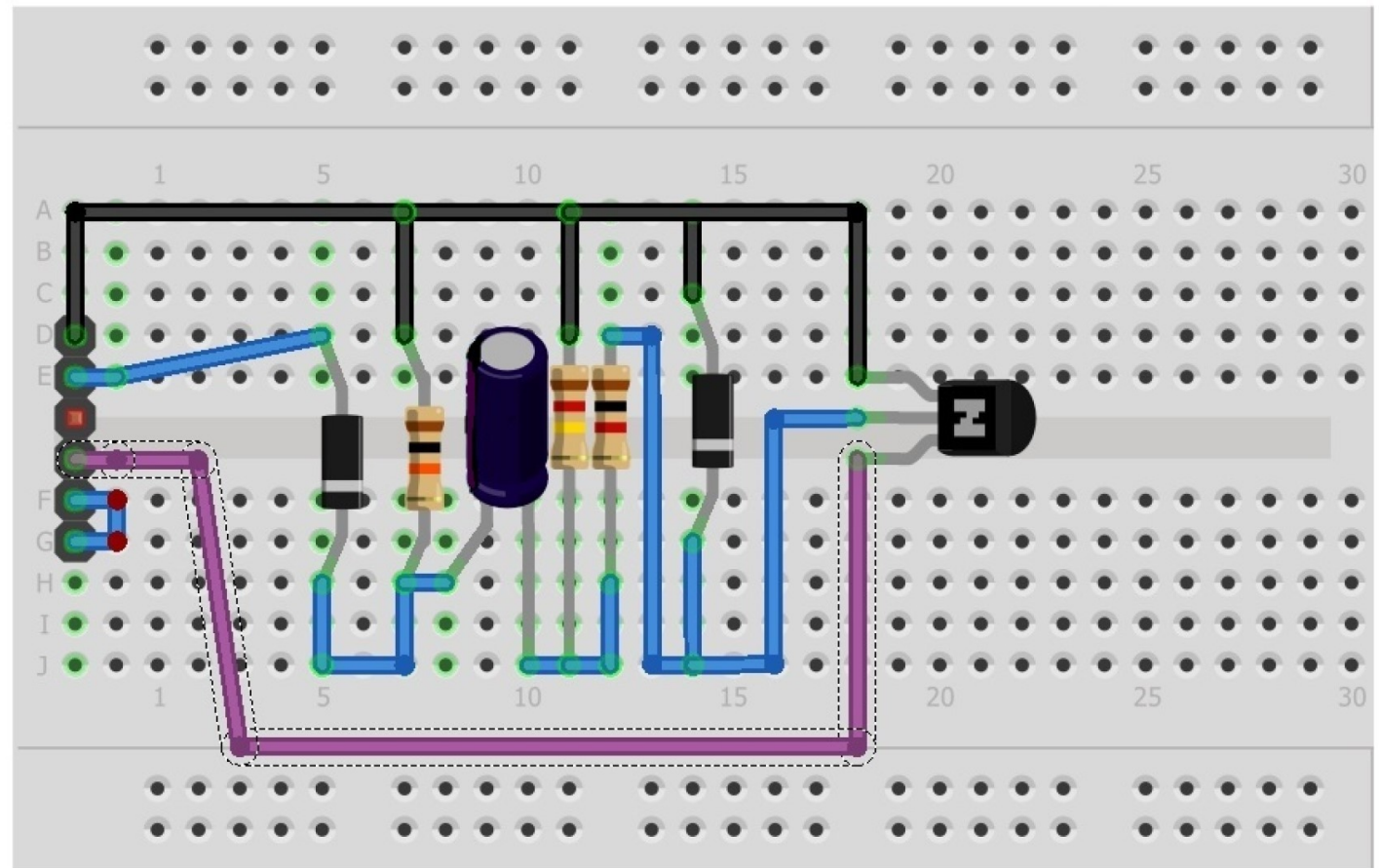
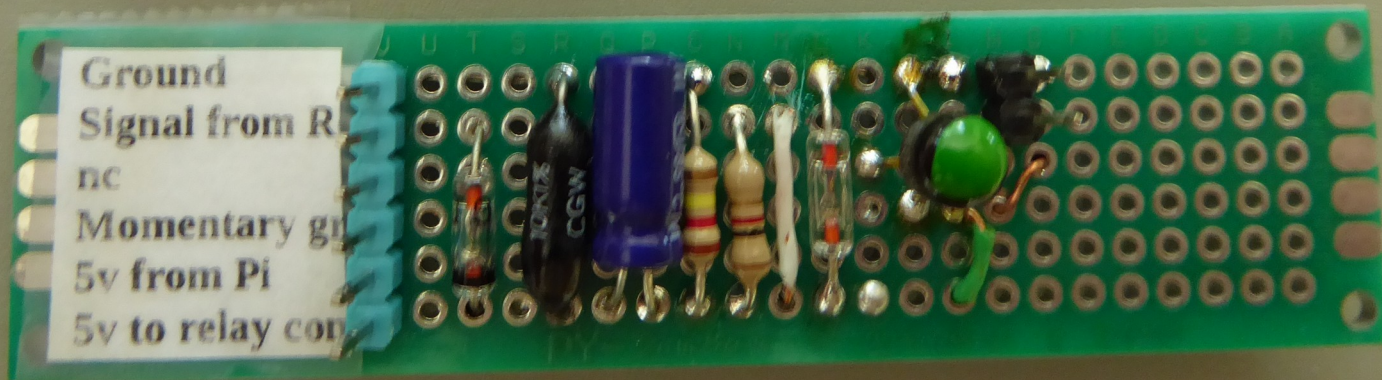
