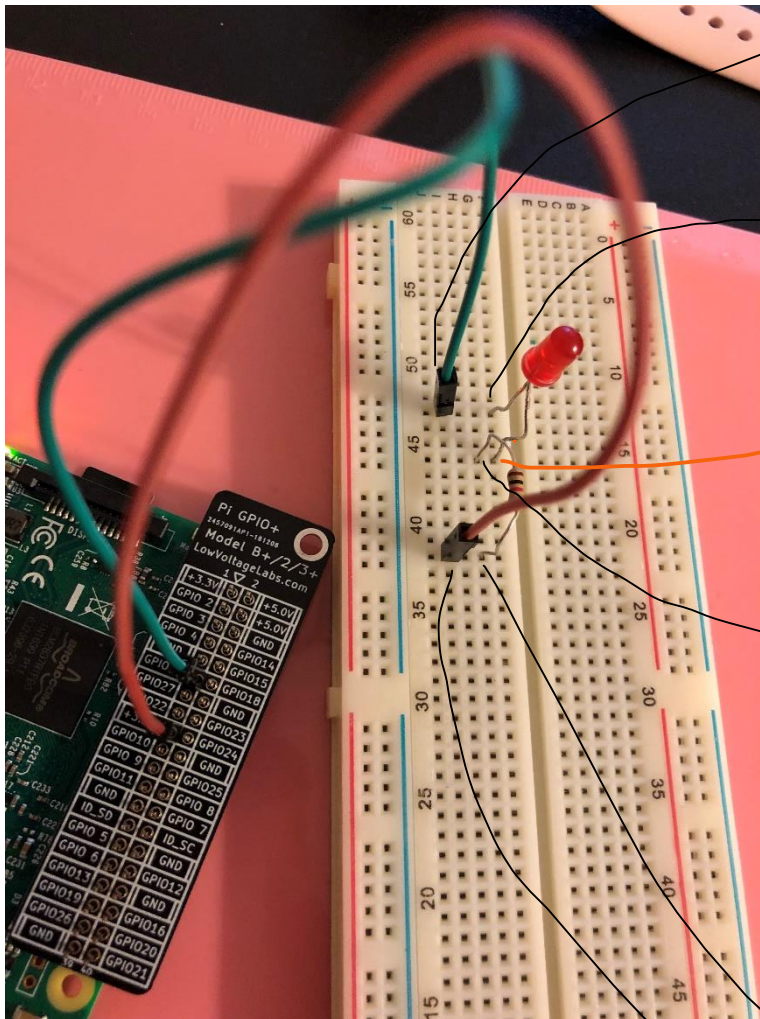


Karishma Kapur

2/12/19

Lab 2

Here is my Breadboard Circuit:



Connecting GPIO 17 to the same pathway as the negative end of the LED

This is the negative end of the LED (cathode)

This is the positive end of the LED (anode)

This is the resistor in the same path as the positive end of the LED, to limit the current given from the breadboard, to make sure the LED light doesn't get blown out. ( $r = v/i$ ) and LED current is roughly 20 mamp.

