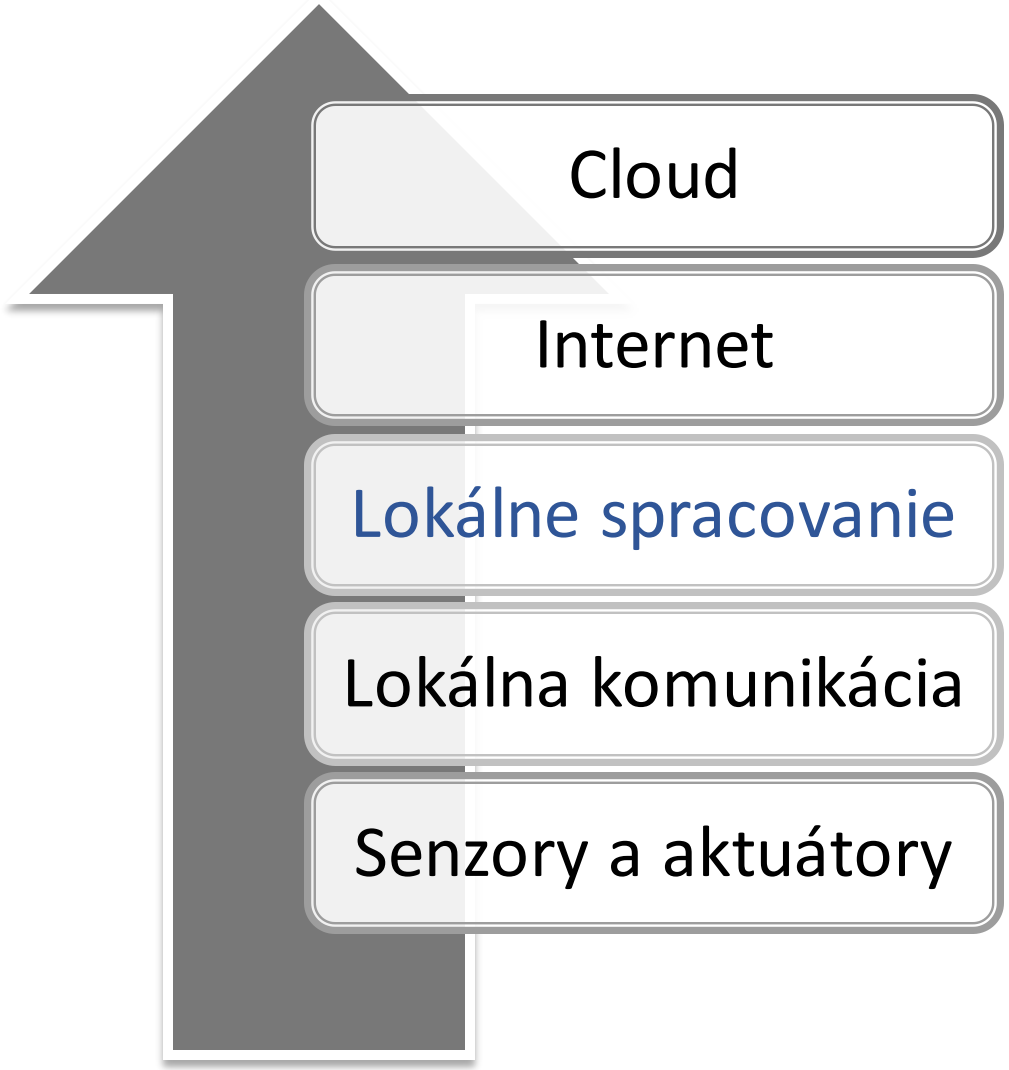
Raspberry Pi

Základy internetu vecí

ÚINF PF UPJŠ

EURÓPSKA ÚNIA
Európsky sociálny fond
Európsky fond regionálneho rozvoja
OPERAČNÝ PROGRAM
ĽUDSKÉ ZDROJE
MINISTERSTVO ŠKOLSTVA, VEDY, VÝSKUMU A ŠPORTU SLOVENSKEJ REPUBLIKY
UNIVERZITA MATEJA BELA

