



MakerLab
Ljubljana

ICT academy

Internet stvari: Raspberry PI

LTFE

Laboratorij za telekomunikacije

LMMFE

Laboratorij za multimedijo

IKT

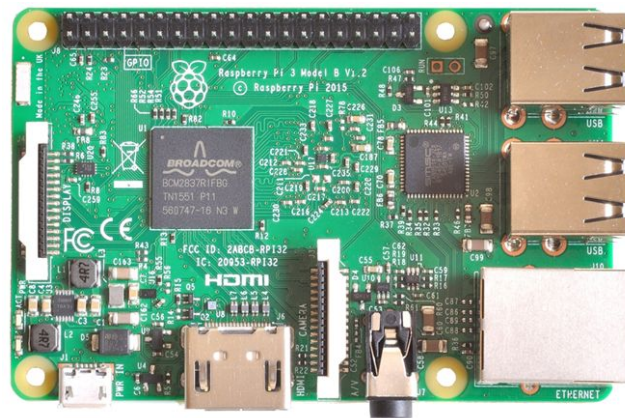
Katedra za informacijske
in komunikacijske
tehnologije



Univerza v Ljubljani
Fakulteta za *elektrotehniko*

Ljubljana, september 2021

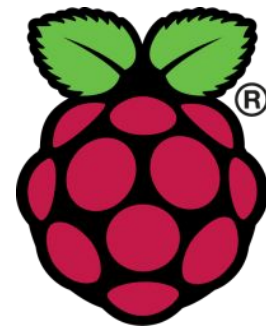
Del 1: Uvod v delo z RaspberryPI



Namestitev OS na SD kartico

- Prenesemo sliko z operacijskim sistemom iz strani:
<https://www.raspberrypi.org/software/operating-systems/>
- Namestimo orodje **Raspberry Pi Imager** iz strani
<https://www.raspberrypi.org/software/>, ki nam omogoča pisanje slike na SD kartico.
- Vstavimo SD kartico v računalnik.
- Zaženemo Raspberry Pi Imager in namestimo OS.
 - Izberemo možnost, da počistimo vsebino celotne kartice pred pisanjem.ž
- **Gradiva za delavnico lahko najdete na:**
 - <https://github.com/leon11s/raspberry-pi-icta>

Kaj je Raspberry Pi?



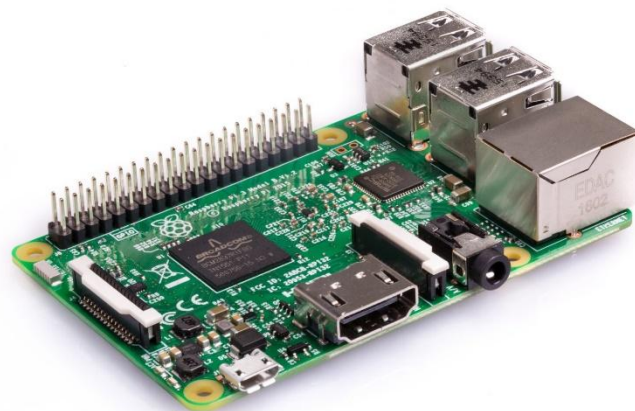
- Poceni računalnik v velikosti kreditne kartice
- Razvit l. 2009 - Raspberry Pi foundation (UK)
- <5W porabe
- Linux podpora
- Ima USB porte, ki omogočajo da priključimo različne naprave
- Prvotna ideja: izobraževanje -> nizka cena -> omejitve
- Zaradi nizke cene je prišlo do uporabe na ostalih področjih
- Uradna stran: <https://www.raspberrypi.org/>
- Ogromno gradiva na internetu
- ***Na kratko: poceni Linux računalnik z GPIO pini***

Raspberry Pi

- Modeli:
 - Raspberry Pi 1 model B (+) in A (februar 2012)
 - Raspberry Pi 2 model B (februar 2015)
 - Raspberry Pi 3 model B (februar 2016)
 - Raspberry Pi 3 model B+ (marec 2018)
 - Raspberry Pi 4 model B (2019)
- Cena okoli 35€ + dokup ostale opreme (napajalnik, SD kartica)
- Slabša HW zaščita
 - Z nepazljivostjo je možno uničiti čip
 - GPIO pini delujejo na 3.3V – med pini pa je tudi 5V napajanje, ki v primeru kontakta z 3.3V pinom lahko prekuri tisti pin ali cel čip
 - Občutljivost na statično elektriko
- Uporaba: nadomestek namiznega računalnika, domači strežnik, nadomestek za mikrokrmilnik

