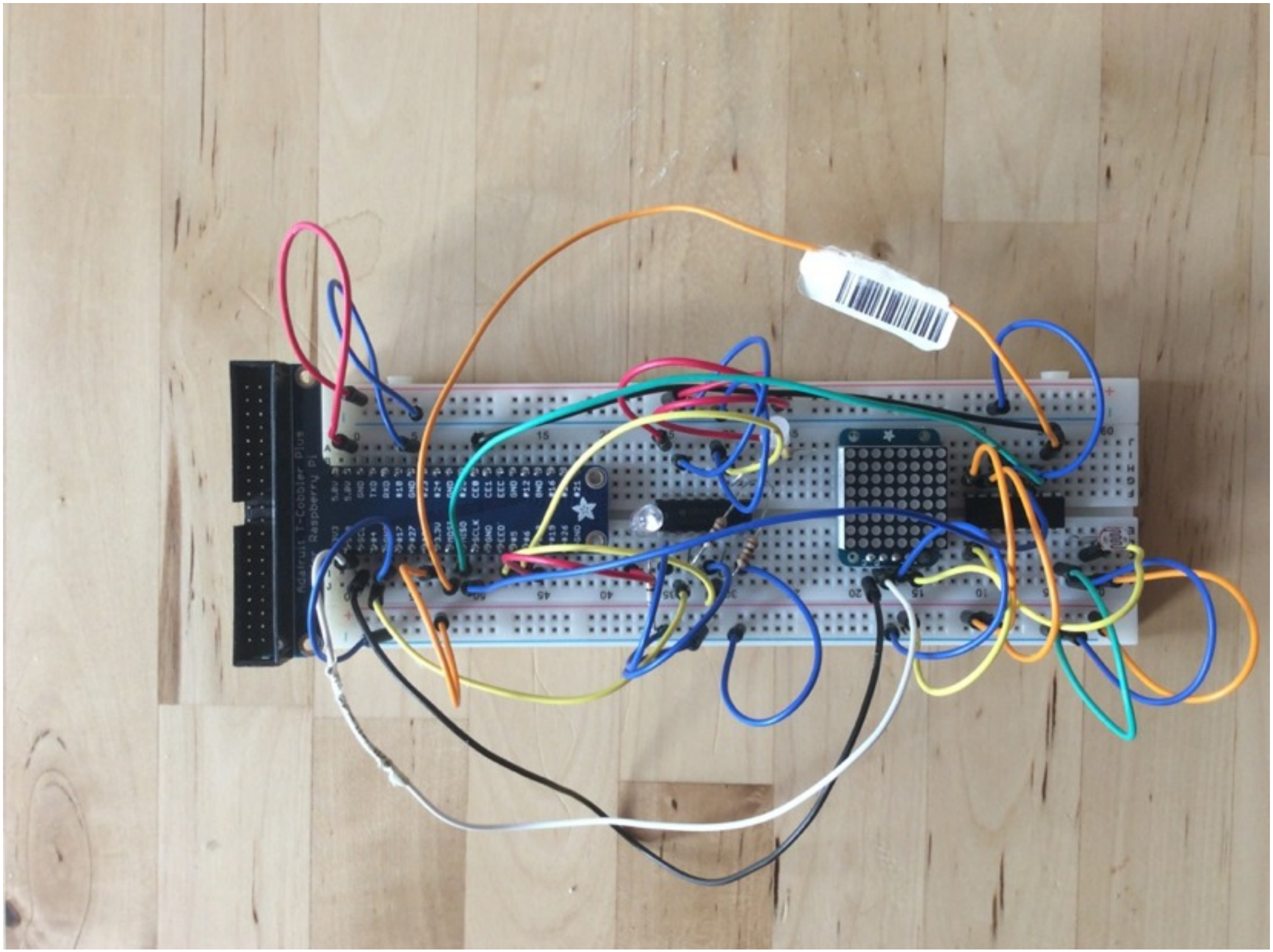
Intro to Raspberry Pi



Intro to Raspberry Pi

Essential Parts

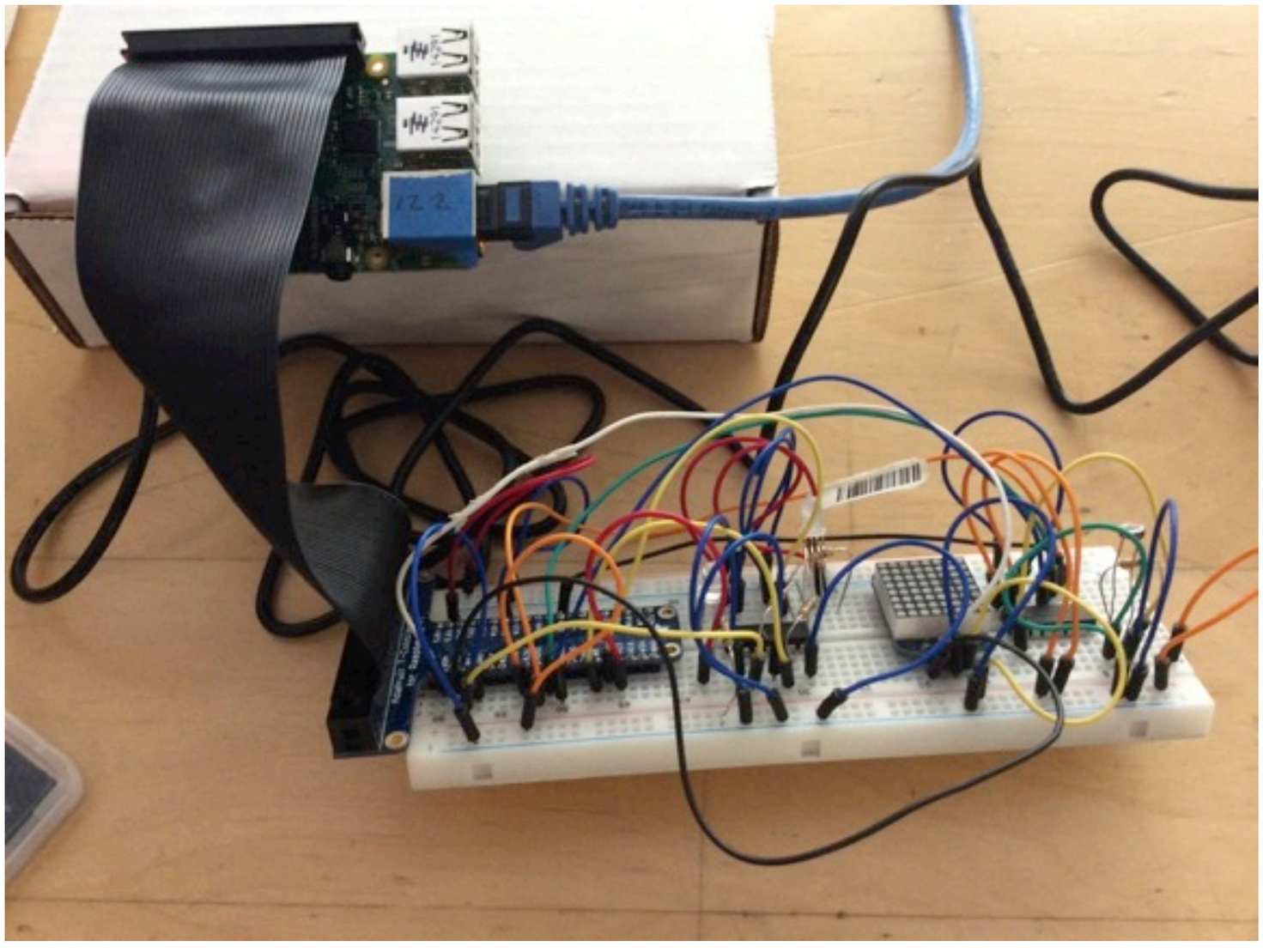
- Network Cable
- µUSB Power Cord
- µSD Card (already installed)

Intro to Raspberry Pi