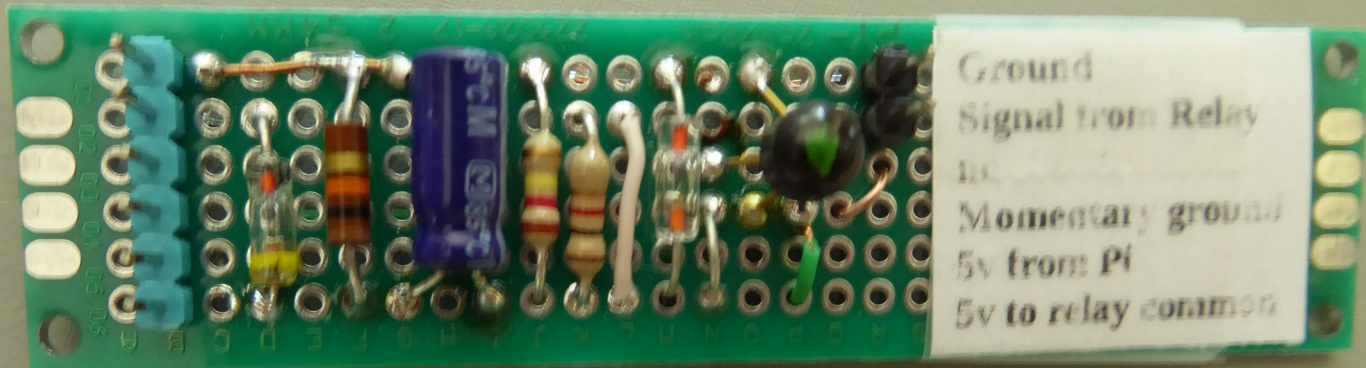
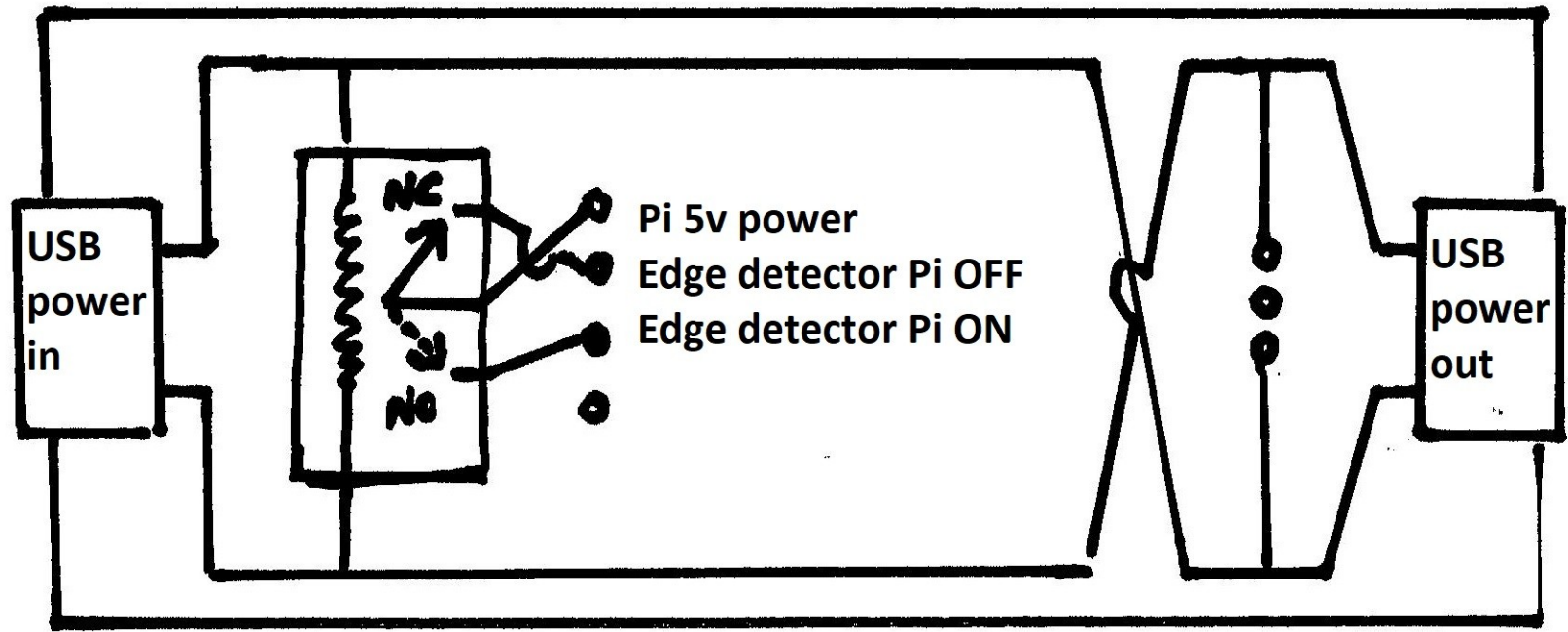