Raspberry Pi 3 Model B

- Mi bomo uporabljali RP3 Model B
- Specifikacije:
 - 1.2Ghz 64-bit quad-core ARMv8 CPU
 - 802.11n Wireless LAN
 - 1 GB RAM
 - Bluetooth 4.1
 - Bluetooth Low Energy (BLE)
 - 40-pin header with GPIOs
 - audio out, an SD card slot, an Ethernet port, 4 USB ports, camera interface (CSI), display interface (DSI), and a micro USB (napajanje)
- Operacijski sistem: Linux
- Priporočeno napajanje je 5V in 2,5A



Hardware	Specifications
SoC	Broadcom BCM2836 (CPU, GPU, DSP, SDRAM)
CPU	900 MHz quad-core ARM Cortex A7 (ARMv7 instruction set)
GPU	Broadcom VideoCore IV @ 250 MHz
More GPU info	OpenGL ES 2.0 (24 GFLOPS); 1080p30 MPEG-2 and VC-1 decoder (with license); 1080p30 h.264/MPEG-4 AVC high-profile decoder and encoder
Memory	1 GB (shared with GPU)
USB ports	4
Video input	15-pin MIPI camera interface (CSI) connector
Video outputs	HDMI, composite video (PAL and NTSC) via 3.5 mm jack
Audio input	I ² S
Audio outputs	Analog via 3.5 mm jack; digital via HDMI and I ² S
Storage	MicroSD
Network	10/100Mbps Ethernet
Peripherals	17 GPIO plus specific functions, and HAT ID bus
Power rating	800 mA (4.0 W)
Power source	5 V via MicroUSB or GPIO header
Size	85.60mm × 56.5mm
Weight	45g (1.6 oz)

Raspbian

- **Operacijski sistem prilagojen za RaspberryPi**
 - Zasnovan na Debian Linux
 - Vsebuje vse potrebne gonilnike za RaspberryPi
 - Vsebuje tudi vse dodatne programe in skripte potrebne za delovanje
- **Slika pripravljena za namestitev na SD kartico**
 - Potrebno dodatno orodje za kopiranje na kartico
 - Disk imager, dd, ipd.
- **<http://raspbian.org/>**
 - cca. 1.3GB – z grafičnim vmesnikom
 - cca. 292MB – minimalno (brez grafičnega vmesnika)
- **Orodje NOOBS – slika za avtomatsko instalacijo Raspbian in nekaterih ostalih distribucij**

Del 2: Priprava okolja in prva uporaba

Del 3: Osnove uporabe Linux terminala

Pregled ukazov

- ls
- pwd
- cd
- su
- sudo
- exit
- mv
- cp
- rm
- mkdir
- nano
- shutdown
- ifconfig
- iwconfig

Del 4: Konfiguracija, omrežne nastavitve in varnost

- **Upravljanje na daljavo (SSH)**
- **Pridobimo kodo za tečaj iz računa Github**
- **Nastavitev statičnega IP naslova**
- **Požarni zid**

Sprememba default gesla

- Vsak RP, ki ima nameščen Raspbian ima default uporabniško ime (pi) in geslo (raspberry)
- S tem imamo root dostop do RPja
- Spremenimo geslo:
 - Možnost 1) preko raspi-cofig aplikacije

```
sudo raspi-config
```

- Možnost 2) preko terminala

```
passwd  
sudo passwd pi
```

Dodajanje novega uporabnika

- Omejena dovoljenja
 - `sudo adduser <ime_uporabnika>`
 - Uporabnik dobi v home mapi svojo mapo
- Če želimo zamenjati porabnika v terminalu uporabimo ukaz
 - `su <ime_uporabnika>`
- Preverimo kateri uporabnik smo
 - `whoami`
 - Oz. pogledamo v terminalu
- Dodamo uporabniku sudo pravice
 - `sudo adduser <ime_uporabnika> sudo`
 - Preverimo če imamo sudo pravice
 - Npr. `sudo mkdir neki`
- Odstranimo uporabnika
 - `sudo userdel -r <ime_uporabnika>`
 - `-r` odstrani mapo uporabnika

