

Vyšší odborná škola a Střední průmyslová škola elektrotechnická Olomouc, Božetěchova 3

**PRAKTICKÁ ZKOUŠKA   
Z  ODBORNÝCH PŘEDMĚTŮ**

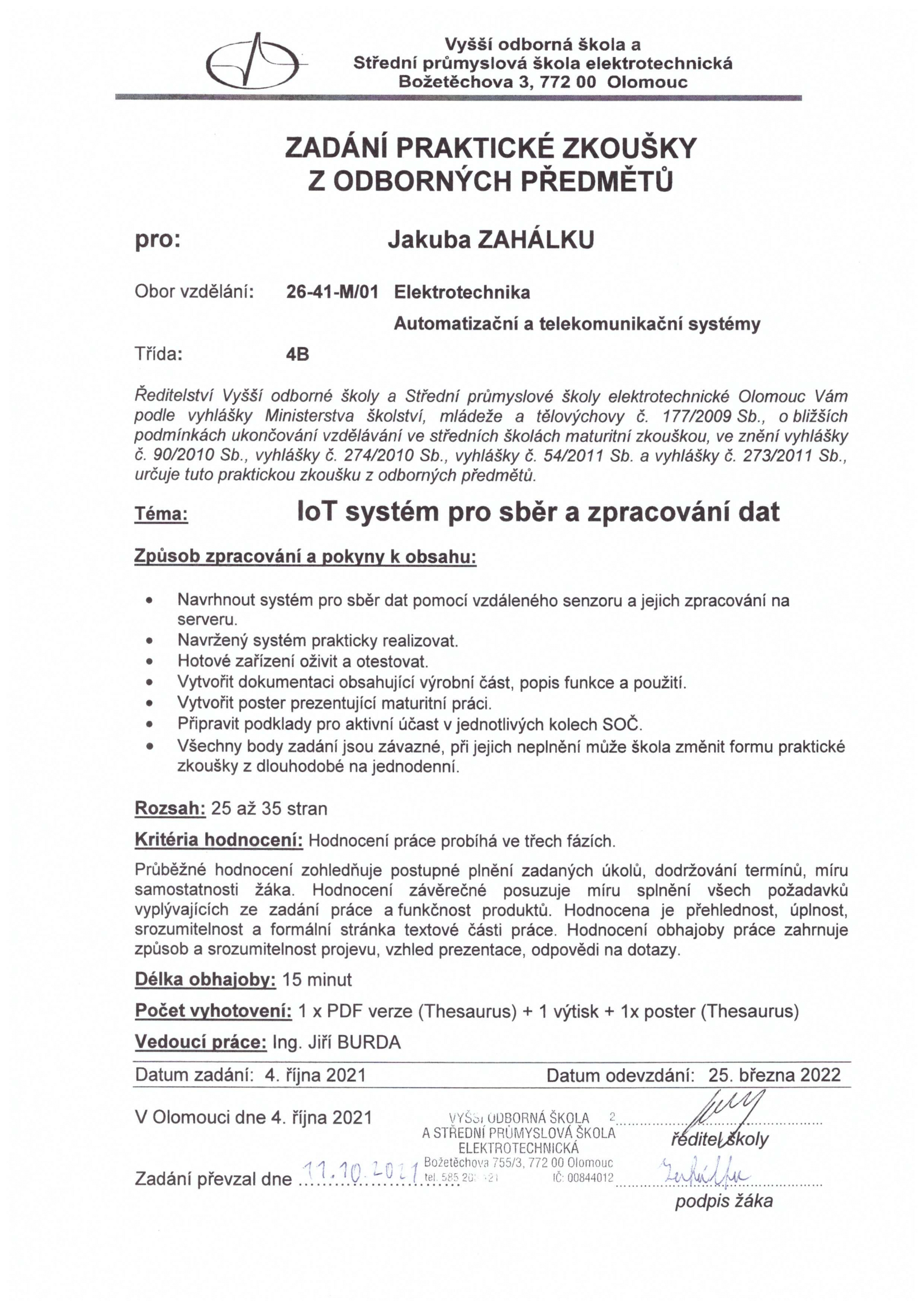
IoT systém pro sběr a zpracování dat

Autor **Jakub Zahálka**

Obor **Elektrotechnika**

Vedoucí práce **Ing. Jiří Burda**

Školní rok **2021/2022**



Prohlašuji, že jsem praktickou zkoušku z odborných předmětů vypracoval samostatně a všechny prameny jsem uvedl v seznamu použité literatury.

……………………………  
 Jakub Zahálka

Chtěl bych vyslovit poděkování panu Ing. Jiřímu Burdovi za odborné konzultace a poskytnuté informace.

……………………………  
 Jakub Zahálka

Prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé práce nebo její části se souhlasem školy.

……………………………  
 Jakub Zahálka

Abstrakt

Tato práce se zabývá návrhem a konstrukcí systému určenému k sběru a zpracování dat. Tento projekt byl započat z nutnosti měřit průtok vody při pronájmu tělocvičny a při odhalování úniku vody. Celý projekt se v základu zabýval jen o měření vody pomocí bezkontaktního senzoru, ale postupem času byl zobecněn a přizpůsoben k dalším formám vstupu z různých čidel. Finální zařízení je postaveno   
na mikrokontroleru ESP32S2 z důvodu WIFI konektivity. ESP32S2 posílá změřené hodnoty periodicky na server přes protokol MQTT, kde jsou informace zpracovány a přidány do databáze. Graf je poté z těchto dat vykreslen na webu. Jednou   
z možností vstupu je 5 analogově/digitálních vstupů. Na horní lištu jsou vyvedeny 2 piny přímo z mikrokontroleru. Jako vstup je možné využít také 2 proudových smyček 4 – 20 mA. Poslední typ vstupu jsou protokoly Mbus a RS485. Mbus je složen z 2 zenerových diod a 4 tranzistorů. RS485 je zprostředkován integrovaným obvodem MAX485 doplněným o několik pasivních součástek. Jako výstup je možné použít 4 relátka s svorkami COM, NC, NO nebo také 3 NPN mosfet výstupy. Na boční střaně sestavy je rozšiřující kontektor pro připojení různých modulů. Rozšiřující konektor má na sobě vyvedené programovací rozhrani, I2C, 2 volné piny a různé větve napájení. Celé zařízení je budováno s myšlenkou rozšíření a univerzálnosti. Z tohoto důvodu se využívají standardní vstupy typické pro automatizační prostředí. Rozšiřitelnost systému je zajištěna protokolem MQTT. Za dobu fungování systému si sám sebe zaplatil a odhalil různé úniky vody mimo vyučovací hodiny. Díky nastavitelnému varovnému systému je možné upozornit na různé hodnoty   
a zprávu o nich odeslat na předem definovaný e-mail nebo Telegram. Využitelnost této dlouhodobé maturitní práce je v celé řadě automatizačních prostředí, ale také jako jádro chytrého domu.

Obsah

[Obsah 5](#_Toc98984381)

[Úvod 7](#_Toc98984382)

[1. Prototyp 8](#_Toc98984383)

[1. Návrh schématu 10](#_Toc98984384)

[1.1 Mikroprocesor 10](#_Toc98984385)

[1.1.1 Enable procesoru 11](#_Toc98984386)

[1.2 Napájení 11](#_Toc98984387)

[1.3 Vstupy a výstupy 13](#_Toc98984388)

[1.3.1 Napěťové děliče pro ADC 13](#_Toc98984389)

[1.3.2 Proudová smyčka 14](#_Toc98984390)

[1.3.3 RS485 15](#_Toc98984391)

[1.3.4 Relé výstupy 16](#_Toc98984392)

[1.3.5 Mosfet výstupy 18](#_Toc98984393)

[1.3.6 433 MHz komunikace 19](#_Toc98984394)

[1.3.7 Meterbus 20](#_Toc98984395)

[2. Návrh DPS 21](#_Toc98984396)

[2.1 Rozložení součástek 22](#_Toc98984397)

[2.2 Návrh a pravidla pro kreslení 22](#_Toc98984398)

[2.3 Realizace a postup pájení DPS 24](#_Toc98984399)

[3. Tvorba krabičky 25](#_Toc98984400)

[3.1 3D render krabičky 25](#_Toc98984401)

[3.2 3D tisk 27](#_Toc98984402)

[4. Komunikace a vizualizace 29](#_Toc98984403)

[4.1 MQTT 29](#_Toc98984404)

[4.2 Bridge 29](#_Toc98984405)

[4.3 Databáze 29](#_Toc98984406)

[4.4 Grafana 30](#_Toc98984407)

[4.5 OTA 30](#_Toc98984408)

[5. Demo program 31](#_Toc98984409)

[5.1 Displej 31](#_Toc98984410)

[5.2 Bezdrátová komunikace 433MHz 31](#_Toc98984411)

[5.3 Protokol MQTT 31](#_Toc98984412)

[5.4 ArduinoOTA 31](#_Toc98984413)

[Závěr 33](#_Toc98984414)

[Seznam použité literatury 35](#_Toc98984415)

[Seznam použitých zkratek 36](#_Toc98984416)

[Seznam obrázků a tabulek 37](#_Toc98984417)

[Přílohy 39](#_Toc98984418)

Úvod

V této práci se budu zabývat tvorbou a návrhem kompletního zařízení určenému k dohledu nad různými systémy. Budu vytvářet vlastní desku plošných spojů osazenou procesorem ESP32S2. Samotná deska bude obsahovat analogově/digitální vstupy a relé/mosfet výstupy. Na desce budou obvody pro komunikaci přes různé protokoly. Pro komunikaci přes internet bude používán protokol MQTT přímo určený k tomuto typu aplikace. Snímaná data budou periodicky posílána na server, kde budou zapisována do databáze. Z této databáze bude čerpat Grafana, která z těchto dat vytvoří graf. Grafana se také postará   
o upozornění na havarijní stavy pomocí zprávy na Telegram. Dané téma jsem   
si zvolil kvůli nutnosti měřit průtok vody ve škole. Postupem času jsem se rozhodl pro větší univerzálnost celé sestavy a přidat další možnosti vstupů a rozšiřitelnosti na větší plochu působení.

# Prototyp

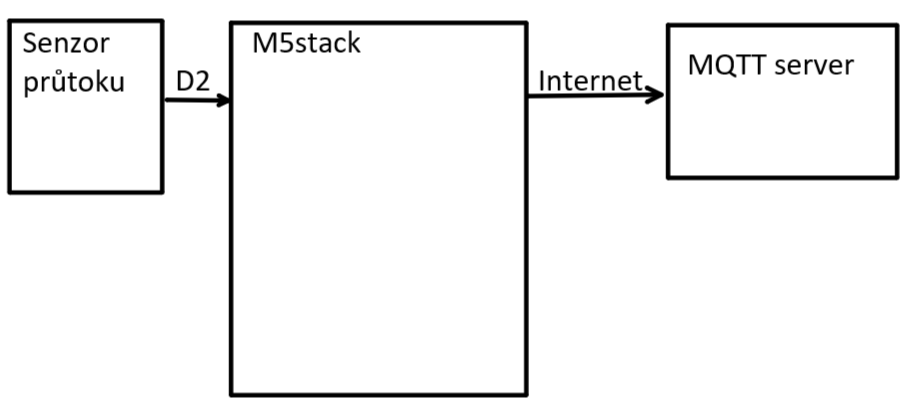
Původní prototyp byl tvořen vývojovou sada M5Stack. Samotné čidlo pro snímaní průtoku vody musí být kompatibilní s daným vodoměrem. Čidlo se pohybuje v cenové relaci do 1000 Kč a je schopné komunikace přes protokol Meterbus (dále jen Mbus) nebo pulzně přes stáhnutí měřícího pinu k zemi. Přes protokol Mbus   
je možné vyčíst i číslo vodoměru, ale k tomu je potřeba speciálního obvodu pro komunikaci přes Mbus. Data z měřiče jsou shromažďována a každých 10 vteřin odesílána na MQTT server. Zde jsou převzata programem, který je přesměruje   
a zapíše do databáze influx. Z této databáze jsou data vizualizována programem Grafana.