IoT - komponenty



Hardvér

Softvér

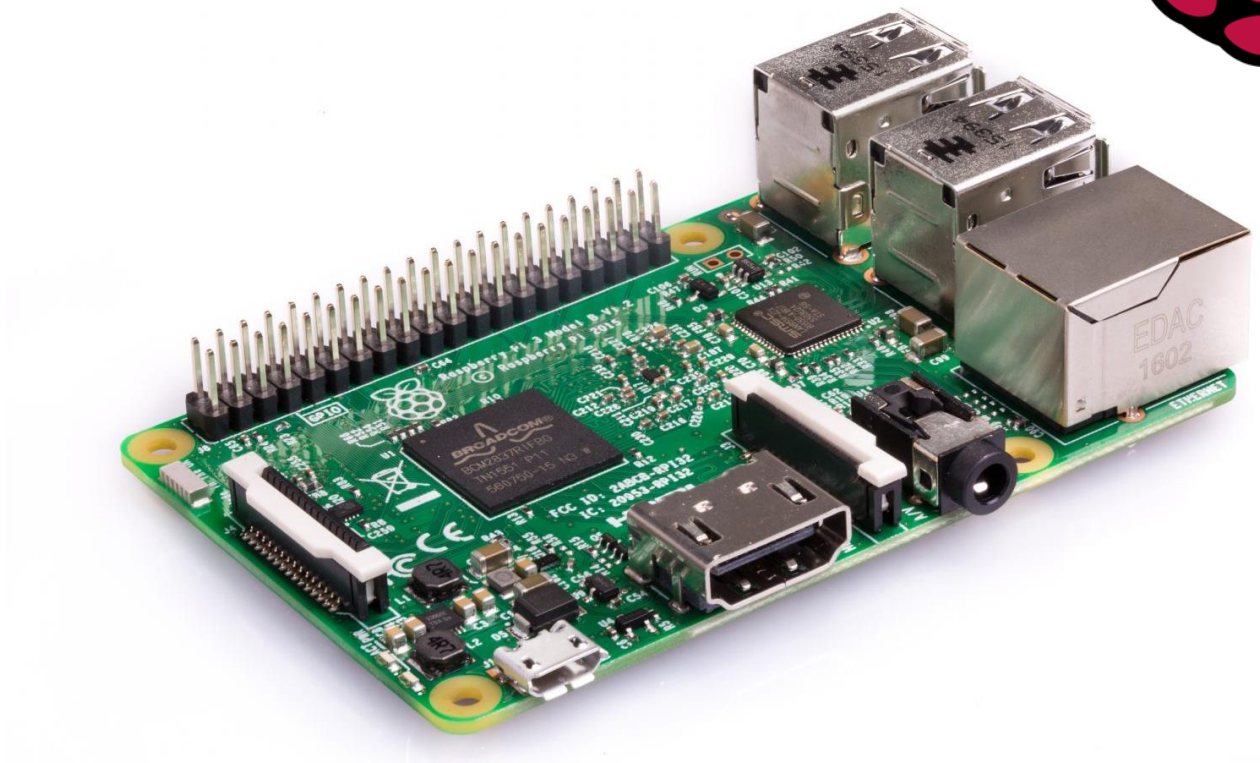
Protokoly

Hardvér: Jednodoskový počítač (SBC)