Zahtevamo geslo za sudo za vsako operacijo

- Po defaultu ni zahtevano
- Če imamo napravo povezano direktno na internet je to dobra praksa, saj napadalcu preprečimo nastavitve
- Postopek:

```
sudo nano /etc/sudoers.d/010_pi-nopasswd
```

```
pi ALL=(ALL) PASSWD: ALL
```

- Shranimo datoteko

Preverimo posodobitve sistema

- Posodobimo vse pakete na sistemu:

```
sudo apt-get update
```

```
sudo apt-get dist-upgrade
```

- Redno posodabljanje modula za ssh-server

```
sudo apt-get install openssh-server
```

Avtomatizacija opravil

- Raspberry Pi Cron Jobs
- Uporabljammo za zaganjanje skript ob določenem času
- Skript ni potrebno več ročno zagnati, ampak se to zgodi v ozadju
- Možna uporaba:
 - Upravljanje back-upov
 - Posodobitve
 - Pošiljanje mailov
 - Dejansko vse kar želimo, da se ponavlja na določen interval
- Zagon:
 - `sudo crontab -e`
 - Pri prvem zagonu potrebno izbrati urejevalnik besedila
 - Izberemo Nano (2 + Enter)

- Izdelamo skripto
 - `sudo nano /home/pi/RunACronTask.sh`
- V skripto prilepimo naslednjo kodo in shranimo datoteko:
 - `#!/bin/sh`
 - `date >>/home/pi/cron.txt`
 - Ta skripta doda trenutni čas in datum v datoteko `corn.txt`
- Damo naši skripti dovoljenje za zagon:
 - `sudo chmod +x /home/pi/RunACronTask.sh`
- Run crontab with the -e flag to edit the cron table:
 - `crontab -e`
- V datoteko dodamo ukaz, kdaj in katero skripto zaženemo
 - `*/1 * * * * pi /home/pi/RunACronTask.sh`

- The command `crontab` (cron table) is used to edit the list of scheduled tasks in operation, and is done on a per-user basis; each user (including root) has their own crontab.

```
# * * * * *    command to execute
# T T T T T
# | | | | |
# | | | | |
# | | | | |_____ day of week (0 - 7) (0 to 6 are Sunday to Saturday,
or use names; 7 is Sunday, the same as 0)
# | | | |_____ month (1 - 12)
# | | |_____ day of month (1 - 31)
# | |_____ hour (0 - 23)
# |_____ min (0 - 59)
```

Avtomatsko posodabljanje

- Nova skripta: `nano /home/pi/Update.sh`
- Kopiramo v skripto:
 - `#!/bin/bash -e`
 - `/usr/bin/sudo apt-get update`
 - `/usr/bin/sudo apt-get -y upgrade`
 - `/usr/bin/sudo rpi-update`
 - `/usr/bin/sudo shutdown -r now`
- Dovoljenje za zagon: `chmod +x /home/pi/Update.sh`
- Nastavimo Cron Job:
 - `crontab -e`
 - V crontab (every Monday at 1:00 am):
 - `* 1 * * 1 pi /home/pi/Update.sh`

Namestimo fail2ban

- V primeru uporabe RPja kot serverja (ssh, webserver...) imamo v požarnem zidu „luknje“, da dovolimo prometu skozi
- Fail2ban je skener v Python-u, ki preverja log datoteke za morebitne sumljive dogodke (večkratni brutute-force vstopi) in posodobi požarni zid, da to preprečimo
- Namestitev

```
sudo apt-get install fail2ban
```

- Default nastavitve se nahajajo naslednji datoteki:
/etc/fail2ban/jail.conf
 - Te datoteke ne smemo spreminjati

- Če želimo spreminjati nastavitve to storimo v datoteki:
 - `/etc/fail2ban/jail.local`
 - `sudo nano /etc/fail2ban/jail.local`
- V ustvarjeni datoteki nastavimo konfiguracijo za SSH:

```
[ssh]

enabled = true
port = ssh
filter = sshd
logpath = /var/log/auth.log
bantime = 900
banaction = iptables-allports
findtime = 900
maxretry = 3
```

- Za uveljavitev novih nastavitvev, zaženemo:
 - `sudo service fail2ban restart`
- Več o delovanju [fail2ban](https://www.fail2ban.org/)