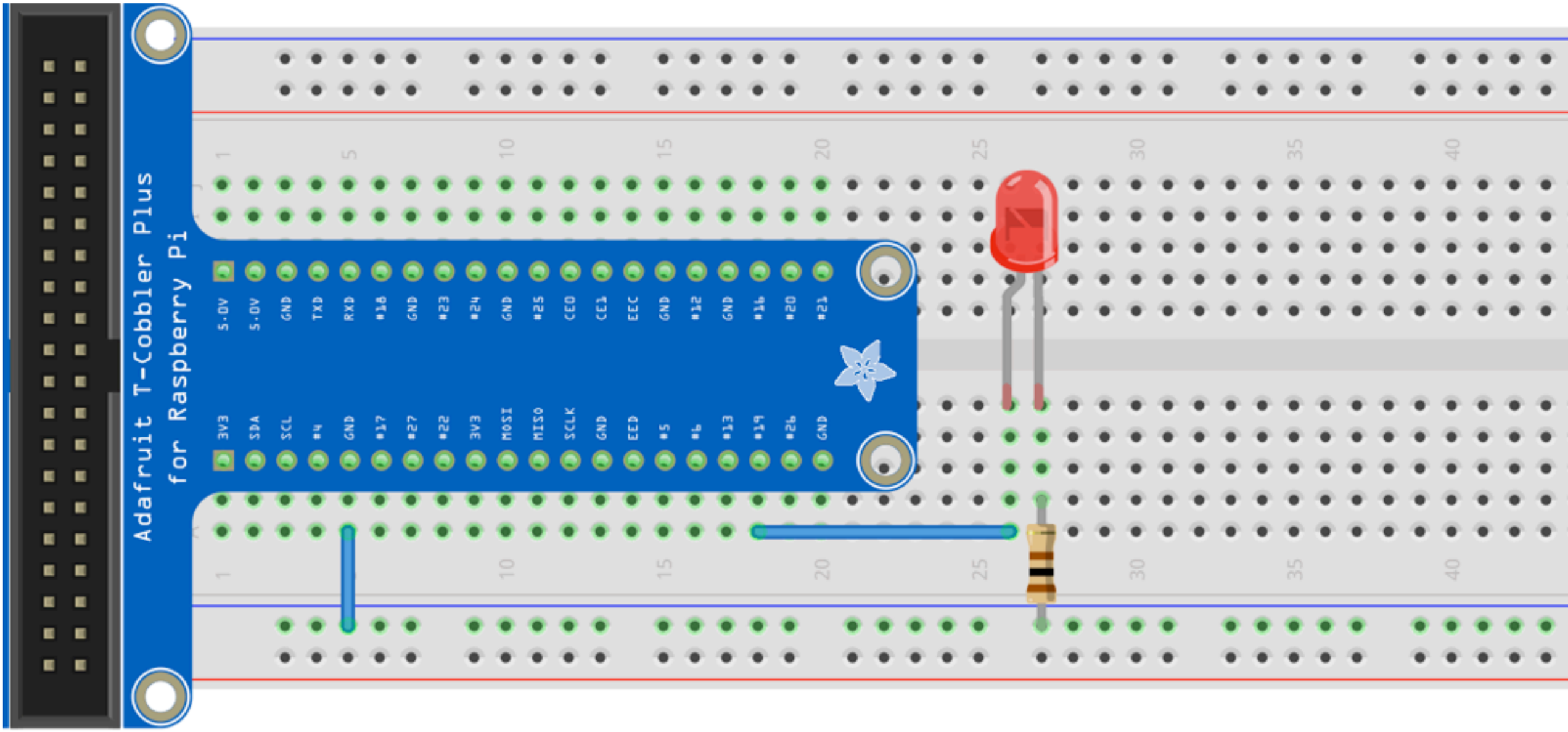
```
dmduser$ ssh pi@172.16.36.XXX
pi@raspberrypi ~ $ sudo python
Python 2.7.3 (default, Mar 18 2014, 05:13:23)
[GCC 4.6.3] on linux2
Type "help", "copyright", "credits" or "license" for
more information.
>>>
```

