



Ekonomická
fakulta
Faculty
of Economics

Jihočeská univerzita
v Českých Budějovicích
University of South Bohemia
in České Budějovice

Jihočeská univerzita v Českých Budějovicích
Ekonomická fakulta
Katedra aplikované matematiky a informatiky

Bakalářská práce

Vývoj aplikace s využitím platformy Arduino

Vypracoval: Tomáš Linhart
Vedoucí práce: doc. Ing. Ladislav Beránek, CSc.

České Budějovice 2021

JIHOČESKÁ UNIVERZITA V ČESKÝCH BUDĚJOVICÍCH

Ekonomická fakulta

Akademický rok: 2019/2020

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Tomáš LINHART
Osobní číslo: E17836
Studijní program: B6209 Systémové inženýrství a informatika
Studijní obor: Ekonomická informatika
Téma práce: Vývoj aplikace s využitím platformy Arduino
Zadávající katedra: Katedra aplikované matematiky a informatiky

Zásady pro vypracování

Cílem práce je navrhnout automatické otevírání dveří u příbytku hospodářských zvířat (např. slepic) s použitím open source platformy Arduino. Součástí práce bude analýza dostupných řešení, bude popsán způsob automatického ovládání včetně modelu a bude zpracována mobilní aplikace umožňující vzdálené nastavení systému automatického ovládání dveří. Součástí práce bude i zhodnocení, včetně ekonomického.

Metodický postup:

1. Teoretický popis dostupných řešení, popis možností open source platformy Arduino.
2. Popis automatického ovládání včetně schématu.
3. Návrh, popis vývoje a implementace mobilní aplikace a jeho otestování.
4. Zhodnocení, vypracování doporučení a závěrů.

Rozsah pracovní zprávy: 40 – 50 stran
Rozsah grafických prací: dle potřeby
Forma zpracování bakalářské práce: tištěná

Seznam doporučené literatury:

1. BENES, P. (2005). *Automatizace a automatizační technika*. Brno: CP Books.
2. CASTLEDINE, E., EFTOS, M., & WHEELER, M. (2013). *Vytvoříme mobilní web a aplikace pro chytré telefony a tablety*. Brno: Computer Press.
3. ROUBAL, J., & HUŠEK, P. (2011). *Regulační technika u příkladech*. Praha: BEN – technická literatura.
4. SELECKÝ, M. (2016). *Arduino: uživatelská příručka*. Brno: Computer Press.

Vedoucí bakalářské práce: doc. Ing. Ladislav Beránek, CSc.
Katedra aplikované matematiky a informatiky

Datum zadání bakalářské práce: 25. října 2019
Termín odevzdání bakalářské práce: 30. srpna 2020

JIHOČESKÁ UNIVERZITA V ČESKÝCH BUDEJOVICÍCH
Fakulta ekonomická
Katedra ekonomické informatiky
ZADÁNÍ BAKALÁŘSKÉ PRÁCE
(číslo zadání: 1/2020)

[Faint, mostly illegible text, likely the main body of the assignment or a list of references.]


doc. Dr. Ing. Dagmar Škodová Parmová
děkanka

JIHOČESKÁ UNIVERZITA
V ČESKÝCH BUDEJOVICÍCH
EKONOMICKÁ FAKULTA
Studentská 13 (261)
370 05 České Budějovice


doc. RNDr. Tomáš Mrkvička, Ph.D.
vedoucí katedry

V Českých Budějovicích dne 11. května 2020

Prohlašuji, že svou bakalářskou práci jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47 zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to v nezkrácené podobě vypuštěním vyznačených částí archivovaných Ekonomickou fakultou elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách, a to se zachováním mého autorského práva k odevzdanému textu této kvalifikační práce. Souhlasím dále s tím, aby toutéž elektronickou cestou byly v souladu s uvedeným ustanovením zákona č. 111/1998 Sb. zveřejněny posudky školitele a oponentů práce i záznam o průběhu a výsledku obhajoby kvalifikační práce. Rovněž souhlasím s porovnáním textu mé kvalifikační práce s databází kvalifikačních prací Theses.cz provozovanou Národním registrem vysokoškolských kvalifikačních prací a systémem na odhalování plagiátů.

V Českých Budějovicích, dne

.....
Tomáš Linhart

Poděkování

Rád bych poděkoval panu doc. Ing. Ladislavu Beránkovi, CSc., za odborné vedení při zpracování mé bakalářské práce, jeho cenné připomínky a rady. Zároveň velké díky patří kolegovi Ing. Vojtěchu Navrátilovi za odbornou konzultaci v oblasti hospodářských zvířat.

Abstrakt

Tato bakalářská práce popisuje návrh automatického otevírání dveří u příbytku hospodářských zvířat (například slepic) a ovládání pomocí mobilní aplikace. Dokument obsahuje popis řídicí platformy Arduino, různé zapojení komponentů, možnosti automatického otevírání dveří, výběr vhodných komponentů a návrh konstrukce automatického otevírání dveří. V této práci je popsán systém, zahrnující funkční model a vývoj mobilní aplikace s ukázkou dostupných funkcí automatického otevírání dveří. Práce obsahuje ekonomický závěr a porovnání se současnými tržními příležitostmi, včetně přehledu konkurenčních výrobků.

Klíčová slova: systém otevírání dveří, automatizace, Arduino, Blynk

Abstract

This bachelor thesis describes the design of an automatic door opening system for livestock (hens) and remote control using a mobile application. Document includes description of the control unit – Arduino development board, different ways of wiring components, possibilities of an automatic door opening system, a selection of appropriate components and design of construction of the automatic door opening system. In this work describes creation of the system including a functional model and mobile application development with demonstration of available functions for remote control of the automatic door opening system. The thesis contains an economic conclusion and a comparison with current market opportunities, including an overview of competing products and their market share.

Keywords: door opening system, automation, Arduino, Blynk

Obsah

1. ÚVOD	1
2. CÍLE PRÁCE	2
3. SEZNAM POUŽITÝCH ZKRATEK	3
4. TEORETICKÁ ČÁST	4
4.1. ARDUINO	4
4.1.1. <i>Historie Arduina</i>	5
4.1.2. <i>Arduino UNO</i>	6
4.2. STEJNOSMĚRNÝ MOTOR	8
4.3. H-MŮSTEK	9
4.4. FOTOREZISTOR	10
4.5. WI-FI MODUL	11
4.6. SOFTWARE	12
4.6.1. <i>Arduino IDE</i>	12
4.6.2. <i>Fritzing</i>	13
4.6.3. <i>Android</i>	14
4.6.4. <i>Blynk</i>	15
4.7. KONKURENČNÍ VÝROBKY	16
5. PRAKTICKÁ ČÁST	19
5.1. VÝBĚR HLAVNÍCH KOMPONENT	19
5.2. NÁVRH MODELU	27
5.3. ELEKTRICKÉ ZAPOJENÍ	29
5.4. SCHÉMA ZAPOJENÍ	31
5.5. VÝVOJ APLIKACE	32
5.6. TESTOVÁNÍ	43
6. ZÁVĚR	44
I. SUMMARY	46
II. SEZNAM POUŽITÝCH ZDROJŮ	48
III. SEZNAM OBRÁZKŮ	50
IV. SEZNAM TABULEK	51
V. SEZNAM PŘÍLOH	52
VI. PŘÍLOHY	53

1. Úvod

Prvotní myšlenku mi vnukl kolega ze Zemědělské fakulty Jihočeské univerzity Ing. Vojtěch Navrátil, který řešil problém se zabezpečením u svého chovu slepic. Hlavním důvodem potřeby zabezpečení příbytku je ochrana zvířat před predátory v nočních hodinách jako je například kuna, liška případně zbloudilý pes. Automatické otevírání dveří by mělo usnadnit člověku práci a zároveň zajistit neustálý přehled o příbytku hospodářských zvířat. Kolega mi také poskytl cenné informace z oblasti chovu hospodářských zvířat. Základním předpokladem pro správnou funkci systému automatického otevírání/zavírání dveří byl fakt, že slepice se řídí podle slunce, kdy jim teoreticky už nehrozí nebezpečí od výše zmíněných šelem. Proto je možné systém řídit pomocí předem nastavených časů dle západu/východu slunce, případně pomocí senzoru intenzity světla. Vše by mělo být ovládatelné přes mobilní aplikaci k zajištění maximálního pohodlí.

V dnešní době začíná být téměř vše kolem nás řízeno počítači. Wifi připojení je už samozřejmostí a mobilní sítě jsou na vzestupu, čemuž ještě přispěje zavedení 5G sítě na našem území. Odvětví chytrých domácností zažívá v posledních letech obrovský rozmach. Je možné celý dům ovládat z mobilní aplikace, od zabezpečení, přes vytápění, osvětlení až po ozvučení každé místnosti. Proto by ani toto odvětví nemělo zůstat pozadu a využívat dostupné technologie k usnadnění a zjednodušení každodenního života.

2. Cíle práce

Cílem mé práce je navrhnout systém automatického otevírání dveří u příbytku hospodářských zvířat, v mém případě slepic.

Celý systém je postavený na platformě Arduino. Ta vyhodnocuje nastavené časy, případně porovnává údaje ze senzoru intenzity světla. Na základě těchto informací ovládá mechanismus dveří dle potřeby. Zároveň je propojena pomocí Wi-Fi modulu s mobilní aplikací k usnadnění ovládání. Ke zvedání dveří je použit stejnosměrný motor s převodovkou.

Bakalářská práce se skládá z teoretické a praktické části. V teoretické části popisují jednotlivé prvky použité při konstrukci: řídicí jednotku, motor, senzor, napájení. Dále také použitý software k návrhu schémat a samotnému vývoji aplikace a přehled nejvýznamnějších konkurenčních výrobků. V praktické části popisují vlastní konstrukci zařízení a vývoj mobilní aplikace.

V závěru shrnuji praktické poznatky a řešené problémy v průběhu práce. Zároveň uvádím možné vylepšení a zdokonalení systému.

3. Seznam použitých zkratek

Tabulka 1: Seznam použitých zkratek

LED	Light - Emitting Diode
IDE	Integrated Development Environment
USB	Universal Serial Bus
GND	Ground (uzemnění)
A/D	Analog/Digital
IoT	Internet of Things
ICSP	In Circuit Serial Programming
DC Motor	Direct Current Motor
GSM	Groupe Spécial Mobile
GPS	Global Positioning System
Wi-Fi	Wireless Fidelity
PCB	Printed Circuit Board
RPM	Revolutions Per Minute

4. Teoretická část

4.1. Arduino

Arduino je malý jednodeskový počítač založený na mikrokontrolerech ATmega od firmy Atmel. Základním principem je jednoduché ovládání hardwaru i softwaru. Arduino dokáže číst vstupy jako např. zachycení světla pomocí senzoru, zmáčknutí tlačítka nebo přijetí Twitter zprávy a na základě toho aktivovat motor, rozsvítit LED diodu nebo publikovat příspěvek online. Můžete říct svému Arduino, co má dělat pomocí instrukcí mikrokontroleru na desce. K tomu je využíván programovací jazyk Arduino (založený na jazyku Wiring) a Arduino Software (IDE) založené na Processingu. (What is Arduino?, 2018)

Existuje více verzí Arduino desek. Mezi základní tzv. vstupní úroveň patří desky Arduino UNO, Arduino LEONARDO, Arduino NANO a Arduino MICRO. Tyto desky patří mezi nejpoužívanější, hlavně díky své jednoduchosti a malé velikosti. Mezi tzv. pokročilé desky patří například model Arduino MEGA, který má 54 digitálních a 16 analogových pinů a je tím pádem určen k řešení komplexnějších projektů, kde je potřeba připojit mnoho periférií. (Arduino Products, 2021)

K Arduino je možné připojit Arduino shieldy, které rozšiřují možnosti platformy. Mezi oficiální patří například Ethernet shield, WiFi shield umožňující připojení Arduina k internetu. Dále také Relé nebo Motor shield zajišťující snadné ovládání elektromotorů. (Voda, 2017)

4.1.1. Historie Arduina

Od startu Arduino projektu v roce 2005 do roku 2013 se prodalo přes 700 000 (Selecký, 2016) oficiálních desek. Počet prodaných neoficiálních kopií byl více než dvojnásobný, přibližně přes několik milionů Arduino desek. V dnešní době už je nedohledatelný i přibližný počet kvůli obrovské produkci neoficiálních kopií v Číně. Popularita Arduina roste víc a víc díky velké komunitě, která realizuje mnoho úžasných projektů za pomoci této open source platformy. Práce s platformou je jednoduchá, rychlá a relativně snadno naučitelná. Největší výhodou Arduina, oproti jiným platformám je, že ji mohou celkem snadno používat ke svým projektům i lidé bez technického vzdělání a potřeby dopodrobna rozumět elektronickým souvislostem. Kolem Arduina se vytvořila obrovská komunita sdílející si své kódy a návrhy zapojení. (McRoberts, 2010)

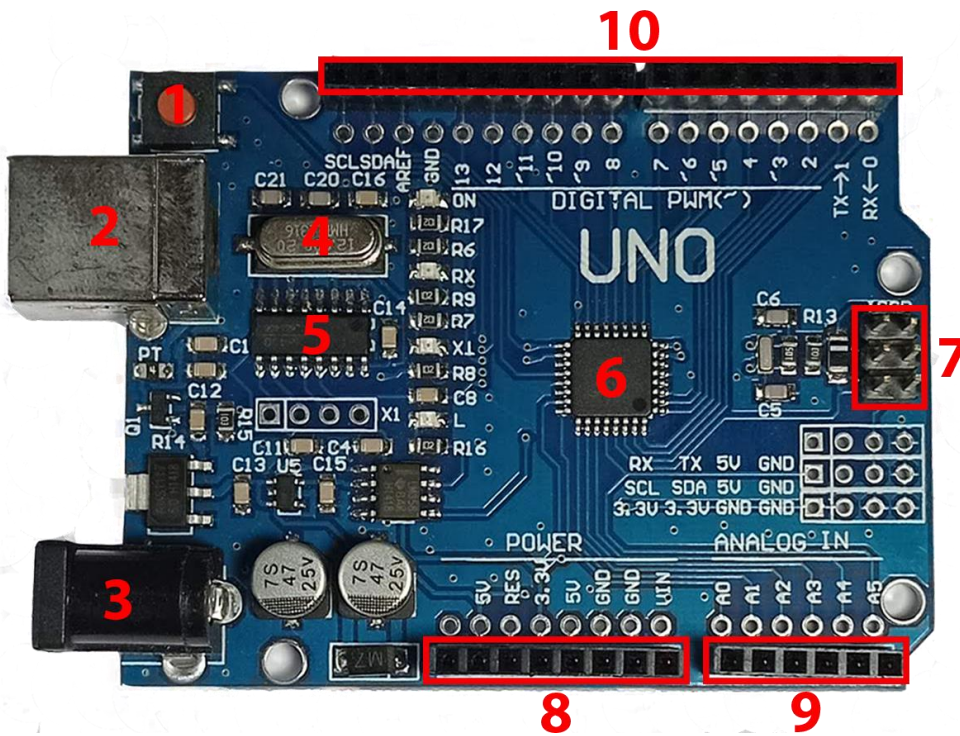
4.1.2. Arduino UNO

Nejpoužívanější deskou je Arduino UNO, disponuje poměrně velkým množstvím vstupních a výstupních pinů a dá se k němu připojit mnoho periférií, proto jsem si tento typ vybral k realizaci mého projektu.

