

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

Vyšší odborná škola a Střední průmyslová škola, Jičín, Pod Koželuhy 100

18-20-M/01 Informační technologie

IQRF výukový simulační modul

Individuální praktická maturitní práce

Jičín, 2024/2025

David Horáček



Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

Obor vzdělávání: Informační technologie

Číslo oboru: 18-20-M/01

Třída: I4B

Školní rok: 2024-2025

Téma maturitní práce: IQRF výukový simulační modul

Vedoucí maturitní práce: Bc. Jiří Klíma

Oponent maturitní práce: Ing. Ladislav Vik

Téma zpracuje žák: David Horáček

Zadání maturitní práce:

Maturitní práce se zaměřuje na rozvoj IoT sítí prostřednictvím naučné desky, která simuluje venkovní prostory ulice s využitím bezdrátových technologií. Deska bude zahrnovat technologii IQRF a využití sonických senzorů, relé a led diod. Žák bude využívat automatizační vývojové prostředí Node-RED, běžícím na zařízení Raspberry Pi. K projektu bude také přiložen podrobný manuál, který uživatelům poskytne veškeré potřebné informace pro instalaci, konfiguraci a provoz systému.

Hlavní cíle práce:

- Vytvoření naučné desky s pomocným manuálem pro vývoj automatizačního systému.
- Simulace bezdrátové technologie IQRF pro specifické příklady.
- Vytvoření dashboardů ovládání a sběr informací v prostředí Node-RED.
- Implementace Raspberry Pi do systému.
- Vytvoření manuálu pro instalaci a ovládání panelu.

Způsob zpracování a kritéria hodnocení jsou uvedena v dokumentu Metodický návod na tvorbu individuální maturitní práce, který je umístěn v týmu individuálních maturit prostřednictvím MS Teams.

Termín zadání práce 11. 10. 2024

Řešitel práce předloží vedoucímu práce svoje řešení k posouzení (elektronicky prostřednictvím Microsoft Teams):

- konzultace (úvodní řešerše, návrh koncepčního řešení): 1. 11. 2024
- konzultace (první průběžné řešení): 29. 11. 2024
- konzultace (druhé průběžné řešení): 17. 12. 2024
- konzultace (třetí průběžné řešení): 17. 1. 2025
- konzultace (návrh konečného řešení s návrhem prezentace): 28. 2. 2025

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

Odevzdání práce:

**17. 3. 2025 elektronicky přes MS Teams 18. 3. 2025 do 12.00 hod.
odevzdat jeden výtisk IMP v kroužkové vazbě**

Vypracovanou práci včetně všech příloh odevzdá žák elektronicky prostřednictvím aplikace Microsoft Teams, dále pak jednu vytištěnou písemnou práci předsedům předmětových komisí strojní a elektro, případně vedoucímu. Vytištěné práce slouží jako doklad pro případnou kontrolu dosaženého stupně vzdělání a zůstávají v majetku školy. Jsou uloženy v archivu a v knihovně školy, přístupné žákům školy ke studiu.

.....
Podpis studenta

.....
Podpis vedoucího práce

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

Prohlašuji, že jsem individuální praktickou maturitní práci zpracoval(a) samostatně a uvedl(a) v ní všechny prameny, literaturu a ostatní zdroje, které jsem použil(a).

V Jičíně dne

Podpis žáka:.....

Anotace

Tato práce se zabývá konceptem chytrého města. Byla vytvořena výuková deska, která simuluje venkovní prostředí ulice a ukazuje, jak efektivně využít IoT ke snížení infrastruktury. Město je monitorováno bezdrátovou technologií IQRF. Pro vizualizaci získaných informací bylo použito vývojové prostředí Node-RED. Zařízení Raspberry Pi slouží jako pracovní stanice celého systému. Jako součást práce byly zformovány metodické listy, které slouží ke snadnému pochopení a implementaci IoT řešení.

Klíčová slova

IoT, IQRF, Node-RED, Raspberry Pi, Didaktická pomůcka

Annotation

This thesis deals with concepts of smart city. A teaching board has been created that simulated an outdoor street environment and shows how to effectively use IoT to reduce infrastructure. The city is monitored by wireless IQRF technology. The Node-RED development environment was used to visualize the obtained information. Device Raspberry Pi serves as workstation for the entire system. Part of the work are method sheets, which serves for easy understanding and implementation of IoT solution.

Key words

- IoT, IQRF, Node-RED, Raspberry Pi, Didactic tool

Obsah

Úvod.....	9
1 Teoretická část.....	10
1.1 Internet věcí.....	10
1.2 IQRF.....	10
1.2.1 IQMESH	11
1.2.2 Transciever	11
1.2.3 IQRF OS	12
1.2.4 DPA framework (V4.31)	12
1.2.5 Bonding proces	14
1.2.6 IQRF IDE	14
1.2.7 Bezpečnost	14
1.2.8 Další zařízení.....	14
1.2.9 IQRF SDK.....	15
1.3 Komunikační protokoly	15
1.3.1 SPI.....	15
1.3.2 UART	16
1.4 Node-RED.....	16
1.4.1 Využití/Praktické řešení	16
1.4.2 Architektura.....	16
1.4.3 Funkce a možnosti.....	17
1.4.4 Bezpečnost	17
1.4.5 Node-RED a umělá inteligence.....	17
1.5 Raspberyy Pi	17
1.5.1 Hardware Raspberry Pi 4	17
1.5.2 Software	18
1.5.3 Možnosti využití	18
2 Praktická část	19
2.1 Výstavba IQRF sítě.....	19
2.1.1 Úvod do projektu	19

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

2.1.2	Nastavení transceiveru	20
2.1.3	Vlastnosti chování TR.....	21
2.1.4	Vytvoření IQMESH sítě	22
2.1.5	Custom DPA Handler	23
2.1.6	Senzor měření vzdálenosti – Custom DPA Handler.....	24
2.2	Raspberry Pi	27
2.2.1	Plánovaný postup práce/očekávání/problémy.....	27
2.2.2	Postup řešení	29
2.2.3	Příprava Rpi	29
2.2.4	Konfigurace RPI.....	30
2.2.5	MQTT	31
2.2.6	IQRF Gateway Daemon.....	31
2.2.7	IQRF Gateway Daemon WebApp.....	32
2.2.8	Node-RED	34
2.2.9	Tvorba flows a dashboardu	36
2.2.10	Tvorba Node-RED pro sonický senzor.....	40
2.3	Návrh desky	43
2.3.1	3D tisk.....	44
2.3.2	Skládání finálního modelu	44
2.3.3	Displej zapojení.....	45
Závěr		46
Zdroje a literatura.....		47
Seznam příloh.....		50

Obrázek 1 - Popis všech vlastností IQMESH [5]	11
Obrázek 2 - Model transceiveru se schématickými popisky	12
Obrázek 3 - Fyzické modely IQRF s periferií	19
Obrázek 4 - Chyba při nahrání plug-inu.....	20
Obrázek 5 - Zjištění OS TR modulu.....	20
Obrázek 6 - Porovnání starších DPA a nových.....	21
Obrázek 7 - Připojený TR modul do CK-USB-04A.....	21

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

Obrázek 8 - Náhled na IQMESH Network manager.....	22
Obrázek 9 - Map View sítě.....	23
Obrázek 10 - Makra Iot Stater KITu.....	24
Obrázek 11 - Zapojený sonický senzor k modulu DK-EVAL-04a.....	24
Obrázek 12 - Přidané soubory v hlavičce.....	25
Obrázek 13 - Inicializace pinů.....	26
Obrázek 14 - Zpracování periferie	26
Obrázek 15 - Zdržení pomocí operace nop	26
Obrázek 16 - Čtení hodnoty z pinu, výpočet vzdálenosti.....	27
Obrázek 17 - Software balenaEtcher.....	29
Obrázek 18 - Funkce GPIO pinů [23]	30
Obrázek 19 - Vložení KON-RASP-01 [11].....	30
Obrázek 20 - Výběr repositáře shodný s distribucí [26].....	32
Obrázek 21 - Nastavení SPI interface	33
Obrázek 22 - Služba Daemon ve webovém rozhraní	33
Obrázek 23 - Sekce zařízení v Network manager	34
Obrázek 24 - Node-RED dashboard s daty ze sítě	35
Obrázek 25 – Node-RED IQRf overview.....	36
Obrázek 26 – Node-RED Connection MQTT.....	37
Obrázek 27 – Node-RED IQRf DEMO.....	39
Obrázek 28 - Ukázkový příklad z dokumentace IQRf[27].....	41
Obrázek 29 - Označené uzly, které byly přidány pro HC-SR04	42
Obrázek 30 - Přidaná část do záložky IQRf DEMO	42
Obrázek 31 - Parsování dat pro dashboard.....	42
Obrázek 32 - Ukázka mType možností	43
Obrázek 33 - Výsledek zobrazený v Node-RED.....	43
Obrázek 34 - K40 whisper návrh řešení	44
Obrázek 35 - Finální testování displeje se světly	45

ÚVOD

IoT dnes představuje neodmyslitelnou součást našeho života. Jednou z těchto technologií je IQRf, bezdrátová síťová platforma navržená pro nízkoenergetickou a spolehlivou komunikaci mezi zařízeními. Klíčovou roli hraje právě v konceptu Internetu věcí (IoT), který umožňuje monitorování, automatizaci různých prvků či správu městského prostředí.

Tato práce se zaměřuje na návrh a realizaci výukového simulačního modulu, který demonstruje principy chytré infrastruktury města s využitím IQRf. Hlavním cílem je vytvoření modelu IoT sítě, který umožní studentům lépe porozumět fungování těchto technologií. Součástí práce je také dokumentace a metodické listy, které slouží k usnadnění výuky a praktické aplikace získaných znalostí.

Struktura práce je rozdělena na dvě části. Nejprve je popsán teoretický základ, který zahrnuje podstatné informace a principy IoT sítě a IQRf. Na tyto znalosti pak navazuje praktická část, která obsahuje konfigurace, testování a dosažený výsledek.

1 TEORETICKÁ ČÁST

V této kapitole jsou představeny teoretické základy. Podrobně jsou vysvětleny všechny použité technologie, které jsou nutné k vzájemnému propojení IoT sítí.

1.1 Internet věcí

Internet věcí (anglicky Internet of Things, zkráceně IoT) je síť fyzických zařízení, vozidel, objektů a spotřebičů, které jsou vybaveny senzory, softwarem a síťovou konektivitou. [1]

Historie IoT ve zkratce vychází z myšlenky označit všechny objekty identifikátorem a připojit je do internetu a tím je spravovat. Koncept IoT byl poprvé formulován v roce 1999. Termín „Internet of Things“ byl užít Kevinem Ashtonem, který pracoval pro společnost Procter & Gamble, kde jeho hlavní mise byla zautomatizovat a zefektivnit práci. RFID čipy byly použity k monitorování zboží. Každopádně to nebylo zcela první IoT řešení. V roce 1982 na Univerzitě Carnegie Mellon bylo úkolem zjistit, zda má cenu chodit k automatu na Coca-Colu. [2] Automat byl připojen do místní sítě a byly přidány senzory. Byly poskytovány informace o stavu zásob a teplotě nápojů v reálném čase. Od té doby byl Internet věcí využíván téměř ve všech odvětvích. Průmysl, energetika, zdravotnictví, doprava, logistika, zemědělství, chytrá města či domácnosti jsou jen ukázkou, že toto téma změnilo celý svět.

Rozmach IoT dokazují i čísla. Podle statistik je aktuálně přes 18 miliard těchto zařízení a podle předpovědi toto číslo brzy přesáhne 30 miliard v následujících letech. [3] Dnešní doba ukazuje, jak mocné je spojení IoT a cloudu. Data se posílají sama do cloudu, kde se ukládají, zpracovávají a mohou být snadno použita na řešení jiných problémů. V rozvoji odvětví Big Data přichází teprve plná síla využitelnosti chytrých zařízení. Učení AI (umělá inteligence) obvyklých či neobvyklých jevů, reagování na změnu, informování na stavy, které možná nastanou, bude směrem, kudy IoT půjde v dalších letech. [1]

Internet věcí je zařízení, které má senzory. Pomocí nich měří různé hodnoty. V síti komunikuje pomocí síťových protokolů a pro přenos dat využívá datových protokolů. Data se posílají do systému či cloudu, kde se vyhodnocují a dále zpracovávají.

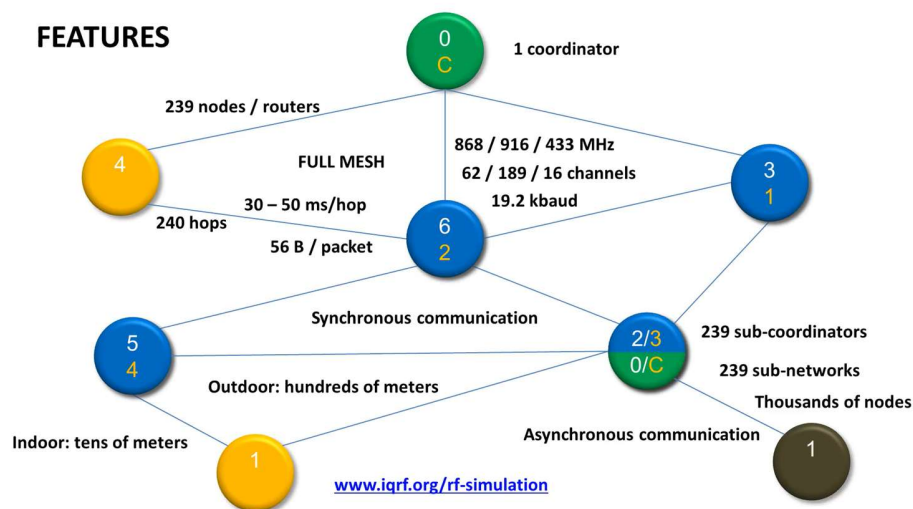
1.2 IQRF

V rámci Internetu věcí hraje technologie IQRF (Intelligent Quality Radio Frequency) významnou roli hned v několika oblastech. Jde o technologii vyvinutou českou společností MICRORISC a je chráněna několika patenty. IQRF je platforma určená pro bezdrátovou komunikaci s nízkou spotřebou energie. Přenos dat probíhá v pásmu sub-GHz (433 MHz, 868 MHz a pro USA 916 MHz) a jde o bezlicenční pásma, tudíž nevyžaduje poplatky za přenos ani jinou infrastrukturu externích operátorů.

Pro propojení se využívá tzv. mesh topologie. V případě IQRF se jedná o protokol IQMESH. Komunikaci zajišťují speciální transceivery. Díky spojení těchto transcieverů s mesh sítí může dosah přesahovat až stovky metrů. Sít' je pak řízena koordinátorem, který řídí další nody (uzly). Pro větší sítě či cloudová řešení musí být v síti i gateway. [4]

1.2.1 IQMESH

Směrovací protokol se nazývá IQMESH. Jde o robustní a spolehlivý komunikační protokol, který byl vyvinutý přímo pro IQRF. Byl navrženou tak, aby umožnil komunikaci velkému počtu zařízení v rámci jedné sítě. V jedné IQRF síti může být až 240 transcieverů. IQMESH tedy vytváří mesh sít', tzn. každý uzel je připojen na všechny další, co má v dosahu. Je založen na optimalizovaném zaplavení (node hopping), proto každý node může fungovat jako přenosový bod, čímž zajišťuje několik cest, kudy mohou data jít, pokud nějaký node selže. IQMESH je zároveň flexibilní protokol a dokáže se automaticky přizpůsobit změnám v síti.

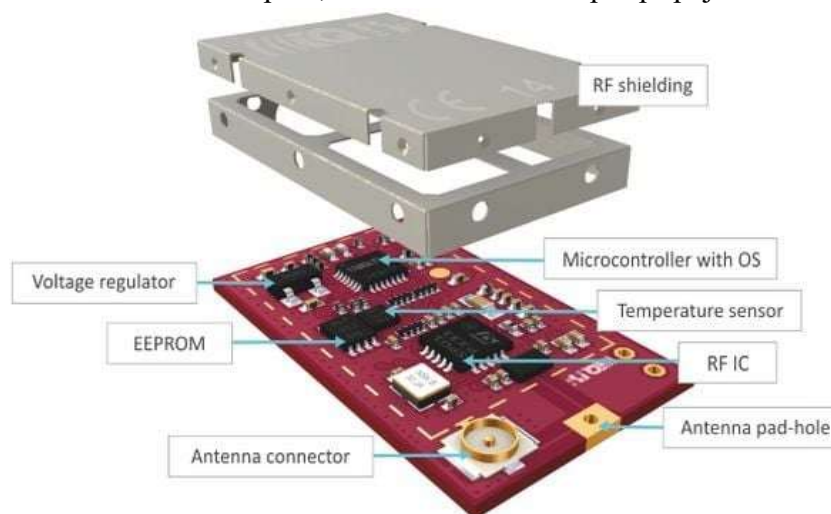


Obrázek 1 - Popis všech vlastností IQMESH [5]

1.2.2 Transciever

Transciever je jedno z klíčových zařízení v IQRF technologii. Zajišťuje obousměrnou bezdrátovou komunikaci. Kombinuje funkce vysílače a přijímače. Jsou to malé plošné spoje o velikosti SIM karty a umožňují přijímat i vysílat radiové vlny. Mezi nejdůležitější části patří řídicí mikrokontroler s OS, externí EEPROM paměť, senzor teploty, červená a

zelená LED a několik dalších pinů, které mohou sloužit pro připojení dalších zařízení.



Obrázek 2 - Model transceiveru se schématickými popisky

1.2.3 IQRF OS

IQRF vyvinula pro běh aplikací svůj operační systém. Je určen pro řízení komunikace bezdrátových sítí. Obsahuje sadu předdefinovaných funkcí a umožňuje vývoj aplikací v jazyce C. IQRF OS také umožňuje řízení pomocí DPA protokolu. Tento protokol zjednodušuje ovládání jednotlivých zařízení bez větších znalostí jazyka C nebo síťových znalostí. Operační systém má na starost také plno dalších funkcí, jako je správa mesh sítě, automatická oprava, řízení spotřeby energie nebo bezpečnost přenosu dat. Zároveň jde o kompatibilní systém, tudíž usnadňuje integraci do různých řešení. [6] [7]

1.2.4 DPA framework (V4.31)

Direct Peripheral Access (dále jen DPA) je součástí technologie IQRF a tvoří most mezi hardwarem, komunikací a aplikační logikou. Jedná se o komunikační protokol, který zjednodušuje vývoj či správu IQMESH sítě. Komunikace probíhá přes komunikační protokol SPI nebo UART. Uživatel je schopen prostřednictvím jednoduchých příkazů ovládat předdefinovanou periferii. IQRF obsahuje sadu funkcí s účelem zrychlit a zjednodušit práci. Pokud uživatel potřebuje vlastní funkcionalitu, může použít Custom DPA Handler. [8]

Struktura zprávy se skládá z 4 povinných parametrů.