Ground  
Signal from relay  
nc  
Momentary ground  
5v from Pi  
5v to relay common





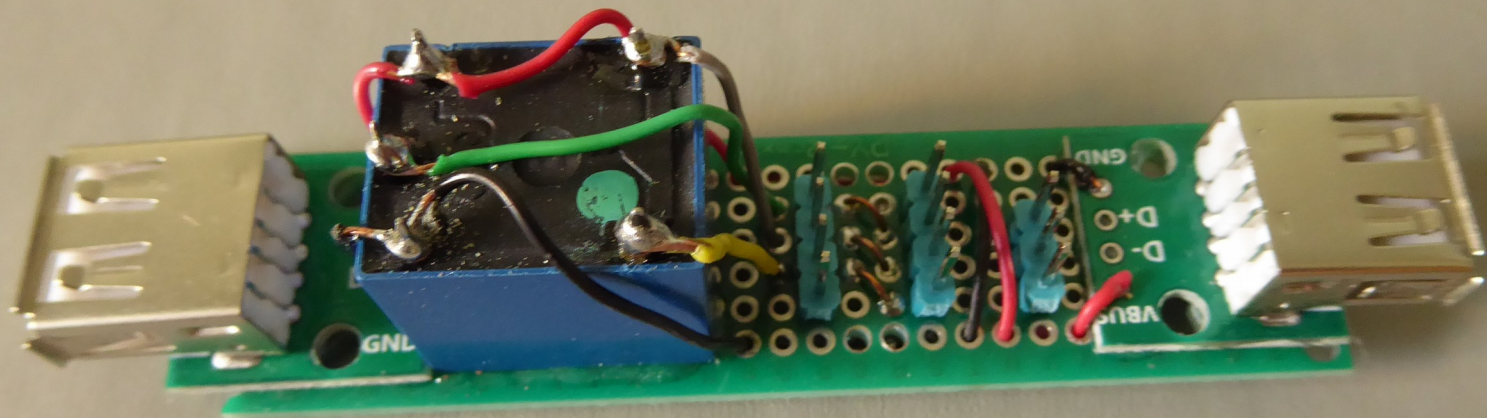




Note, no Pi ground

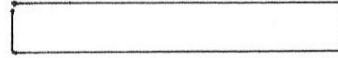
LED power light connection





## Power Supply Board

Vout 3.3v



Yellow

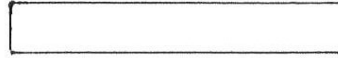
Cap 10uF  
Vout to G

LD117V33C  
3.3v Regulator

Cap 0.1uF  
Vin to G

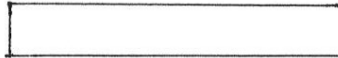
G Vout Vin

Ground

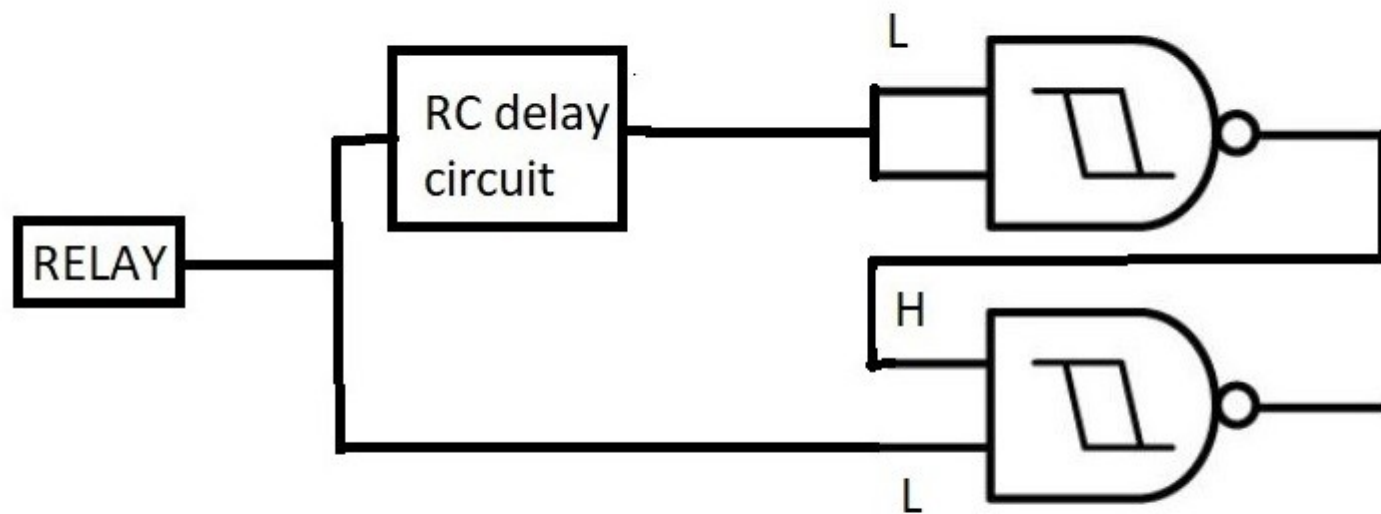


Black

Vin 5v



Red



**H = 100k to VCC**

**L = 100k to ground**

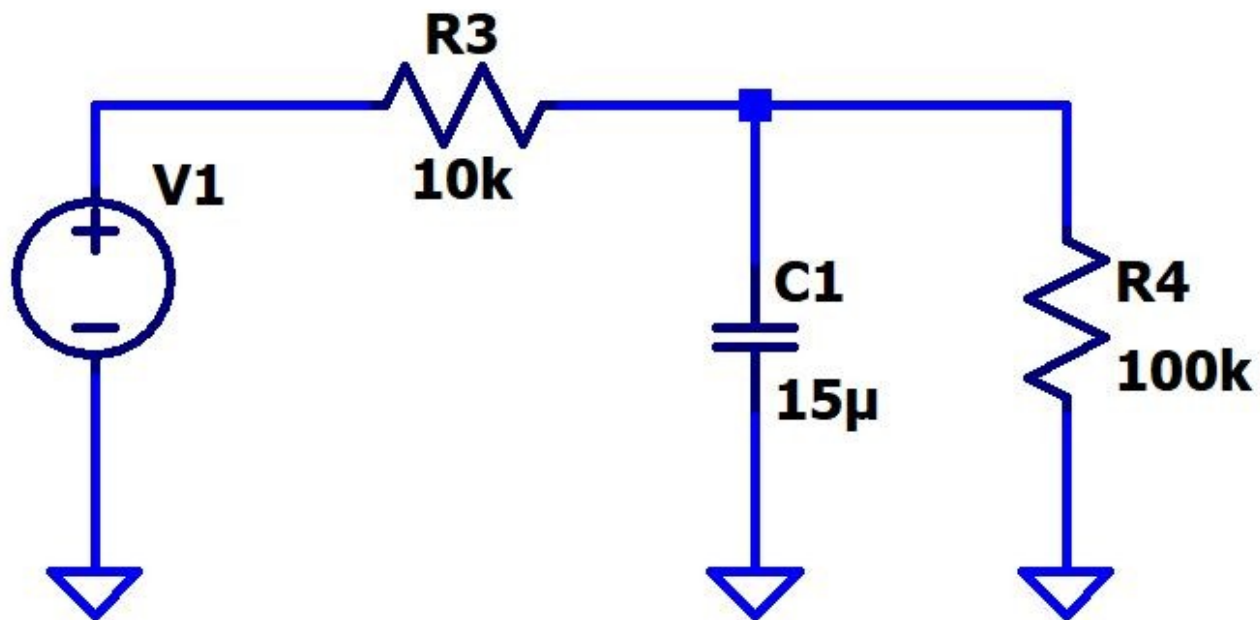
Event/Time	Relay Status NO side	Gate 1		Gate 2		
		In	Out	A In	B In	Out
Standby	Open	L	H	H	L	H
<-----Pi in standby mode----->						
Amp on	Closed		L		H	H
Amp on +RC Delay time	Closed		H		L	H
<-----Pi running----->						
Amp off	Open	L	H	H	L	H

Event/Time	Relay Status NC side	Gate 3		Gate 4		
		In	Out	A In	B In	Out
Standby	Closed		H		L	H
<-----Pi in standby mode----->						
Amp on	Open	L	H	H	L	H
<-----Pi running----->						
Amp off	Closed		L		H	H
Amp off +RC Delay time	Closed		H		L	H
<-----Pi in standby mode----->						
Amp off	Closed		H		L	H