Obrázek 1 – Vývojová sada M5Stack Core



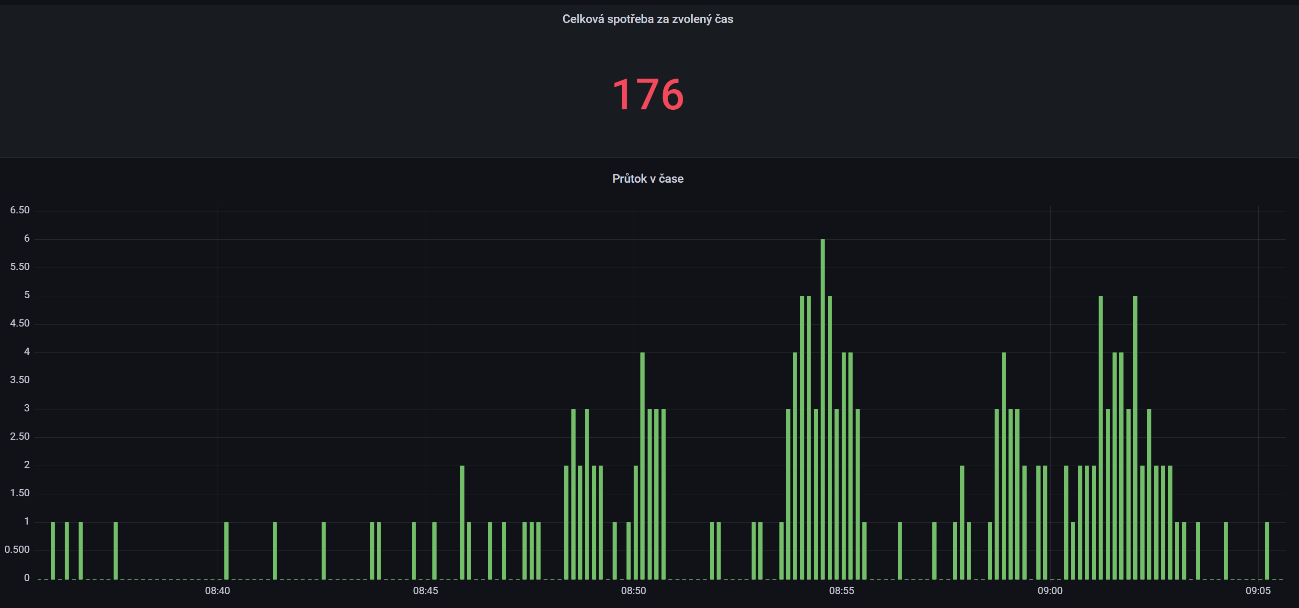
Zdroj: [https://www.distrelec.cz](https://www.distrelec.cz/)

Obrázek 2 - Blokové schéma prototypu



Zdroj: Vlastní zpracování

Obrázek 3 - Ukázka grafu průtoku vody



Zdroj: Vlastní zpracování

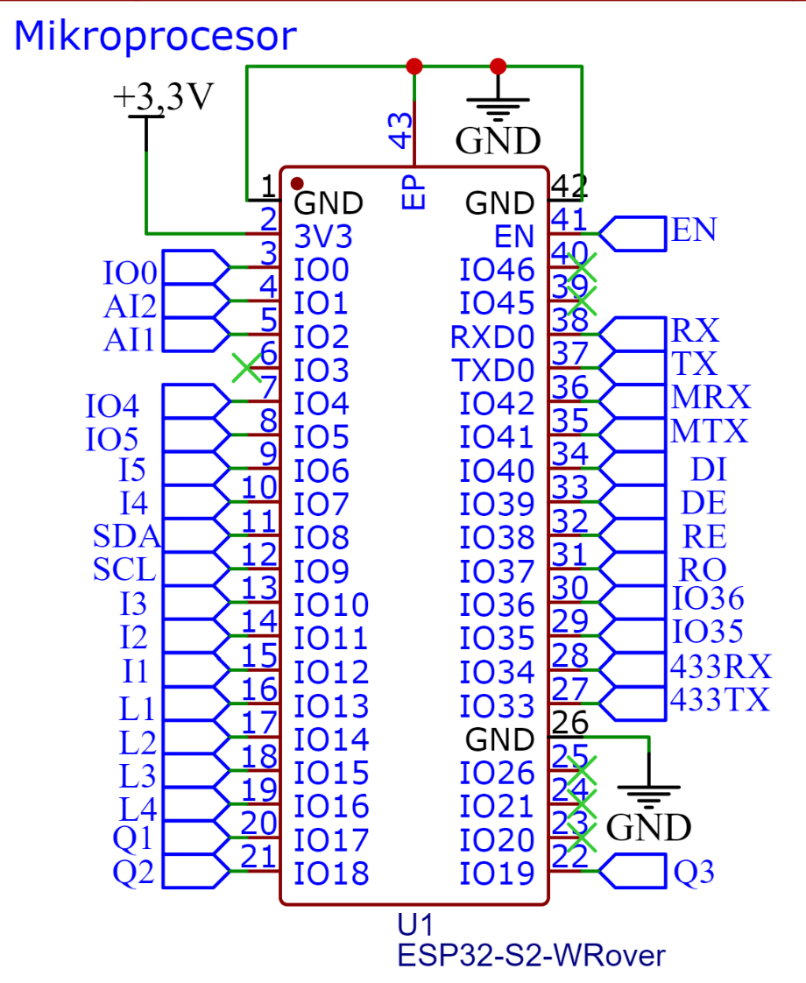
# Návrh schématu

Před tvorbou desky plošných spojů je nutno nejdříve vytvořit schéma zapojení. Zvolený procesor pro tuto aplikaci je ESP32S2 Wrover od firmy Espressif. Tento procesor je kompletní balíček všech potřebných součástek a pro základní zapojení stačí jen 3,3V napájení. Nejdříve jsem si našel veškeré potřebné součástky na webu TME a poté jsem začal s kreslením schématu. Kompletní schéma naleznete v   
[Příloze č. 2](#_Toc98982905).

## Mikroprocesor

Pro základní funkčnost procesoru potřebujeme připojit napájení 3,3V a pin EN(enable). Pin EN slouží jako resetovací pin procesoru a je důležitý v případě potřeby restartu i samotného startu. Na jednotlivé GPIO piny jsou napojeny veškeré vstupní či výstupní periferie. Některé piny byly ponechány volné kvůli špatné dostupnosti na desce nebo neměly využití.

Obrázek 4 - Schéma zapojení mikroprocesoru

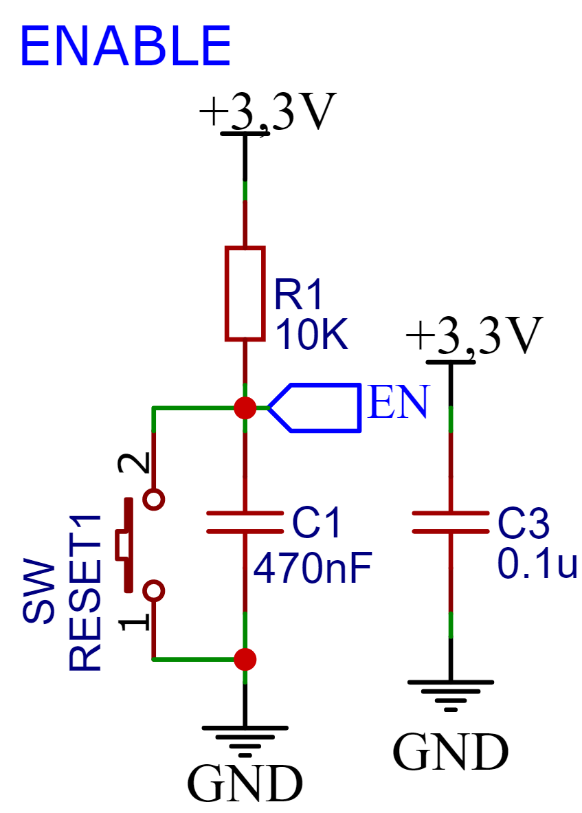


Zdroj: Vlastní zpracování

### Enable procesoru

Procesor má takzvané bootstrap piny, které mu určují různé startovací podmínky. Pin EN je připojen 10k rezistorem na 3,3V a bez něj by procesor nenastartoval. Kondenzátor C1 slouží pro programování procesoru. V případě absence tohoto kondenzátoru je potřeba při každém programování ESP vyresetovat ručně. Kondenzátor C3 je vyrovnávací kondenzátor, který zmírňuje krátké propady v napájení procesoru při vypínání nebo zapínání vnitřních tranzistorů.

Obrázek 5 - Schéma doprovodných součástek pro CPU

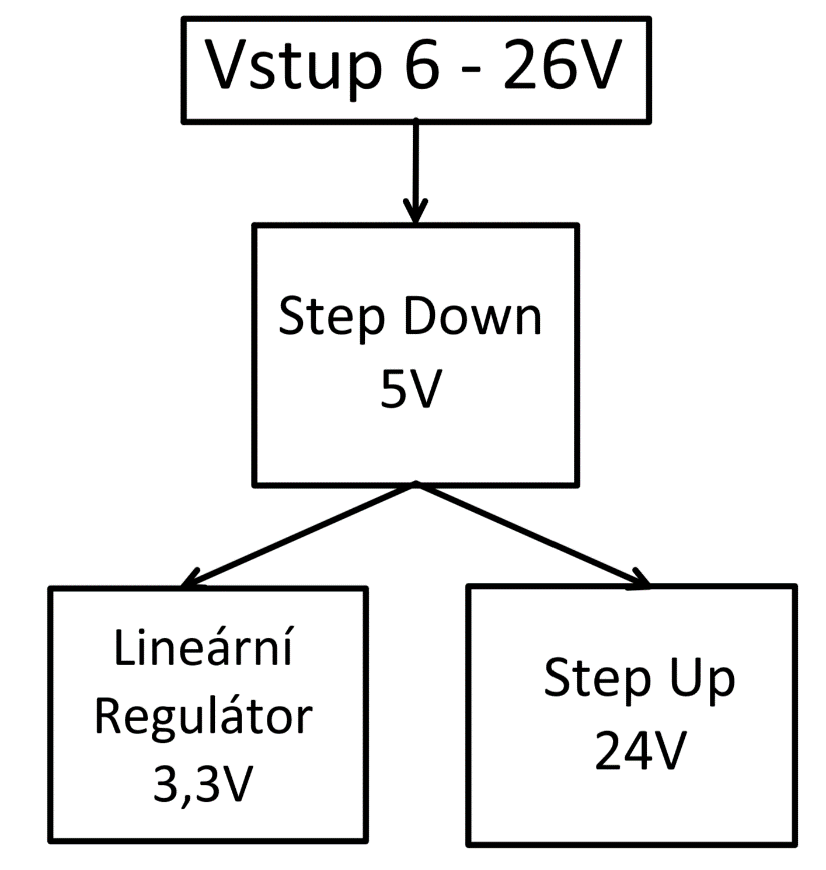


Zdroj: Vlastní zpracování

## Napájení

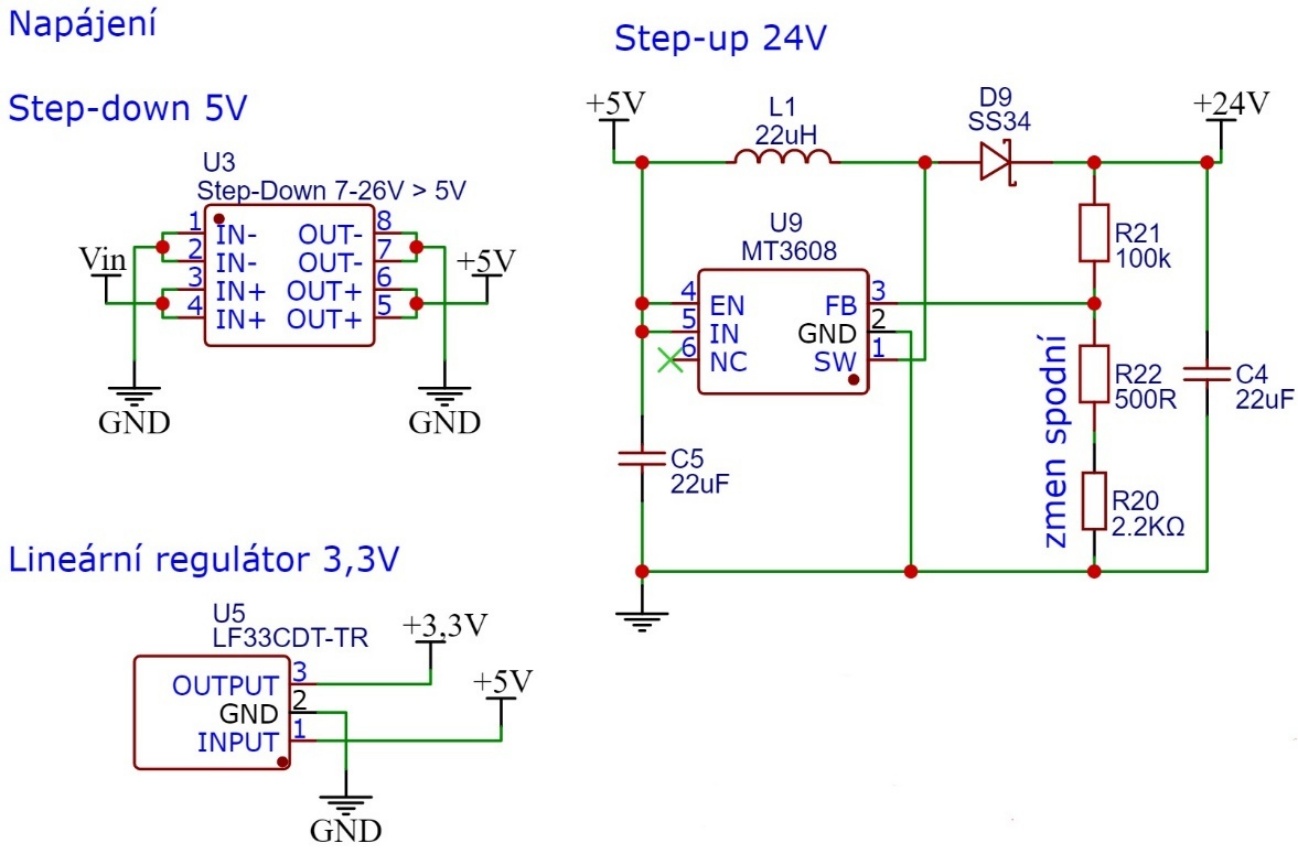
Dalším důležitým prvkem je samotné napájení. Maximální napájecí napětí celého systému je 26V. Kvůli nutnosti různých napájecích napětí pro různé komponenty   
se zde nachází celkem 3 různé napájecí obvody. Na vstupu je připojen 5V step-Down regulátor realizovaný čínským modulem. Na 5V větev je poté napojen lineární regulátor, který obstará 3,3V pro procesor. Obvod pro komunikaci Mbus potřebuje více jak 21V pro jeho správnou funkčnost, a proto jsem zvolil napětí 24V, které   
je často používaným napětím v automatizaci a je možné jej využít pro další účely.