Parametry desky Arduino UNO:

- Pracovní napětí: 5 V
- Vstupní napětí: 7 – 12 V
- Maximální vstupní napětí: 6 – 20 V
- Max. proud na 1 pin: 20 - 40 mA
- Počet digitálních vstupně-výstupních pinů: 14 (6 z nich podporuje PWM)
- Počet analogových vstupů: 6
- Flash paměť: 32 kB
- EEPROM: 1 kB
- SRAM: 2 kB (Aqeel, 2018)

Obrázek 1: Arduino UNO



Zdroj: Vlastní tvorba, 2021

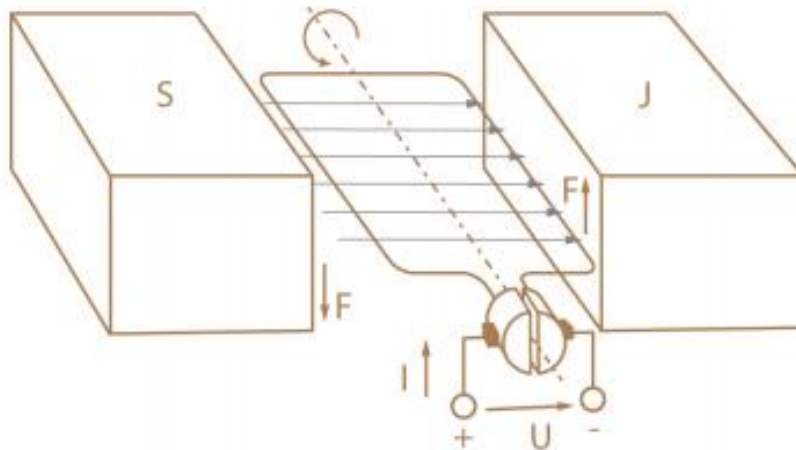
- 1) Tlačítko reset umožňující restartování nahraného programu
- 2) USB konektor pro připojení k počítači
- 3) Napájecí konektor 5V
- 4) Krystalový oscilátor
- 5) USB převodník
- 6) Atmega328 mikrokontroler
- 7) ICSP hlavice pro externí programování.
- 8) Napájecí výstupy
- 9) Analogové piny
- 10) Digitální piny

4.2. Stejnosměrný motor

Stejnosměrný motor neboli DC motor (direct current motor) je točivý stroj napájený stejnosměrným proudem. Skládá se ze statoru (pevná část) a rotoru (pohyblivá část), zvaného též kotva. U nejjednodušších stejnosměrných motorů tvoří stator permanentní magnety, tyto motory patří do skupiny s cizím buzením. Rotor neboli kotva je tvořena elektromagnety. Pomocí komulátoru je měněn směr procházejícího proudu kotvou. Komulátor je tvořen uhlíkovými kartáči, které jsou permanentně spojeny se státorem a vodivými lamelami, které rotují společně s kotvou. Směr a rychlost otáčení se reguluje pomocí H-můstku.

Princip stejnosměrného motoru objevil v roce 1873 Zénobe Gramme. Historicky patří mezi nejstarší elektrické stroje, které se z počátku používaly jako generátory pro výrobu stejnosměrného proudu (například Dynamo). V současné době jsou postupně nahrazovány střídavými regulovanými pohony s asynchronními motory, a to zejména proto, že jsou složitější a nákladnější. (Navrátil, 2015)

Obrázek 2: Princip stejnosměrného motoru



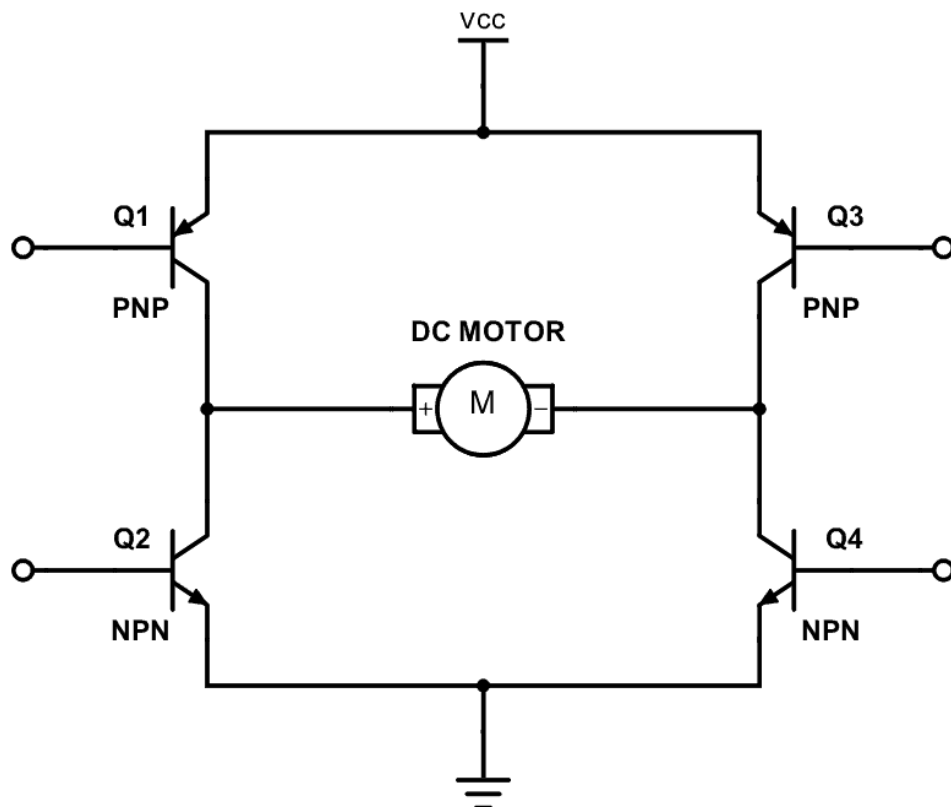
Zdroj: Převzato z Novák, M., & Pech, J. Robotika pro střední školy: Programujeme Arduino. Jihočeská univerzita v Českých Budějovicích, Pedagogická fakulta, 2020

Na obrázku 2 je vidět princip stejnosměrného motoru. Ke svorkám motoru je přiváděn stejnosměrný proud, který prochází vodiči kotvy. Protože se vodiče nachází v magnetickém poli, které tvoří permanentní magnety statoru, působí na ně síla, díky které se motor otáčí. Základním jevem je zde indukce. (Novák & Pech, 2020)

4.3. H-můstek

H-můstek je elektrický obvod tvořený tranzistory sloužící pro řízení stejnosměrného motoru. Příklad zapojení tohoto obvodu je na obrázku 3. Obvykle se prodávají moduly obsahující dva H-můstky. Tyto moduly se dají použít k řízení jednoho krokového motoru nebo dvou stejnosměrných motorů (rychlost i směr otáčení). K H-můstku se připojuje externí napájení pro daný motor, protože nelze připojit přímo k Arduino, které na svých pinech neposkytuje potřebný proud, zejména při spuštění motoru. H-můstek zároveň slouží jako ochrana řídicí desky proti zpětnému proudu, který by mohl motor generovat a poškodit tak nějakou část Arduina. (Paštyka, 2017)

Obrázek 3 Příklad zapojení H-můstku

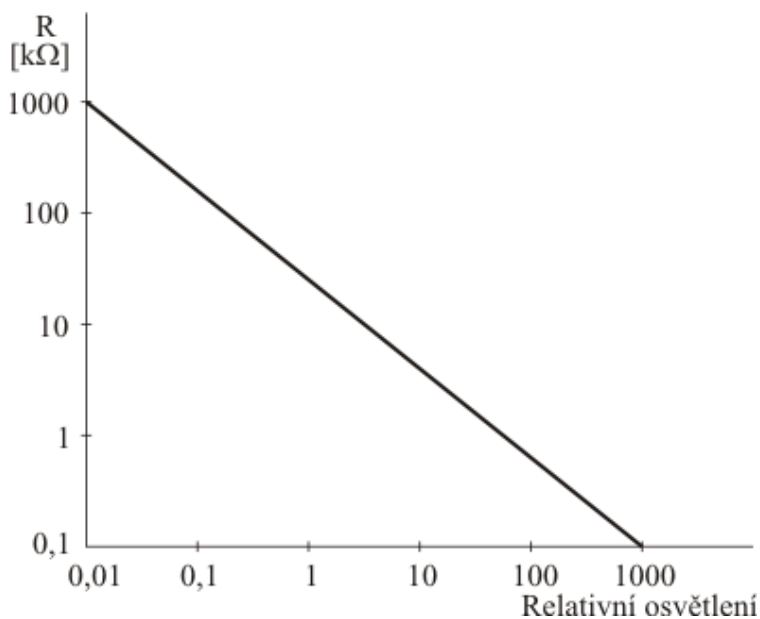


Zdroj: Převzato z webových stránek <https://www.build-electronic-circuits.com/h-bridge/>, 2021

4.4. Fotorezistor

Fotorezistor je pasivní elektronická součástka bez PN přechodu. Princip je založen na fotoelektrickém jevu. Se zvyšující se intenzitou dopadajícího světla se snižuje odpor, respektive se zvyšuje elektrická vodivost. Odbor fotorezistoru se zmenšuje exponenciálně, ale lze linearizovat. Grafem závislosti elektrického odporu na osvětlení je poté v logaritmickém měřítku přímka jako na obrázku číslo 4.

Obrázek 4 Závislost odporu fotorezistoru na intenzitě osvětlení



Zdroj: Převzato z Filip Albert, 2009

Fotorezistory se liší podle svých vlastností jako je například odpor, rychlost odezvy, doba náběhu, doba sestupné hrany a rozsah provozních teplot. Velkou výhodou je snadné použití, protože je lze využít jak u stejnosměrných, tak i u střídavých obvodů. Dalšími výhodami je například citlivost nebo nízká cena. Mezi nevýhody patří dlouhá doba odezvy, hlavně při rychlé změně intenzity světla, která ale u tohoto projektu nevádí. (Frohn, 2006)

4.5. Wi-Fi modul

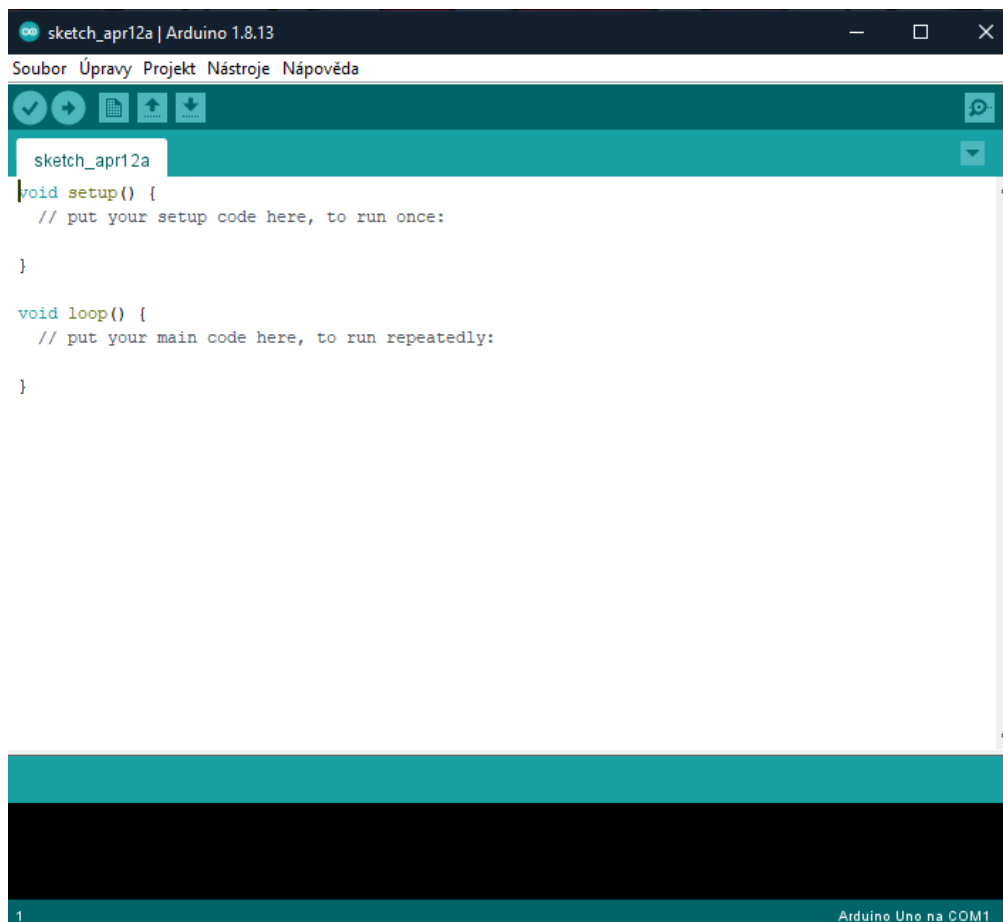
K zajištění vzdálené komunikace a propojení systému s mobilní aplikací je potřeba využít komunikační modul. Mezi běžně používané technologie patří Bluetooth nebo Wi-Fi. Bluetooth technologie by byla v tomto případě nepraktická, protože funguje na relativně krátkou vzdálenost (kolem 10 metrů) a mou snahou je, aby se zařízení dalo ovládat odkudkoli, což uživateli zajistí maximální pohodlí. Proto jsem se rozhodl využít Wi-Fi modul. K platformě Arduino je možné připojit velké množství různých Wi-Fi modulů, které se liší velikostí, druhem zapojení nebo zpracováním antény. Moduly využívající technologii Wi-Fi umožňují oboustrannou komunikaci, slouží tedy jako přijímač a vysílač zároveň. Komunikace probíhá na frekvenci 2,4 GHz.