- NADR

„Node Address“ je základem směrování pro IQMESH. NADR je 2-bajtový parametr pro síťové adresování (tzn. 16-bitový). Přesto lze využít pouze 1 nižší bajt. Vyšší bajt není podporovaný. Adresy mají své významy:

0x00 = IQMESH Coordinator

0x01 – 0xEF = IQMESH Node Addresses (1-239)

0xFF = Broadcast Address (zpráva pro všechny nody v síti)

0xFE = Temporary Address (dočasná adresa, např. při bondování)

- PNUM

„Peripheral Number“ je jednobitový parametr, který určuje, jaká periferie je adresována. Podle hodnot jsou rozděleny do různých skupin:

0x00 – 0x1F = vestavěné periferie

0x20 – 0x3E = pro uživatelské periferie

0x40 – 0x7F – rezervováno IQRF

Další nejsou přístupné.

- PCMD

„Peripheral Command“ je jednobitový parametr, který pracuje společně s PNUM a definuje, jakou operaci provést s danou periferií. Bit 7 je jeden z nejdůležitějších, označuje odpověď.

- HWPID

„Hardware Profile ID“ je dvoubitový parametr, který specifikuje funkcionalitu a chování zařízení. Určuje, jaká uživatelská periferie je implementována. Slouží pro ověření, kontrolu a bezpečnost. Předchází spuštění neplatných příkazů nebo nekompatibilních zařízení.

- Volitelně PData

„Peripheral Data“ je volitelné pole s proměnlivou délkou (max. 56bytes). Nese přidaná data příkazů nebo odpovědí. Používá se pro čtení z paměti a periferií. Na základě své délky ovlivňuje čas prostupnosti sítě.

DPA zprávy se dále dělí na DPA Request, DPA Confirmation, DPA Notification, DPA Response.

FRC – Fast Response Command je vložená periferie v IQRF, která umožňuje rychlý sběr dat z několika nodů v síti. Hlavním cílem je rychlost vybírání dat stejného typu pomocí jednoho požadavku. PNUM příkazu FRC je 0x0D.

1.2.5 Bonding proces

Aby mohly prvky tvořit síť, je nutné je nejprve propojit. Tento proces se nazývá bonding a je zásadní pro fungování mesh sítě. Při bondingu je nový node přidáván do sítě řízené koordinátorem, který následně umožňuje jeho správu a ovládání prostřednictvím bezdrátové komunikace. Bonding lze provést několika způsoby:

- Local bonding je mechanismus, kdy je node přidán na základě požadavku a komunikace s koordinátorem.
- Smart bonding je mechanismus, kdy je node přidán ručně za pomoci zadání IBK a MID.
- Remote bonding je složitější mechanismus, kdy je node předbondován a následně přidán do sítě.

1.2.6 IQRF IDE

IQRF IDE je další z ekosystému IQRF. Jedná se o mocný softwarový program, který je navržen pro efektivní vývoj, správu, programování, testování IQRF aplikací. Software nabízí mnoho funkcí, usnadňuje tak práci vývojářům. Nabízí také správu IQMESH sítě, využití DPA protokolů a monitorování zařízení. IQRF IDE nabízí user-friendly prostředí. Je vhodné pro začátečníky i pokročilé.

1.2.7 Bezpečnost

Bezpečnost jde ruku v ruce s operačním systémem. S novým OS přichází také vylepšené zabezpečení. Protože jde znovu o klíčový aspekt, obzvlášť pokud jde o bezdrátovou komunikaci, proto IQRF implementuje hned několik vrstev ochrany. Rozděleny jsou do tří bezpečnostních prvků (přístupové, síťové, uživatelské šifrování) založených na standardu AES-128. [9] Tento algoritmus zajišťuje, aby byla všechna data v síti šifrována. Při přidávání nových zařízení do sítě probíhá bonding. Jde o způsob bezpečného párování. Při bondování musí být zadána citlivá data jako je NID, heslo nebo síťová adresa, proto jsou přenášena přístupovým šifrováním. Kontrolní součet nebo časový tag lze považovat za další ochranné opatření.

1.2.8 Další zařízení

IQRF disponuje širokou škálou vývojových prostředků. Při vývoji se transceiver nemusí pájet k jinému plošnému spoji, ale stačí ho pouze zasunout do konektorů SIM. Níže budou představeny vývojové prostředky, které budou využity při této práci.

CK-USB-04a

Tento modul je vývojový a programovací prostředek určený pro práci s transceiverem zasunutým v SIM konektoru. Je vybaven malým mikrokontrolerem, který umožňuje připojení k počítači přes USB. Není vybaven akumulátorem, proto je napájen přes USB port počítače. CK-USB-04a obsahuje dvě tlačítka. První slouží pro

reset transceiveru a druhé pro uživatelské použití. Může fungovat jako gateway propojující síť s PC.

DK-EVAL-04a

Tento modul je možné připojit přes piny na další periferie nebo rozšiřující vybavení. DK-EVAL-04a není vybaven mikrokontrolerem, takže ho není možno připojit k PC. Má akumulátor, který lze nabíjet. Obsahuje totožná tlačítka s CK-USB-04a. Pro zapnutí tohoto modulu je nutné připojit spínač, tzv. jumper. [10]

DDC-SE-01

Při připojení na modul DK-EVAL-04a se nabízí využití další senzorů. Na tomto modulu lze nalézt teplotní senzor, fotorezistor a potenciometr. DDC-SE-01 podporuje připojení dalšího zařízení přes sběrnici I2C.

DDC-RE-01

Tento modul může mít široké praktické využití. Je vybaven dvěma relé, která jsou schopna ovládat spotřebiče do napětí 24 V a do výkonu 12 W.

KON-RASP-01

Modul (Shield) je vyvinutý pro použití s Raspberry Pi s klíčovými vlastnostmi. Lze vložit IQRF transceiver. Modul je připojený skrz GPIO piny. Napájený skrz Raspberry Pi. Pro připojení musí být Raspberry Pi vypnuto, to platí i pro připojení transceiveru. [11]

1.2.9 IQRF SDK

IQRF SDK (Software Development Kit) je platforma navržená pro jednoduchý vývoj softwarových aplikací pro zařízení s IQRF transceivery. Podporuje programovací jazyky jako Java nebo C (++) a je integrována do vývoje na zařízeních, jako je Raspberry Pi. SDK poskytuje knihovny, které usnadňují komunikaci se zařízeními IQRF přes SPI nebo GPIO.

1.3 Komunikační protokoly

1.3.1 SPI

Serial Peripheral Interface je sériový periferní komunikační protokol používaný k propojení mikrokontrolérů s periferními zařízeními. SPI je synchronní protokol. Pro komunikaci je využíván hodinový signál, který zajišťuje synchronizaci mezi zařízeními. Rozděluje 2 zařízení – Master X Slave. Master generuje signál a řídí komunikaci. Slave odpovídá na požadavky. Ke komunikaci využívá 4 hlavní vodiče. Výhodou je rychlost, full-

důležitá komunikace a jednoduchost. Pokud je používáno větší množství zařízení, nastane problém s počtem pinů. [12]

1.3.2 UART

Universal Asynchronous Receiver-Transmitter je sběrnice sloužící pro asynchronní sériový přenos dat mezi zařízeními. Jde o široce využívaný modul v mikrokontrolerech, počítačích a dalších elektronických zařízeních. Umožňuje jednoduchou výměnu dat přes sériové rozhraní bez potřeby synchronního hodinového signálu. Komunikace probíhá pomocí dvou hlavních vodičů. Tx (Transmitter) je vodič, kterým zařízení odesílá data. RX (Receive) je vodič, kterým zařízení přijímá data. Komunikace nevyžaduje sdílený hodinový signál, ale ta to musí být předem nastavena stejná přenosová rychlost (baud rate) kvůli správné synchronizaci dat. Každý datový rámec obsahuje Start bit, Datové bity, Paritní bit, Stop bit. Mezi výhody patří jednoduchost, spolehlivost díky paritnímu bitu nebo nízký počet vodičů. Na druhou stranu jde o pomalejší protokol, který vyžaduje nastavení přenosové rychlosti. [13]

1.4 Node-RED

Node-RED je velmi výkonný open-source nástroj, který usnadňuje integraci různých systémů, technologií a služeb bez nutnosti velkých znalostí programování. Propojuje hardware, API a online služby, čímž vytváří mnoho zajímavých a nových řešení. Za tímhle účelem byl vytvořen společností IBM Emerging Technology Services a aktuálně je jednou z částí OpenJS Foundation. Programování probíhá přímo v prohlížečovém rozhraní (browser-based) pomocí drag-and-drop. [14] To dělá jednoduché zapojení mnoha uzlů pohromadě. Jde o stále vyvíjený software a díky podpoře komunity dochází k neustálému vývoji.

1.4.1 Využití/Praktické řešení

V IoT světě je Node-RED skvělý nástroj pro vývoj aplikací. Poskytuje snadnou správu, analýzu a automatizaci dat v reálném čase klidně od několika IoT zařízení najednou. Podporuje mnoho běžných protokolů (MQTT, HTTP) pro přenos dat mezi zařízeními a cloudem. Uživatelům umožňuje vizualizovat data a zobrazovat je pomocí dashboardů. V rámci automatizace lze nastavit mnoho pravidel, jak se má zařízení chovat v různých situacích a podmínkách. Data je také možné ukládat do databáze, nebo odesílat přímo pro analýzu. Je ideální pro použití v domácnostech, průmyslu, zemědělství, tak i pro chytrá města.

1.4.2 Architektura

Node-RED je postaven na platformě Node.js. Ten je zodpovědný za zpracování událostí. Je to asynchronní běhové prostředí pro JavaScript. Node-RED právě díky tomu zvládne pracovat s událostmi a asynchronními procesy. Node.js pokrývá serverovou část, která běží nezávisle na klientovi a zajišťuje stabilitu a rychlou komunikaci. GUI je webová aplikace

(editor), kde vše jde řešit vizuálně. Pomocí jednotlivých nodů (uzly – funkční blok) lze tvořit flows (toky – řetězec, který má nějaký výsledek), které pak mají logický význam. Všechny nody, jejich propojení a data se ukládají ve formátu JSON.

1.4.3 Funkce a možnosti

Tento nástroj nabízí přizpůsobení různým požadavkům pomocí rozmanitých funkcí, knihoven a balíčků. Nody jsou základními stavebními prvky. Jsou to pomyslné bloky kódu, které mají danou funkci. Flows jsou uspořádané sekvence kódů, které mají výsledek. Node-RED obsahuje standartní knihovnu, která má mnoho běžných funkcí. Základní vstupní a výstupní nody nebo pro debug. Pokud paleta balíčků nestačí, pak je k dispozici npm (Node Package Manager), kde snadno package nainstalovat. Například Dashboard je jednou z nich.

1.4.4 Bezpečnost

Node-RED nabízí řadu bezpečnostních funkcí. Při přihlášení do editoru může být nakonfigurováno přihlášení a vyžadování hesla pro přístup k tokům. RBAC je vlastnost, kterou Node-RED získal až o něco déle. Vytvoří více úrovní přístupových práv. To znamená, že některým uživatelům lze přidělit pouze omezený přístup. Přenos dat mezi klientem a serverem zajišťují protokoly TLS/SSL. [15] Dalším prvkem by mělo být správné naložení s daty. Jelikož se data ukládají ve formátu JSON, což je obvykle uloženo v souborovém systému, tak jsou snadno přístupná. Pro vyšší zabezpečení lze data ukládat do zabezpečené databáze.

1.4.5 Node-RED a umělá inteligence

Už v této době lze zaznamenat umělou inteligenci v Node-RED. Je možné pracovat s balíčky pro AI, které mají predikci, rozpoznávání, rozpoznávání obrazu nebo umí rozpoznat něco neobvyklého. Pokud potřebujeme naučit AI konkrétní situace, jsou zde předtrénované modely. TensorFlows.js může být dotrénován, jelikož používá neuronové sítě. Cloudy v dnešní době určitě využívají AI a lze implementovat umělou inteligenci do Node-RED skrz cloudové řešení.

1.5 Rasperry Pi

Rasperry Pi je malý výkonný jednodeskový počítač, který se využívá často ve vývojářských projektech, vzdělávání nebo IoT aplikací. Díky své ceně je velmi dostupný. První vznikl roku 2012 s myšlenkou rozšíření znalostí o počítačové technice. Díky vlastnostem, kterými disponuje, se hodil do mnoha průmyslových řešení.

1.5.1 Hardware Rasperry Pi 4

Hardwarová vrstva poskytuje veškerou základní výbavu pro samostatné fungování. Základem je CPU - ARM Cortex-A72 s frekvencí až 1,5 GHz. RAM paměť je dostupná ve velikostech 2 GB, 4 GB a 8 GB (v projektu využíváno 4 GB). Úložiště využívá microSD,

protože nemá vlastní pevný disk. Na microSD je uložen OS a data. Pro připojení další periférie slouží 4 x USB port (2.0 a 3.0), 2 x HDMI port, Ethernet, Wifi, Bluetooth, a A/V port. Nejvýznamnější částí jsou GPIO (General Purpose Input/Output) piny, které umožňují komunikaci přímo s hardwarovými komponenty. Raspberry Pi je osazeno 40 piny. Piny mají své funkce (power pins, General purpose – digitální piny, se speciální funkcí – SPI, UART I2C). [16]

1.5.2 Software

Na Raspberry lze nahrát mnoho operačních systémů. Oficiální se však nazývá Raspberry Pi OS a dříve byl také nazýván Raspbian. Ten je založen na Linuxové distribuci Debian. Debian je jednou z nejstarších Linux distribucí a je zkompileován pro procesory ARM. Pokud jde o průmyslové použití v IoT a není potřeba Linux, existuje také operační systém Windows IoT Core, který je proto určen.

1.5.3 Možnosti využití

Raspberry Pi zvládá operace od primitivních až po složité. Proto je vhodný do mnoha projektů od domácího použití po řešení projektů s umělou inteligencí. Často slouží jako server nebo řídicí prvek sítě, který dokáže ovládat ostatní zařízení. V případě práce o IQRF bude výukový simulační modul představovat právě řídicí prvek.

2 PRAKTICKÁ ČÁST

V této části se popisuje realizace projektu založeného na IQRF technologii. Jsou zde vysvětleny jednotlivé kroky vedoucí k dokončení práce, včetně problémů, které se během řešení objevují, a jejich vysvětlení. Pro lepší přehlednost se celý proces dělí do několika pod částí.

2.1 Výstavba IQRF sítě

Nejprve byla připravena všechna zařízení určená pro sběr dat a řízení v rámci jedné sítě. Pro tento účel bylo využito čtyř nodů a jednoho koordinátoru. Počátečním krokem bylo zprovoznění sítě, přičemž každý prvek byl nakonfigurován pro správnou komunikaci v rámci IQMESH technologie.



Obrázek 3 - Fyzické modely IQRF s periférií

2.1.1 Úvod do projektu

Konfigurace sítě je realizována prostřednictvím softwaru IQRF IDE, který zajišťuje její kompletní správu a ovládání. Nejprve je nezbytné stáhnout vývojové prostředí z oficiálních stránek IQRF.org, následně rozbalit instalační balíček a provést jeho instalaci.

Při vytváření nového projektu je k dispozici několik možností konfigurace. Vzhledem k tomu, že základním prvkem sítě je mesh topologie, je vhodné využít předpřipravený projekt „IQMESH demo“, který obsahuje veškeré potřebné soubory a usnadňuje konfiguraci sítě.

V této práci byl využit předpřipravený projekt ze starter kitu IoT-StarterKit-01-demo, který obsahuje předdefinované knihovny a makra určená pro řízení sítě prostřednictvím DPA

příkazů. Tato makra výrazně zjednodušují správu a ovládání sítě, což umožňuje efektivnější práci s IQMESH technologií.

Na úvod je nutné zdůraznit, že vytvoření IQMESH sítě představuje komplexní proces, který vyžaduje předem připravené nody pro zajištění správné funkčnosti a stability sítě. Níže je uveden postup, podle kterého byla konfigurace sítě realizována.

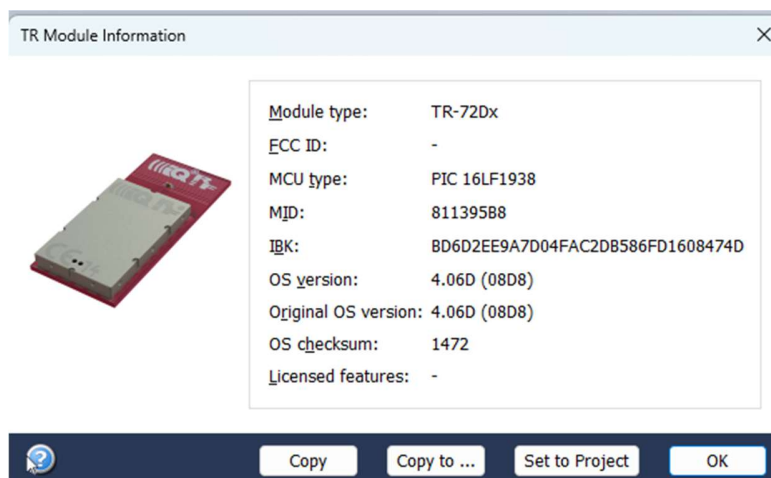
2.1.2 Nastavení transceiveru

IQRF platforma se neustále vyvíjí, a proto je nutné před konfigurací ověřit kompatibilitu zařízení a verze operačního systému, který jím bude nahrán. Plug-iny (předpřipravené hardwarové profily – HWP) určené pro nody a koordinátor, které jsou součástí ukázkového projektu, nejsou kompatibilní s novějšími verzemi OS.

V současnosti je aktuální verze transceiverového OS 4.06D, zatímco hardwarové profily obsažené v projektu jsou kompatibilní pouze s verzemi 4.00D a 4.02D. Z tohoto důvodu dochází při pokusu o nahrání plug-inu do transceiveru k chybě. Tento problém lze snadno vyřešit, existují dva jednoduché způsoby: 1 - Downgrade OS na požadovanou verzi pomocí nástroje Change IQRF OS Wizard. 2 - Stažení a použití nových kompatibilních plug-inů, odpovídajících aktuální verzi OS.