Intro to Raspberry Pi

Connect Cable to RasPi towards board
Cable only fits into Breadboard 1 way



GPIO & Red LED

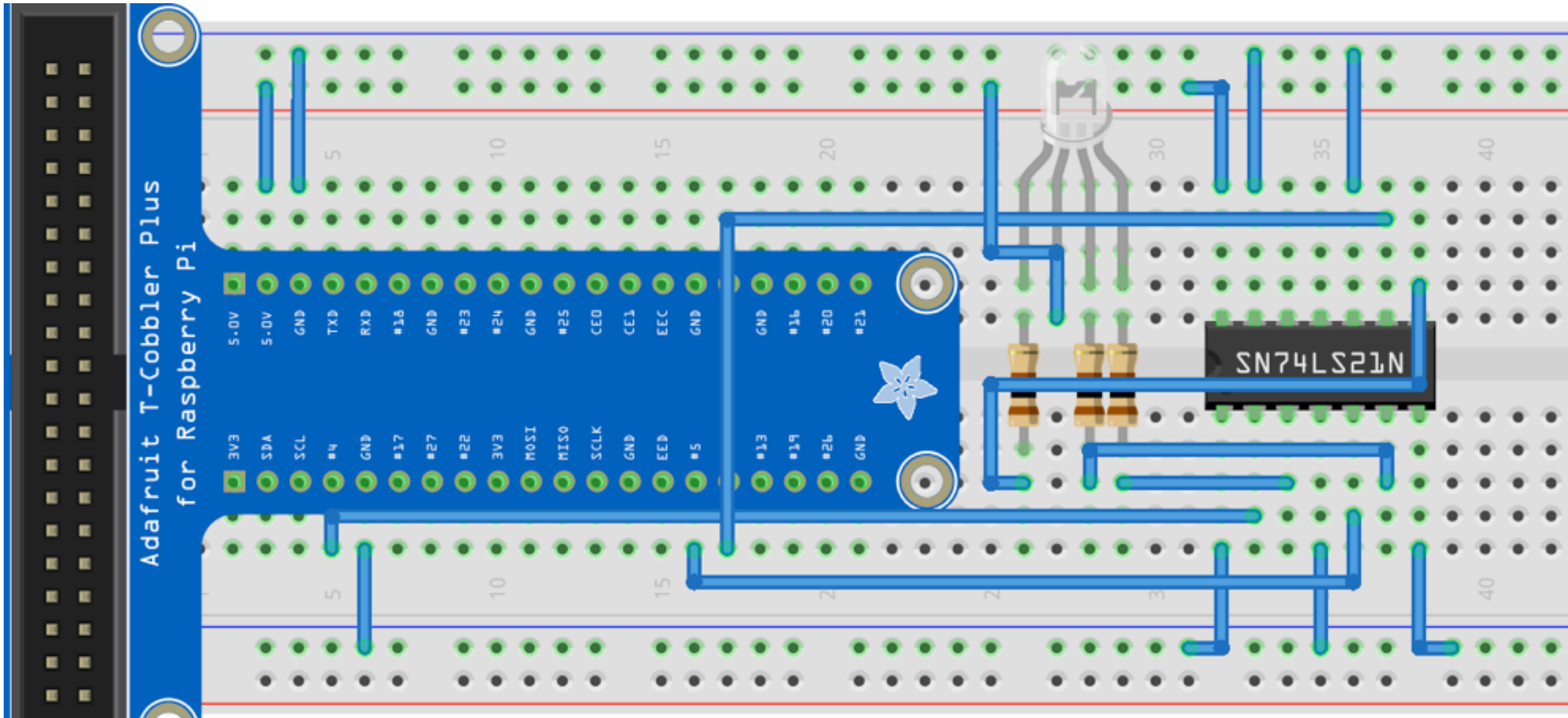


- Long leg of LED to Pin 13
- Short leg to resistor, then to Ground

GPIO & Red LED

```
pi@raspberrypi ~ $ sudo python
>>> import RPi.GPIO as GPIO
>>> GPIO.setmode(GPIO.BCM)
>>> GPIO.setup(13, GPIO.OUT)
>>> GPIO.output(13, True)
```

Level Shifter & RGB LED



- Longest leg of LED to 5V
- Other legs to Resistors, then Level Shifter
- Level Shifter inputs to Pins 4, 5, & 6