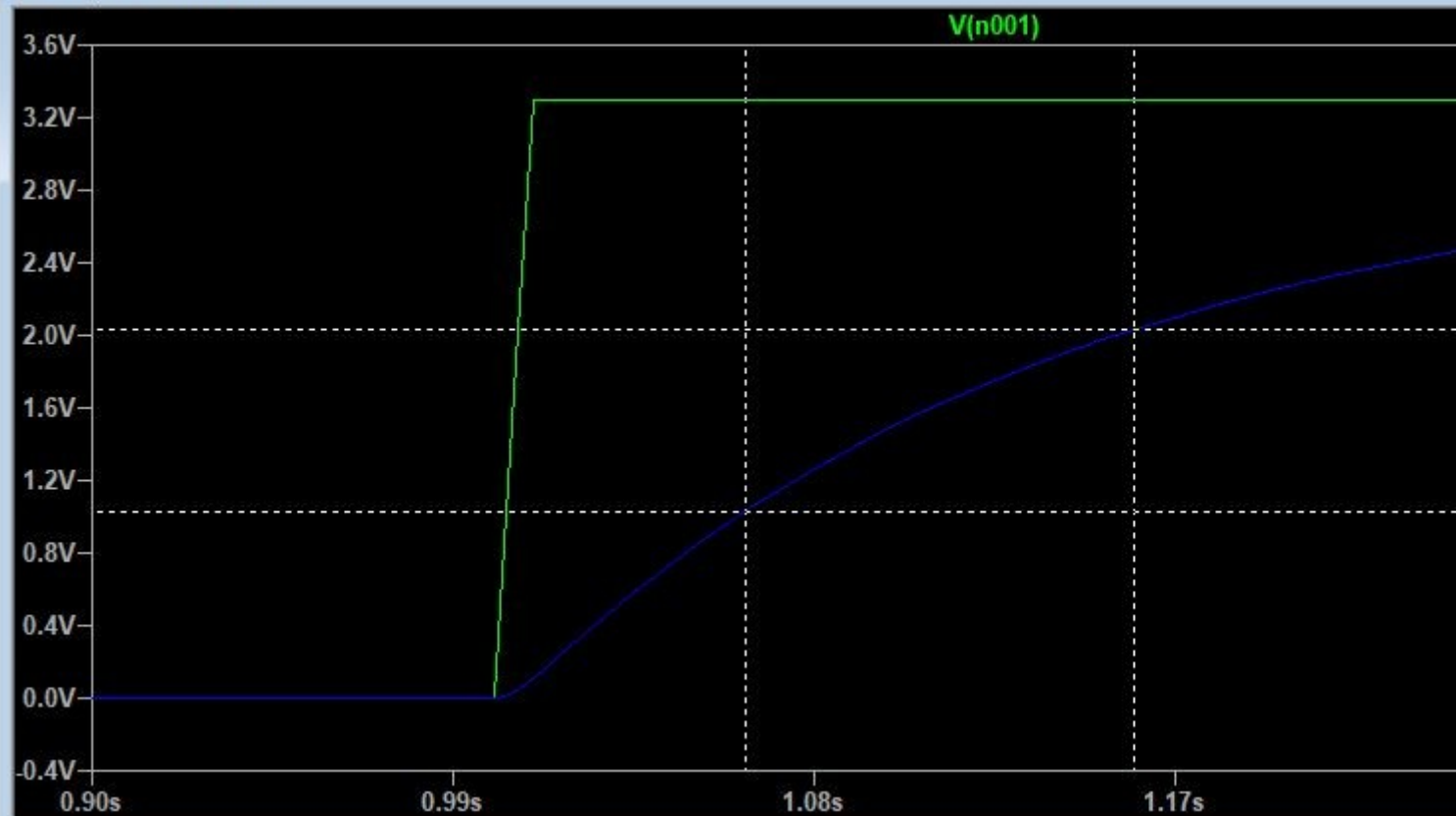
**PULSE(0 3.3 1s 0.01 0.01 5s 10s 2)**

Rectangular Snip

