

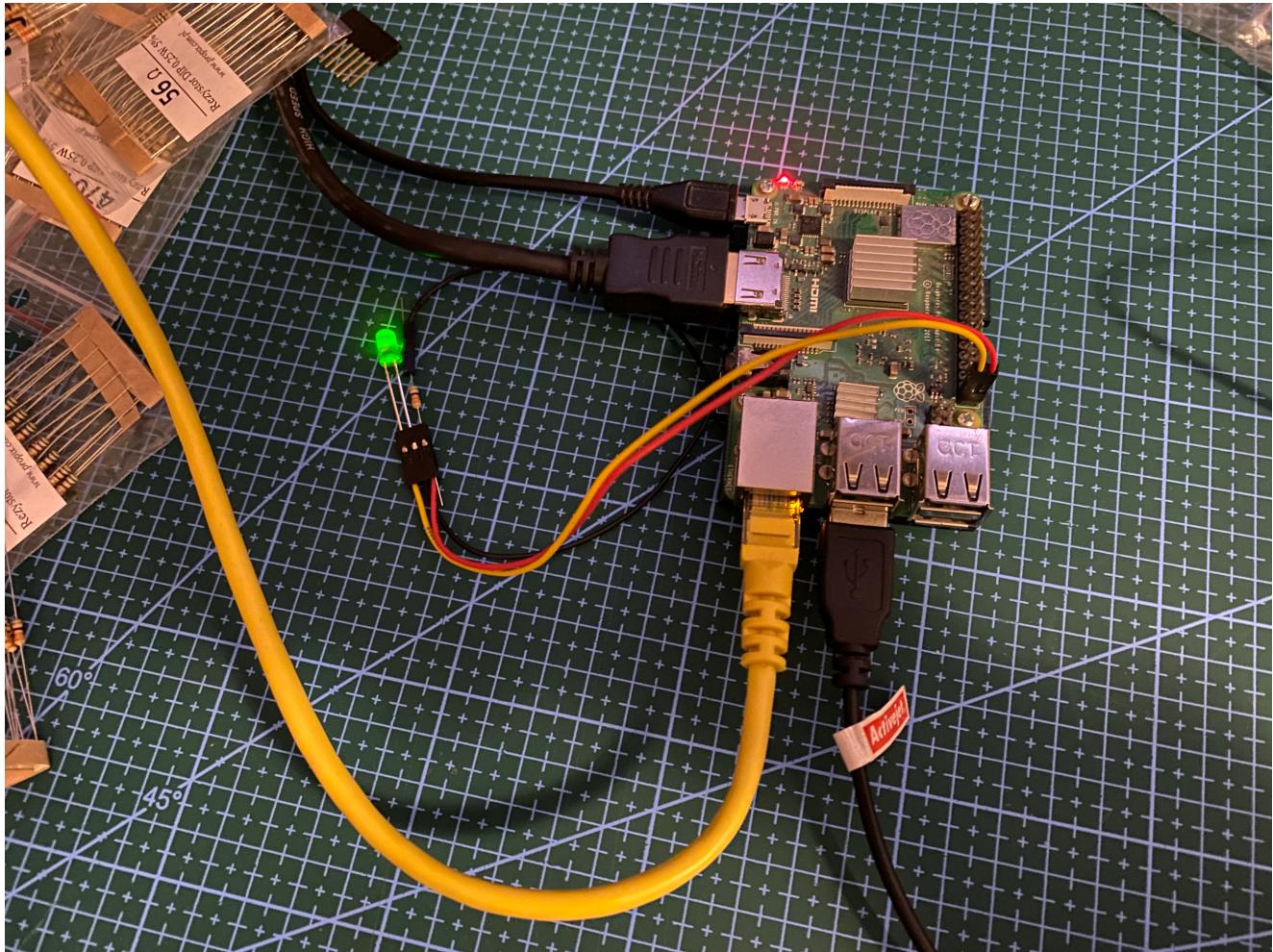
Baraniecki Karol Byczko Maciej	Prowadzący: Dr inż. Dominik Żelazny	Numer ćwiczenia laboratoria 17
PT 16:30 TP	Temat ćwiczenia: Raspberry Pi	Ocena:
Grupa: D	Data wykonania: 9 grudnia 2021	

