

Diagnostická sestava pro měření spotřeby bezpečnostních kamer

Bc. Richard Konečný

Diplomová práce
2022



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav elektroniky a měření

Akademický rok: 2021/2022

ZADÁNÍ DIPLOMOVÉ PRÁCE

(projektu, uměleckého díla, uměleckého výkonu)

Jméno a příjmení: Bc. Richard Konečný
Osobní číslo: A20157
Studijní program: N1032A020003 Bezpečnostní technologie, systémy a management
Specializace: Bezpečnostní technologie
Forma studia: Prezenční
Téma práce: Diagnostická sestava pro měření spotřeby bezpečnostních kamer
Téma práce anglicky: Diagnostic Set for Power Measurement of Security Cameras

Zásady pro vypracování

1. Vypracujte literární rešerši zaměřenou na současné trendy kamerových systémů s ohledem na měření spotřeby.
2. Specifikujte vývojovou platformu Arduino pro ovládání bezpečnostní kamery a měření spotřeby.
3. Navrhněte a realizujte prototyp diagnostické sestavy.
4. Vytvořte programové vybavení pro diagnostickou sestavu.
5. Ověřte funkčnost navržené sestavy.

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam doporučené literatury:

1. PETZOLD, Charles. Mistrovství ve Windows Presentation Foundation: [aplikace = kód + markup]. Brno: Computer Press, 2008. Mistrovství. ISBN 9788025121412.
2. KAČMÁŘ, Dalibor. Programujeme .NET aplikace ve Visual Studiu .NET. Praha: Computer Press, 2001. Všechny cesty k informacím. ISBN 80-7226-569-5.
3. LUKÁŠ, Luděk. Bezpečnostní technologie, systémy a management I. 1. vyd. Zlín: VeRBuM, 2011, 316 s. ISBN 978-80-87500-05-7.
4. VODA, Zbyšek. Průvodce světem Arduina. Vydání druhé. Bučovice: Martin Stříž, 2017. ISBN 978-80-87106-93-8.
5. MANN, Burkhard. C pro mikrokontroléry: ANSI-C, kompilátory C, spojovací programy –linkery, práce s ATMEL AVR a MSC-51, příklady programování v jazyce C, nástroje pro programování, tipy a triky ... Praha: BEN – technická literatura, 2003. μ C & praxe. ISBN 80-7300-077-6.
6. LOVEČEK, Tomáš a Peter NAGY. Bezpečnostné systémy: kamerové bezpečnostné systémy. Žilina: Žilinská univerzita, 2008, 283 s. Vysokoškolské učebnice. ISBN 9788080708931.

Vedoucí diplomové práce: **Ing. Stanislav Kovář, PhD.**
Ústav bezpečnostního inženýrství

Datum zadání diplomové práce: **3. prosince 2021**

Termín odevzdání diplomové práce: **23. května 2022**

doc. Mgr. Milan Adámek, Ph.D. v.r.
děkan



Ing. Milan Navrátil, Ph.D. v.r.
ředitel ústavu

Ve Zlíně dne 7. února 2022

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně;
- byl/a jsem seznámen/a s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití jen připouští-li tak licenční smlouva uzavřená mezi mnou a Univerzitou Tomáše Bati ve Zlíně s tím, že vyrovnání případného přiměřeného příspěvku na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše) bude rovněž předmětem této licenční smlouvy;
- beru na vědomí, že pokud bylo k vypracování diplomové práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně, dne 1. května 2022

Bc. Richard Konečný, v. r.
podpis studenta

ABSTRAKT

Cílem práce je navrhnout a realizovat diagnostickou (monitorovací) sestavu pro IP kamery na bázi platformy Arduino. V teoretické části práce se student seznámí se současným stavem v oblasti, principy kamerových systémů a platformou Arduino. V úvodu praktické části se bude věnovat měření spotřeby IP kamer, pomocí Hallova senzoru, při variabilních provozních stavech. Následovat bude návrh a realizace diagnostické sestavy, včetně aplikace pro monitorování obrazu. Součástí práce bude i metoda aktivace/deaktivace IP kamery pro účely úspory energie. V závěru práce bude otestována funkčnost sestavy, včetně navržených metod aktivace a deaktivace.

Klíčová slova: diagnostická sestava, IP kamera, platforma Arduino, RFID čtečka, Hallův senzor

ABSTRACT

The thesis aims to design and implement a diagnostic (monitoring) set for IP cameras based on the Arduino platform. In the theoretical part of the work, the student gets acquainted with the current state in the field, the principles of camera systems, and the Arduino platform. The introduction to the practical part will focus on measuring the IP cameras consumption using a Hall sensor in variable operating conditions. This part will be followed by designing and implementing a diagnostic kit, including an image monitoring application. The work will also include activating/deactivating the IP camera for energy-saving purposes. The conclusion of the work will consist of assembly functionality testing, including the proposed activation and deactivation methods.

Keywords: diagnostic set, IP camera, Arduino platform, RFID reader, Hall sensor

Rád bych poděkoval vedoucímu diplomové práce panu Ing. Stanislavu Kovářovi, Ph.D. za vedení, užitečné rady a připomínky během vypracování diplomové práce.

Prohlašuji, že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

OBSAH

ÚVOD	8
I TEORETICKÁ ČÁST	9
1 KAMEROVÉ SYSTÉMY	10
1.1 ANALOGOVÉ KAMEROVÉ SYSTÉMY	10
1.2 DIGITÁLNÍ KAMEROVÉ SYSTÉMY	10
1.2.1 Hlavní podstata funkce IP kamery	11
1.2.2 Konstrukce IP kamery	12
1.2.3 Komunikace IP kamery	17
1.3 HYBRIDNÍ KAMEROVÉ SYSTÉMY	19
1.4 SPOTŘEBA KAMEROVÝCH SYSTÉMŮ	20
1.5 SOUČASNÉ TRENDY V OBLASTI KAMEROVÝCH SYSTÉMŮ.....	21
1.6 INTEGRACE KAMEROVÝCH SYSTÉMŮ.....	22
2 SPECIFIKACE PLATFORMY ARDUINO	24
2.1 ARDUINO SHIELDY	25
2.2 PROGRAMOVÁNÍ ARDUINA.....	25
2.2.1 Proměnné.....	26
2.2.2 Datové typy	27
2.3 PROPOJENÍ ARDUINA S POČÍTAČEM	28
2.4 SÉRIOVÁ KOMUNIKACE	29
II PRAKTICKÁ ČÁST	30
3 NÁVRH A REALIZACE PROTOTYPU	31
3.1 POUŽITÉ KOMPONENTY	32
3.1.1 Arduino Mega	32
3.1.2 RFID čtečka karet RC522	33
3.1.3 LCD display	34
3.1.4 IR senzor	35
3.1.5 Modul relé 5 V	36
3.1.6 Ethernet Shield W5100	37
3.1.7 RTC hodiny reálného času	38
3.1.8 Hallův senzor proudu	39
3.1.9 Paměťová karta	43
3.1.10 Ostatní komponenty	43
3.2 NAPÁJENÍ PROTOTYPU	45
3.2.1 Napájení pomocí bateriového boxu	45
3.2.2 Napájení adaptérem.....	46
3.3 SCHÉMA PROTOTYPU	46
3.4 VYTVOŘENÁ DIAGNOSTICKÁ SESTAVA	47
3.5 POUŽITÁ IP KAMERA	53
3.6 PoE PŘEPÍNAČ	53
4 PROGRAMOVÉ VYBAVENÍ	56

4.1	VÝVOJOVÉ DIAGRAMY	56
4.2	POPIS ZDROJOVÉHO KÓDU V ARDUINO IDE	58
4.2.1	Importování knihoven	59
4.2.2	Nastavení pro vybrané komponenty.....	60
4.2.3	Nastavení ID karty, která bude sloužit pro vypnutí IP kamery	61
4.2.4	Kontrola microSD karty	61
4.2.5	Načtení hodnot z Hallova senzoru proudu	61
4.2.6	Detekování pohybu pomocí IR senzoru	62
4.2.7	Ukládání data a času na microSD kartu	62
4.2.8	Ukládání naměřených dat pomocí Hallova senzoru na microSD kartu	63
4.2.9	Kontrola přiložené RFID karty	63
4.2.10	Přiložení správné RFID karty.....	65
4.2.11	Přiložení nesprávné RFID karty	66
4.2.12	IR senzor nedetekuje pohyb	66
4.3	POPIS ZDROJOVÉHO KÓDU WPF APLIKACE V C#.....	67
4.3.1	Návrh WPF aplikace	67
4.3.2	WPF aplikace pro sledování videa.....	68
4.3.3	Vytváření GUI.....	68
4.3.4	Importování knihoven	69
4.3.5	Vytvoření proměnných.....	69
4.3.6	Přiřazení proměnných k předdefinovaným funkcím.....	70
4.3.7	Tlačítko pro připojení IP kamery	70
4.3.8	Tlačítko pro zapnutí Motion Detection	71
4.3.9	Vytvoření události pro Motion Detection	71
4.3.10	Uložení snímku pomocí tlačítka Snímek	72
5	OVĚŘENÍ FUNKČNOSTI.....	74
5.1	KONTROLA MICROSD KARTY	74
5.2	NEDETEKOVÁN POHYB	74
5.3	DETEKOVÁN POHYB, SPRÁVNÁ RFID KARTA	75
5.4	DETEKOVÁN POHYB, NESPRÁVNÁ RFID KARTA.....	76
5.5	SLEDOVÁNÍ VIDEA PŘES WPF APLIKACI.....	77
5.6	ZHODNOCENÍ MĚŘENÍ.....	79
5.7	ANALYZOVÁNÍ NAMĚŘENÝCH HODNOT	81
5.7.1	Měření s IP kamerou Axis.....	82
5.7.2	Měření s IP kamerou Vivotek	85
	ZÁVĚR	87
	SEZNAM POUŽITÉ LITERATURY.....	88
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	92
	SEZNAM OBRÁZKŮ	95
	SEZNAM TABULEK.....	99
	SEZNAM PŘÍLOH.....	100

ÚVOD

Kamerové systémy lze řadit mezi prvky vnitřní i vnější prostorové ochrany. Kamerové systémy zaznamenaly významný progres v posledních letech způsobený zejména rozšířením internetu a zvýšením kvality přenosových médií. IP kamery v dnešní době lze nalézt jak v rodinných domech, veřejných institucích, na nádražích, letištích, ve městech, nákupních centrech a na spoustu dalších místech, kterými denně procházíme. Některé kamery mohou být umístěny skrytě tak, aby nebyly zpozorovatelné, oproti tomu jiné kamery mají i preventivní charakter a instalují se veřejně. Na trhu se lze setkat s maketami IP kamer, které nejsou používány k záznamu, ale pouze mají odradit pachatele. Nicméně takové řešení není ideální, ačkoliv maketa představuje levné řešení a působí varovně, neposkytuje žádnou ochranu.