- 5 Ground lines on Level Shifter

Level Shifter & RGB LED

```
>>> GPIO.setup(4, GPIO.OUT)
>>> GPIO.setup(5, GPIO.OUT)
>>> GPIO.setup(6, GPIO.OUT)
>>> GPIO.output(4, True)
>>> GPIO.output(5, True)
```

Level Shifter & RGB LED

```
>>> r = GPIO.PWM(6, 100)
>>> r.start(50)
>>> r.ChangeDutyCycle(100)
>>> GPIO.output(5, False)
>>> g = GPIO.PWM(5, 100)
>>> g.start(50)
>>> b = GPIO.PWM(4, 100)
```

Logic Voltage Levels

What V counts as True?

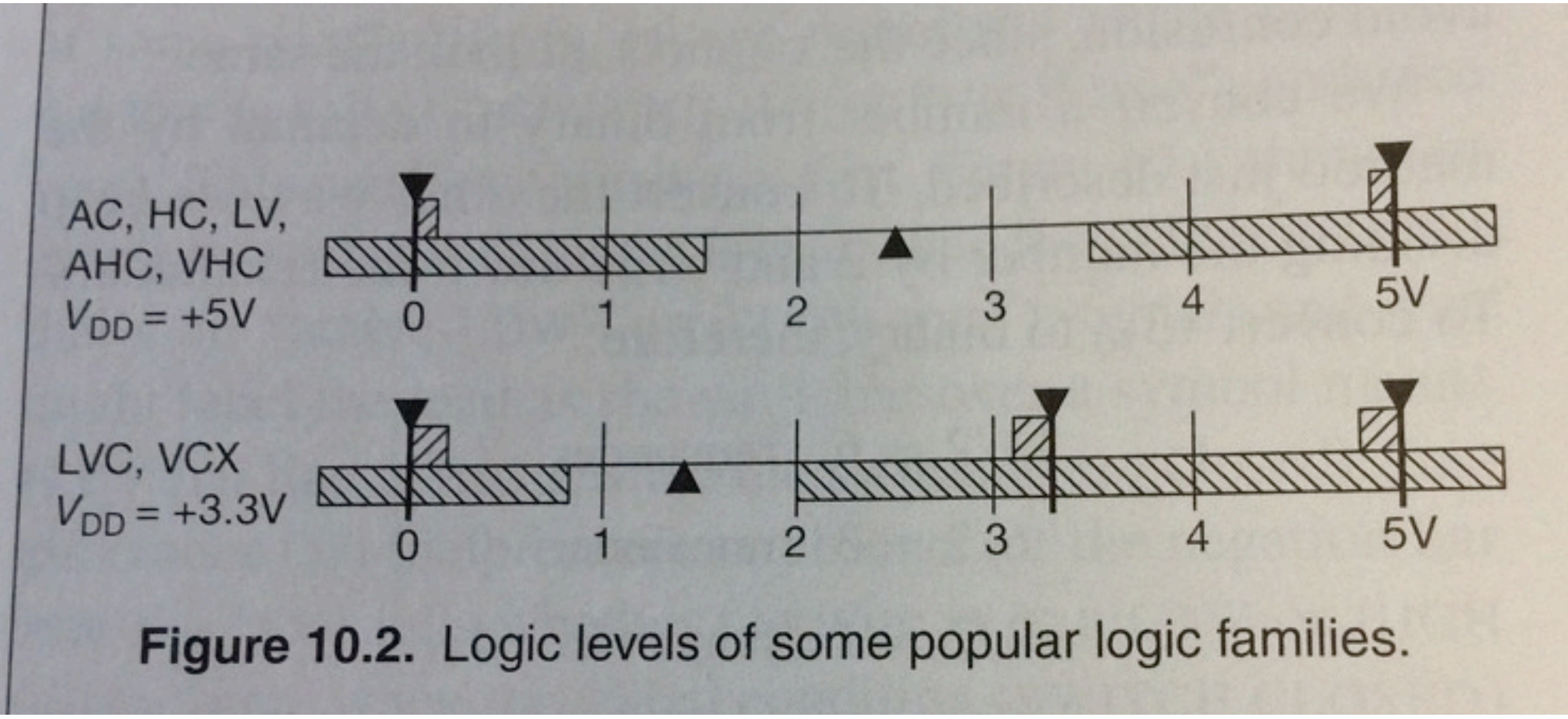


Figure 10.2. Logic levels of some popular logic families.

