**ML01\_RPI-instalace\_OS-wiringPi**

**Téma:** Architektura mikropočítače RPI a připojení periférií. Instalace OS Raspbian s provedením základní konfigurace a ověřením funkcí GPIO. Vývojové prostředí Geany a jeho nastavení, tvorba jednoduchého programu v jazyce C. Konfigurace funkčních uzlů v prostředí NodeRED a propojení pomocí diagramu toků (flow) podle dílčí úlohy.

**Cíl hodiny:** Umět připojit k RPI napájení a periférie. Nainstalovat na micro SD kartu OS Raspbian a provést základní systémová nastavení. Připojit přes nepájivé pole LED diodu a příkazy z terminálu ovládat GPIO piny pro výstup na LED diodu. Nainstalovat prostředí NodeRED a vytvořit v něm dílčí úlohu pro blikání LED.

**Doba trvání:** 4x45min

**Pomůcky:** Rasberry Pi 3, PC se SW pro nahrání obrazu OS na microSD kartu, nepájivé pole, LED dioda, rezistor 220 Ω, propojovací vodiče.

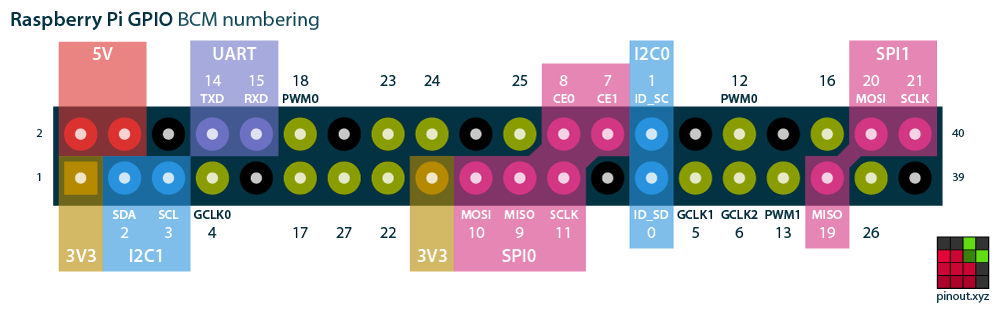
**Zdroje informací:** <https://pinout.xyz/pinout/wiringpi> , <http://wiringpi.com/>

1. **Rozhraní Raspberry Pi 3 model B**

RPI používá rozhraní USB, HDMI, Ethernet, Wi-Fi, Bluetooth a je napájeno pomocí micro USB konektoru. Speciálním portem pro připojení grafického dotykového displeje je port DSI (Display Serial Interface) a port CSI (Camera Serial Interface) pro připojení kamerového modulu. Další popis je soustředěn na rozhraní typické pro RPI – GPIO.

**GPIO**

Jedná se dvouřadý 40-ti pinový konektor GPIO, který umožňuje připojení, jak samostatných vstup/výstupních digitálních pinů, tak standardizovaných sběrnic, např. UART, SPI, I2C, 1-WIRE. Ve výchozím nastavení jsou vypnuté a dají se zapnout jak v grafickém prostředí, tak i v příkazovém řádku. Piny GPIO pracují s 3,3V logikou a na 40-ti pin konektoru rozhraní jsou kromě signálových pinů piny napájecí (3,3V, 5V) a zemnící. Poněvadž jsou signálové piny vyvedeny přímo na čip, je nutná velká obezřetnost a dodržení logické úrovně 3,3V.



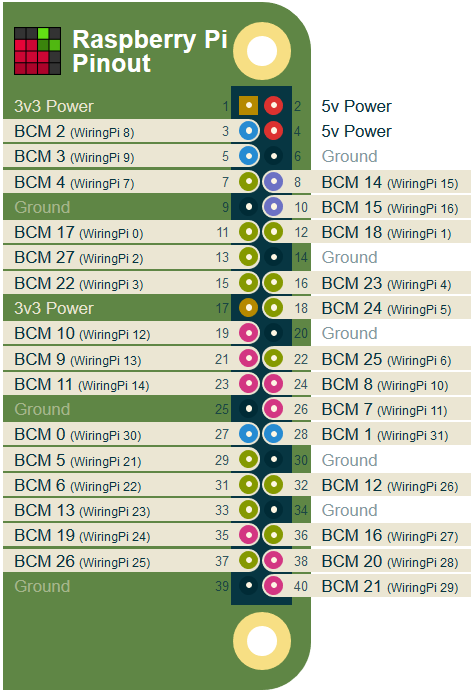
Standardizované sběrnice na RPI 3

**WiringPi**

Vytváří společnou platformu a soubor funkcí pro přístup k jednotlivým pinům pro více jazyků, např. jazyk C, Python. WiringPi používá vlastní schéma číslování pinů odlišné jak od fyzických pozic pinů, tak i od číslování GPIO pinů (BCM). V jazyce C se určí způsob číslování pinů příkazy:

int wiringPiSetup (void) ; // číslování WiringPi

int wiringPiSetupGpio (void) ; //číslování GPIO (BCM)



Číslování pinů WiringPi

1. **Operační systém pro Raspberry Pi 3 model B**

Pro RPI existuje více druhů operačních systémů, ale nejpoužívanější univerzální operační systém je Raspbian. Většinou se stažený obraz operačního systému nahraje na SD micro kartu, ze které RPI bootuje.

**Instalace OS Raspbian na Raspberry Pi**

Žák si stáhne obraz \*.img nebo \*.zip OS Raspbian with desktop (pro práci v grafickém prostředí). Pomocí nástroje pro zápis obrazu na micro SD kartu (Rufus, Etcher, atd.) vytvoří bootovací paměťovou kartu s nahraným obrazem. Při této operaci může dojít na školním počítači k problému s právy (jsou vyžadována práva administrátora).

**OS Raspbian**

Raspbian je bezplatný operační systém založený na Debianu optimalizovaném pro hardware RPI. Raspbian poskytuje více než čistý operační systém a dodává se s více než 35 000 balíky, předkompilovaným softwarem dodávaným ve formátu pro snadnou instalaci na RPI. Součástí současné desktopové verze jsou kromě kancelářského balíčku také speciální vývojové nástroje:

* BlueJ Java IDE – vývojové prostředí pro programovací jazyk Java
* Geany Programmers Editor – vývojové prostředí pro programovací jazyk C, C++
* Node-Red - programovací nástroj založený na bázi flow-based programming (FBP).
* Greenfoot Java IDE - vývojové prostředí pro programovací jazyk Java pracující s grafickými objekty
* Mathematica – matematický a vědecký softwarový balíček
* Python 2 (IDLE) nebo Python 3 (IDLE) - vývojové prostředí pro programovací jazyk Python
* Scratch – grafické programovací prostředí
* Sense HAT Emulator – emulace rozšiřujícího modulu pro RPI
* Sonic Pi – hudební vývojové prostředí
* Wolfram (LXTerminal) – terminál programovacího jazyka Wolfram pro matematické a vědecké výpočty

**Seznámení s Rasberry GPIO Breadboard Shield**

Pro vyvedení GPIO pinů z RPI je možné použít plochý vícežilový kabel se 40-ti pinovým konektorem, který se připojí na Breadboard Shield zapojený do nepájivého pole. Tato zakončovací destička má přehledný popis pinů. Neobsahuje však žádnou elektronickou ochranu.



GPIO Breadboard Shield

**Nastavení OS Raspbian**

* Žák připojí RPI do LAN sítě IoT podle situace na metalickou síť nebo k WiFi.
* Login do OS Raspbian má jméno „pi“ a heslo „raspberry“.

Pro snazší práci se soubory si žáci v terminálu nainstalují midnight commander (mc) příkazem *apt-get install mc* a provedou update a následně upgrade OS (Nejlépe pomocí příkazu *sudo apt update && sudo apt upgrade –y*. S parametrem –y se už nemusí potvrzovat instalace). Tato instalace trvá dlouho, je vhodné využít čas k zapojení nepájivého pole k vyzkoušení ovládání GPIO pinů z terminálu.

**Vyzkoušení ovládání GPIO pinů pomocí příkazů z terminálu**

Pro vyzkoušení ovládání GPIO pinů na Raspberry Pi je zapotřebí připojit přes nepájivé pole LED diodu např. na wiringpi pin 21, kterou budou žáci rozsvěcovat a zhasínat pomocí příkazů:

gpio mode 21 output // nastavení wiringpi pinu 21 na výstup

gpio write 21 1 // 1 - rozsvícení LED diody

gpio write 21 0 // 0 - zhasnutí LED diody

Obsah obrázku obvod, hodiny

Popis byl vytvořen automaticky

Zapojení LED diody na wiringpi pin 21

**Záloha operačního systému**

Pozn.: tato operace vyžaduje více času a není bezpodmínečně nutná ke splnění dílčí úlohy.