4.6. Software

4.6.1. Arduino IDE

Arduino IDE je open source software používaný ke psaní a kompilování kódu pro Arduino moduly. Software je vydáván společností Arduino a jeho nejnovější verze je volně ke stažení na oficiálních stránkách Arduino.cc. Je dostupný pro operační systémy Windows, Linux a macOS. Běží na platformě Java a je dodáván s vestavěnými funkcemi a příkazy, které hrají zásadní roli při ladění, úpravách a kompilaci kódu v prostředí Arduino IDE. V prostředí se využívá programovací jazyk Wiring, který se používá pro programování mikrokontroleru bez specifických znalostí hardwaru. Tento jazyk vychází z programovacích jazyků C a C++. (Fezari & Al Dahoud, 2018)

Obrázek 5: Prostředí Arduino IDE

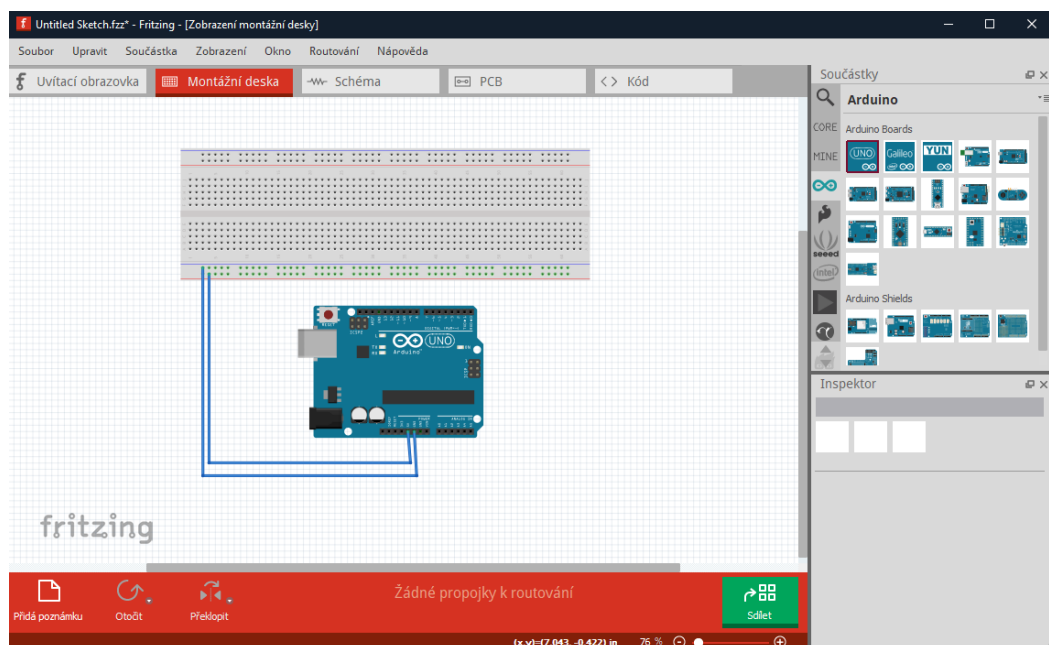


Zdroj: Vlastní tvorba, Arduino IDE, 2021

4.6.2. Fritzing

Fritzing je open source software sloužící k jednoduchému návrhu elektrického zapojení platformy Arduino. Jak je vidět na obrázku číslo 3, program nabízí režim Montážní deska, ve kterém lze jednoduše vizuálně navrhnout propojení Arduina s ostatními komponenty, kterých software obsahuje nespočet, případně je zde možnost naimportovat vlastní součástky. Druhým režimem je Schéma, ve kterém je následně zobrazeno schéma elektrického zapojení Arduina. V režimu PCB je možnost návrhu vlastního plošného spoje. Program také zastává funkci Arduino IDE a je zde možné Arduino přímo programovat. (Fritzing, 2021)

Obrázek 6: Prostředí Fritzing



Zdroj: Vlastní tvorba, Fritzing, 2021

4.6.3. Android

Operační systém Android je nejpoužívanější mobilní operační systém na světě. Systém je založen na Linuxovém jádře a je vyvíjen společností Google. Android je primárně navrhován pro smartphony a tablety. Protože Android je open source, stal se tak nejrychleji rostoucím operačním systémem. Díky své otevřené povaze se stal oblíbeným pro mnoho spotřebitelů a vývojářů. Vývojáři softwaru mohou navíc snadno upravovat a vylepšovat funkce tak, aby splňoval nejnovější požadavky mobilní technologie.

Android uživatelé stáhnou více než 1,5 miliardy aplikací a her z Google Play každý měsíc. Díky silnému vývojovému rámci uživatelé a vývojáři softwaru jsou schopni vytvářet vlastní aplikace pro širokou škálu zařízení. Některé klíčové vlastnosti operačního systému Android jsou: Práce s aplikačním rámcem, Integrovaný prohlížeč, Optimalizovaná grafika, SQLite, Podpora médií, technologie GSM, Bluetooth, Edge, Wi-Fi, fotoaparát, GPS. Pro pomoc vývojářům Android poskytuje sadu pro vývoj softwaru Android (SDK). Ta umožňuje programování v jazyce Java. Sada také obsahuje debugger, knihovny, emulátor sluchátek založený na QEMU (Quick Emulator), dokumentace, ukázkový kód a výukové programy. (Int. Journal of Engineering Research and Applications, 2014)

4.6.4. Blynk

Společnost Blynk pochází z New Yorku a vznikla v roce 2014. Zabývá se vývojem mobilní aplikace, která umožňuje propojení nejrůznějších typů zařízení. Jejich hlavním posláním, jak uvádějí na svých webových stránkách, je dělat IoT (Internet of Things) neboli Internet věcí jednodušší, přístupnější a praktičtější. (Blynk IoT platform: for business and developers, 2020)

Internet of Things (Internet věcí) nazýván také jako Internet of Everything (Internet všeho) nebo Industrial Internet (Průmyslový internet), dále jenom IoT, je novým technologickým paradigmatem představovaným jako globální síť strojů a zařízení schopných vzájemné interakce. IoT je uznáván jako jedna z nejdůležitějších oblastí budoucích technologií a získává obrovskou pozornost od široké škály průmyslových odvětví. (Lee & Lee, 2015)

Aplikace společnosti Blynk je dostupná na mobilních telefonech se systémem Android verze 4.2+ a iOS verze 9+. K propojení hardwaru s aplikací je potřeba využít rozšiřující knihovny Blynk Library, které zajistí komunikaci mezi hardwarem, Blynk Cloud serverem a vytvořenou aplikací. Je možné propojit přes 400 hardware modulů, mezi které patří i platforma Arduino. Připojení hardwaru k Blynk Cloud serveru je zajišťováno pomocí autorizačního tokenu, který je vygenerován při založení nového projektu. Příprava samotné aplikace je velice intuitivní a je zde připraveno mnoho předdefinovaných příkladů. (Blynk IoT platform: for business and developers, 2020)

4.7. Konkurenční výrobky

Na českém trhu se vyskytuje několik různých systémů pro automatické otevírání dvířek, především od zahraničních výrobců. Všechny systémy ale fungují na stejném principu a žádný z nich nenabízí propojení s mobilní aplikací k zajištění maximálního komfortu uživatele.

Mezi nejrozšířenější patří systém od britské společnosti Chicken Guard, která nabízí 3 varianty: Standard, Premium a Extreme. Tyto varianty se pohybují v cenové relaci přibližně od 4 384 Kč až do 5 531 Kč a obsahují pouze ovládací mechanismus bez dvířek, které je možné dokoupit za cenu 1 296 Kč. Nejlevnější z variant, verze Standard, ovšem nenabízí možnost ovládání pomocí světelného senzoru, ale pouze pomocí přednastaveného času. Všechny systémy je možné napájet buď pomocí USB kabelu nebo pomocí 4 AA baterií. Na obrázku 7 je ukázána nejlepší z variant Extreme, kde je na levé straně vidět výřez pro fotorezistor sloužící k zjištění intenzity venkovního světla. (Zemědělské potřeby M+S, 2021)

Obrázek 7: Systém automatického otevírání dvířek ChickenGuard Extreme



Zdroj: Převzato z webových stránek výrobce <https://www.chickenguard.com/product/extreme/>

Další alternativou vyskytující se na českém trhu je systém od německé společnosti Albert Kerbl GmbH, který je vybaven jak senzorem intenzity světla, tak nastavitelným časovačem. Umožňuje napájení ze sítě na 230 V nebo 4 x AA baterie. Balení ovšem neobsahuje dvířka jako u předešlého výrobce. Jeho pořizovací cena je přibližně 3 450 Kč. (Automatické zavírání kurníku do 2,5 kg, 2021)

Obrázek 8: Systém automatického otevírání dvířek Kerbl



Zdroj: Převzato z webových stránek prodejce <https://www.kamir.cz/>, 2021

Jediným českým zástupcem je firma Hexpeak s.r.o. s jejich systémem automatických dveří Chickdoor. Zařízení je vybaveno interním nebo externím snímačem intenzity světla v závislosti na tom, jestli se systém instaluje dovnitř nebo vně kurníku. Nabízí výběr mezi bateriovým nebo síťovým napájecím zdrojem. Cena tohoto systému se pohybuje přibližně kolem 3 349 Kč a balení tohoto produktu jako jediné obsahuje i nerezový rám včetně dveří a plastové průhledné stříšky k ochraně proti vniknutí vody. Tato společnost k datu 04.04.2021 ladí poslední detaily a chystá sériovou výrobu systému Chickdoor SMART, který by měl podporovat ovládání pomocí mobilní aplikace. (Automatická dvířka Chickdoor, 2021)

Obrázek 9: Systém automatického otevírání dveří Chickdoor



Zdroj: Převezato z webových stránek výrobce <https://chytrykurnik.cz/>, 2021

5. Praktická část

V této části se zaměřuji na návrh a samotnou realizaci automatického otevírání dveří. Popisuji jednotlivé fáze od výběru komponentů, vývoj aplikace a následné testování.

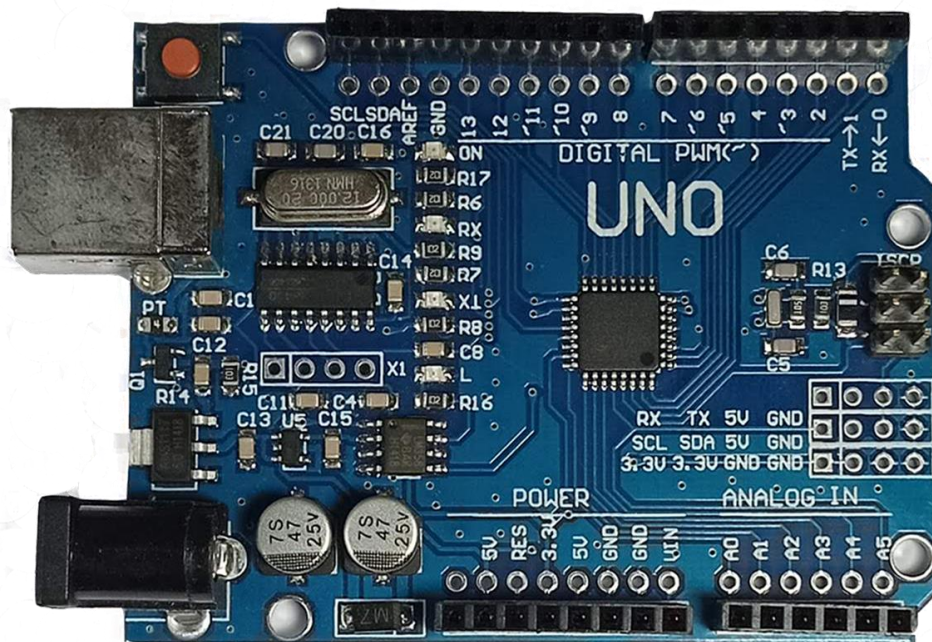
5.1. Výběr hlavních komponentů

Celá práce je založena na použití platformy Arduino. U modelu není zvláště kladený důraz na velikost, proto jsem se rozhodl použít nejpoužívanější model Arduino UNO, který mi poskytuje možnost připojení většího množství zařízení na rozdíl od Arduino NANO, které bych volil v případě potřeby zmenšování velikosti modelu. Dále k zajištění pohybu dveří jsem vybral stejnosměrný motor. K ověření skutečné polohy dveří jsem použil jazýčkové kontakty. K automatickému ovládní systému bylo potřeba měřit intenzitu venkovního světla, k čemuž jsem zvolil fotorezistor. Nakonec bylo potřeba zajistit bezdrátovou komunikaci s aplikací pomocí Wi-Fi modulu. Komponenty byly vybírány s ohledem na finanční nenáročnost a dostupnost.

Řídící deska Arduino UNO

Jak jsem výše zmiňoval, nebyl jsem omezen velikostí řídicí desky, proto jsem zvolil variantu Arduino UNO, která je nejrozšířenější a má proto také nejlepší technickou podporu v případě, že by nějaké komponenty nefungovaly správně. Deska je osazena USB 2.0 type B konektorem k přímému propojení s počítačem a není tak potřeba žádný externí programátor k nahrávání programu. Podrobnější popis desky je na obrázku 1 na straně 7. Z důvodu snižování nákladů jsem zvolil klon originálního Arduina UNO, který má stejné parametry, tato možnost je tu díky tomu, že platforma Arduino je open source.

Obrázek 10: Arduino UNO



Zdroj: Vlastní fotografie, 2021

Stejnoseměrný motor JGA25-370

K pohonu celého ústrojí jsem zvolil kartáčový stejnosměrný motor JGA25-370 od výrobce Eyewink. Motor je opatřen převodovkou, dané poměry a parametry jsou uvedeny v tabulce číslo 2. Vhodné napětí pro tento motor je 12 V. Průměr motoru je 25 mm, průměr hřídele 4 mm a hmotnost 80 g. (Motor JGA25-370 12V s převodovkou)

Motor byl volen tak, aby byl schopen utáhnout bez problémů i těžší dvířka. I když základním předpokladem je, že dvířka obydlí by měla vážit do 1 kg.