Obrázek 6 - Blokové schéma napájení



Zdroj: Vlastní zpracování

Obrázek 7 - Schéma napájecího systému

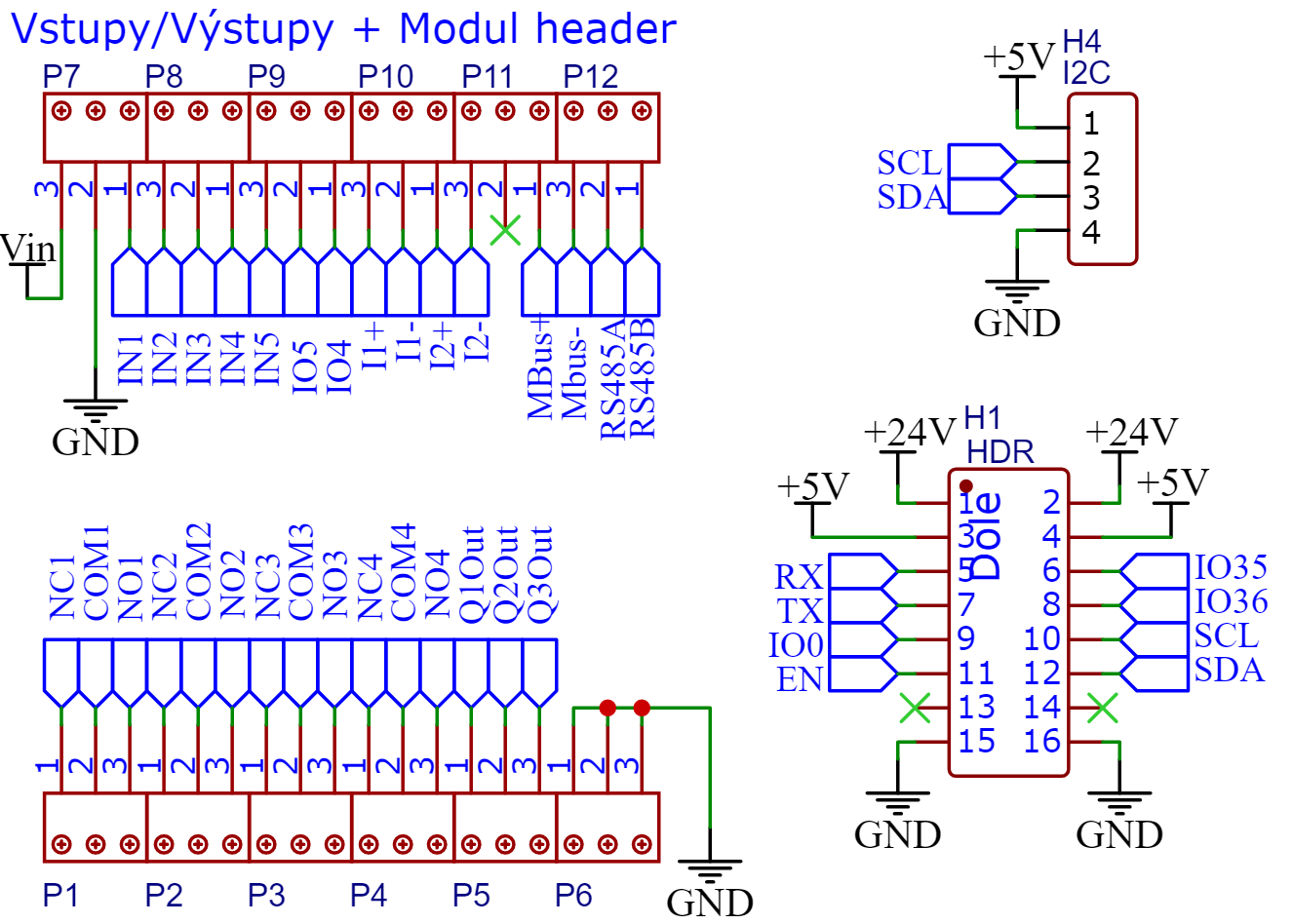


Zdroj: Vlastní zpracování

## Vstupy a výstupy

Vstupy a výstupy jsou realizovaný šroubovými terminály do desky plošných spojů   
a rozšiřujícím konektorem na boční straně. Nahoře se nachází vstupní periferie   
a na dolní straně se nachází výstupy. Na boku se nachází konektor pro rozšiřující moduly.

Obrázek 8 - Schéma rozložení jednotlivých vstupů/výstupů + boční konektor pro moduly



Zdroj: Vlastní zpracování

### Napěťové děliče pro ADC

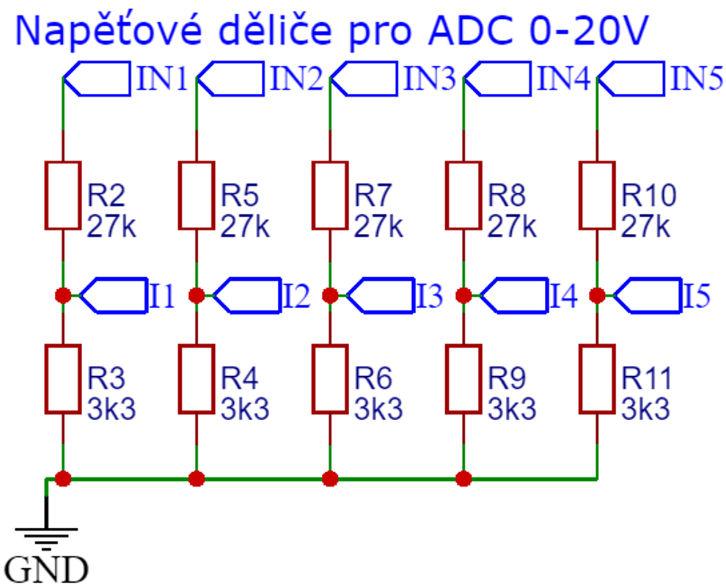
Zařízení disponuje 5 analogově/digitálními vstupy, které obsahují napěťový dělič pro rozšíření měřitelné oblasti. Při plném rozsahu se na dolním rezistoru ukáže 2,5V, což je maximální měřitelné analogové napětí přímo vestavěného ADC. Tyto vstupy lze použít i jako digitální vstupy. Maximální roszah v mé konfiguraci je 23V.

(1)

(2)

Do vzorce (1) dosadím maximální měřitelné napětí ADC za U. Za R dosadím hodnotu zvoleného rezistoru. Výsledkem vzorce (1) je celkový proud procházející napěťovým děličem v plném rozsahu. Druhá rovnice se dá použít na výpočet maximálního rozsahu napěťového děliče. Za I dosadím dříve vypočítaný celkový proud. Za poslední parametr R dosadím celkový odopor děliče.

Obrázek 9 - Napětově děliče pro ADC





Zdroj: Vlastní zpracování

### Proudová smyčka

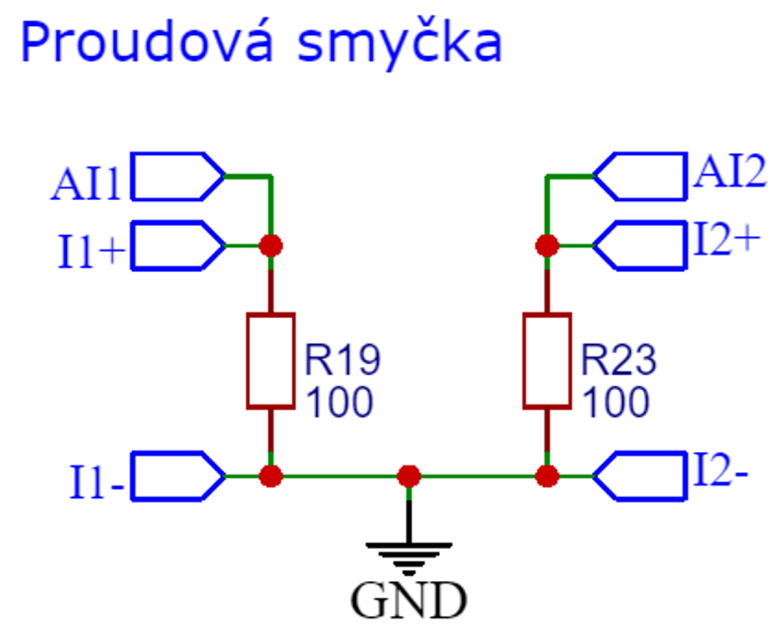
Zařízení obsahuje jednoduchou proudovou smyčku 4-20mA. Rezistory mají hodnotu 100Ω a při proudu 20 mA to odpovídá 2V. Toto napětí je potom měřeno ADC.

(1)

(2)

Vzorec (1) jsem použil pro vypočítání hodnoty rezistorů pro proudovou smyčku.   
Za U jsem dosadil mnou zvolené maximální napětí 2V. Za proud jsem zvolil maximální proud senzorů čili 20mA. Druhý vzorec se využívá již při samotném výpočtu reálné hodnoty proudu.

Obrázek 10 - Schéma proudové smyčky

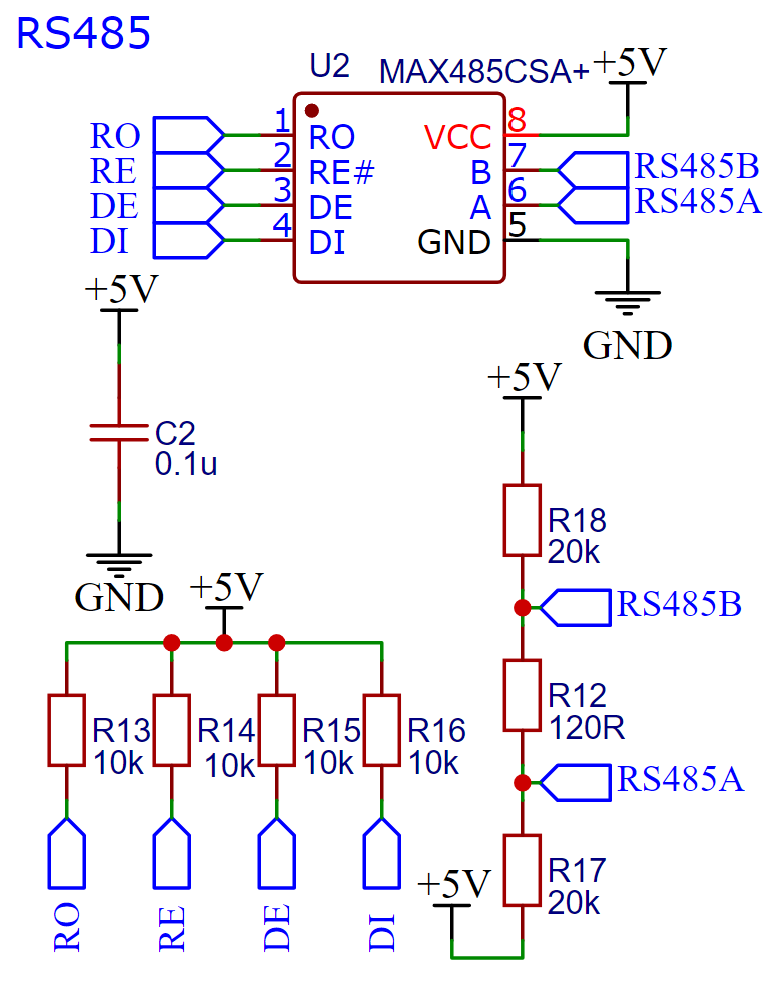


Zdroj:Vlastní zpracování

### RS485

Dalším komunikačním protokolem je protokol RS485, který je zprostředkován integrovaným obvodem MAX485CSA+. V základní dvou vodičové verzi je komunikace poloduplexní a schopna dosáhnout vzdálenosti 1,2km bez opakovače. Maximální počet zařízení je 32 a v jednu chvíli může vysílat jen jedno zařízení.   
Je přímo navržen pro kroucenou dvojlinku. Na vodiče RS485A a RS485B je zapojen rezistor R12 o hodnotě 120Ω a rezistory 20k R17,R18 jako pull- up celého vedení. Tyto rezistory jsou potřeba pro samotnou komunikaci. Vstup DI a výstupy RO,RE,DE jsou připojeny na procesor a mají svůj 10k pull-up rezistor. V testování jsem zjistil, že tam nejsou potřeba, a proto jsem je na finálním výrobku neosadil.

