

Real-world Automation with Arduino

Bob Igo

`bob@igo.name`

CPOSC 2012

Slides at <http://bob.igo.name/?cat=21>

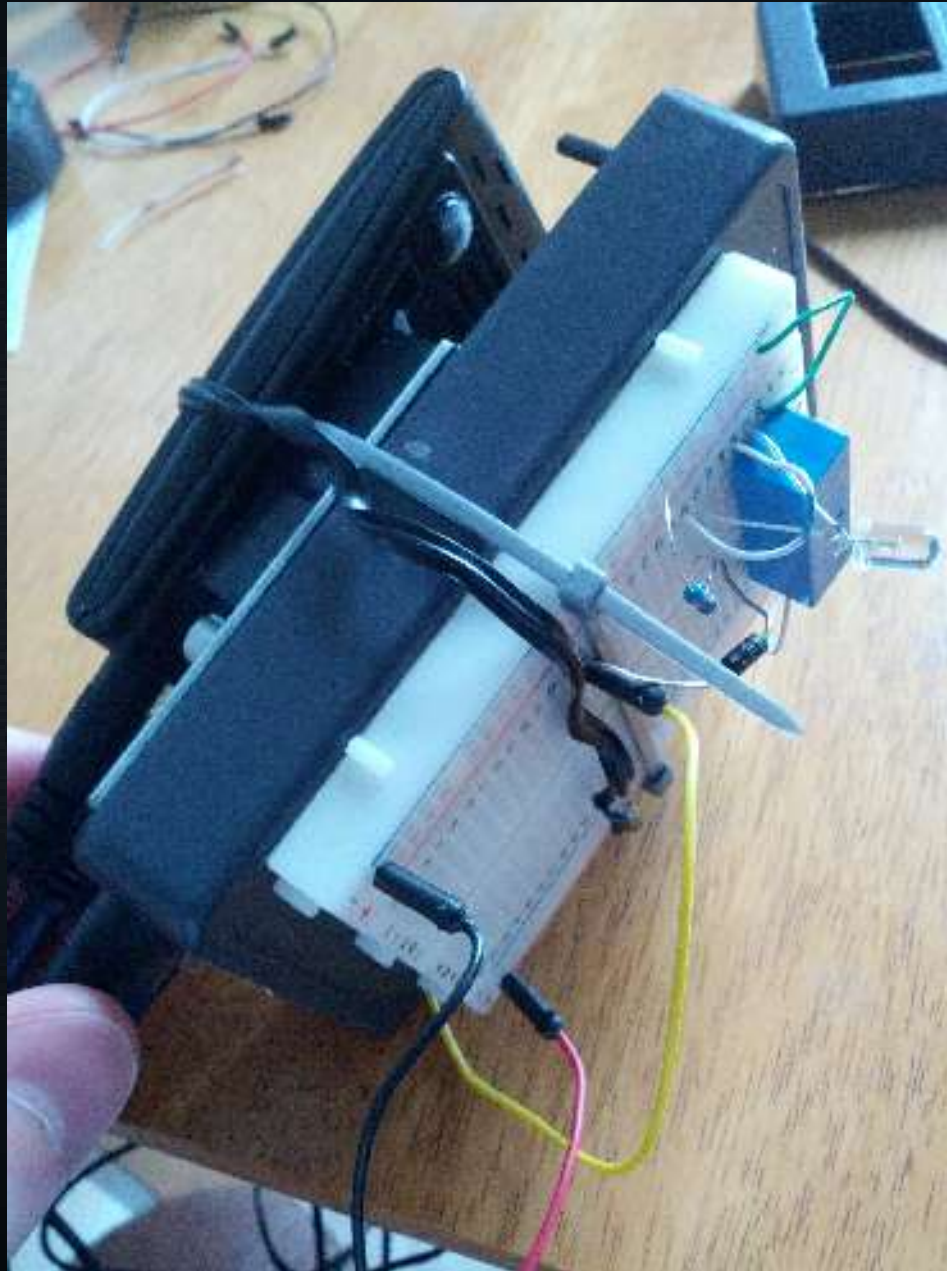
Problem 1: Crappy Garage Door Opener Reception

- My garage door can't hear either of our openers very well.
 - tried all basic troubleshooting steps
- Obvious next step:
 - make my house sense my car and open the garage door

Solution 1: Smart Garage Door Opener

- Requirements:
 - 1: Know when I've driven away in a car.
 - 2: Know when I've returned to the house.
 - 3: Open the door when I've returned.
 - 4: Prevent unauthorized usage.

Final Physical Form



Know when I've driven away in a car

- What happens on my Android phone when I've left the house in a car?

GPS location changes

- Speed > running/walking
- WAP signal goes away
- Car bluetooth (usually) connected

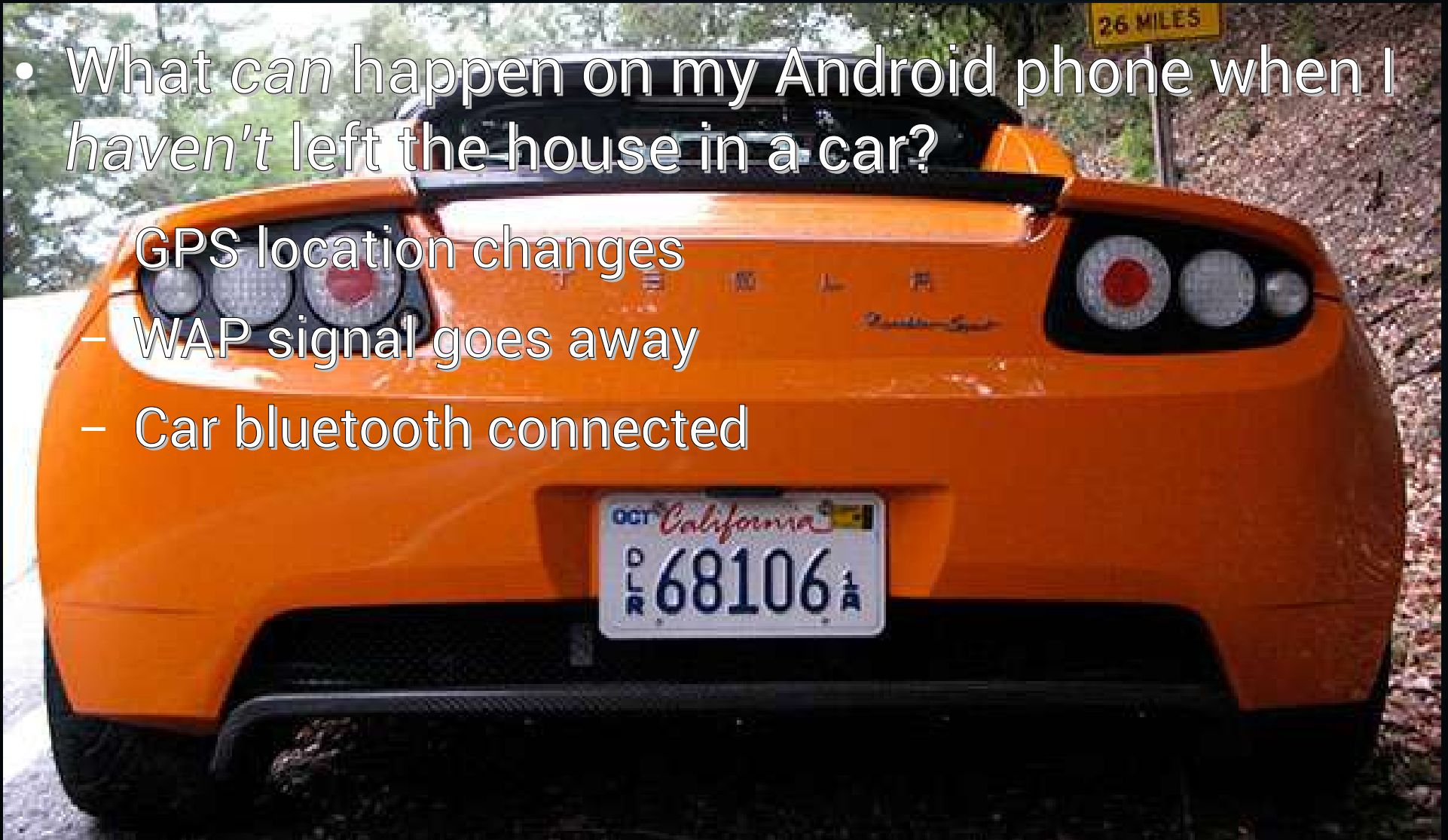


Know when I've driven away in a car

- What can happen on my Android phone when I haven't left the house in a car?