1 Zagadnienia do opracowania

1. Urządzenia RPI (wersja 4)
2. • **Raspberry Pi** - platforma komputerowa stworzona przez Raspberry Pi Foundation. Urządzenie składa się z pojedynczego obwodu drukowanego i zostało wymyślone, by wspierać naukę podstaw informatyki.
 - interfejs GPIO (numeracja pinów) - Wejście-wyjście ogólnego przeznaczenia, GPIO (od ang. general-purpose input/output), pin interfejsu do komunikacji między elementami systemu komputerowego (na przykład między mikroprocesorem a urządzeniami peryferyjnymi).
3. Środowisko programistyczne
4. • zasady pracy z systemem Linux - NIE WYKONUJ KOMENDY `rm -rf /*`
 - interfejs wiring Pi - biblioteka do obsługi GPIO
 - język python, zasady instalacji pakietów (pip) - język programowania wysokiego poziomu ogólnego przeznaczenia, o rozbudowanym pakiecie bibliotek standardowych, którego ideą przewodnią jest czytelność i klarowność kodu źródłowego. Jego składnia cechuje się przejrzystością i zwięzłością.
 - sterowanie sygnałami wejściowymi/wyjściowymi
 - RPi.GPIO - Raspberry Pi GPIO - Moduł do sterowania diodami
5. Zestaw prototypowy
 - picoboard - Zestaw prototypowy dla Raspberry Pi
 - zasady podłączania diody - diodę należy podłączyć przez opornik aby się nie spalił pin w Raspberry.
 - interfejs 1-wire i czujnik temperatury DS18B20

2 Zadania do wykonania

1. Testy urządzenia
 - podłączyć urządzenia: zasilanie + kabel ethernet



- określenie adresu IP (podłączyć monitor, adres IP podany będzie na ekranie (ew. komenda ifconfig))

```

Plik Edycja Karty Pomoc
pi@raspberr... ~ pi@raspberr... ~
pi@raspberrypi:~ $ ip -c a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP group default qlen 1000
    link/ether b8:27:eb brd ff:ff:ff:ff:ff:ff
    inet 192.168.1.40/24 brd 192.168.1.255 scope global dynamic noprefixroute eth0
        valid_lft 36154sec preferred_lft 75354sec
    inet6 fe80::1b2f:195b:7e4:3049/64 scope link
        valid_lft forever preferred_lft forever
3: wlan0: <NO-CARRIER,BROADCAST,MULTICAST,UP> mtu 1500 qdisc pfifo_fast state DOWN group default qlen 1000
    link/ether b8:27:eb:15:77:a9 brd ff:ff:ff:ff:ff:ff
4: tun0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UNKNOWN group default qlen 100
    link/none
    inet 10.8.0.1/24 brd 10.8.0.255 scope global tun0
        valid_lft forever preferred_lft forever
    inet6 fe80::3959:770:42f0:4b51/64 scope link stable-privacy
        valid_lft forever preferred_lft forever
pi@raspberrypi:~ $ scrot

```

- zalogowanie na konsole via putty (proszę nie zmieniać hasła) (dostęp: pi, raspberry)

