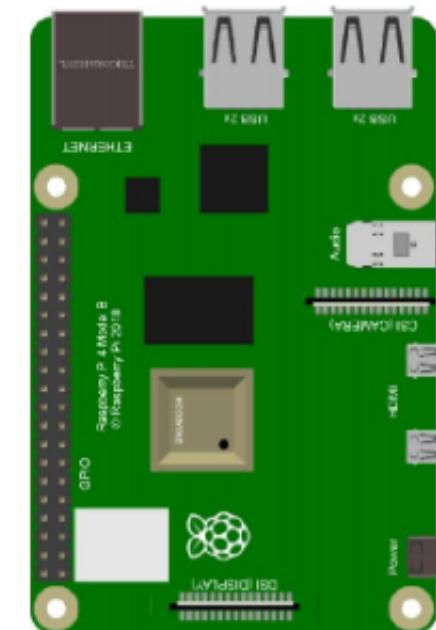
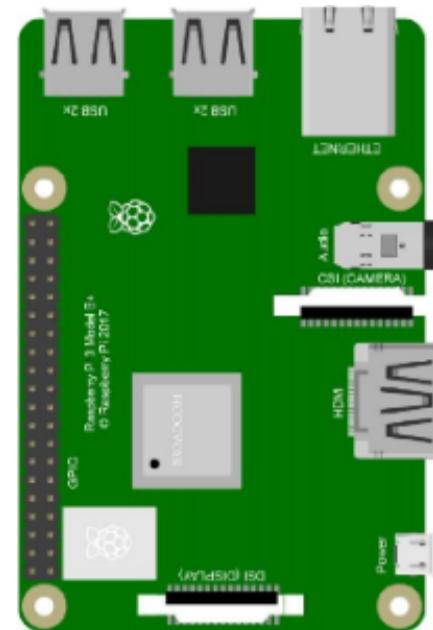


# Raspberry Pi Dersleri

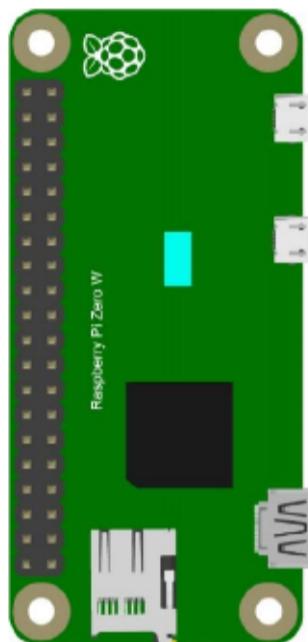
Raspberry Pi 4 Model B / CAD Cizimi:



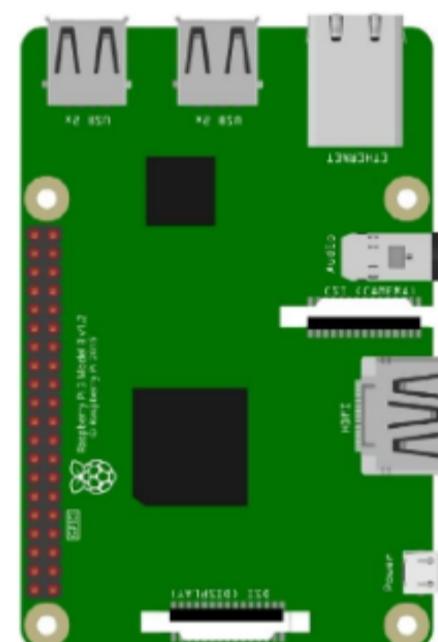
Raspberry Pi 3 Model B+ / CAD Cizimi:



Raspberry Pi Zero W / CAD Cizimi:



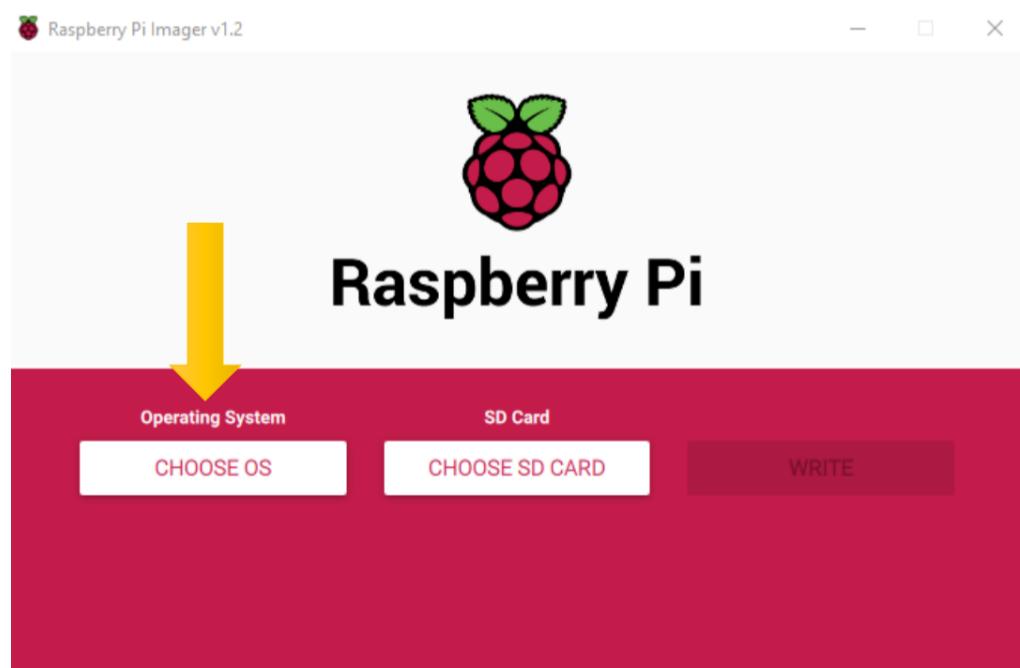
Raspberry Pi 3 Model B / CAD Cizimi:



## Raspberry Pi Gereken Sarj Adaptörleri

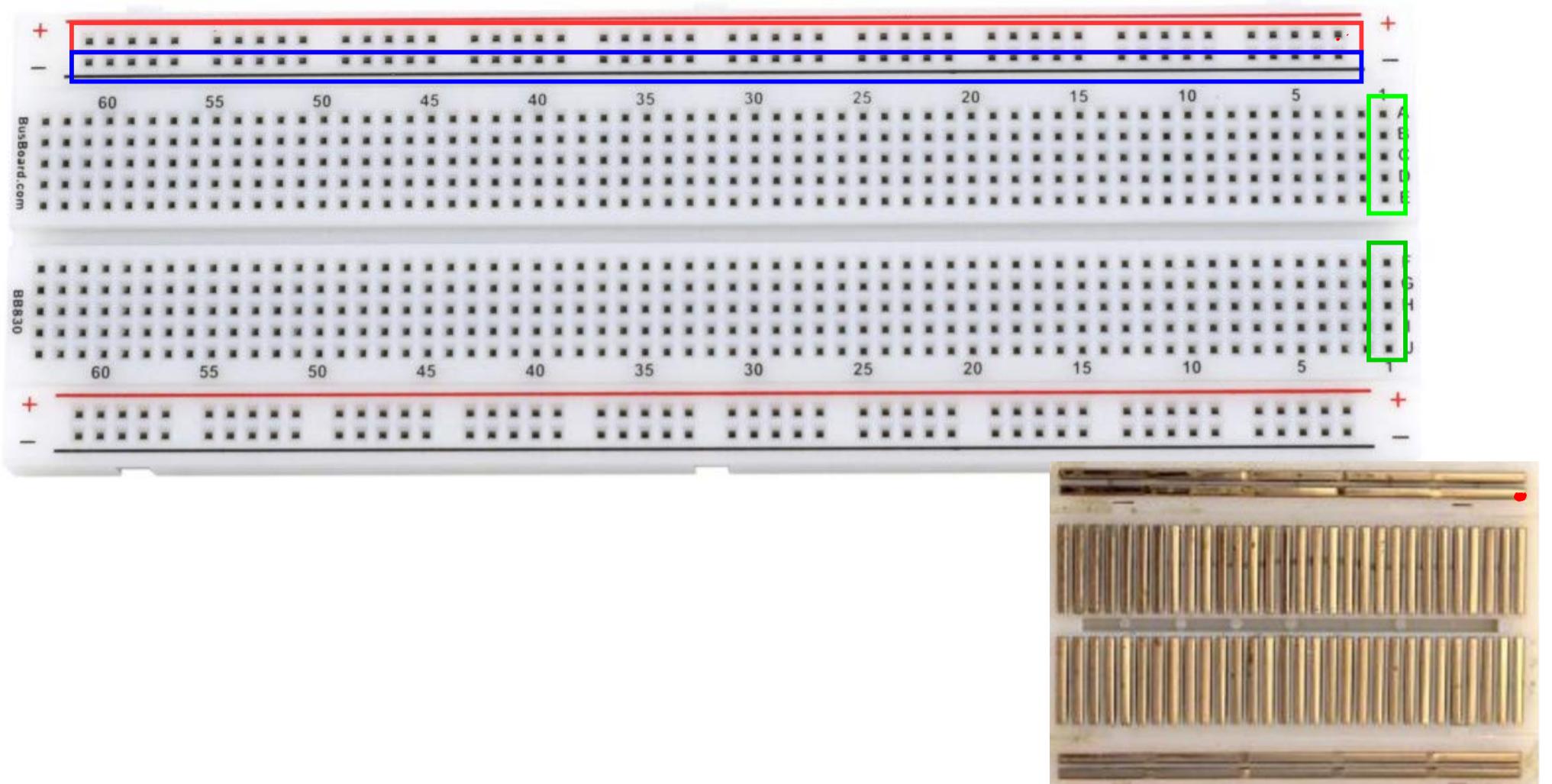
Raspberry Pi Model A	700mA
Raspberry Pi Model B	1.2A
Raspberry Pi Model A+	700mA
Raspberry Pi Model B+	1.8A
Raspberry Pi 2 Model B	1.8A
Raspberry Pi 3 Model B	2.5A
Raspberry Pi 3 Model A+	2.5A
Raspberry Pi 3 Model B+	2.5A
Raspberry Pi 4 Model B	3.0A
Raspberry Pi Zero W	1.2A
Raspberry Pi Zero	1.2A

## Raspberry Pi Imager



Bilgi: Ctrl+Shift+X ek Özellikler Acar!

# Devre Tahtası (Breadboard)



## GPIO Haritası

i<sup>2</sup>C: Inter-Integrated Circuit

SCL: Serial Clock

SDA: Serial Data

GPCLK: General Purpose  
Clock (Fix Frekans  
Ayarlanabilir Pin)

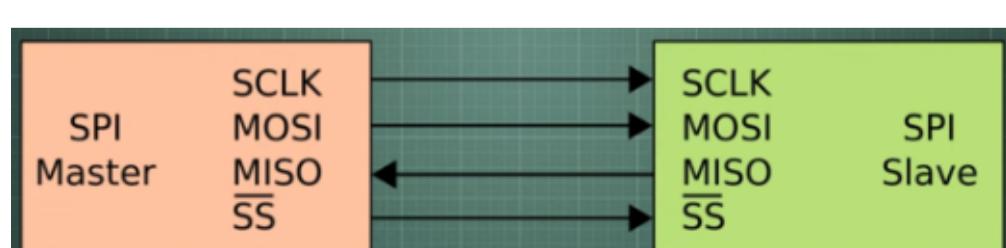
		J8		
Power	+3,3V	(1)	(2)	+5V Power
{I2C}	SDA1	(3)	(4)	+5V Power
{I2C}	SCL1	(5)	(6)	GND
GPCLK0	GPIO4	(7)	(8)	GPIO14 {UART} TXD0
	GND	(9)	(10)	GPIO15 {UART} RXD0
	GPIO17	(11)	(12)	GPIO18 PCM_CLK
	GPIO27	(13)	(14)	GND
	GPIO22	(15)	(16)	GPIO23
Power	+3,3V	(17)	(18)	GPIO24
SPI0_MOSI	GPIO10	(19)	(20)	GND
SPI0_MISO	GPIO9	(21)	(22)	GPIO25
SPI0_SCLK	GPIO11	(23)	(24)	GPIO8 SPI0_CE0_N
	GND	(25)	(26)	GPIO7 SPI0_CE1_N
{ID EEPROM}	ID_SD	(27)	(28)	ID_SC {ID EEPROM}
GPCLK1	GPIO5	(29)	(30)	GND
GPCLK2	GPIO6	(31)	(32)	GPIO12 PWM0
PWM1	GPIO13	(33)	(34)	GND
PCM_FS	GPIO19	(35)	(36)	GPIO16
	GPIO26	(37)	(38)	GPIO20 PCM_DIN
	GND	(39)	(40)	GPIO21 PCM_DOUT