Atom & rmate

Atom text editor (on Mac)
Packages menu > Remote Atom > Start Server

```
dmduser$ ssh -R 52698:localhost:52698 pi@172.16.36.XXX
pi@raspberrypi ~ $ rmate rgb.py
```

Setting up a new RasPi

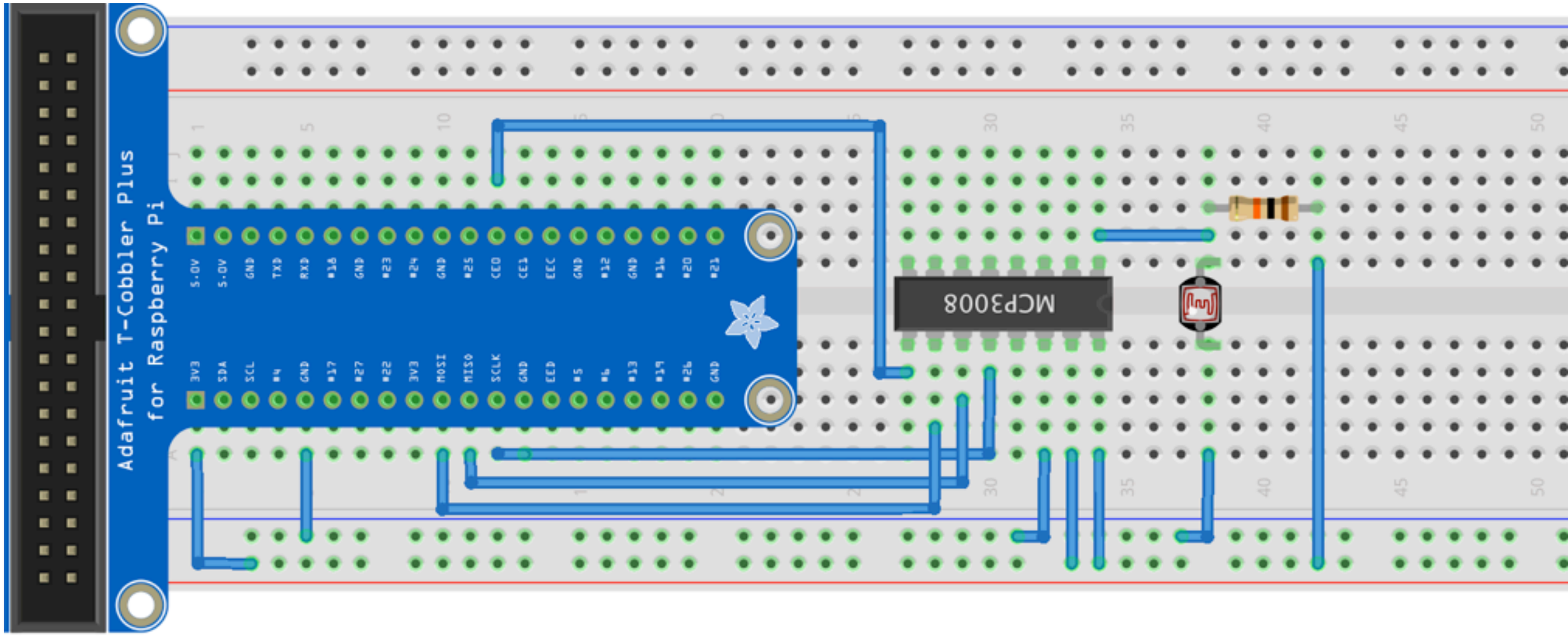
I did a few things for you before class:

Raspian Linux https://www.raspberrypi.org/downloads/
copy to SD card

Adafruit Occidentalis Bootstrapper
https://github.com/adafruit/Adafruit-Pi-Finder
(IP Address, more hardware drivers)

sudo gem install rmate

Photoresistor, ADC, & SPI



- Ground -> Photoresistor -> Resistor -> 3.3V
- ADC input -> junction Photoresistor & Resistor
- ADC connects to SCLK, MISO, MOSI, CE0
- 8 analog inputs on one side of ADC