**.tran 15s**

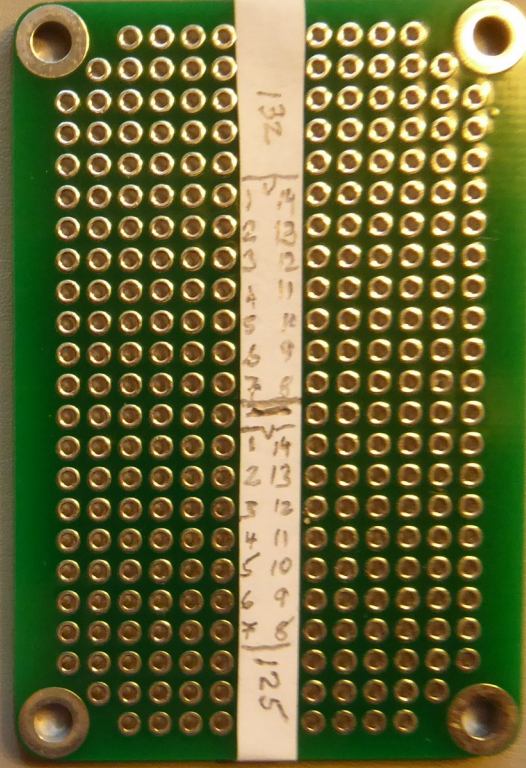
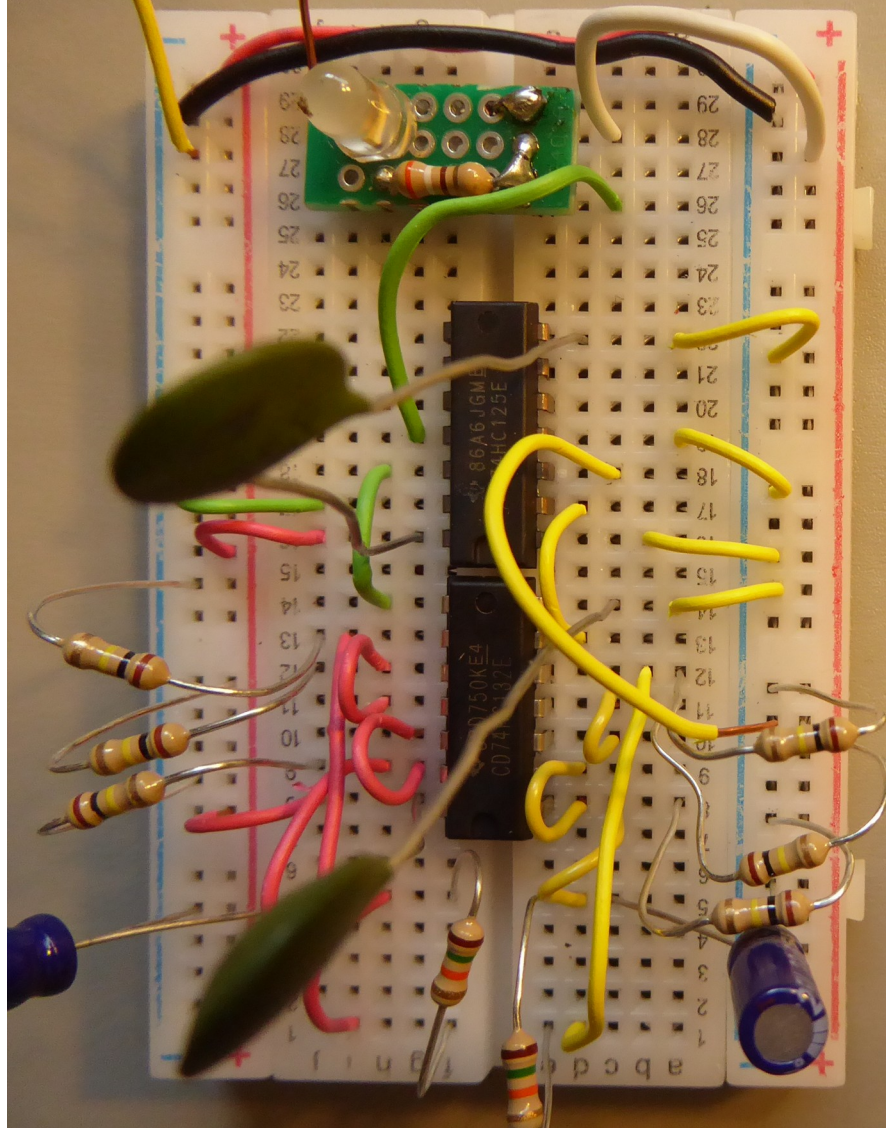