GPS location changes

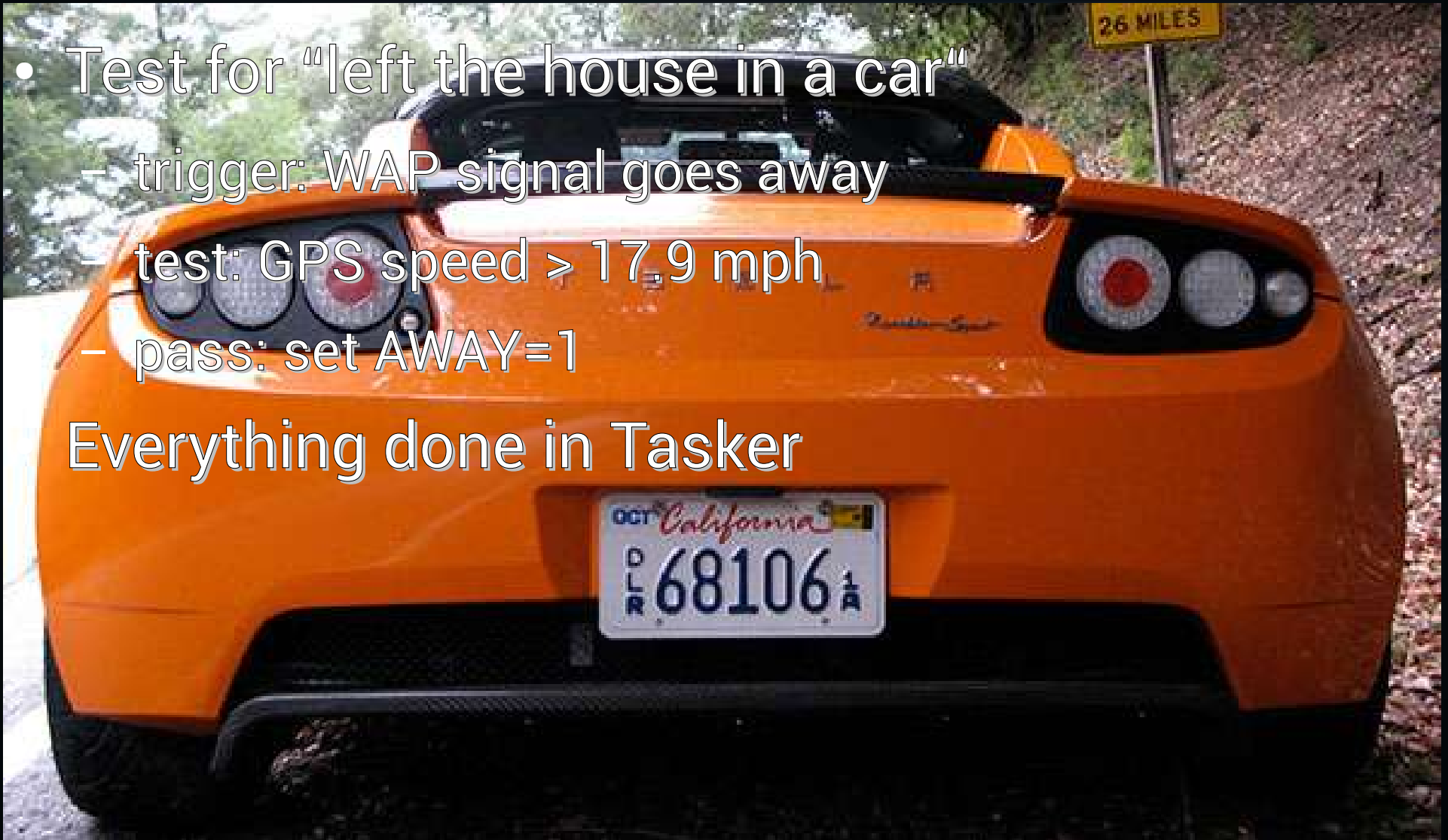
- WAP signal goes away
- Car bluetooth connected



Know when I've driven away in a car

- Test for "left the house in a car"
 - trigger: WAP signal goes away
 - test: GPS speed > 17.9 mph
 - pass: set AWAY=1

Everything done in Tasker



Know when I've returned to the house

- Test for "returned to the house"
 - trigger: WAP signal appears
 - test: `AWAY == 1`
- Upon return, get a URL
- Everything done in Tasker



2010 Tesla Roadster Sport

Know when I've returned to the house

- VZW-specific bug
 - In a call: can't associate with WAP



2010 Tesla Roadster Sport

Open the door when I've returned

- Requirements:
 - I need something that runs a web server
 - wires between it and the garage door opener
 - I need it to throw a switch (the opener)
 - 5VDC, 1.5mA
 - + all the standard implicit requirements
- Arduino can do all of that (with some help)

Open the door when I've returned

- Arduino + Ethernet Shield
 - mini web server that can throw a relay
- Android + Tasker
 - gets the relevant URL when I've returned home

Open the door when I've returned

- Example Arduino web server does 99%
 - additional code:
 - set the relay's control pin to OUTPUT
 - when server responds, set pin HIGH for 500ms

Prevent unauthorized usage

- Currently, WPA2 mostly does this
 - prevents anyone who lacks my WPA2 passphrase from discovering the URL on my LAN
 - although it's in my Tasker task



Prevent unauthorized usage

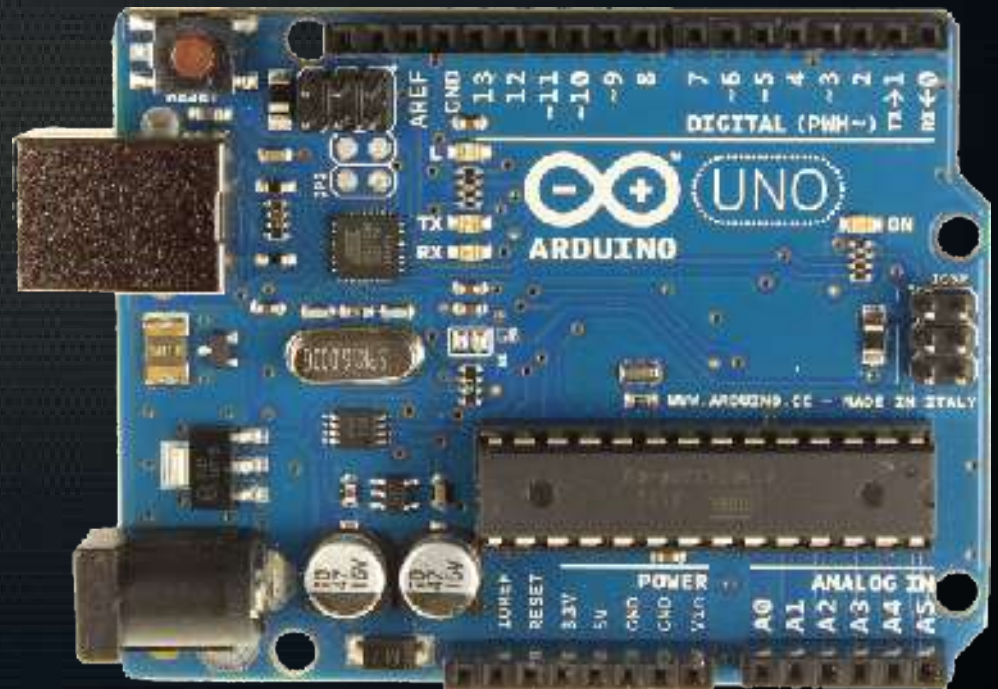
- Eventually
 - rolling codes as sub-URLs, synced between the Arduino server and the phone



Arduino Code

- Arduino IDE: Examples → Ethernet → Web Server
- Plus a relay controlled by Arduino pin 5:

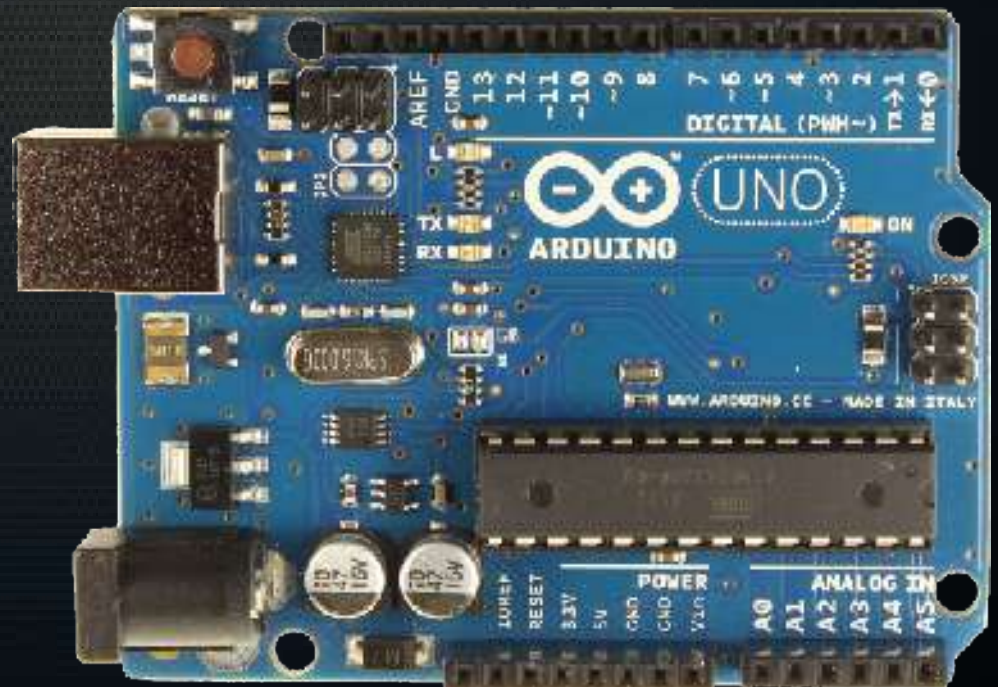
```
- pinMode(5, OUTPUT);  
- void briefRelayThrow() {  
    digitalWrite(5, HIGH);  
    delay(500);  
    digitalWrite(5, LOW);  
}
```