SPI

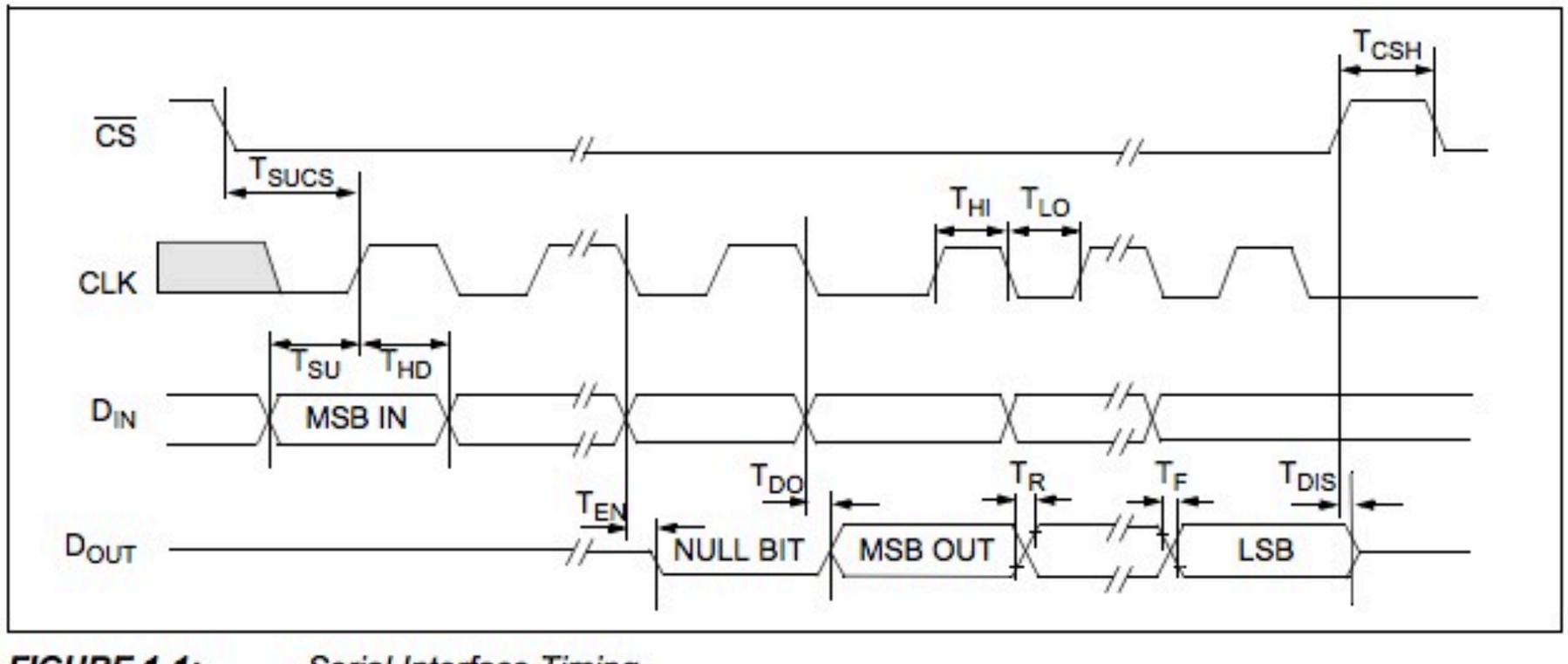


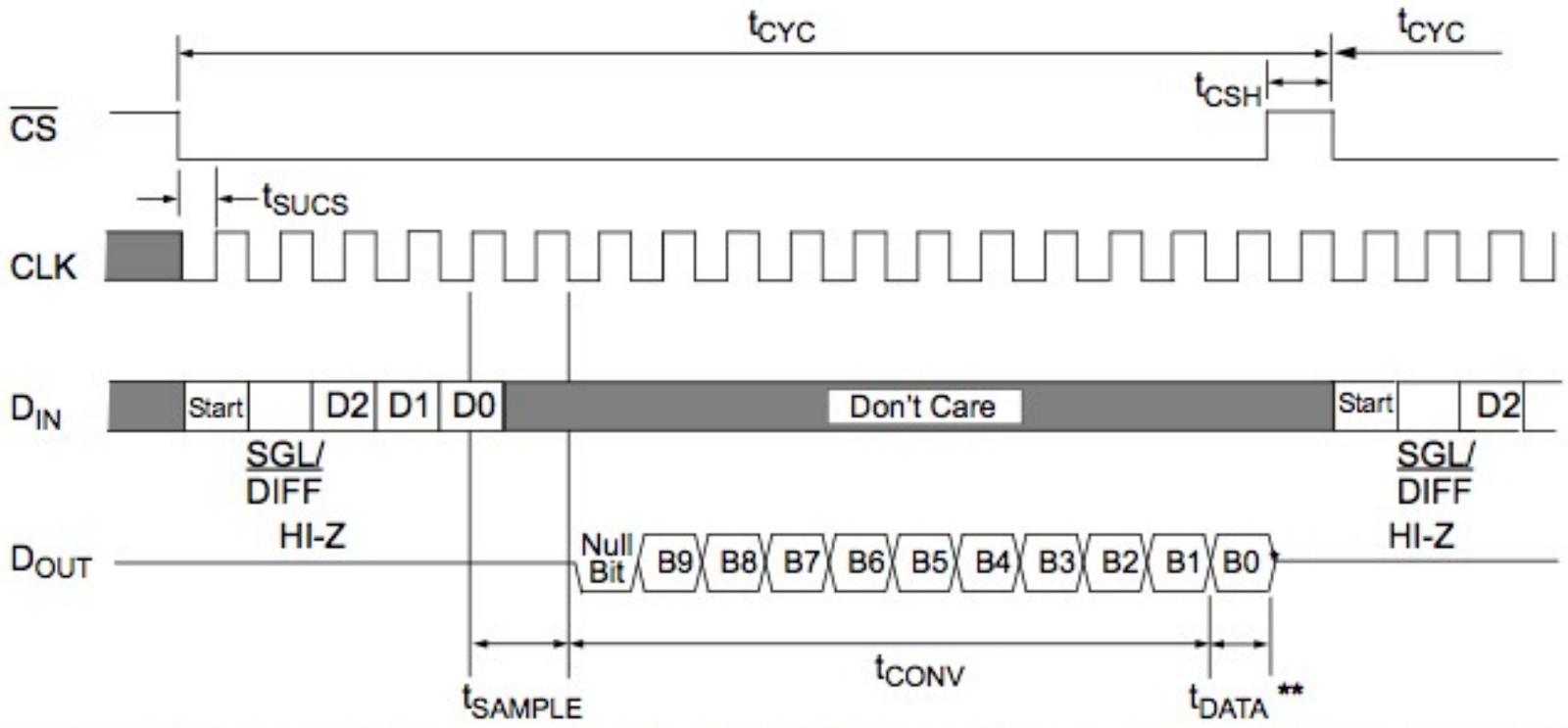
FIGURE 1-1: *Serial Interface Timing.*

- Chip Select Pin - one per chip, other pins shared
- Clock - read data on rising edge of clock
- MOSI - from RasPi to ADC
- MISO - from ADC to RasPi

MCP3008 ADC

TABLE 5-2: CONFIGURE BITS FOR THE MCP3008

Control Bit Selections				Input Configuration	Channel Selection
Single /Diff	D2	D1	D0		
1	0	0	0	single-ended	CH0
1	0	0	1	single-ended	CH1
1	0	1	0	single-ended	CH2
1	0	1	1	single-ended	CH3
1	1	0	0	single-ended	CH4
1	1	0	1	single-ended	CH5
1	1	1	0	single-ended	CH6
1	1	1	1	single-ended	CH7
0	0	0	0	differential	CH0 = IN+ CH1 = IN-