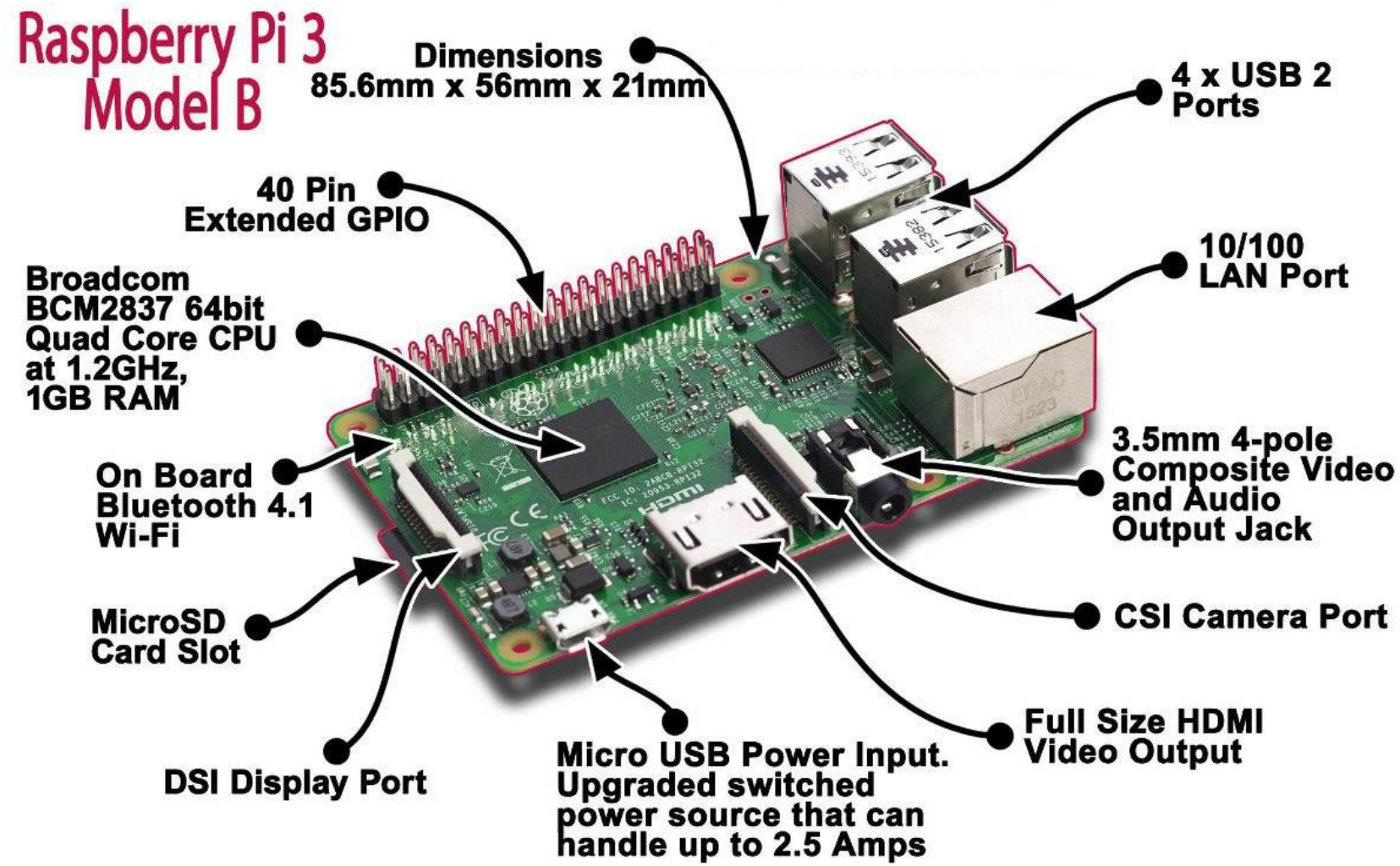
- **Kompletný počítač na jednej doske**
- Prvý SBC: dyna-micro (1976)