Arduino Code

- Add a call to `briefRelayThrow()` in `loop()`, after
`client.println("HTTP/1.1 200 OK");`
`client.println("Content-Type: text/html");`
`client.println();`

`briefRelayThrow();`

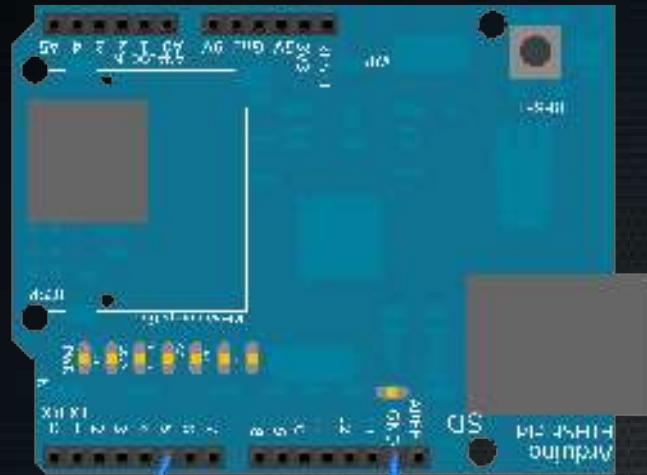


Networking

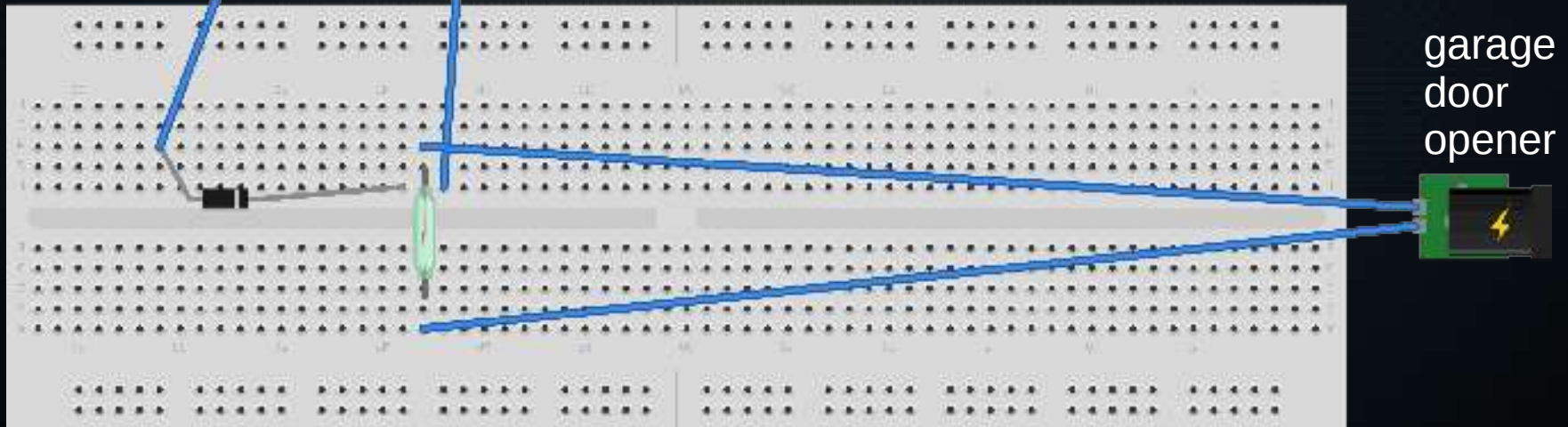
- I couldn't do a wire run to the garage
 - no ethernet
 - no PoE
- I used a wireless bridge
 - Netgear WNCE2001
 - WAP → WNCE2001 → ethernet shield



Arduino Wiring Diagram



NOTE: No LED modeled



Made with  Fritzing.org

Parts List

- Arduino Uno R3
- Arduino Ethernet Shield R3
- 5VDC SPST REED relay
 - RadioShack #275-0232
- IN4004 rectifier diode
 - RadioShack #276-1103
- (optional Netgear WNCE2001)
- (optional status LEDs and 1kOhm resistors)
- smartphone (ideally, Android with Tasker)
- protoboard, wires

Tasker Side

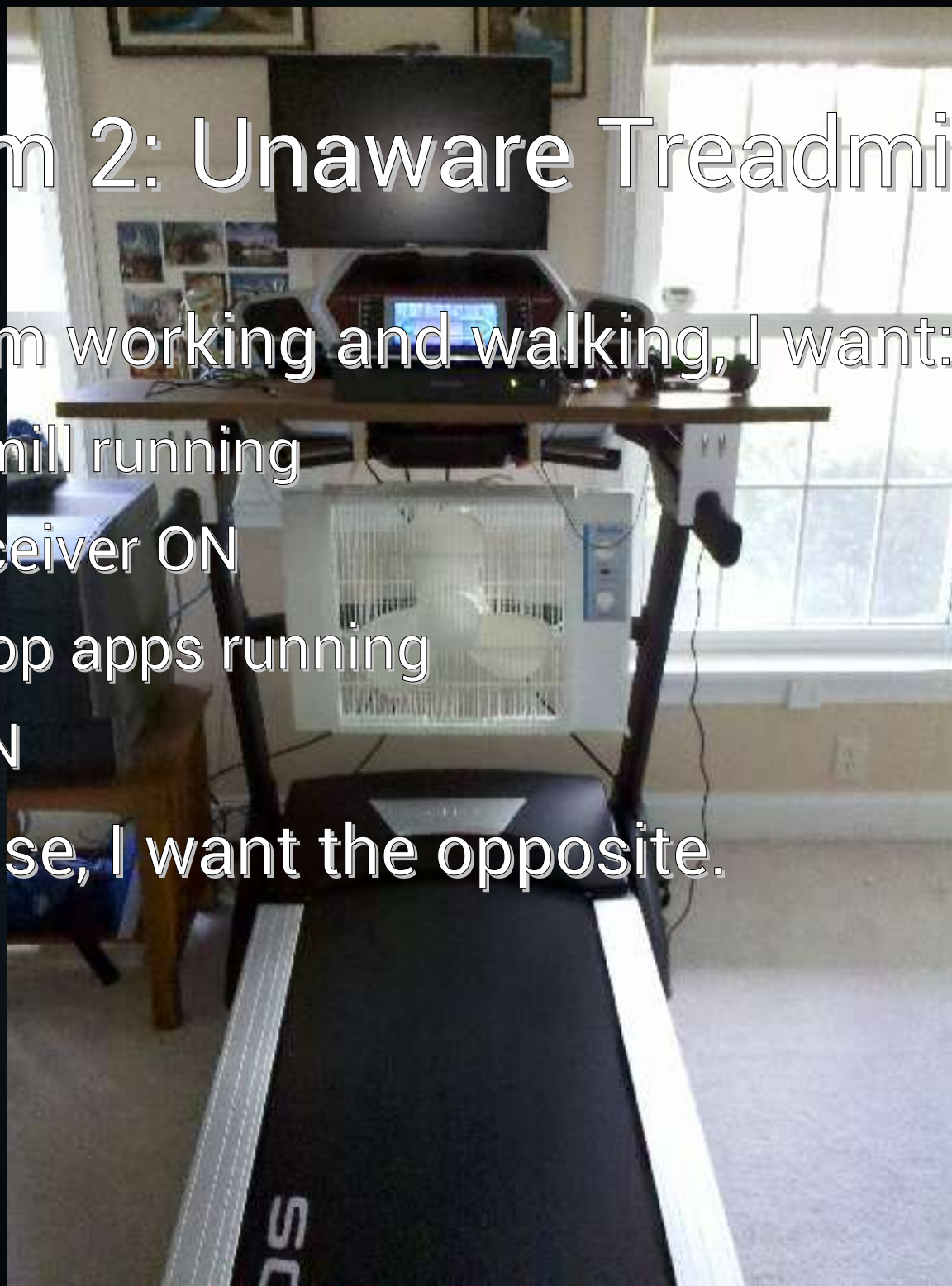
- See

<https://github.com/Human/igo-tasker>

Questions?