V posledních letech dochází k vylepšování IP kamer o další funkce, jedná se především o ovládání a sledování záznamu přes mobilní telefon prostřednictvím aplikace. Moderní IP kamery obsahují např. LED osvětlení, které je aktivováno při detekci narušitele ve střeženém prostoru, nebo třeba ukládání záznamu na microSD kartu. Dále IP kamera může být vybavena reproduktorem pro případnou komunikaci např. s kurýrem. Mezi časté aplikace kamerových systémů lze řadit: rozpoznávání tváří, rozpoznávání RZ automobilů na dálnici, čtení QR kódů a nespočet dalších aplikací.

Kamerové systémy lze využívat samostatně, nebo v kombinaci s dalšími bezpečnostními aplikacemi. V případě kombinace se mluví o integrovaných poplachových systémech. Často využívanou integrací bývá spojení přístupového systému s kamerovým systémem.

Měření spotřeby představuje jeden z nejdůležitějších parametrů při návrhu kamerových systémů. Kamerové systémy se navrhují pro 24hodinový provoz, což v případě instalace desítek či stovek kamer nemusí být, z hlediska spotřeby, pro investora zanedbatelný faktor. Proto mnoho odběratelů hledá kompromis mezi kvalitou obrazu, poskytovanými funkcemi a spotřebou energie. Proto vznikl požadavek propojení IP kamer např. s detektory pohybu, které aktivují IP kamery pouze v případě, když se ve střeženém objektu nachází osoba.

Technologický vývoj jde každým dnem stále víc kupředu, což zapříčiní širší nabídku na trhu se zbožím pro zákazníka. Širší sortiment a vyšší kompatibilita jednotlivých komponent nabízí možnost vytvoření vlastního modelu nebo celého funkčního systému, který bude reflektovat skutečný stav reálných komponent, se kterými se lze běžně setkat.

I. TEORETICKÁ ČÁST

1 KAMEROVÉ SYSTÉMY

Základní rozdělení kamerových systémů:

- **analogové** – v současné době jsou méně rozšířeným druhem kamerových systémů, pracují s analogovým signálem, který je spojitý v čase,
- **digitální** – poskytují vzdálený přístup pro správu kamery, v současnosti jsou považovány za nejrozšířenější druh kamerových systémů, poskytují široké možnosti analýzy zpracovaného obrazu,
- **hybridní** – spojení analogových a digitálních kamerových systémů. [1], [2]

Kamerové systémy upravuje norma ČSN EN 62676-1-1 (334592): Dohledové videosystémy pro použití v bezpečnostních aplikacích. Tato norma vešla v platnost 1.9.2014. Byla vytvořena jako náhrada normy ČSN EN 50132-1, která platila od prosince 2010 do konce roku 2016. [3]

1.1 Analogové kamerové systémy

Analogové kamery jsou propojeny pomocí koaxiálního kabelu. Pro ukládání kamerového záznamu je využíván Videocassette recorder (dále jen VRC). Maximální délka nekomprimovaného záznamu se uvádí přibližně 8 hodin. V případě využití více analogových kamer se používá multiplexor. Jedná se o typ kamerových systémů, který se v současné době používá jen zřídka, tento systém byl nahrazen využitím nových technologií. [1], [2]

1.2 Digitální kamerové systémy

Síťová kamera neboli IP kamera je zařízení, které se skládá ze systému a počítače, výsledkem je nezávisle pracující celek. Příjem a odesílání dat probíhá pomocí počítačové sítě. IP kamera se skládá z: obrazového snímače, procesoru (mikroprocesor pro další porty), paměti, objektivu a potřebného komunikačního rozhraní. Další příslušenství kamery se odvíjí na základě umístění IP kamery. Mezi takové příslušenství lze řadit: kamerové držáky, které zajišťují minimalizaci otřesů a vibrací nebo možnost směrového nastavení IP kamery. Dalším příslušenstvím mohou být kamerové kryty, které lze rozdělit na venkovní a vnitřní. Venkovní kryty se instalují z důvodu ochrany před povětrnostními podmínkami, v některých případech mohou být vybaveny vyhříváním a krytím IP 67. Primárním cílem vnitřních krytů je ochrana proti mechanickému poškození neboli antivandal a maskování IP kamery. Problematika antivandal je detailně rozebrána v publikaci [4]. Antivandal je označení pro stupeň

ochrany vůči mechanickému poškození. Požadavky jsou zahrnuty v převzaté evropské normě pro ČR, ČSN EN 62262. V normě jsou obsaženy klasifikace stupňů ochrany, jež zajišťují ochranu zařízení vůči vnějším nárazům nebo vlivům. Norma také specifikuje značení stupňů ochrany, nebo zkoušky pro verifikaci požadavků. Napájení IP kamery je nejčastěji zajištěno pomocí 12 V nebo 24 V. Samotné napájení lze provést dvěma způsoby, buď pomocí samostatných napájecích kabelů, nebo pomocí datových vodičů Power over Ethernet (dále jen PoE). Napájení pomocí PoE se provádí po stávajících datových vodičích, není potřeba dalších vodičů pro napájení. IP kamera je připojena pouze jedním vodičem, tento vodič zajišťuje jak datový přenos, tak i napájení IP kamery. IP kamerové systémy bývají instalovány s cílem identifikace osob, detekce pohybu ve střežené oblasti, a hlavně k ochraně majetku a osob. [1], [2]

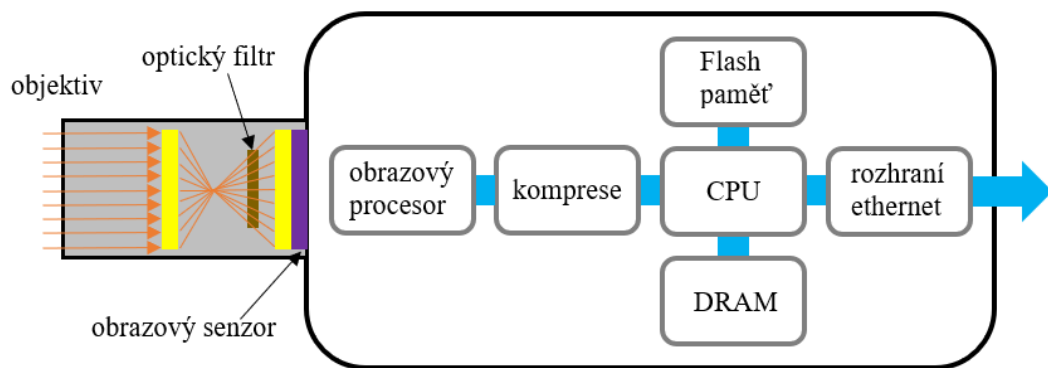
1.2.1 Hlavní podstata funkce IP kamery

Obrazový čip kamery nevnímá pozorovanou scénu jako obraz, jako je tomu u lidského oka, ale jako světlo o různých vlnových délkách, které je následně pomocí Complementary Metal-Oxide-Semiconductor (dále jen CMOS) čipu nebo Charge-coupled device (dále jen CCD) čipu dále převáděno na elektrický náboj. Tento elektrický náboj je dále akumulován v samostatných světločivných buňkách. Následně je výstupem z čipu buď digitální signál, pokud byl použit CMOS čip, nebo analogový signál v případě použití CCD čipu. Ještě před dopadem světla na snímací čip, musí světlo projít objektivem kamery. Funkce, která zajišťuje vykreslování se označuje Modulation Transfer Function (dále jen MTF), a také definuje obraz vzhledem k ohniskové vzdálenosti objektivu. Pro zajištění průchodu pouze určité vlnové délky, se kterou IP kamera pracuje je provedeno s využitím optického filtru. Uvedený filtr je umístěn mezi světlo citlivým čipem a objektivem. [1]

Do obrazového procesoru, někdy označovaného jako Digital Signal Processor (dále jen DSP) proběhne odeslání digitálního signálu. Pokud byl pořízen analogový signál, tak nejdříve nastane jeho převod na digitální signál a ten je poté odeslán. DSP dále pracuje se signálem v digitální podobě a využívá přitom velkého množství různých funkcí pro zvýšení kvality videa. Cílem úprav je zajistit požadovanou kvalitu scenerie, kterou kamera zaznamenává. Za účelem snížení velkého objemu dat je provedena komprese s využitím různých kompresních algoritmů. [1]

Na Obr. 1 jsou blokově znázorněny jednotlivé části IP kamery. Jedná se především o Central Processing Unit (dále jen CPU), Flash paměť a Dynamic Random Access Memory (dále jen

DRAM). Komunikace s externími zařízeními je zajištěna prostřednictvím obou pamětí a CPU. Funkce, které probíhají uvnitř kamery, jedná se např. o nastavování IP kamery nebo spuštění různých operací jsou rovněž zajišťovány procesorem. Jakmile je digitální videosignál podroben kompresi a zpracován, tak nastane jeho odeslání prostřednictvím daného komunikačního rozhraní do cílového zařízení na základě dané struktury systému. [1]



Obr. 1. Blokové schéma funkce IP kamery. [1]

1.2.2 Konstrukce IP kamery

Objektiv – primárním cílem objektivu je vykreslit zmenšený obraz pozorované scény na určité místo na fotocitlivém prvku, vytvořený obraz nesmí obsahovat žádné rušivé a negativní prvky. Objektiv je mechanická část kamery složena ze soustavy čoček uspořádaných v optické ose, lze tedy říct, že jsou opticky vycentrované. Při samotném ostření nebo přibližování, kdy dochází současně ke změně ohniskové vzdálenosti se jednotlivé části objektivu pohybují. [2], [5]

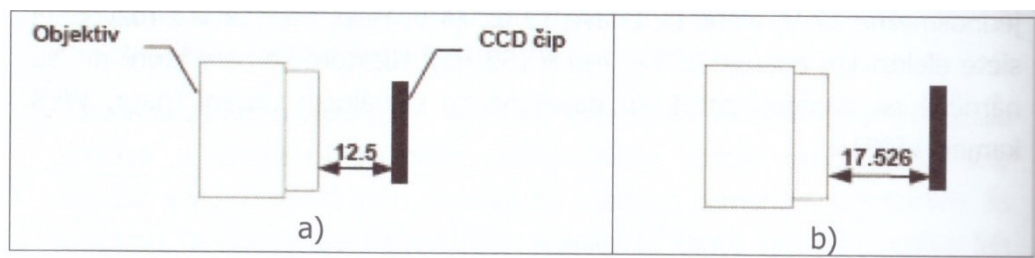
Mezi hlavní parametry, které je nutné řešit při výběru objektivu lze řadit:

- uchycení objektivu,
- ohnisková vzdálenost,
- světelnost,
- clona objektivu,
- možnost nastavení clony a ohniskové vzdálenosti,
- hloubka ostrosti. [5]

Druhy objektivů:

- **normální objektiv** – objektiv pracuje s úhlem záběru přibližně 50°, tato velikost úhlu odpovídá přibližně úhlu záběru lidského oka,