Del 5: Uvod v GPIO

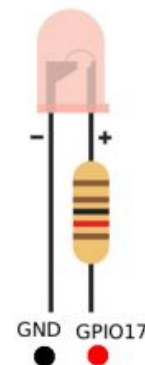


3.3V PWR	1	2	5V PWR
I2C1 SDA	3	4	5V PWR
I2C1 SCL	5	6	GND
GPIO 4	7	8	UART0 TX
GND	9	10	UART0 RX
GPIO 17	11	12	GPIO 18
GPIO 27	13	14	GND
GPIO 22	15	16	GPIO 23
3.3V PWR	17	18	GPIO 24
SPI0 MOSI	19	20	GND
SPI0 MISO	21	22	GPIO 25
SPI0 SCLK	23	24	SPI0 CS0
GND	25	26	SPI0 CS1
Reserved	27	28	Reserved
GPIO 5	29	30	GND
GPIO 6	31	32	GPIO 12
GPIO 13	33	34	GND
GPIO 19	35	36	GPIO 16
GPIO 26	37	38	GPIO 20
GND	39	40	GPIO 21

Uporaba GPIO pinov

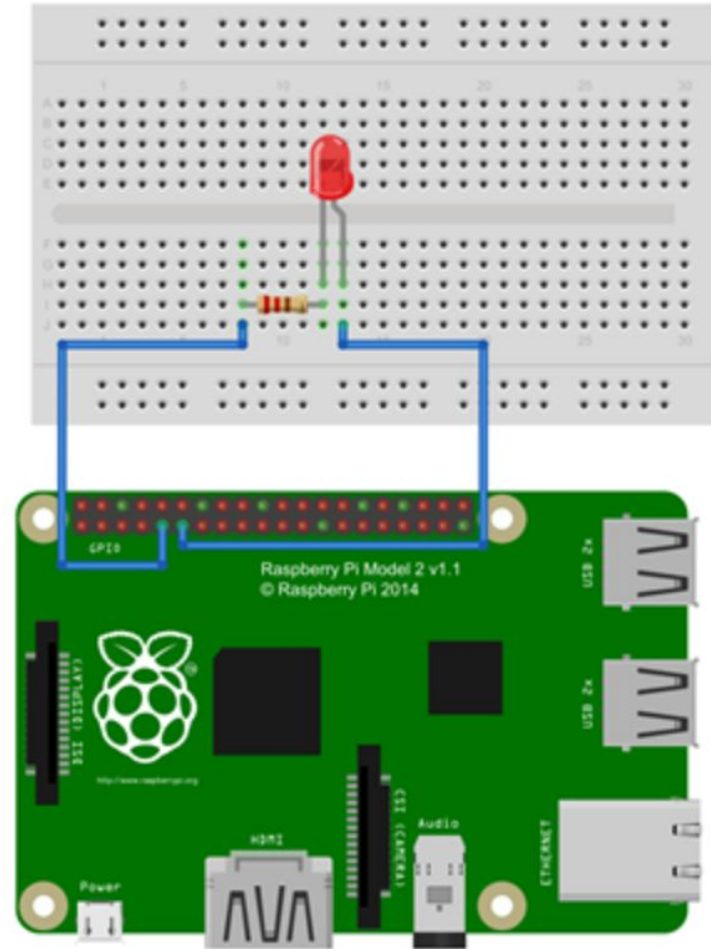
- RaspberryPi izklopite
- LED diodo priklopite med Ground pin (09) in GPIO17 (GPIO GEN0) (11).
- To sta tretji in četrty pin v levi vrsti (označena z piko).
- Dodamo upor
- RaspberryPi prižgite nazaj.

Pin#	NAME		NAME	Pin#
01	3.3v DC Power		DC Power 5v	02
03	GPIO02 (SDA1 , I2C)		DC Power 5v	04
05	GPIO03 (SCL1 , I2C)		Ground	06
07	GPIO04 (GPIO_GCLK)		(TXD0) GPIO14	08
09	Ground		(RXD0) GPIO15	10
11	GPIO17 (GPIO_GEN0)		(GPIO_GEN1) GPIO18	12
13	GPIO27 (GPIO_GEN2)		Ground	14
15	GPIO22 (GPIO_GEN3)		(GPIO_GEN4) GPIO23	16
17	3.3v DC Power		(GPIO_GEN5) GPIO24	18
19	GPIO10 (SPI_MOSI)		Ground	20
21	GPIO09 (SPI_MISO)		(GPIO_GEN6) GPIO25	22
23	GPIO11 (SPI_CLK)		(SPI_CE0_N) GPIO08	24
25	Ground		(SPI_CE1_N) GPIO07	26
27	ID_SD (I2C ID EEPROM)		(I2C ID EEPROM) ID_SC	28
29	GPIO05		Ground	30
31	GPIO06		GPIO12	32
33	GPIO13		Ground	34
35	GPIO19		GPIO16	36
37	GPIO26		GPIO20	38
39	Ground		GPIO21	40