Obrázek 4 - Chyba při nahrání plug-inu

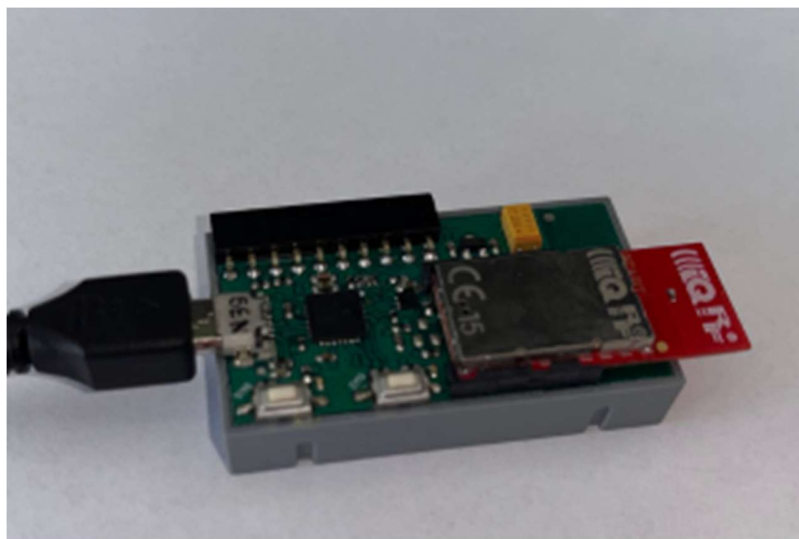


Obrázek 5 - Zjištění OS TR modulu



Obrázek 6 - Porovnání starších DPA a nových

Nahrání HWP (i jiné konfigurace) TR modulů probíhá přes USB rozhraní, které je připojeno k programátoru CK-USB-04A. Vložení TR modulu se provádí za současného stisknutí reset tlačítka a zasunutí do SIM konektoru, aby nedošlo k poškození zařízení. Zelená LED v levém rohu v IDE bliká v případě, že zařízení je rozpoznáno a komunikuje s PC.



Obrázek 7 - Připojený TR modul do CK-USB-04A

Soubor se do transceiverů nahraje kliknutím pravého tlačítka a výběru upload (také stisknutím klávesy F10). Na nody v síti se nahraje plug-in s názvem „DPA-Node-STD-7xD-V417-220224“, který řekne transceiveru, aby se choval jako nod. Pro koordinátor je to plug-in „DPA-Coordinator-STD-7xD-V417-220224“.

2.1.3 Vlastnosti chování TR

Konfigurace transceiverů ovlivňuje zásadně jejich chování a výdrž. V části projektu „TR Configuration“ jsou soubory s příponou .xml, což je strukturovaný datový soubor obsahující informace o nastavení. Parametry frekvence komunikace (868 MHz) a kanálu (52) se ponechávají na výchozí úrovni. Pro všechny TR toto musí být stejné, aby došlo ke komunikaci mezi jednotlivými zařízeními. Zabezpečení je další nezbytnou podmínkou pro IQMESH, a proto je nastavováno heslo. Dále nastavení DPA:

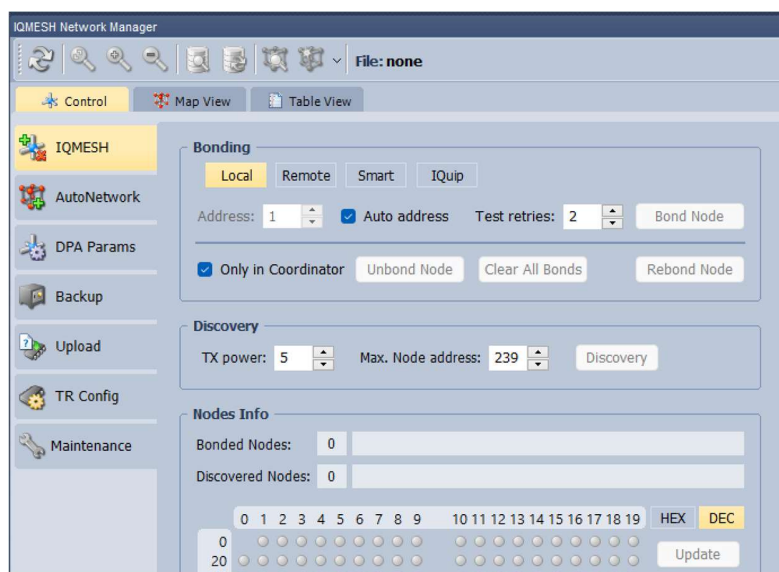
- Node bez připojené periferie má nejjednodušší konfiguraci.

- Node s připojenou periferií vyžaduje nastavení custom DPA Handler.
- Koncový node může mít vypnuté routování (routing off) – není nutné.
- Koordinátor zajišťuje funkci FRC pro rychlý sběr dat a má povolenou komunikaci přes SPI.

2.1.4 Vytvoření IQMESH sítě

Pokud byla před tím zařízení součástí jiné sítě, je potřeba všechny předchozí vazby mezi nody a koordinátorem zrušit. V koordinátoru se to dělá pomocí funkce Clear All Bonds, která je v části IQMESH Network Manager. U nodů je potřeba podržet obě tlačítka naráz, poté pustit reset tlačítko (horní) a počkat, dokud nezhasne rozsvícená zelená LED. Jakmile zhasne, druhé tlačítko musí být puštěno. Úspěšné provedení signalizuje blikající červená LED na nodu.

Bondování může proběhnout různými cestami. V této práci je využito lokálního bondování.

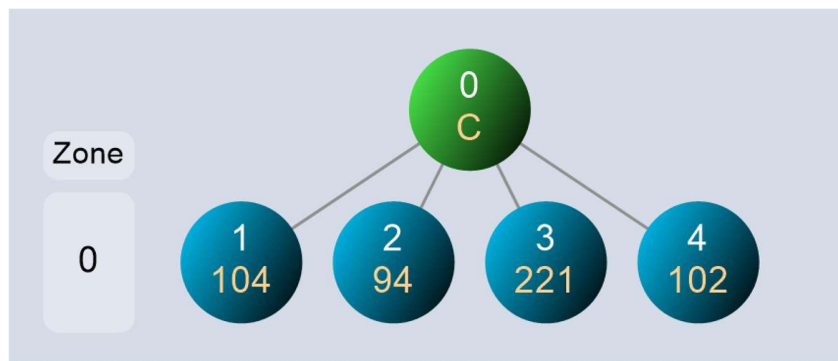


Obrázek 8 - Náhled na IQMESH Network manager

Bondování – proces, při kterém node získává:

- logickou adresu,
- network ID,
- síťové heslo,
- konfiguraci komunikačního kanálu od koordinátora.

V průběhu procesu Discovery se mapují nody a přiřazují se jim VRN (Virtual Routing Number). Tímto krokem se také vytváří směrovací tabulka. Přejitím do záložky „map view“ lze vidět náhled sítě.



Obrázek 9 - Map View sítě

Tímto procesem je IQMESH připravená. Správně nastavenou síť lze ověřit pomocí DPA příkazu k rozsvícení červené LED diody na kterémkoliv nodu.

2.1.5 Custom DPA Handler

Poslední důležitou částí jsou zdrojové kódy. Jde o kompilované programy napsané v jazyce C pro mikrokontrolery. Nazývají se „Custom DPA Handler“ a umožňují přizpůsobit chování transceiverů. Mnoho příkladů už je vytvořeno právě od společnosti MICRORISC a lze je využít. Soubor musí být před nahráním zkompilovaný do formátu .hex.

V této výukové pomůcce je pracováno s periferními zařízeními, pro které už jsou Custom DPA Handler napsány, tak i s periferií, pro kterou bylo nutné napsat vlastní zdrojový kód.

Pro modul DDC-RE-01, který obsahuje relé, lze využít „0C02_BinaryOutput-Template“, který pracuje s relé. Díky starter-kitu jsou zde i připravená makra, pomocí kterých lze tato relé ovládat. Kód .c je tedy potřeba kompilovat a poté ho uploadovat do TR.



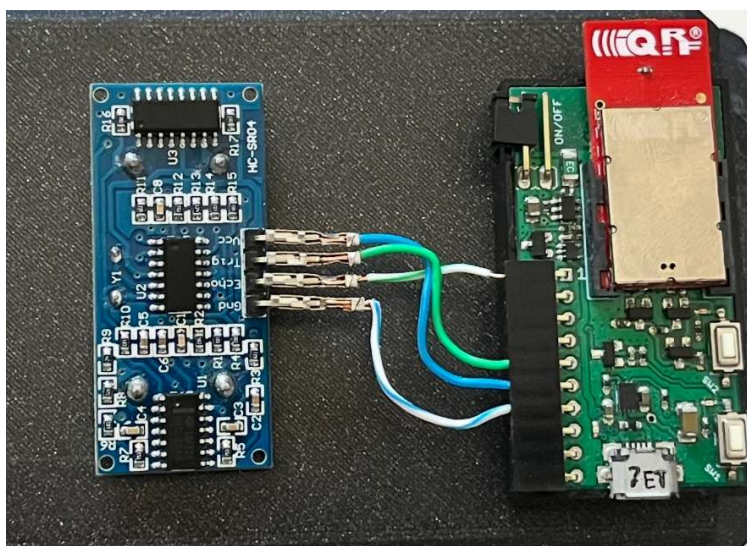
Obrázek 10 - Makra Iot Starter KITu

DDC-SE-01 je druhý modul, který má především senzory. Byl zde využit stejný postup, pouze jiný handler „0002_DDC-SE01“.

2.1.6 Senzor měření vzdálenosti – Custom DPA Handler

Zapojení ultrazvukového senzoru HC-SR04 probíhá přímým propojením s modulem DK-EVAL-04A. Tento senzor slouží jako ultrazvukový měřič vzdálenosti a umožňuje přesné měření vzdálenosti od objektů v definovaném rozsahu.

Senzor disponuje 4 piny - VCC, TRIG, ECHO a GND. VCC je připojeno na 5 V, což odpovídá pinu Vbat na modulu DK-EVAL-04A, GND je připojeno na ground (také pin GND), TRIG pin (Výstup) je zapojen do RA0 – což odpovídá pinu C1 na modulu. Echo je zapojen do RC5 (C8). Pro ovládání tohoto senzoru je použitý custom DPA Handler napsaný v jazyce C. [17]



Obrázek 11 - Zapojený sonický senzor k modulu DK-EVAL-04a

Rozbor C kódu

Kód je rozdělen na více částí, které zajišťují použití DPA Frameworku, který umožňuje ovládat IQRF zařízení. V hlavičce programu jsou přidány soubory nutné pro práci s DPA protokolem. Celý kód je uvnitř funkce CustomDpaHandler() s návratovým typem funkce bit (v běžném jazyce C se používá bool (C99 +), ale některé embedded systémy používají stále tento typ pro úsporu paměti).

```
#include "IQRF.h"  
#include "DPA.h"  
#include "DPAcustomHandler.h"
```

Obrázek 12 - Přidané soubory v hlavičce

Funkce clrwdt() je v kódu proto, aby zajistila, že mikrokontroler nebude resetován, pokud běží správně. Tato funkce resetuje watchdog timer (WDT) v mikrokontroléru. Jde o bezpečnostní mechanismus, který resetuje procesor, pokud nastane nějaká chyba. Běží neustále na pozadí.

Přepínání mezi DPA událostmi – funkce GetDpaEvent() zjišťuje, jaká událost byla vyvolána. Možné události:

- DpaEvent_Interrupt – Je to metoda, která značí hardwarové přerušení, které předává CARRY jako flag (příznak), tj. speciální bit, který se používá pro určité operace – zde buď bude 1 (TRUE) nebo 0 (FALSE).
- DpaEventInit – Inicializace/nastavení pinů.
- DpaEvent_Idle – Tato událost je volána, pokud zařízení je v klidném stavu nečinnosti (idle), mohlo by být využito k úspoře energie nebo například k monitorování stavu.
- DpaEvent_DpaRequest – Zde je hlavní logika programu.

Inicializace pinů probíhá přes registry LAT a TRIS, ty jsou součástí mikrokontroléru, který je uvnitř IQRF modulů (TR-72). TRIS určuje pinu, zda bude vstupem nebo výstupem. $TRISx.y = 0$; znamená, že se jedná o výstup (1 by značila vstup). LAT (Latch) nastavuje výstupní logickou hodnotu – 0 = LOW, 1 = HIGH. Pokud se LAT nastaví jako vstup (TRIS = 1), nemá žádnou funkci. Tudiž funguje pouze jako výstupní pin. Pro práci s piny je určený i PORT - čte hodnotu ze vstupního pinu. $PORTx.y == 1 \{ \}$. [18]

```
case DpaEvent_Init:
    LATA.0 = 0;
    TRISA.0 = 0; // RA0 jako výstup (trig)
    TRISC.5 = 1; // RC5 jako vstup (echo)
    break;
```

Obrázek 13 - Inicializace pinů

Zpracování požadavků periferie je v části kódu, která obsahuje DPA požadavky související s periferií v zařízení. Nastaví se počet periferních zařízení, která budou připojená na zařízení (UserPerNr = 1). Je také přidána do seznamu periferií a je jí přiřazené ID (PNUM_USER + 0). Potřeba je nastavit identifikátor HWPID, který je určen pro identifikaci zařízení a HWPIDver pro verzi zařízení.

```
IfDpaEnumPeripherals_Else_PeripheralInfo_Else_PeripheralRequest()
{
    // We implement 1 user peripheral
    _DpaMessage.EnumPeripheralsAnswer.UserPerNr = 1;
    FlagUserPer(_DpaMessage.EnumPeripheralsAnswer.UserPer, PNUM_USER + 0);
    _DpaMessage.EnumPeripheralsAnswer.HWPID = 0x000F;
    _DpaMessage.EnumPeripheralsAnswer.HWPIDver = 0x1234;

DpaHandleReturnTRUE:
    return TRUE;
```

Obrázek 14 - Zpracování periferie

Měření samotné vzdálenosti proběhne přes zapnutí LATA.0 = 1. Pin musí být high po dobu 10 μ s, toho je docíleno pomocí nop() – trvá 1 μ s. Poté se čeká na začátek ECHO signálu. Přepočet vzdálenosti pak vychází z fyzikálních parametrů zvuku ve vzduchu (při 20° C (343 m/s)). [19]

```
LATA.0 = 1;
nop2(); nop2(); nop2(); nop2(); nop2();
nop2(); nop2(); nop2(); nop2(); nop2();
LATA.0 = 0;
```

Obrázek 15 - Zdržení pomocí operace nop

```
uns16 time = 0;
while(PORTC.5 == 0){} // Čekání na vzestupnou hranu
while(PORTC.5 == 1) { time++; } // Měření délky pulzu

uns16 distance = time / 26;

_DpaMessage.Response.PData[0] = (char)distance;
_DpaMessage.Response.PData[1] = (char)(distance / 256);
_DpDataLength = 2;
```

Obrázek 16 - Čtení hodnoty z pinu, výpočet vzdálenosti

Kód byl poskytnutý bývalým studentem Tomášem Weisssem.

Vysvětlení některých pojmů

Embedded systémy – zařízení, které obsahuje mikroprocesor, paměť a I/O, má omezené zdroje, má jednu dedikovanou funkci, je to často součástí nějakého většího zařízení

2.2 Raspberry Pi

2.2.1 Plánovaný postup práce/očekávání/problémy

Pokud síť funguje správně, pokračuje se v práci s Raspberry Pi. Postupováno bylo podle návodu od IQRF. [20]

Předpokládá se využití Raspberry Pi 4 společně s novým Raspberry Pi OS (64-bit Debian GNU/Linux 12 Bookworm). Pro začátek se provádí vypálení image na SD kartu pomocí Raspberry Pi Imageru, pokračuje se připojením periférií a základním nastavením systému.

Dále se stahují potřebné technologie pro zprostředkování a přenos dat (middleware, tedy software sloužící jako prostředník mezi aplikacemi). Provádí se konfigurace IQRF Gateway Daemon WebApp, instalace Node-RED a testování.

Během práce se objevily první problémy. Vyzkoušel se image přímo poskytnutý na IQRF, avšak při pokusu o použití Raspberry Pi 4 s Raspbian Lite (Raspberry Pi OS) se zobrazuje chybová hláška: "This board requires newer software." Firmware již není kompatibilní s Raspberry Pi 4.

Na kartu byl vypálen nejnovější OS pro Rpi. Problém nastal hned, když na Raspberry Pi byl nahrán nekompatibilní repositář, který nebyl určen pro specifickou distribuci. Konkrétně byl použit repositář pro Debian Buster (10) stable testing, zatímco běžící distribuce byla Debian Bookworm (12). Problémy byly identifikovány při kontrole, zda služba běží, přičemž několik procesů bylo hned ukončeno. Při odstraňování problémů se zjistila záměna distribucí. Pro zajištění čistého systému byla data vymazána nejjednodušším způsobem

přeformátováním karty, případně nevratným krokem pomocí příkazu `sudo rm -rf /home/pi/*`. Následně byla distribuce opravena na aktuální verzi dostupnou z oficiálních stránek <https://repos.iqrf.org/>.

Proces stažení byl opakován a provedeny kontrolní testy, které ukazovaly, že služba běží správně. Pokračovalo se ke stažení Gateway Daemon WebApp, což je webové rozhraní pro správu a řízení bezdrátové IQRF sítě. V tomto rozhraní se nastavuje rozhraní SPI pro komunikaci. [21] Po restartu služby by měla být IQRF síť přístupná skrze Network Configurator.

Zde nastal třetí problém – síť se znovu nedokázala správně načíst. Nebylo možné rozpoznat koordinátor ani načíst další zařízení. Hledání příčiny tohoto problému bylo náročné. Přezkoumání chybových hlášení ukázalo, že problém spočíval v neschopnosti načíst TR modul. Podle logů měl být problém v konfiguraci rozhraní. Hláška vypadala následovně:

Jan 12 19:26:26 raspberrypi iqrfgd2[2787]: Error: Cannot get TR parameters msg => interface to DPA coordinator is not working - verify (CDC or SPI or UART) configuration.

Bylo zkušeno povolení SPI jak prostřednictvím `raspi-config`, tak úpravou v konfiguračním souboru, nicméně problém přetrvával. [22] Po kontaktování společnosti MICRORISC bylo sděleno, že nový kernel 6.x momentálně není podporován (zjištění informace `uname -a`), a bylo doporučeno využít starší, kompatibilní verzi.