UART: Universal Asynchronous  
Receiver Transmitter



EEPROM: Electrically  
Erasable Programmable  
Read-Only Memory

PCM: Pulse-Code Modulation

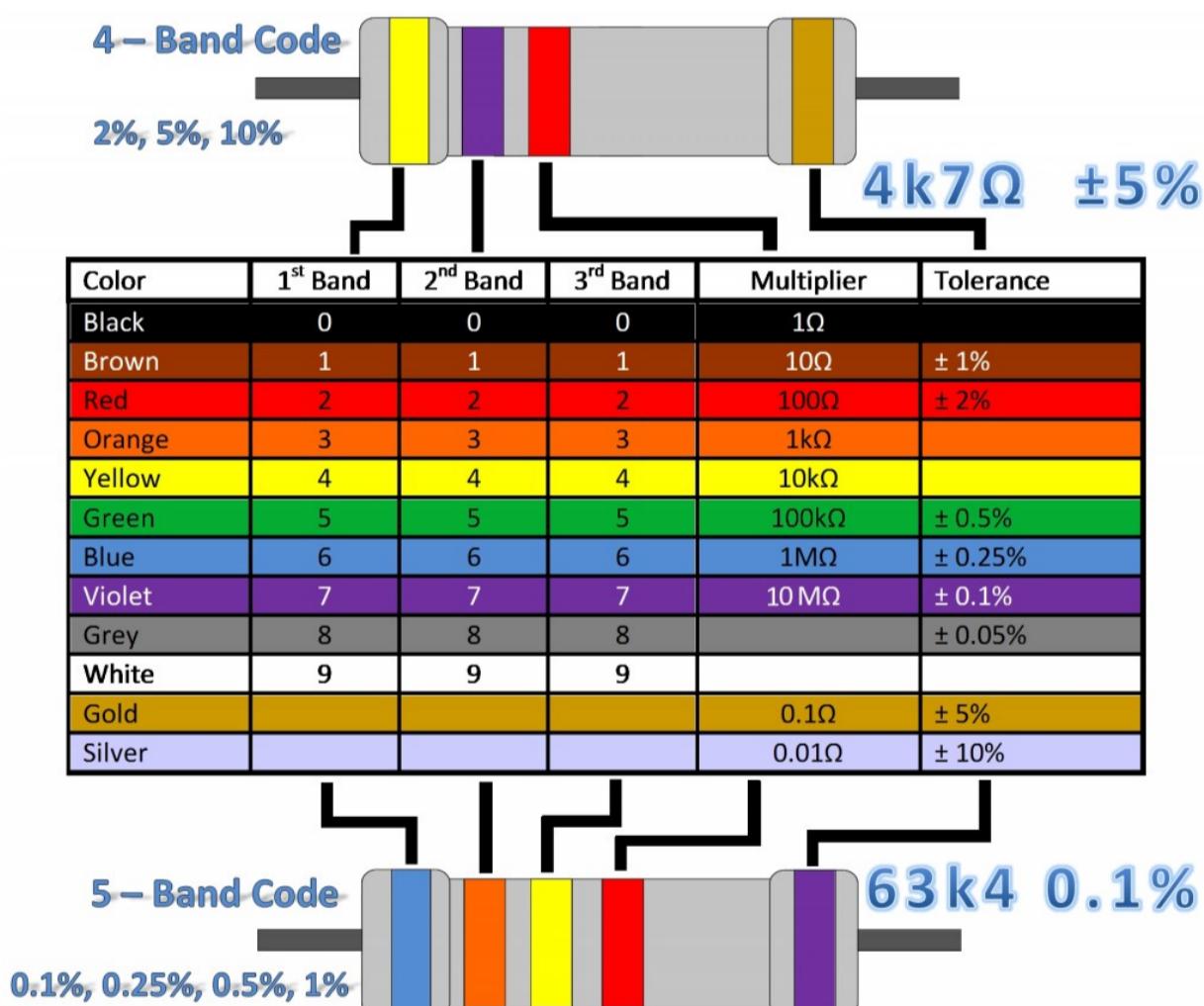


wiringPi Pin	BCM GPIO	Name	Header	Name	BCM GPIO	wiringPi Pin
—	—	3.3v	1   2	5v	—	—
8	R1:0/R2:2	SDA	3   4	5v	—	—
9	R1:1/R2:3	SCL	5   6	0v	—	—
7	4	GPIO7	7   8	TxD	14	15
—	—	0v	9   10	RxD	15	16
0	17	GPIO0	11   12	GPIO1	18	1
2	R1:21/R2:27	GPIO2	13   14	0v	—	—
3	22	GPIO3	15   16	GPIO4	23	4
—	—	3.3v	17   18	GPIO5	24	5
12	10	MOSI	19   20	0v	—	—
13	9	MISO	21   22	GPIO6	25	6
14	11	SCLK	23   24	CE0	8	10
—	—	0v	25   26	CE1	7	11
30	0	SDA.0	27   28	SCL.0	1	31
21	5	GPIO.21	29   30	0V	—	—
22	6	GPIO.22	31   32	GPIO.26	12	26
23	13	GPIO.23	33   34	0V	—	—
24	19	GPIO.24	35   36	GPIO.27	16	27
25	26	GPIO.25	37   38	GPIO.28	20	28
		0V	39   40	GPIO.29	21	29

wiringPi Pin	BCM GPIO	Name	Header	Name	BCM GPIO	wiringPi Pin
-----------------	-------------	------	--------	------	-------------	-----------------

Direnc Kodu  
Sokak Ta Sayıları Gibi



# Wiring Pi Yüklemesi:

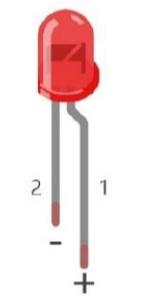
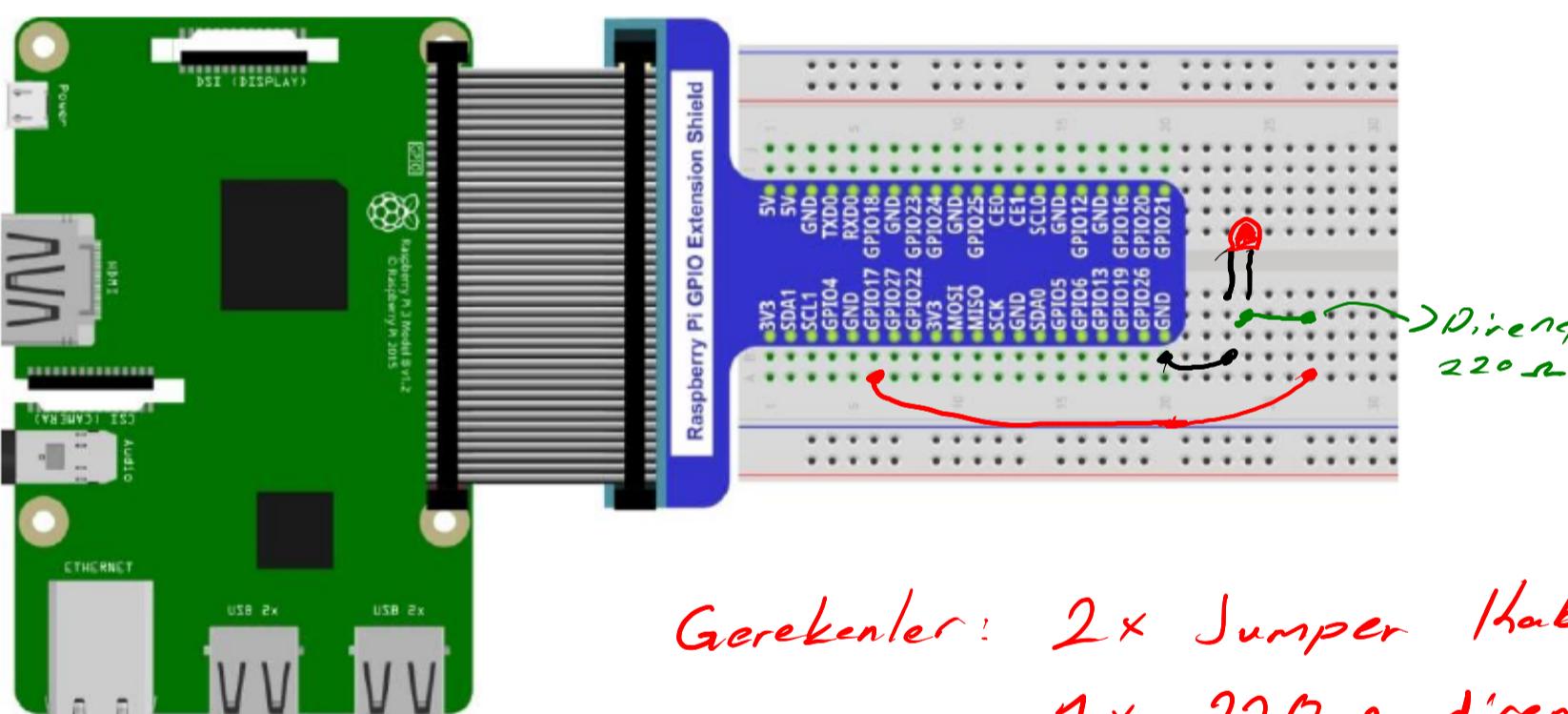
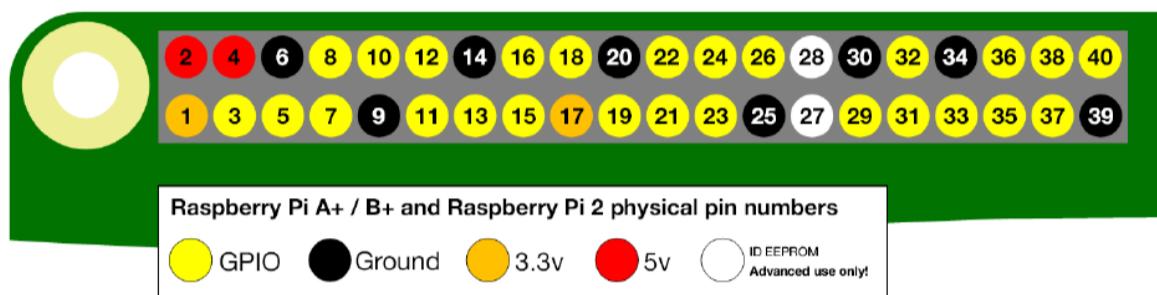
```
sudo apt-get update
git clone https://github.com/WiringPi/WiringPi
cd WiringPi
./build
```

Test et.