Vaja 1: Led osnovno

- Led on/off
- PIN 9 -> GND
- PIN 11 -> GPIO17



Koda

- Ustvarimo novo datoteko -> *nano 01_led_osnovno.py*

- Koda

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)
GPIO.setwarnings(False)
GPIO.setup(17,GPIO.OUT)
print("LED on")
GPIO.output(17,GPIO.HIGH)
time.sleep(2)
print("LED off")
GPIO.output(17,GPIO.LOW)
GPIO.cleanup()
```

- Zagon programa -> *sudo python3 01_led_osnovno.py*

Vaja 2: Led blink

- Ustvarimo novo datoteko -> *nano 02_led_blink.py*

- Koda

```
import RPi.GPIO as GPIO
```

```
GPIO.setmode(GPIO.BCM)
```

```
GPIO.setup(17, GPIO.OUT)
```

```
p = GPIO.PWM(17, 0.5)
```

```
p.start(1)
```

```
input('Press enter to stop!')
```

```
print("Stopping!")
```

```
p.stop()
```

```
GPIO.cleanup()
```

- Zagon programa -> *sudo python3 02_led_blink.py*

Vaja 3: Led PWM

- Ustvarimo novo datoteko LED.py -> *nano 03_led_pwm.py*
- Koda

```
import time
import RPi.GPIO as GPIO

GPIO.setmode(GPIO.BOARD)
GPIO.setup(11, GPIO.OUT)

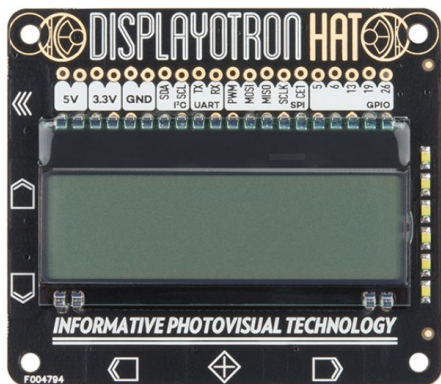
p = GPIO.PWM(11, 50) # channel=11 frequency=50Hz
p.start(0)

try:
    while True:
        for dc in range(0, 101, 5):
            p.ChangeDutyCycle(dc)
            time.sleep(0.1)
        for dc in range(100, -1, -5):
            p.ChangeDutyCycle(dc)
            time.sleep(0.1)
except KeyboardInterrupt:
    pass
finally:
    print("Stopping!")
    p.stop()
    GPIO.cleanup()
```

- Zagon programa -> *sudo python3 03_led_pwm.py*

Del 6: Upravljanje z LED preko spletnega vmesnika

Del 7: Individualne vaje



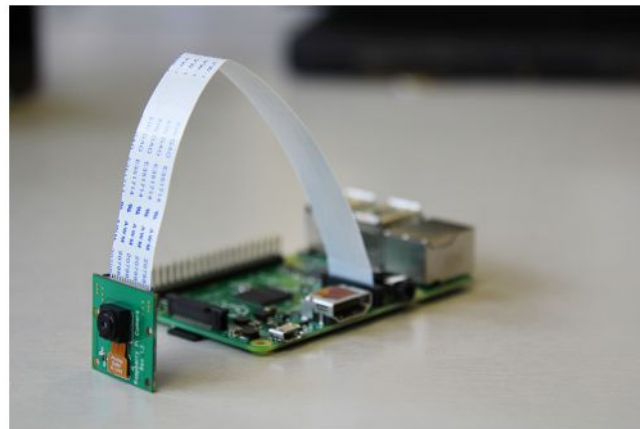
Display-O-tron HAT



Sense HAT



Rainbow HAT



Kamera

Vabljeni k sodelovanju!

info@ltfe.org

+386 1 476 8988

www.ltfe.org

LTFE

Laboratorij za telekomunikacije

LMMFE

Laboratorij za multimedijo

IKT

Katedra za informacijske
in komunikacijske
tehnologije



Univerza v Ljubljani
Fakulteta *za elektrotehniko*