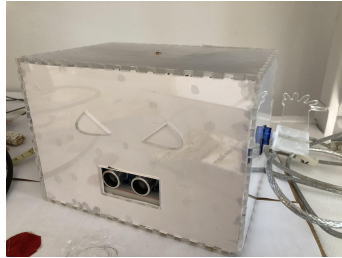
pesKey bot

A decorative graphic in the top-left corner consisting of several overlapping circles and rings in various colors: pink, orange, teal, and lime green. Some shapes have dashed outlines.

How often do you....

FORGET YOUR KEYS?

Hello!



Meet PesKey bot



1

How she works







Parts/Digital Fabrication Techniques We Used

- ◎ Ultrasonic Sensor
- ◎ Light Dependent Resistor
- ◎ LEDS
- ◎ Micro- Servo Motors
- ◎ Laser Cut Acrylic

Interaction



The receiving, processing/reflecting, and communication of information between two or more parties/objects.



Our Thought Process

Idea

Create something useful, entertaining, and interactive

Solution: animating a keyholder.

Input: distance from door

Processing: Determining if key is present

Output: Waving arms, alarm, blinking lights



Steps

Test distance sensor, servos, speaker, ldr, leds separately

Write out logic/ pseudo code

Implement the parts together



First Prototype





User Testing Feedback

- ⦿ Speaker too quiet
- ⦿ Arms moving too slow
- ⦿ Distance sensor in wrong position
- ⦿ LEDs not visible enough
- ⦿ Need to add a key sensor

Changes We Made

Changed to a
Grove Speaker
for better audio
quality

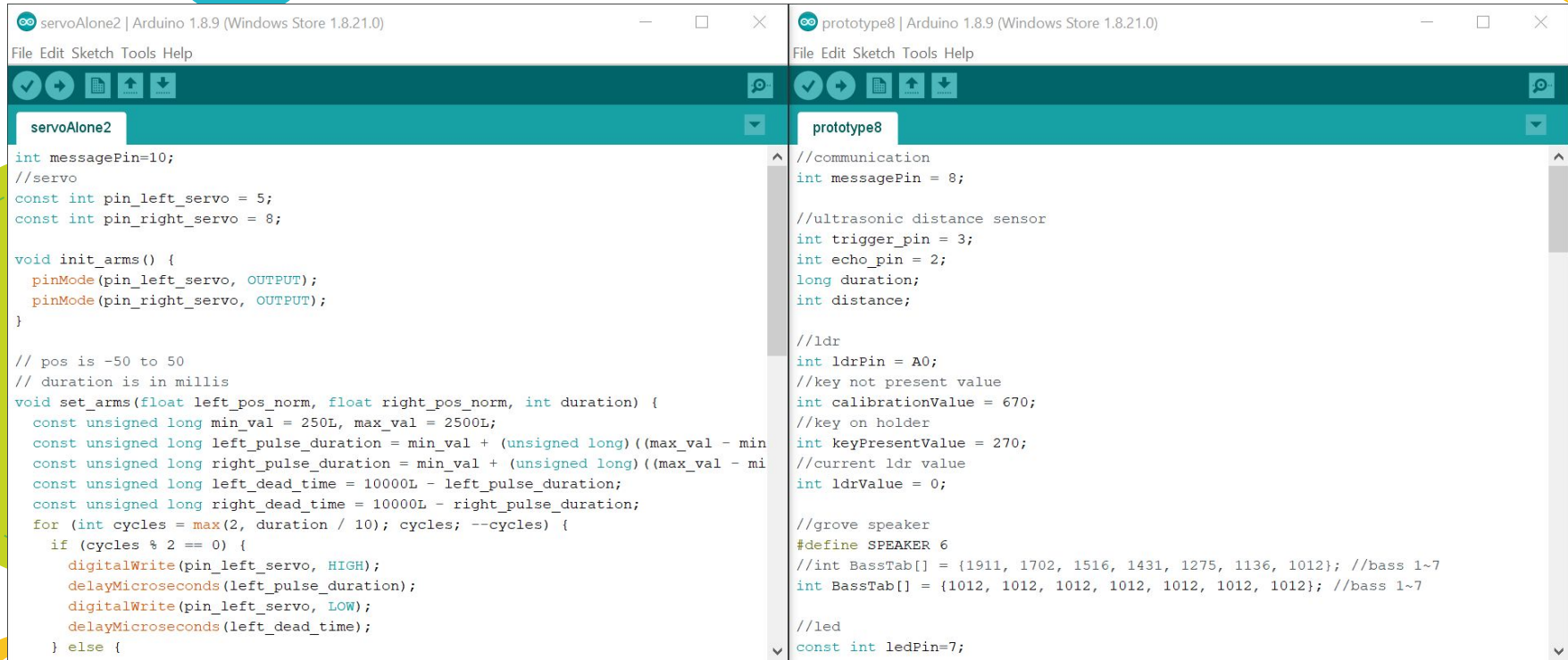
Separated the
servo motor
from the main
circuit

Oriented the
distance sensor
on the front
face of the bot

Laser cut using
transparent
acrylic

Added an Idr
sensor to
detect if key is
present

Final Code



The image shows two side-by-side windows of the Arduino IDE, version 1.8.9 (Windows Store 1.8.21.0). The left window is titled 'servoAlone2' and the right window is titled 'prototype8'. Both windows show code for controlling servos and an ultrasonic sensor.

```
servoAlone2
int messagePin=10;
//servo
const int pin_left_servo = 5;
const int pin_right_servo = 8;

void init_arms() {
  pinMode(pin_left_servo, OUTPUT);
  pinMode(pin_right_servo, OUTPUT);
}

// pos is -50 to 50
// duration is in millis
void set_arms(float left_pos_norm, float right_pos_norm, int duration) {
  const unsigned long min_val = 250L, max_val = 2500L;
  const unsigned long left_pulse_duration = min_val + (unsigned long)((max_val - min_val) * left_pos_norm);
  const unsigned long right_pulse_duration = min_val + (unsigned long)((max_val - min_val) * right_pos_norm);
  const unsigned long left_dead_time = 10000L - left_pulse_duration;
  const unsigned long right_dead_time = 10000L - right_pulse_duration;
  for (int cycles = max(2, duration / 10); cycles; --cycles) {
    if (cycles % 2 == 0) {
      digitalWrite(pin_left_servo, HIGH);
      delayMicroseconds(left_pulse_duration);
      digitalWrite(pin_left_servo, LOW);
      delayMicroseconds(left_dead_time);
    } else {
      digitalWrite(pin_right_servo, HIGH);
      delayMicroseconds(right_pulse_duration);
      digitalWrite(pin_right_servo, LOW);
      delayMicroseconds(right_dead_time);
    }
  }
}

prototype8
//communication
int messagePin = 8;

//ultrasonic distance sensor
int trigger_pin = 3;
int echo_pin = 2;
long duration;
int distance;

//ldr
int ldrPin = A0;
//key not present value
int calibrationValue = 670;
//key on holder
int keyPresentValue = 270;
//current ldr value
int ldrValue = 0;

//grove speaker
#define SPEAKER 6
//int BassTab[] = {1911, 1702, 1516, 1431, 1275, 1136, 1012}; //bass 1-7
int BassTab[] = {1012, 1012, 1012, 1012, 1012, 1012, 1012}; //bass 1-7

//led
const int ledPin=7;
```

Thanks!



Any questions?