Obrázek 11 - Schéma obvodu pro komunikaci RS485

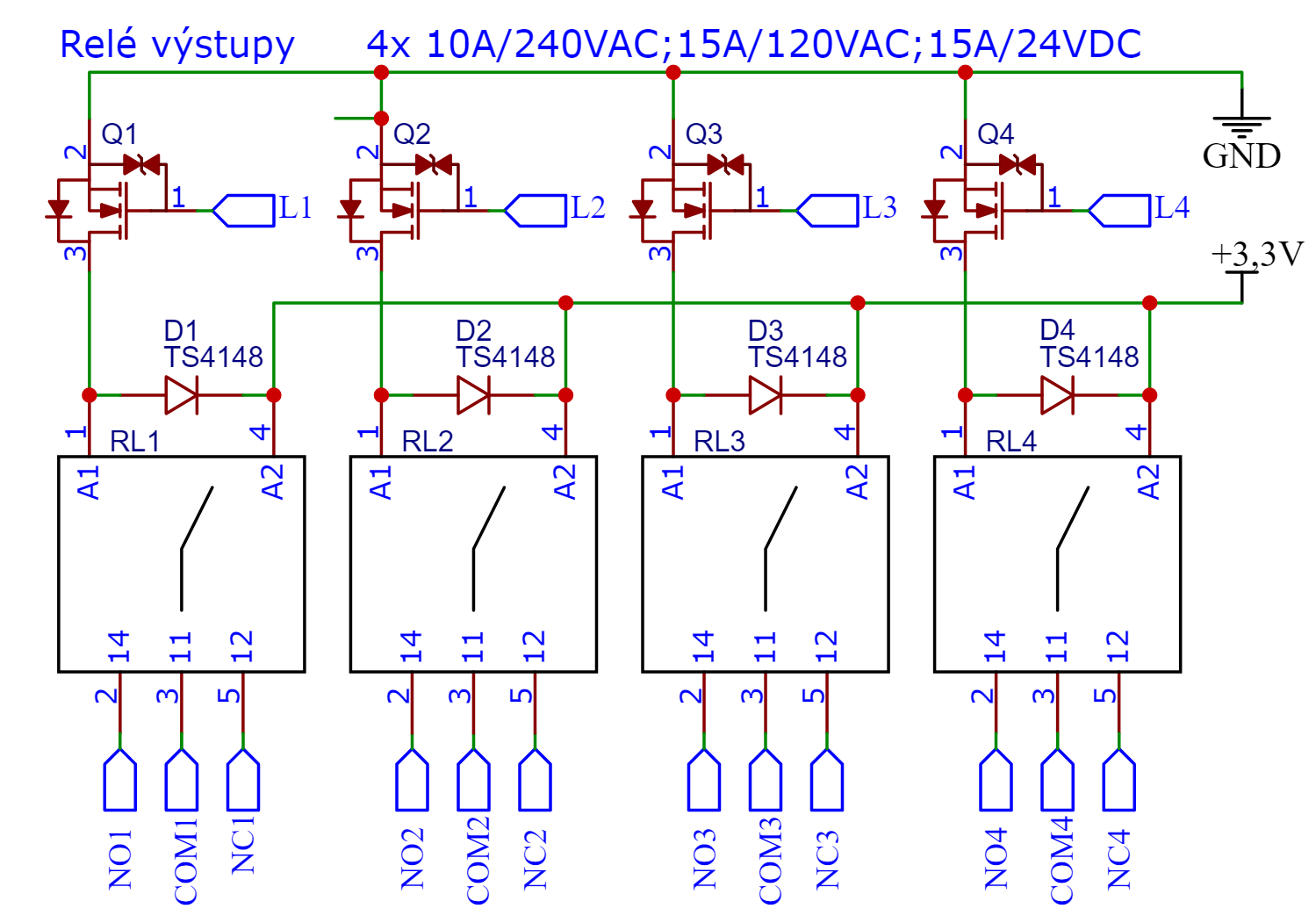


Zdroj: Vlastní zpracování

### Relé výstupy

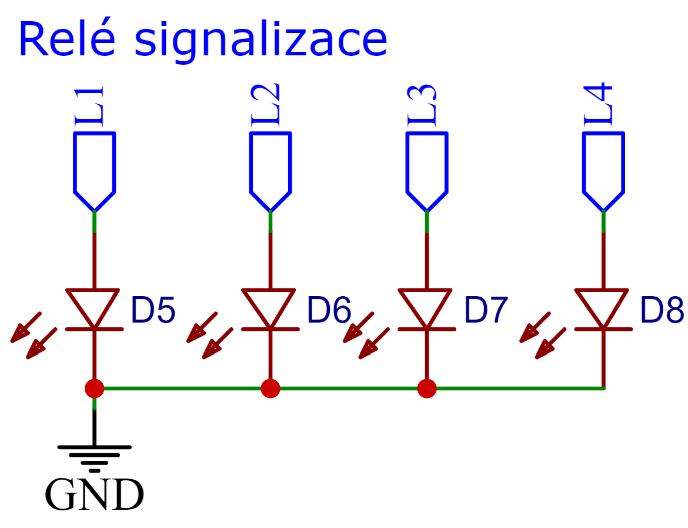
Relé jsou připojeny na 3,3V větev a jsou spínány NPN tranzistorem na zem.   
Na signální vodič je připojena i LED dioda, která slouží k signalizaci. Na cívku relé jsou napojeny spínací diody v závěrném směru. Jsou zde z důvodu zpětného rázu   
při vybíjení cívky. Relé má vyvedeny 3 spínací vývody. Je zde dostupná svorka COM, NC a NO. Vývod NC (normally closed) je v klidovém stavu sepnutý s COM, ale jakmile zaktivujeme relé, tak se kontak rozpojí. Kontakt NO (normally open) je v klidovém stavu rozepnutý, ale v aktivním stavu se spojí COM.

Obrázek 12 - Schéma zapojení relé výstupů



Zdroj: Vlastní zpracování

Obrázek 13 - Schéma zapojení relé signalizace



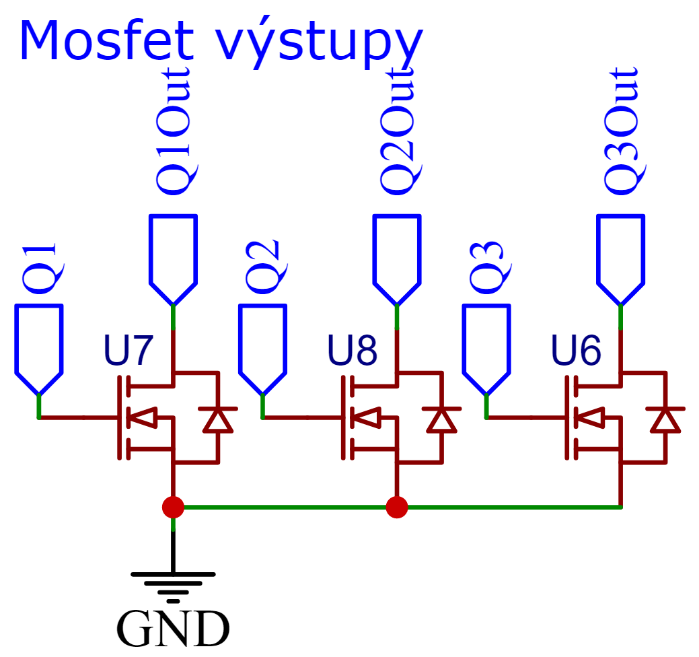
Zdroj: Vlastní zpracování

Signalizační diody mají větší pracovní napětí než je logická úroveň ESP32, proto   
je nebylo nutné chránit rezistorem.

### Mosfet výstupy

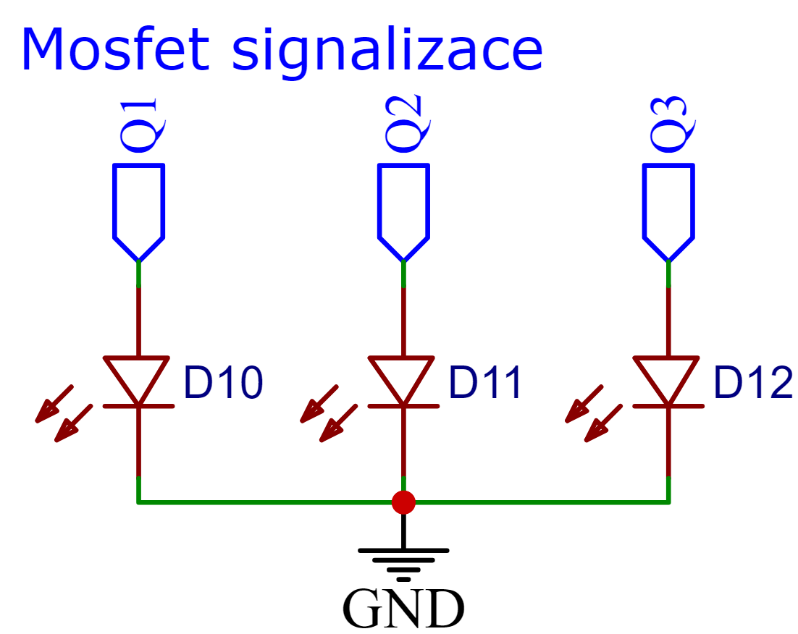
Pro realizaci mosfet výstupů jsem využil NPN Mosfetu. Stejně jako relé i mosfety jsem opatřil signalizační LED diodou. Na samotné desce jsem přidal další GND svorky, aby nedocházelo k přetížení vstupní GND svorky.

Obrázek 14 - Schéma mosfet výstupů



Zdroj: Vlastní zpracování

Obrázek 15 - Schéma mosfet signalizace

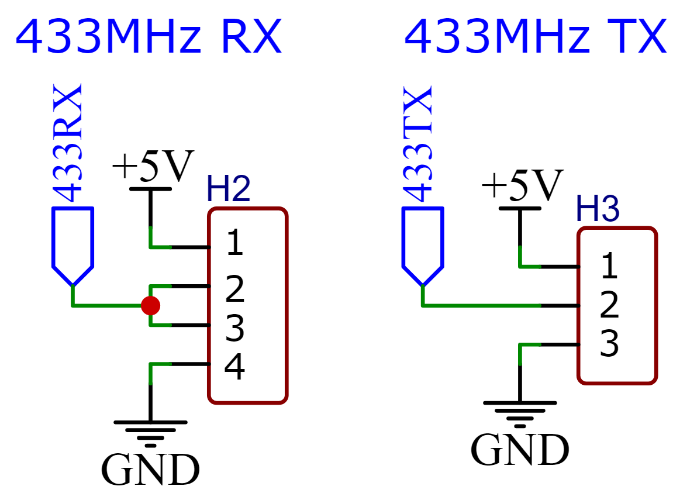


Zdroj:Vlastní zpracování

### 433 MHz komunikace

Pro možnost ovládání levných čínských zásuvek a čtení dat z meteo stanic jsem zvolil levné čínské moduly na frekvenci 433 MHz. Tyto moduly umožní možnost ovládání těchto „chytrých“ zásuvek přes wifi. Popřípadě je možné je využít například pro vylepšení starého bezdrátového domovního zvonku na zvonek chytrý s upozorněním na mobilní telefon. Dalším využitím může být třeba vzdálené ovládání vjezdové brány a podobně.