Obrázek 11: Stejnoseměrný motor JGA25-370



Zdroj: Vlastní fotografie, 2021

Tabulka parametrů motoru JGA25-370

Tabulka 2: Parametry motoru JGA25-370

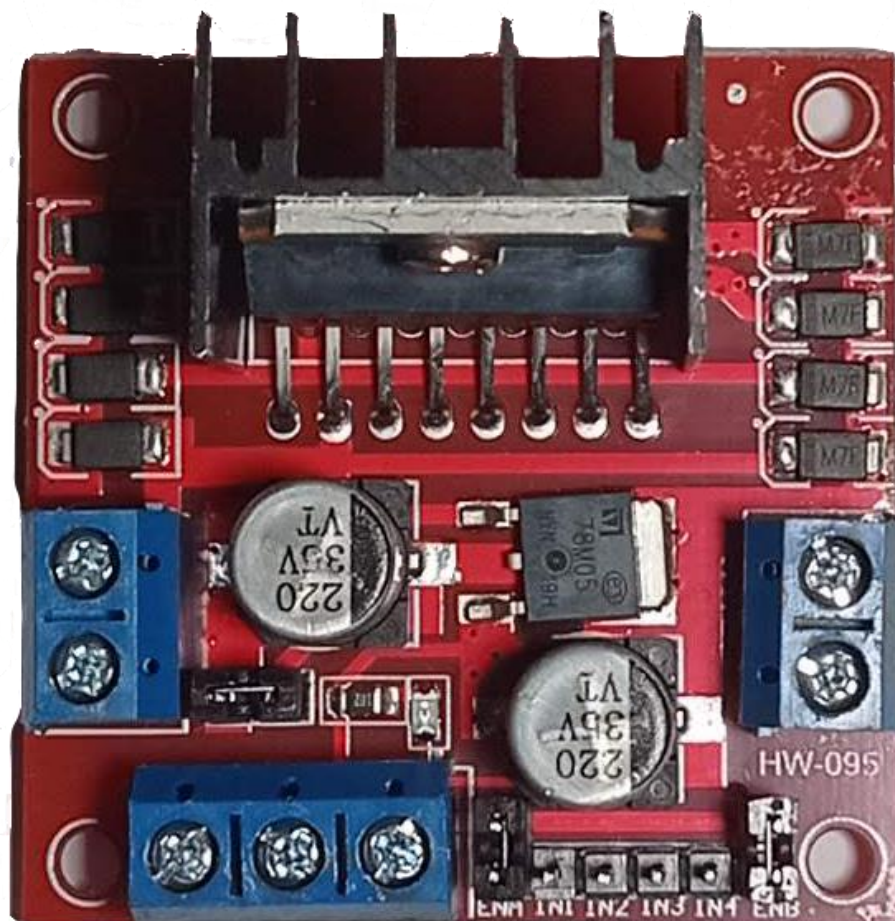
Redukční poměr	4.4	9.6	21.3	35	45	78	103	171	226	377	500
Proud bez zatížení (mA)	60	60	60	60	60	60	60	60	60	60	60
Otáčky bez zatížení (RPM)	1360	620	280	170	130	77	60	35	26	16	12
Jmenovitý točivý moment kg*cm	0.1	0.22	0.5	0.8	1	1.8	2.4	4	5.2	8.4	9
Jmenovité otáčky (RPM)	1000	450	220	130	100	60	46	27	20	12	9
Jmenovitý proud (A)	0.45	0.45	0.45	0.45	0.45	0.45	0.45	0.45	0.45	0.45	0.45
Maximální točivý moment (kg*cm)	0.35	0.75	1.7	2.8	0.36	6.2	8.2	9	Maximální zatížení 9 kg		
Stop proud (A)	1.3	1.3	1.3	1.3	1.3	1.3	1.3	1.3	Nepodporuje stání		
Délka	17		19	21		23		25		27	

Zdroj: Převzato ze webových stránek prodejce Laskakarduinu.cz, 2021

Modul H-můstku L298N

K řízení rychlosti a směru otáčení DC motoru jsem zvolil H-můstek L298N. Tento modul může řídit jeden dvoufázový krokový motor, jeden čtyřfázový krokový motor nebo dva stejnosměrné motory. Maximální proud zatížení je 3A, při dlouhodobé zátěži maximálně 2A. Pro logické řízení modulu je využíváno napětí 5V. Maximální napájecí napětí je 46V a napětí na výstupu lze regulovat pomocí řídicí logiky od 5V do 35V. Tento modul obsahuje také vestavěný 5V stabilizátor napětí, který je možné použít k napájení Arduino desky. (SparkFun Electronics, 2000)

Obrázek 12: H-můstek L298N

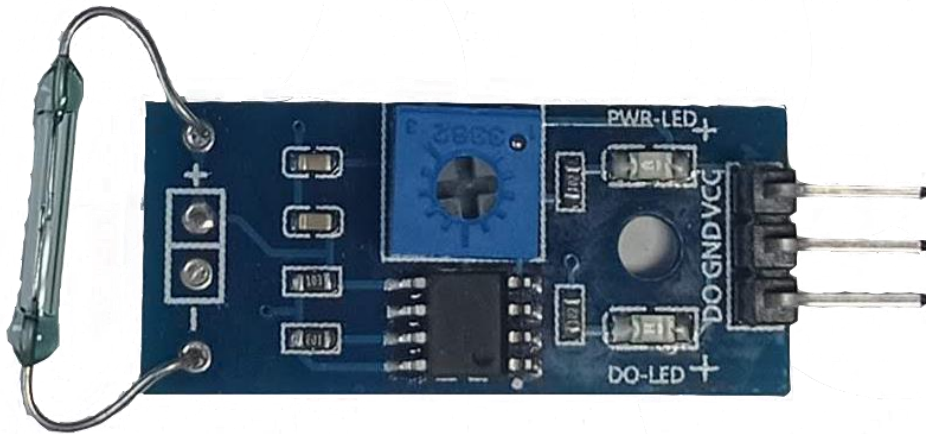


Zdroj: Vlastní fotografie, 2021

Jazýčkový kontakt

Z bezpečnostních důvodů jsem využil na horní a spodní hraně otvoru jazýčkové kontakty, které budou kontrolovat, zdali jsou dveře opravdu zavřené nebo otevřené. Jazýčkový kontakt je mechanický spínač reagující na přítomnost magnetického pole, proto na horní a spodní straně dvířek jsou umístěny malé magnety, které aktivují tyto snímače. Použitý modul zahrnuje jazýčkový kontakt a komparátor. Na desce je trimr, který slouží k nastavení citlivosti. Výstupem je logická 0 nebo 1, přičemž 0 znamená sepnutý kontakt při proudu 15mA. Provozní napětí je 3,3V – 5V. Deska obsahuje předvrtaný otvor pro upevnění.

Obrázek 13: Jazýčkový kontakt s komparátorem LM393

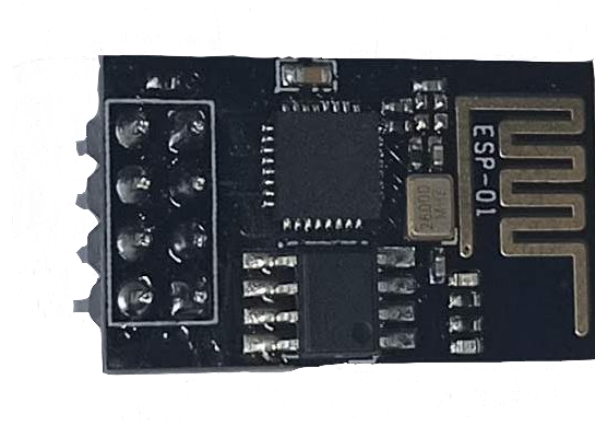


Zdroj: Vlastní fotografie, 2021

Komunikační modul

Pro komunikaci systému s Blynk Cloud serverem a následně s aplikací jsem využil Wi-Fi modul ESP-01. Jak je vidět na obrázku číslo 14, modul má integrovanou anténu přímo na desce. K napájení musí být použito napětí 3,3V, vyšší napětí by mohlo modul poškodit. Modul má integrovaný 32-bitový procesor o frekvenci 160 MHz a 8 Mb flash paměť, lze jej tedy využít i samostatně ke spuštění jednoduché aplikace. K přenášení dat využívá pásmo 2,4 GHz.

Obrázek 14: Wi-Fi modul ESP-01



Zdroj: Vlastní fotografie, 2021

Fotorezistor

K měření intenzity světla k následnému rozeznání dne a noci jsem použil fotorezistor. Jak z důvodu jednoduchého zapojení, tak i kvůli nízké ceně. Nevýhodou bývá dlouhá odezva při velkých změnách intenzity světla, což se ale netýká tohoto projektu, kde stmívání bude pozvolné.

Zdroj napájení

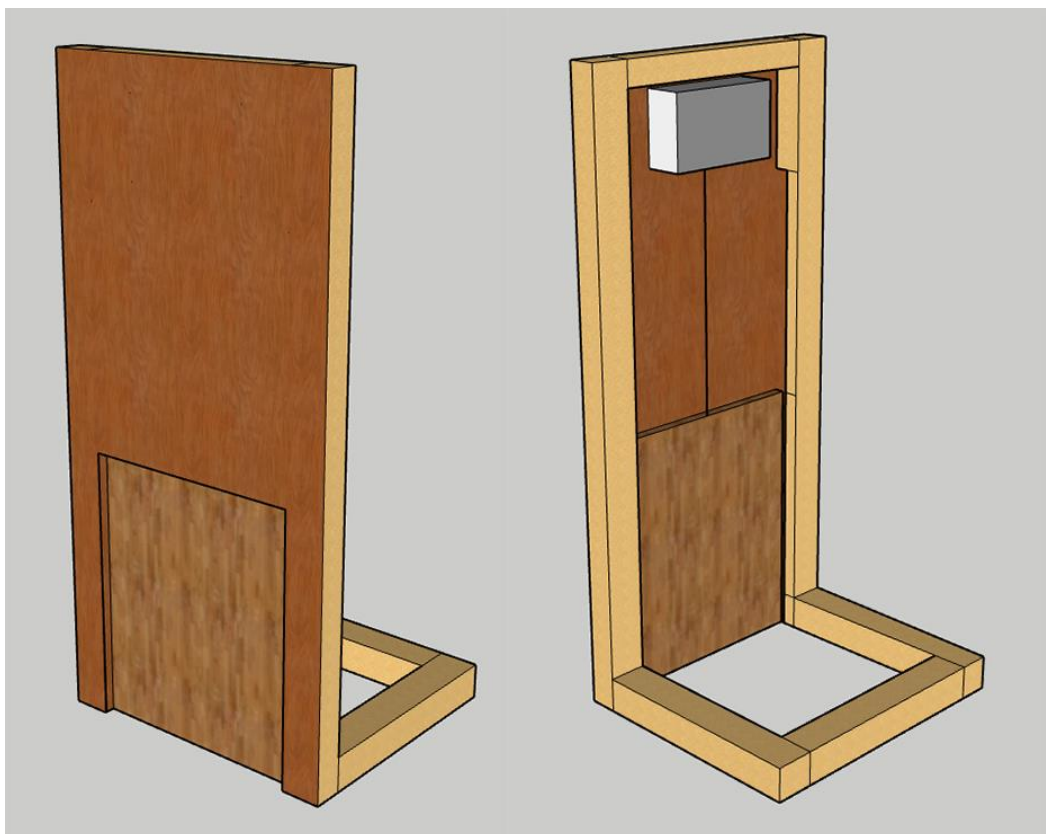
Systém automatického ovládání dvířek je celý napájen ze sítě. K tomuto řešení jsem se rozhodl z důvodu eliminace možných problémů s bateriemi nebo solárními články, u kterých by mohlo docházet ke kolísání napětí, nedostatečnému proudu, případně vlivem povětrnostních podmínek ke snižování kapacity baterie, což by vedlo k ohrožení bezpečnosti kurníku. Zároveň by tato řešení vyžadovala rozsáhlé propočty a úpravy jak použitých komponentů, tak i samotného programu.

Hlavním problémem je napájení stejnosměrného motoru JGA25-370. K zajištění napětí 12V jsem použil regulovatelný stejnosměrný zdroj MW9115GS. Výstupní napětí tohoto zdroje se dá snadno regulovat pomocí otočného přepínače na 3V/4,5V/6V/7,5V/9V/12V. Výstupní proud je 1,5A. K napájení platformy Arduino jsem použil USB kabel připojený do standardního 5V adaptéru, který se používá například k nabíjení mobilních telefonů.

5.2. Návrh modelu

Vhodnými materiály ke konstrukci modelu by mohla být kombinace hliníkových profilů a pozinkovaného plechu, případně tvrzeného plastu. Tyto materiály jsou velice odolné vůči povětrnostním podmínkám, především pak vlhkosti, která by časem mohla poškodit dřevěná dvířka. Práce s těmito materiály je ale složitější, hlavně z důvodu, že neodpustí sebemenší nepřesnosti. Proto jsem se rozhodl ke konstrukci modelu zvolit jako materiál dřevo, práce s ním je jednodušší a v případě komplikací například se zasekáváním dvířek je možné díl rychle nahradit, případně upravit. Na obrázku 15 je návrh modelu, vytvořený v programu SketchUp.