- **širokouhlý objektiv** – v porovnání s normálním objektivem je zde ohnisková vzdálenost kratší, což se projeví širším záběrem snímku. Často se lze setkat s označením „rybí oko“, kde je úhel záběru 180° ,
 - **teleobjektiv** – u tohoto typu objektivu je zorný úhel užší, což dovoluje vyplnit celý snímek např. vzdáleným objektem, obraz objektu je poté zvětšený,
 - **telecentrické objektivy** – výhodou daného typu objektivu je možnost potlačení perspektivy, jelikož objektivem prochází jen paprsky, které jsou rovnoběžné s optickou osou. [2]
- a) **Uchycení objektivu** – na základě uchycení objektivu lze definovat následující rozdělení:
- uchycení typu CS (a),
 - uchycení typu C (b). [5]



Obr. 2. Objektivy s uchycením typu CS a C. [5]

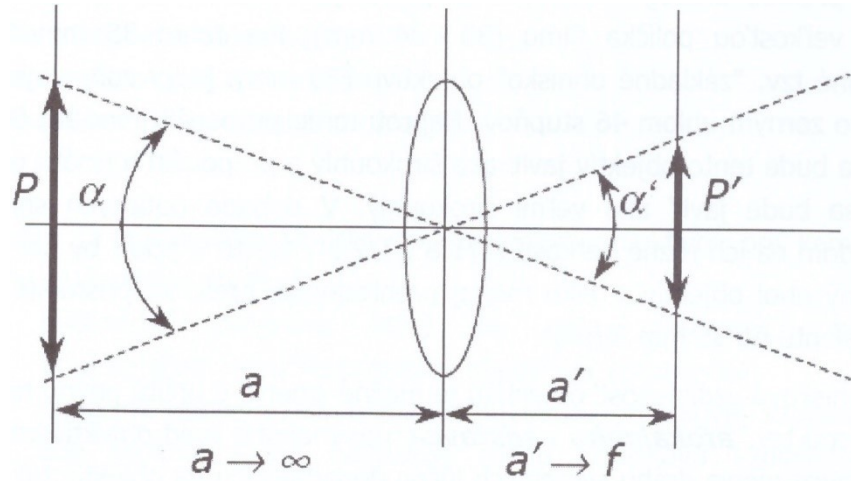
U obou typů uchycení je pro připevnění použit stejný závit. Rozdíl nastává v předepsané vzdálenosti roviny zadní čočky objektivu od optického snímače kamery. Objektivy typu C lze využít jak s kamerami C, tak i s kamerami CS, podmínkou je ale použití C/CS adaptéru. Jedná se o 5 mm redukci ve formě kroužku, která se našroubuje na příslušný závit. [5]

- b) **Ohnisková vzdálenost f (focus)** – jedná se o vzdálenost za objektivem, která je měřena od optického středu objektivu k rovině snímače, nejčastěji je ohnisková vzdálenost měřena v mm. Ohnisková vzdálenost ovlivňuje šířku záběru a zároveň také úhel záběru. Lze tedy říct, že čím kratší je ohnisková vzdálenost, tím větší je úhel záběru objektivu. [2]

Ohniskovou vzdálenost některých objektů lze plynule měnit pomocí zařízení na změnu ohniskové vzdálenosti nazývaného jako transfokátor. Objekty, které disponují proměnlivou ohniskovou vzdáleností lze označit jako zoom. [5]

- f – ohnisková vzdálenost [mm]
- a – vzdálenost snímaného objektu [m]

- a – vzdálenost snímacího prvku [mm]
- P – šířka snímaného objektu [m]
- P' – šířka zaostřovací roviny [mm]
- α – velikost snímaného úhlu kamerou [°][5]



Obr. 3. Zobrazení optickým systémem. [5]

Uvedené rovnice popisují vyjádření ohniskové vzdálenosti. [5]

$$\frac{P'}{P} = \frac{a'}{a} \quad (1.1)$$

$$\frac{1}{a} + \frac{1}{a'} = \frac{1}{f} \quad (1.2)$$

Typy objektivů na základě ohniskové vzdálenosti:

- **pevná ohnisková vzdálenost (Fixed Focal Length)** – ohnisková vzdálenost je pevně nastavena z výroby a není možné ji měnit,
 - **proměnná ohnisková vzdálenost (Varifocal Length)** – ohniskovou vzdálenost lze ručně nastavovat pomocí otáčení určité části objektivu,
 - **elektronicky řízená ohnisková vzdálenost (Motor Zoom)** – nastavení probíhá pomocí servomotoru. [2], [5]
- c) **Světelnost** – jedná se o maximální schopnost přijímat světlo, zároveň platí, že s klesajícím číslem světelnosti je tato funkce vyšší. V praxi to znamená, že při ukládání obrazu s vyšší světelností (menší clonové číslo) je možné při určité scéně zmenšit čas uzávěrky a tím eliminovat možnost rozmazání snímků. [5]
- d) **Clona objektivu** – jedná se o mechanické zařízení, které je využíváno k regulování množství světla procházejícího objektivem. V rozsahu od maximální propustnosti

(otvor o plné velikosti) a zároveň nejmenšího clonového čísla, až po maximální uzavření clony (otvor o nejmenší velikosti). Hodnota nastavené clony, nebo otvoru v objektivu přímo působí na rozlišovací schopnost, tím se myslí kvalita, se kterou je objektiv schopen vykreslit nasnímaný obraz. Při přivírání clony dochází nejdříve k zmenšení rozlišovací schopnosti objektivu. Jakmile je ale clonový otvor příliš malý, tak se rozlišovací schopnost zhoršuje. Podíl ohniskové vzdálenosti a průměru vstupního otvoru je clonové číslo. Na základě ovládní clony je možné použití následujících objektivů. [5]

- **Objektivy s pevnou clonou** – využití nacházejí u kamer, které jsou vybaveny režimem elektronické závěrky Automatic Electronic Shutter, (dále jen AES). Nebo obvodem pro automatické řízení zisku Automatic Gain Control, (dále jen AGC), jedná se o elektronický obvod, který udržuje výstupní videosignál na konstantní úrovni.
 - **Objektivy s manuálním nastavením clony**
 - **Objektiv s automatickou clonou – (Auto Iris – AI)** – tyto objektivy lze rozdělit na dva typy:
 - **standardní ovládaný videosignál (VIDEO AI)** – umístění elektroniky i mechanické části se nachází v těsné blízkosti,
 - **pasivní ovládaný jednosměrným signálem (DC AI)** – využívá galvanický princip clony. Uvnitř objektivu se nachází pouze mechanická část servomotoru. Pro řízení se využívá elektronika, která se nachází v pouzdru kamery. Neupravený signál z optického snímače je upraven pomocí elektronického obvodu v kameře, tento proces je označován jako DSP. [5]
- e) **Hloubka ostrosti (Deep of field – DOF)** – jedná se o určitý rozsah, ve kterém nastává zobrazení předmětů s ještě příznivou ztrátou rozlišení určitých detailů, předměty jsou ještě ostré. Není možné určit přesnou hranici mezi ostrým a neostrým objektem na fotografii. Hyperfokální vzdálenost objektivu je taková vzdálenost objektivu, která je nastavená na zaostřovací skruži, při nastavené vzdálenosti začínají být ostře zobrazené objekty až do nekonečna. Parametry, které ovlivňují hloubku ostrosti jsou: [5]
- clona objektivu,
 - vzdálenost fotografovaného objektu,

- ohnisková vzdálenost,
- velikost plochy optického snímače. [5]

IP kamery prošly v posledních letech značným zlepšením po hardwarové stránce, zároveň nastalo i zlepšení softwarových aplikací pro sledování záznamu z IP kamery. Mezi nejznámější výrobce IP kamer lze řadit např.: Axis, Vivotek, Hikvision. [1]

Následně zde bude provedeno rozdělení na základě současných trendů na trhu společně s popisem každé kategorie.

Fixní IP kamery:

- uvedený typ IP kamery má pevně určený směr natočení, který nelze vzdáleně měnit. Tento druh kamery lze doplnit různými typy objektivů, může se jednat o teleobjektiv, klasický nebo širokoúhlý. Fixní IP kamery bývají používány v prostorech, kde je vyžadováno snímání pouze určité scény a zároveň se požaduje fyzická viditelnost kamery. Na základě umístění se nabízí případné doplnění kamery o ochranný kryt pro vnitřní nebo venkovní prostředí. Mezi často používaný typ lze řadit fixní dome kamery, tyto kamery jsou už v prvotním provedení doplněny o dome kryt. Dome kryt má specifický tvar, podle kterého lze tuto kameru snadno identifikovat. Předností dome kamery je specifická diskrétnost, zajištění na základě neprůhledného materiálu, ze kterého je vyrobeno krytí objektivu. Není proto možné jednoznačně určit, který prostor je kamerou snímán. Mezi nevýhody tohoto typu kamery lze zařadit nemožnost výměny objektivu. [1]

PTZ IP kamery:

- označení PTZ je převzato ze tří anglických slov: Pan (pohyb po horizontální ose), Tilt (pohyb po vertikální ose) a Zoom (přiblížení). Polohovací mechanismy lze ovládat jak automaticky, tak i manuálně, což zajistí značné rozšíření těchto typů kamer především pro bezpečnostní aplikace. Obsluha nejčastěji provádí ovládání PTZ IP kamery pomocí joysticku nebo klávesnice. Následně jsou zde uvedeny jednotlivé druhy PTZ IP kamer: [1]
 - mechanické IP PTZ kamery: tyto kamery nacházejí uplatnění zejména pro monitorování vnitřních prostor,
 - nemechanické IP PTZ kamery: výhodou daného provedení je neslyšitelnost veškerých pohybů, které kamera vykonává. V současné době se vyskytují

v provedení s megapixelovými obrazovými snímači a širokoúhlým objektivem.

- IP PTZ DOME kamery: tato kategorie využívá nejrůznějších moderních funkcí současných IP kamerových systémů. Kamery jsou vhodné pro vnitřní i vnější použití. Poskytují neomezený pohyb v samostatných osách a při spojení s dome krytem je zajištěna určitá diskrétnost a obtížněji se lokalizuje objektiv kamery. [1]

1.2.3 Komunikace IP kamery

Každá IP kamera neboli síťová kamera má svoji IP adresu a obslužné funkce, které zajišťují síťovou komunikaci. Kamerová jednotka obsahuje software pro web server, File Transfer Protocol (dále jen FTP) server, FTP klienta a e-mailového klienta. Řídící procesor společně s pamětí DRAM a Flash pamětí zajišťuje komunikaci s webovým serverem, komunikaci se sítí, řízení programových I/O nebo případně inteligentní analýzu obrazu. [1]

Na základě fungování IP kamery v ethernetové síti vzniká kompatibilnější zařízení pro záznam videa. V takovém případě nejsou nutné další periferie pro ovládání, stačí jen přímé spojení k počítači nebo Digital Video Recorder (dále jen DVR), které jsou opatřeny s využitím Video Management System (dále jen VMS) softwarem. [1]

Komunikační část IP kamery se skládá ze tří základních částí:

- **hardwarové komunikační rozhraní IP kamery**

Pro datový přenos z IP kamery je možné využít různé typy konektorů. Některé IP kamery mohou disponovat celou řadou specifických konektorů, oproti tomu jiné IP kamery mohou mít zase méně konektorů nezbytných pro datový přenos. Jelikož nabídka IP kamer na trhu je celkem široká, tak každý výrobce nabízí určitou specifickou aplikaci, aby zaujal zákazníka. Nejčastěji se pro přenos videa využívá koaxiální kabel, UTP kabel, případně konektory pro umístění antén zajišťující bezdrátovou komunikaci skrze WiFi. V některých případech může být IP kamera doplněna o logické I/O, výstup pro připojení externího mikrofону pro zajištění lepší kvality zvuku nebo konektor pro napájení. Jak již bylo zmíněno výše, tak napájení lze také realizovat přes UTP kabel prostřednictvím technologie PoE. Připojení UTP kabelu bývá realizováno prostřednictvím konektoru RJ45. [1]

Pro připojení externích zařízení k IP kameře lze využít již zmíněné logické vstupy a výstupy. Pokud je požadován přenos informací z Poplachových zabezpečovacích a tísňových systémů

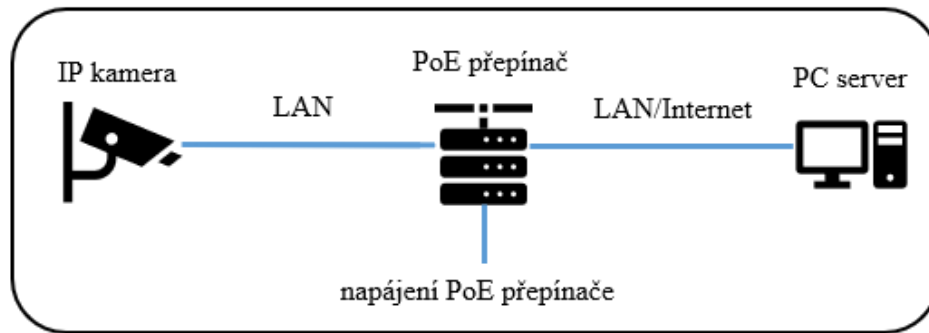
(dále jen PZTS) nebo Access Control System (dále jen ACS) jsou především využívány logické vstupy. Procesor IP kamery provede vyhodnocení logických informací, na základě těchto informací je provedeno např. spuštění nahrávání, změna pozice IP kamery, nebo další předem nastavené funkce. Oproti tomu logické výstupy lze použít pro odeslání informace do externích zařízení. Může se jednat např. o turnikety, ovládání brány, dveřní zámky, případně další prvky inteligentních systémů budov, které vykonávají svou činnost samostatně na základě logického výstupu. [1]

Názorným příkladem může být rozpoznávání Registrační značky (dále jen RZ) u vozidel pomocí IP kamery. IP kamera nejdříve provede nasnímání a uložení obrazu RZ. Následně je provedena kontrola s databází, ve které jsou uloženy RZ, pro které je přístup povolen. Pokud je RZ shodná, tak nastane odeslání logické informace na řídicí jednotku brány, která provede automatické otevření brány.

Častějším případem, kde se lze setkat s rozpoznáváním RZ jsou kontroly na dálnicích. Kde dochází ke kontrole, zda jedoucí vozidlo po dálnici má zaplacenou dálniční známku. Dříve byla dálniční známka ve formě nálepky, kterou bylo potřeba nalepit na čelní sklo. V současné době došlo k nahrazení fyzické dálniční známky známkou elektronickou, ke které je přiřazena určitá RZ.

- **přenosové technologie síťového videa**

V současné době mezi nejpoužívanější počítačový protokol pro komunikaci lze řadit Internet Protocol (dále jen IP) protokol. Jedná se o základní protokol, který je používán pro internet, e-mail a také v podstatě pro jakoukoliv novou síť. Silnou předností daného protokolu je totožná funkce jak pro malé firmy, tak i pro velké společnosti, případně nabízí možnost rozšíření o další komponenty. Postupem let nastalo zdokonalení ethernetu, což zapříčinilo i zvyšování přenosové rychlosti datové sítě. Jednotlivé IP kamery je možné připojovat prostřednictvím standardu 100BASE-T, který poskytuje datový přenos o rychlosti 100 Mbit/s s využitím kroucené dvojlinky. Pokud je požadováno připojení více IP kamer, tak se využívá síťový prepínač (Network Switch), tento prepínač také poskytuje zpětnou kompatibilitu s dřívějším, méně používaným standardem Ethernetu 10Base-T. Tyto síťové prepínače poté nesou označení 10/100 prepínače. Nabízí se také možnost tyto prepínače doplnit technologií PoE a zajistit tak napájení IP kamer přes datovou síť. Z důvodu stále se zvyšujících nároků na přenosovou rychlost se kromě UTP kabelů využívají také optická vlákna, které lze nalézt nejčastěji u páteřních vedení propojující rozsáhlé sítě. [1]



Obr. 4. Připojení IP kamery přes PoE přepínač. [1]

- **komunikace IP kamery v síti**

Každé zařízení, které se připojí k internetu musí obsahovat jedinečnou IP adresu. IP adresa se skládá celkem ze čtyř částí, které jsou označovány jako oktety. Mezi jednotlivými částmi se nachází tečka, každá část obsahuje číslo z rozsahu 0-255. Prakticky se lze setkat s IP adresami v rozsahu od čtyř 0 po čtyřikrát 255. IP kameru lze považovat za koncové zařízení, které musí být osazené síťovou kartou Network Interface Card (dále jen NIC). Na základě vytvořené topologie sítě je provedeno adresování paketů přes přepínače do koncových zařízení v síti. Mezi nejpoužívanější protokol pro přenos dat přes datovou síť je využíván protokol TCP/IP, tento protokol plní zároveň funkci nosiče pro jiné protokoly. [1]

1.3 Hybridní kamerové systémy

Častým důvodem vytváření hybridního kamerového systému je rozšíření nebo doplnění stávajícího analogového kamerového systému o další kamery, které už jsou digitální. Důvodem k rozšíření stávající analogové struktury o novou digitální strukturu bývají nejčastěji finanční náročnost, nebo limit připojení kamer. Jelikož Network Video Recorder (dále jen NVR), DVR atd. mají jen omezený počet slotů pro připojení koaxiálního kabelu. Může se např. jednat o chráněné objekty, ve kterých se nachází analogové kamerové systémy. Tyto systémy mohou být plně funkční, ale kvůli chráněnému interiéru není možné provést kompletní nové rozvody pro digitální kamery. Z toho důvodu jsou zachovány analogové kamerové systémy v interiéru a digitální kamery jsou umístěny pouze ve venkovních prostorech, kde jsou kamery nejčastěji umístěny na plotech nebo na veřejném osvětlení a není tak problém nových rozvodů. Prvkem mezi analogovými a pouze IP kamerami jsou HD-SDI kamerové systémy. Tyto systémy jsou ideální v místech, kde byla již stávající kabeláž zavedena. Předností je jednoduchost instalace a kompatibilita různých záznamových zařízení

s odlišnými výrobci. HD-SDI kamerové systémy poskytují hybridní systém pro analogové, IP kamery a HD-SDI kamery, nebo třeba nekomprimované video bez zpoždění. [1],[2]

Další možností spojení stávající koaxiální kabeláže s IP kamerami je pomocí systému Veracity, jehož prvky jsou připojeny na oba konce koaxiálního vedení. Na jeden z těchto prvků je připojena pomocí PoE LAN IP kamera, druhý prvek je propojen pomocí PoE LAN se Switchem. Veracity tedy představuje adaptér, který obsahuje vstup pro koaxiální kabel a POE LAN, přičemž je nutné uvažovat dva kusy adaptéru na jedno vedení koaxiálního kabelu. [2]

1.4 Spotřeba kamerových systémů

Spotřeba je důležitý parametr, který se posuzuje nejčastěji při návrhu rozsáhlých kamerových systémů. Pokud se jedná pouze o jednu IP kameru, která bude v provozu 24 hodin, tak spotřeba takové kamery v porovnání s větším množstvím kamer bude zanedbatelná. Ale v případě, že se jedná o rozmístění několika desítek až stovek IP kamer v objektu (může se jednat o rezidenční objekty, firemní objekty nebo třeba obchodní centra), tak spotřeba takového množství IP kamer může být značně vyšší. Při stále zvyšujících cenách za energii není pro firmu ekonomické, aby byly IP kamery neustále v provozu, i když třeba na záznamu nic není. Proto výrobci kamer přicházejí s funkcemi, které aktivují IP kameru pouze v případě, jestliže je v prostoru detekován pohyb. Tím je zajištěno, že IP kamery nejsou neustále v provozu, ale jejich zapnutí nastane pouze při detekování pohybu. Detekce pohybu může být zajištěna pomocí PIR detektoru, mikrovlnného detektoru, IR detektoru a dalších různých typů detektorů. Na základě použitého typu detektoru může být zajištěna detekce pouze lidí nebo i věcí, které nevyzařují tepelné záření.

Tato práce se věnuje vytvoření diagnostické sestavy, pomocí které bude možné aktivovat IP kameru pouze v případě, kdy bude detekován pohyb pomocí IR senzoru. Tím bude zajištěno, že IP kamera nebude stále aktivní, ale její zapnutí nastane pouze při detekci narušitele. Diagnostická sestava byla vytvořena za účelem snížení spotřeby IP kamery. Dále pomocí Hall-ova senzoru bude probíhat měření odebíraného proudu na vodiči a vnitřního napětí na Hall-ově senzoru. Pro měření spotřeby lze využít velké množství měřících přístrojů, může se jednat o např.: digitální měřiče spotřeby, wattmetry, proudové senzory a další. Měření spotřeby lze realizovat také např. pomocí Hall-ova senzoru proudu, který bude využit v diagnostické sestavě, Hallův senzor je detailněji popsán v kapitole 3.1.8.

1.5 Současné trendy v oblasti kamerových systémů

Velkým přínosem pro IP kamery bylo využití přísvitu infračerveným světlem určeného pro noční monitoring. InfraRed (dále jen IR) přísvit umožňuje IP kamerám zobrazit snímanou scénu i v naprosté tmě. Nevýhodou IR přísvitu je nezachování původních barev snímané scény, výsledný video signál je reprezentován v černobílém spektru. IR přísvit může být dvojího typu, buď vestavěný uvnitř IP kamery, nebo může být použit jako externí zařízení. IR diody používané pro IR přísvit u IP kamery nejčastěji využívají infračervené světlo o vlnové délce 850 nm nebo 940 nm. Uspořádání IR diod pro přísvit může být buď parabolické nebo v seskupení jako soustředné kružnice. [1], [2]

Další zajímavou aplikací mohou být termokamery, jejich princip spočívá na záznamu tepelného záření, které je vyzařováno objektem nebo osobou ve snímané scéně. Snímaný obraz není nijak ovlivněn přítomností viditelného světla. Předností tepelného záření může být lepší průchod kouřem nebo mlhou. Termokamery nacházejí široké uplatnění, může se jednat o vyhledání ohnisek požáru, noční vidění, detekce pohybu osob, monitorování úniku tepla z budovy, kontrola potrubí a spoustu dalších aplikací. [2]

Mezi trendy kamerových systémů, lze zařadit: radarové technologie pro detekování pohybu prostřednictvím elektromagnetických vln. Často využívanou funkcí je detekce zvuků, kdy lze nastavit, aby kamerové systémy reagovaly na konkrétní specifický zvuk. Může se jednat o rozbití oken nebo křik. Další zajímavou funkcí, kterou některé IP kamery obsahují je rozsvícení světla v IP kameře při detekování pohybu z chráněné zóny, světlo může případného pachatele vystrašit. [6]

Řada výrobců nabízí aplikaci do chytrého telefonu, přes kterou lze ovládat IP kameru. Lze např. přes chytrý telefon komunikovat třeba s kurýrem, přes reproduktor zabudovaný v IP kameře.