Obrázek 16 - Schéma zapojení modulů pro bezdrátovou komunikaci

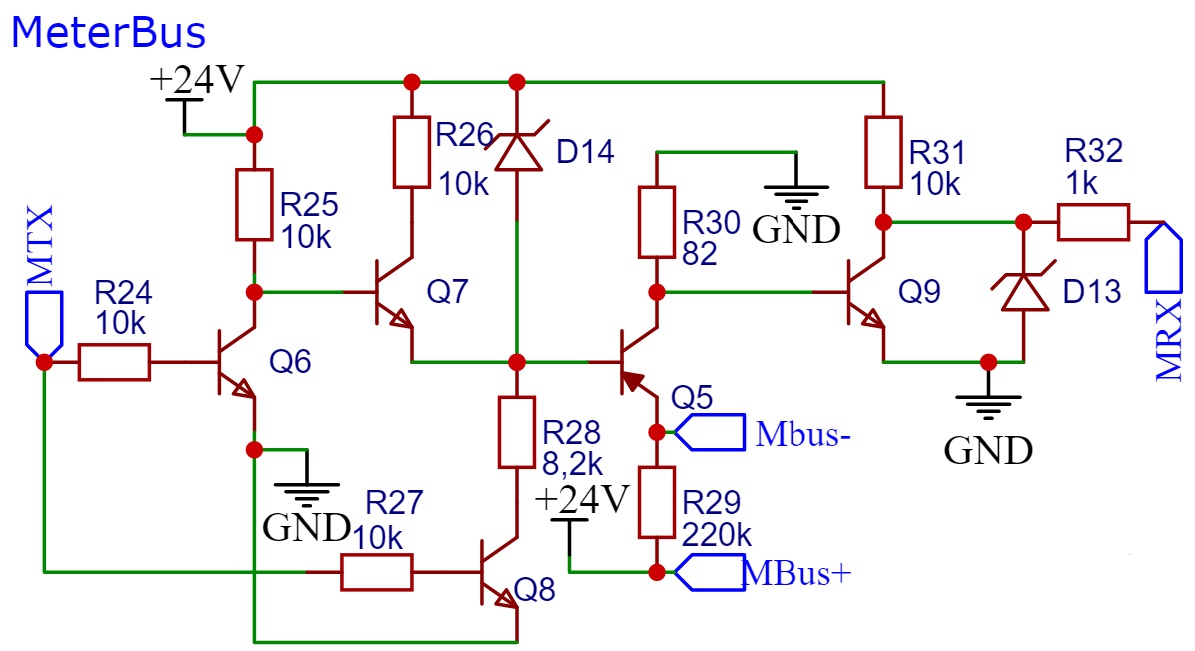


Zdroj: Vlastní zpracování

### Meterbus

Meterbus je protokol který není moc rozšířený v komunitě a většinou je využíván pro proprietární systémy. Samotný protokol je založen na proudové smyčce a jeho komunikace probíhá pomocí zatěžování komunikační linky. Tyto proudy jsou poté převedeny na TTL logiku. Toto schéma bylo převzato a upraveno z [githubu](https://github.com/roarfred/AmsToMqttBridge/tree/master/Electrical/MBus_Simulator).

Obrázek 17 - Schéma zapojení převodníku Meterbus

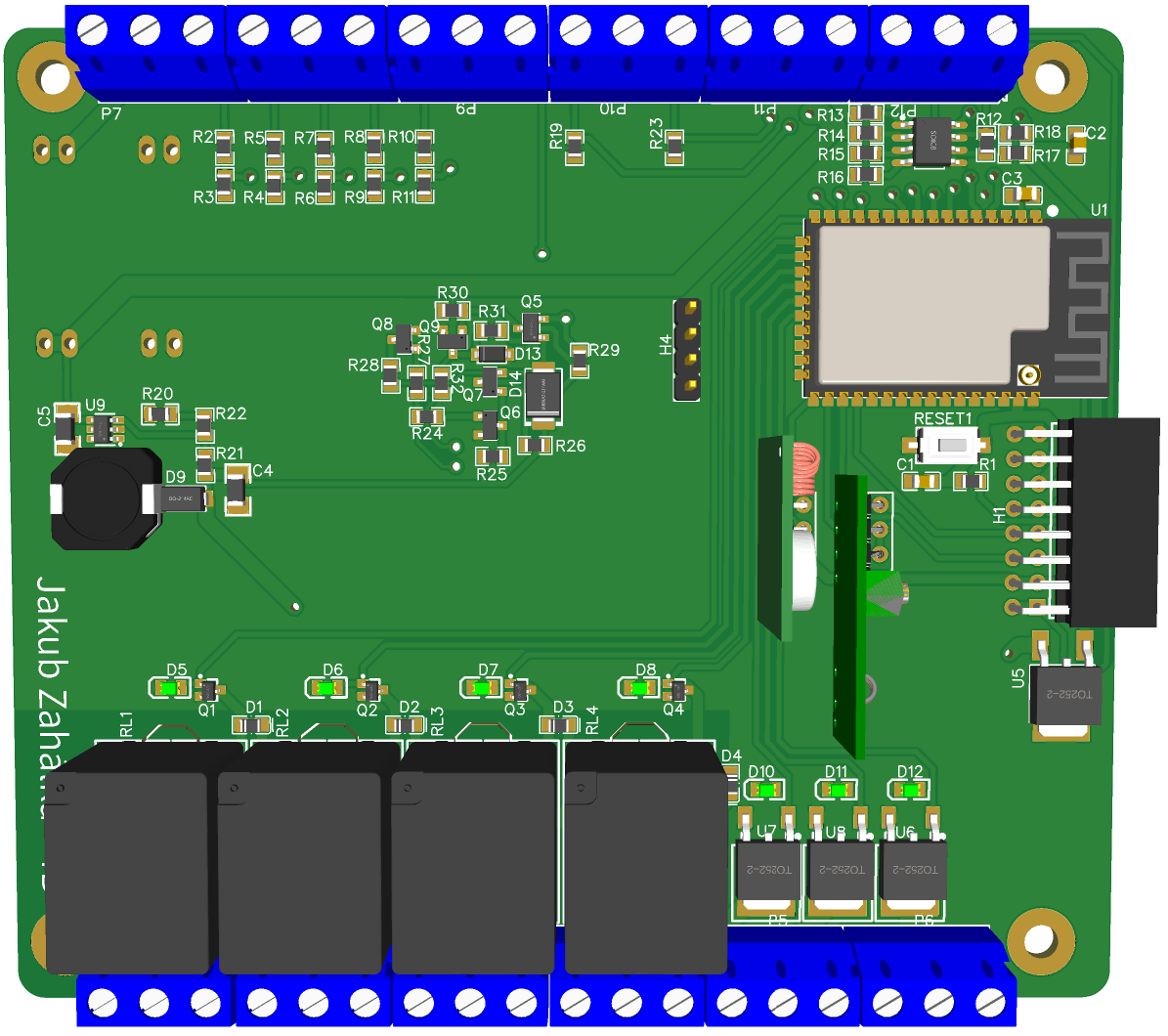


Zdroj: Vlastní zpracování

# Návrh DPS

Deska plošných spojů byla navrhnuta také v programu Easyeda a její 3D model byl později využit při výrobě krabičky na 3D tiskárně. Prvně jsem začal s plánováním přibližných rozložení součástek. Abych mohl pokračovat v práci na krabičce   
a celkovému systému uchycení, tak jsem vytvořil okraje desky a boční konektor pro rozšíření. Tento postup jsem zvolil z důvodu experimentování se speciálním úchytem na din lištu vytisknutého s integrovanou pružinou a print in place spojem.

Obrázek 18 - 3D render hotové DPS

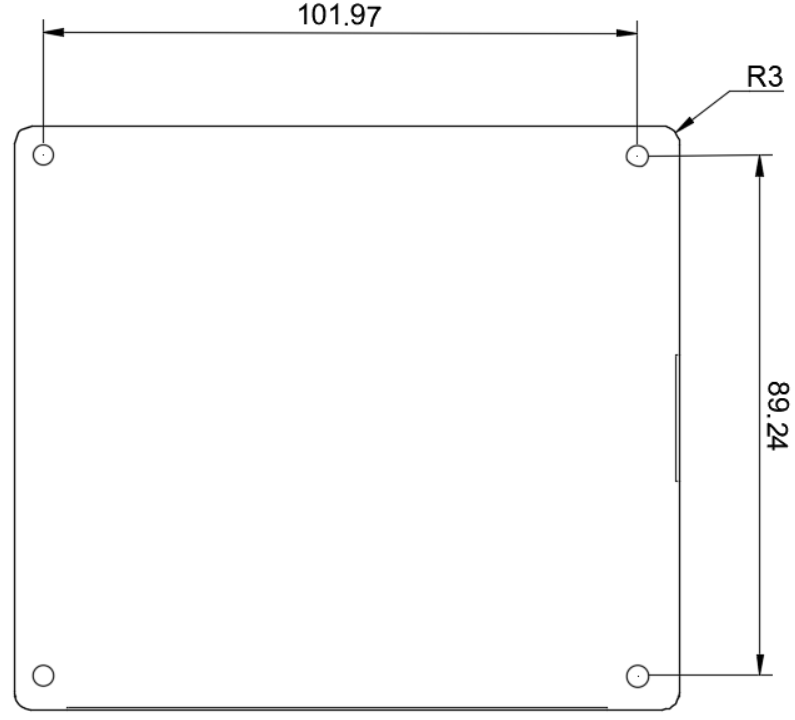


Zdroj: Vlastní zpracování

## Rozložení součástek

Při návrhu samotného rozložení součástek jsem se inspiroval u PLC automatů. Vstupy a komunikační protokoly jsou vyvedeny na horní části zařízení. Dolní část obsahuje relé a mosfet výstupy. Na boku se nachází rozšiřující konektor, který obsahuje programovací rozhraní a další vývody pro připojení rozšiřujících modulů.