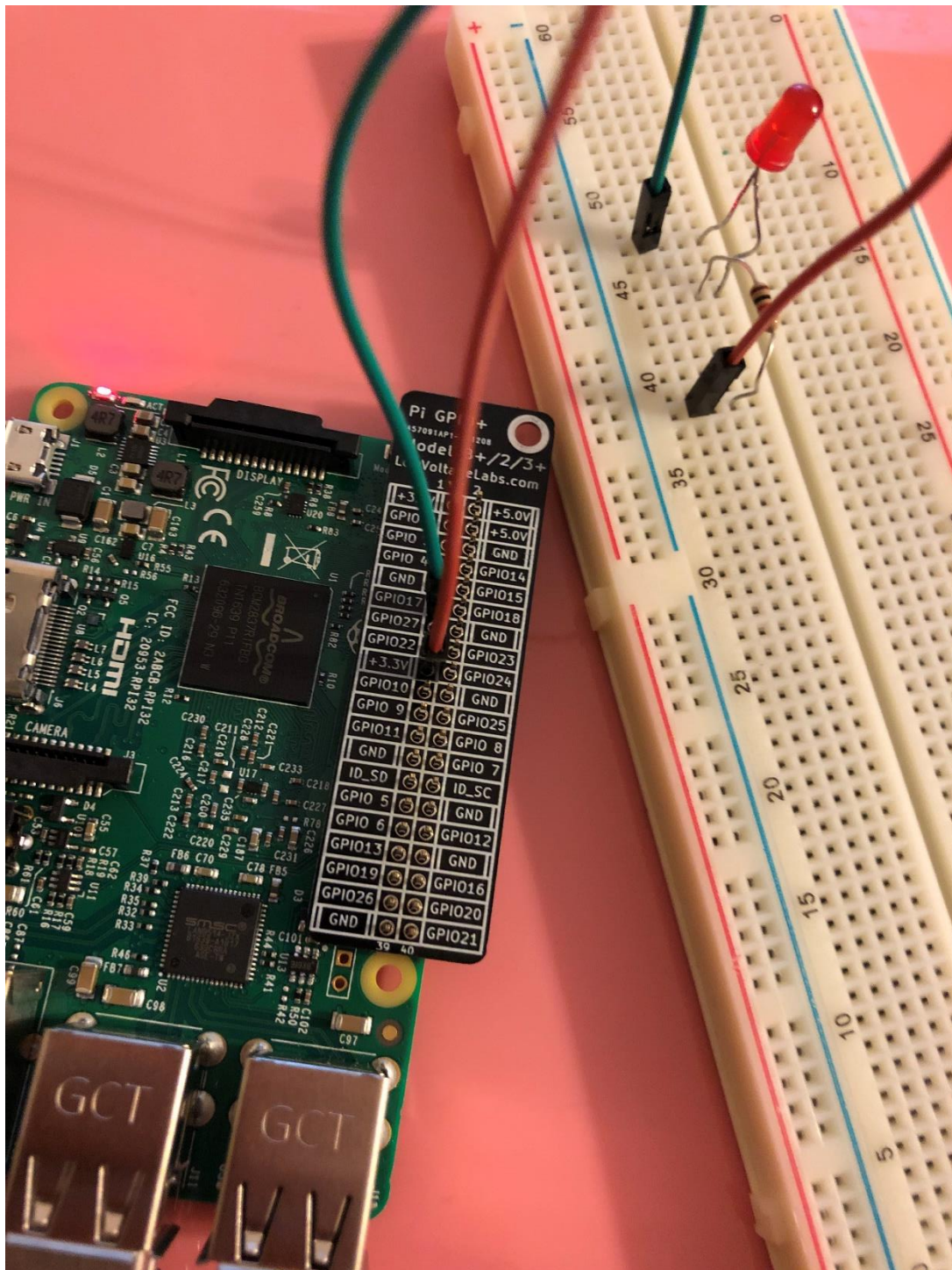
Volt was  $3.3 / 20 \text{ mamp} = 165$  roughly.

I tried to use 220 ohm for the resistor, but it didn't work. So I went up and 920 worked

The other end of resistor, which is also helping to limit the current.

Connecting +3.3V from breadboard to the same pathway as the resistor.

Closer look at breakboard circuit:



Running the code:

1) Compiling code

```
raspberrypi: ~/MyPrograms
enter a letter: k
hangman string is "HANGM"
is your current word: key-----

e enter a letter: b
hangman string is "HANGM"
is your current word: keyb----

se enter a letter: o
hangman string is "HANGM"
is your current word: keybo---

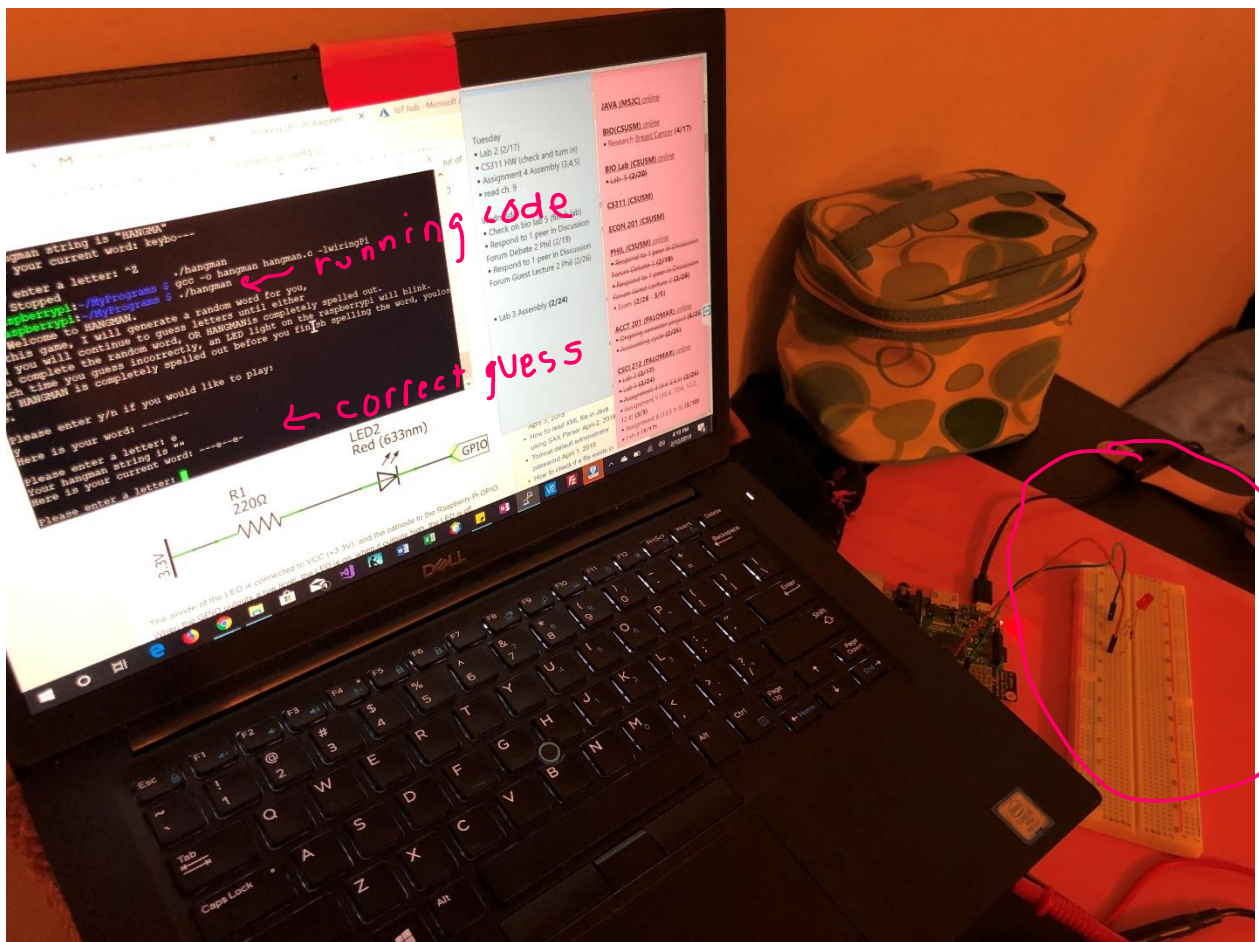
ase enter a letter: q
on
off
r hangman string is "HANGMA"
e is your current word: keybo--

ase enter a letter: ^Z
+ Stopped ./hangman
raspberrypi:~/MyPrograms $ gcc -o hangman hangman.c -lwiringPi
```

Running it with `-lwiringPi` because I need to link the code to make sure the LED light program in `hangman.c` (under function name `turnonLED` and included with header `wiringPi.h`) will work.



- 2) Running the code and telling game I want to play ('y') and entering a correct guess. Light does not turn on because I guessed correctly.



3) I enter another letter, but I guess incorrectly, so LED light turns on