Raspberry Pi

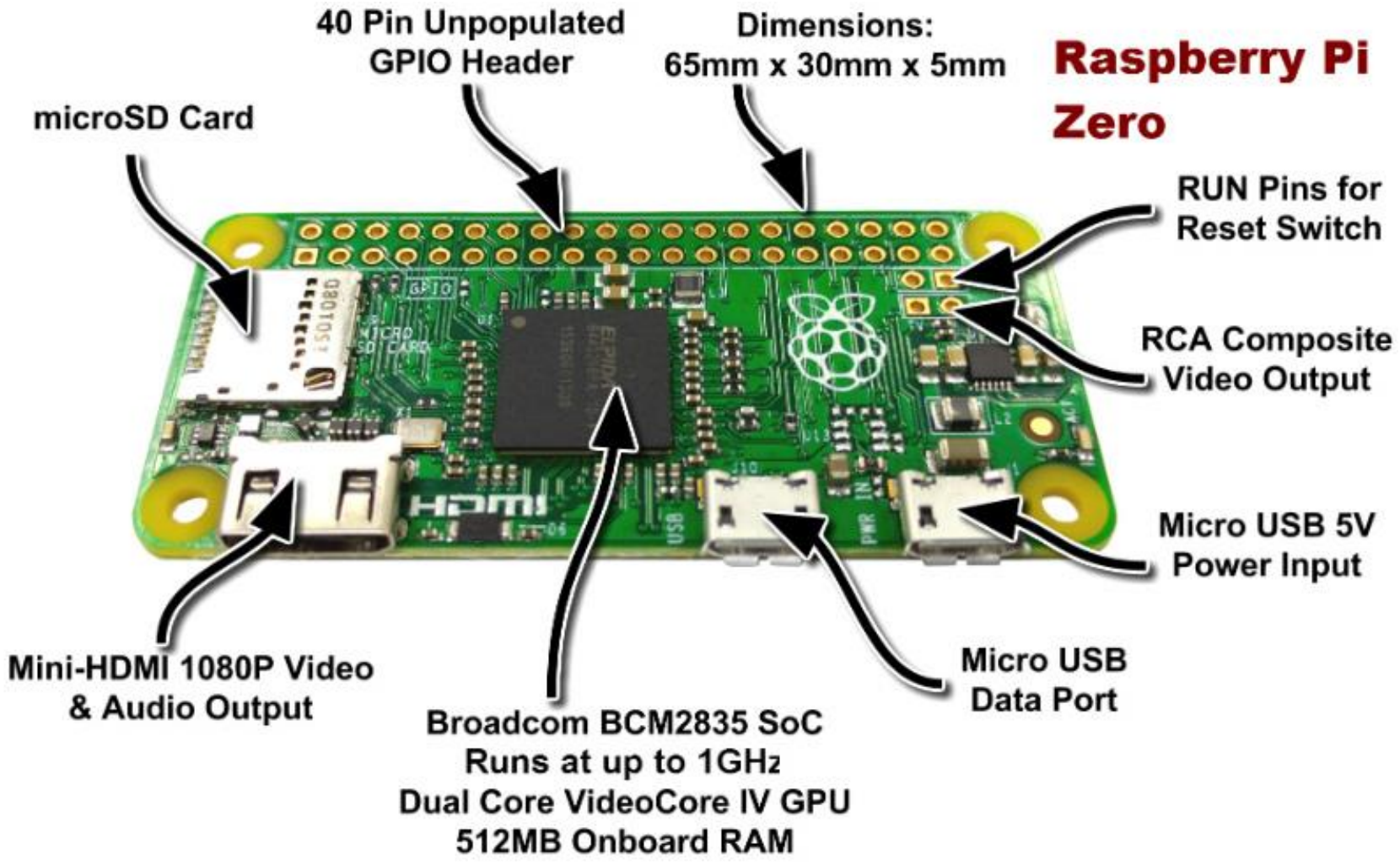
- Jednodoskový počítač (vyvinutý pre edukačné účely):
  - CPU: 1.2 GHz 64-bit quad-core **ARM** Cortex-A53
  - SDRAM: **1 GB**
  - 5V, 300 mA (1.5 W)
  - 17× GPIO
  - MicroSDHC slot
  - 10/100 Mbit/s Ethernet,
  - 802.11n wireless
  - Bluetooth 4.1
  - HDMI, USB
  - **Linux, Windows 10 IoT Core**
  - **40 €**



Raspberry Pi 3 (~38 EUR)



Raspberry Pi Zero (~5 USD/10 USD)