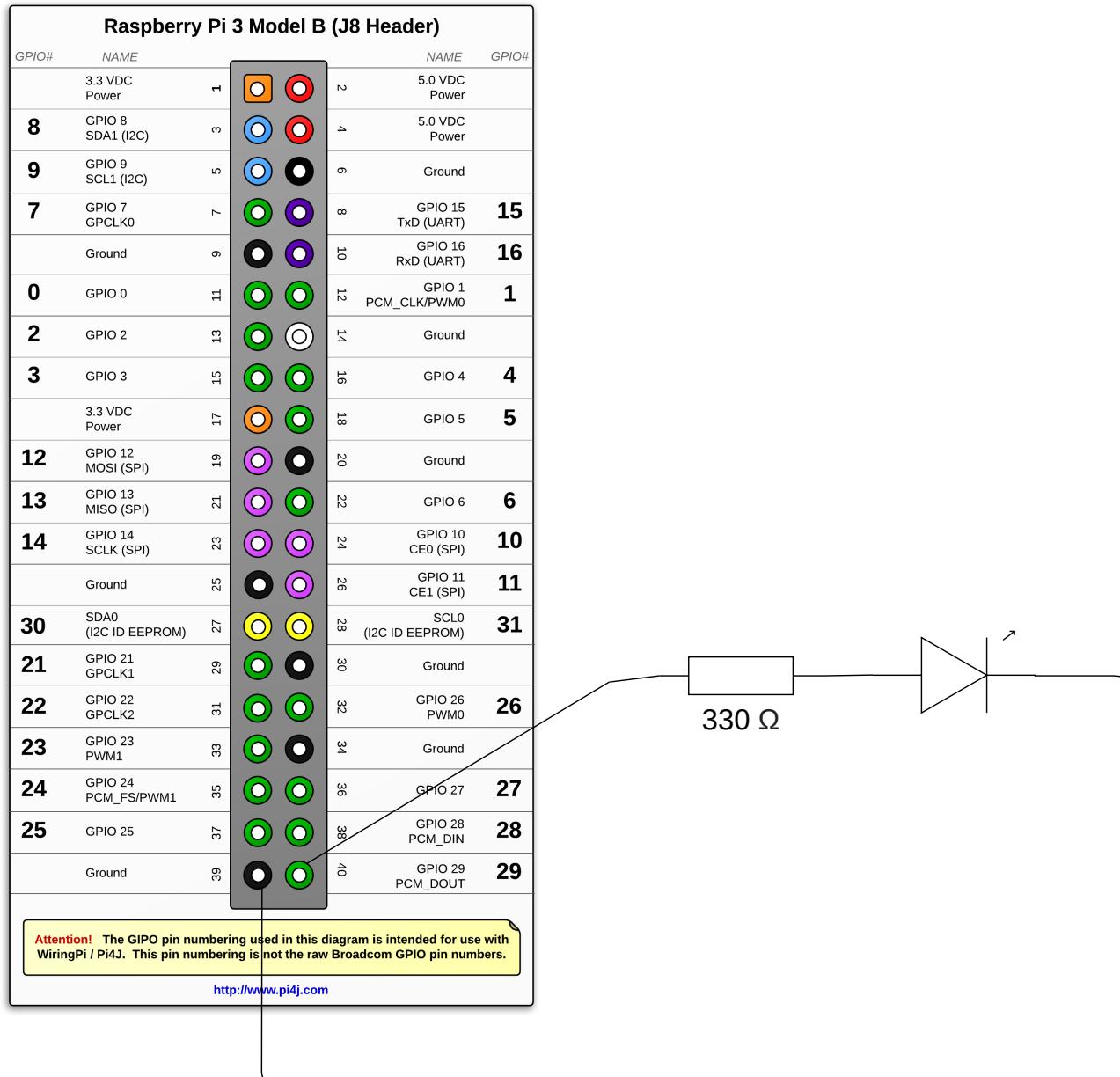
Zalogowaliśmy się za pomocą OpenSSH w terminalu, gdyż PuTTY wymaga dodatkowych instalacji a robi to samo.

The screenshot shows a terminal window with a dark background and light-colored text. It displays the following session:

```
pi@raspberrypi: ~  
karol@ktp ~ % ssh pi@192.168.1.40  
pi@192.168.1.40's password:  
Linux raspberrypi 5.10.63-v8+ #1459 SMP PREEMPT Wed Oct 6 16:42:49 BST 2021 aarch  
64  
  
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: Thu Dec 9 18:11:51 2021 from 192.168.1.19  
pi@raspberrypi:~ $ █
```

2. Sterowanie diodą

- podłączenie diody i rezystora do RPi (przed podłączaniem zasilania proszę poprosić prowadzącego o sprawdzenie połączeń)



- sterowanie z poziomu programu gpio

```

1 gpio -1 mode 40 out # ustawienie pinu jako wyjscie
2 gpio -1 write 40 1 # wlaczenie diody
3 gpio -1 write 40 0 # wylaczenie diody
4 gpio -1 blink 40 # wlaczenie trybu migania diody

```

- program w pythonie

```

1 import time
2 import RPi.GPIO as GPIO
3
4 GPIO.setmode(GPIO.BCM)
5 PIN = 40
6
7 GPIO.setup(PIN, GPIO.OUT)
8 while True:
9     time.sleep(0.1)
10    GPIO.output(PIN, 0)
11    time.sleep(0.1)
12    GPIO.output(PIN, 1)
13

```

3 Wnioski

Film Video z działającym układem

Zadanie było w miarę proste, pogubiliśmy się przy numeracji pinów a poza tym wszystko potoczyło się bez większych problemów.