Zde jsou ukázány další chybové logy:

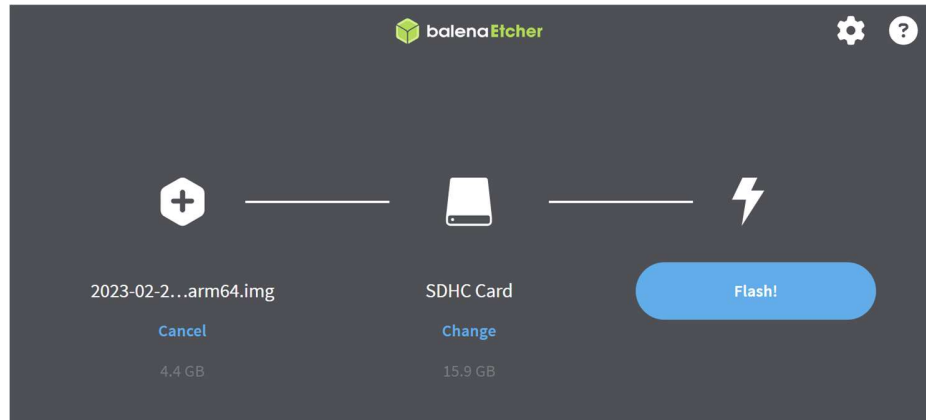
journalctl -u iqrfgd2.service -f

Jan 12 19:26:22 raspberrypi iqrfgd2[2787]: Error during opening file: No such file or directory
Jan 12 19:26:22 raspberrypi iqrfgd2[2787]: Error during opening file: No such file or directory
Jan 12 19:26:26 raspberrypi iqrfgd2[2787]: Error: Cannot get TR parameters msg => interface to DPA coordinator is not working - verify (CDC or SPI or UART) configuration
Jan 12 19:26:26 raspberrypi iqrfgd2[2787]: Error: Interface to DPA coordinator is not ready - verify (CDC or SPI or UART) configuration
Jan 12 19:26:26 raspberrypi iqrfgd2[2787]: [IQRF Repository cache] Cache is up to date.
Jan 12 19:26:26 raspberrypi iqrfgd2[2787]: [IQRF Repository cache] Loading cache ...
Jan 12 19:26:26 raspberrypi iqrfgd2[2787]: [IQRF Repository cache] Cache successfully loaded.
Jan 12 19:26:26 raspberrypi iqrfgd2[2787]: Cannot load required package for: os="0000" dpa="0000"
Jan 12 19:26:26 raspberrypi iqrfgd2[2787]: Loaded package for: os="08B1" dpa="0300"

2.2.2 Postup řešení

Při práci se postupuje podle doporučení a dokumentace MICRORISC. Pro realizaci se využívá obraz systému 2023-02-21-raspbian-bullseye-arm64.img.xz, který představuje verzi Linuxu 11 (Bullseye). Systém běží na jádru verze 5.15, což je kompatibilní verze kernelu.

Obraz systému se stahuje a vypaluje na SD kartu pomocí softwaru BalenaEtcher.



Obrázek 17 - Software balenaEtcher

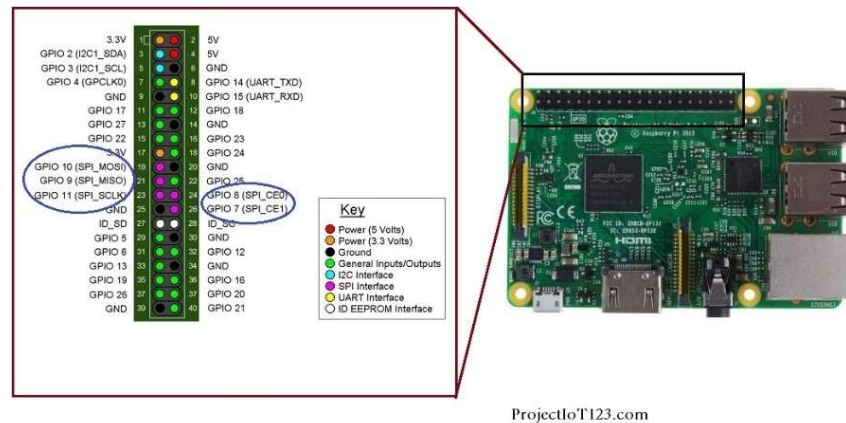
2.2.3 Příprava Rpi

SD karta s vypáleným obrazem systému se vkládá do Raspberry Pi. Současně se připojuje napájecí zdroj, který by měl poskytovat napětí přibližně 5 V a proud přibližně 3 A. Používá se zdroj kompatibilní s modulem, aby se předešlo poškození zařízení. Dále se připojuje zobrazovací zařízení (displej), na kterém se bude pracovat, a k Raspberry Pi se

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

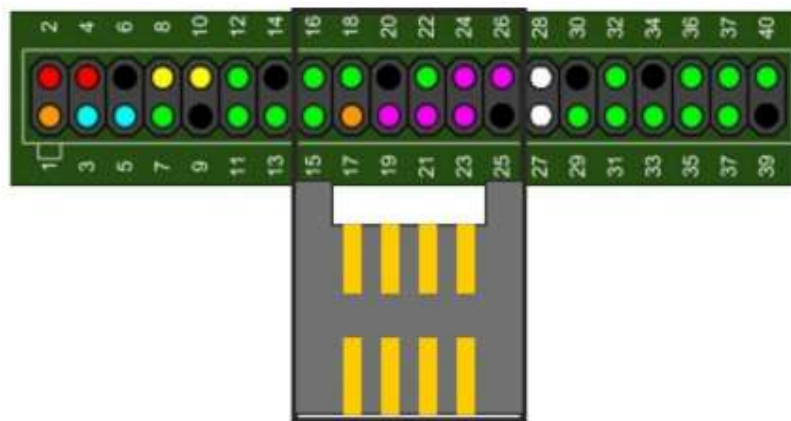
připojuje ethernetový kabel pro připojení k síti. K správě systému se připojují také myš a klávesnice. [8]

Aby se předešlo vkládání IQRF modulu KON-RASP-01 za běhu zařízení, což je podle dokumentace nevhodné, připojuje se modul předem. [11]



Obrázek 18 - Funkce GPIO pinů [23]

GPIO piny jsou chráněny pomocí analogového spínače, což minimalizuje riziko jejich poškození. Pro komunikaci s transceivery využívá IQRF SPI rozhraní, přičemž Raspberry Pi je implementováno jako SPI slave.



Obrázek 19 - Vložení KON-RASP-01 [11]

2.2.4 Konfigurace RPI

Výchozí konfigurace pro ukázkou se nastavuje s přihlašovacími údaji: *user=pi*, *password=Raspberry*. Následně je potřeba systém aktualizovat na aktuální verzi pomocí příkazu *sudo apt-get update && sudo apt-get -y full-upgrade*.

SPI

Pro povolení SPI rozhraní se upravuje konfigurační soubor s příkazem *sudo nano /boot/config.txt*. V tomto souboru se odkomentuje řádek *dtoverlay=spi=on*, tím se rozhraní povolí. Aby se změny projeví, je nutné systém restartovat příkazem *sudo reboot*.

2.2.5 MQTT

Instalace MQTT brokeru na Raspberry Pi. *Message Queuing Telemetry Transport* (MQTT) je komunikační protokol navržený pro zařízení s omezenými zdroji a nízkou šířkou pásma. Slouží jako prostředník mezi klienty, kteří posílají nebo přijímají zprávy. Tento protokol nachází široké uplatnění v IoT sítích, kde jsou data zpracovávána brokerem, monitorována a zobrazována uživateli. [24]

Pro práci je možné vybírat z mnoha dostupných brokerů. V tomto prostředí se pracuje s *Mosquitto*, což je open-source řešení (další možnosti HiveMQ, Eclipse MQTT). Standardní komunikační port MQTT je 1883.

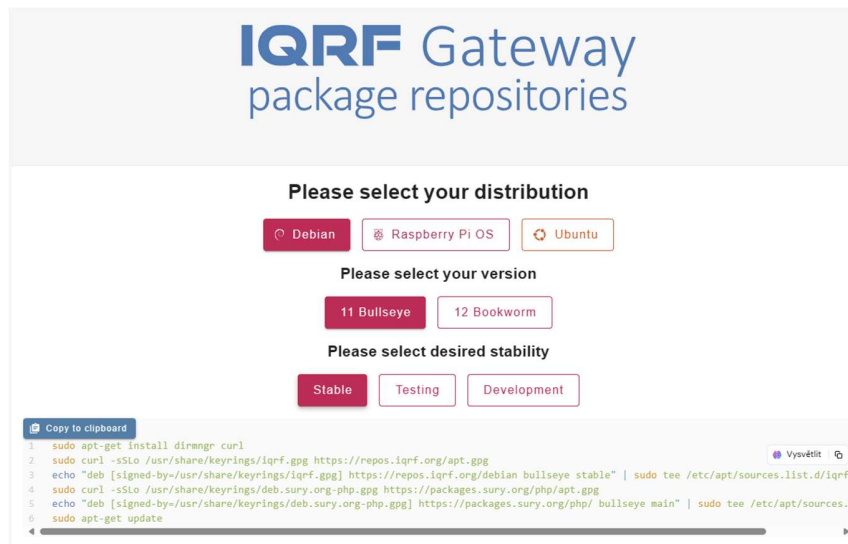
Instalace samotné služby tedy zajišťuje funkci MQTT serveru. Po instalaci se broker spustí automaticky jako systémová služba. Download proběhne spuštěním příkazu - *sudo apt-get install -y mosquitto mosquitto-clients*. Důležité je, aby instalace proběhla bez chyb a mosquitto fungoval - *systemctl status mosquitto.service* – ověření služby. Pokud by bylo potřeba nastavit jiná nastavení, potom se musí postupovat přes hlavní konfigurační soubor (Nastavení portů, zabezpečení, anonymní přístup).

2.2.6 IQRG Gateway Daemon

Softwarový proces běžící na Linuxu umožňuje komunikaci mezi IQRG sítěmi a externími aplikacemi. Na Linuxu je *Daemon* proces navržený tak, aby běžel na pozadí bez přímé interakce s uživatelem. Jeho úkolem je poskytovat služby nebo vykonávat specifické úlohy kontinuálně nebo na požádání. *IQRG Gateway Daemon* monitoruje a spravuje komunikaci mezi transceivery. *IQRG Gateway* jsou fyzická zařízení vybavená softwarem, která propojují IQRG síť s dalšími systémy. Slouží jako most mezi bezdrátovou sítí IQRG a systémy, jako jsou *MQTT* nebo *HTTP*. [25]

Před instalací samotné služby je nutné přidat některé balíčky, bez kterých by instalace nebyla možná. Mezi tyto balíčky patří správce klíčů, které umožňují komunikaci s keyservery a správu GPG klíčů. Klíče zajišťují, že balíčky z repositáře jsou ověřené a bezpečné pro použití. Následně se přidává nový repositář odpovídající verzi distribuce, který je dostupný na oficiálním IQRG webu. Po přidání repositáře je vhodné aktualizovat seznamy

balíčků a provést instalaci samotné služby příkazem: `sudo apt-get update && sudo apt-get install -y iqr-f-gateway-daemon`. Po instalaci je potřeba ověřit, že služba funguje správně.



Obrázek 20 - Výběr repozitáře shodný s distribucí [26]

2.2.7 IQRF Gateway Daemon WebApp

Pokud je služba *active (running)*, pokračuje se instalací další části, kterou je webová aplikace *IQRF Gateway Daemon WebApp*. Jedná se o webové uživatelské rozhraní určené pro správu sítě, konfiguraci gatewaye, diagnostiku a monitorování.

Hlavní funkcí této aplikace je konfigurace brány, která umožňuje definovat mapování mezi transceiverem a kontrolérem. Přizpůsobuje komunikaci pomocí parametrů SPI a UART. Aplikace nabízí také vzdálený přístup přes webový prohlížeč a možnost správy přístupů pro více uživatelů.

Pro instalaci *IQRF Gateway Daemon WebApp* je nutné mít aktivního *Daemona*, který slouží jako backend pro WebApp. Instalace aplikace se provádí příkazem: `sudo apt-get install -y iqr-f-gateway-webapp`. Po instalaci je aplikace dostupná prostřednictvím adresy <http://localhost> nebo <http://127.0.0.1>. *Localhost* je označení pro počítač, na kterém běží aplikace nebo proces, a používá se pro testování bez potřeby připojení k externí síti.

Při prvním vstupu do webového rozhraní je nutné vytvořit *credentials* neboli přístupové údaje. Aby modul, na kterém je připojen transceiver, mohl správně komunikovat přes SPI, je potřeba nastavit komunikační rozhraní. Pro výchozí pozici modulu lze využít předpřipravený modul *KON-RASP-01*, který má defaultní hodnoty. Pro „Power enable GPIO pin“, který má účel ovládání napájení transceiveru, musí být aktivováno nastavení tohoto pinu na logickou hodnotu 1 (GPIO pin 23). Další „Bus SPI GPIO pin“ je využíván jako součást SPI komunikace. Je potřebné zajištění správné konfigurace, aby mohl IQRF modul a Rpi

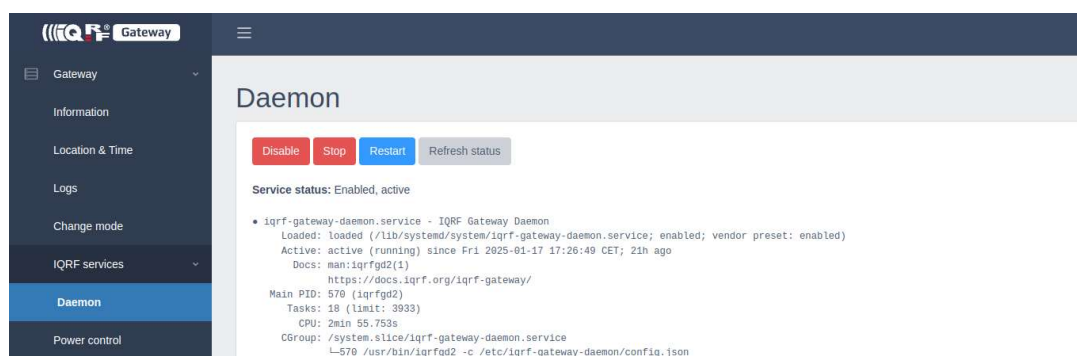
Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

komunikovat prostřednictvím SPI (GPIO pin 7). Poslední „Programming Mode Switch GPIO Pin“ je zodpovědný za přepínání modů transceiveru.

The screenshot shows a web-based configuration interface for the IQRf Gateway. It features several input fields for GPIO pins: 'Power enable GPIO pin' (set to 23), 'Bus enable GPIO pin' (set to 7), and 'Programming mode switch GPIO pin' (set to 22). There are also checkboxes for 'Enable SPI reset' (checked) and 'Use communication interface pins' (unchecked). Below these are three empty input fields for 'I2C interface enable GPIO pin', 'SPI interface enable GPIO pin', and 'UART interface enable GPIO pin'. A 'Save' button is located at the bottom left of the configuration area.

Obrázek 21 - Nastavení SPI interface

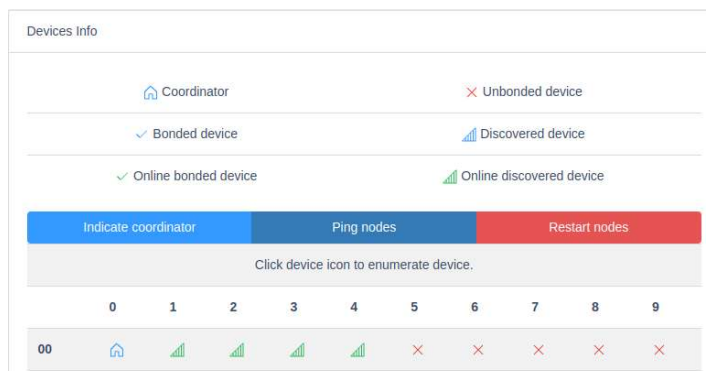
Aby se změny projevíly, je nutné restartovat službu. To se provádí v nastavení gatewaye v sekci *Daemon – Restart*.



Obrázek 22 - Služba Daemon ve webovém rozhraní

Správná funkčnost systému se ověřuje prostřednictvím Network Manageru. V sekci zařízení jsou zobrazovány všechny přibondované nody, což umožňuje snadnou kontrolu připojení. Pomocí tlačítka *Indicate Coordinator* se aktivuje červená LED na koordinátoru, která začne blikat a potvrzuje správnou komunikaci. Webová aplikace (WebApp) zároveň poskytuje možnost odesílání DPA příkazů, což umožňuje nejen ověřovat funkčnost, ale také

efektivně spravovat celou síť. Na závěr se provádí restart Raspberry Pi. Pokud služba naběhne i po novém spuštění, potvrzuje se správná funkčnost systému.



Obrázek 23 - Sekce zařízení v Network manager

2.2.8 Node-RED

Node-RED je vizuální prostředí ideální pro IoT a automatizační aplikace. Nabízí snadné vizuální propojení komponent a datových toků, přičemž běží na *Node.js*. Při instalaci na Raspberry Pi je potřeba zajistit přítomnost dalších balíčků. Klíčovým nástrojem je *curl*, který slouží pro stahování obsahu z internetu. Ten se instaluje příkazem: `sudo apt-get install -y curl`. Pro instalaci *Node-RED* se využívá skript, který se spouští příkazem: `bash <(curl -sL https://raw.githubusercontent.com/node-red/linux-installers/master/deb/update-nodejs-and-nodered)`. Tento skript provádí několik důležitých kroků: Odstraňuje starou verzi *Node-RED*, pokud je již nainstalována. Kontroluje a aktualizuje instalaci *Node.js*. Instaluje *Node-RED* pomocí *npm* (*Node Package Manager* – správce balíčků pro *Node.js*). Nastavuje základní konfigurační soubor dle uživatele.

Zajištění automatického spuštění při startu systému je potřeba provést povolení: `sudo systemctl enable nodered.service` a `sudo systemctl start nodered.service`.

Pro práci s *Node-RED* je potřeba doinstalovat některé z dalších palet nodů. Jednou z nich je *node-red-dashboard*, která umožňuje vytváření uživatelského rozhraní pro vizualizaci dat. Toto rozhraní je po instalaci přístupné na adrese <http://localhost:1880/ui>. Instalace *node-red-dashboard* se provádí přímo v prostředí *Node-RED* prostřednictvím správy palet nebo lze i příkazem v terminálu.