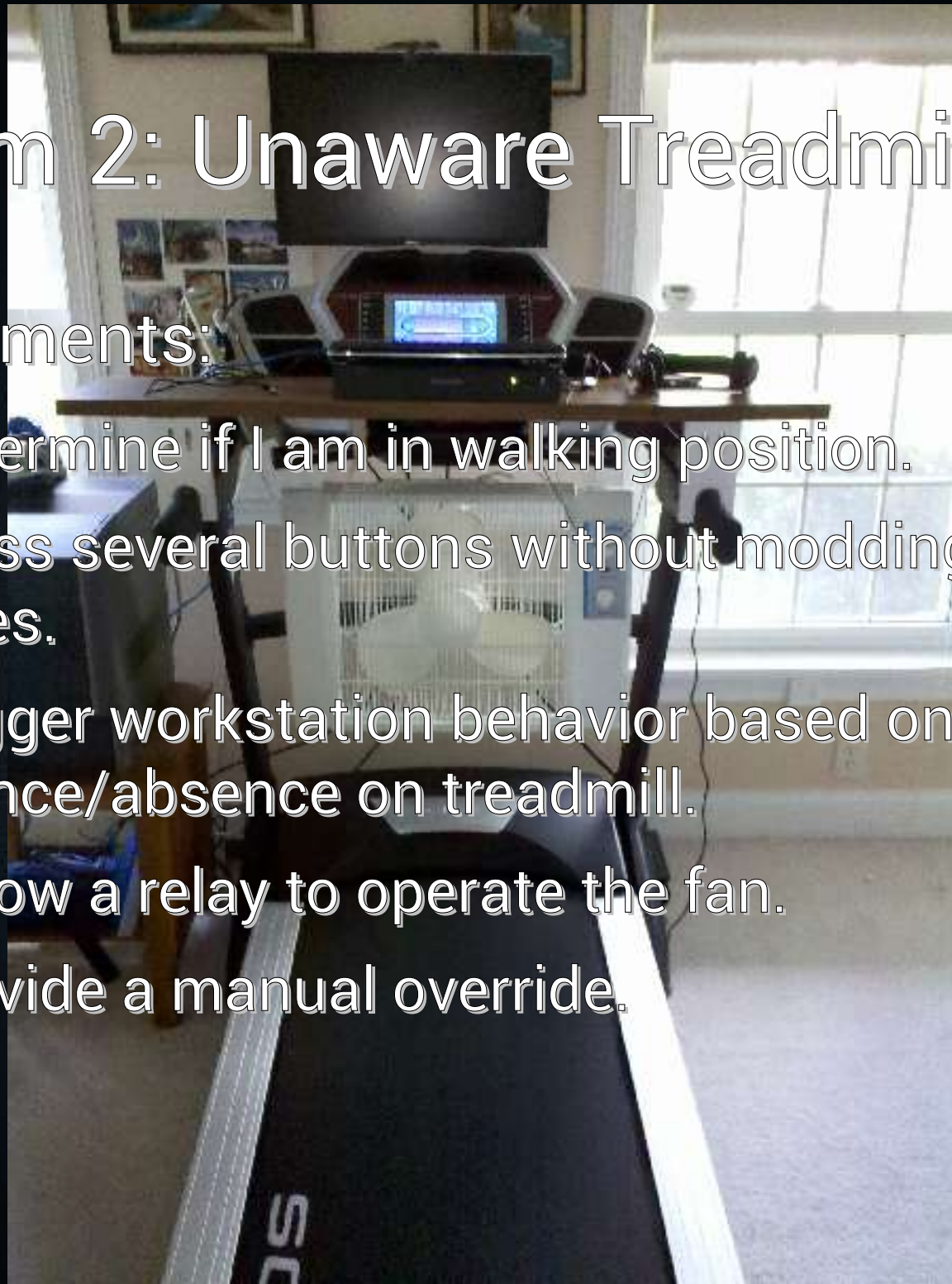
Problem 2: Unaware Treadmill Desk

- When I'm working and walking, I want:
 - treadmill running
 - AV receiver ON
 - desktop apps running
 - fan ON
- Otherwise, I want the opposite.

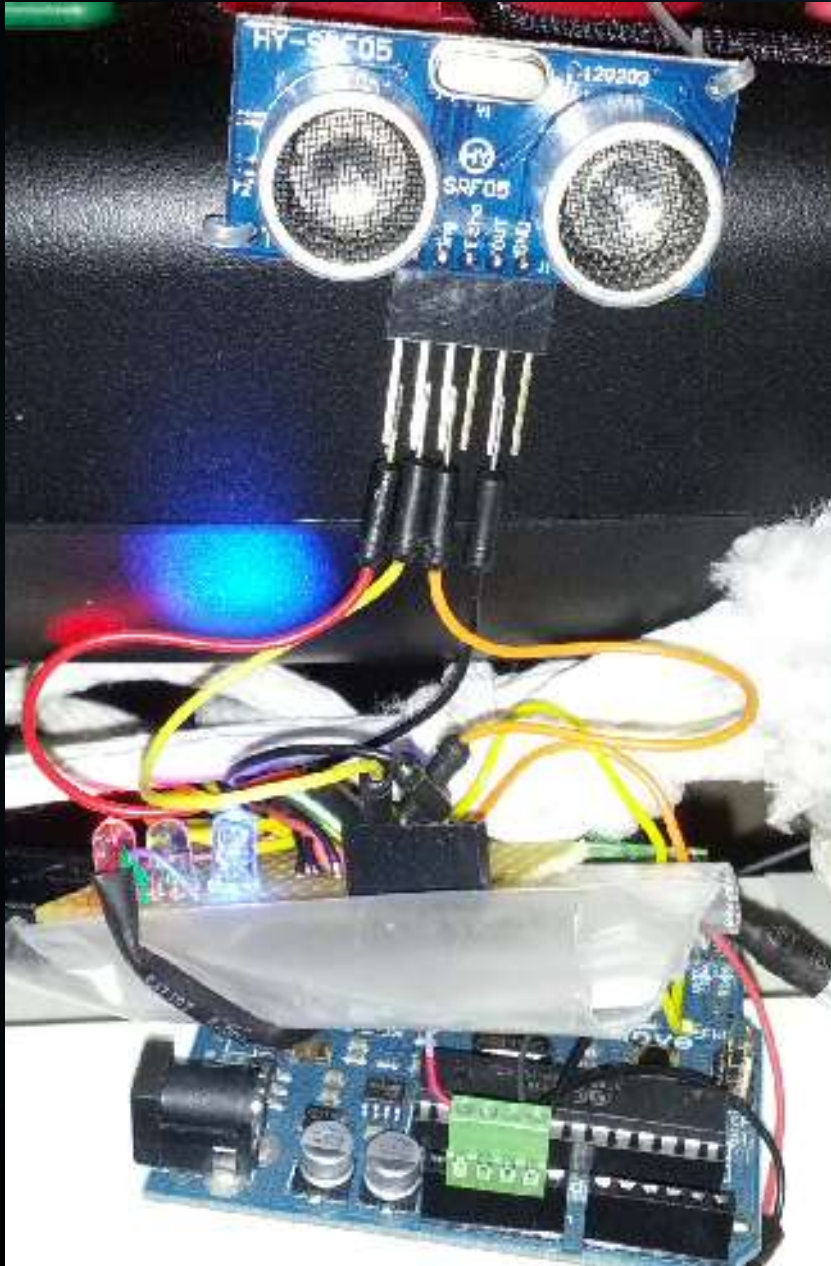


Problem 2: Unaware Treadmill Desk

- Requirements:
 - 1: Determine if I am in walking position.
 - 2: Press several buttons without modding the devices.
 - 3: Trigger workstation behavior based on my presence/absence on treadmill.
 - 4: Throw a relay to operate the fan.
 - 5: Provide a manual override.