Obrázek 19 - Nákres rozpoložení vstupů a výstupů + konektor modulu

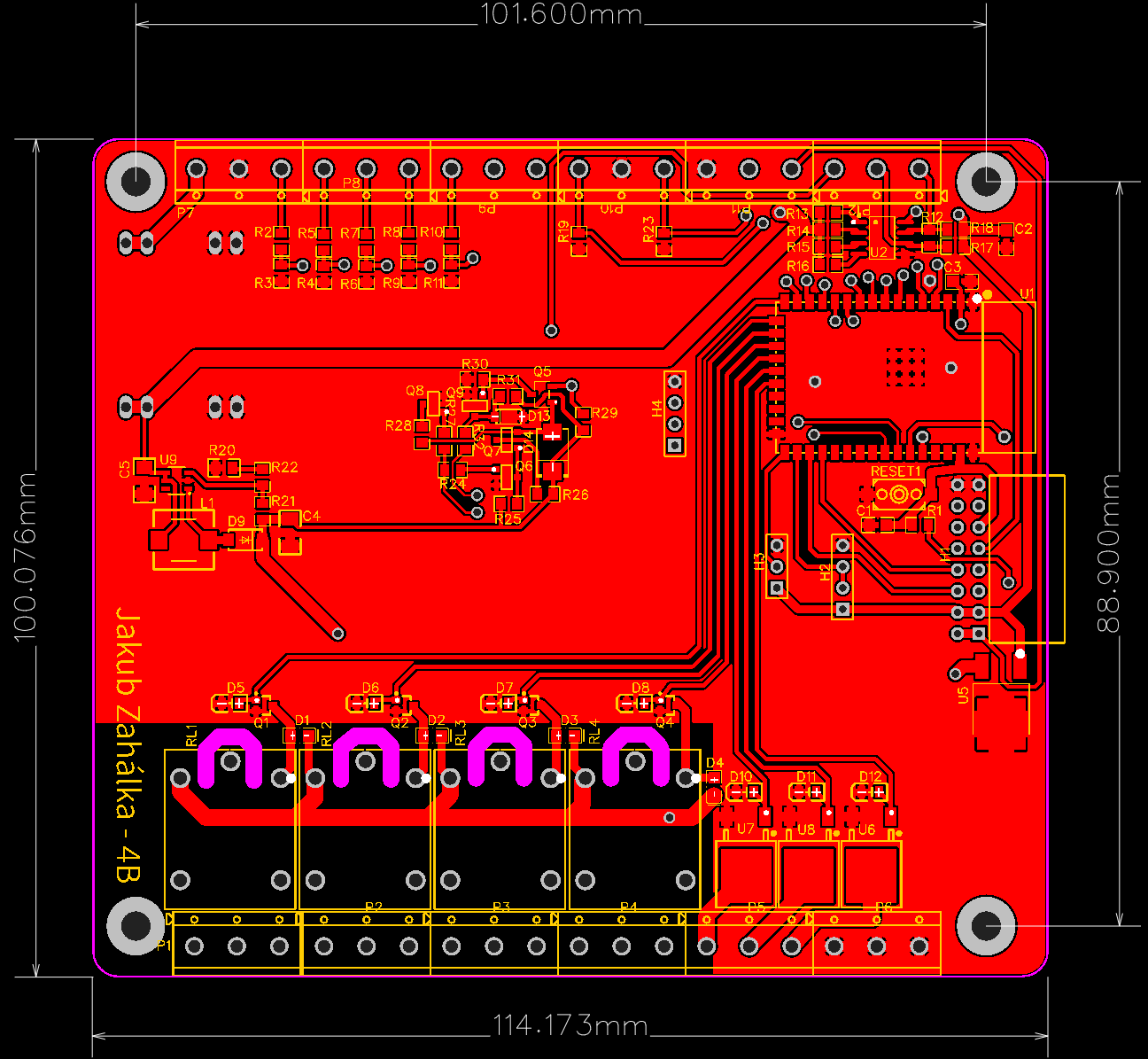


Zdroj: Vlastní zpracování

## Návrh a pravidla pro kreslení

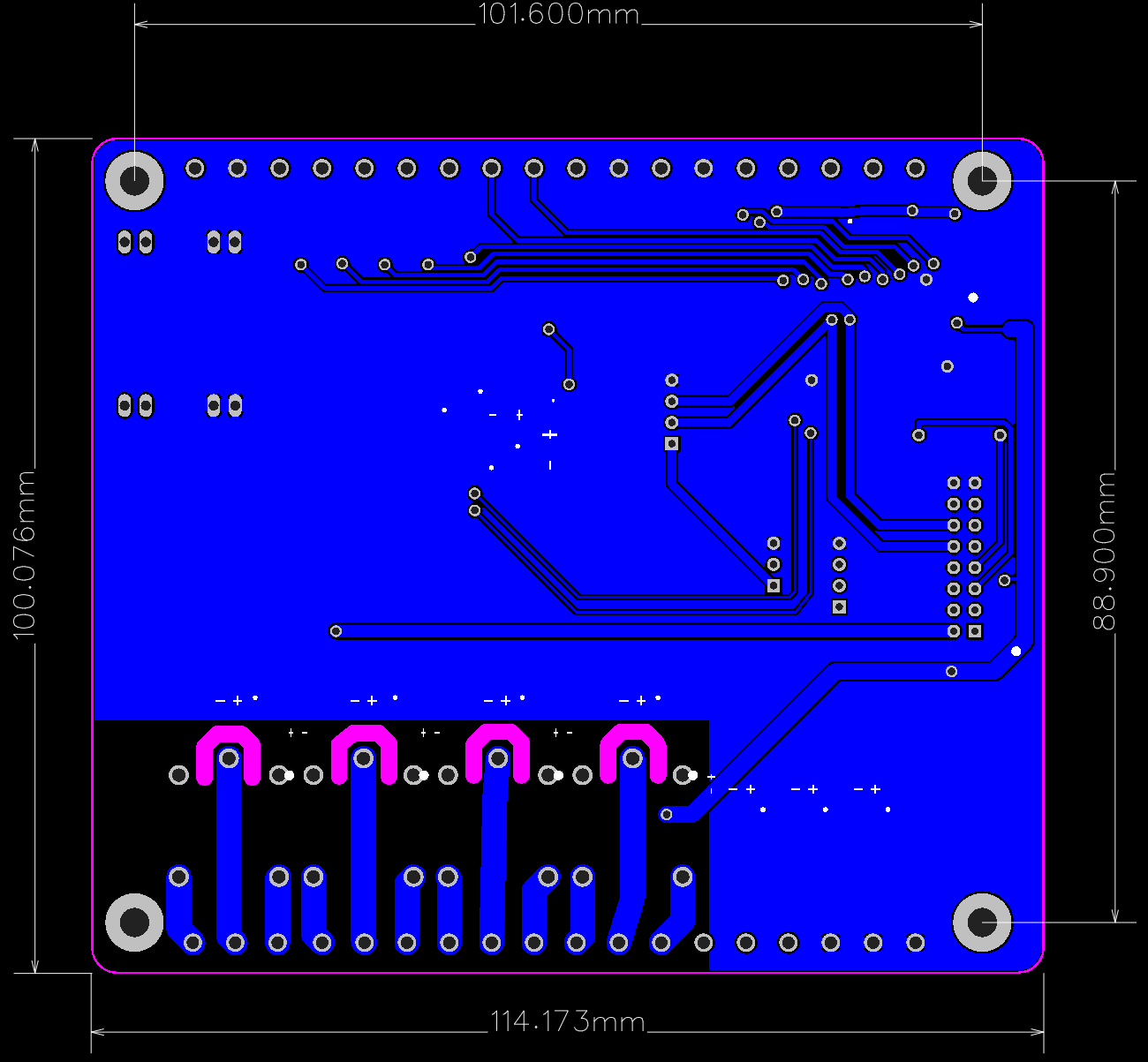
Původně měla deska být vyrobena na školní frézce, a proto je tloušťka cesty vždy větší jak 0,5mm. Nakonec jsem se ale rozhodl pro profesionální výrobu u čínského výrobce JLCPCB. Primární typ součástek jsem zvolil SMD. Tato technologie je novější a jednotlivé obvody se dají zmenšit. Pro samotný package jsem zvolil pouzdra rezistorů a kondenzátorů o velikosti 0805, které jsou malé, ale stále se dobře pájí v domácích podmínkách. Pod rezistory 0805 je možné vést spoj, ačkoliv s trochu menší tloušťkou cesty než 0,5mm. Prokovy jsou o děrovém průměru 0,9mm   
a celkový průměr 1,4mm. Tyto rozměry byly původně zvoleny z důvodu vlastnoruční výroby DPS, ale jak jsem již dříve zmiňoval, tak jsem si nechal desku vyrobit profesionálně.

Obrázek 20 - Vrchní vrstva DPS



Zdroj: Vlastní zpracování

Obrázek 21 - Spodní vrstva DPS



Zdroj: Vlastní zpracování

## Realizace a postup pájení DPS

Profesionálně vyrobenou desku jsem postupně osadil. Začal jsem od nejmenších pasivních součástek a postupně se propracoval do konce. První na řadě byly rezistory, poté kondenzátory a další malé součástky většinou o velikosti 0805. THT komponenty jsem nechal jako poslední kvůli jejich výškovému rozdílu. Pokud bych začal od THT komponent, tak by jsem si stížil samotné osazování. Kvůli chybě v mém návrhu jsem musel realizovat spoje DE, DI, RE, RO pomocí lakovaného drátu. Tyto spoje obsluhují integrovaný obvod MAX485CSA+. Po osazení jednotlivých komponent jsem osadil jednotlivé hotové moduly. Konkrétně vysílač a příjmač 433MHz a napájecí step down 5V.

# Tvorba krabičky

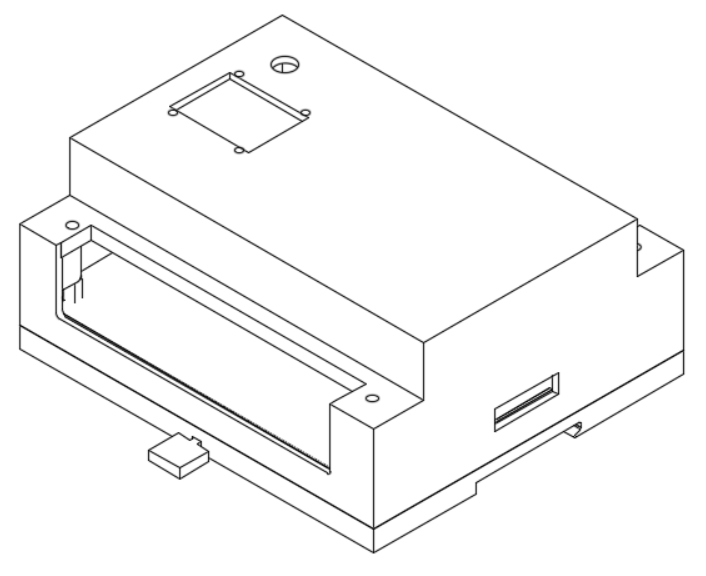
Pro tvorbu krabičky budu využívat program Fusion 360 od firmy Autodesk. V tomto programu navrhnu samotný model který potom převedu na různé rendery nebo nákresy.

Již do počátku jsem měl v plánu celou sestavu upevnit na DIN lištu. Po prozkoumání různých mechanismů jsem se rozhodl pro mechanismus s integrovanou pružinou   
a print in place spojem, který se staral o posun. V první fázi jsem navrhl spodní plát pro montáž již s funkčním mechanismem pro uchycení. Do tohoto dílu jsem poté vložil závitové vložky určené pro plast. V dalším kroku jsem přišrouboval distanční sloupky pro nadzvednutí DPS. Samotnou desku jsem poté zajistil dalšími distančními sloupky, které potom budou držet horní část krabičky. Z programu EASYEDA jsem si vyexportoval model mého plošného spoje, okolo kterého jsem navrhl horní část krabičky. V krytu jsem umístil OLED displej a otvor pro montáž SMA konektoru na připojení antény.

## 3D render krabičky

Různé rendery jsem vyexportoval přímo z prostředí Fusionu. Fusion   
je profesionální nástroj přesně určený k těmto aplikacím a je založený na návaznosti parametrů. V případě že změním jeden rozměr, tak by se měly ostatní změny přizpůsobit. Fusion podporuje animační přechody, automatické nákresy   
či pevnostní simulace. Rozměry krabičky jsou 120×105×53,5. Všechny uvedené rozměry jsou v mm. ([Příloha č. 4](#_Toc98982907))

Obrázek 22 - Nákres hotové krabičky z  rohového pohledu



Zdroj: Vlastní zpracování

Obrázek 23 – Nákres hotové krabičky z horního pohledu

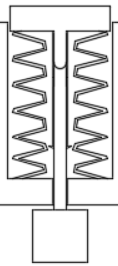


Zdroj: Vlastní zpracování

## 3D tisk

Samotný tisk jsem provedl na tiskárně Prusa MK3S. Tisk spodní části zabral 10 hodin. Horní část zabrala 12 hodin. V horní části jsem se neobešel bez podpor   
a ty jsem po dokončení tisku musel odstranit. Pružina, která byla součástí tisku,   
je ve své podstatě jen několik spojů, které se omezeně zdeformují. Pružina se tiskla ve „vzduchu“, respektive měla mezeru mezi povrchy. Díky tomu se poté mohla rozpohybovat pomocí šroubováku. Jistící páka se pohybuje v drážce a byla vytisknuta postupem print in place. Tato metoda slouží ke konstrukci nerozebíratelných spojů. Celá spodní část je jen jeden model a tisk proběhl   
bez problémů. Horní kryt jsem navrhl s pomocí modelu osazené desky vyexportovaného z EASYEDA. Tento model byl však ve špatném měřítku, a proto jsem jej musel zmenšit na reálnou velikost.

Obrázek 24 - Ukázka jistícího mechanismu



Zdroj: Vlastní zpracování

# Komunikace a vizualizace

Tato sekce rozebírá jednotlivé protokoly, jejich použití a implementaci. Je zde také nastíněný základní princip fungování jednotlivé komunikace.

## MQTT

MQTT je komunikační protokol přímo určený k tomuto typu užití. Jeho základem   
je takzvaný broker, který se stará o samotné připojení. Na tento broker se připojují klienti, které mu potom předávají informace a dochází k obousměrné komunikaci. Broker je takový univerzální hub. Klient po připojení provede akci subscribe,   
při kterém řekne, že chce dostávat novinky z určitých větví. Pokud někdo jiný pošle na stejnou větev nějakou informaci, tuto informaci porovná a podle ní může něco zapnout nebo vypnout. Další akce, co provádí klient, je publish, který „zveřejní“ zprávu či měřený údaj do předem určené větve. Protokol MQTT má spoustu dalších funkcí, ale v mé práci jsem využil pouze základních. Největší výhodou MQTT je jeho rozšiřitelnost. Na všechny zařízení stačí jeden broker, který obslouží všechny jeho klienty. Klienti se poté identifikují client\_ID, které slouží jako identifikátor.

## Bridge

Tento program slouží jako univerzální most mezi MQTT a databází influx. Je napsaný v programu Python a jeho jediným úkolem je předávání dat do databáze s časovým razítkem. Jako klient se připojí na MQTT broker, odkud se subscribne   
do specifikované větve. Data, co přijdou do této větve, opatří časovým razítkem  
a zapíše je do databáze.

## Databáze

Databáze je úložiště pro data. Jako hlavní databázi jsem zvolil program InfluxDB, založený na časových udajích. Tato databáze je také přímo kompatibilní s Grafanou, která je využívána pro tvorbu grafů.

## Grafana

Grafana je open-source program přímo uzpůsobený k interaktivní tvorbě grafů.   
Má již vestavěný celý systém pro správu a úpravu jednotlivých panelů. Další výhodou je možnost upozornění na havarijní stavy přes škálu různých kanálů. Přímo v  Grafaně jdou provádět i různé matematické operace, ale tyto úkony se provádí   
na straně uživatele. V případě velkých sad dat (dny, týdny) dojde k dočasnému „zamrznutí“. Proto je vhodné mít databázi na serveru s SSD diskem a rychlým připojením k internetu. Za dodržení těchto parametrů je možné rychle načítat   
i velké sety dat. Načtená data je možné vyexportovat do formátu .csv a dále   
je zpracovat například v programu Microsoft Excel. Ukázku grafu si můžete prohlédnout v [Příloze č. 5.](#_Toc98984145)

## OTA

OTA neboli over the air je knihovna určená k vzdálené aktualizaci firmwaru.   
Pro základní funkčnost je potřeba přístup na stejnou síť jako programované zařízení. Knihovna podporuje nastavení přístupových údajů pro aktualizace. Samotné aktualizace se nejdříve nahrají do paměti a po dokončení stahování   
se automaticky nahrají do hlavní paměti. V případě problému s aktualizací se vrátí k poslední verzi před nahráním. OTA jsem zvolil z důvodu udržitelnosti IoT zařízení. Většinou se podobné zařízení neudržují.