gpio readall										Pi 4B
BCM	wPi	Name	Mode	V	Physical	V	Mode	Name	wPi	BCM
		3.3v			1    2			5v		
2	8	SDA.1	ALTO	1	3    4			5v		
3	9	SCL.1	ALTO	1	5    6			0v		
4	7	GPIO. 7	IN	1	7    8	0	IN	TxD	15	14
		0v			9    10	1	IN	RxD	16	15
17	0	GPIO. 0	IN	0	11    12	0	IN	GPIO. 1	1	18
27	2	GPIO. 2	IN	0	13    14			0v		
22	3	GPIO. 3	IN	0	15    16	0	IN	GPIO. 4	4	23
		3.3v			17    18	0	IN	GPIO. 5	5	24
10	12	MOSI	IN	0	19    20			0v		
9	13	MISO	IN	0	21    22	0	IN	GPIO. 6	6	25
11	14	SCLK	IN	0	23    24	1	IN	CE0	10	8
		0v			25    26	1	IN	CE1	11	7
0	30	SDA.0	IN	1	27    28	1	IN	SCL.0	31	1
5	21	GPIO.21	IN	1	29    30			0v		
6	22	GPIO.22	IN	1	31    32	0	IN	GPIO.26	26	12
13	23	GPIO.23	IN	0	33    34			0v		
19	24	GPIO.24	IN	0	35    36	0	IN	GPIO.27	27	16
26	25	GPIO.25	IN	0	37    38	0	IN	GPIO.28	28	20
		0v			39    40	0	IN	GPIO.29	29	21

## 1. Led Yakma

Hatırlatma!!!



Gerekeler: 2x Jumper Kablosu  
1x 220 Ω direnç  
1x Led

```
import RPi.GPIO as GPIO
import time

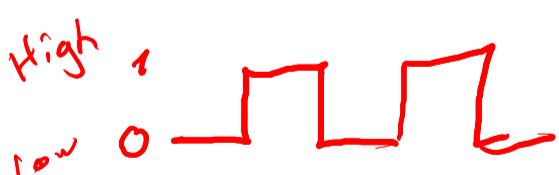
ledPin = 11 # Fiziksel GPIO Pin tanimi

def kurulum():
    GPIO.setmode(GPIO.BOARD)          # GPIO Fiziksel modunda calistirir
    GPIO.setup(ledPin, GPIO.OUT)        # GPIO Pin kurulumu
    GPIO.output(ledPin, GPIO.LOW)       # Pin Dusuk Frekansta baslasin
    print('Kurulum basladi. Kurulan pin: %d' %ledPin)

def dongu():
    while True:
        GPIO.output(ledPin, GPIO.HIGH)   # Led Pin Frekansini yukselt
        print(">>> LED Acilmistir!")
        time.sleep(1)
        GPIO.output(ledPin, GPIO.LOW)    # Led Pin Frekansini dusur
        print(">>> LED Kapanmistir!")
        time.sleep(1)

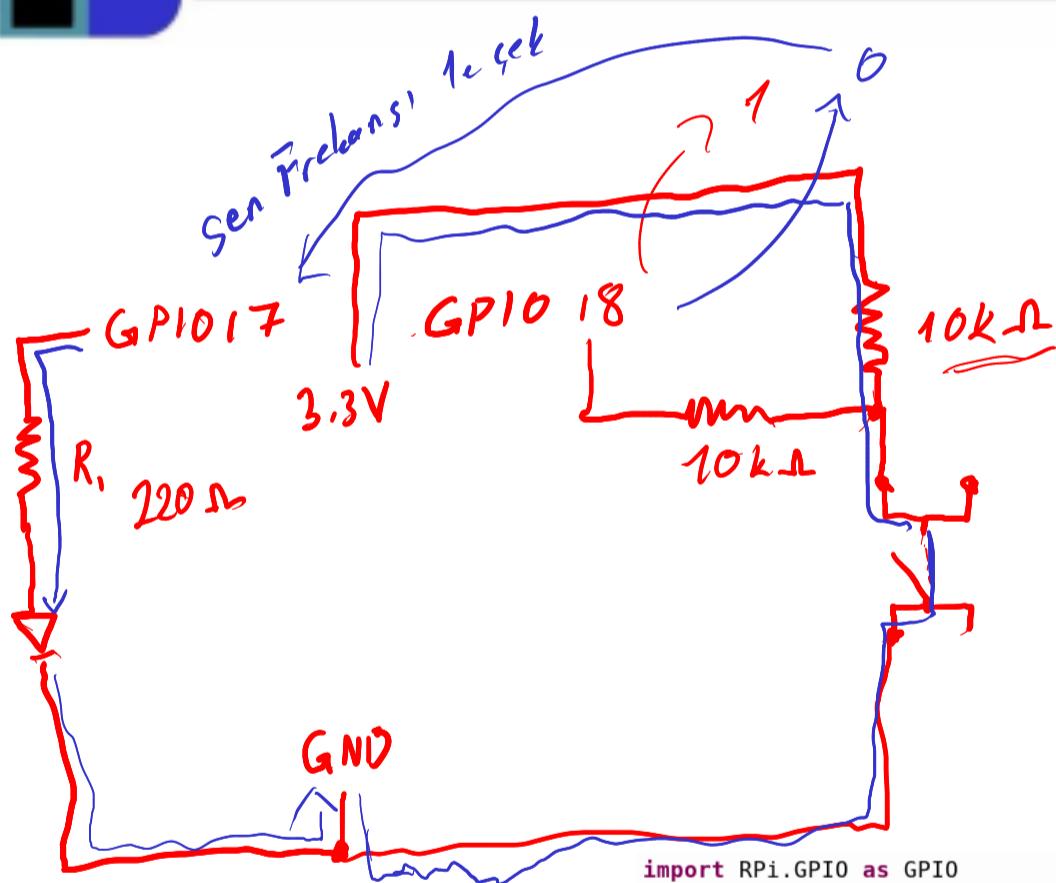
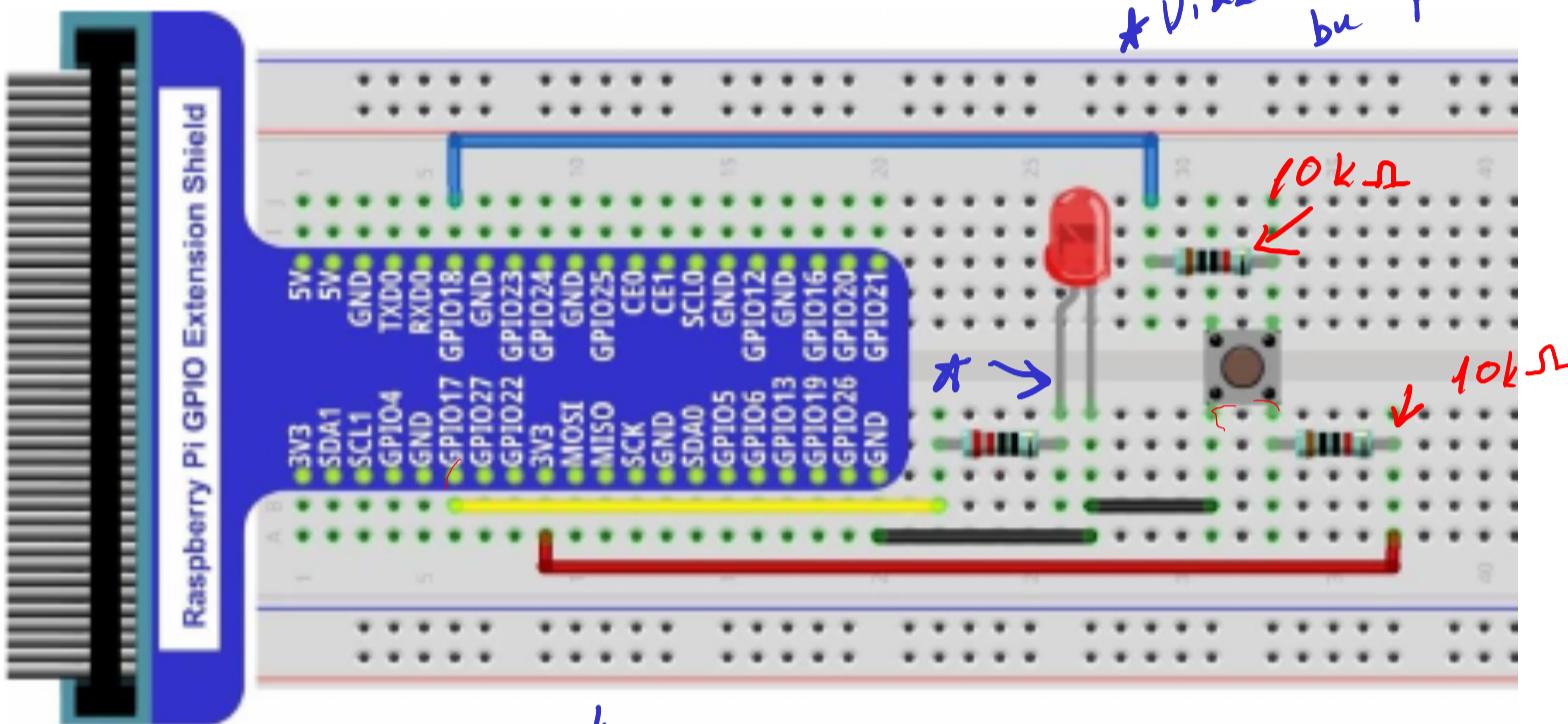
def bitis():
    GPIO.cleanup()                   # GPIO Cikislarini sifirlar

if __name__ == '__main__':
    print("PROGRAM BASLIYOR!")
    kurulum()
    try:
        dongu()
    except KeyboardInterrupt:
        bitis()                         # Klavye ile CTRL+C mudehalesi sirasinda
```