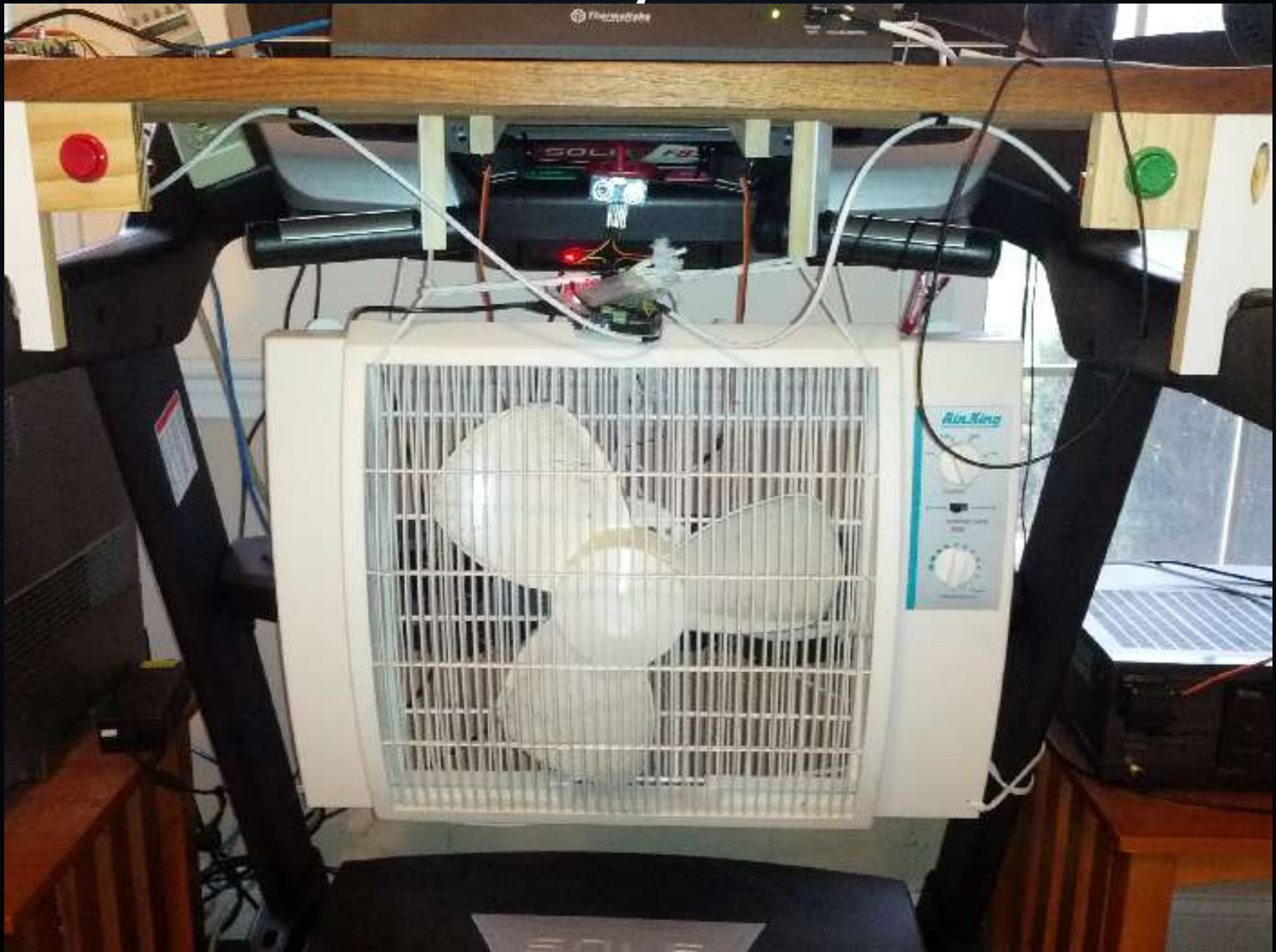
Current Physical Form



Current Physical Form



Current Physical Form

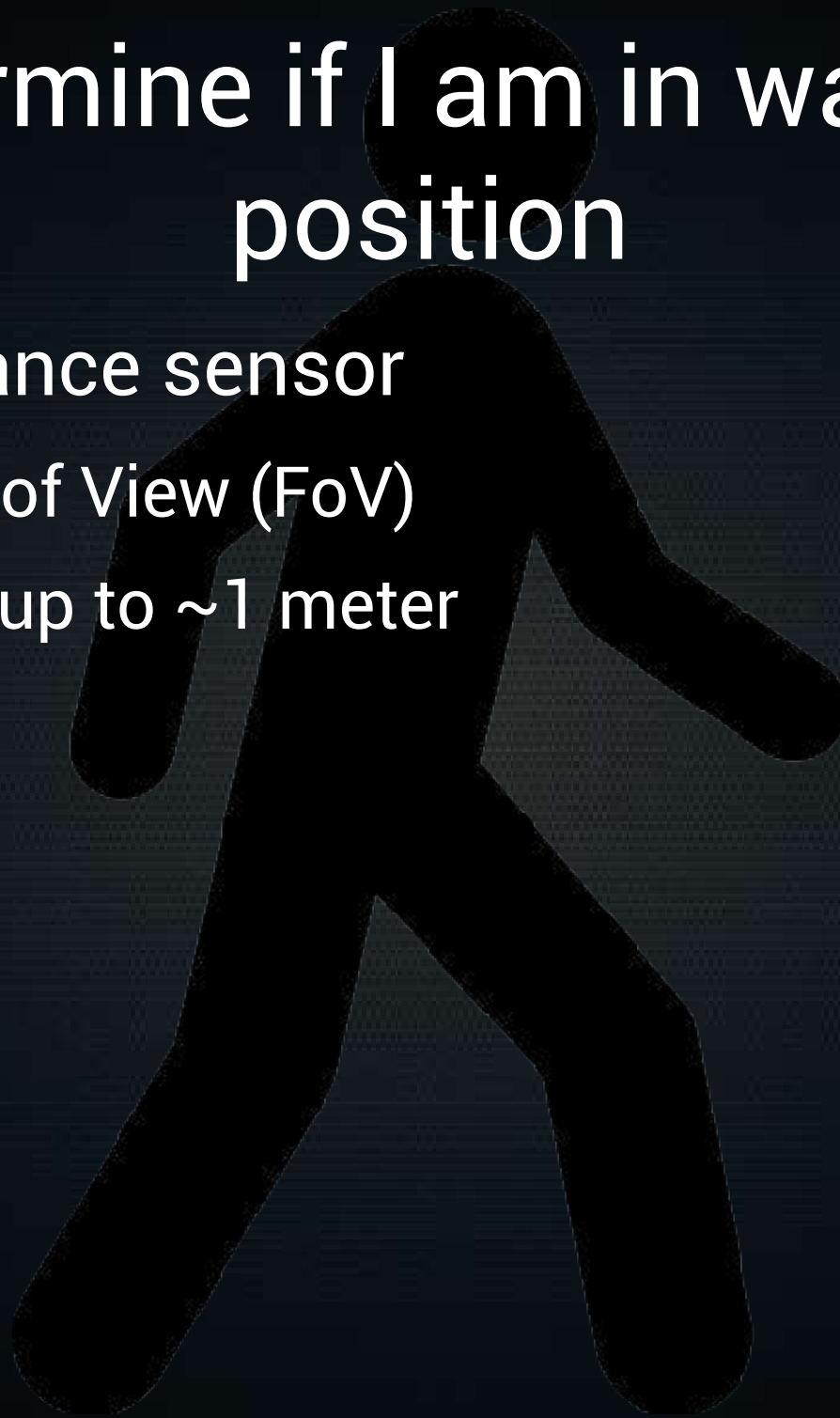


Workstation Side



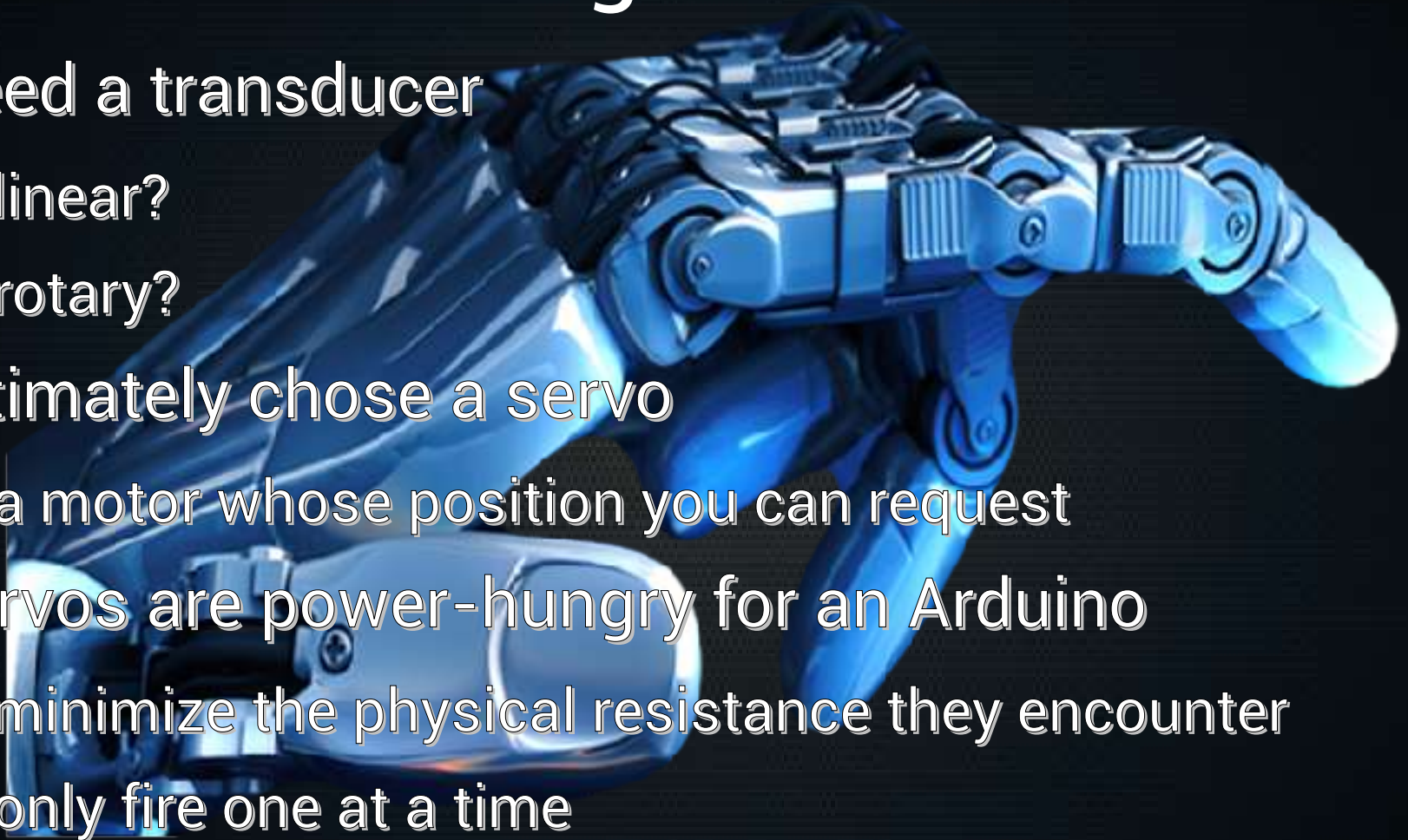
Determine if I am in walking position

- sonic distance sensor
 - 15° Field of View (FoV)
 - accurate up to ~1 meter



Press several buttons without modding the devices

- Need a transducer
 - linear?
 - rotary?
- Ultimately chose a servo
 - a motor whose position you can request
- Servos are power-hungry for an Arduino
 - minimize the physical resistance they encounter
 - only fire one at a time