IP kamery nachází široké uplatnění pro rozpoznávání tváří, nejčastěji na letištích, nádražích nebo ve městech. V případě širokého pokrytí IP kamer může operátor jednoduše přepínat mezi jednotlivými IP kamerami a zjistit tak např. pohyb podezřelé osoby nebo automobilu.

Další často využívanou funkcí je rozpoznávání tváře neboli Face Detection. V současné době dochází k širokému rozšíření IP kamer, analogové kamery jsou sice na ústupu, nicméně podíl jejich výskytu je stále široký.

1.6 Integrace kamerových systémů

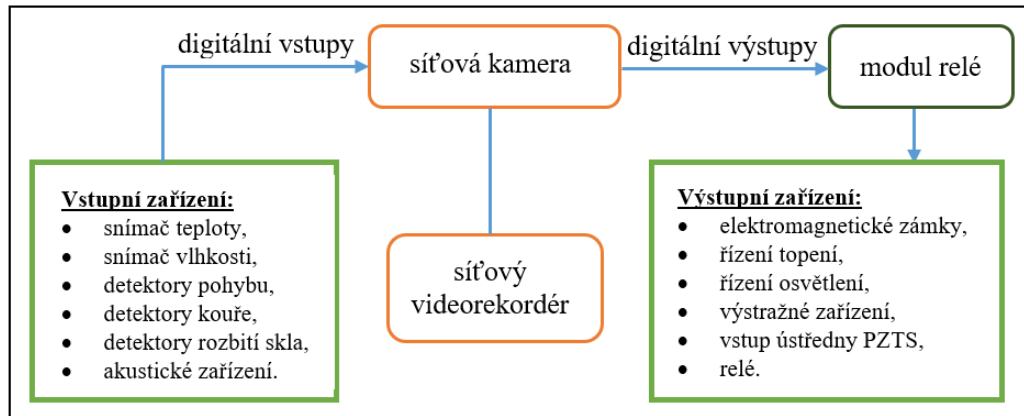
Kamerové systémy mohou být využívány samostatně, nebo v kombinaci s dalšími bezpečnostními aplikacemi. Pokud jsou kamerové systémy využívány společně s dalšími aplikacemi, tak lze tyto systémy nazývat: integrované poplachové systémy (dále jen IPS). V takovém systému lze např. definovat stavy, které nastanou při aktivaci některé další aplikace. Za často využívaný typ integrace lze považovat spojení přístupového systému s kamerovým systémem. [7]

Při integraci: systémy kontroly vstupu (dále jen SKV)/dohledové videosystémy (dále jen DV), tak princip spočívá v komunikaci řídicí jednotky SKV pomocí protokolu TCP/IP s prvkem DV (digitální videorekordér). Vytvořená integrace poskytuje možnost nastavení událostí v SKV, které zapříčiní aktivování záznamu na digitálním videorekordéru. [7]

Příkladem takového systému může být videotelefon. Jedná se o zařízení, které kombinuje domácí telefon společně s elektronickým vrátným a přenosem video signálu. Systém obsahuje malou zabudovanou kameru nejčastěji se širokoúhlým objektivem, přísvitem v IR oblasti nebo ve viditelném spektru. Pro přenos videosignálu je využíván kroucený pár sdělovacího kabelu. Vnitřní ovládací zařízení lze realizovat za použití přenosné obrazovky nejčastěji o úhlopříčce 5 palců, která poskytuje kvalitní obraz a poměrně velkou obrazovou plochu pro obsluhu. Životnost takového zařízení není příliš vysoká, jelikož často dochází ke krádeži nebo k vandalismu. [5]

Na Obr. 5 je vyobrazena integrace pomocí síťové kamery. S využitím digitálních vstupů a výstupů je možné nastavovat vazby mezi stavy vstupních zařízení a činnostmi kamery (nebo nastavení odezvy výstupních zařízení). Nejčastěji se lze s takovou integrací v praxi setkat při zajišťování následujících činností:

- záběry z kamery jsou odesílány, jakmile nastane aktivace definovaného vstupu,
- regulace topení, klimatizace, osvětlení a jiných aplikací při detekci pohybu osob ve snímaném obraze,
- definování výstupních (reakčních událostí) podle úrovně intenzity zvuku,
- přímá signalizace poplachu s využitím zařízení, které je připojeno na výstup kamery,
- detekce poplachu při vniknutí je signalizována na ústřednu PZTS, současně je jeden z jejich vstupů připojen na výstup kamery,
- odemykání nebo zamykání dveří podle příslušného stavu vstupních zařízení. [7]



Obr. 5. Integrace s využitím síťové kamery. [7]

2 SPECIFIKACE PLATFORMY ARDUINO

Prvotní zmínky o platformě Arduino pochází z italského Interaction Design Institute. Hlavním cílem konstruktérů Arduina bylo vytvořit snadno programovatelné a levné vývojové sady pro studenty. Arduino se stalo mezi studenty oblíbené a nastalo jeho rozšíření do celého světa. Na rozšíření se především podíleli jednotliví uživatelé či programátoři, kteří své vytvořené projekty a schémata sdílejí většinou na sociálních sítích nebo na různých diskusních fórech. Samotná platforma Arduino je považována za Open Source projekt. Na trhu lze nalézt velké množství desek Arduino, řada z nich také vznikala ve spolupráci se společností např. Intel. Některé desky Arduino vynikají svým specifickým tvarem nebo velikostí, díky čemuž mohou být používány pro řadu zajímavých projektů. [8]

Vedle originálních desek Arduino lze koupit i neoficiální desky Arduino, označované jako klony. Rozdíl mezi originální deskou Arduino a klonem je především v ceně, absenci oficiálního nápisu a odstínu desky Arduino. Dalším rozpoznatelným rysem je, že v názvu je často uvedeno *-duino* (jelikož název Arduino podléhá ochraně autorskými právy, názvy jako *-duino* a další podobné jsou ale přípustné). Důvodem vzniku klonů jsou veřejně přístupná schémata a rozpis použitých součástek na výrobu základní desky. Nic tak nebrání tomu, aby si kdokoliv sestavil Arduino třeba doma v garáži. Originální desky Arduino jsou zeleno modré, oproti tomu klony jsou tmavě modré. Platforma Arduino bývá tvořena třemi hlavními částmi:

- základní deskou s mikropočítačem např. Arduino Mega,
- vývojovým prostředím pro programování např. Arduino Integrated Development Environment (dále jen IDE),
- knihovnamí, které je potřeba nejdříve importovat do software. [8]

Platforma Arduino je vhodná jak pro začínající samouky, tak i pro pokročilé programátory, jelikož lze najít spoustu základních příkladů i projektů, které jsou obtížnější. Případně se nabízí možnost vytvoření celého vlastního projektu. [8]

Z pohledu kompatibility je Arduino kompatibilní s velkou řadou senzorů, snímačů a dalších komponent, např.: infračervený senzor, ultrazvukový senzor, Hallův senzor proudu a spoustu dalších komponent. Připojením těchto senzorů lze celkem levně vytvořit zajímavé měřicí nebo detekční obvody. [8]

Každá deska Arduino je tvořena procesorem, nejčastěji se jedná o procesory od firmy Atmel, případně firmy Intel. Základní deska také obsahuje vstupní a výstupní piny pro připojení dalších komponent. Pro napájení se využívá nejčastěji Universal Serial Bus (dále jen USB) port typu B, který se nachází u desky Arduino Mega, nebo napájecí konektor, kterým disponuje většina desek Arduino. [8]

Celý sortiment desek má typický design a význačnou modrou barvou. Na trhu se lze setkat s deskami, které mají za svým názvem přidáno označení Rev3 nebo R3. Uvedené označení podává informaci o čísle verze konkrétní desky. V praxi to znamená, mírné změny v modelech nebo odlišný design, nicméně takové úpravy nevyžadují konstrukci nové desky. Mnoho desek obsahuje převodník, který zprostředkovává komunikaci mezi počítačem a mikrokontrolerem. Ostatní desky kvůli úspoře místa využívají externích převodníků, případně těch zabudovaných v procesorech. Na některých deskách lze nalézt port RJ45, slot na microSD kartu nebo více USB portů. Mezi nejznámější desky Arduino lze řadit: Mini, UNO, nebo Mega 2560. [8]

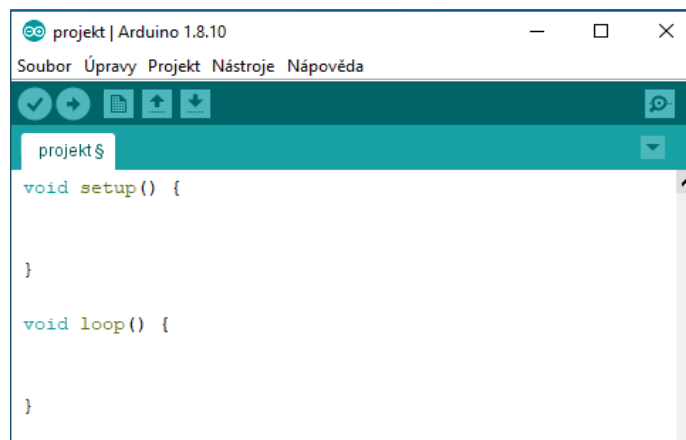
2.1 Arduino shieldy

Shieldy lze přirovnat k WiFi kartě, grafické kartě nebo zvukové kartě, které jsou připojeny k počítači. Tyto periférie jsou připojovány za účelem dosažení kvalitnějšího poslechu hudby nebo třeba lepšího grafického zpracování obrazu. U Arduina jde v podstatě o stejný princip, jen v trochu menším provedení. Arduino shieldy jsou využívány k doplnění funkcí, kterými základní deska nedisponuje. Na trhu se lze setkat s rozsáhnou nabídkou shieldů. Nejpoužívanější shieldy jsou: Ethernet shield, WiFi shield, Motor shield a spoustu dalších. Propojení shieldů s deskou Arduina je provedeno pomocí průchozích pinů, které jsou na shieldu obsaženy. Umístění průchozích pinů přesně odpovídá umístění pinů na desce Arduina. Nicméně je důležité věnovat pozornost kompatibilitě shieldu s deskou Arduina, aby nenastala kolize. [8]

2.2 Programování Arduina

Programování platformy Arduina lze provést v různých softwarech, např: C++, C, ale mezi nejběžnější spadá Arduino IDE, které lze bezplatně stáhnout z oficiálních stránek Arduina. Při otevření nového projektu jsou předdefinovány dvě smyčky. První smyčka je označena jako void setup, zde se nachází část kódu, která se provede po aktivaci tlačítka reset, nebo při uložení programu do desky Arduina, tento kód se opakuje pouze jednou. Kód, který se

opakuje po celou dobu se zapisuje do smyčky void loop. Základ každého kódu obsahuje tyto dvě smyčky, void setup a void loop. [8], [9]



Obr. 6. Vývojové rozhraní Arduino IDE.