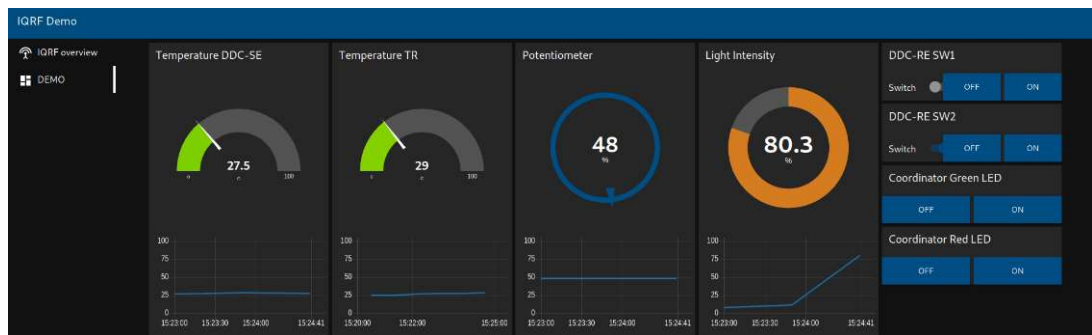
Od IQRF je dostupný *starter-kit*, který usnadňuje integraci *Node-RED* do IoT projektů. Součástí starter-kitu jsou šablony pro *Node-RED*, které umožňují vizualizaci dat z IQRF zařízení. Díky přiloženým modulům je zajištěna komunikace mezi *Node-RED* a IQRF přes *MQTT*. Repositář starter-kitu se klonuje z *GitLab* do domovské složky na Raspberry Pi pomocí příkazu: `git clone https://gitlab.iqrf.org/alliance/iot-starter-kit.git`. Tím se vytvoří nová složka *iot-starter-kit*, která obsahuje všechny soubory z repositáře. Tyto soubory je

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

následně potřeba zkopírovat do adresáře s konfigurací *Node-RED*, konkrétně do složky: */home/pi/.node-red*. V závěru proběhne restart Node-RED pro načtení nových změn. Celý postup:

```
cd /home/pi
git clone https://gitlab.iqrf.org/alliance/iot-starter-kit.git
cd iot-starter-kit/install
cp rpi-board/node-red/* /home/pi/.node-red
sudo systemctl restart nodered.service
```

Pokud je služba správně spuštěná (lze ověřit pomocí `systemctl status nodered.service`), na adrese <http://localhost:1880> by měly být viditelné IQRF flows pro komunikaci. V případě, že IQMESH síť funguje, jednotlivé nody komunikují s koordinátorem, který následně komunikuje s Raspberry Pi. Daemon pracuje ve spojení s MQTT a Node-RED. Pokud jsou všechny tyto komponenty správně propojeny a funkční, měla by být data zobrazena na dashboardu dostupném na adrese <http://localhost:1880/ui>.



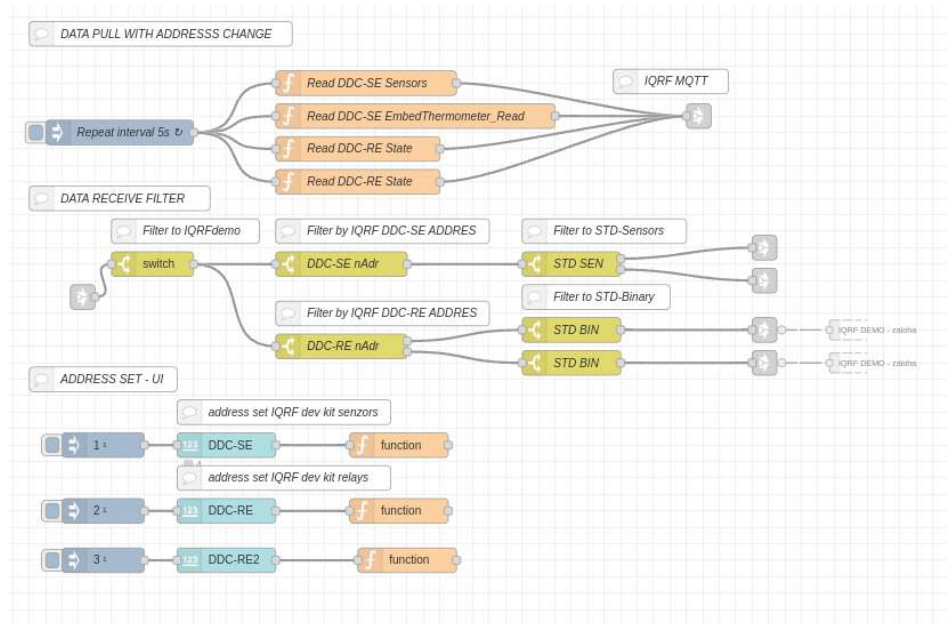
Obrázek 24 - Node-RED dashboard s daty ze sítě

2.2.9 Tvorba flows a dashboardu

Celý proces je založen na konfiguraci nodů v prostředí *Node-RED*. Funkčnost jednotlivých nodů a *flows* bude vysvětlena na konkrétních ukázkách. Flows jsou tvořeny třemi záložkami.

Popis začíná záložkou *IQRF Overview*.

Dashboard *IQRF Overview* slouží k přehledu IQRF zařízení, ke kterým jsou připojeny moduly s rozšiřující periferií. V tomto případě se jedná o nody s modulem DDC-SE-01 a dva moduly s relé DDC-RC-01.



Obrázek 25 – Node-RED IQRF overview

Nastavení správné adresy nodů

V dolní části rozhraní, označené „ADDRESS SET - UI“, se provádí nastavení logické adresy nodu. Používají se následující nody:

- **Inject** – při startu nastaví výchozí adresy.
- **UI_numeric** – součást knihovny Dashboard, umožňuje ruční nastavení hodnoty.
- **Function** – ukládá adresu do globální proměnné pomocí `global.set('DDC-RE2', msg.payload);`.

Horní část flow slouží k odeslání zpracovaných dat do MQTT out:

- **Inject** – každých 5 sekund provádí aktualizaci pro kontrolu dat.
- **Function** – získává adresy z globální proměnné a následně pracuje s IQRD DPA API, což umožňuje ovládání zařízení v síti.

```

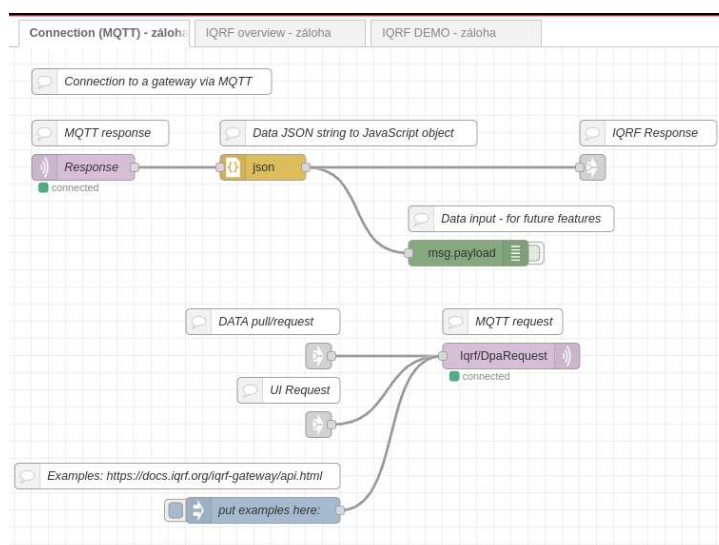
var adress=global.get("DDC-RE");
var request=
{
    "mType": "iqrFBinaryoutput_SetOutput", //typ zprávy jaké operace se provádí, odlišení zpráv, aby se
    //zpracovaly odpovídajícím způsobem
    "data": {
        "msgId": "IQRFDemo", //zde se nastavuje msgId pro oddělení různých typů zpráv a přehlednosti
        "req": {
            "nAdr": 1, //výchozí adresa
            "param": {
                "binOuts": []
            }
        },
        "returnVerbose": true
    }
};
request.data.req.nAdr=adress; //nastavení ručně nastavené adresy
msg.payload=request; //uložení do msg
return msg;

```

- **Link out** – propojuje více flows bez nutnosti přímého propojení dráty. V tomto případě posílá data do MQTT.

Prostřední část flow bude popsána později. Nyní bude představena část Connection (MQTT).

Připojení k MQTT brokeru



Obrázek 26 – Node-RED Connection MQTT

Sekce Connection (MQTT) zajišťuje připojení k MQTT brokeru a zpracování zpráv ze/do systému IQRF.

Přehled hlavních nodů:

- **Link in** – umožňuje propojení více flows bez přímého propojení drátem.
- **MQTT out** – odesílá zprávy do brokeru. Nastavují se parametry:
 - Adresa serveru: 127.0.0.1:1883
 - *Topic*: iqrf/DpaRequest (určuje, že zprávy jsou směřovány do IQRF)
 - *QoS* (Quality of Service) – potvrzení doručení zprávy
- **MQTT in** – naslouchá zprávám na MQTT brokeru (127.0.0.1:1883). Při přijetí zprávy ji předává jako msg.payload, topic, qos.
- **JSON** – převádí přijatá data z formátu JSON na objekt JavaScriptu.
- **Link out** – odesílá zpracovaný objekt zpět do IQRF Overview, kde dochází k dalšímu filtrování a zpracování dat.

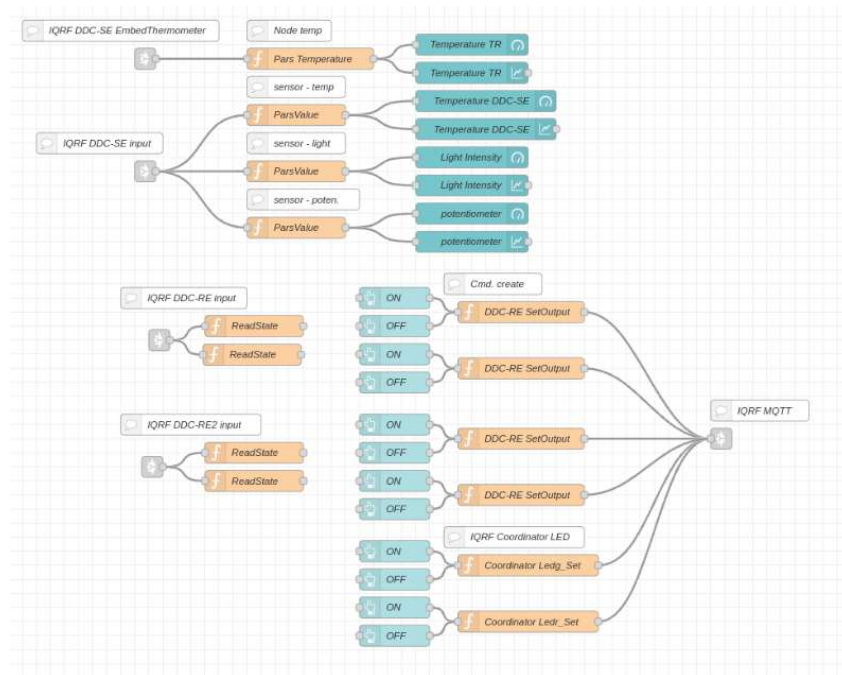
Směrování zpráv v sekci *IQRF Overview*

V této části dochází k podmíněnému směrování zpráv a jejich filtrování podle specifických kritérií.

- **Switch** – provádí směrování zpráv na základě hodnoty msg.payload.data.msgId, pokud msgId == "IQRFdemo", zpráva je dále zpracována. Tento mechanismus umožňuje rozpoznání zpráv IQRF a jejich odlišení od ostatních MQTT zpráv.
- **Switch 2** – filtruje adresy nodů na základě payload.data.rsp.nAdr. Propouští pouze zprávy odpovídající dané globální proměnné (například DDC-RE2). Tento krok zajišťuje, že systém pracuje pouze s daty ze specifických zařízení.
- **Switch 3** – rozděluje zprávy podle hodnoty msg.payload.mType, která určuje typ prováděné operace. Pro DDC-SE (senzorové zařízení):
 - "iqrfSensor_ReadSensorsWithTypes" – určuje čtení hodnot senzorů IQRF.
 - "iqrfEmbedThermometer_Read" – identifikuje čtení teplotních dat z IQRF Embed Thermometer.
- **Link out** – přeposílá filtrované zprávy do poslední sekce IQRF DEMO, kde probíhá jejich konečné zpracování.

IQRF DEMO

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100



Obrázek 27 – Node-RED IQRf DEMO

Záložka *IQRf DEMO* obsahuje několik function nodů v kombinaci s dashboard nody z knihovny node-red-dashboard. Function nody provádějí selekci dat a dále pracují se strukturou DPA handleru.

Klíčovou roli hrají parametry *hwpld*, které určují ovládání výstupů pro různé typy zařízení a podle indexů identifikují jednotlivé komponenty modulů.

- Pro DDC-SE modul jsou senzory ukládány do pole *sensors*, přičemž každý senzor má svůj specifický index.
- Pro DDC-RE modul function node ukládá přijatou hodnotu a vytváří požadavek na změnu stavu binárního výstupu. Indexy určuje, které relé má být výstupem.

Tento mechanismus umožňuje efektivní zpracování dat a následné ovládání připojených IQRf zařízení v prostředí *Node-RED*.

```
var stateOfRel=msg.payload;
msg.payload={
  "mType": "iqrBinaryoutput_SetOutput",
  "data": {
    "msgId": "IQRfdemo",
    "req": {
      "nAdr": 2,
      "param": {
        "binOuts": [ {
          "index": 1,
          "state": true }
        ]
      }
    }
  }
}
```

```
    }},  
    "returnVerbose": true  
  }  
};  
msg.payload.data.req.param.binOuts[0].state=stateOfRel;  
msg.payload.data.req.nAdr=global.get("DDC-RE");  
if (global.get("DDC-RE-hwpId")==3074) {  
  msg.payload.data.req.param.binOuts[0].index=3; }  
if (global.get("DDC-RE-hwpId")==18434) {  
  msg.payload.data.req.param.binOuts[0].index=1; }  
return msg;
```

Zpracování hodnot v UI nodech

Zpracování hodnot – pracují s proměnnou {{value}}, která obsahuje aktuální data.

Vizualizace – zobrazují hodnoty senzorů nebo stav výstupů zařízení. (chart, gauge)

Ovládací prvky – umožňují interakci s IQRF zařízeními, například změnu stavu binárních výstupů nebo nastavení parametrů.

Díky propojení s function nody UI nody přehledně zobrazují zpracovaná data a umožňují jejich řízení v reálném čase.

2.2.10 Tvorba Node-RED pro sonický senzor

Pro správnou komunikaci je nezbytné porozumět fungování IQRF Gateway Daemon API. V případě, kdy není využíván IQRF standard, je nutné použít IQRF Generic API, což je univerzální formát pro přenos dat přes MQTT do IQRF sítě. [27]

Pro odesílání DPA příkazů se využívá IQRF raw formát. Struktura Request/Response je zapisována do pole rData, přičemž jednotlivé hodnoty jsou rozděleny na dvojbajtové sekvence, které jsou odděleny tečkou. Tento formát zajišťuje konzistentní a správný přenos příkazů v rámci IQMESH sítě.

- nAdrL . nAdrH . PNUM . PCMD . HWPID_L . HWPID_H . PData



```
Response example iqrRaw v1.0.0 Copy X
1 {
2   "mType": "iqrRaw",
3   "data": {
4     "msgId": "testRaw",
5     "timeout": 1000,
6     "rsp": {
7       "rData": "00.00.06.83.00.00.00.44"
8     },
9     "raw": {
10      "request": "00.00.06.03.ff.ff",
11      "requestTs": "2018-08-31T12:48:35.887733",
12      "confirmation": "",
13      "confirmationTs": "",
14      "response": "00.00.06.83.00.00.00.44",
15      "responseTs": "2018-08-31T12:48:35.915579"
16    },
17    "insId": "iqrfgd2-1",
18    "statusStr": "ok",
19    "status": 0
20  }
21 }
```

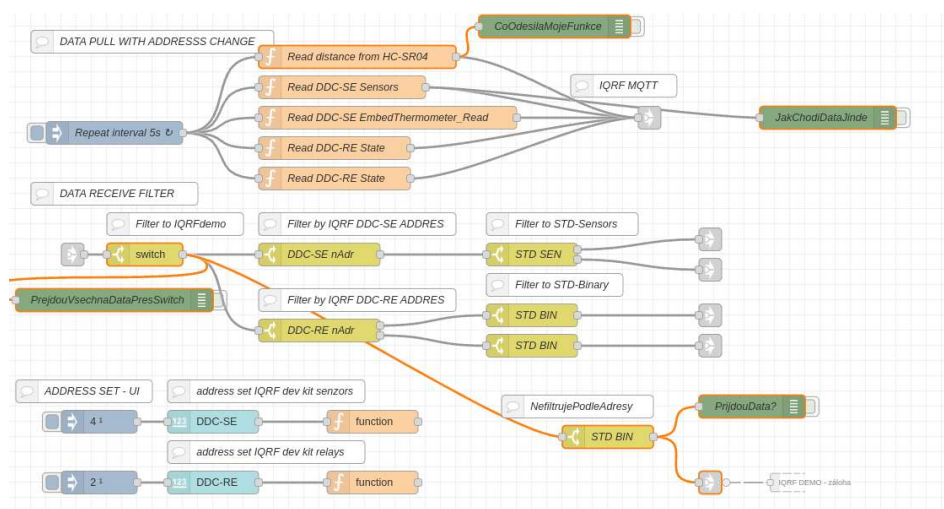
Obrázek 28 - Ukázkový příklad z dokumentace IQRf[27]

Pro správné sestavení DPA příkazu je nutné dodržet přesnou strukturu a odpovídající hodnoty.

Pokud je node umístěn na adrese 1, první bajt (nižší byte) bude obsahovat hodnotu 01. Následně se nastaví: PNUM – uživatelská periferie s hodnotou 20, PCMD – nastavena na 00, HWPID – univerzální hodnota FF.FF.

Výsledný příkaz bude vypadat takto: "rData": "01.00.20.00.FF.FF". Tento příkaz je následně vložen do funkce, která je připojena k link out, odkud se odešle na MQTT out.