Obrázek 15: 3D návrh modelu



Zdroj: Vlastní tvorba, SketchUp, 2021

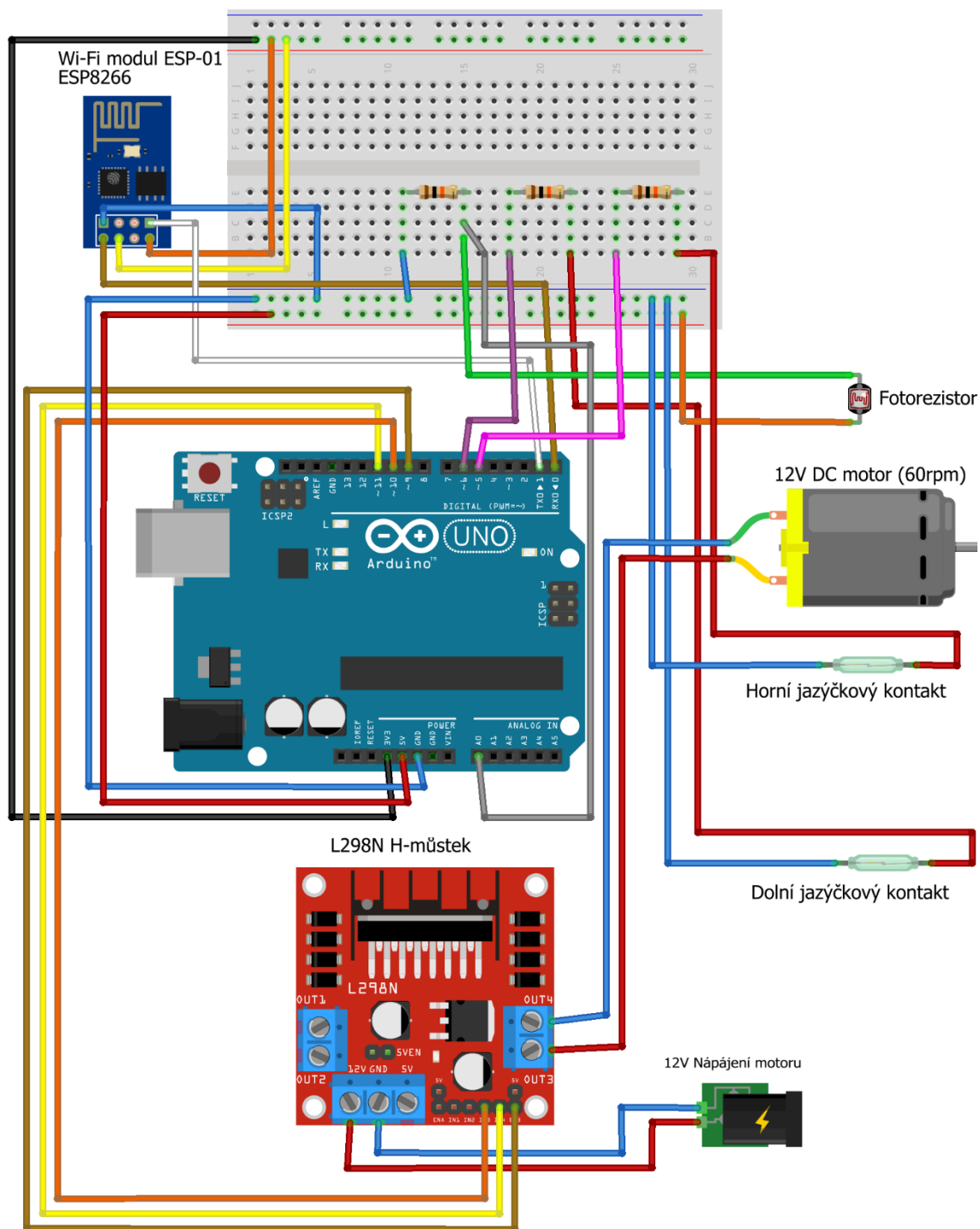
Jak je patrné z obrázku 15, konstrukce předváděcího modelu je velmi jednoduchá. Nosný rám je zhotoven ze smrkových hranolů, které jsou pevně spojeny pomocí vrutů do tvaru písmene L. Zezadu na vnitřní straně hranolů jsou vyříznuty drážky umožňující pohyb dvířek pouze vertikálním směrem. Přední čelo modelu, znázorňující vnější stěnu kurníku, je vyříznuto z truhlářské překližky z důvodu nízké hmotnosti, která přispěje ke snadnému manipulování a převozu celého modelu. Dvířka jsou taktéž z truhlářské překližky, která je dostatečně pevná a zároveň lehká, což eliminuje možnost přehřívání motoru, které by mohlo vést k poruše motoru nebo řídicí elektroniky a tím k ohrožení bezpečnosti kurníku. Na zadní straně je připevněná ochranná plastová krabice obsahující řídicí elektroniku, včetně motoru s navíjecím zařízením.

Model slouží pouze jako ukázka projektu. Při reálném použití v kurníku by se rozměry a uchycení dvířek muselo přizpůsobit daným možnostem. Například by nebylo možné umístit dvířka s řídicí elektronikou uvnitř kurníku, kvůli nedostatku místa. Řešením by bylo umístit systém z venkovní strany kurníku zároveň tedy i hranoly s vodícími drážkami dvířek.

5.3. Elektrické zapojení

Před samotnou realizací jsem si připravil vizualizaci a schéma elektrického zapojení v programu Fritzing, který jsem zmiňoval v teoretické části. Během vytváření bylo nutné vhodně rozvrhnout jednotlivé piny Arduina. Například DC motor řízený pomocí H-můstku L298N je nutné připojit na digitální piny s PWM modulací, která je pouze na pinech 3,5,6,9,10 a 11. Tyto piny jsou na desce značeny vlnovkou. Komunikační modul ESP-01 má při programování prohozené piny RX a TX, které jsou u Arduina umístěné pod čísly 0 a 1. Na obrázku 16 je znázorněno výsledné zapojení po procesu programování, které je použito v mém modelu. Na obrázku vizualizace elektrického zapojení není vidět napájení platformy Arduino pomocí USB kabelu a 5V adaptéru. Vizualizaci zapojení pinů jsem dále používal při programování aplikace. Elektrické schéma zapojení je vidět na obrázku číslo 17.

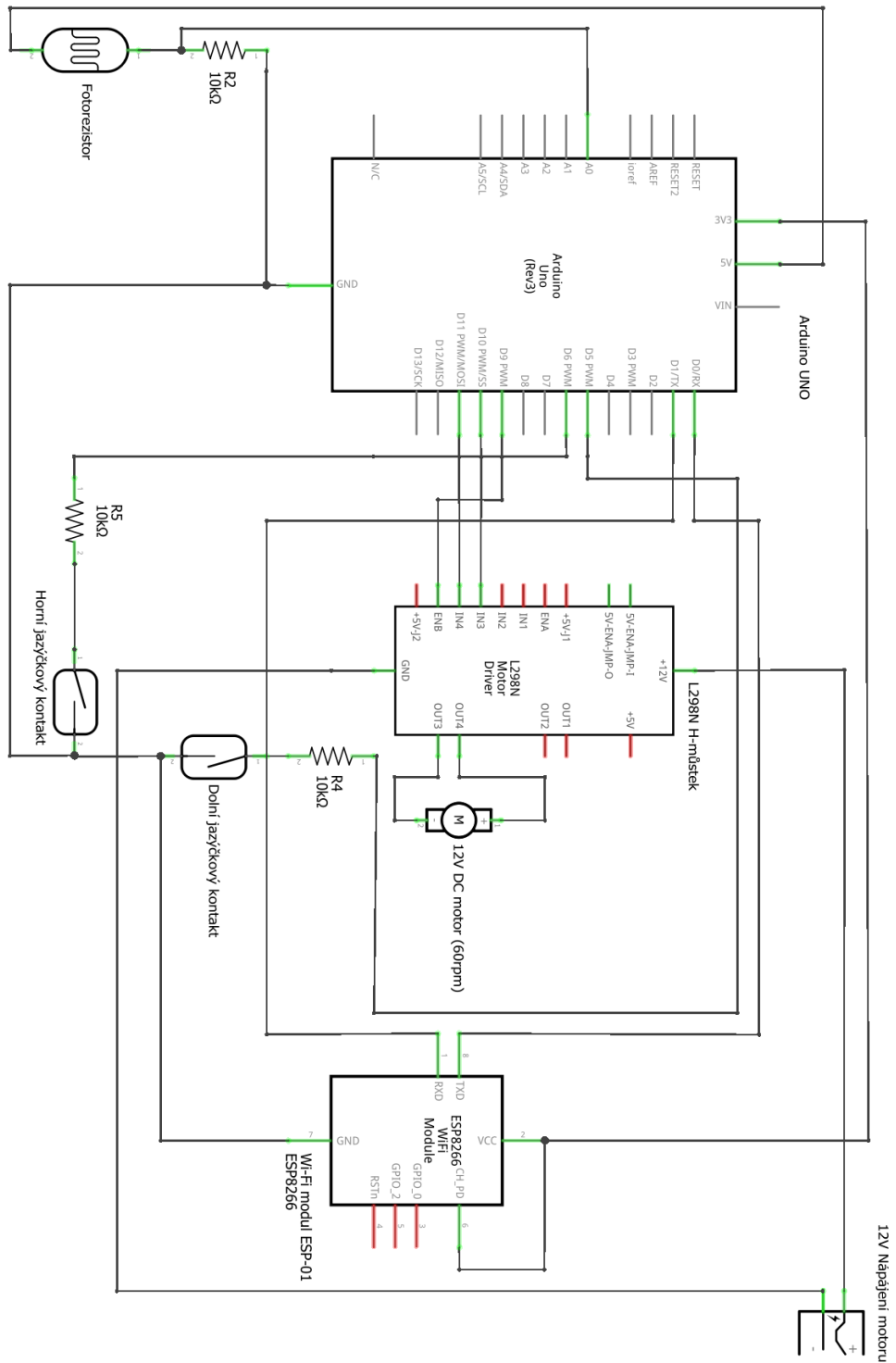
Obrázek 16: Vizualizace elektrického zapojení



Zdroj: Vlastní tvorba, Fritzing, 2021

5.4. Schéma zapojení

Obrázek 17: Schéma elektrického zapojení



Zdroj: Vlastní tvorba, Fritzing, 2021

5.5. Vývoj aplikace

K vývoji jsem se rozhodl využít mobilní aplikaci americké společnosti Blynk. Tato společnost se zabývá vývojem své aplikace od roku 2014. Nabízí propojení více než 400 modulů a zároveň spolupracuje se společností Arduino, jejíž výrobek ve svém systému používám. Je to tedy ideálním řešením k zajištění maximální kompatibility a spolehlivosti systému.

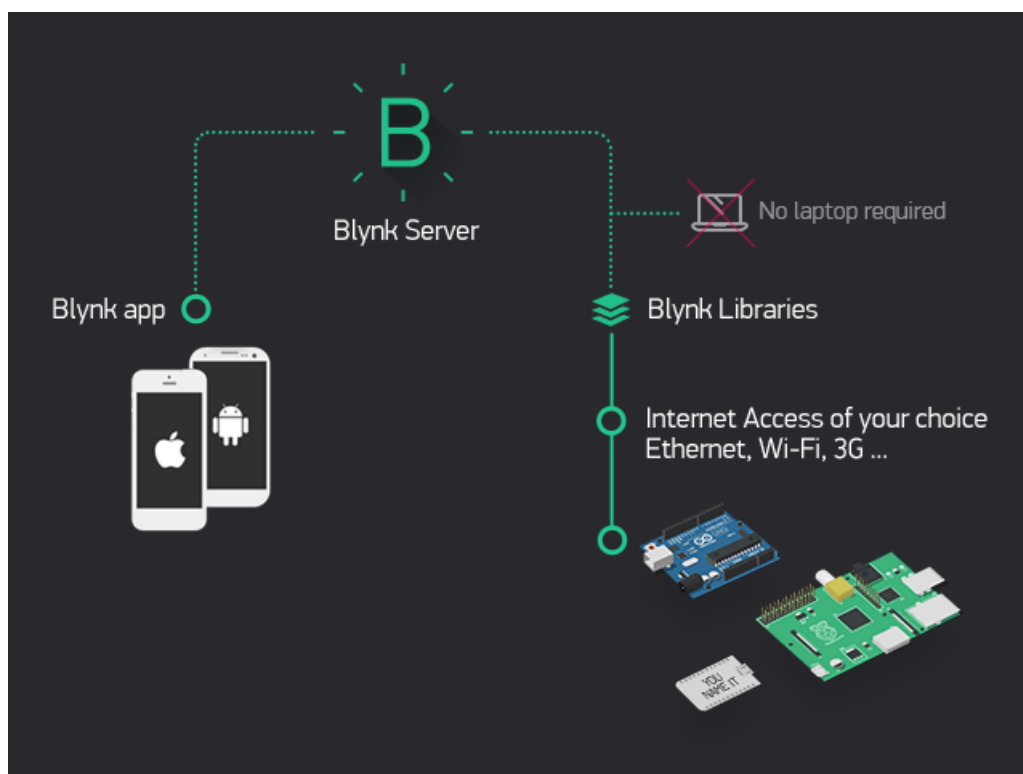
Blynk nabízí 3 cenové kategorie. První kategorií je „Developer Plan“, který je určený pro vývojáře a soukromé užívání. Tato kategorie je zcela zdarma, a proto jsem se rozhodl využít právě toto řešení. Zahrnuje mobilní aplikaci pro Android nebo iOS, přístup k Blynk Cloud a uchovávání dat, možnosti připojení až 5 zařízení a knihovny pro práci s jakýmkoliv zařízením.

Pokud je aplikace vyvíjena ke komerčním účelům, je zde připraven „Startup Plan“, který zahrnuje publikování aplikace na Google Play a App Store pod jménem vlastní společnosti, logem a vlastní ikonou. Je zde umožněn přístup k Business Cloud Serveru, neomezené množství připojených zařízení a uživatelů, odborná pomoc s vydáním aplikace. Tato kategorie je zpoplatněná měsíční sazbou od 415 dolarů měsíčně, což je přibližně 8 900 Kč.

Třetí kategorie „Business Plan“ zahrnuje vše jako výše zmíněný „Startup Plan“ a k tomu nabízí vytvoření webové aplikace rozšířenou o možnost správy všech svých zařízení, uživatelů a organizací. Zároveň je zde přístup k datové analytice. Cena této kategorie je od 1 000 dolarů měsíčně neboli přibližně 21 500 Kč. (Blynk IoT platform: for business and developers, 2020)

Před samotným vývojem jsem musel zjistit, jak vůbec funguje celé propojení mobilní aplikace s platformou Arduino. Jak je vidět na obrázku číslo 18, internetové propojení může být zajištěno jakýmkoliv způsobem například Ethernet kabelem, bezdrátově pomocí Wi-Fi případně mobilní sítí, záleží na použitém modulu. Do hlavního programu, který se nahrává do řídicí platformy, musí být importovány Blynk knihovny a následně musí být použity modifikované příkazy, aby aplikace byla schopna dané zařízení správně ovládat. Do programu se také přidává autorizační token, který je vygenerován při založení nového projektu v mobilní aplikaci Blynk. Ten slouží k autorizaci zařízení při připojení k Blynk Cloud serveru a přiřazení ke správnému projektu. Blynk server zajišťuje oboustrannou komunikaci mezi vyvíjenou aplikací a ovládanou platformou. (Blynk IoT platform: for business and developers, 2020)

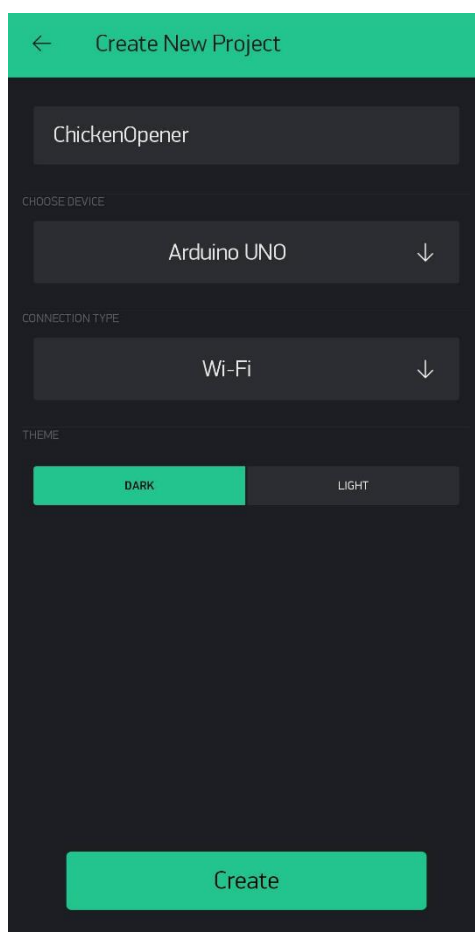
Obrázek 18: Schéma propojení platformy s aplikací Blynk



Zdroj: Převzato z webových stránek Blynk <https://blynk.io/>, 2021

K vývoji mé aplikace jsem využil mobilní telefon Xiaomi Note 9 LTE s operačním systémem Android verze 10, na který jsem nainstaloval z oficiálního obchodu Google Play mobilní aplikaci Blynk verze 2.27.28. Při otevření aplikace je třeba se zaregistrovat standartně pomocí e-mailu nebo je zde možnost pomocí Facebook účtu. Po otevření aplikace jsem si vytvořil nový projekt, jak je znázorněno na obrázku číslo 19.