## 2. Button ile Led Yakma

Gereklilikler: 5x Jumper Kablo  
1x Button  
1x Led  
1x 220  $\Omega$  Direnç  
2x 10k  $\Omega$  Direnç



```
import RPi.GPIO as GPIO
import time

ledPin = 11 # Fiziksel GPIO Pin tanimi
btnPin = 12 # Fiziksel GPIO Button Pin Tanimi

def kurulum():
    GPIO.setmode(GPIO.BOARD)          # GPIO Fiziksel modunda calistirir
    GPIO.setup(ledPin, GPIO.OUT)        # GPIO Pin kurulumu
    GPIO.setup(btnPin, GPIO.IN, pull_up_down=GPIO.PUD_UP) # GPIO Pinini bas kaldir moduna sokacak
    print('Kurulum basladi.')

def dongu():
    while True:
        if GPIO.input(btnPin) == GPIO.LOW:
            GPIO.output(ledPin, GPIO.HIGH) # Ledi yak
            print(">>> LED yanıyor")
        else: # Diger button durumu
            GPIO.output(ledPin, GPIO.LOW)
            print(">>> LED Yanmiyor")

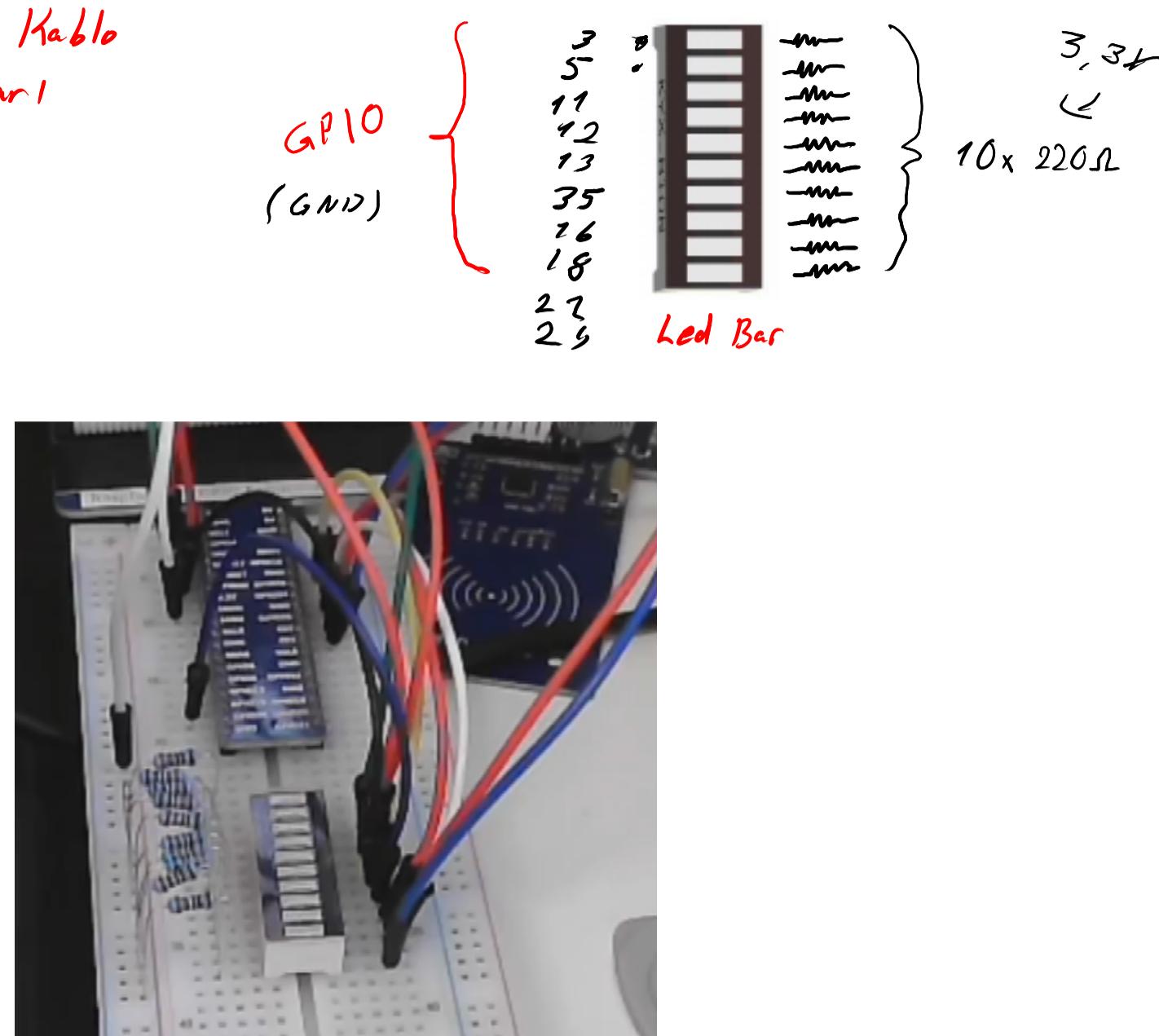
def bitis():
    GPIO.cleanup()                    # GPIO Cikislarini sifirlar

if __name__ == '__main__':
    print("PROGRAM BASLIYOR!")
    kurulum()
    try:
        dongu()
    except KeyboardInterrupt:
        bitis()                         # Klavye ile CTRL+C müdahalesi sirasinda
```