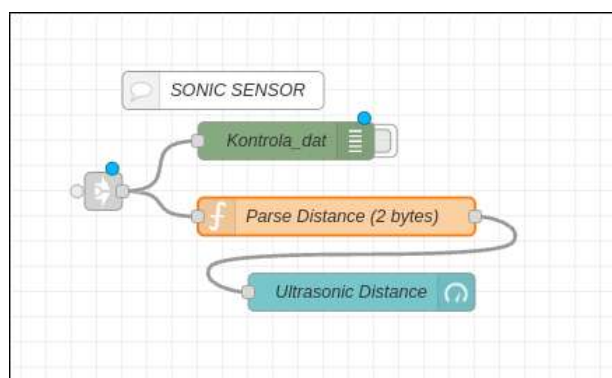
Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100



Obrázek 29 - Označené uzly, které byly přidány pro HC-SR04

Zpětná odpověď přichází prostřednictvím MQTT in na topic `Iqrf/DpaResponse`. Odpovědní pole `rsp.rData` obsahuje DPA rámec s výsledkem, včetně dvou bajtů pro data z `CustomDpaHandleru`.

Filtrování je potřeba na základě `msgID` – používá se IQRf DEMO jako identifikátor a typu zprávy (`mType`) – na základě `iqrfRaw` se data odesílají do funkce, která provádí parsování pro dashboard.



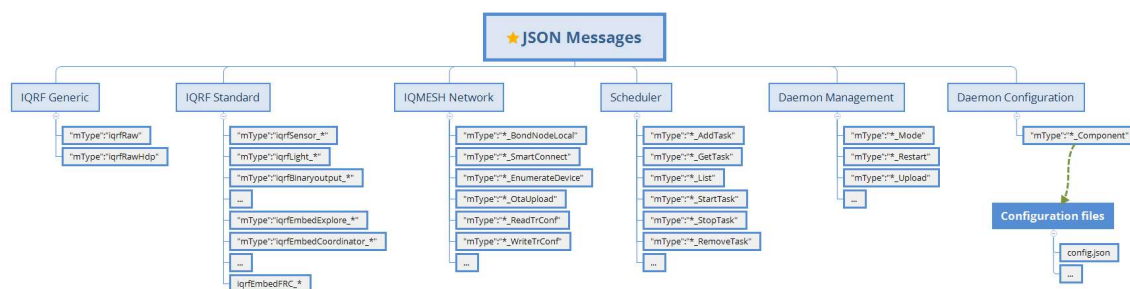
Obrázek 30 - Přidaná část do záložky IQRf DEMO

```

1 // "01.00.20.00.0f.00.00.46.c1.00" data v podobném formátu chodí ze senzoru
2 // získány v iqrF webapp
3 let rDataString = msg.payload.data.rsp.rData;
4 let arr = rDataString.split('.'); //rozdělení dat podle tečky
5 let hexByte = arr[8]; // data jsou na deváté pozici
6
7 let value = parseInt(hexByte, 16); // Převédeme hex => decimal
8
9 // value pro dashboard
10 msg.payload = value;
11 return msg;
12

```

Obrázek 31 - Parsování dat pro dashboard



Obrázek 32 - Ukázka mType možnosti

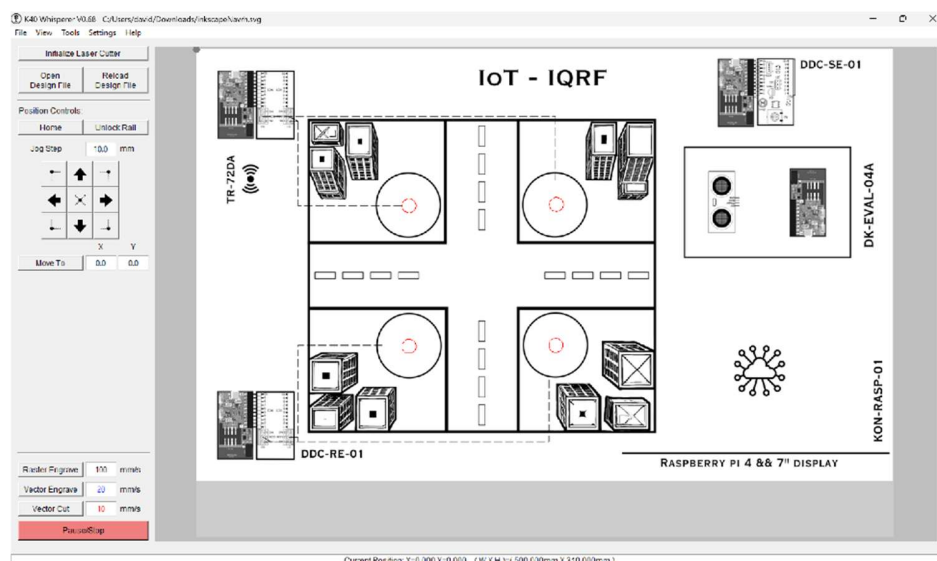


Obrázek 33 - Výsledek zobrazený v Node-RED

2.3 Návrh desky

Při tvorbě návrhu desky bylo nejprve nutné promyslet celkový návrh a rozložení jednotlivých prvků, aby byla efektivně demonstrována funkčnost chytré ulice. Za tímto účelem byl využit generativní nástroj ChatGPT pro vytvoření počáteční vizuální podoby [28], která sloužila jako základní návrh, který poukázal na to, že toto řešení je všestranné, řeší obecný problém a je možné implementovat ho na jakýkoliv problém.

Následně byla přepracována úprava v inkscape. Jedná se o nástroj pro tvorbu vektorové grafiky. Ve webové aplikaci Canva byly upraveny texty a vylepšena vizuální stránka návrhu. SVG soubor byl pak nahrán do programu K-40 Whisper, který pracuje s laserem. Tímto způsobem vznikla deska.



Obrázek 34 - K40 whisper návrh řešení

Při návrhu byla využita schémata modulů z datasheetů z IQRF za účelem lepší demonstrace prvků. Důležitým aspektem bylo také promyšlení způsobu, jakým bude na desce vizuálně znázorněna bezdrátová komunikace, jelikož tato technologie není přímo viditelná. Byla znázorněna přerušovanou čarou. Deska byla také doplněna o grafické prvky pro lepší pochopení studentů.

2.3.1 3D tisk

Pro lepší efekt fyzické podoby byl využit 3D tisk, který demonstruje lampy, kontejner a umožňuje uchycení modulů na desku.

Moduly byly staženy z [printables.com](https://www.printables.com), cults3d.com a bylo s nimi nakládáno dle licencí.

- Lampa - autor Nikki, [streetLamp](#) [29]
- Kontejner – autor CosyMaker79, [GarbageBin](#) [30]
- Držák na boxy – autor Tbah, [remixovaný IQRF box](#) [31]
- Držák na display s uchycením řídicí jednotky – autor hallbraden, [drzak](#) [32]

2.3.2 Skládání finálního modelu

Při skládání všeho dohromady přišlo na propojení hardwaru samotného IQRF se zbytkem. K modulu DK-EVAL-04A se připojí druhý modul, který obsahuje relé (DDC-RE-01). Jak relé ovládat, bylo vysvětleno výše. To samé se provede i s modulem se senzory (DDC-SE-01). Vše se dá na určené místo.

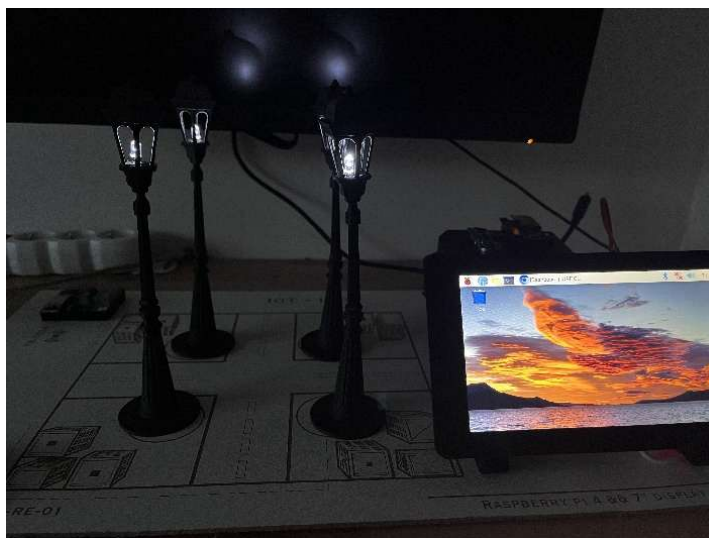
DDC-SE-01 nepotřebuje jinou periférii na připojení, proto stačí, když bude takto samostatně. Reprezentovat může bytový dům, kde se sleduje teplota, na večer při tmě se

mohou rozsvítit světla nebo pomocí potenciometru upravovat intenzitu světel či řídit průtok klimatizace.

K vytvoření funkčních lamp, které jsou připojeny na relé IQRF modulu je za potřebí LED dioda s rezistorem o hodnotě $220\ \Omega$. Komponenty byly připájeny, čímž se zajistila funkčnost. Lampy jsou stavěny tak, aby nebyly hned poškozeny, například při usazování na desku či manipulaci s ní.

2.3.3 Displej zapojení

Pro připojení displeje je potřeba užít plochý kabel (15 pinový) a napájecí kabely. Jelikož je snaha o co největší ušetření infrastruktury, proto se bude napájet displej přímo přes GPIO pinů. 5 V z RPI na pinu 2 a GND na pinu 6. Plochý kabel se připojí k Raspberry a displej funguje. [33]



Obrázek 35 - Finální testování displeje se světly

ZÁVĚR

Tato práce se zaměřila na návrh a realizaci výukového simulačního modulu s využitím technologie IQRf, který slouží k demonstraci principů chytré infrastruktury města. Cílem bylo vytvořit funkční model, ale také zpřístupnit studentům technologie využívané v IoT sítích.

Během realizace se objevila řada problémů, především u softwarového nastavení nebo vzájemné komunikace jednotlivých prvků. Přesto se podařilo vyvinout plně funkční didaktickou pomůcku, která studentům umožní lépe pochopit fungování IQRf sítě a její možné využití v praxi.

Výsledkem této práce je funkční IoT síť postavená na technologii IQRf. Doprovodná dokumentace poskytuje jasný přehled o použitých technologiích, jejich implementaci a správě, zatímco metodické listy pomáhají krok za krokem s nastavením celého systému. Tento materiál je snadno použitelný pro výuku a slouží zároveň jako podpora při praktické maturitní zkoušce.

ZDROJE A LITERATURA

- [1] IoT Port. Co to je IoT [online]. 2020. [cit. 2024-10-24]. Dostupné z: <https://www.iotport.cz/iot-novinky/ostatni-clanky-o-iot/co-to-je-iot>
- [2] REALBUSINESS.CO.UK. From 1982 Coca-Cola vending machine to the latest trend: what the internet of things means for business [online]. [cit. 2025-03-04]. Dostupné z: <https://realbusiness.co.uk/from-1982-coca-cola-vending-machine-to-latest-trend-what-the-internet-of-things-means-for-business>
- [3] Statista. Number of Internet of Things (IoT) connected devices worldwide from 2019 to 2030 [online]. 2025 [cit. 2024-10-28]. Dostupné z: <https://www.statista.com/statistics/1183457/iot-connected-devices-worldwide/>
- [4] IQRF Tech s.r.o. ZÁKLADY TECHNOLOGIE IQRF. Interní materiál získaný ve škole v rámci programu Smart School, 2017. [cit. 2024-10-28].
- [5] IQMESH, IQRF. YouTube [online video]. [cit. 2025-03-04]. Dostupné z: <https://www.youtube.com/watch?v=UynZCPe2k3U>
- [6] IQRF Tech s.r.o. User's Guide IQRF OS 4.06G for TR-7xG [PDF online]. 2023. [cit. 2025-03-04]. Dostupné z: [Downloads - IQRF](#)
- [7] IQRF Tech s.r.o. Tech Guide—IQMESH Network Deployment [PDF online]. 2023. [cit. 2025-03-04]. Dostupné z: [Downloads - IQRF](#)
- [8] ChatGPT. ChatGPT [online]. [cit. 2024-10-31]. Dostupné z: <https://chatgpt.com/>.
Úprava textu
- [9] MICRORISC s.r.o. IQRF DPA v4.31 Framework: Technical Guide [PDF online]. 2024. [cit. 2025-03-04]. Dostupné z: [Downloads - IQRF](#)
- [10] IQRF Tech s.r.o. DK-EVAL-04A IQRF Development Kit: User's Guide [PDF online]. 2021. [cit. 2025-03-04]. Dostupné z: [Downloads - IQRF](#)
- [11] MICRORISC s.r.o. KON-RASP-01 IQRF Shield for Raspberry Pi: User's Guide [PDF online]. 2015. [cit. 2025-03-04]. Dostupné z: [Downloads - IQRF](#)
- [12] Wikipedie. Serial Peripheral Interface [online]. [cit. 2024-10-31]. Dostupné z: https://cs.wikipedia.org/wiki/Serial_Peripheral_Interface
- [13] Wikipedie. UART [online]. [cit. 2024-10-31]. Dostupné z: <https://cs.wikipedia.org/wiki/UART>

- [14] Node-RED. About [online]. [cit. 2024-10-31]. Dostupné z: <https://nodered.org/about/>
- [15] Node-RED. Securing Node-RED [online]. [cit. 2025-03-04]. Dostupné z: <https://nodered.org/docs/user-guide/runtime/securing-node-red>
- [16] Raspberry Pi Foundation. Raspberry Pi 4 Model B Datasheet [PDF online]. [cit. 2024-10-31]. Dostupné z: <https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-datasheet.pdf>
- [17] OPENAI. ChatGPT [online]. [cit. 2025-03-04]. Dostupné z: <https://chat.openai.com/>. Získání informací o pinech.
- [18] OPENAI. ChatGPT [online]. [cit. 2025-03-04]. Dostupné z: <https://chat.openai.com/>. Získání informací o registrech LAT a TRIS.
- [19] Návod Drátek. Měřič vzdálenosti ultrazvukový [online]. [cit. 2025-03-04]. Dostupné z: [Měřič vzdálenosti ultrazvukový | Návod Drátek](#)
- [20] iqrfsdk. iot-starter-kit/install/rpi-board/GW-SbS-INSTALL.md at master – iqrfsdk/iot-starter-kit – GitHub [online]. [cit. 2025-03-04]. Dostupné z: [iot-starter-kit/install/rpi-board/GW-SbS-INSTALL.md at master · iqrfsdk/iot-starter-kit · GitHub](#)
- [21] IQRf Tech s.r.o. Tech Guide—IQRf SPI TR-7xD+7xG [PDF online]. 2023. [cit. 2025-03-04]. Dostupné z: [Downloads - IQRf](#)
- [22] Enabling the SPI port in Raspberry Pi. [online]. [cit. 2024-10-31]. Dostupné z: [Enabling the SPI port In Raspberry Pi -](#)
- [23] Projectiot. Enabling the SPI port in Raspberry Pi [online obrazový dokument]. [cit. 2024-12-16]. Dostupné z: [Enabling the SPI port In Raspberry Pi -](#)
- [24] MQTT. MQTT.org [online]. [cit. 2025-01-13]. Dostupné z: <https://mqtt.org/>
- [25] IQRf Tech s.r.o. IQRf Gateway installation guide [PDF online]. 2023. [cit. 2025-03-04]. Dostupné z: [Downloads - IQRf](#)
- [26] IQRf Repository. Repos.iqrf.org [online]. [cit. 2025-03-04]. Dostupné z: <https://repos.iqrf.org/>
- [27] Daemon API v2.6.x | IQRf Gateway documentation. [online]. [cit. 2024-10-31]. Dostupné z: [Daemon API v2.6.x | IQRf Gateway documentation](#)

[28] ChatGPT. ChatGPT [online]. [cit. 2024-10-31]. Dostupné z: <https://chatgpt.com/>.
Generativní nástroj, který navrhnul vizuální podobu desky.