Do USB čtečky karet vložíme stejně velkou mikro SD kartu, na kterou chceme zálohovat. Jména disků (karet) zjistíme pomocí příkazu:

*~ $ lsblk*

Kopii provedeme příkazem (pokud je cílová karta /dev/sdb) s parametrem výpisu postupu:

~ $ sudo dd bs=4M if=/dev/mmcblk0 of=/dev/sdb status=progress conv=fsync

**Vzdálený přístup přes USB RNDIS**

V případě použití Raspberry Zero nelze přistupovat k RPI přes periferní zařízení (klávesnice, myš, případně monitor), můžeme k RPI přistupovat pomocí síťového rozhraní Kindle USB RNDIS. Získá tak IP adresu a přistoupí k RPI přes konzoli SSH, nebo graficky přes VNC viewer. K tomu je potřeba mít tyto utility nainstalované včetně ovladače Kindle USB RNDIS pro daný operační systém (např. ro WIN10 kindle\_rndis.inf).

Pokud je nutno instalovat ovladač na PC jako nepodepsaný, zadejte s právy admin do konzoly PC příkazy pro odblokování instalace nepodepsaných ovladačů:

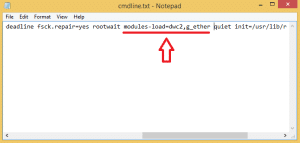
*bcdedit /set testsigning on*

*bcdedit /set loadoptions DISABLE\_INTEGRITY\_CHECKS*

*bcdedit.exe /set NOINTEGRITYCHECKS ON*

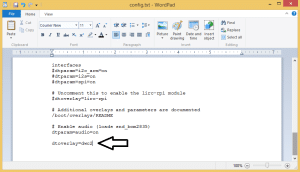
V samotném RPI je nutno nastavit konfigurační soubory v oblasti boot. Soubor cmdline.txt má záznam v jediném řádku, na konec tohoto řádku přidejte za rootwait:

modules-load=dwc2,g\_ether



Do souboru config.txt přidat za poslední řádek:

dtoverlay=dwc2



IP adresu RPI připojeného přes USB RNDIS je možné nastavit na RPI staticky v souboru etc/network/interfaces:

*allow-hotplug usb0*

*iface usb0 inet static*

*address 192.168.7.2*

*netmask 255.255.255.0*

*network 192.168.7.0*

*broadcast 192.168.7.255*

*allow-hotplug wlan0*

*iface wlan0 inet dhcp*

*wpa-conf /etc/wpa\_supplicant/wpa\_supplicant.conf*

aby byla přístupná nadále Wifi, je nutno zadat v souboru /etc/wpa\_supplicant/wpa\_supplicant :

*ctrl\_interface=DIR=/var/run/wpa\_supplicant GROUP=netdev*

*update\_config=1*

*network={*

*ssid="myNetworkName"*

*psk="myNetworkPassword"*

*key\_mgmt=WPA-PSK*

*}*

1. **Seznámení s vývojovým prostředím Geanny**

Geanny je programovací prostředí v anglickém jazyce již nainstalované spolu s jinými aplikacemi v OS Raspbian. Žáci v něm mohou tvořit programy v jazyce C. Při ukládání je potřeba použít koncovku zdrojového souboru jazyka C (názevsouboru.c).

Pro tvorbu programů je vhodné využít knihovnu vzorových příkladů WiringPi, kterou nainstalujeme pomocí příkazů:

$ git clone git://git.drogon.net/wiringPi

$ cd ~/wiringPi

$ git pull origin

Po této instalaci je možné ovládat GPIO piny podle číslování GPIO (BCM), např. pro LED diodu na pinu wiringPi 21 platí pin GPIO 5 a příkazy se zadávají s parametrem –g:

gpio –g mode 5 output // nastavení GPIO pinu 5 na výstup

gpio –g write 5 1 // 1 - rozsvícení LED diody

gpio –g write 5 0 // 0 - zhasnutí LED diody

**Základní lišta programu:**

File (Soubor) – základní operace s novým projektem: nový, otevřít, uložit a vlastnosti programu, ukončení programu.

Edit (Úpravy) – možnosti kopírování, vkládání kódu, přesun na řádek, najít, možnosti odsazení.