Zde se nachází screen z vývojového prostředí Arduino IDE. Uprostřed lze vidět smyčku void setup a void loop společně se složenými závorkami. Pro ověření kódu, zda se v něm nenacházejí chyby lze využít kulaté tlačítko umístěné vlevo. Při zjištění chyb nastane zobrazení informací ve spodní části. Druhé tlačítko zleva může být využito k nahrání kódu do desky Arduina. Jakmile je nahrání dokončeno, tak se zobrazí nápis: uloženo. Zobrazení sériového monitoru probíhá pomocí aktivování tlačítka se symbolem lupa na pravé straně. [8]

2.2.1 Proměnné

Jako proměnná se označuje pozice v paměti, do níž se provede ukládání dat. U proměnné lze rozlišovat jméno, typ dat, která budou ukládána a hodnotu. Proměnou lze využít tam, kde dochází k častému opakování stejných hodnot. Např. časová prodleva u zapínání LED diod. V případě nastavení časové prodlevy ručně pro pět LED diod, bude nutné při změně prodlevy ručně přepisovat všech pět hodnot. Při použití proměnné stačí změnit pouze jednu hodnotu. Využití proměnných je uvedeno na Obr. 7 až Obr. 9. [8]

```
byte x; //deklarace proměnné x

x = 12; //přiřazení hodnoty 12 do proměnné x

//spojení dvou předešlých operací do jedné operace
byte q = 12;
```

Obr. 7. Deklarace proměnné a přiřazení hodnoty do proměnné.

```
prodleva_LED_1(15);  
prodleva_LED_2(15);  
prodleva_LED_3(15);  
prodleva_LED_4(15);  
prodleva_LED_5(15);
```

Obr. 8. Nastavení časové prodlevy bez proměnné.

```
byte K = 15;  
  
prodleva_LED_1(K);  
prodleva_LED_2(K);  
prodleva_LED_3(K);  
prodleva_LED_4(K);  
prodleva_LED_5(K);
```

Obr. 9. Nastavení časové prodlevy s proměnnou.

Funkce v obou případech je stejná, časová prodleva u všech LED diod je nastavena na 15 ms. Nicméně při změně prodlevy v prvním případě je nutné upravit všech pět hodnot, ale v případě druhém stačí upravit pouze nadefinovanou proměnnou. Proměnná, která byla použita má název *K*, je datového typu *byte* a její hodnota je 15.

2.2.2 Datové typy

Ke každé proměnné je přiřazen datový typ, který podává informaci o tom, jaká data se mohou v proměnné nacházet. Může se jednat o logické hodnoty (true/false), znaky, či čísla. [8] Následující řádky popisují nejčastěji používané datové typy a jejich specifikace

Číselné datové typy:

- **byte** – má velikost 8 bitů a je využíván k ukládání celých čísel v rozsahu 28 hodnot – 0 až 255,
- **integer** – datový typ nachází uplatnění pouze u celých čísel, přičemž rozsah je závislý na typu procesoru – tedy od záporných do kladných hodnot, se středovou hodnotou přibližně 0,
- **long** – tento datový typ je používán k ukládání celočíselných 32bitových hodnot,
- **float** – datový typ nachází uplatnění při práci čísel s desetinnou čárkou. [8]

Logický datový typ:

- **boolean** – je využíván pouze pro ukládání dvou hodnot, tedy true (pravda), nebo false (nepravda). [8]

Znakový datový typ:

- **char** – pro ukládání jednoho konkrétního znaku textu je využit právě datový typ char, který je uložen jako číselná hodnota v ASCII tabulce,
- **string** – používá se pro uložení řetězce znaků, nejčastěji bývá popisován pomocí pole znaků. [8]

```
byte prom_1 = 12; //byte

int prom_2 = 500; //integer

long prom_3 12121212; //long

float prom_4 = 2.876; //float

boolean prom_5 = true; //boolean

char prom_6 = 'A'; //char
char prom_6 = 65;

String prom_7 = ""; //String
```

Obr. 10. Přehled datových typů.

2.3 Propojení Arduina s počítačem

Desku Arduino Mega lze propojit s počítačem pomocí USB kabelu. Pro propojení je nutné využít USB kabel, který bude mít na jednom konci USB konektor typu A pro připojení do počítače a na druhém konci USB konektor typu B pro připojení k desce Arduino Mega. Jakmile je provedeno propojení, tak se na počítači objeví instalační okno. Pokud je instalace provedena v pořádku, pak je nutné vybrat sériový port. Tato operace se provede v záložce nástroje -> port -> vybrat sériový port, ke kterému je deska Arduino připojena. Z oznamovací lišty na Obr. 11 lze vyčíst informace o připojení desky Arduino, jedná se o Arduino Mega 2560, které je připojené na port COM7. [8]

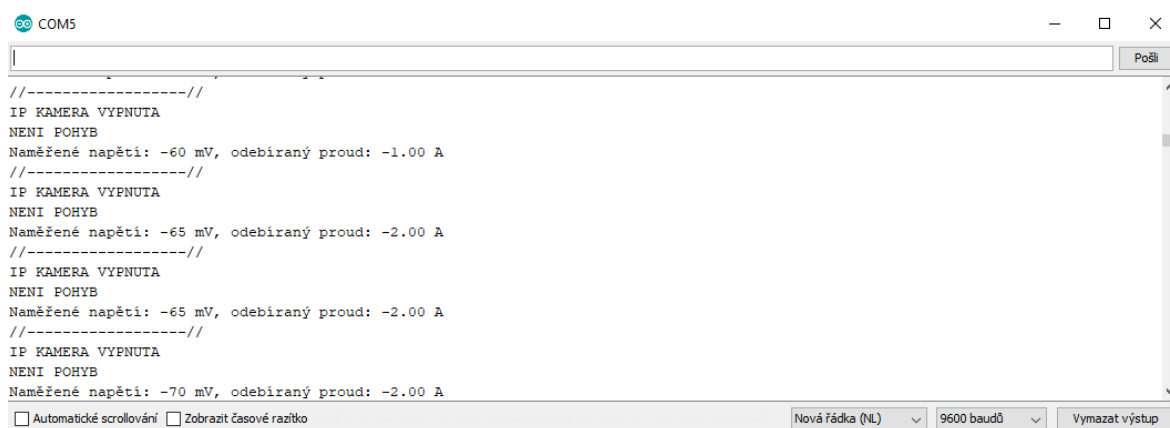


Obr. 11. Informace o připojení na port COM7.

2.4 Sériová komunikace

Pro práci se sériovou komunikací je potřeba otevřít záložku sériový monitor, čímž se zobrazí okno, ve kterém lze vkládat příkazy a sledovat výpis hlášek. S využitím sériové komunikace lze posílat určitá data do desky Arduina, případně sériovou komunikaci lze využít pro výpis dat. Sériová komunikace nachází uplatnění především při ladění programu. Programátor tak může pomocí výpisu na sériový monitor zjistit místo vzniku chyb. Aby mohla být sériová komunikace zahájena, je potřeba napsat do smyčky void setup příkaz *Serial.begin(9600)*. Hodnota v závorce označuje rychlost komunikace mezi počítačem a Arduinem, která odpovídá počtu přenosů za sekundu. U komunikace s počítačem není výběr rychlosti na uživateli, lze použít jen několik málo definovaných hodnot. Běžné komunikační rychlosti, se kterými se lze setkat jsou 1200, 2400, 4800, 9600, 19200, 38400, 57600 a 115200 Bd, jedná se o násobky 1200 Bd. Nicméně nejčastěji používanou hodnotou je hodnota 9600 baudů (tedy 9600 bitů za sekundu), která odpovídá osminásobku hodnotě 1200 Bd (8 bitů na baud). Tato hodnota se většinou uvádí v datasheetu výrobku. Vypsání textu na sériový monitor probíhá pomocí příkazu *Serial.println(„text“)*. [8]

Na Obr. 12 se nachází rozhraní sériového monitoru s aktuálním výpisem informací. Vlevo dole se nachází dvě zatrhávací tlačítka pro automatické rolování a pro zobrazení časového razítka.



Obr. 12. Otevření sériového monitoru.

II. PRAKTICKÁ ČÁST

3 NÁVRH A REALIZACE PROTOTYPU

Cílem práce bylo navrhnout a realizovat diagnostickou sestavu pro měření spotřeby IP kamery. Monitorovací sestava bude ovládat činnost IP kamery pomocí IR senzoru a Radio Frequency Identification (dále jen RFID) čtečky. Pro sledování videa z IP kamery byla vytvořena aplikace ve Visual Studiu.

Stručný popis funkce diagnostické sestavy pro měření spotřeby IP kamery:

základem diagnostické sestavy byla platforma Arduino. Zapnutí IP kamery nastane při detekci pohybu narušitele pomocí Infrared Radiation (dále jen IR) senzoru. Jakmile IR senzor detekuje pohyb, tak nastane sepnutí relé, které následně provede zapnutí IP kamery a rozsvítí se modrá Light Emitting Diode (dále jen LED dioda). Vypnutí IP kamery bude možné pouze s využitím SPRÁVNÉ karty, jedná se o RFID kartu pracující na frekvenci 13,56 MHz. Po přiložení SPRÁVNÉ karty na RFID čtečku nastane rozsvícení zelené LED diody po dobu 2000 ms, zároveň nastane zhasnutí modré LED diody a IP kamera je vypnuta. Pokud by někdo přiložil NESPRÁVNOU kartu, tak se na 2000 ms rozsvítí červená LED dioda, ale IP kamera zůstane stále zapnuta.

Po celou dobu provozu diagnostické sestavy se na Liquid Crystal Display (dále jen LCD displej) budou vypisovat oznámení, zda je IP kamera zapnuta, vypnuta, nebo zda je přiložena správná nebo nesprávná RFID karta.

Jelikož zobrazovací kapacita LCD displeje jsou pouze dva řádky o 16 znacích, tak není možné na LCD displej vypisovat podrobnější informace. Pro tento účel budou podrobnější informace vypisovány na sériový monitor. Pokud není detekován pohyb pomocí IR senzoru, tak hodnoty vnitřního napětí na Hallově senzoru i odebíraný proud na vodiči budou nulové. Pokud nastane detekce IR senzorem, tak se na sériový monitor vypíše naměřené vnitřní napětí na Hallově senzoru a odebíraný proud na vodiči při zapnuté IP kameře, dále se zobrazí aktuální datum a čas, kdy byl pohyb detekován. Po přiložení SPRÁVNÉ karty nastane na sériovém monitoru zobrazení informací o přiložení správné RFID karty, pokud by náhodou někdo přiložil nesprávnou RFID kartu, tak se na sériový monitor vypíše jedinečné ID nesprávné karty. Sériový monitor se především využívá, pokud je požadováno sledování informací o aktuální činnosti určitého systému. Nevýhodou sériového monitoru je, že vypsaná data nelze zálohovat. Z toho důvodu se budou všechny informace, které se zobrazí na sériovém monitoru zároveň ukládat na microSD kartu do souboru *data.txt*. Po vložení microSD karty do notebooku lze informace zobrazit a případně zálohovat.