[29] NIKKI. Desktop Street Lamp [3D model online]. [cit. 2025-03-04]. Dostupné z: [Desktop Street Lamp od autora Nikki | Stáhněte si zdarma STL model | Printables.com](#)

[30] COSYMAKER79. GarbageBin (kontejner) [3D model online]. [cit. 2025-03-04].
Dostupné z: [Garbage bin od autora CosyMaker79 | Stáhněte si zdarma STL model | Printables.com](#)

[31] TBAH. SKADIS Dremel drill bit box holder [3D model online]. [cit. 2025-03-04].
Dostupné z: [SKADIS Dremel drill bit box holder od autora Tbah | Stáhněte si zdarma STL model | Printables.com](#). Upravený model

[32] HALLBRADEN. Raspberry Pi 7" touch display stand [3D model online]. [cit. 2025-03-04]. Dostupné z: [Free STL file Raspberry Pi 7" touch display stand - 3D printer model to download - Cults](#)

[33] Raspberry PI Touchscreen Connection for 7in. Touch Display [online]. [cit. 2024-10-31]. Dostupné z: [Raspberry PI Touchscreen Connection for 7in. Touch Display](#)

[] CALVO, Isidro; GIL-GARCÍA, José Miguel; RECIO, Igor; LÓPEZ, Asier; QUESADA, Jerónimo. Building IoT Applications With Raspberry Pi and Low Power IQRf Communication Modules [online]. [cit. 2025-03-04]. Dostupné z: [Building IoT Applications with Raspberry Pi and Low Power IQRf Communication Modules](#)

[] Jiri Skovranek, Martin Pies a Radovan Hajovsky. Use of the IQRf and Node-RED technology for control and visualization in an IQMESH network. IFAC-PapersOnLine [online]. 2018 [cit. 2024-10-31]. Dostupné z: <https://www.sciencedirect.com/science/article/pii/S2405896318309157?via%3Dihub>

[] Sobeslav, Vladimir & Horalek, Josef. (2020). A Smart Parking System Based on Mini PC Platform and Mobile Application for Parking Space Detection. Mobile Information Systems. 2020. [cit. 2024-10-31]. Dostupné z: [\(PDF\) A Smart Parking System Based on Mini PC Platform and Mobile Application for Parking Space Detection](#)

[] Root.cz. Bezdrátový přenos s IQRf: přizpůsobení chování transceiveru [online]. [cit. 2025-03-04]. Dostupné z: [Bezdrátový přenos s IQRf: přizpůsobení chování transceiveru - Root.cz](#)

[] Root.cz. Bezdrátový přenos s IQRf: čtení z externích čidel [online]. [cit. 2025-03-04]. Dostupné z: [Bezdrátový přenos s IQRf: čtení z externích čidel - Root.cz](#)

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

SEZNAM PŘÍLOH

Příloha č. 1: Metodický list č.1

Příloha č. 2: Metodický list č. 2

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

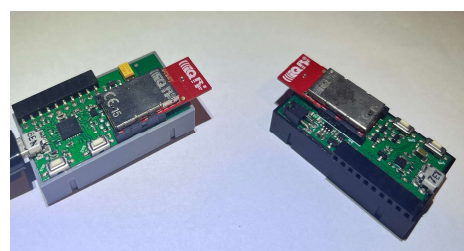
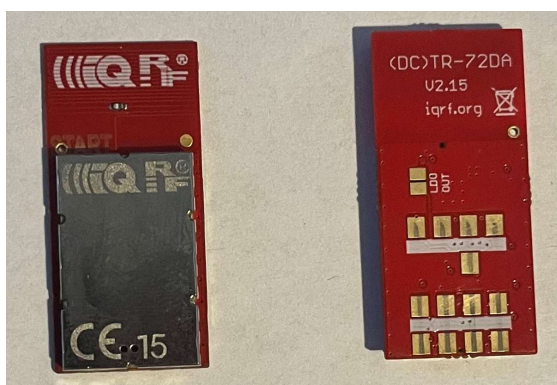
PŘÍLOHA Č. 1: METODICKÝ LIST Č.1

NÁZEV LISTU	ZÁKLADY PRÁCE S IQRF, BONDING PROCES
ZASAZENÍ DO VÝUKY	ELEKTRONICKÉ POČÍTAČE, IoT, INFORMAČNÍ TECHNOLOGIE
ZASAZENÍ DO SKUPINY	4. ROČNÍK SŠ, KONKRÉTNĚ SPŠ A VOŠ JIČÍN
PŘEDPOKLAD VYUŽITÍ METODICKÉHO LISTU	CVIČENÍ ZAMĚŘENÉ NA IQRF S VYUŽITÍM U PRAKTICKÉ MATURITNÍ ZKOUŠKY
CÍL VÝUKY	NAUČIT ŽÁKY ZÁKLADNÍ PRINCIPY NASTAVENÍ IQRF SÍTĚ, PŘEDEVŠÍM BONDOVÁNÍ NODŮ
VSTUPNÍ ZNALOST	ŽÁK MÁ PŘEDSTAVU O IoT, BYL SEZNÁMEN S TEORETICKOU ČÁSTÍ IQRF (ROZUMÍ POJMŮM MESH TOPOLOGY, NODE, COORDINATOR, BONDING, DISCOVERY, DPA PŘÍKAZ)
POPIS VÝUKY	TEORETICKÝ ÚVOD, UKÁZKA SESTAVENÍ, SAMOSTATNÁ PRÁCE ŽÁKŮ PO JEDNOM ČI SKUPINĚ PO DVOU (V ZÁVISLOSTI NA ZNALOSTI ŽÁKŮ ČI POČTU HARDWARU)
PŘÍPADNÉ PROBLÉM	POŠKOZENÝ HARDWARE, NODE PŘIPOJENÝ NA JINÝ KOORDINÁTOR, MOŽNÉ RF RUŠENÍ ŽÁKŮ NAVZÁJEM
PRAKTICKÁ ČÁST	SEZNÁMENÍ S HARDWAREM NASTAVENÍ TRANSCEIVERU BONDING

ZÁKLADY IQRF

POPIS:	PRO VYTVOŘENÍ SÍTĚ MUSÍ BÝT NAKONFIGUROVÁN TR MODUL. VLOŽENÍ TRANSCEIVERU DO KONFIGURÁTORU. POKUD JE KONFIGURÁTOR PŘIPOJEN K POČÍTAČI, MUSÍ BÝT STISKNUTÉ RESETOVACÍ TLAČÍTKO! TRANSCEIVER BUDE NASTAVEN PRVNĚ JAKO BĚŽNÝ NODE A NAKONEC BUDE NASTAVEN KOORDINÁTOR. KOORDINÁTOR SI PŘIBONDUJE NODY. VZNIKNE MESH SÍŤ.
--------	---

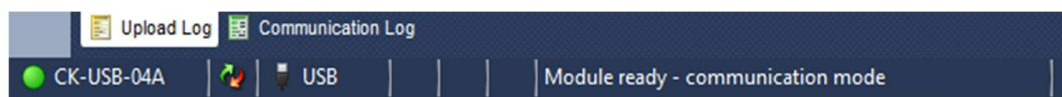
PŘÍPRAVA HARDWARU



VLOŽENÍ TRANSCEIVERU DO MODULU A
PŘIPOJENÍ K PC

OTEVŘÍT IDE

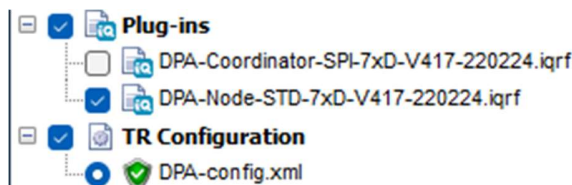
POKUD TR SPRÁVNĚ KOMUNIKUJE S POČÍTAČEM, MĚLO BY SE TOTO OBJEVIT V DOLNÍ ČÁSTI IDE.



ABY BYLO MOŽNÉ VYTVOŘIT IQMESH SÍŤ, JE NUTNÉ VYTVOŘIT NODY A KOORDINÁTOR.

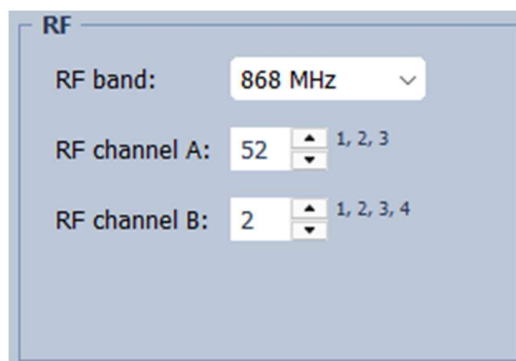
VYTVÁŘENÍ NODU

V PRAVÉ ČÁSTI IDE S NACHÁZÍ NABÍDKA “PROJECT”. TA SE SKLÁDÁ Z VÍCE MOŽNOSTÍ. V TUTO CHVÍLI JSOU DŮLEŽITÉ ČÁSTI “PLUG-INS” A “TR CONFIGURATION”. ZDE NASTAVÍME CHOVÁNÍ NODU.



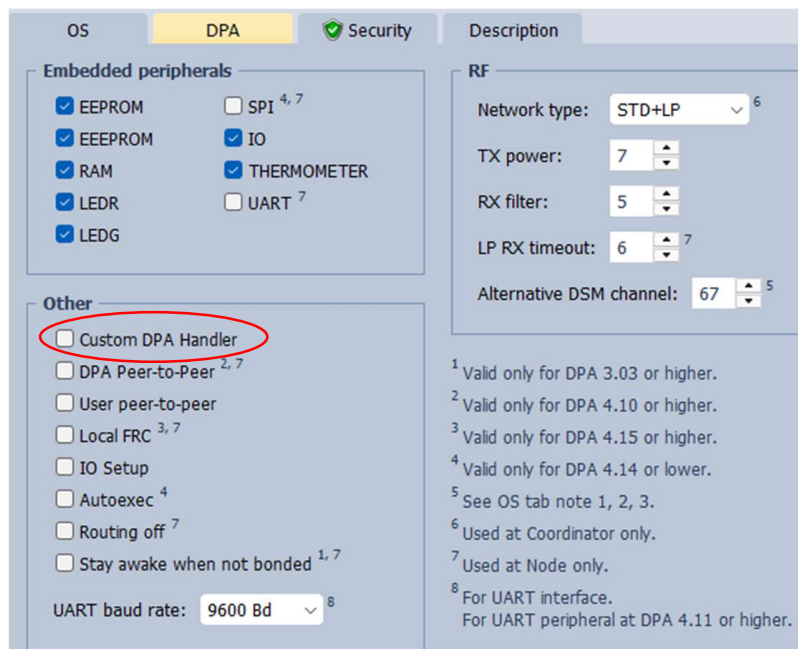
PLUG-IN JE POMĚRNĚ JEDNODUCHÝ, STAČÍ POUZE VYBRAT A NAHRÁT.

TR CONFIGURATION BUDEME MUSET UPRAVIT.



OS – NASTAVENÍ RF PÁSM A KOMUNIKAČNÍHO KANÁLU.

! V RÁMCI SÍTĚ MUSÍ BÝT HODNOTY TOTOŽNÉ, JINAK SE TR NEDOMLUVÍ.



NASTAVENÍ SEKCE DPA V TUTO CHVÍLI POSTAČÍ BEZE ZMĚNY. POKUD BUDU POTŘEBA PRACOVAT S CUSTOM DPA HANDLERY, JE NUTNO JE POVOLIT V TÉTO ČÁSTI.

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

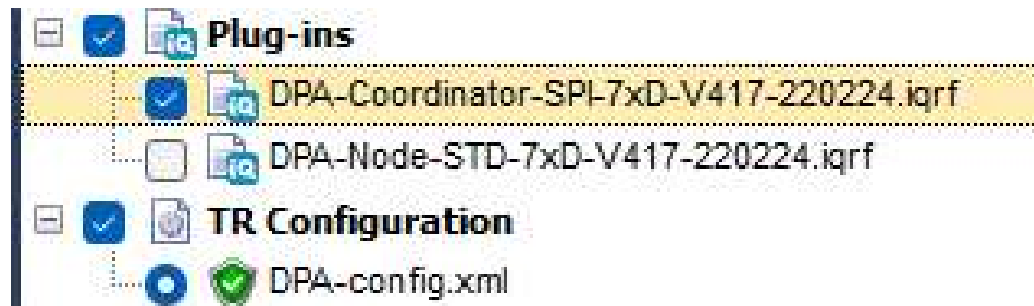
The screenshot shows a configuration window with four tabs: OS, DPA, Security, and Description. The Security tab is active, indicated by a green shield icon. Inside the Security tab, there is a section titled 'Access Password' with a checked checkbox. Below this, there is a 'Format' dropdown menu set to 'ASCII'. A 'Value' field contains a password represented by 10 dots. To the right of the password field is an eye icon with a small '1' next to it. Below the password field, there is a yellow progress bar and the text 'Password strength: Medium'. At the bottom right of the password field, the text '8/16' is displayed.

IQMESH síť VYŽADUJE, ABY BYLO NASTAVENÉ I ZABEZPEČENÍ, PROTO JE POTŘEBA NASTAVIT HESLO V ZÁLOŽCE SECURITY.

TENTO POSTUP SE ZOPAKUJE PRO VŠECHNY NODY V SÍTI.

VYTVÁŘENÍ KOORDINÁTORU

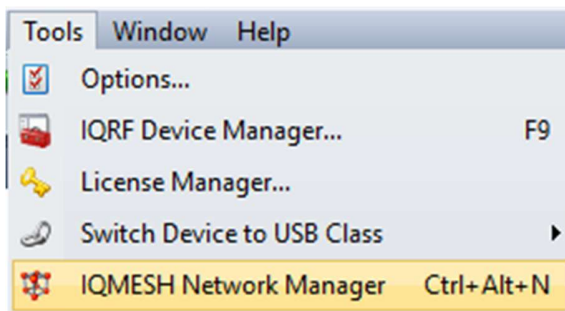
! TOTOŽNÝ POSTUP JAKO U NODU, JEDINÁ ZMĚNA V PLUG-INU.



TAKHLE MÁME PŘIPRAVENÝ HARDWARE PRO VYTVOŘENÍ MESH SÍTĚ.

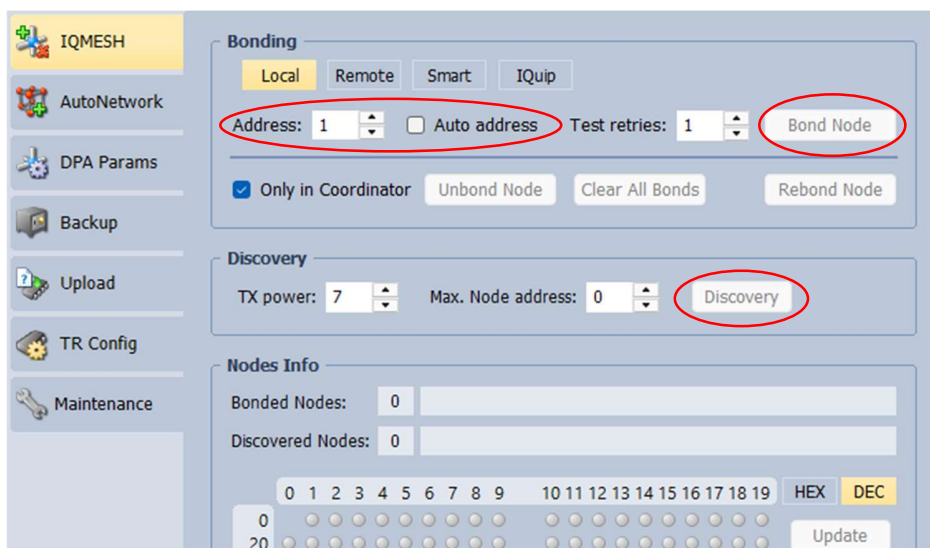
BONDING PROCES

ABY BYLO MOŽNÉ SE DOSTAT K VYTVOŘENÍ SÍTĚ, JE POTŘEBA OTEVŘÍT V IDE ZÁLOŽKU S NÁZVEM IQMESH NETWORK MANAGER NEBO OTEVŘÍT POMOCÍ KLÁVESOVÉ ZKRATKY.

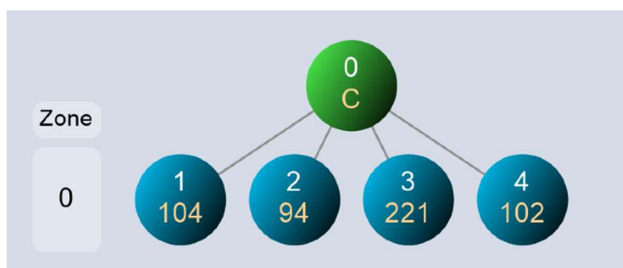


OTEVŘE SE MANAGER, KDE LZE PROVÉST BONDING

1. NASTAVENÍ ADRESY
2. KLIKnutí NA “BOND NODE”
3. STISKnutí UŽIVATELSKÉHO TLAČÍTKA NA NODU SE ZAŘÍZENÍ PŘIBONDUJE
4. OBJEVÍ SE INFORMACE O ÚSPĚŠNÉM PŘIBONDOVÁNÍ
5. KLIKnutíM NA DISCOVERY SE SPUSTÍ PROCES PROZKOUMÁNÍ NODŮ A PROBĚHNE VYTVOŘENÍ MAPY.



MAPA SÍTĚ V ZÁLOŽCE “MAP VIEW”

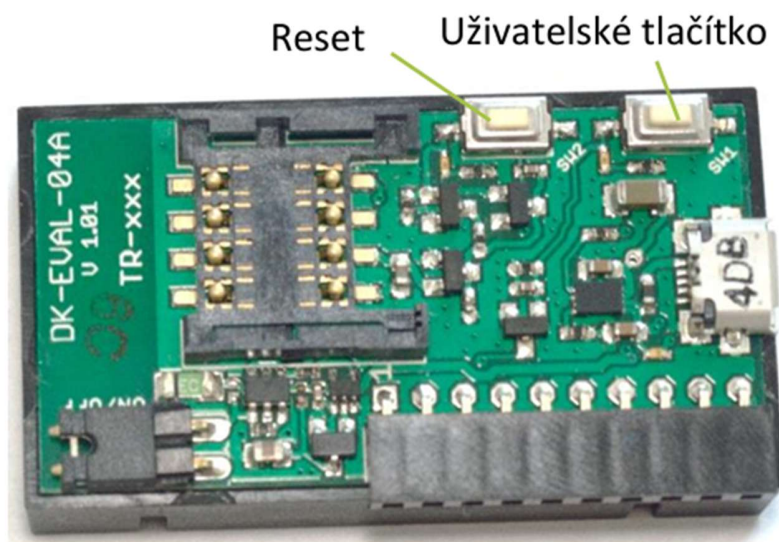


ODBONDOVÁNÍ

POKUD NASTANE SITUACE, ŽE NOD UŽ BYL PŘIBONDOVÁN K JINÉMU KOORDINÁTORU, MUSÍME HO ODBONDOVAT, ABY HO BYLO MOŽNÉ PŘIPOJIT K JINÉMU KOORDINÁTORU.

TO SE PROVEDE STISKNUTÍM A DRŽENÍM OBOU TLAČÍTEK SOUČASNĚ S NÁSLEDNÝM PUŠTĚNÍM TLAČÍTKA RESET. V TUTO CHVÍLI BY MĚLA SVÍTIT ZELENÁ LED, JAMILE ZHASNE, JE TŘEBA PUSTIT I UŽIVATELSKÉ TLAČÍTKO.

NA NODU BY MĚLA ZAČÍT BLIKAT ČERVENÁ LED, KTERÁ ZNAČÍ, ŽE JE NOD PŘIPRAVENÝ NA BONDOVÁNÍ.