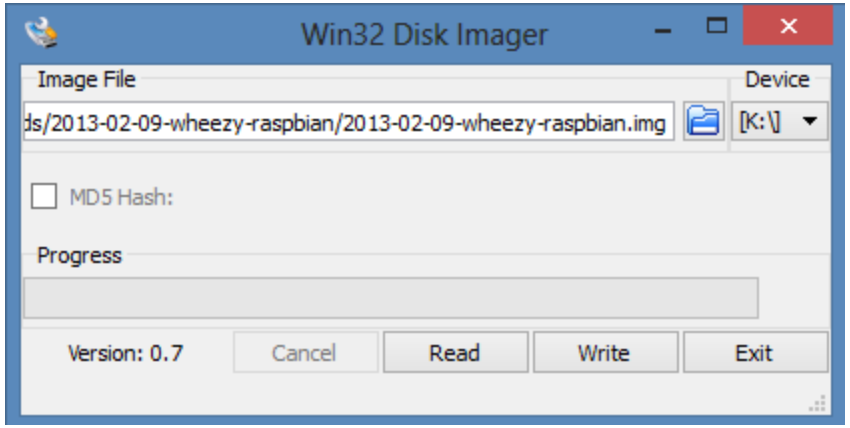


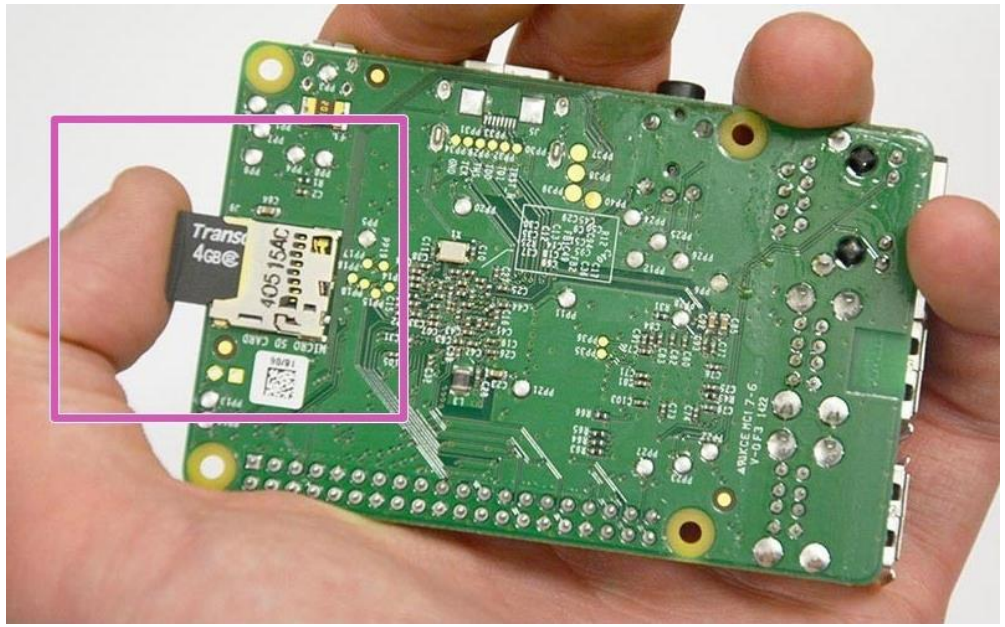
SoC – System on Chip

- **Integrovaný obvod**, ktorý obsahuje všetky **základné časti** počítača:
  - CPU, GPU, RAM,
    WiFi, Bluetooth, DSP (audio, video)…
  - *Broadcom BCM-2837* (na RPi3)
  - *Broadcom BCM-2836* (na RPi2)
    - **ARM** procesor