### 3. Led Barı Yakınla

Gerekliler: 10x 220Ω Direnç  
11x Jumper Kablo  
1x Led Barı

+3,3V	(1)	2	+5V
GPIO2	(3)	4	+5V
GPIO3	(5)	6	GND
GPIO4	(7)	8	GPIO14
GND	(9)	10	GPIO15
GPIO17	(11)	12	GPIO18
GPIO27	(13)	14	GND
GPIO22	(15)	16	GPIO23
+3,3V	(17)	18	GPIO24
GPIO10	(19)	20	GND
GPIO9	(21)	22	GPIO25
GPIO11	(23)	24	GPIO8
GND	(25)	26	GPIO7
ID_SD	(27)	28	ID_SC
GPIO5	(29)	30	GND
GPIO6	(31)	32	GPIO12
GPIO13	(33)	34	GND
GPIO19	(35)	36	GPIO16
GPIO26	(37)	38	GPIO20
GND	(39)	40	GPIO21



```

import RPi.GPIO as GPIO
import time

ledPin = [3, 5, 11, 12, 13, 35, 16, 18, 22, 24] # Fiziksel GPIO Pin tanimi

def kurulum():
    GPIO.setmode(GPIO.BOARD)
    GPIO.setup(ledPin, GPIO.OUT)
    GPIO.output(ledPin, GPIO.HIGH)
    print('Kurulum basladi.')

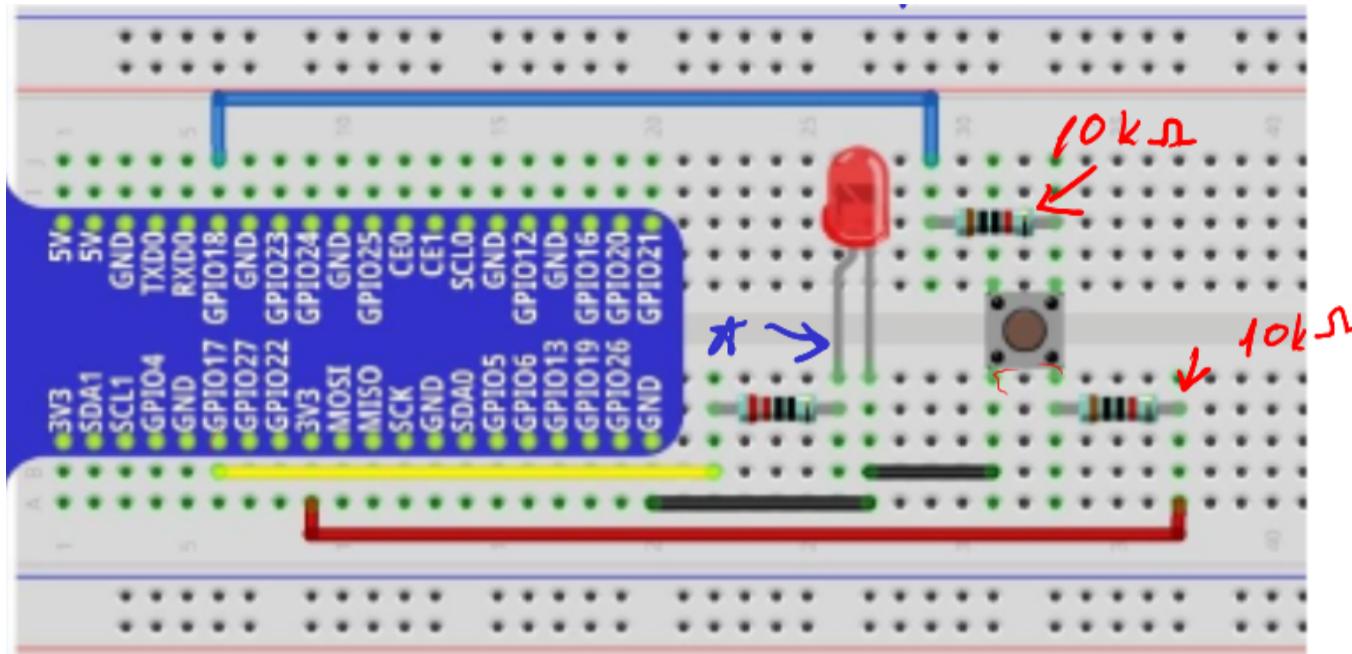
def dongu():
    while True:
        for pin in ledPin:
            GPIO.output(pin, GPIO.LOW)
            time.sleep(0.1)
        for pin in ledPin:
            GPIO.output(pin, GPIO.HIGH)
            time.sleep(0.1)

def bitis():
    GPIO.cleanup() # GPIO Cikislarini sifirlar

if __name__ == '__main__':
    print("PROGRAM BASLIYOR!")
    kurulum()
    try:
        dongu()
    except KeyboardInterrupt: # Klavye ile CTRL+C müdahalesi sirasinda
        bitis()

```

## 4. Led Buton V2



```
import RPi.GPIO as GPIO
import time

ledPin = 11 # Fiziksel GPIO Pin tanimi
btnPin = 12 # Fiziksel GPIO Button Pin Tanimi
ledDurum = False # Led Yanmiyor durumu

def kurulum():
    GPIO.setmode(GPIO.BOARD)          # GPIO Fiziksel modunda calistirir
    GPIO.setup(ledPin, GPIO.OUT)        # GPIO Pin kurulumu
    GPIO.setup(btnPin, GPIO.IN, pull_up_down=GPIO.PUD_UP) # GPIO Pinini bas kaldir moduna sokacak
    print('Kurulum basladi.')

def buttonIslem(channel): # Butona basildiginda bu fonksiyon calisacaktir
    global ledDurum
    print('Buton Islem GPIO Durumu:%d' %channel)
    ledDurum = not ledDurum # Her tıklamada durumu tersine cevirecek (AC/KAPA = TRUE/FALSE)

    if ledDurum:
        print("Led Acik")
    else:
        print("Led Kapali")

    GPIO.output(ledPin, ledDurum)

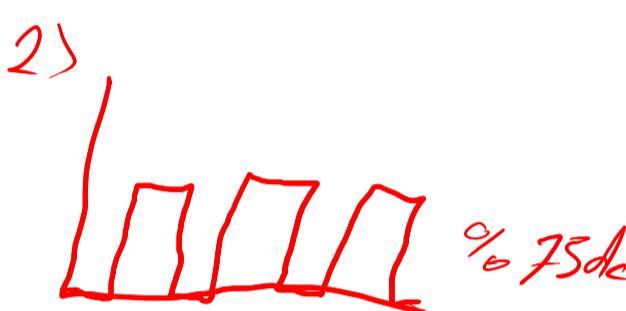
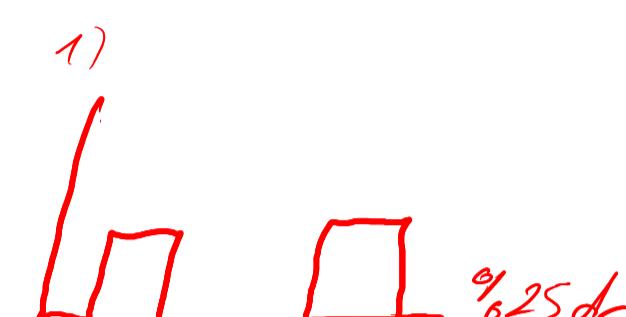
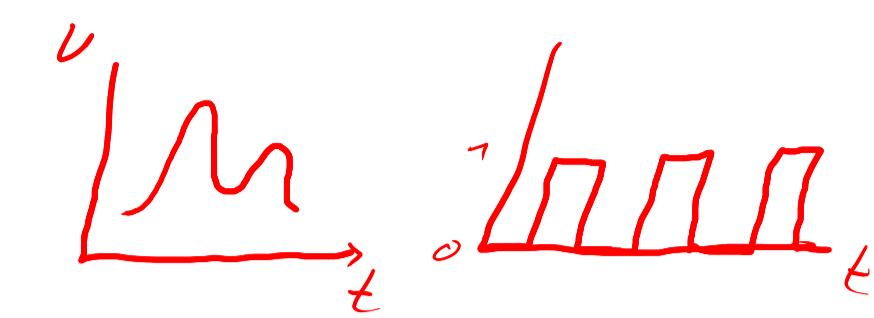
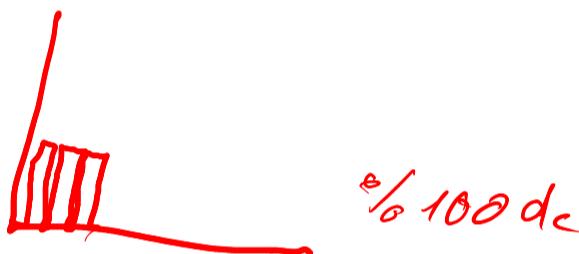
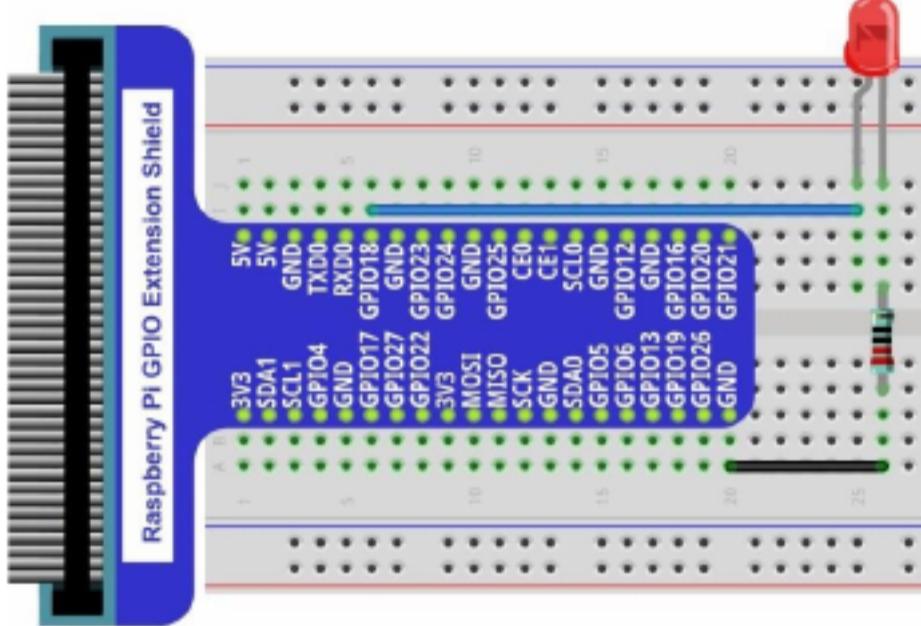
def dongu():
    GPIO.add_event_detect(btnPin, GPIO.FALLING, callback=buttonIslem, bouncetime=300) # Butona basilinca
    while True:
        pass # Programin acik kalmasini saglayan sonsuz pass döngüsü

def bitis():
    GPIO.cleanup()                      # GPIO Cikislarini sifirlar

if __name__ == '__main__':
    print("PROGRAM BASLIYOR!")
    kurulum()
    try:
        dongu()
    except KeyboardInterrupt:           # Klavye ile CTRL+C müdahalesi sirasinda
        bitis()
```