PŘÍLOHA Č. 2: METODICKÝ LIST Č.2

NÁZEV LISTU	TVORBA IQRG GATEWAY
ZASAZENÍ DO VÝUKY	ELEKTRONICKÉ POČÍTAČE, IoT, INFORMAČNÍ TECHNOLOGIE
ZASAZENÍ DO SKUPINY	4. ROČNÍK SŠ, KONKRÉTNĚ SPŠ A VOŠ JIČÍN
PŘEDPOKLAD VYUŽITÍ METODICKÉHO LISTU	CVIČENÍ ZAMĚŘENÉ NA IQRG S VYUŽITÍM U PRAKTICKÉ MATURITNÍ ZKOUŠKY
CÍL VÝUKY	VYSVĚTLENÍ STUDENTŮM VYUŽITÍ GATEWAY V IQRG, PRÁCE S RASPBERRY PI, VSTUP DO PROTOKOLŮ SPOJENÝCH S IoT, MQTT
VSTUPNÍ ZNALOST	ŽÁK PROŠEL METODICKÝ LIST Č. 1. UMÍ PRACOVAT S IQRG A PROVÁDĚT PŘÍKAZY DPA, ZÁKLADNÍ ZNALOST LINUX
POPIS VÝUKY	TEORETICKÝ ÚVOD, UKÁZKA SESTAVENÍ, SAMOSTATNÁ PRÁCE ŽÁKŮ PO JEDNOM ČI SKUPINĚ PO DVOU (V ZÁVISLOSTI NA ZNALOSTI ŽÁKŮ ČI POČTU HARDWARU)
PŘÍPADNÉ PROBLÉM	POŠKOZENÝ HARDWARE, NODE PŘIPOJENÝ NA JINÝ KOORDINÁTOR, MOŽNÉ RF RUŠENÍ ŽÁKŮ NAVZÁJEM
PRAKTICKÁ ČÁST	SEZNÁMENÍ S IQRG GATEWAY NASTAVENÍ RASPBERRY PI INSTALACE SLUŽEB

IQRF GATEWAY

POPIS	ZÁKLADEM JE RASPBERRY PI, NA KTERÉ SE PŘIPOJÍ IQRF ADAPTÉR PRO BEZDRÁTOVOU KOMUNIKACI - BUDE TVOŘIT LOKÁLNÍ GATEWAY. NÁSLEDNĚ SE STÁHNE A NAINSTALUJE SLUŽBA MQTT PRO ZAJIŠTĚNÍ VÝMĚNY DAT MEZI ZAŘÍZENÍMI A SLUŽBA IQRF GATEWAY DAEMON. SPRÁVA A VIZUALIZACE DAT PROBÍHÁ SKRZE IQRF GATEWAY DAEMON WEBAPP A VÝVOJOVÉ PROSTŘEDÍ NODE-RED.
-------	---

PŘÍPRAVA HARDWARU

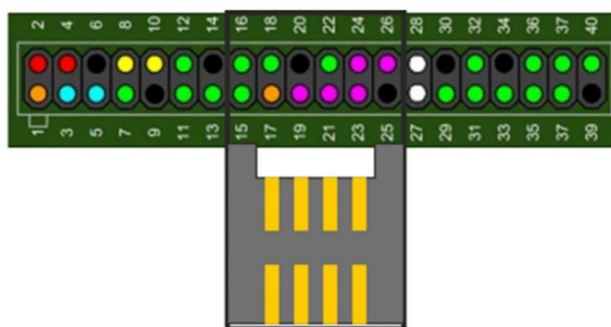


PŘEDTÍM NEŽ JE RPI ZAPNUTO(PŘIPOJENO KE ZDROJI), JE DOBRÉ MÍT VŠECHNO PŘIPOJENO, ABY DOŠLO KE SPRÁVNÉMU ZAVEDENÍ KOMPONENT.

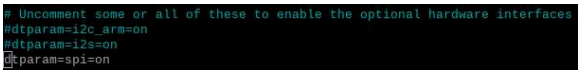
PRO PRÁCI JE VHDNÉ MÍT KLÁVESNICI, MYŠ A MONITOR.

ETHERNET PRO PŘIPOJENÍ K INTERNETU.

ADAPTER PRO TR SE PŘIPOJÍ NA PINY PODLE PŘEDLOHY NÍŽE.
(RPI MUSÍ BÝT VYPNUTÉ PŘI VKLÁDÁNÍ!)



KONFIGURACE RASPBERRY

NASTAVENÍ UŽIVATELE + ZÁKLADNÍ NASTAVENÍ USER, PASSWORD	<i>USER=PI, PASSWORD=RASPBERRY - VÝCHOZÍ</i> <i>USER=IQRF, PASSWORD=IQRF</i>
AKTUALIZACE SYSTÉMU	<i>sudo apt-get update && sudo apt-get -y full-upgrade</i>
POVOLENÍ SPI V KONFIGURAČNÍM SOUBORU JE POTŘEBA NAJÍT TENTO ŘÁDEK A ODKOMENTOVAT HO, ULOŽIT KONFIGURAČNÍ SOUBOR	<i>sudo nano /boot/config.txt</i> <i>"#DTPARAM=SPI=ON"</i> 
RESTARTOVAT RPI PRO PROJEVENÍ ZMĚN SE MUSÍ RASPBERRY RESTARTOVAT	<i>sudo reboot</i>

INSTALACE SLUŽEB

MQTT - INSTALACE MQTT BROKERU	<i>sudo apt-get install -y mosquitto mosquitto-clients</i>
INSTALACE SAMOTNÉ SLUŽBY TEDY ZAJIŠŤUJE FUNKCI MQTT SERVERU. PO INSTALACI SE BROKER SPUSTÍ AUTOMATICKY JAKO SYSTÉMOVÁ SLUŽBA. SLOUŽÍ JAKO PROSTŘEDNÍK MEZI KLIENTY, KTERÍ POSÍLAJÍ NEBO PŘIJÍMAJÍ ZPRÁVY.	

TEST SLUŽBY, ZDA INSTALACE PROBĚHLA SPRÁVNĚ → *SYSTEMCTL STATUS MOSQUITTO.SERVICE*

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

IQRF GATEWAY
DAEMON

```
sudo apt-get install -y dirmngr apt-transport-https lsb-release ca-certificates
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys
9C076FCC7AB8F2E43C2AB0E73241B9B7B4BD8F8E
sudo apt-get install dirmngr curl
sudo curl -sSLo /usr/share/keyrings/iqrf.gpg https://repos.iqrf.org/apt.gpg
echo "deb [signed-by=/usr/share/keyrings/iqrf.gpg] https://repos.iqrf.org/debian
bullseye stable" | sudo tee /etc/apt/sources.list.d/iqrf.list
sudo curl -sSLo /usr/share/keyrings/deb.sury.org-php.gpg
https://packages.sury.org/php/apt.gpg
echo "deb [signed-by=/usr/share/keyrings/deb.sury.org-php.gpg]
https://packages.sury.org/php/ bullseye main" | sudo tee
/etc/apt/sources.list.d/php.list
sudo apt-get update && sudo apt-get install -y iqrf-gateway-daemon
```

SOFTWAREOVÁ SLUŽBA DAEMON JE URČENÁ PRO BĚH NA LINUXU. ZAJIŠŤUJE MNOHO FUNKCÍ MEZI KTERÉ PATŘÍ KOMUNIKACE SE ZAŘÍZENÍMI IQRF PŘES SPI ROZHRANÍ, SPRÁVU IQRF SÍTĚ, PŘENOS DAT MEZI IQRF SÍTÍ A JINÝMI SYSTÉMY POMOCÍ MQTT NEBO PODPORA DPA PROTOKOLU.

IQRF Gateway package repositories

Please select your distribution

☒ Debian ☐ Raspberry Pi OS ☐ Ubuntu

Please select your version

☒ 11 Bullseye ☐ 12 Bookworm

Please select desired stability

☒ Stable ☐ Testing ☐ Development

Copy to clipboard

```
1 sudo apt-get install dirmngr curl
2 sudo curl -sSLo /usr/share/keyrings/iqrf.gpg https://repos.iqrf.org/apt.gpg
3 echo "deb [signed-by=/usr/share/keyrings/iqrf.gpg] https://repos.iqrf.org/debian bullseye stable" | sudo tee /etc/apt/sources.list.d/iqrf.
4 sudo curl -sSLo /usr/share/keyrings/deb.sury.org-php.gpg https://packages.sury.org/php/apt.gpg
5 echo "deb [signed-by=/usr/share/keyrings/deb.sury.org-php.gpg] https://packages.sury.org/php/ bullseye main" | sudo tee /etc/apt/sources.l
6 sudo apt-get update
```

Vysvětlit

[IQRF GATEWAY PACKAGE REPOSITORIES](#)

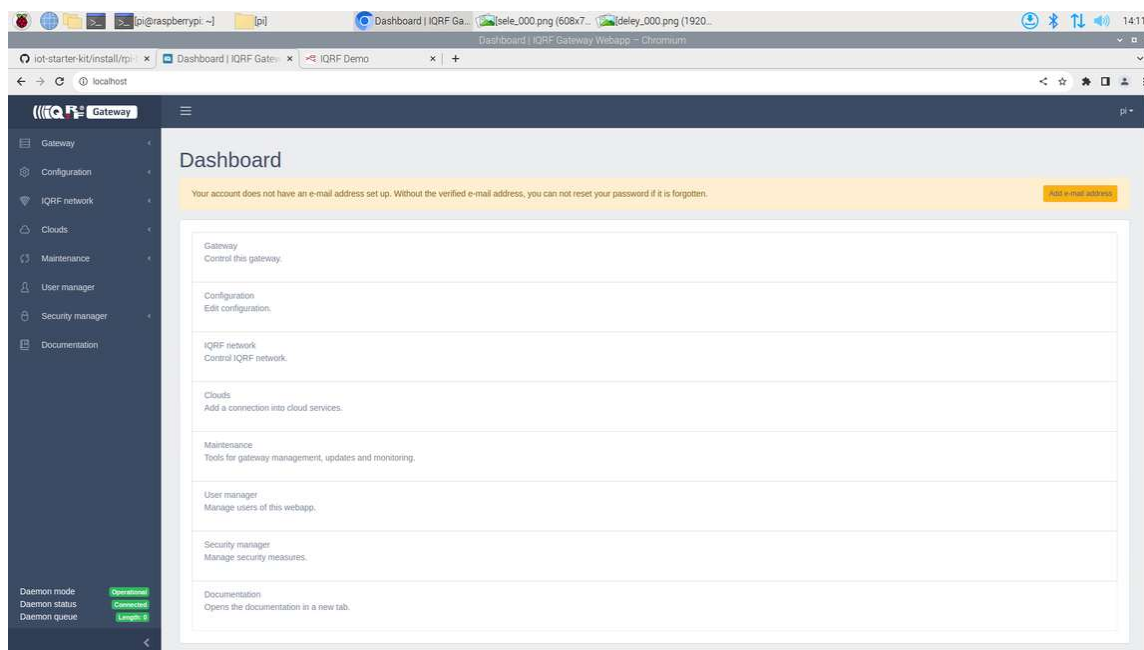
TEST SLUŽBY, ZDA INSTALACE PROBĚHLA SPRÁVNĚ → `SYSTEMCTL STATUS IQRF-GATEWAY-DAEMON.SERVICE`

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

IQRF GATEWAY DAEMON WEBAPP	<code>sudo apt-get install -y iqrif-gateway-webapp</code>
WEBOVÉ UŽIVATELSKÉ ROZHRANÍ URČENÉ PRO SPRÁVU SÍTĚ, KONFIGURACI GATEWAYE, DIAGNOSTIKU A MONITOROVÁNÍ. HLAVNÍ FUNKCÍ TÉTO APLIKACE JE KONFIGURACE BRÁNY, KTERÁ UMOŽŇUJE DEFINOVAT MAPOVÁNÍ MEZI TRANSCEIVEREM A KONTROLÉREM. PŘÍZPŮSOBUJE KOMUNIKACI POMOCÍ PARAMETRŮ SPI A UART.	
PŘÍSTUPNÁ NA HTTP://LOCALHOST NEBO HTTP://127.0.0.1 .	

KONFIGURACE IQRF GATEWAY DAEMON WEBAPP

1. VYTVOŘIT UŽIVATELE, PŘESKOČIT NASTAVENÍ SMTP SERVERU
2. NASTAVIT KOMUNIKAČNÍ ROZHRANÍ NA SPRÁVNĚ GPIO PINY
3. RESTART SLUŽBY PRO PROJEVENÍ ZMĚN
4. OVĚŘENÍ FUNKČNOSTI SLUŽBY - KOMUNIKACE POMOCÍ DPA PŘÍKAZU
5. SPRÁVA SÍTĚ SKRZE NETWORK MANAGER



ÚVODNÍ WEBOVÉ PROSTŘEDÍ

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100



KONFIGURACE IQRF SPI INTERFACE

V NABÍDCE NAPRAVO, V SEKCI “CONFIGURATION” SE NACHÁZÍ POLOŽKA INTERFACES, VE KTERÉ SE NAKONFIGURUJE SPI INTERFACE.



NÍŽE V AVAILABLE SPI LZE NALÉZT PŘEDPŘIPRAVENÉ “MAKRO” PRO ADAPTER NA RPI. VÝBĚR “SET MAPPING” NASTAVÍ SPI ROZHRANÍ.

☒ Enable SPI reset

Power enable GPIO pin: 23

Bus enable GPIO pin: 0 ✓

Programming mode switch GPIO pin: -1 ✓

☐ Use communication interface pins

I2C interface enable GPIO pin: [empty]

SPI interface enable GPIO pin: [empty]

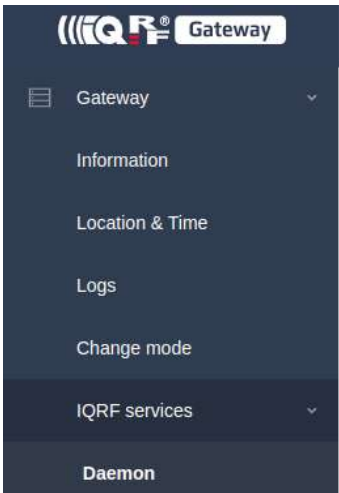
UART interface enable GPIO pin: [empty]

Available SPI mappings:

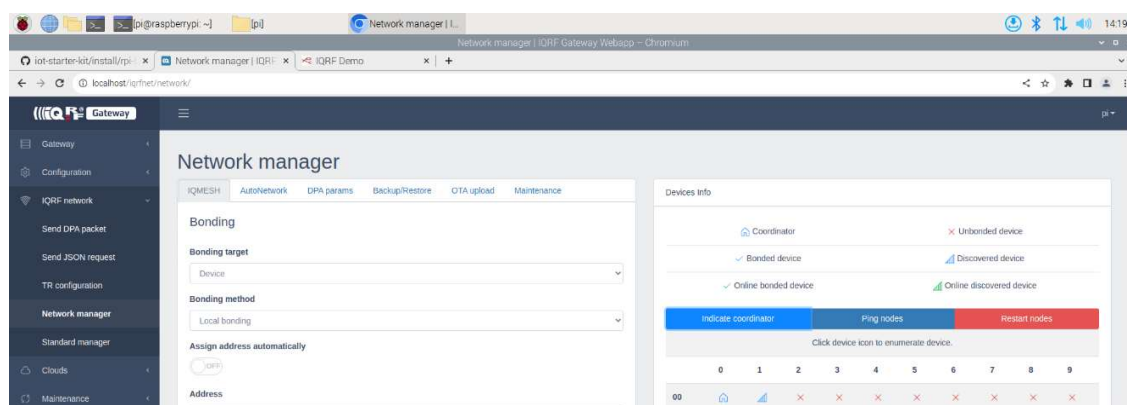
-
- KON-RASP-01

TLAČÍTKEM “SAVE” ULOŽIT A NYNÍ JE POTŘEBA RESTART CELÉ SLUŽBY.

Vyšší odborná škola a Střední průmyslová škola,
Jičín, Pod Koželuhy 100

	HNEJ V PRVNÍ ČÁSTI NABÍDKY V ZÁLOŽCE IQRf SERVICES JE POLOŽKA DAEMON. Daemon Disable Stop Restart Refresh status Service status: Enabled, active RESTART.
---	---

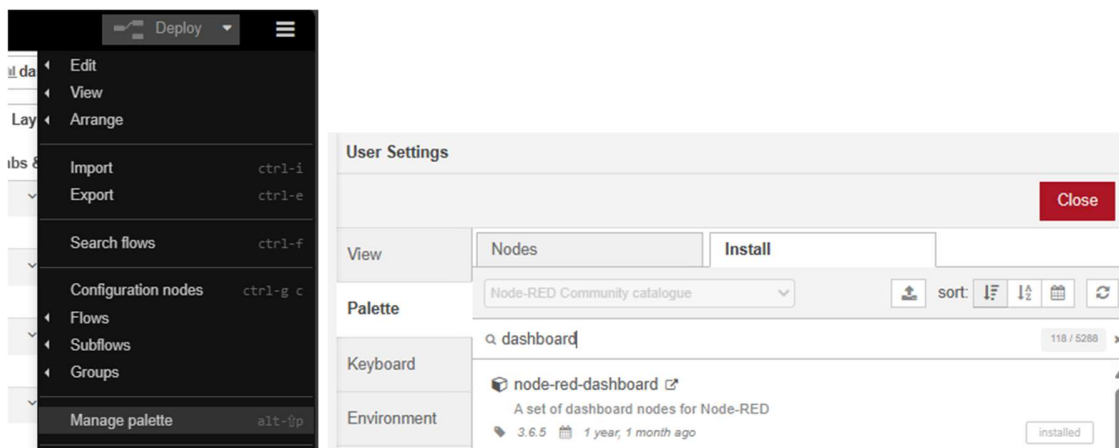
OVĚŘENÍ FUNKČNOSTI LZE ZASLÁNÍM DPA PŘÍKAZU NEBO SAMOTNÉ ZOBRAZENÍ SÍTĚ V NETWORK MANAGER.



NODE-RED	<pre>sudo apt-get install -y curl bash <(curl -sL https://raw.githubusercontent.com/node- red/linux-installers/master/deb/update-nodejs- and-nodered)</pre>
POVOLENÍ SLUŽBY	<pre>sudo systemctl enable nodered.service sudo systemctl start nodered.service</pre>
PŘÍSTUPNÝ NA HTTP://LOCALHOST:1880 .	

INSTALACE DASHBOARD DO NODE-RED

VE VÝCHOZÍM STAVU NODE-RED NEOBSAHUJE STAŽENÉ BALÍČKY PRO DASHBOARD. JE POTŘEBA JE TEDY NAJÍT V PALETĚ A NAINSTALOVAT.



PO INSTALACI JE DASHBOARD PŘÍSTUPNÝ PŘES ADRESU [HTTP://LOCALHOST:1880/UI](http://localhost:1880/ui). V TUTO CHVÍLI ZDE NIC NEBUDE.

KLONOVÁNÍ STARTER-KITU, KTERÝ USNADNÍ PRÁCI S NODE-RED V PROJEKTU

```
cd /home/pi
git clone https://gitlab.iqrf.org/alliance/iot-starter-kit.git
cd iot-starter-kit/install
cp rpi-board/node-red/* /home/pi/.node-red
sudo systemctl restart nodered.service
```

! POZOR, JAK SE JMENUJE DOMOVSKÁ SLOŽKA. POKUD SE USER JMENUJE JINAK, JE POTŘEBA UPRAVIT.

KONTROLA, ZDA SLUŽBA BĚŽÍ SPRÁVNĚ → `SYSTEMCTL STATUS NODERED.SERVICE`

VÝSLEDEK → [HTTP://LOCALHOST:1880/UI](http://localhost:1880/ui)