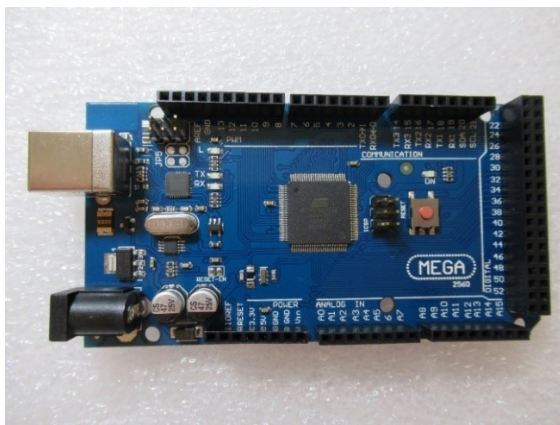
Pro sledování aktuálního videa z IP kamery je využita knihovna tříd pro přípravu grafického rozhraní Windows Presentation Foundation (dále jen WPF) skládající se z XAML a C# příkazů. Aplikace disponuje tlačítkem pro připojení a odpojení IP kamery, včetně zobrazovacího okna pro sledování aktuálního streamu z kamery. V aplikaci lze nastavit nespočet funkcí, např. čtení Quick Response (dále jen QR) kódů, rozpoznávání obličeje, vytvoření snímku obrazu nebo třeba definování funkce Motion detection pro zvýraznění pohyblivých objektů a spoustu dalších funkcí. Aby bylo zajištěno úspěšné propojení aplikace s IP kamerou, tak je potřeba při vytváření aplikace správně zakomponovat IP adresu IP kamery. Důležitou podmínkou je připojení do stejné sítě internet, ve které je přes switch připojena IP kamera.

3.1 Použité komponenty

Pro konstrukci navržené diagnostické sestavy posloužily níže uvedené komponenty. Všechny použité komponenty pochází od firmy Arduino, tužíš je zaručena vzájemná kompatibilita komponent. Každá součástka je doplněna o fotografii, zapojení vývodů, popis funkce a specifikace.

3.1.1 Arduino Mega

Na trhu lze nalézt buď originální desku Arduino Mega, nebo klon Arduino Mega. Rozdíl mezi těmito deskami spočívá pouze v ceně, barvě základní desky, popisku a odlišném umístění některých součástek. Pro tuto sestavu byla použita originální deska Arduino Mega. Deska Arduino Mega je osazena procesorem ATmega2560 s taktovací frekvencí 16 MHz. Na desce lze nalézt celkem 54 digitálních vstupů/výstupů. Dále deska obsahuje Inter-Integrated Circuit (dále jen I2C) rozhraní, charakterizované piny Serial Clock (dále jen SCL) a Serial Data (dále jen SDA), které jsou využívány např. pro připojení LCD displeje. Deska Arduino Mega pracuje s provozním napětím 5 V, na desce lze také nalézt port 3,3 V pro připojení např. RFID čtečky. Pro celkový reset základní desky lze využít vyvedené resetovací tlačítko na desce Arduina, jedná se o červené tlačítko v pravé části. Programování a propojení s počítačem probíhá nejčastěji přes USB kabel, pro jeho připojení se využívá USB port typu B na desce Arduina. Pro napájení může také posloužit napájecí konektor v levé spodní části, ke kterému lze připojit adaptér nebo bateriový box. [10]



Obr. 13. Arduino Mega.

3.1.2 RFID čtečka karet RC522

RFID čtečka obsahuje integrovanou anténu nacházející se na základní desce. Pro napájení čtečky se používá 3,3 V s provozním proudem 13 až 26 mA. RFID čtečka pracuje s frekvencí 13,56 MHz. Provozní teplota RFID čtečky je udávána v rozmezí od $-20\text{ }^{\circ}\text{C}$ do $80\text{ }^{\circ}\text{C}$. Uvedenou čtečku lze připojit jak k Arduino, tak i k Raspberry Pi a mnoha dalším platformám. Komunikace je zajištěna prostřednictvím Serial Peripheral Interface (dále jen SPI) sběrnice. Základní deska čtečky je osazena celkem osmi vývody pro připojení. V Tab. 1 jsou uvedeny jednotlivé vývody, které lze na čtečce najít a piny na desce Arduina, ke kterým jsou připojeny. Při vytváření programu je nutné nejdříve importovat knihovnu pro RFID čtečku: **MFRC522.h**. [11], [12]

Tab. 1. Zapojení vývodů z RFID čtečky na piny Arduina.

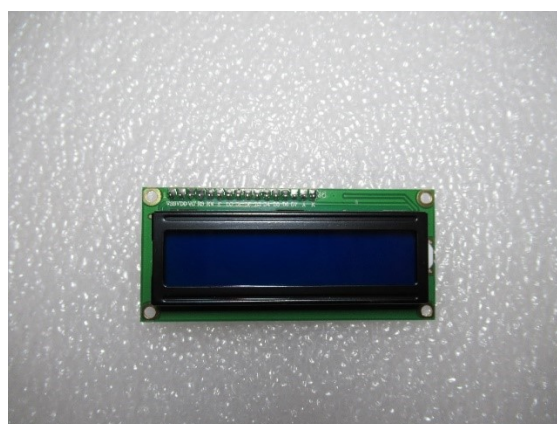
vývody na RFID čtečce	piny na Arduinu
VCC	3,3 V
GND	GND
RESET	5
SDA (synchronní data)	53
SCK (clock)	52
MOSI (master OUT/slave IN)	51
MISO (master IN/slave OUT)	50
IRQ (přerušeni)	není zapojeno



Obr. 14. RFID čtečka RC522.

3.1.3 LCD display

LCD displej náleží mezi výstupní zařízení, které dovedou zobrazit uložený text. Zobrazení textu probíhá na dvou řádcích po šestnácti znacích. Napájení displeje je zajištěno pomocí 5 V. Na zadní straně displeje se nachází I2C převodník. Uvedený převodník lze využít ke snížení počtu připojovaných pinů. Při použití převodníku je nutné připojit pouze piny označené SCL, SDA, VCC a GND ke stejnojmenným pinům na desce Arduina. Komunikace s platformou Arduino probíhá přes Two-Wire Interface (dále jen TWI) rozhraní. Pro lepší viditelnost informací na LCD displeji se využívá modré podsvícení, podsvícení lze regulovat potenciometrem umístěným na zadní straně displeje. Někdy může nastat situace, že na displeji není nic vidět, je proto nutné pootočit s potenciometrem. Při vytváření programu je nutné nejdříve importovat knihovnu pro LCD displej: `LiquidCrystal_I2C.h`. [13], [14]



Obr. 15. LCD displej.

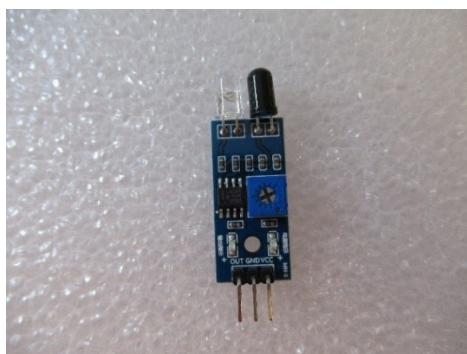


Obr. 16. LCD displej a I2C převodník.

3.1.4 IR senzor

Infračervený senzor překážek je osazen dvěma diodami, první plní funkci vysílače a druhá zajišťuje funkci přijímače. Vysílací infračervená dioda je používána pro vysílání paprsku a přijímací dioda pro příjem odraženého paprsku od překážky. Princip spočívá v tom, že vysílač vyšle signál o dané frekvenci. V případě, že signál narazí na překážku, tak nastane odražení signálu do přijímače. Překážkou může být např. člověk, který se pohybuje v detekční oblasti. Detekční dosah lze regulovat pomocí potenciometru. Přibližný dosah se uvádí 0 až 6 cm. Provozní napětí se udává 5 V. Infračervený senzor obsahuje tři vývody: napájení, uzemnění a výstupní pin, který je připojen na pin č. 3 na desce Arduina. Signalizace na senzoru je zajištěna dvojicí zelených LED diod.

Pokud svítí pouze jedna LED dioda, tak není paprsek přerušen, jakmile se ale rozsvítí druhá LED dioda, tak nastane přerušování paprsku, což indikuje, že IR senzor detekuje pohyb. [15]



Obr. 17. IR senzor.

3.1.5 Modul relé 5 V

Jedná se o 1 kanálové relé s fotoelektrickou izolací, relé obsahuje indikaci napájecího napětí. Pracovní napětí se uvádí 5 V. Rozhraní 1 kanálového relé je typu Normally Open (dále jen NO). Relé lze ovládat stejným způsobem jako např. rozsvícení a zhasnutí LED diody. V případě, že výstupní pin, na který je připojena LED dioda, má hodnotu HIGH (5 V), nastane rozsvícení LED diody. Pokud ale má výstupní pin hodnotu LOW (0 V), LED dioda zhasne. U relé nastává stejná situace s tím rozdílem, že některé moduly mohou mít přehozené logické úrovně. V takovém případě by spínání LED diody nastávalo při 0 V (LOW) a rozpínání při 5 V (HIGH). Relé celkem obsahuje šest vývodů pro připojení, tři vstupní piny pro připojení k Arduino a tři výstupní pro připojení např. IP kamery. Na výstupní piny lze připojit zařízení, které bude ovládáno. S využitím relé lze spínat např. osvětlení nebo IP kamery. Tab. 2 uvádí zapojení vstupních pinů, které jsou připojeny k desce Arduino. [16], [17]

Tab. 2. Vstupní piny na relé.

vstupní vývody na relé	piny na Arduino
D+	5 V
D-	GND
IN	13

Výstupní piny se připojují k zařízení, které se bude ovládat, v tomto případě se jedná o IP kameru. Výstupní vývody jsou sice tři, ale pro připojení zařízení jsou využívány pouze dva. Připojení se provede pomocí svorky COM a dále je potřeba vybrat, zda zařízení bude připojeno smyčkou Normally Closed (dále jen NC) nebo NO.



Obr. 18. Modul relé 5 V.