Search (Vyhledávání) – možnost nají určitý text v souboru, jít na určitý řádek

View (Zobrazení) – nastavení zobrazení lišt a oken v programu

Document (Dokument) – nastavení souboru: programovací jazyk, kódování, znaky pro konce řádků, odstavců

Project (Projekt) – otevření nového nebo již napsaného kódu v souboru (projektu)

Build (Sestavení) – nastavení parametrů kompilace a sestavení, způsob spuštění kompilace a sestavení kódu

Tools (Nástroje) – výchozí nastavení, manažér pluginů

Help (Nápověda)

Ve volbě Build/Set Build Commands nastavit způsob kompilace programu a způsob vytvoření spustitelného souboru pro doinstalovanou knihovnu wiringPi:

Compile: gcc -Wall –c „%f“ –l wiringPi

Build: gcc -Wall –o „%e“ „%f“ –l wiringPi

Obsah obrázku snímek obrazovky

Popis byl vytvořen automaticky

Nastavení kompilace ve vývojovém prostředí Geanny

Příklad vytvoření programu pro blikání LED

Inspirací může být program blink.c v adresáři wiringPi/examples ve kterém upravíme číslo wiringPi GPIO pinu. Zdrojový program se zkompiluje klávesou F8 a sestaví (případně včetně kompilace) klávesou F9. Výpisy z překladu a kompilace se zobrazí v terminálu, kde je možné výsledný program spustit (ve stejné složce jako byl vytvořen) příkazem například: ./blink.

1. **Seznámení s prostředím NodeRED**

Node-Red je vývojové grafické prostředí, které je předinstalované v OS Raspbian. Pokud by instalace chyběla, je možné ji provést příkazem, příkaz provede rovněž update:

*~ $* bash <(curl -sL https://raw.githubusercontent.com/node-red/raspbian-deb-package/master/resources/update-nodejs-and-nodered)

Node-Red je možné spustit z hlavního menu nebo z terminálu příkazem node-red-start. Pokud se má NodeRED nastartovat po zapnutí RPI, je nutné toto nastavení zadat příkazem:

*~ $ sudo systemctl enable nodered.service*

Konfigurace se provádí v grafickém prostředí v prohlížeči na IP adrese, kde Node-Red běží (přidělená IP adresa RPI z AP – 192.168.100.100 nebo na RPI localhost) s nastavením portu 1880. Node-Red umožňuje sestavení diagramu toků dat z vývojových funkčních bloků propojených mezi sebou:

* Input – inject, mqtt klient, …
* Output – debug, mqtt broker, …
* Function – delay, trigger, switch, …
* Social – email, twitter, …
* Storage – file, …
* Analysis
* Advanced – watch, exec, …
* Raspberry Pi – rpi gpio, …

Jednotlivé funkční bloky se dají po dvojitém kliknutí konfigurovat. U vstupů je potřeba zadat data, která se mají posílat, jakého datového typu budou a název zprávy topic, který je stejný jako u klienta na druhé straně komunikace. U výstupu je nejjednodušší použít funkční blok msg. payload, pomocí kterého se budou data vypisovat vpravo na liště v záložce debug.

Vstupní a výstupní hodnoty uzlů pro sestavený diagram úlohy je možné zobrazit v prohlížeči pomocí rozhraní dashboard na stejné adrese a portu jako diagram toků s přidáním /ui. V diagramu toku je nutné přidat uzly pro paletu dashboard, které je nutné doinstalovat vyhledáním node-red-dashboard:

Obsah obrázku snímek obrazovky, interiér, monitor, počítač

Popis byl vytvořen automaticky

1. **Výsledný diagram dílčí úlohy**

Obsah obrázku snímek obrazovky

Popis byl vytvořen automaticky

Dashboard Blink:

Obsah obrázku snímek obrazovky

Popis byl vytvořen automaticky

1. **Vypracování PL01\_RPI-instalace\_OS-wiringPi**

Při úspěšném zpracování pracovního listu by žáci měli zvládnout:

* Nainstalovat OS Raspbian, nakonfigurovat (včetně připojení na internet) a provést update a upgrade OS.
* Propojit LED diodu přes nepájivé pole s Raspberry Pi.
* Příkazem v konzole ovládat GPIO výstup na LED.
* Aktualizovat prostředí NodeRED
* Sestavit blokový diagram, kde tlačítko ovládá diodu
* Graficky zobrazit tlačítko v dashboardu