Operačný systém na RPi

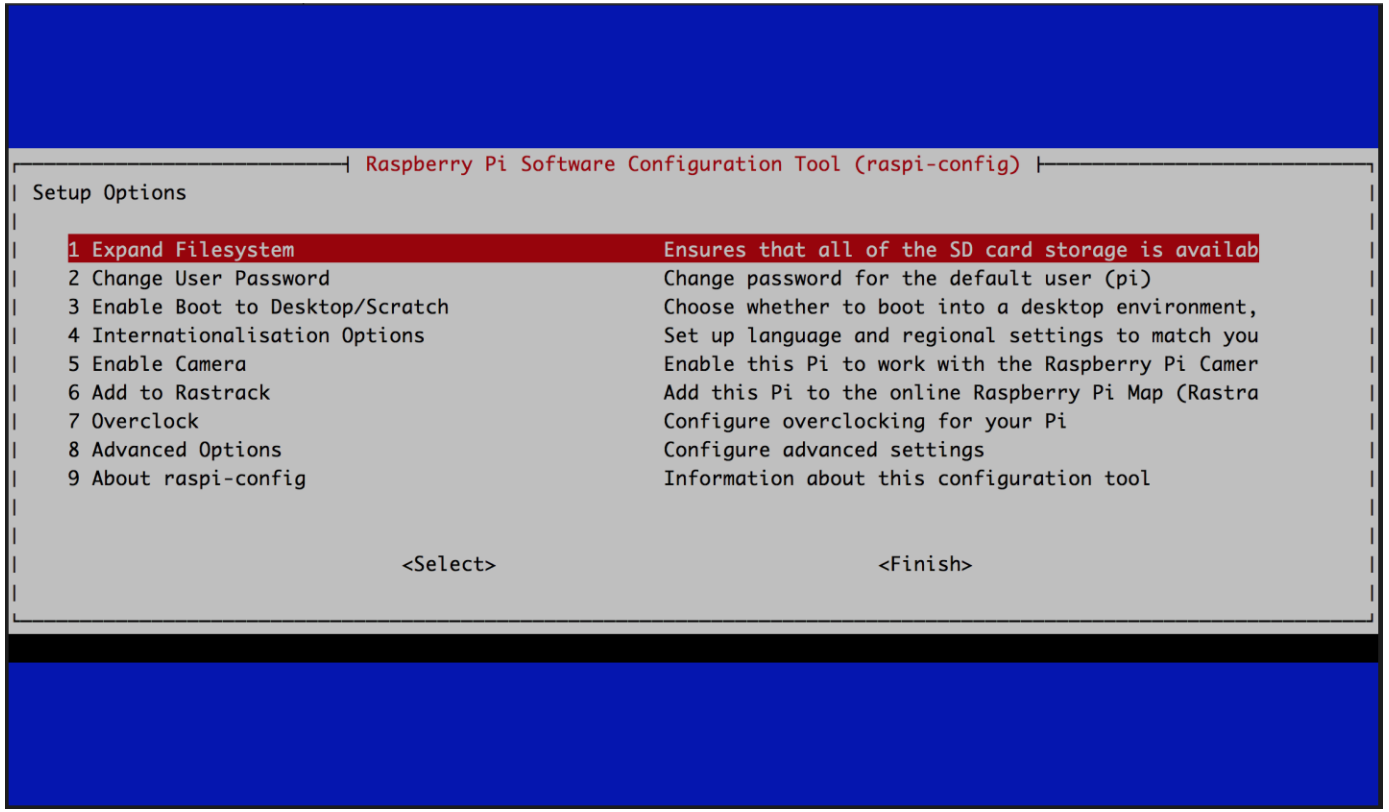
- Operačný systém a súborový systém je uložený na (micro)**SD karte**
  - SSD -> obmedzený počet zápisov
  - RPi nemá hodiny reálneho času (RTC) -> strata času po vypnutí
  - default login: pi/raspberry
- 1. krok: príprava SD karty s operačným systémom