0	0	0	1	differential	CH0 = IN- CH1 = IN+
0	0	1	0	differential	CH2 = IN+ CH3 = IN-
0	0	1	1	differential	CH2 = IN- CH3 = IN+
0	1	0	0	differential	CH4 = IN+ CH5 = IN-
0	1	0	1	differential	CH4 = IN- CH5 = IN+
0	1	1	0	differential	CH6 = IN+ CH7 = IN-
0	1	1	1	differential	CH6 = IN- CH7 = IN+



* After completing the data transfer, if further clocks are applied with $\overline{CS}$ low, the A/D converter will output LSB first data, then followed with zeros indefinitely. See Figure 5-2 below.

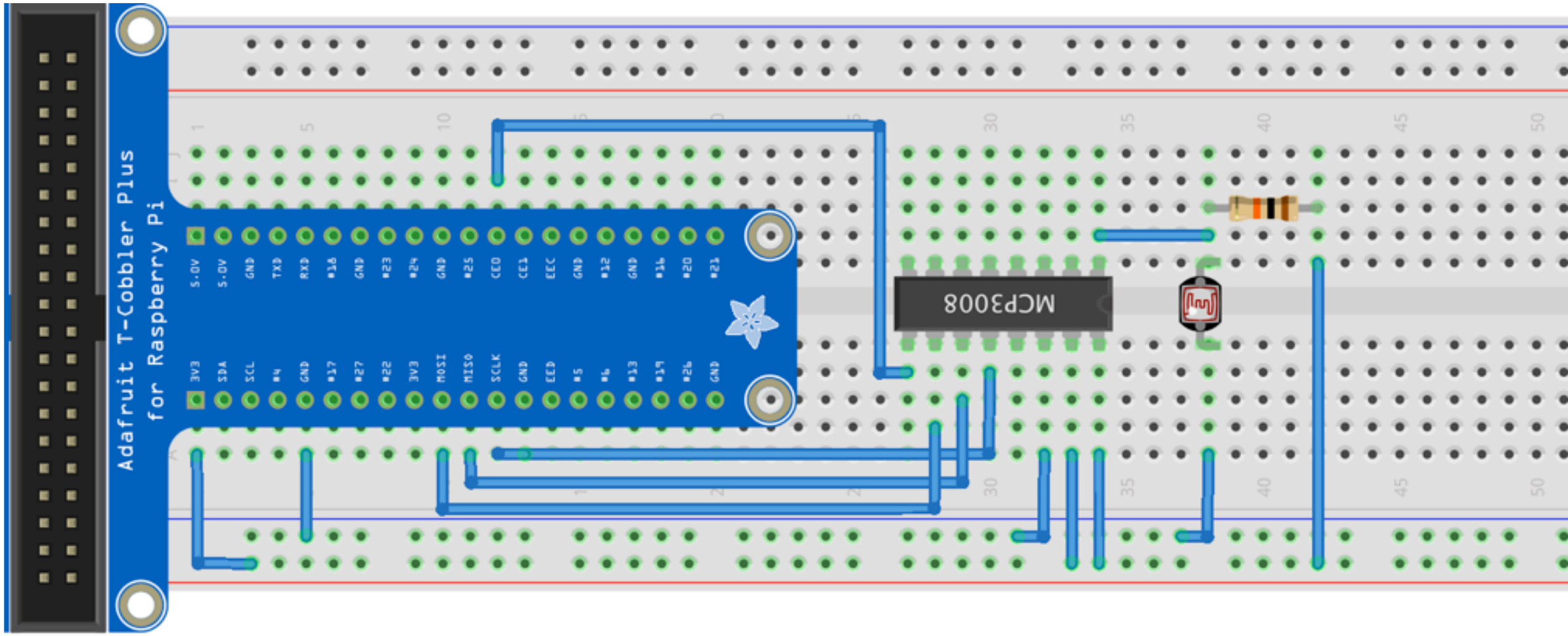
** t_{DATA}: during this time, the bias current and the comparator powers down while the reference input becomes a high-impedance node.