# Demo program

Pro základní test jsem napsal demo program v prostředí arduino IDE, přes které jsem i samotné zařízení programoval. Jednotlivé funkce jsem postupně zprovoznil   
a otestoval. Demo program se věnuje základním funkcionalitám desky.

## Displej

Pro zobrazování základních informaci jsem použil OLED displej s rozlišením 128x64. Samotná komunikace probíhá přes sběrnici I2C a řízení displeje zajišťuje knihovna SSD1306 of firmy Adafruit. Při startu displej ukáže logo školy, které jsem překonvertoval na bitmapu. Tuto bitmapu pak displej vykreslí. Po dokončení startovací sekvence na displej vypíše IP adresu modulu. V případe nějakého problému při fungování se na displeji zobrazí hláška informující o problému.

## Bezdrátová komunikace 433MHz

Pro komunikaci na frekvenci 433 MHz využívám knihovny rc-switch. Jedná   
se o základní knihovnu určenou pro levné čínské moduly či ovladače a umožňuje   
mi ovládat různé zařízení fungující na této frekvenci. Tato knihovna je schopna ovládat zvonky, brány a další podobná zařízení.

## Protokol MQTT

Komunikace s MQTT brokerem je zajištěna knihovnou pubsubclient. Tato knihovna spolupracuje s knihovnou wifi, která obsluhuje wifi připojení. Používám   
ji jen v základní konfiguraci, pro akce subscribe a publish. O ovládání výstupů se stará funkce callback, která porovná přijaté zprávy na odebíraných kanálech.   
Při nepovedeném připojení na server se na displeji objeví hláška „MQTT ERROR“.

## ArduinoOTA

Pro OTA aktualizace využívám vestavěné knihovny ArduinoOTA. Tato knihovna vyžaduje nějaký typ připojení k síti a uživatel, který chce nahrát firmware na desku, potřebuje znát přístupové údaje k programování. Velkým úskalím tohoto způsobu je, že firewall může blokovat a většinou i blokuje. Vyšší verzí by bylo OTA, které   
se provádí automatickým stahováním binárního souboru z nějaké volně dostupné stránky. Tento typ jsem ale pro demo program nevyužil.

Závěr

Cílem mé práce bylo navrhnout a realizovat IoT zařízení určené k dohledu   
nad různými systémy. Díky volnému zadání jsem byl schopen přidat spoustu různých funkcí. Tyto funkce rozšířily samotnou možnost užití, kvůli kterým je systém schopný zastat pozici hlavní řídící jednotky chytrého domu i podobných jiných prostředí. Moje práce se dá rozdělit na 6 bloků:

1. Díky použití napěťového děliče je možné využít vstupu do 23V a tato specifikace se hodí právě při užívání snímačů z oboru automatizace. Tyto vstupy jde také využít jako digitální vstupy s nastavitelnou úrovní logiky. Dále jsem zařízení opatřil proudovou smyčkou. Tato smyčka dokáže měřit automatizační snímače s proudovým výstupem 4-20mA a zpřístupní řadu nových snímačů. V případě potřeby je pro uživatele dostupné rozhraní RS485, které bývá často používané v průmyslu. Je možné jej třeba využít pro přidání vzdáleného dohledu k starší výrobní lince. Jednotlivá data jsou periodicky posíláná na server kde jsou zpracována.
2. Výstupy jsou realizovány klasickými relé a dalšími mosfet výstupy. Během návrhu jsem dodržel izolační vzdálenost cest připojujících k relé, a proto mohu spínat i 230V spotřebič. Mosfet výstupy mají přidané GND konektory navíc, které slouží pro zkrácení cesty k výstupu. Díky tomuto přístupu jsem eliminoval případný problém s přetížením vstupu desky. Boční konektor sloužící pro rozšiřující modul je opatřen programovacím rozhraním, komunikací I2C, napájením a nějakými volnými GPIO piny. Tato skutečnost mi umožňuje případné rozšíření s vlastním modulem nebo dalšími snímači či jinými periferiemi.
3. DPS jsem nechal vyrobit u JLCPCB a byl jsem velice spokojen s jejich kvalitou.

Návrh desky by šel optimalizovat, ale stále jej hodnotím velmi povedeně. V další verzi bych vyměnil čínský modul za vlastní obvod přímo na DPS.

1. U krabička vytisknuté na 3D tiskárně jsem využil technik, které jsem dříve neznal. Pro správné uchycení jsem využil závitové vložky do plastu a dosáhl jsem téměř perfektního výsledku. Nejvíce pokročilý je mechanismus pro uchycení na DIN lištu. U jeho tvorby jsem využil hned několika různých postupů a díky tomu jsem mohl vytisknout celý spodní plát jako jeden kus bez potřeby externích součástí. Horní kryt jsem navrhl pomocí 3D modelu desky a po vytisknutí seděl na první pokus.
2. Samotný použitý software je velice dobře rozšiřitelný či vyměnitelný. Použitý protokol MQTT je schopný obsluhovat velké množství zařízení. Tato zařízení na sobě nejsou závislá a vzájemně na sobě nezávisí. Bridge je most mezi protokolem MQTT a databází influx, ale je jednoduché ho vyměnit za jiný, který bude data posílat do jiné databáze nebo je bude třeba ukládat do .csv souboru. Databázi Influx je možné vyměnit za jakoukoli jinou po případné úpravě mostu. Grafanu jsem zvolil kvůli její jednoduchosti a již fungujícím administrátorskému systému. Velké plus je možnost posílání notifikací v případě havárie. Tento systém je však možné vyměnit za jakýkoli jiný a není nutné se na něj vázat.
3. OTA protokol je vhodný pro vzdálenou aktualizaci firmwaru zařízení a případné vylepšení systému.

Celé zařízení se vyšplhalo s výrobní cenou na něco okolo 2000Kč. Tato cena je vyšší, než jsem původně předpovídal, ale tato skutečnost dává smysl. V porovnání s jinými alternativami je na tom velice solidně a je mnohem levnější než konkurence. Celou práci hodnotím velmi pozitivně, i přes některé její nedostatky. Během práce jsem se toho spoustu nového naučil a prozkoumal pro mě neobjevené oblasti.

Seznam použité literatury

[1] WILLIAMS, Tim. *The circuit designer´s companion*. 2nd ed. Amsterdam: Elsevier, 2005. ISBN 978-0750663700.

[2] *ESP32S2 - Get started* [online]. Online: Espressif, 2018 [cit. 2022-01-18]. Dostupné z: <https://docs.espressif.com/projects/esp-idf/en/latest/esp32s2/get-started/>

[3] *ESP32 MQTT: Publish and Subscribe with Arduino IDE* [online]. Online: Random Nerd Tutorials, 2018 [cit. 2022-01-15]. Dostupné z: <https://randomnerdtutorials.com/esp32-mqtt-publish-subscribe-arduino-ide/>

Seznam použitých zkratek

**IoT** Internet of Things

**DPS** Deska plošných spoů

**SMD** Surface Mount Device

**THT** Through Hole Technology

**MQTT** MQ Telemetry Transport

**Mbus**  Meterbus

**ADC** Analog Digital Convertor

**GPIO** General Purpose Input Output

**PLC** Programmable Logic Controller

**OTA** Over The Air

**TTL** Transistor Transistor Logic

Seznam obrázků a tabulek

[Obrázek 1 – Vývojová sada M5Stack Core 8](#_Toc98982418)

[Obrázek 2 - Blokové schéma prototypu 9](#_Toc98982419)

[Obrázek 3 - Ukázka grafu průtoku vody 9](#_Toc98982420)

[Obrázek 4 - Schéma zapojení mikroprocesoru 10](#_Toc98982421)

[Obrázek 5 - Schéma doprovodných součástek pro CPU 11](#_Toc98982422)

[Obrázek 6 - Blokové schéma napájení 12](#_Toc98982423)

[Obrázek 7 - Schéma napájecího systému 12](#_Toc98982424)

[Obrázek 8 - Schéma rozložení jednotlivých vstupů/výstupů + boční konektor pro moduly 13](#_Toc98982425)

[Obrázek 9 - Napětově děliče pro ADC 14](#_Toc98982426)

[Obrázek 10 - Schéma proudové smyčky 15](#_Toc98982427)

[Obrázek 11 - Schéma obvodu pro komunikaci RS485 16](#_Toc98982428)

[Obrázek 12 - Schéma zapojení relé výstupů 17](#_Toc98982429)

[Obrázek 13 - Schéma zapojení relé signalizace 17](#_Toc98982430)

[Obrázek 14 - Schéma mosfet výstupů 18](#_Toc98982431)

[Obrázek 15 - Schéma mosfet signalizace 18](#_Toc98982432)

[Obrázek 16 - Schéma zapojení modulů pro bezdrátovou komunikaci 19](#_Toc98982433)

[Obrázek 17 - Schéma zapojení převodníku Meterbus 20](#_Toc98982434)

[Obrázek 18 - 3D render hotové DPS 21](#_Toc98982435)

[Obrázek 19 - Nákres rozpoložení vstupů a výstupů + konektor modulu 22](#_Toc98982436)

[Obrázek 20 - Vrchní vrstva DPS 23](#_Toc98982437)

[Obrázek 21 - Spodní vrstva DPS 24](#_Toc98982438)

[Obrázek 22 - Nákres hotové krabičky z  rohového pohledu 26](#_Toc98982439)

[Obrázek 23 – Nákres hotové krabičky z horního pohledu 27](#_Toc98982440)

[Obrázek 24 - Ukázka jistícího mechanismu 28](#_Toc98982441)

Přílohy

[Příloha č. 1 Poster k maturitní práci 40](#_Toc98984141)

[Příloha č. 2 Kompletní schéma 41](#_Toc98984142)

[Příloha č. 3 Rozložený pohled celé sestavy 42](#_Toc98984143)

[Příloha č. 4 Nákres krabičky 43](#_Toc98984144)

[Příloha č. 5 Graf po dnech 44](#_Toc98984145)