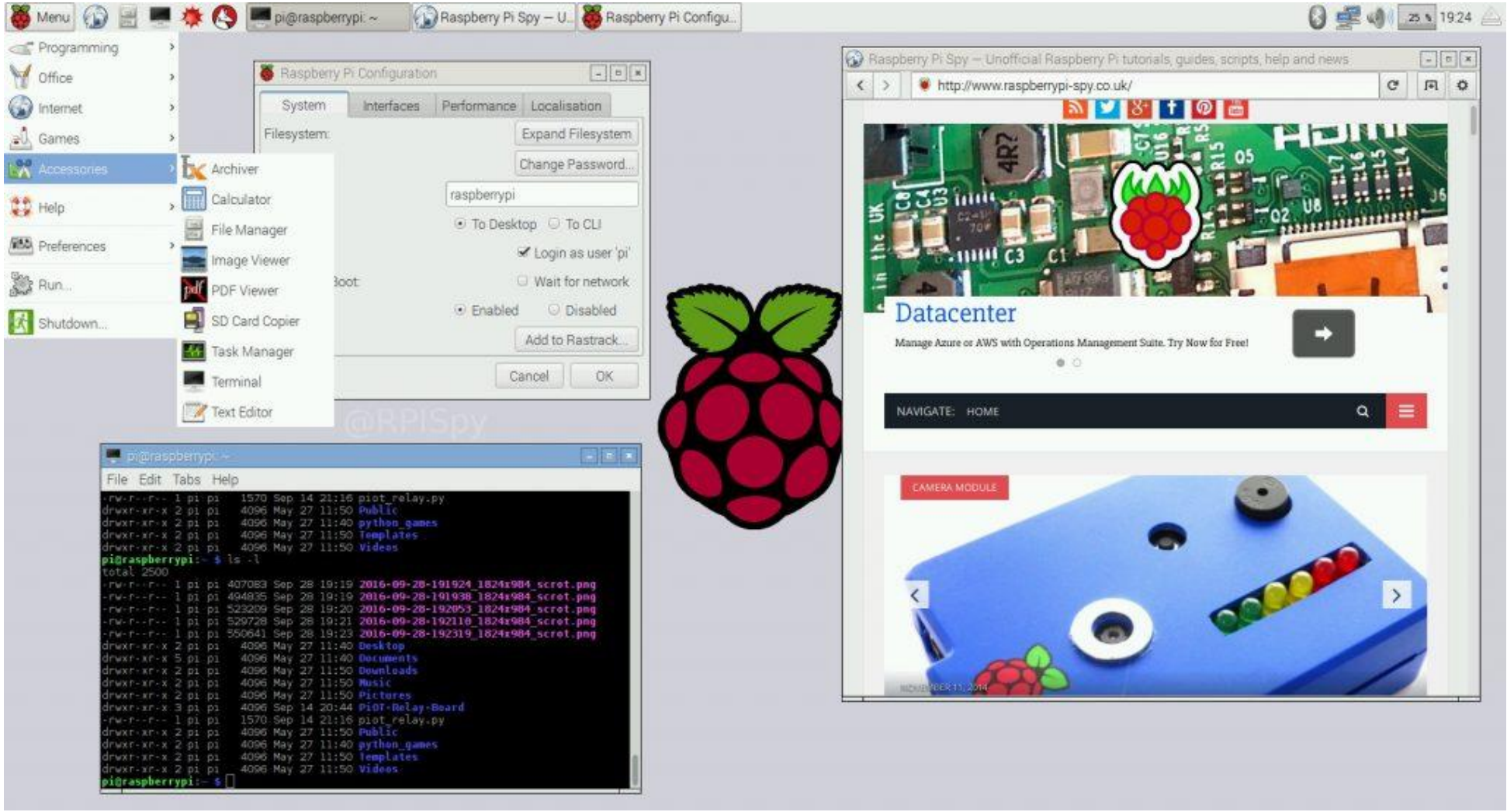


sudo raspi-config

- Nastavenie systému cez `raspi-config`
  - Aktivácia/deaktivácia SSH, I2C, SPI, kamery, …
  - Bootovacie nastavenia
  - Pretaktovanie
  - …



Raspbian – grafické používateľské rozhranie



Raspbian = Linux

- Raspbian a ďalšie OS pre RPi sú **plnohodnotné operačné systémy** a štandardné **linuxové systémy**:
  - `ifconfig -a`
    - konfigurácia siete
  - `/etc/wpa_supplicant/wpa_supplicant.conf`
    - konfigurácia WiFi
  - `lsusb`
    - vypísanie pripojených USB zariadení
  - `/dev`
    - pripojené zariadenia
  - `udev`
    - správca zariadení pre Linux kernel
    - `/etc/udev/rules.d/` - možnosť statického mapovania zariadení

