

Time Sitting Tracker (TST) Instructable

Physical and Tangible Computing

Eric Rudat and Prayash Thapa

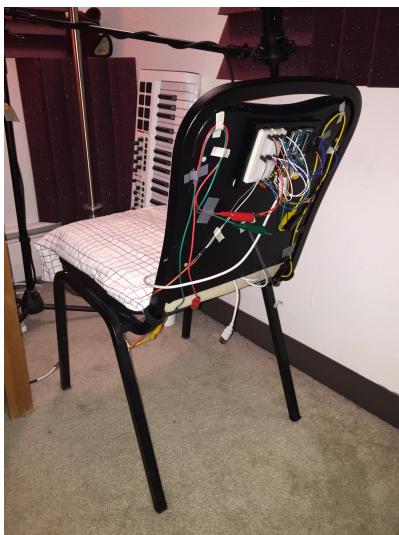
Brief Introduction

The time sitting tracker is a chair designed to notify users if they have been in a seated position for too long. Studies show that excessive sitting, especially in a continuous fashion, can reduce the average person's life expectancy. Thus, we attempt to alleviate this issue by tracking the amount of time people sit for in one period.

The chair features a microcontroller, in the form of an Arduino Uno, which handles input from four force sensitive resistor (FSRs) featured on the chair. This data is then passed to a Raspberry Pi to be parsed, via a serial port connection. Then, after a user-specified amount of time the Pi will send a text message to the user's phone, notifying them that they have been sitting for too long.

Materials Needed

- Raspberry Pi 2 Model B
 - Prior knowledge of how to obtain IP address of Raspberry Pi assumed
- USB cord
- Cell Phone with texting capabilities
- Custom Chair featuring:
 - Arduino Uno mounted on the back
 - 4 Force Sensitive Resistors (FSRs) attached to the Aruduino
 - Wiring to connect all the components
 - Foam for added comfort
 - Cover for aesthetic purposes



Step-by-Step Instructable

1. Login into your Raspberry Pi remotely using **ssh**.

```
ssh pi@xxx.xxx.xxx.xxx  
where 'xxx.xxx.xxx.xxx' is your Pi's IP address
```

2. Enter your Pi's password. *Default = raspberry*

```
pi@xxx.xxx.xxx.xxx's password:
```

3. You should now see something like this in your terminal window.

```
Linux raspberrypi 3.18.7-v7+ #755 SMP PREEMPT Thu Feb 12  
17:20:48 GMT 2015 armv7l  
  
The programs included with the Debian GNU/Linux system  
are free software;  
the exact distribution terms for each program are  
described in the  
individual files in /usr/share/doc/*copyright.  
  
Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to  
the extent  
permitted by applicable law.  
Last login: Wed Apr 29 01:57:15 2015 from 192.168.2.1  
pi@raspberrypi ~ $
```

4. Clone the following repository into any directory in your Pi's file system.

```
git clone https://github.com/erudat/TimeSittingTracker.git
```

5. Download the following <https://github.com/korylprince/pygvoicelib>
6. Follow the usage instructions, which can be found by opening up a Python 2 shell and entering the following commands.

```
>>> import pygvoicelib  
>>> help(pygvoicelib)
```

7. Add the file *pygvoicelib.py* to the repository you created in step 4.
8. Upload the following code from any computer to the Arduino Uno, using the Arduino Application.

```
int fsrPin_0 = 0;  
int fsrPin_1 = 1;  
int fsrPin_2 = 2;  
int fsrPin_3 = 3;  
  
int fsrRead_0;  
int fsrRead_1;  
int fsrRead_2;  
int fsrRead_3;  
  
int average01;  
int average23;
```

```
void setup(void) {
    Serial.begin(9600);
    pinMode(13, OUTPUT);
    pinMode(12, OUTPUT);
}

void loop(void) {
    fsrRead_0 = analogRead(fsrPin_0);
    fsrRead_1 = analogRead(fsrPin_1);
    fsrRead_2 = analogRead(fsrPin_2);
    fsrRead_3 = analogRead(fsrPin_3);

    average01 = (fsrRead_0 + fsrRead_1) / 2;
    average23 = (fsrRead_2 + fsrRead_3) / 2;

    // Conditional Handles Sending Data over Serial Port
    if (average01 >= average23) {
        Serial.println(average01);
    }
    else {
        Serial.println(average23);
    }

    // Conditional Handles Turning LED On/Off
    if (fsrRead_0 > 0 && fsrRead_1 > 0) {
        digitalWrite(13, HIGH);
    } else {
        digitalWrite(13, LOW);
    }

    if (fsrRead_2 > 0) {
        digitalWrite(12, HIGH);
    } else {
        digitalWrite(12, LOW);
    }

    delay(1000);
}
```

9. Connect your Arduino to the Raspberry Pi using the USB cord.



Be sure it is connected to the lower left USB port!

10. From the ssh prompt change directories, **cd**, to the where you cloned the Git repository.

11. Now it is time to execute the python script. From the cloned directory enter this command.

```
python chair.py
```

12. You should now see something like this.

```
Please enter the number you wish  
to send notifications to: xxxxxxxxxxxx  
Please enter the amount of time you wish  
to elapse before notification is sent: xx
```

13. Enter the phone number you wish to receive text messages at.
14. Enter the amount of time (in seconds) you want the chair to track for.
15. The program should now be running, and the chair is ready to use. You will see something like this at the terminal prompt.

```
Running...
To exit press 'CTL-C'
```

16. At any point you can exit the python script by pushing 'CTL-C' at the ssh terminal prompt.