FIGURE 5-1: *Communication with the MCP3004 or MCP3008.*

Datasheet has all the details:

- How many bits per message
- What they all mean

Photoresistor, ADC, & SPI



- Ground -> Photoresistor -> Resistor -> 3.3V
- ADC input -> junction Photoresistor & Resistor
- ADC connects to SCLK, MISO, MOSI, CE0
- 8 analog inputs on one side of ADC

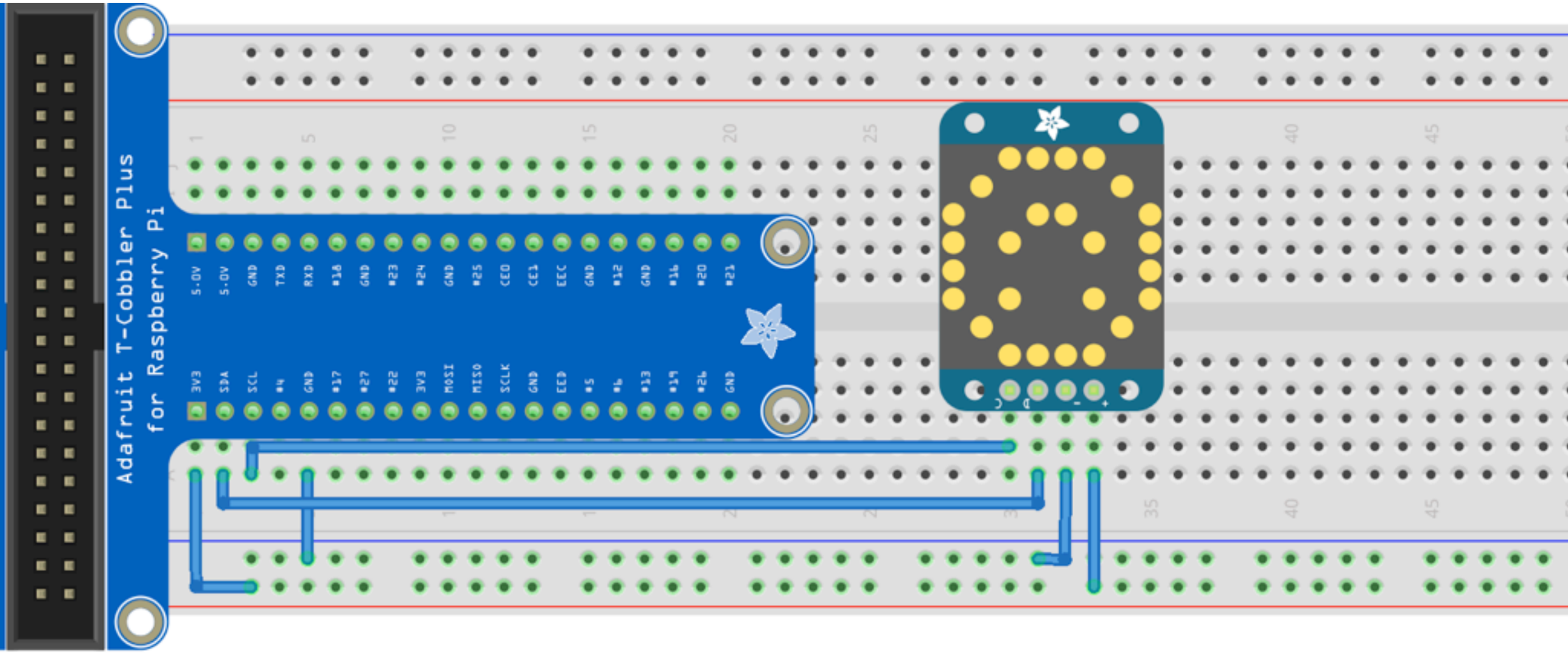
Photoresistor, ADC, & SPI

```
adc = spidev.SpiDev()
adc.open(0,0)

def readadc(channel):
    if ((channel > 7) or (channel < 0)):
        return -1
    r = adc.xfer2([1,(8+channel)<<4,0])
    adcout = ((r[1]&3) << 8) + r[2]
    return adcout

while true:
    print(readadc(0))
    time.sleep(1)
```

8x8 LED Matrix & I²C



- LED + to 3.3V
- LED - to Ground
- LED C to RasPi SCL
- LED D to RasPi SDA