GPIO na RPi

Alternate Function						Alternate Function
	3.3V PWR	1	2	5V PWR		
I2C1 SDA	GPIO 2	3	4	5V PWR		
I2C1 SCL	GPIO 3	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	GPIO 10	19	20	GND		
SPI0 MISO	GPIO 9	21	22	GPIO 25		
SPI0 SCLK	GPIO 11	23	24	GPIO 8	SPI0 CS0	
	GND	25	26	GPIO 7	SPI0 CS1	
	Reserved	27	28	Reserved		
	GPIO 5	29	30	GND		
	GPIO 6	31	32	GPIO 12		
	GPIO 13	33	34	GND		
SPI1 MISO	GPIO 19	35	36	GPIO 16	SPI1 CS0	
	GPIO 26	37	38	GPIO 20	SPI1 MOSI	
	GND	39	40	GPIO 21	SPI1 SCLK	

GPIO: 3.3 V

vs.
Arduino 5V

- žiadny ADC
- žiadne analógové piny
- zabudované pull-up a pull-down rezistory
- 1 pin s hardvérovým PWM

WiringPi

- **WiringPi** „is a PIN based GPIO access library written in C for the BCM2835, BCM2836 and BCM2837 SoC devices used in all Raspberry Pi.“
  - C knižnica aj utilita gpio
- WiringPi utilita **gpio**:
  - `gpio readall`
  - `gpio [-g] mode <pin> in/out/pwm/up/down/tri`
    - prepínač -g piny v číslovaní BCM_GPIO
  - `gpio [-g] write <pin> 0/1`
  - `gpio [-g] read <pin>`
  - `gpio export <BCM_pin> in/out`
    → /sys/class/gpio

```
#!/bin/bash
while true
do
    cat /sys/class/gpio/gpio23/value
done
```

i2c tools

- Utility na prácu s I²C „z príkazového riadku“
- RPi má dve I²C zbernice:
  - 0 – interné použitie
  - 1 – všeobecné použitie
  - Dostupné zbernice: `ls /dev/*i2c*`
  - SDA, SCL piny majú stále 1.8 kΩ pull-up rezistory
    → tieto piny sú nevhodné na všeobecné použitie

- `i2cdetect -y 1`
- `i2cget -y 1 0x11`
- `i2cset -y 1 0x11 0x01`

Pi4J (pi4j.com)

- Knižnica na prácu z GPIO, I^2C, SPI a sériovou komunikáciou priamo z Javy
- Maven: `com.pi4j.pi4j-distribution 1.1`

- Poznámky:
  - nekompatibilná s novšími kernelmi (staticky pribuildovaná staršia verzia WiringPi)
    - riešenie: 1.2-SNAPSHOT
    - fix:
```
static {
    System.setProperty("pi4j.linking", "dynamic");
}
```
  - NetBeans IDE podporuje spúštanie a debugovanie kódu na vzdialených platformách

Pi4J – práca s GPIO pinmi

```
package sk.upjs.iot;

import com.pi4j.io.gpio.*;

public class PinTest {

    static {
        System.setProperty("pi4j.linking", "dynamic");
    }

    public static void main(String[] args) throws Exception {
        GpioController gpio = GpioFactory.getInstance();
        GpioPinDigitalInput pin = gpio.provisionDigitalInputPin(
            RaspiPin.GPIO_04, PinPullResistance.PULL_UP);
        while (true) {
            System.out.println(pin.getState());
            Thread.sleep(500);
        }
    }
}
```

Pi4J – komunikácia cez I^2C

```
package sk.upjs.iot;

import com.pi4j.io.i2c.*;

public class I2CTest {

    static {
        System.setProperty("pi4j.linking", "dynamic");
    }

    public static void main(String[] args) throws Exception {
        I2CBus i2c = I2CFactory.getInstance(I2CBus.BUS_1);
        I2CDevice device = i2c.getDevice(0x38);
        while (true) {
            System.out.println(device.read());
            Thread.sleep(500);
        }
    }
}
```

Ďakujem za pozornosť