SimpleRCcircuit.raw

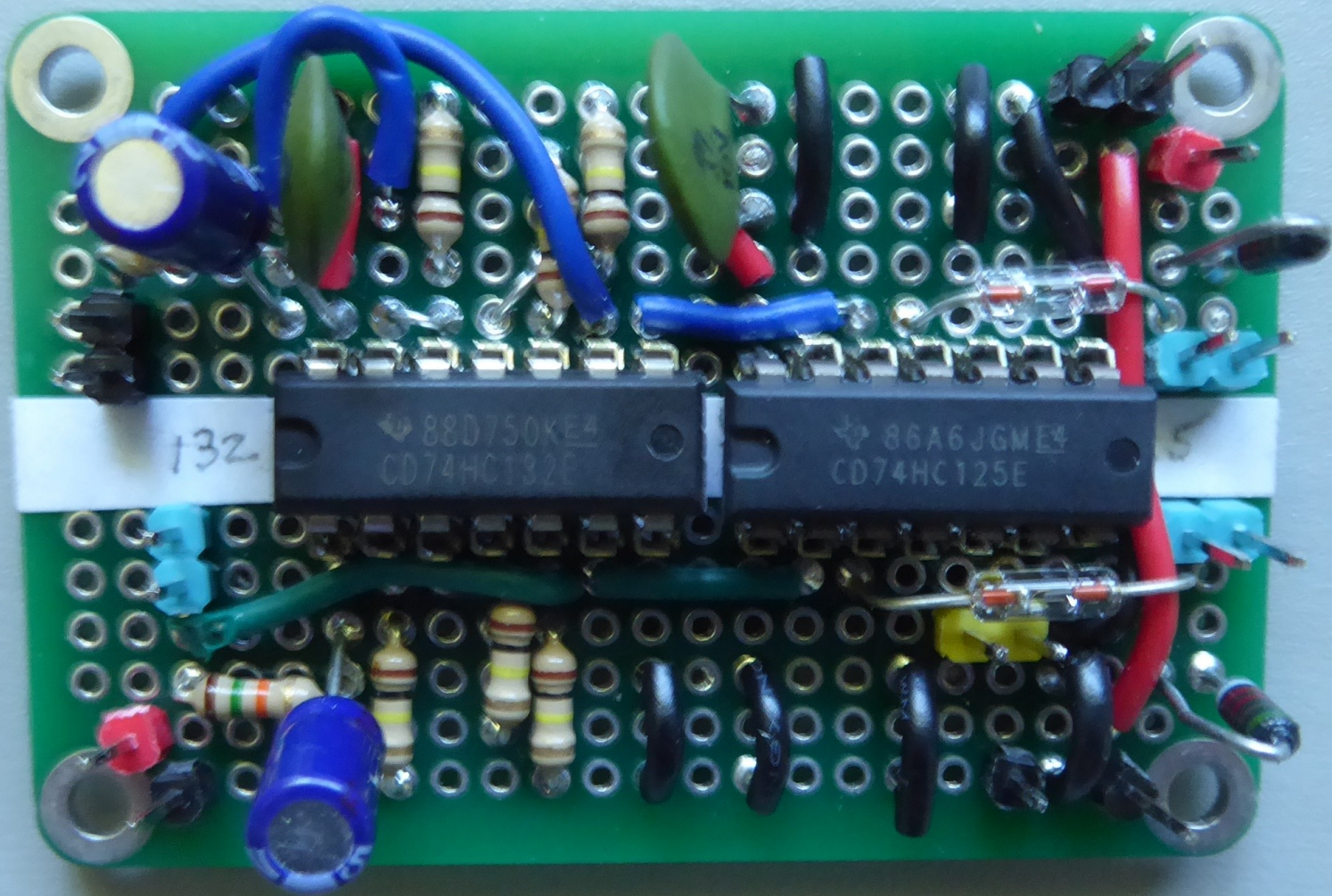




74HC132 Quad 2IN NAND Connections					74HC132 Quad 2IN NAND Connections						
	C3	C2	C1	Pin	Name	Name	Pin	C1	C2	C3	
	RC Delay (NO)		Pin 2	1	1A	VCC	14	VCC	0.1uF to ground		
		100k to ground	Pin 1	2	1B	4B	13	Pin 12	100k to ground	RC Delay (NC)	
			Pin 4	3	1Y	4A	12	Pin 13			
		100k to VCC	Pin 3	4	2A	4Y	11	Pin 10			
	Relay direct (NO)	100k to ground		5	2B	3B	10	Pin 11	100k to VCC		
Turn on signal			Buffer 1A (pin2)	6	2Y	3A	9		100k to ground	Relay direct (NC)	
			Ground	7	GND	3Y	8	Buffer 4A (pin 12)			Turn off signal
74HC125 Hex tri-state buffer Connections					74HC125 Hex tri-state buffer Connections						
			Ground	1	1e	VCC	14	VCC	0.1uF to ground		
			NAND 2Y(pin6)	2	1A	4e	13	Ground			
Turn on signal			Pi via diode (pin3)	3	1Y	4A	12	NAND 3Y (pin 8)			
			Ground	4	2e	4Y	11	Pi via diode (pin 40)			Turn off signal
			Pi ON (pin 8)	5	2A	3e	10	Ground			
			ON diode +ve	6	2Y	3A	9	Not used			
			Ground	7	GND	3Y	8	Not used			