Trigger workstation behavior based on my presence/absence on treadmill

- Arduino's USB port is a serial interface
 - send status output over serial to my workstation
 - read status with a daemon
 - react to status with a Sikuli script

Throw a relay to run the fan

- 120VAC + relay + inductive load + solder = scary
 - I chose a pre-made Arduino-ready solution for this



Provide a manual override

- Position of the servos handles some of this
 - servo arm leaves room



Provide a manual override

- One button enters override mode; the other re-enters automatic mode.
 - Red forces a “no walker present” status and disables distance sensor polling
 - Green re-enables distance sensor polling

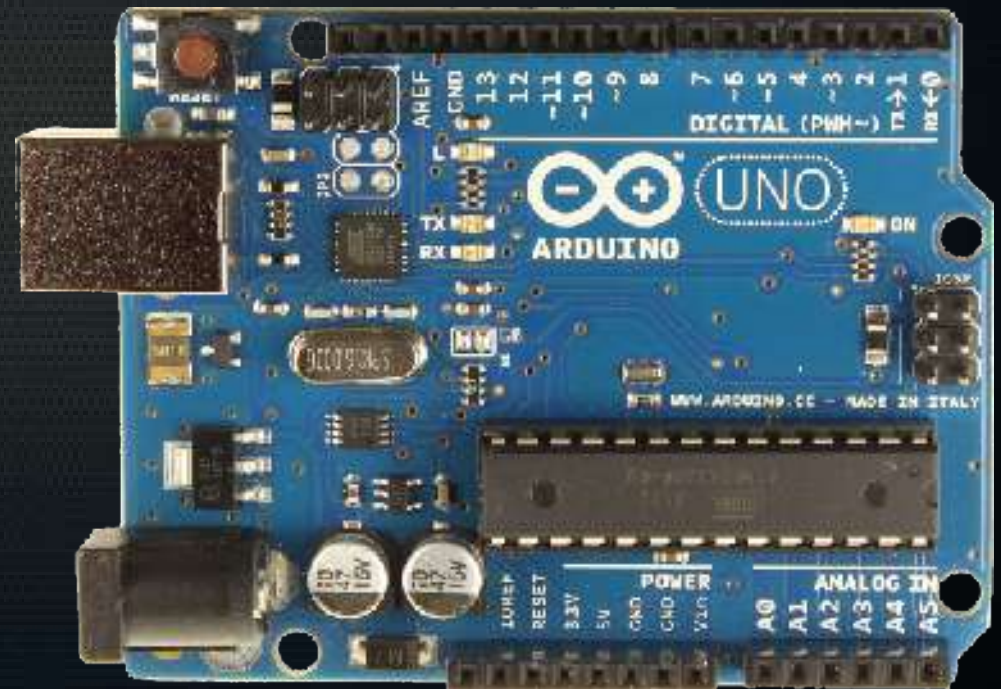


Provide a manual override

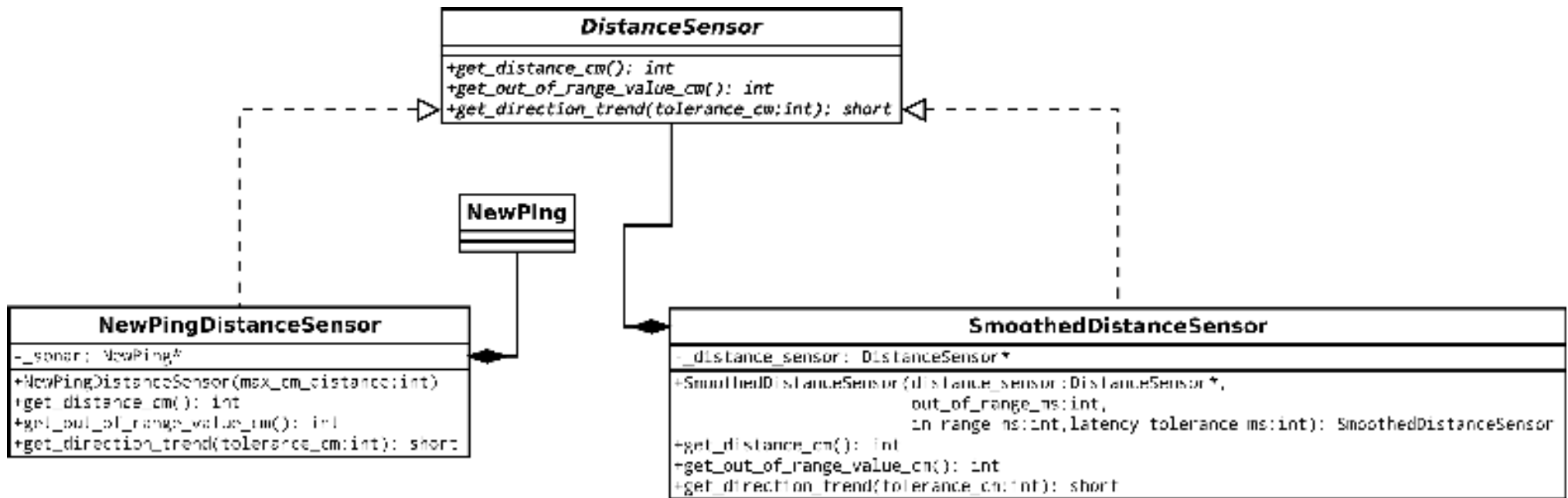
- Added a status LED to communicate override mode state.

Arduino Code

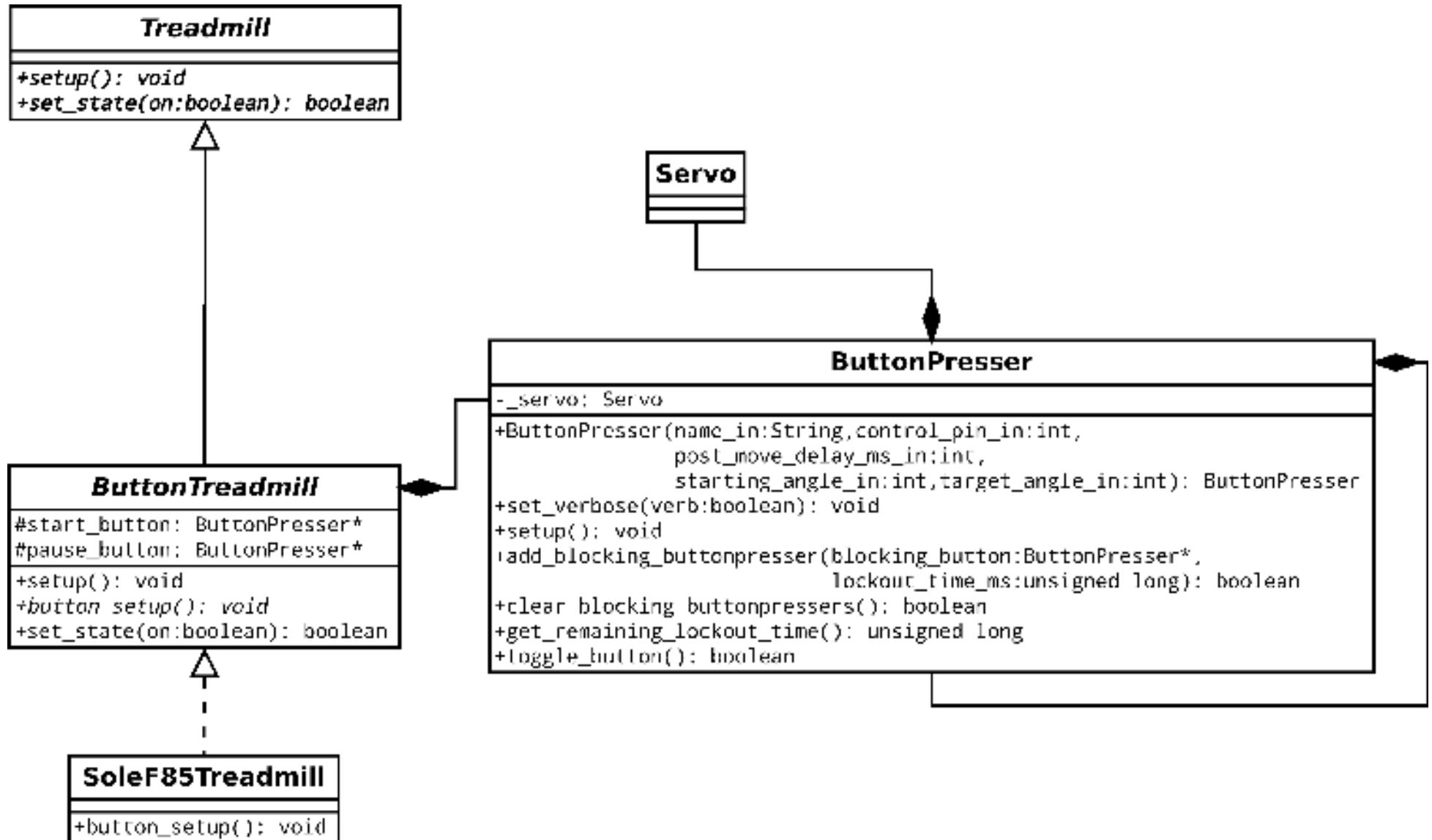
- The DeskAttendant project utilizes several classes to toggle buttons and read from the distance sensor.