## 5. PWM - Analog, Digital Farkı

PWM: Pulse Width Modulation



```
import RPi.GPIO as GPIO
import time

ledPin = 12 # Fiziksel GPIO Pin tanimi

def kurulum():
    global p
    GPIO.setmode(GPIO.BOARD)
    GPIO.setup(ledPin, GPIO.OUT)
    GPIO.output(ledPin, GPIO.LOW)          # GPIO Fiziksel modunda calistirir
                                         # GPIO Pin kurulumu
                                         # Pin Dusuk Frekansta baslasin

    #PWM
    p = GPIO.PWM(ledPin, 500)           # PWM Frekansini 500Hz
    p.start(0)                         # Baslangic Frekansini 0a tanimlar.

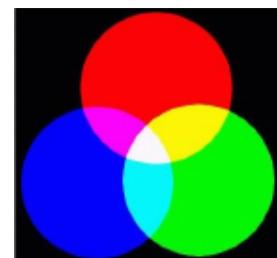
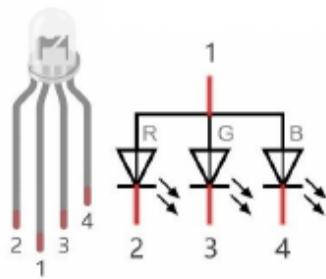
    print('Kurulum basladi. Kurulan pin: %d' %ledPin)

def dongu():
    global p
    while True:
        for dc in range(0,101,1):      # LED Aydinligini yukselt
            p.ChangeDutyCycle(dc)
            time.sleep(0.01)
        time.sleep(1)
        for dc in range(100,-1,-1):    # LED Aydinligini alcalt
            p.ChangeDutyCycle(dc)
            time.sleep(0.01)
        time.sleep(1)

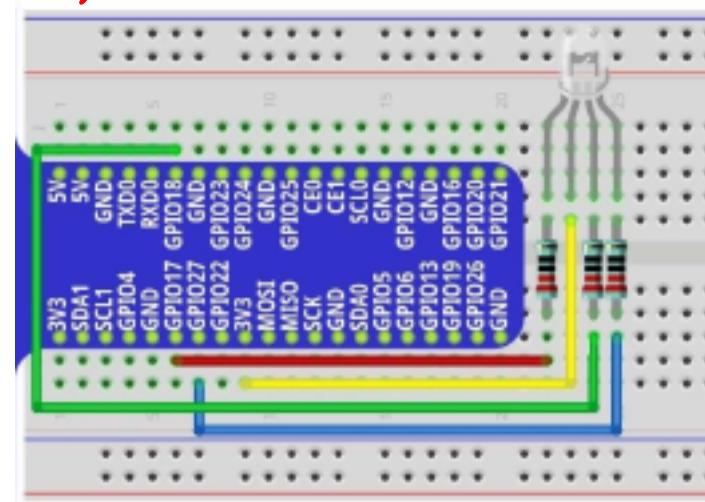
def bitis():
    p.stop()                          # PWM Baslangicini durdur
    GPIO.cleanup()                     # GPIO Cikislarini sifirlar

if __name__ == '__main__':
    print("PROGRAM BASLIYOR!")
    kurulum()
    try:
        dongu()
    except KeyboardInterrupt:
        bitis()                         # Klavye ile CTRL+C mudahalesi sirasinda
```

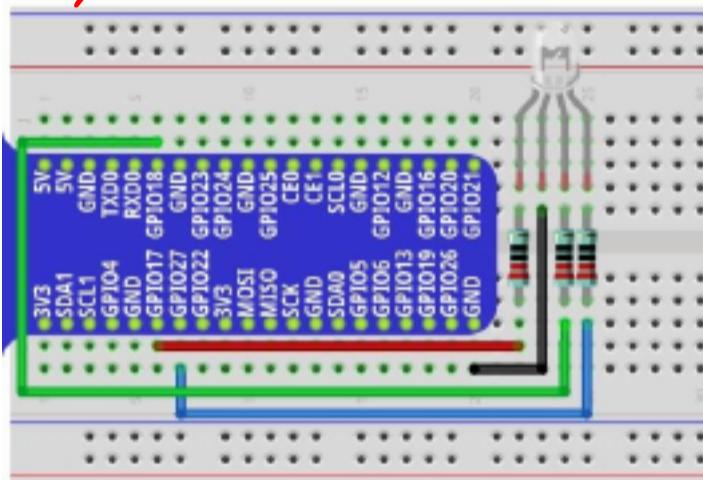
## 6. RGB LED



Tip 1:



Tip 2:



Gerekeler:

3 x 220 $\Omega$  Direng  
1 x RGB Led

```
1 #!/bin/python3
2
3 import RPi.GPIO as GPIO
4 import time
5 import random
6
7 ledPin = [11, 12, 13] # Fiziksel GPIO Pin tanimi
8
9 def kurulum():
10     global pwmK, pwmY, pwmM
11     GPIO.setmode(GPIO.BCM)           # GPIO Fiziksel modunda calistirir
12     GPIO.setup(ledPin, GPIO.OUT)      # GPIO Pin kurulumu
13     GPIO.output(ledPin, GPIO.HIGH)    # Pin Dusuk Frekansta basasin
14     pwmK = GPIO.PWM(ledPin[0], 2000)
15     pwmY = GPIO.PWM(ledPin[1], 2000)
16     pwmM = GPIO.PWM(ledPin[2], 2000)
17     pwmK.start(0)
18     pwmY.start(0)
19     pwmM.start(0)
20
21 def renk_ayar(k_deg, y_deg, m_deg):      # Renk Ayarlamasini RGB degerinin Frekansini ayarlayan fonksiyon
22     pwmK.ChangeDutyCycle(k_deg)
23     pwmY.ChangeDutyCycle(y_deg)
24     pwmM.ChangeDutyCycle(m_deg)
25
26 def dongu():
27     while True:
28         k = random.randint(0,100)
29         y = random.randint(0,100)
30         m = random.randint(0,100)
31         renk_ayar(k, y, m)
32         print('k=%d, y=%d, m=%d' %(k, y, m))
33         time.sleep(2)
34
35 def bitis():
36     pwmK.stop()
37     pwmY.stop()
38     pwmM.stop()
39     GPIO.cleanup()                   # GPIO Cikislarini sifirlar
40
41
42 if __name__ == '__main__':
43     print("PROGRAM BASLIYOR!")
44     kurulum()
45     try:
46         dongu()
47     except KeyboardInterrupt:        # Klavye ile CTRL+C mudahalesi sirasinda
48         bitis()
```

## 7. SMBUS & I<sup>2</sup>C

SMBUS = System Management Bus

BUS Nedir?

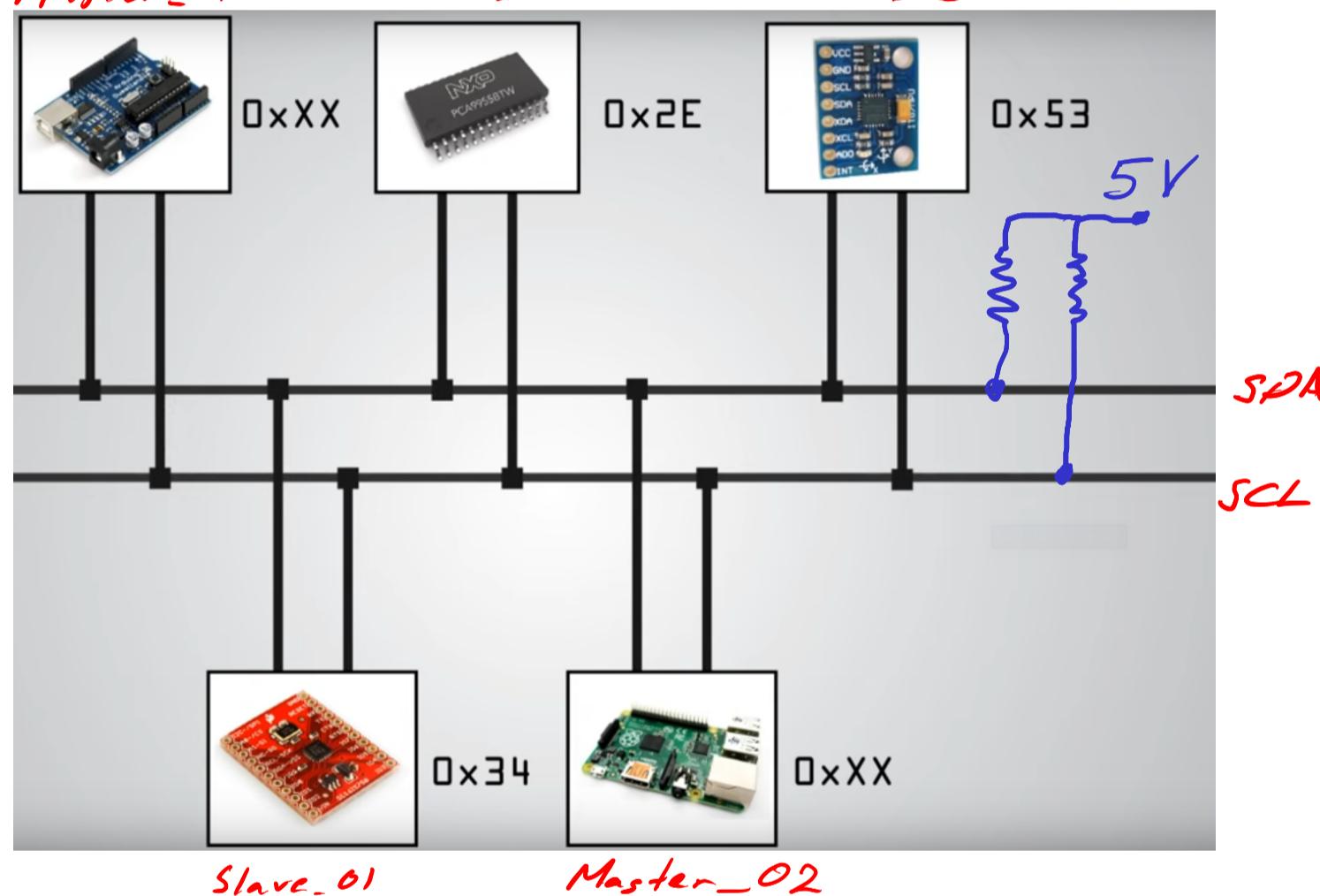
Bus, Veri aktarımını genel bir yolla sağlayan sistemdir.

ÖRN

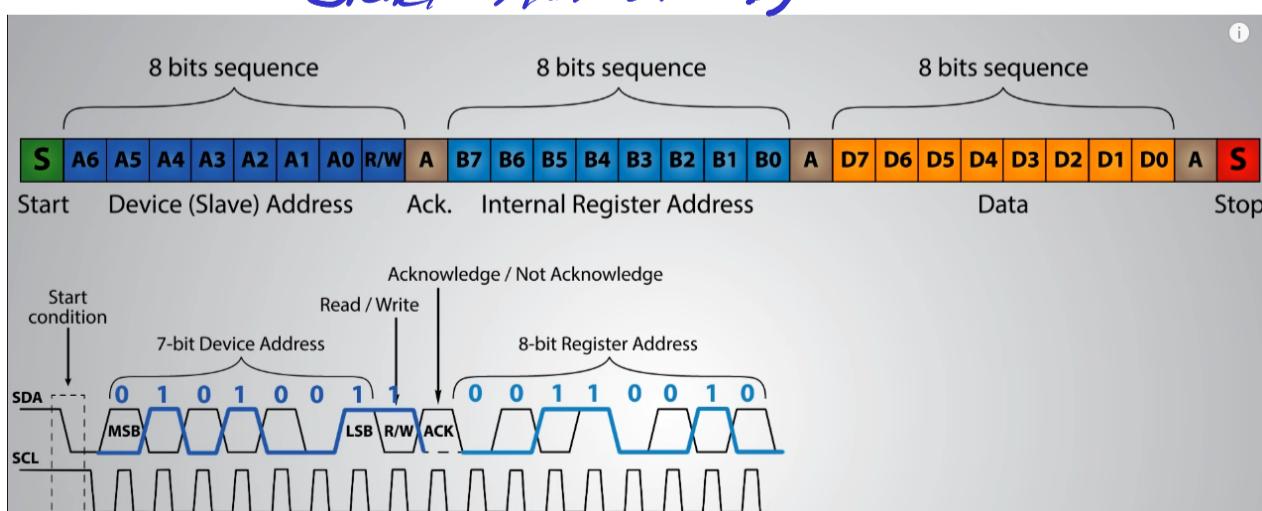
USB = Universal Serial Bus demektir!

SMBUS temel görevi Voltajı ve ls. ölçümü yapmaktadır.

Master\_01      Slave\_02      Slave\_03



Genel Kütür :)



Cihaz: 01010011 ise adresi = 0x53 olur