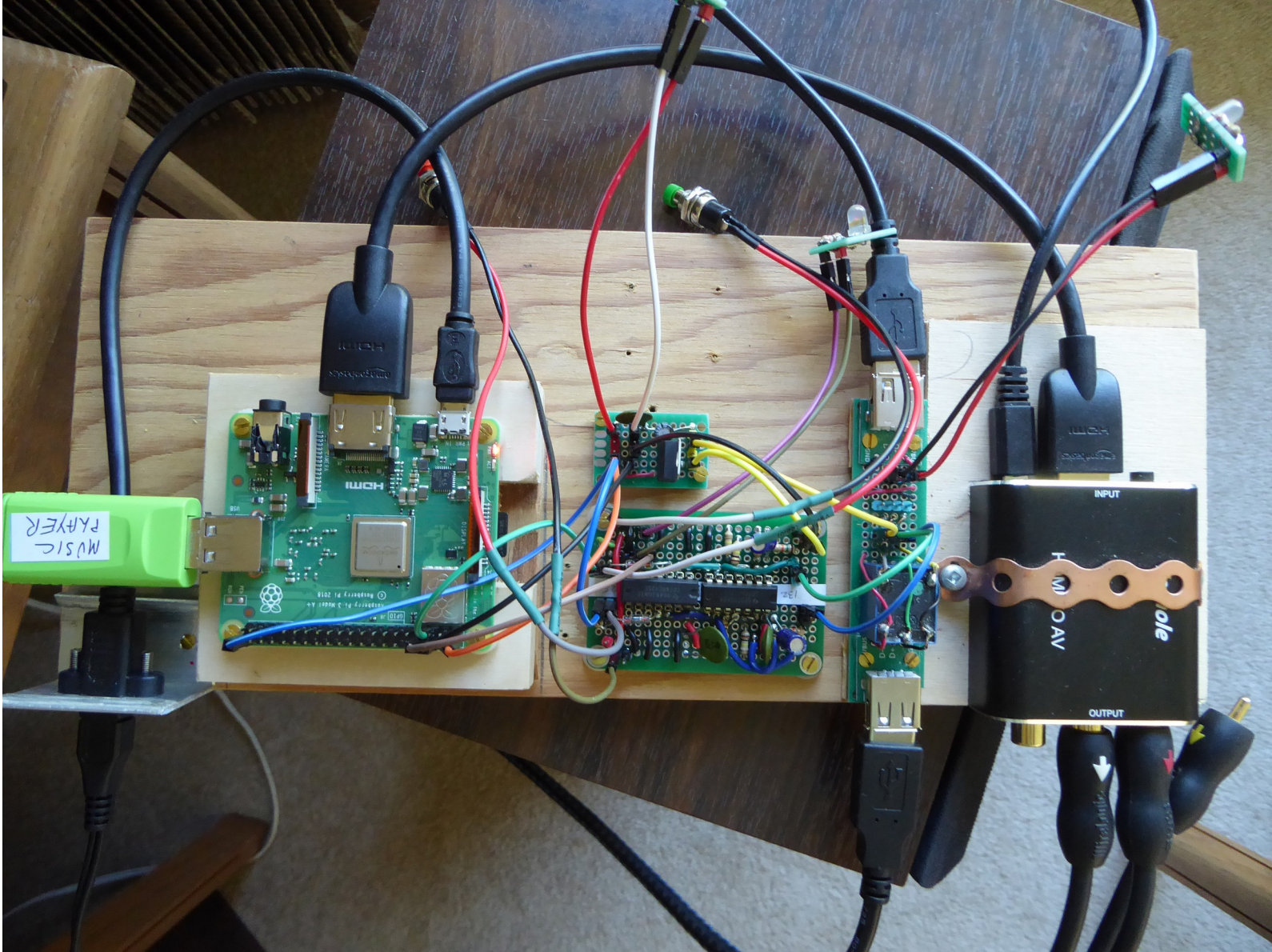


	Terminal	Connect to	Cable colour
<b>Power Board</b>			
	Red	5V on Pi (pin2)	Orange
	Black	Pi ground (pin6)	Black
	Black	Trigger board ground	Black
	Yellow	Trigger board +ve	Yellow
	Yellow	Relay common	Yellow
	Red	Pi powered LED	Red
	Black	Pi powered LED	Black
<b>Pi 3A</b>			
	Pin 8	Trigger board ON light (Yellow pin, 74125 pin5)	Brown
	Pin 2	Power board 5V input	Orange
	Pin 6	Power Board ground	Black
	Pin 5	Trigger Board ON pin	Green
	Pin 40	Trigger board OFF pin	Blue
<b>Relay Board</b>			
	Relay Common	Power Board 3.3v	Yellow
	Relay NO	Trigger Board ON side	Green
	Relay NC	Trigger board OFF side	Blue
	Indicator out	LED unit	Black
	Indicator out	LED unit	Red
<b>Trigger Board</b>			
	Power +ve	Power Board 3.3v	Yellow
	Power ground	Power Board ground	Black
	Relay NC	Relay Board	Blue
	Relay NO	Relay Board	Green
	Momentary ON	Pi pin 5	Green
	Momentary ON	Green PB	White
	Ground	Green PB	White
	Momentary OFF	Pi pin 40	Blue
	Momentary OFF	Red PB	Grey
	Ground	Red PB	Brown
	ON light signal	Pi pin 8	Brown
	Pi Running LED	LED Unit	Brown
	Ground	LED Unit	Purple
<b>* Blue cable Momentary OFF to Pi is outermost pin</b>			













## Putting pi in "Halt" state

Momentary ground GPIO 21 (pin 40)

crontab contains a command to run a shell script, which in turn runs a python script.

1/ Add to crontab "@reboot sh  
/home/pi/Startscripts/launcher.sh >/home/pi/logs/cronlog 2>&1

2/ Create launcher.sh in Startscripts, and make it executable.

```
#!/bin/sh  
cd /  
cd/home/pi/Startscripts  
sudo python3 01offbutton.py  
cd/
```

3/ Create 01offbutton.py in Startscripts - no need to be executable.

```
#!/usr/bin/env python3  
from gpiozero import Button  
import os  
Button(21).wait_for_press() #(pin40)  
os.system ("sudo poweroff")
```

## **Pi "Wake-up" and "Pi-on" indicator**

**Momentarily grounding pin 5 is a built-in default to wake the Pi from the "Halt" state**

**I used pin 8 (UART TX), this needs to be enabled:  
In /boot/config.txt add "enable\_uart=1"**