Arduino Code

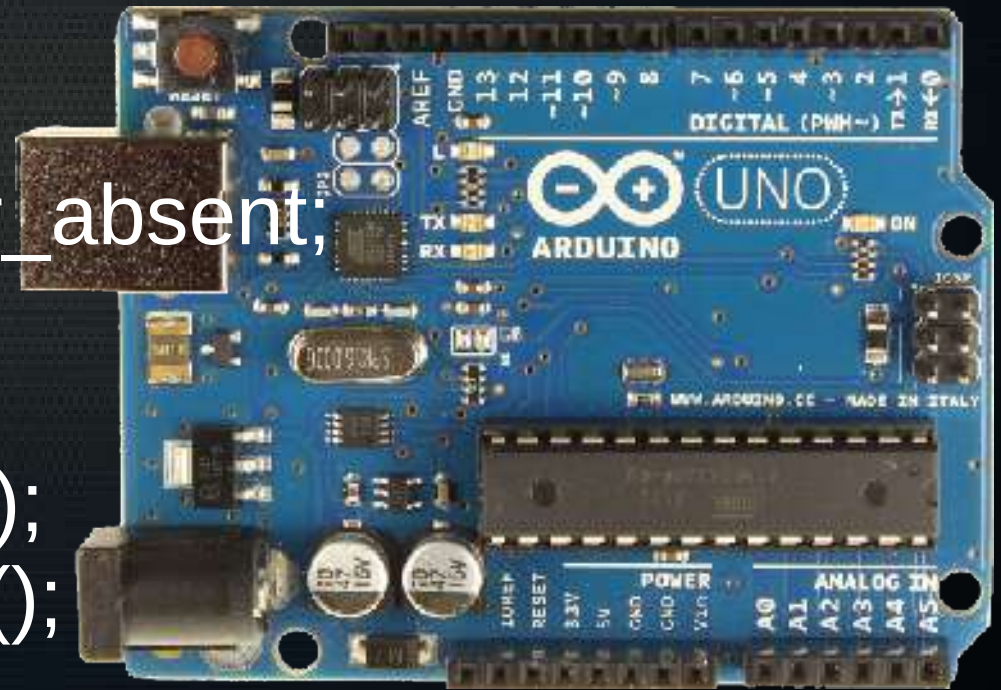


Arduino Code

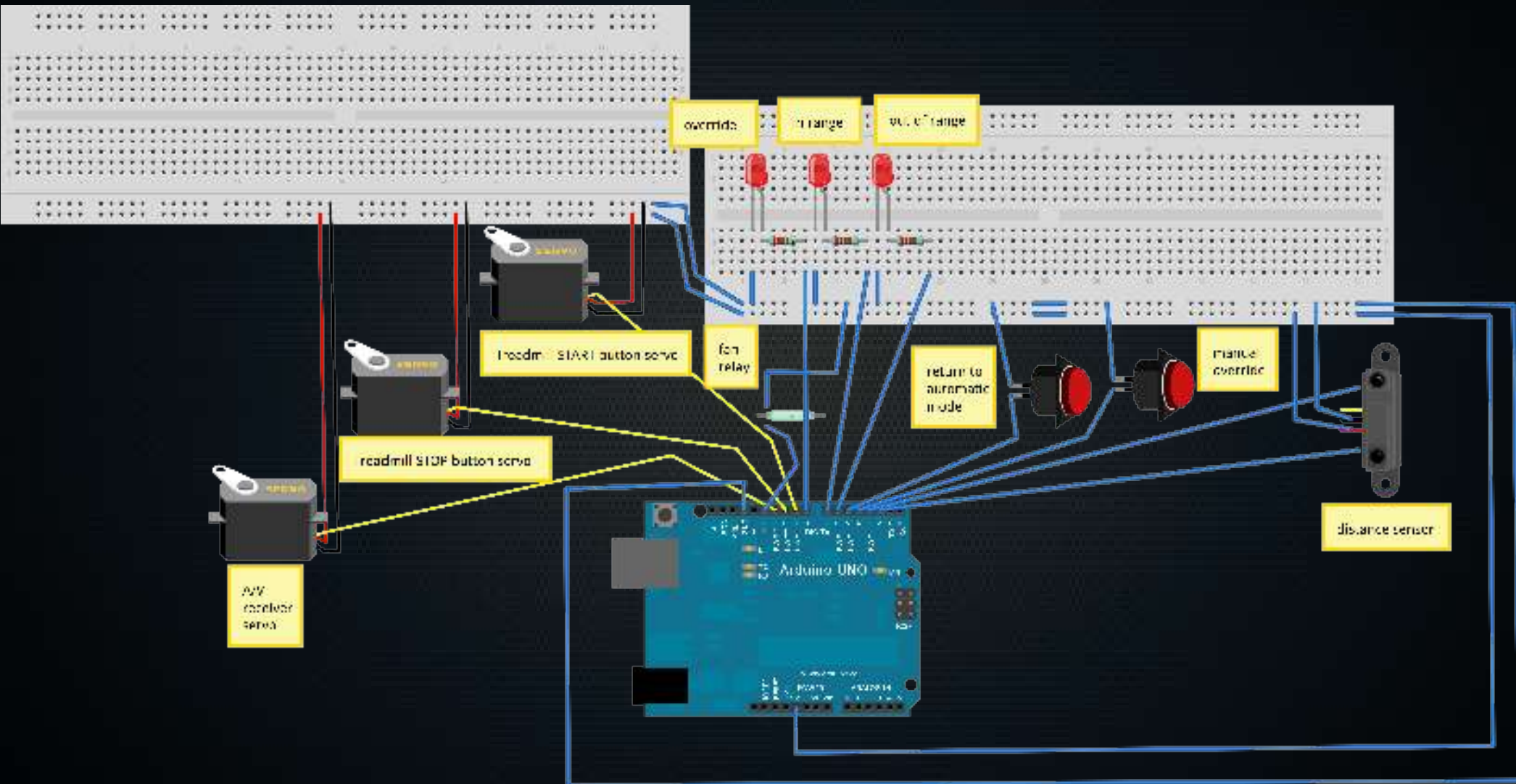


Arduino Code

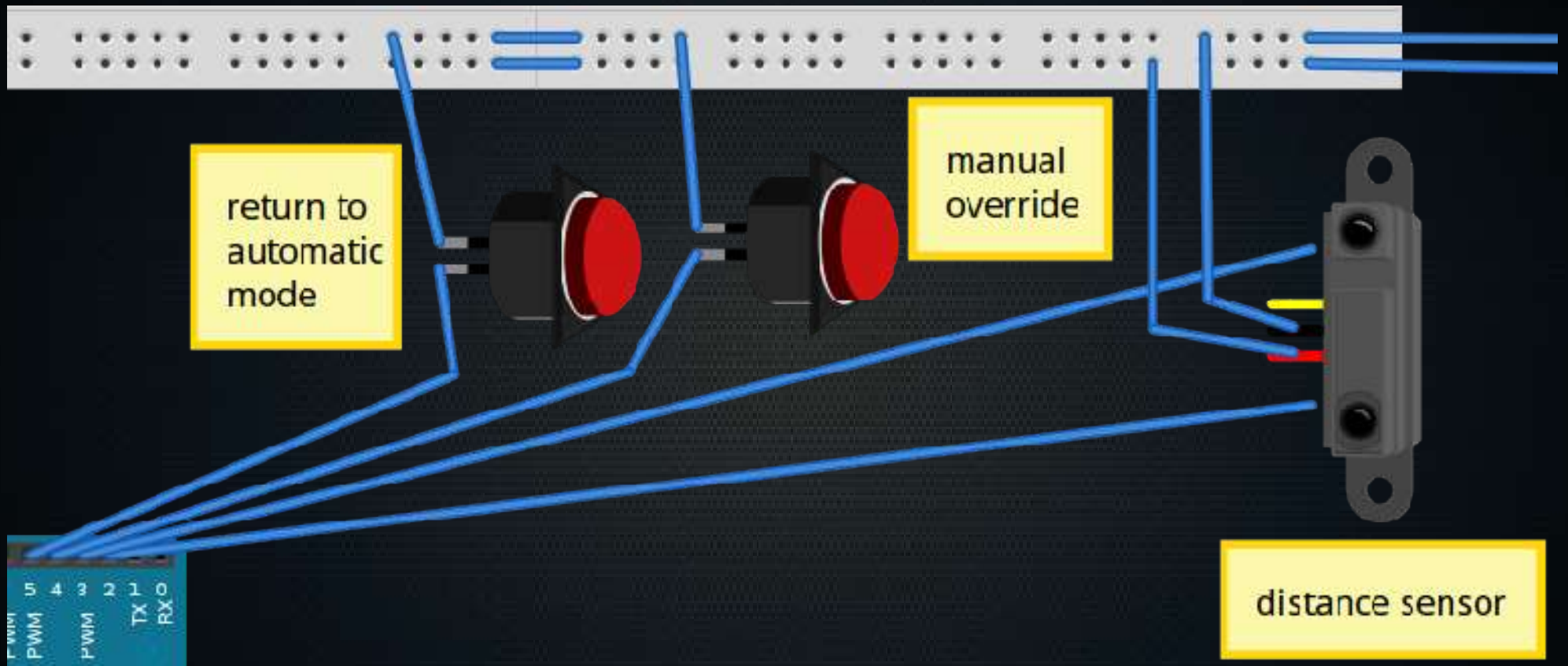
```
void loop() {  
  if (!manual_override) {  
    distance_loop();  
    get_status_from_distance();  
  }  
  else {  
    new_status = walker_absent;  
  }  
  
  process_status_loop();  
  override_button_loop();  
}
```