Příloha č. 6 3D Modely

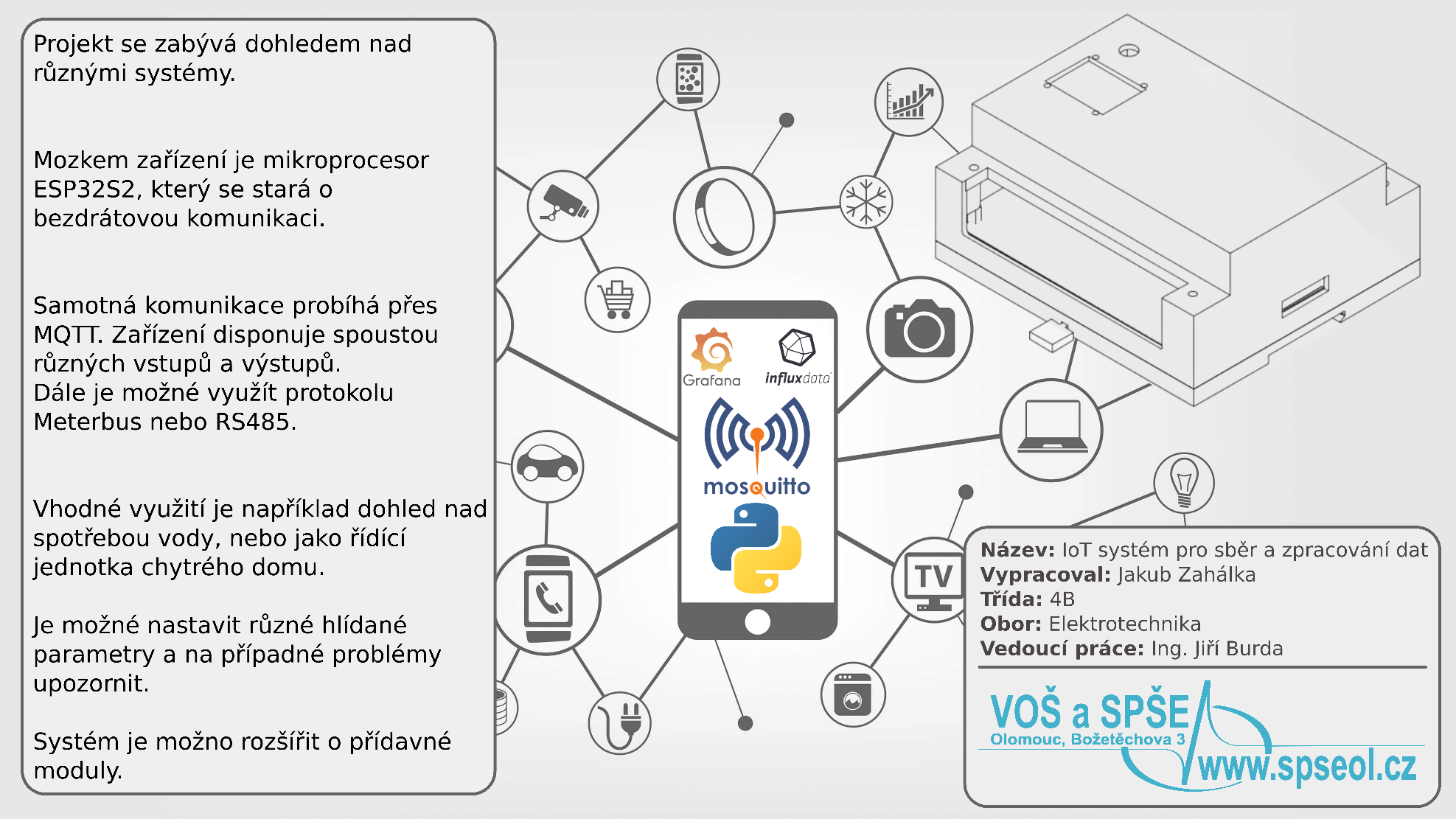
Příloha č. 7 Easyeda

Příloha č. 8 Rendery

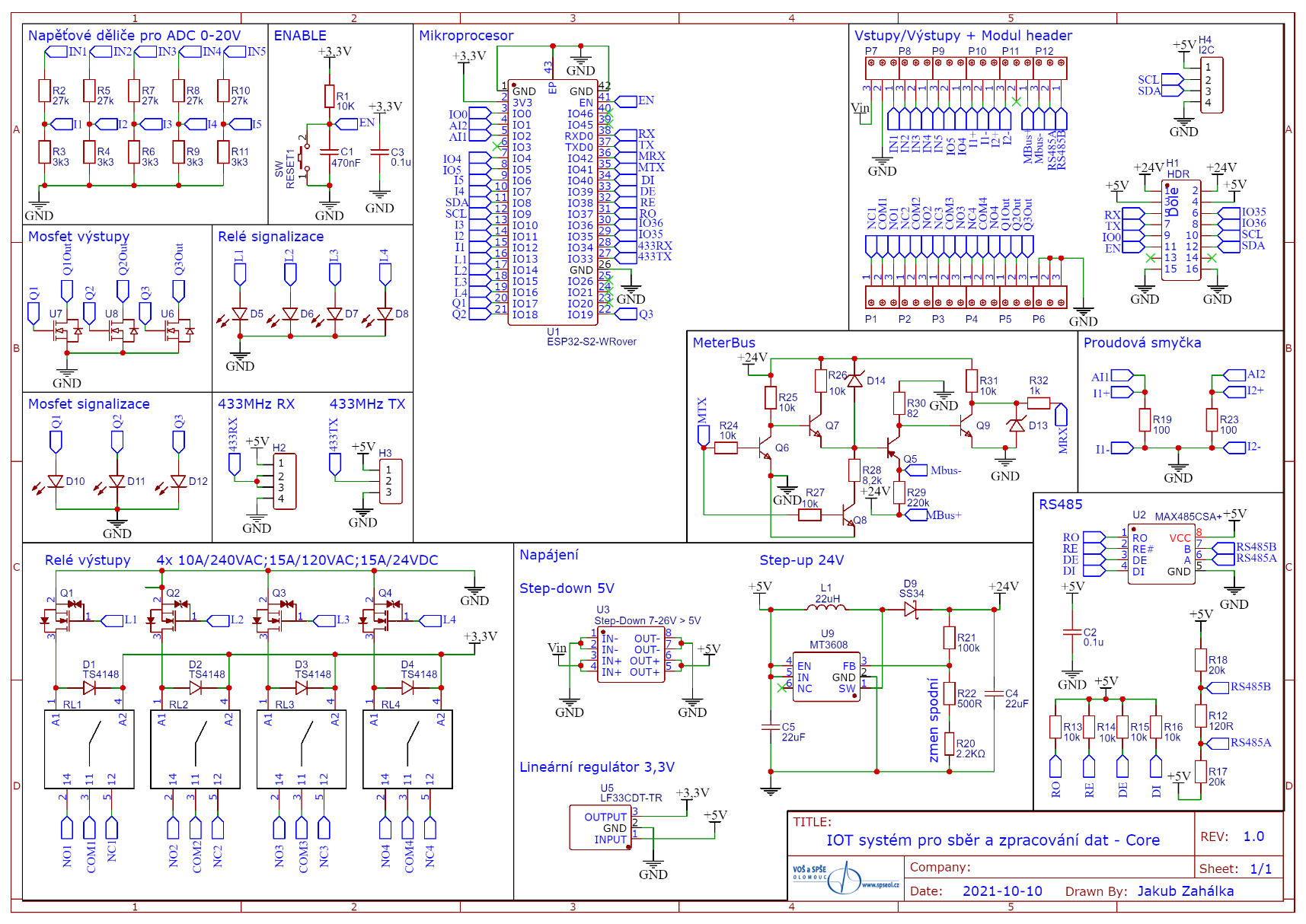
Příloha č. 9 BOM dlouhodobka

Příloha č. 10 DEMO program

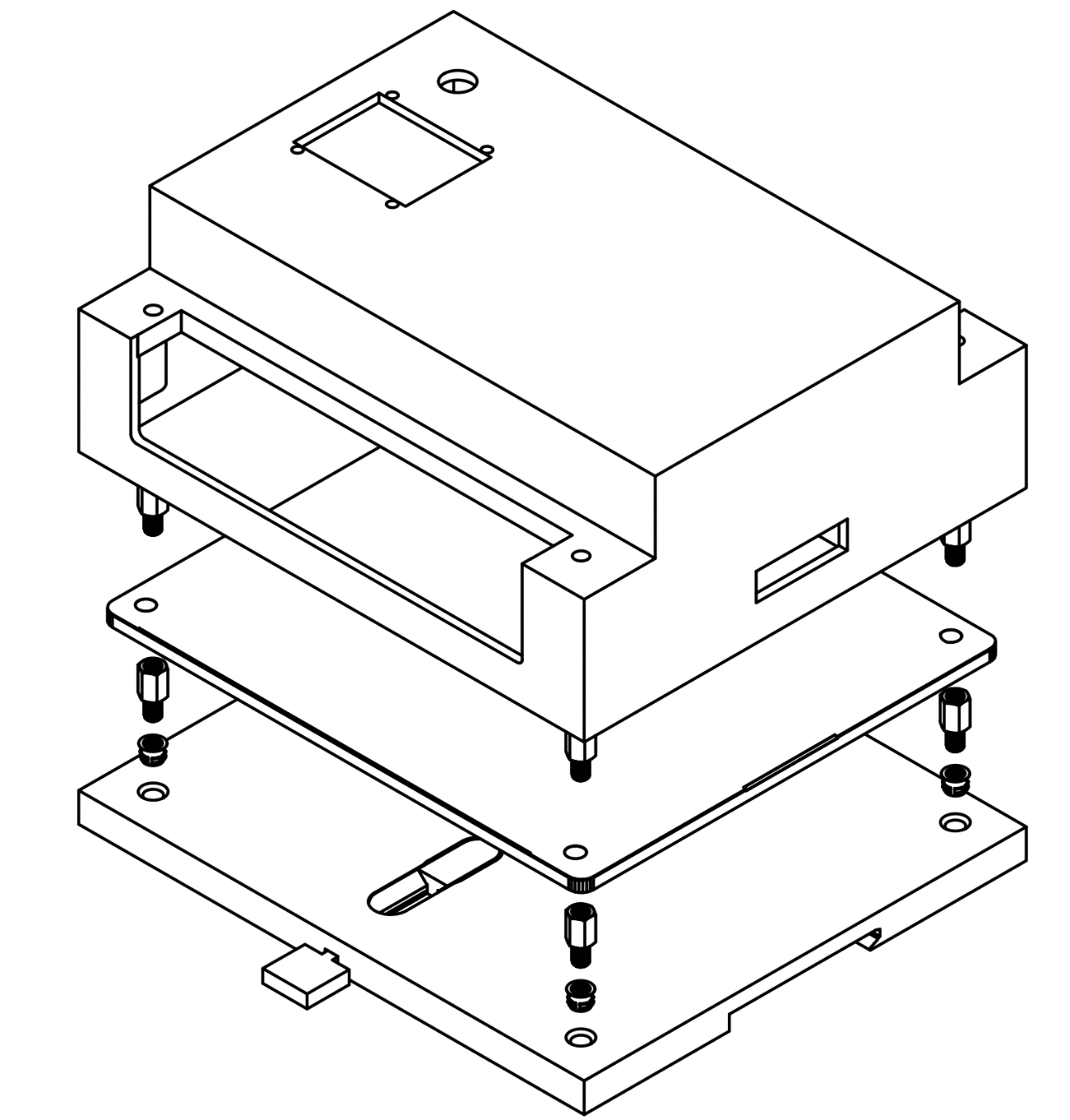
Příloha č. 1 Poster k maturitní práci



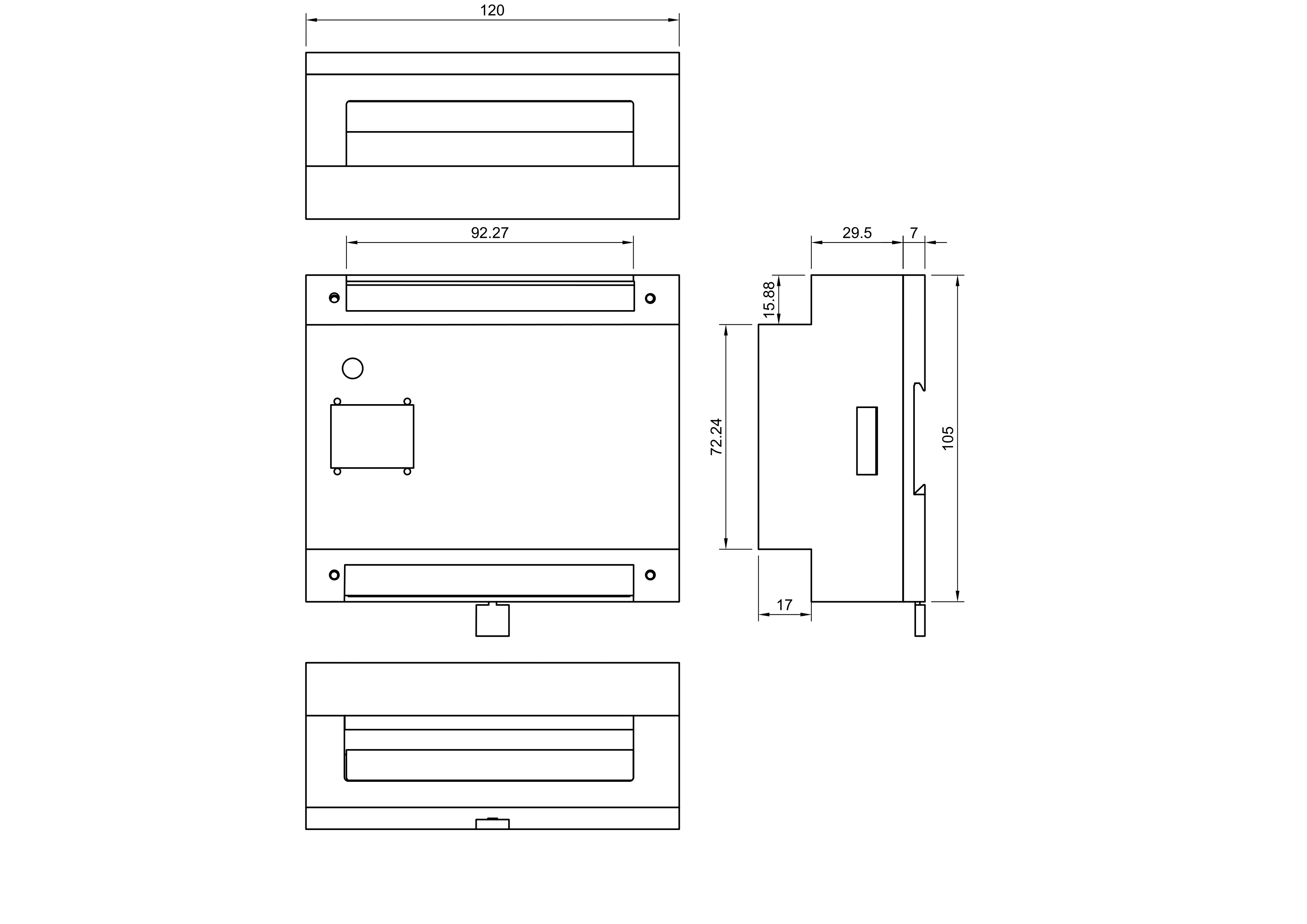
Příloha č. 2 Kompletní schéma



Příloha č. 3 Rozložený pohled celé sestavy



Příloha č. 4 Nákres krabičky



Příloha č. 5 Graf po dnech