Obrázek 19: Blynk - založení projektu

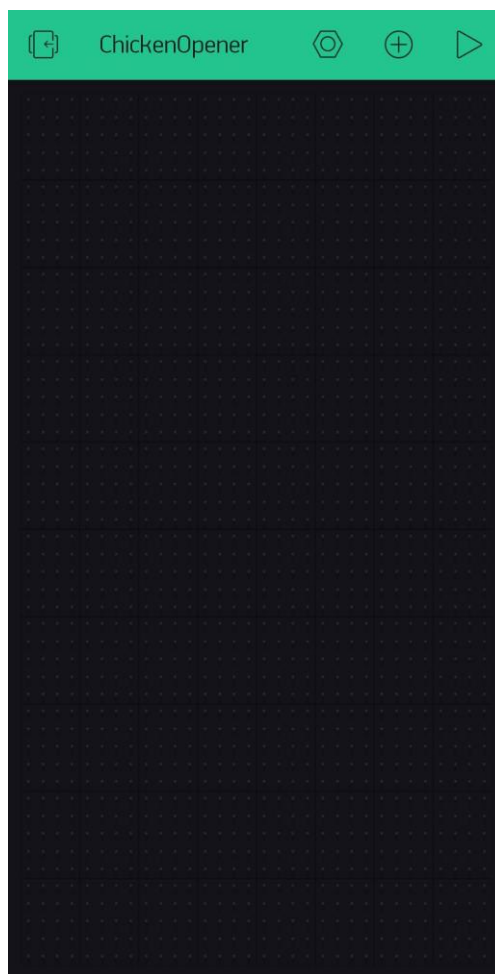


Zdroj: Převezato z aplikace Blynk, 2021

Jako pracovní název jsem zvolil Chicken Opener. Dále jsem zvolil Arduino UNO, jako zařízení, které budu touto aplikací ovládat. Poslední možností je typ komunikace, který bude pomocí Wi-Fi modulu, jak je zmiňováno výše v této práci.

Po vytvoření projektu mi přišel na uvedenou e-mailovou adresu autorizační token, který poté implementuji do zdrojového kódu Arduino aplikace. Můj vygenerovaný autorizační kód je: 9OywXkshTqcSshFzbpYr1uCzv6kzDQUY. Na obrázku číslo 20 je pracovní plocha, která se zobrazí po vytvoření projektu a na které bude probíhat návrh vyvíjené aplikace.

Obrázek 20: Blynk - pracovní plocha

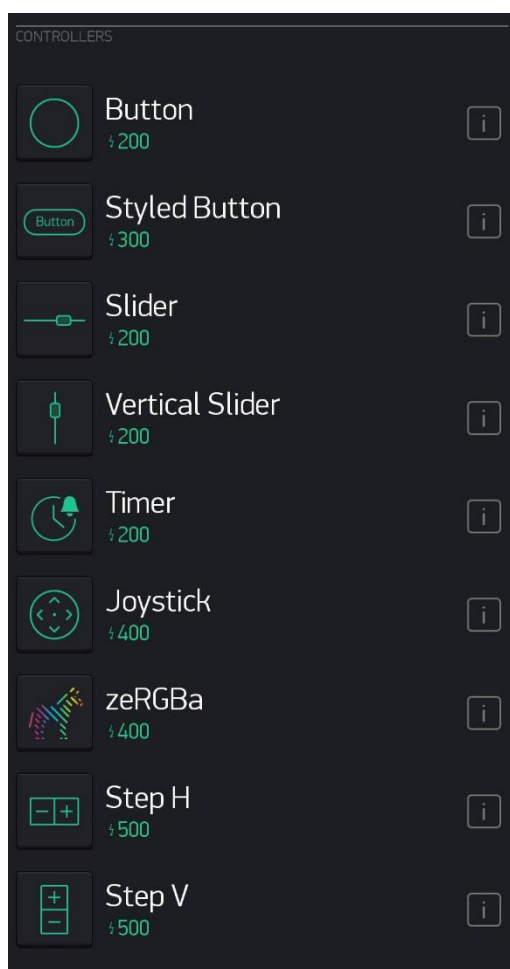


Zdroj: Převzato z aplikace Blynk, 2021

Aplikace Blynk nabízí velké množství takzvaných „widgetů“, což jsou jednoduché předdefinované prvky, pomocí kterých lze vytvářet vlastní aplikaci. Widgets jsou rozděleny dle své funkce do kategorií. V další části představím pouze kategorie využitelné pro mou aplikaci.

První kategorií jsou Controllers, které obsahují ovládací prvky. Dostupné možnosti jsou vidět na obrázku číslo 21.

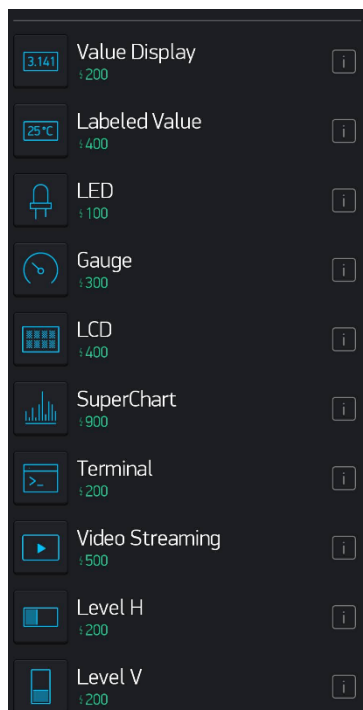
Obrázek 21: Blynk - Controllers widgety



Zdroj: Převezato z aplikace Blynk, 2021

Další kategorií jsou Displays obsahující různé ukazatele hodnot, displeje a podobné. Ukázka této kategorie je na obrázku číslo 22.

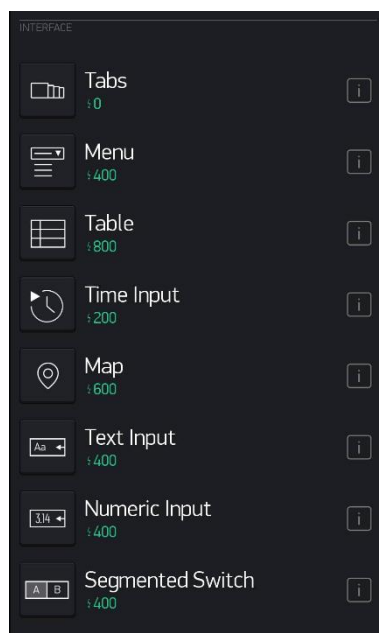
Obrázek 22: Blynk – Displays widgety



Zdroj: Převezato z aplikace Blynk, 2021

Další užitečnou kategorií je Interface obsahující například pole pro vložení času, číselné hodnoty nebo textu, jak je vidět na obrázku číslo 23.

Obrázek 23: Blynk – Interface widgety



Zdroj: Převezato z aplikace Blynk, 2021

Implementace

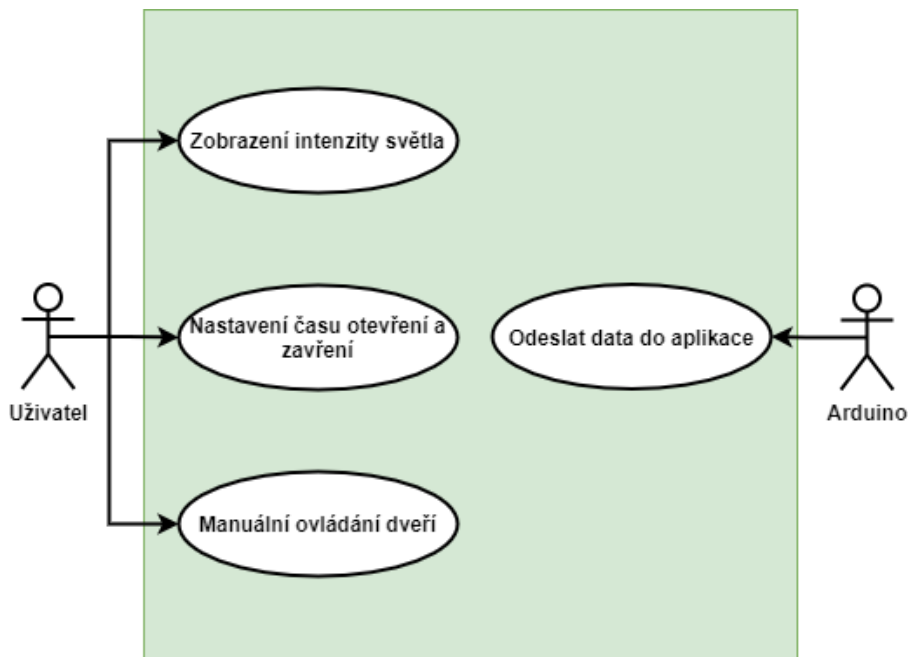
Před samotným vývojem je důležité se podrobně zamyslet, jaké funkce by měla aplikace obsahovat a pro jaké uživatele bude vyvíjena. Cílovými uživateli jsou rekreační chovatelé slepic, kteří hledají jednoduchý systém automatického otevírání dveří. Aplikace by měla být přehledná a snadno ovladatelná.

Funkční požadavky

Funkční požadavky definují základní funkce, které by měla aplikace obsahovat. Při zamyšlení jsem se soustředil na jednoduchost a maximální funkčnost aplikace. Uživatel by neměl být rušen zobrazováním zbytečných hodnot jako je například číselná hodnota intenzity světla. Proto jsem se zaměřil na následující funkční požadavky:

- Aplikace bude zobrazovat intenzitu venkovního světla ve zjednodušené formě – světlo, šero, tma.
- Uživatel bude moci v případě potřeby nastavit čas otevření a zavření.
- Aplikace bude umožňovat manuální ovládání dveří.

Obrázek 24: Digram případů užití

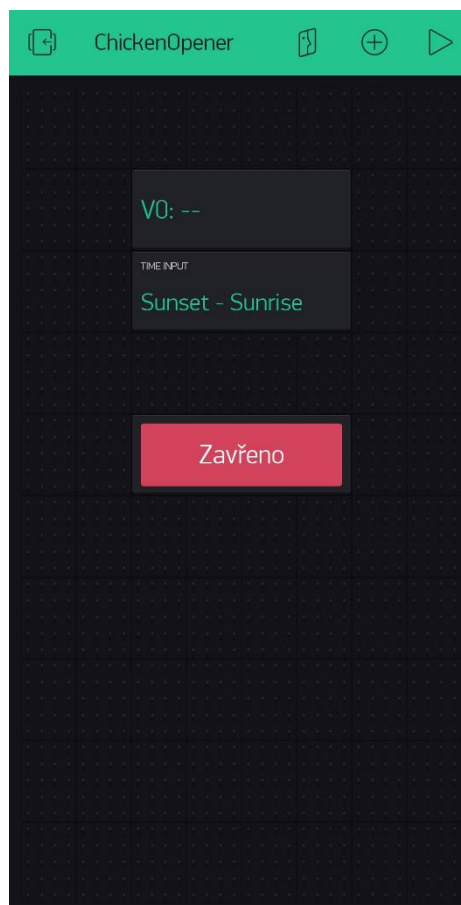


Zdroj: Vlastní tvorba, Draw.io: Diagram Software and Flowchart Maker, 2021

Návrh uživatelského rozhraní

Návrh uživatelského rozhraní je velice jednoduchý a bez zbytečných rušivých elementů. Rozvržení ovládacích prvků v aplikaci je vidět na obrázku číslo 25. Prvním prvkem je Value Display, který zobrazuje jeden ze tří stavů světlo, šero, tma, a to v závislosti na hodnotách získávaných z fotorezistoru. Druhým prvkem je Time input, který umožňuje otevření a zavření dvířek pomocí předem nastaveného času. Třetím prvkem je Button, který zajišťuje funkci manuálního ovládní dvířek. Má dva možné stavy: otevřeno a zavřeno.

Obrázek 25: Blynk - Návrh aplikace

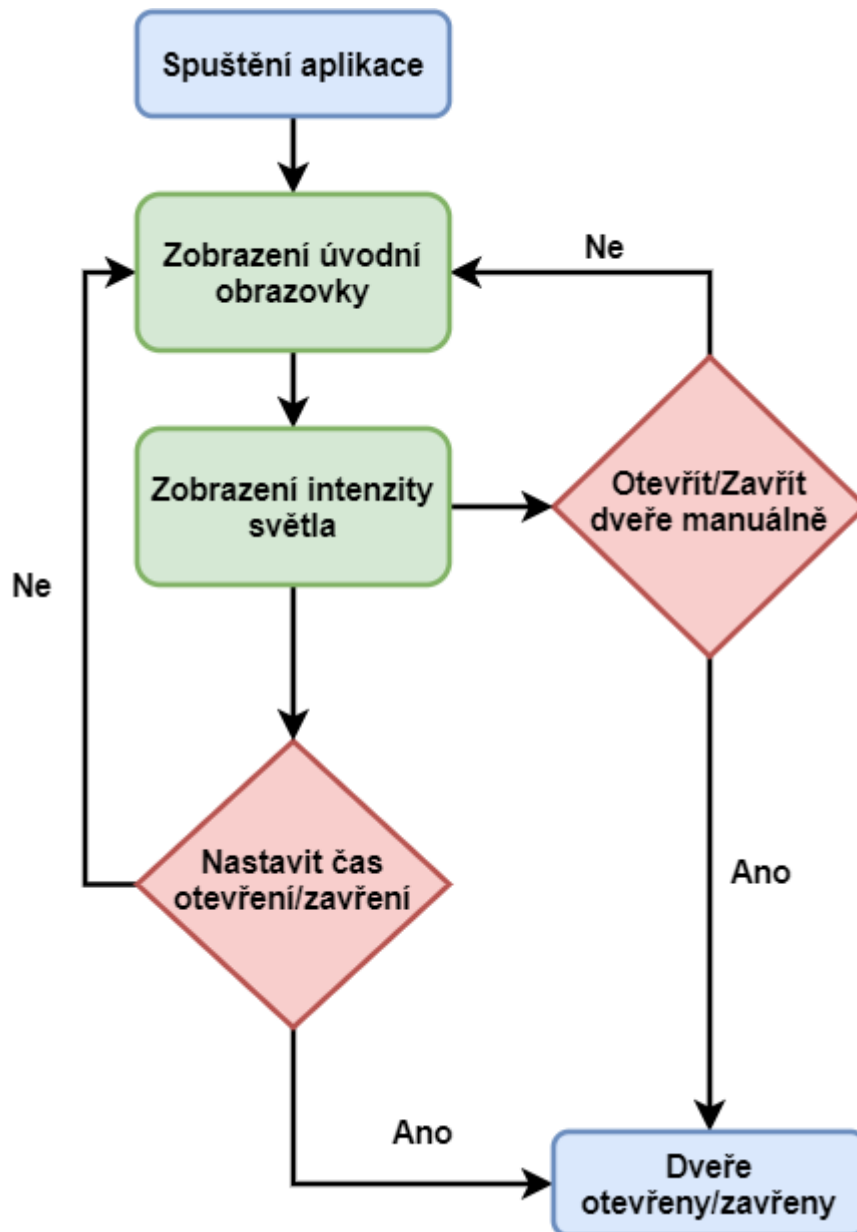


Zdroj: Vlastní tvorba, Blynk, 2021

Všem třem prvkům jsem musel přidělit výstupní virtuální piny, se kterými budu dále pracovat v rámci Arduino programu.