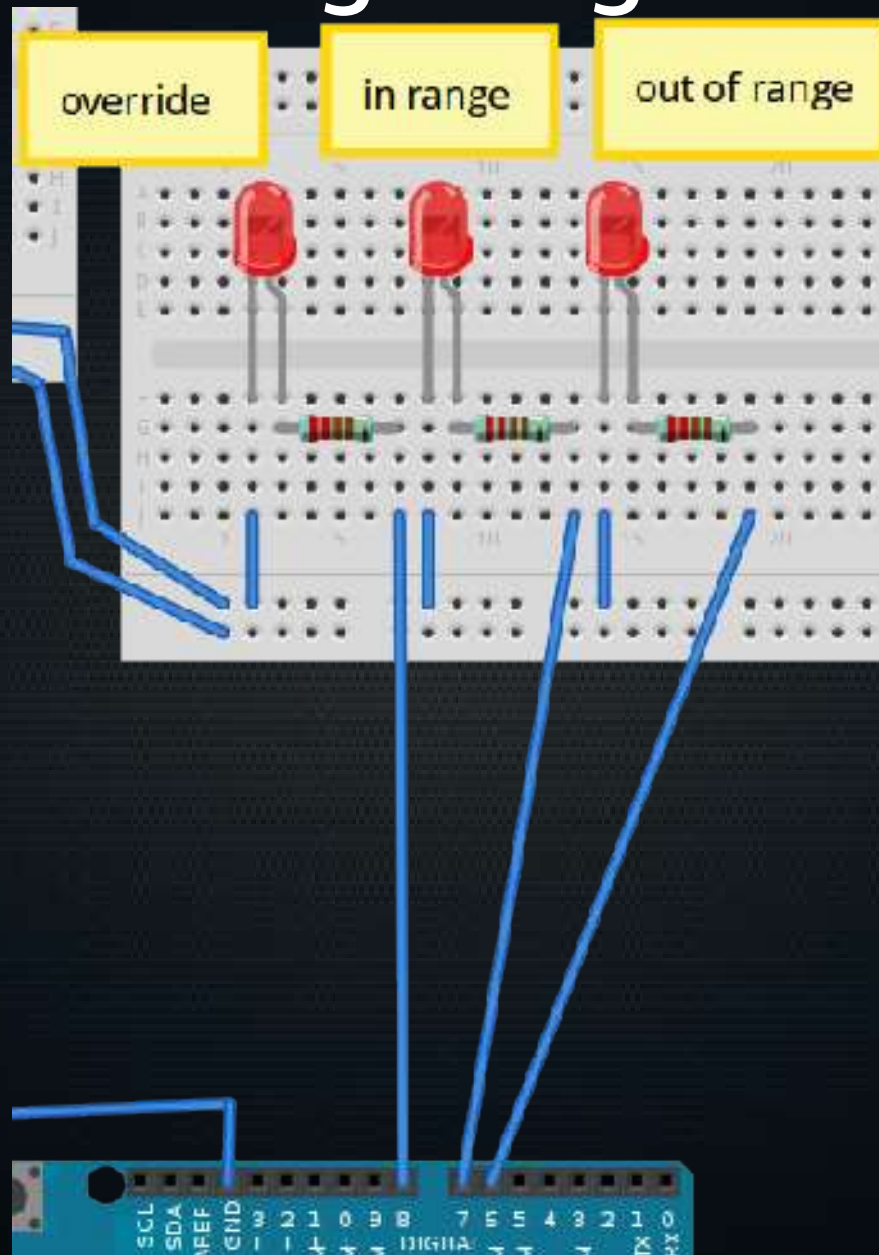
Arduino Wiring Diagram (overview)



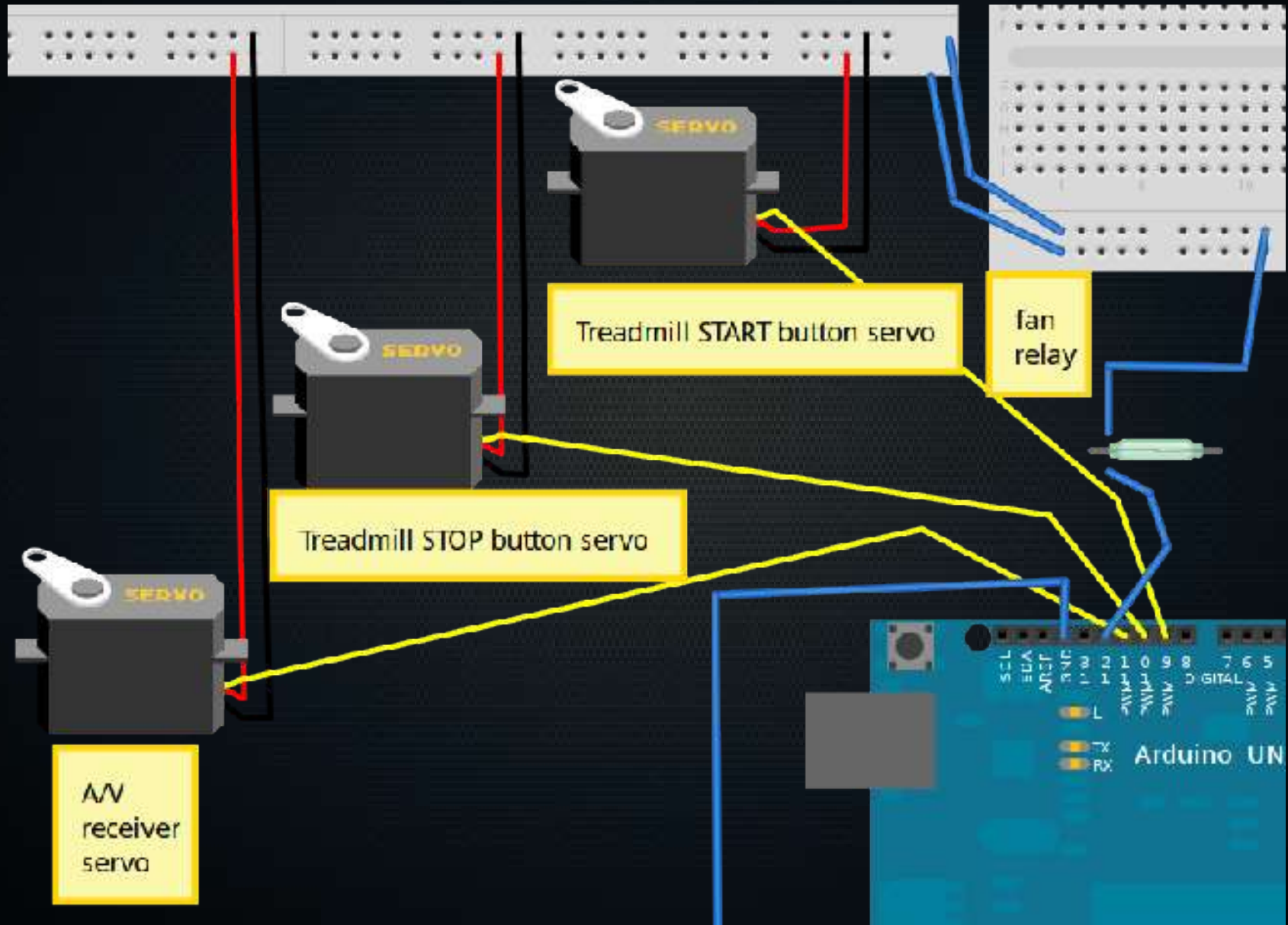
Arduino Wiring Diagram (inputs)



Arduino Wiring Diagram (status)



Arduino Wiring Diagram (actions)



Sources and Parts List

- See

<https://github.com/Human/DeskAttendant>

Questions?