Vývojový diagram mobilní aplikace

Obrázek 26: Vývojový diagram mobilní aplikace



Zdroj: Vlastní tvorba, Draw.io: Diagram Software and Flowchart Maker, 2021

Vývoj aplikace Arduino

Program ovládající Arduino platformu je celý vytvořen v prostředí Arduino IDE. Pro usnadnění ovládání komunikačního modulu a možnost propojení s mobilní aplikací byly přidány externí knihovny rozšiřující prostředí o užitečné příkazy. Z důvodu velkého rozsahu zdrojového kódu jsem základní myšlenku zjednodušil a znázornil pomocí diagramu, který je na obrázku číslo 28. Celý zdrojový kód je v příloze této práce.

Obrázek 27: Ukázka kódu Arduino aplikace

```
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SoftwareSerial.h>

//Vložení autorizačního tokenu k aplikaci Blynk
char auth[] = "90ywXkshTqcSshFzbpYrluCzv6kzDQUY ";

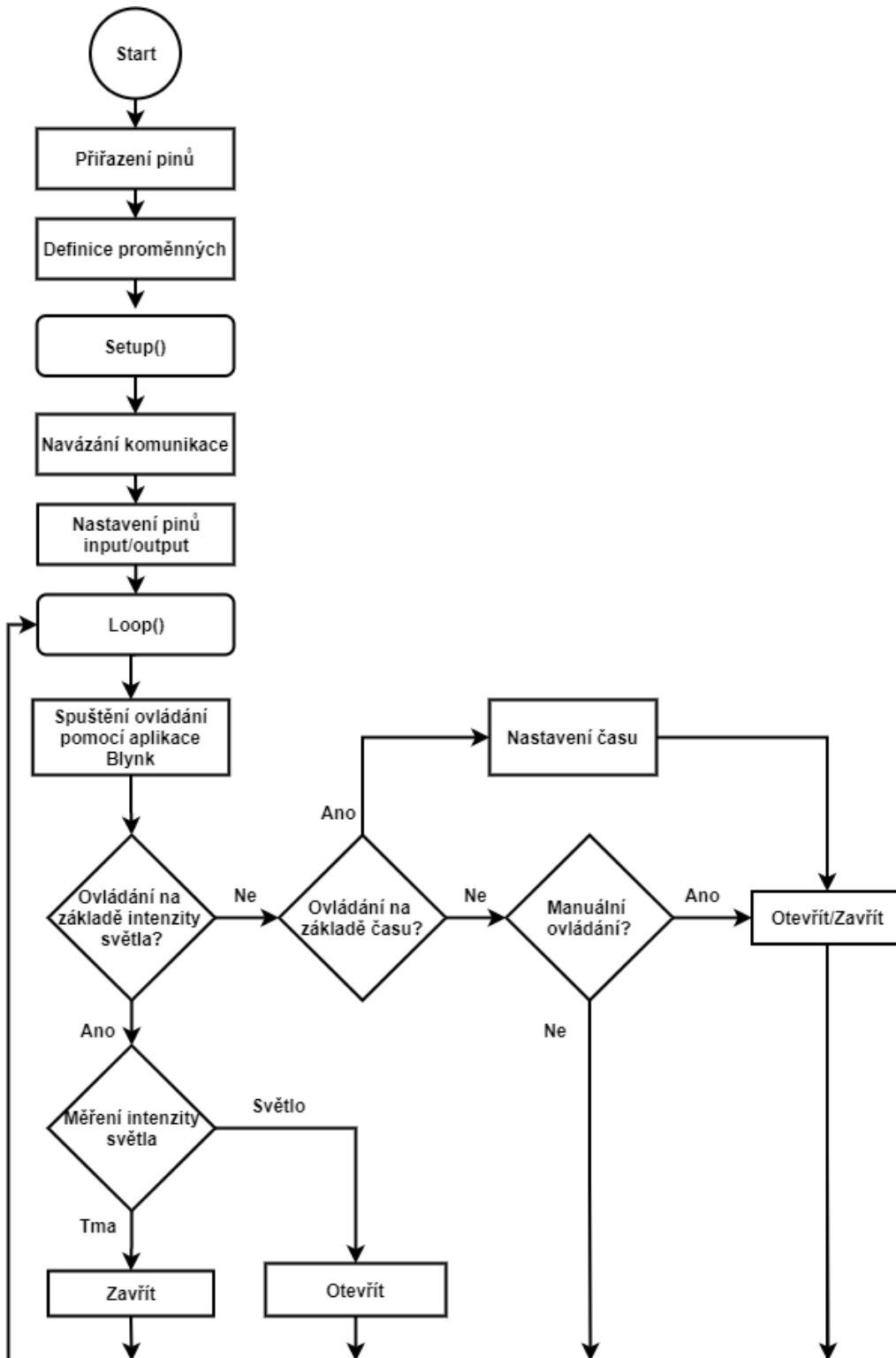
// Zadání přihlašovacích údajů Wi-Fi
// Set password to "" for open networks.
char ssid[] = "YourNetworkName";
char pass[] = "YourPassword";
```

Zdroj: Vlastní tvorba, Arduino IDE, 2021

Na obrázku číslo 27 je ukázka kódu Arduino aplikace. Je zde vidět výše zmiňované přidávání externích knihoven. Dále vložení autorizačního tokenu k aplikaci Blynk, který jsem obdržel na e-mailovou adresu při založení nového projektu. Tomuto tématu se více věnuji na straně 35. Poslední částí je vložení přihlašovacích údajů Wi-Fi sítě, ke které se bude platforma Arduino pomocí Wi-Fi modulu ESP-01 připojovat.

Vývojový diagram aplikace Arduino

Obrázek 28: Vývojový digram aplikace Arduino



Zdroj: Vlastní tvorba, Draw.io: Diagram Software and Flowchart Maker, 2021

Využité knihovny

- **ESP8266WiFi.h** – knihovna zjednodušující práci s Wi-Fi modulem ESP-01.
- **BlynkSimpleESP8266.h** – knihovna umožňující propojení aplikace Blynk s Wi-Fi modulem ESP-01.
- **SoftwareSerial.h** – knihovna umožňující sériovou komunikaci prostřednictvím pinů 0 a 1

5.6. Testování

Po zapojení a sestavení modelu jsem zahájil první fázi testování, při které jsem ověřoval funkčnost bezdrátové komunikace a správné propojení s mobilní aplikací. Při této fázi jsem narazil na problém s Wi-Fi modulem, který nereagoval. Řešení spočívalo ve vzájemném prohození zapojení pinů TX a RX na desce Arduino UNO po naprogramování zařízení.

Ve druhé fázi jsem zkoušel funkčnost motoru a nastavení citlivosti jazýčkových kontaktů. Původně jsem k pohonu používal krokový motor 28BYJ-48 napájený 5V, který se ukázal jako nevyhovující. Rychlost otáčení byla poměrně nízká a točivý moment byl nedostatečný. Proto jsem musel tento motor vyměnit za JGA25-370, který je vhodnější. Zároveň bylo nutné upravit zapojení a zdrojový kód, protože krokový motor měl svou řídicí desku a celý systém ovládání je založen na trochu jiném principu.

Třetí fázi bylo otestování celého modelu s využitím funkcí mobilní aplikace. Veškeré prvky aplikace fungovaly dle návrhu. V této fázi jsem testoval i správnou funkčnost senzoru intenzity světla. Test probíhal 5 dní, kdy jsem při úsvitu a soumraku pozoroval funkčnost systému. Ve všech případech se systém choval dle zvoleného cíle projektu.

6. Závěr

Cílem práce bylo zautomatizování otevírání dveří v chovu domácích zvířat a tím usnadnění každodenní práce, čehož bylo docíleno pomocí jednoduché aplikace vytvořené pomocí softwaru od společnosti Blynk a využitím platformy Arduino, k realizaci vlastního mechanismu otevírání dveří. K plné automatizaci napomáhá fotorezistor vyhodnocující intenzitu venkovního světla. Tím byly naplněny cíle bakalářské práce. Systém je obohacen bezpečnostními prvky, ve formě jazýčkových kontaktů umístěných na horní a spodní hraně otvoru dveří, které kontrolují správnou polohu.

Realizovaný model splňuje požadavky, které jsem si stanovil na začátku práce. Celý systém jsem řádně otestoval, přičemž jsem objevil problém s výkonem použitého krokového motoru, který jsem následně vyměnil za stejnosměrný motor. Po 5denním testování ve venkovním prostředí, které proběhlo dle očekávání, mám osobně dobrý pocit.

Jak jsem uváděl na začátku, v dnešním světě mobilních telefonů, dostupného internetu a chytrých domácností, by se tento systém měl stát samozřejmostí. Osobně si myslím, že poptávka po řešení automatického ovládání, případně ovládání pomocí mobilní aplikace, se bude zvyšovat. Celý tento systém vyšel přibližně na 1 000 Kč, což je v porovnání s konkurencí opravdu znatelný rozdíl. Cena se odvíjí od zvoleného obchodníka, pokud vezmeme v úvahu jenom český trh, ceny jednotlivých součástí se mohou vyšplhat až na trojnásobek za naprosto identickou součástku.

V průběhu testování jsem objevil několik možných oblastí pro zlepšení.

První oblastí je zlepšení způsobu napájení, jako nabízejí konkurenční výrobky, které rozebírám v teoretické části. Varianta napájení pomocí baterie by zjednodušila montáž a zvýšila možnost nasazení systému ve větším počtu kurníků, protože ne každý má jednoduše dostupné síťové napájení. Nejlepší variantou by byla kombinace baterie a solárního panelu, který by během dne dokázal pokrýt spotřebu zařízení a zároveň baterii dobíjet, aby byla připravena na nadcházející noc. Toto řešení by vyžadovalo důkladné propočty a zamyšlení nad celkovou spotřebou dosavadního řešení.

Další oblastí pro zlepšení by bylo například přidání počítání slepic uvnitř a vně kurníku. Toto úloha by se dala řešit světelnou závorou a vhodně umístěným senzorem vzdálenosti, který by určoval směr pohybu.

Zdokonalení by bylo také možné v podobě IP kamery s nočním viděním, která by obraz z vnitřku kurníku přenášela do mobilní aplikace.

I. Summary

The main goal of this bachelor's thesis was to automate the opening of doors in pet breeding and thus facilitate daily work, which was achieved using a simple application created using software from Blynk and using the Arduino platform to implement its own door opening mechanism. A photoresistor evaluating the intensity of outdoor light helps for full automation. This fulfilled the goals of the bachelor's thesis. The system is enriched with security elements, in the form of reel switch located on the upper and lower edge of the door opening, which check the correct position.

The implemented model meets the requirements that I set at the beginning of the work. I tested the whole system properly, discovering a problem with the performance of the stepper motor used, which I then replaced with a DC motor. After 5 days of testing in the outdoor environment, which took place as expected, I personally feel good.

As I said at the beginning in today's world of mobile phones, affordable internet and smart homes, this system should become a matter of course. Personally, I think that the demand for automatic control solutions, or control using a mobile application, will increase. The whole system cost about CZK 1,000, which is a noticeable difference compared to the competition. The price depends on the chosen dealer, if we consider only the Czech market, the prices of individual components can climb up to three times for a completely identical component.

During testing, they identified several possible areas for improvement.

The first area is to improve the way power is offered, as offered by competing products, which I discuss in the theoretical part. The battery-powered variant would simplify installation and increase the possibility of deploying the system in a larger number of henhouses, as not everyone has easily available mains power. The best option would be a combination of a battery and a solar panel, which would be able to cover the consumption of the device during the day and at the same time recharge the battery to be ready for the coming night. This solution would require thorough calculations and reflection on the overall consumption of the current solution.

Another area for improvement would be, for example, the addition of counting hens inside and outside the chicken coop. This solution could be solved by a light barrier and a suitably placed distance sensor, which would determine the direction of movement.

Improvements would also be possible in the form of an IP camera with night vision, which would transmit the image from inside the chicken coop to the mobile application.

II. Seznam použitých zdrojů

- *What is Arduino?. (2018). Arduino.* Retrieved January 23, 2021, from <https://www.arduino.cc/en/Guide/Introduction>
- *Arduino Products. (2021). Arduino.* Retrieved February 5, 2021, from <https://www.arduino.cc/en/Products/Compare>
- Voda, Z. (2017). *Průvodce světem Arduina (Vydání druhé). Martin Stríž.*
- Selecký, M. (2016). *Arduino: uživatelská příručka (přeložil Martin HERODEK).* Computer Press.
- McRoberts, M. (2010). *Beginning Arduino (1st ed.). Apress.*
- Aqeel, A. (2018). *Introduction to Arduino Uno. The Engineering Projects.* Retrieved February 26, 2021, from <https://www.theengineeringprojects.com/>
- Navrátil, V. (2015). *Moderní způsoby řízení stejnosměrných motorů [Bakalářská práce, České vysoké učení technické v Praze, Fakulta elektrotechnická].* https://dspace.cvut.cz/bitstream/handle/10467/62079/F3-BP-2015-Navratil-Vojtech-navratil_vojtech.pdf
- Novák, M., & Pech, J. (2020). *Robotika pro střední školy: Programujeme Arduino. Jihočeská univerzita v Českých Budějovicích, Pedagogická fakulta.* https://imysleni.cz/images/vzdelavaci_materialy/ucebnice-arduino-full-text-reduce-for-web.pdf
- Paštyka, J. (2017). *DC motory řízené pomocí H můstku.* https://embedded.fel.cvut.cz/sites/default/files/kurzy/lpe/h-bridge/H_Bridge.pdf
- Frohn, M. (2006). *Elektronika: polovodičové součástky a základní zapojení. BEN - technická literatura.*
- Fezari, M., & Al Dahoud, A. (2018). *Integrated Development Environment “IDE” For Arduino.* https://www.researchgate.net/profile/Mohamed-Fezari-2/publication/328615543_Integrated_Development_Environment_IDE_For_Arduino/links/5bd8c6d24585150b2b9206df/Integrated-Development-Environment-IDE-For-Arduino.pdf
- Fritzing. (2021). Retrieved April 20, 2021, from <https://fritzing.org/>
- *Int. Journal of Engineering Research and Applications. (2014) (Vol. 4).*

- *Blynk IoT platform: for business and developers. (2020).* Retrieved March 20, 2021, from <https://blynk.io/>
- *Zemědělské potřeby M+S. (2021). Automatické otvírání a zavírání kurníku, otvírání dvířek pro slepice.* Retrieved March 20, 2021, from <https://www.eshop-zemedelske-potreby.cz/>
- *Automatické zavírání kurníku do 2,5 kg. (2021). Zemědělské a chovatelské potřeby eshop Kamír.* Retrieved March 20, 2021, from <https://www.kamir.cz/automat-pro-otevirani-dveri-u-kurniku-do-25-kg>
- *Automatická dvířka Chickdoor. (2021). Chytrý kurník.* Retrieved March 20, 2021, from https://eshop.chytrykurnik.cz/automaticka_dvirka/automaticka-dvirka-chickdoor/
- *Motor JGA25-370 12V s převodovkou. Laskarduino.* Retrieved March 8, 2021, from <https://www.laskarduino.cz/motor-jga25-370-12v-s-prevodovkou/>
- *Dual full-bridge driver L298 (2000). SparkFun Electronics.* Retrieved March 10, 2021, from https://www.sparkfun.com/datasheets/Robotics/L298_H_Bridge.pdf
- *Draw.io: Diagram Software and Flowchart Maker. (2021).* Retrieved April 10, 2021, from <https://app.diagrams.net/>
- *SketchUp. (2021).* Retrieved April 10, 2021, from <https://app.sketchup.com/app?hl=en>

III. Seznam obrázků

Obrázek 1: Arduino UNO.....	7
Obrázek 2: Princip stejnosměrného motoru.....	8
Obrázek 3 Příklad zapojení H-můstku	9
Obrázek 4 Závislost odporu fotorezistoru na intenzitě osvětlení	10
Obrázek 5: Prostředí Arduino IDE	12
Obrázek 6: Prostředí Fritzing.....	13
Obrázek 7: Systém automatického otevírání dveří ChickenGuard Extreme.....	16
Obrázek 8: Systém automatického otevírání dveří Kerbl.....	17
Obrázek 9: Systém automatického otevírání dveří Chickdoor.....	18
Obrázek 10: Arduino UNO	20
Obrázek 11: Stejnosměrný motor JGA25-370.....	21
Obrázek 12: H-můstek L298N	23
Obrázek 13: Jazyčkový kontakt s komparátorem LM393	24
Obrázek 14: Wi-Fi modul ESP-01	25
Obrázek 15: 3D návrh modelu	27
Obrázek 16: Vizualizace elektrického zapojení	30
Obrázek 17: Schéma elektrického zapojení	31
Obrázek 18: Schéma propojení platformy s aplikací Blynk.....	33
Obrázek 19: Blynk - založení projektu.....	34
Obrázek 20: Blynk - pracovní plocha.....	35
Obrázek 21: Blynk - Controllers widgety.....	36
Obrázek 22: Blynk – Displays widgety.....	37
Obrázek 23: Blynk – Interface widgety	37
Obrázek 24: Digram případů užití	38
Obrázek 25: Blynk - Návrh aplikace	39
Obrázek 26: Vývojový diagram mobilní aplikace	40
Obrázek 27: Ukázka kódu Arduino aplikace.....	41
Obrázek 28: Vývojový digram aplikace Arduino	42

IV. Seznam tabulek

Tabulka 1: Seznam použitých zkratk.....	3
Tabulka 2: Parametry motoru JGA25-370	22

V. Seznam příloh

Příloha A1: Zdrojový kód aplikace Arduino část 1	53
Příloha A2: Zdrojový kód aplikace Arduino část 2	54
Příloha A3: Zdrojový kód aplikace Arduino část 3	55
Příloha A4: Zdrojový kód aplikace Arduino část 4	55
Příloha A5: Zdrojový kód aplikace Arduino část 5	56
Příloha A6: Zdrojový kód aplikace Arduino část 6	57
Příloha A8: Zdrojový kód aplikace Arduino část 8	58
Příloha A7: Zdrojový kód aplikace Arduino část 7	58

VI. Přílohy

Příloha A1: Zdrojový kód aplikace Arduino část 1

```
// Knihovny

#define BLYNK_PRINT Serial

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SoftwareSerial.h>

//Vlozeni autorizacniho tokenu k aplikaci Blynk
char auth[] = "90ywXkshTqcSshFzbpYrluCzv6kzDQUY ";

// Zadani prihlasovacich udaju Wi-Fi
// Set password to "" for open networks.
char ssid[] = "YourNetworkName";
char pass[] = "YourPassword";

// Prirazeni pinu

SoftwareSerial EspSerial(0, 1); // Prirazeni pinu RX, TX pro modul ESP-01
// Your ESP8266 baud rate:
#define ESP8266_BAUD 115200
ESP8266 wifi(&Serial);

const int photocellPin = A0;           // Pripojeni fotorezistoru na pin A0
const int enableCoopDoorMotorB = 9;    // Zapnuti motor b - pin 9
const int directionCloseCoopDoorMotorB = 10; // Smer zavirani motor b - pin 10
const int directionOpenCoopDoorMotorB = 11; // Smer otevirani motor b - pin 11
const int bottomSwitchPin = 6;         // Pripojeni spodniho kontaktu na pin 26
const int topSwitchPin = 5;            // Pripojeni horniho kontaktu na pin 27

// Promenne

// Fotorezistor
int photocellReading;                  // Analogove cteni fotorezistoru
int photocellReadingLevel;             // Urovne fotorezistoru (tma, sero, svetlo)
```

Zdroj: Vlastní tvorba, Arduino IDE, 2021

Příloha A2: Zdrojový kód aplikace Arduino část 2

```
// Jazyckove kontakty

// Horni kontakt

int topSwitchPinVal;           // Stav
int topSwitchPinVal2;         // Zpozdeni
int topSwitchState;           // Uchovani stavu

// Dolni kontakt

int bottomSwitchPinVal;       // Stav
int bottomSwitchPinVal2;     // Zpozdeni
int bottomSwitchState;       // Uchovani stavu

// Zpozdeni cteni fotorezistoru
unsigned long lastPhotocellReadingTime = 0;
unsigned long photocellReadingDelay = 600000; // 10 minut

// Zpozdeni kontaktu
unsigned long lastDebounceTime = 0;
unsigned long debounceDelay = 100;
```

Zdroj: Vlastní tvorba, Arduino IDE, 2021

Příloha A3: Zdrojový kód aplikace Arduino část 3

```
// ***** Nastaveni *****  
  
void setup(void) {  
  
  Serial.begin(9600); // initialize serial port hardware  
  Blynk.begin(auth);  
  
  // Dvere kurniku  
  
  // Motor  
  pinMode (enableCoopDoorMotorB, OUTPUT); // Nastavení pinu aktivace motoru jako OUTPUT  
  pinMode (directionCloseCoopDoorMotorB, OUTPUT); // Nastavení pinu motoru smer zavirani jako OUTPUT  
  pinMode (directionOpenCoopDoorMotorB, OUTPUT); // Nastavení pinu motoru smer otevirani jako OUTPUT  
  
  // Jazyckove kontakty  
  
  // Horní kontakt  
  pinMode(topSwitchPin, INPUT); // Nastavení pinu horniho kontaktu jako INPUT  
  digitalWrite(topSwitchPin, HIGH); // Aktivace rezistoru horniho kontaktu  
  // Dolní kontakt  
  pinMode(bottomSwitchPin, INPUT); // Nastavení pinu dolniho kontaktu jako INPUT  
  digitalWrite(bottomSwitchPin, HIGH); // Aktivace rezistoru spodniho kontaktu  
  
}
```

Zdroj: Vlastní tvorba, Arduino IDE, 2021

Příloha A4: Zdrojový kód aplikace Arduino část 4

```
// ***** Funkce *****  
  
// Zaznamenavani intenzity svetla fotorezistorem  
  
void doReadPhotoCell() { // Funkce bude volana opakovane v zavislosti na definovanem zpozdeni (kazdych 10 min)  
  
  photocellReading = analogRead(photocellPin);  
  
  if ((unsigned long)(millis() - lastPhotocellReadingTime) >= photocellReadingDelay) {  
    lastPhotocellReadingTime = millis();  
  
    // Nastaveni stavu fotorezistoru  
    if (photocellReading >= 0 && photocellReading <= 3) { photocellReadingLevel = '1';  
      Blynk.virtualWrite(V0, "Tma"); }  
    else if (photocellReading >= 4 && photocellReading <= 120) { photocellReadingLevel = '2';  
      Blynk.virtualWrite(V0, "Sero"); }  
    else if (photocellReading >= 125 ) {photocellReadingLevel = '3';  
      Blynk.virtualWrite(V0, "Svetlo");}  
  }  
}
```

Zdroj: Vlastní tvorba, Arduino IDE, 2021

Příloha A5: Zdrojový kód aplikace Arduino část 5

```
//Zpozdeni spodniho kontaktu

void debounceBottomReedSwitch() {

  //Zpozdeni dolniho kontaktu
  bottomSwitchPinVal = digitalRead(bottomSwitchPin);      // Cteni vstupni hodnoty a uchovani

  if ((unsigned long)(millis() - lastDebounceTime) > debounceDelay) {    // Zpozdeni 10ms

    bottomSwitchPinVal2 = digitalRead(bottomSwitchPin);    // Kontrolni cteni vstupni hodnoty

    if (bottomSwitchPinVal == bottomSwitchPinVal2) {      // Ujisteni ze obe hodnoty jsou stejne
      if (bottomSwitchPinVal != bottomSwitchState) {      // Zmena stavu
        bottomSwitchState = bottomSwitchPinVal;
      }
    }
  }
}

// Zpozdeni horniho kontaktu
void debounceTopReedSwitch() {

  topSwitchPinVal = digitalRead(topSwitchPin);            // Cteni vstupni hodnoty a uchovani

  if ((unsigned long)(millis() - lastDebounceTime) > debounceDelay) {    // Zpozdeni 10ms

    topSwitchPinVal2 = digitalRead(topSwitchPin);          // Kontrolni cteni vstupni hodnoty

    if (topSwitchPinVal == topSwitchPinVal2) {            // Ujisteni ze obe hodnoty jsou stejne
      if (topSwitchPinVal != topSwitchState) {            // Zmena stavu
        topSwitchState = topSwitchPinVal;
      }
    }
  }
}
```

Zdroj: Vlastní tvorba, Arduino IDE, 2021

Příloha A6: Zdrojový kód aplikace Arduino část 6

```
// Zastavení motoru
void stopCoopDoorMotorB() {
  digitalWrite (directionCloseCoopDoorMotorB, LOW); // Vypnutí zavírání
  digitalWrite (directionOpenCoopDoorMotorB, LOW); // Zapnutí otevírání
  analogWrite (enableCoopDoorMotorB, 0); // Zapnutí motoru, rychlost 0
}

// Zavření dveří
void closeCoopDoorMotorB() {
  digitalWrite (directionCloseCoopDoorMotorB, HIGH); // Zapnutí zavírání
  digitalWrite (directionOpenCoopDoorMotorB, LOW); // Vypnutí otevírání
  analogWrite (enableCoopDoorMotorB, 255); // Zapnutí motoru, maximální rychlost
  if (bottomSwitchPinVal == 0) { // Dokud se nesepe spodní kontakt
    stopCoopDoorMotorB();
  }
}

// Otevření dveří
void openCoopDoorMotorB() {
  digitalWrite (directionCloseCoopDoorMotorB, LOW); // Vypnutí zavírání
  digitalWrite (directionOpenCoopDoorMotorB, HIGH); // Zapnutí otevírání
  analogWrite (enableCoopDoorMotorB, 255); // Zapnutí motoru, maximální rychlost
  if (topSwitchPinVal == 0) { // Dokud se nesepe horní kontakt
    stopCoopDoorMotorB();
  }
}
```

Zdroj: Vlastní tvorba, Arduino IDE, 2021

Příloha A7: Zdrojový kód aplikace Arduino část 7

```
// Automaticke ovladani
void doCoopDoor() {
  if (photoCellReadingLevel == '1') { // Pokud je tma
    if (photoCellReadingLevel != '2') { // Pokud není šero
      if (photoCellReadingLevel != '3') { // Pokud není světlo
        debounceTopReedSwitch(); // read and debounce the switches
        debounceBottomReedSwitch();
        closeCoopDoorMotorB(); // Zavřít dveře
      }
    }
  }
  if (photoCellReadingLevel == '3') { // Pokud je světlo
    if (photoCellReadingLevel != '2') { // Pokud není šero
      if (photoCellReadingLevel != '1') { // Pokud není tma
        debounceTopReedSwitch(); // read and debounce the switches
        debounceBottomReedSwitch();
        openCoopDoorMotorB(); // Otevřít dveře
      }
    }
  }
}

BLYNK_WRITE(V2) // Čtení virtuálního pinu V2
{
  if (param.asInt() == 1) {
    openCoopDoorMotorB();
  }
  else {closeCoopDoorMotorB(); }

  BLYNK_WRITE(V1) // Čtení virtuálního pinu V1
  {
    pinValue = param.asInt();
    if (param.asInt() == HIGH) {
      openCoopDoorMotorB();
    }
    else closeCoopDoorMotorB();
  }
}
```

Zdroj: Vlastní tvorba, Arduino IDE, 2021

Příloha A8: Zdrojový kód aplikace Arduino část 8

```
// ***** the loop *****

void loop() {

  Blynk.run();
  doReadPhotoCell();
  doCoopDoor();

}
```

Zdroj: Vlastní tvorba, Arduino IDE, 2021