3.1.6 Ethernet Shield W5100

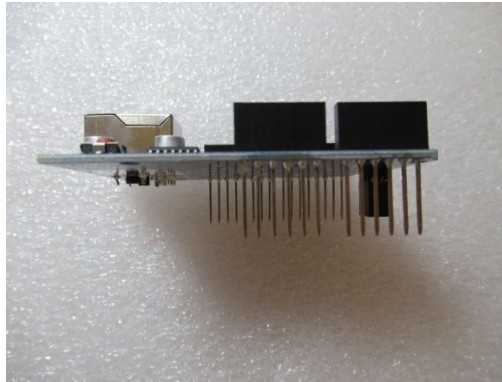
Uvedený modul W5100 lze využít jako server nebo jako klient. Modul je osazen kontrolérem W5100. Může podporovat až čtyři souběžné spojení. Shield obsahuje konektor RJ45 pro připojení síťového kabelu a slot pro paměťové karty typu microSD. Shield může být použit v kombinaci s různými typy desek, např. Mega, Uno atd. Mezi výhody lze zařadit vyvedení tlačítka pro Reset na Shield. Signalizace činnosti Local Area Network (dále jen LAN) portu je zajištěna pomocí indikačních LED diod. Obvodové zapojení W5100 podporuje práci s adresami typu Internet Protocol version 4 (dále jen IPv4), komunikace probíhá rychlostí 10 nebo 100 Mb/s. Komunikace s Arduinem je zajištěna pomocí protokolu SPI. Propojení Shieldu s Arduinem je provedeno pomocí průchozích konektorů na desce, není potřeba připojování pomocí kabelů. Na začátku kódu je potřeba nadefinovat knihovnu pro Ethernet Shield: **Ethernet.h**, tato knihovna je již obsažena v Arduino IDE, takže není potřeba knihovnu importovat. V Tab. 3 jsou uvedeny piny, které Ethernet Shield používá. [18], [19]

Tab. 3. Popis jednotlivých pinů, které využívá Ethernet Shield.

vývody na Ethernet Shieldu	použití pinů
11	SPI
12	SPI
13	SPI
10	CS signál W5100
4	CS signál slotu pro paměťové karty



Obr. 19. Ethernet Shield W5100.



Obr. 20. Ethernet Shield W5100 – vývody na propojení.



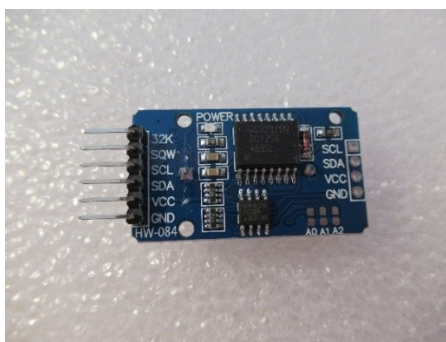
Obr. 21. Propojení Shieldu s Arduinem.

3.1.7 RTC hodiny reálného času

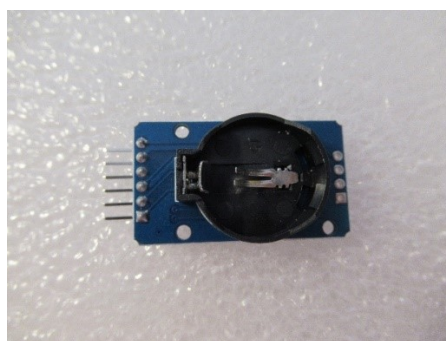
Jedná se o přesný I2C modul reálného času Real-time clock (dále jen RTC) s integrovaným teplotně kompenzovaným krystalovým oscilátorem a krystalem. Modul se skládá ze dvou částí, první obvod DS3231 je označován jako RTC hodiny reálného času. Druhý obvod AT24C32 je paměťový modul typu Electrically Erasable Programmable Read-Only Memory (dále jen EEPROM), kapacita úložiště je 32 KB. Používané rozhraní I2C, maximální přenosová rychlost 400 kHz. Provozní napětí modulu je 5 V. Uplatnění nachází jak v komerční, tak i v průmyslových aplikacích. Modul reálného času udává sekundy, minuty, hodiny, den, datum, měsíc a rok. Dále nabízí uživateli funkce: korekce pro přestupný rok, datum ukončení, možnost nastavení méně než 31 dnů. Data a adresy jsou přenášeny přes I2C sběrnici. Pro správnou funkci je potřeba vložit baterii LIR2032 do patice, která se nachází na druhé straně modulu, baterie zajišťuje pojistku při výpadku napájení. Při vytváření programu je nutné nejdříve importovat knihovnu: **DS3231**. V Tab. 4 je uvedeno připojení vývodů na desku Arduino Mega. [20], [21]

Tab. 4. Zapojení vývodů z RTC modulu na piny Arduina.

vývody na modulu RTC	piny na Arduinu
VCC	5 V
GND	GND
SCL	SCL
SDA	SDA



Obr. 22. RTC hodiny reálného času.



Obr. 23. RTC hodiny reálného času – zadní strana, patice pro vložení baterie.

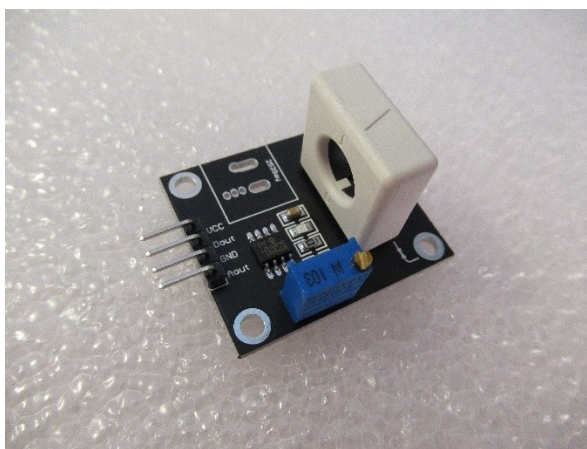
3.1.8 Hallův senzor proudu

Jedná se o vstupní modul, který je používán pro měření proudu procházejícího vodičem, je vybaven měřicím prvkem WCS1700, díky kterému lze měřit stejnosměrný i střídavý proud. Měřicí rozsah je omezen na ± 70 A pro stejnosměrný proud a 50 A pro proud střídavý. Senzor disponuje dvěma výstupy: analogový a digitální. Sepnutí digitálního vstupu nastane při překročení stanovené hodnoty nastavené trimrem. Pro správnou funkčnost je nutné zajistit provozní napětí 5 V. Při samotném měření je nutné protáhnout vodič skrz Hallův senzor proudu. S využitím Hallova senzoru proudu lze měřit spotřebu na IP kameře a zjistit, kdy IP kamera

odebírá největší proud. Pro propojení desky Arduino Mega se senzorem proudu je potřeba propojit celkem tři vodiče. Propojení vodičů je uvedeno v Tab. 5. [22], [23]

Tab. 5. Zapojení vývodů z Hallova senzoru na piny Arduina.

vývody na Hallově senzoru	piny na Arduinu
VCC	5 V
GND	GND
Aout	A0



Obr. 24. Hallův senzor proudu.

Pro analogový pin lze použít jiné volné piny. Pro práci s digitálním výstupem lze použít výstupní pin Dout. Průměr otvoru pro protažení vodiče je 9 mm, lze tedy snadno modulem protáhnout různé typy vodičů. [22], [23]

Na začátku je potřeba nastavit kalibrační konstantu, která odpovídá měření naprázdno. Jedná se o měření, když vodičem, který je protažen přes Hallův senzor neprochází žádný proud. Při takovém měření by měla být hodnota napětí v ideálním případě 0 mV. Měřením naprázdno se stanovila kalibrační konstanta na hodnotu 2478, při uvedené hodnotě bylo prováděno měření. V datasheetu pro Hallův senzor je uvedena přibližná konstanta, která by teoreticky měla odpovídat měření naprázdno. Nicméně tato konstanta se upravila pro zajištění přesnějších výsledků při měření s využitím navržené diagnostické sestavy. Správné stanovení konstanty se zajistilo tak, že při měření byl proud na vodiči i vnitřní hodnota napětí na Hallově senzoru nulová.

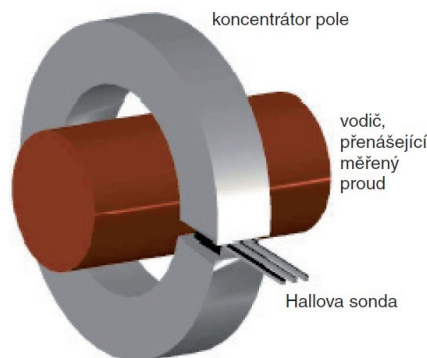
Nejdříve probíhá načtení hodnoty z analogového pinu do proměnné *hodnota*. Ze získané hodnoty proběhne výpočet vnitřního napětí na Hallově senzoru. Dále je potřeba nastavit

napájecí rozsah 0 až 5000. Tento rozsah byl odvozen od hodnoty napájecího napětí 5 V. Jelikož rozlišení, které uvádí výrobce je v jednotkách mV/A i výsledné naměřené napětí bude taktéž v mV, tak hodnota 5 V byla převedena na mV, tomu odpovídá hodnota 5000. Rozsah je uvažován od nuly. Na základě těchto skutečností byl určen rozsah 0 až 5000. Následně analogový rozsah 0 až 1023, jedná se o vnitřní hodnotu AD převodníku s 10bitovým rozlišením 2^{10} , tedy 1024 hodnot, tomu odpovídá rozsah 0 až 1023. [23]

Výpočet hodnoty vnitřního napětí na Hallově senzoru je proveden pomocí funkce map, která je využita pro přepočítání hodnot z jednoho rozsahu do jiného rozsahu. Vstupní hodnoty funkce map jsou: analogová hodnota získaná z Hallova senzoru, původní rozsah (0-1023) a analogový rozsah (0-5000). Výsledkem funkce map je hodnota, která je přepočítaná v novém rozsahu. Na závěr je odečtena kalibrační konstanta, kterou bylo nutné nastavit pro měření naprázdno.

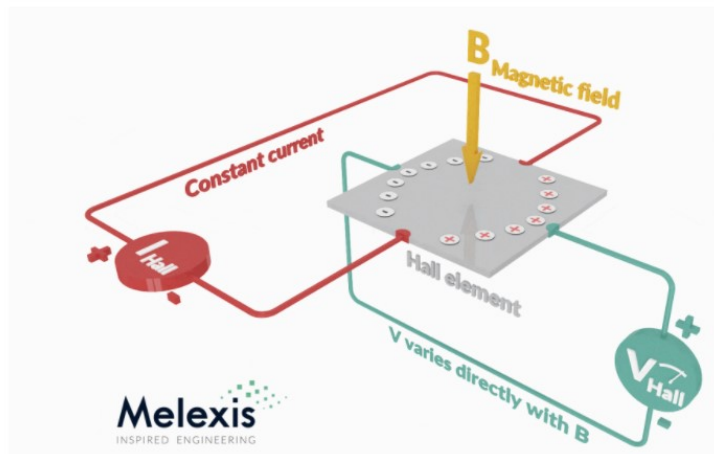
Hodnota proudu je vypočtena tak, že hodnota vnitřního napětí na Hallově senzoru je vydělena rozlišením senzoru WCS1700, které uvádí výrobce, jedná se o hodnotu 32 mV/A ($m\Omega$). [22], [23]

Princip měření pomocí Hallova senzoru proudu je založen na bezdotykovém typu měření. Modul se skládá z feritového jádra, měřeného vodiče a Hallova senzoru. Hallův senzor je magneticky citlivá polovodičová součástka. Tato polovodičová součástka je složena ze senzoru a integrovaného zesilovače. Magneticky propustné feritové jádro je vyrobeno ve specifickém tvaru C s velmi malou mezerou. V uvedené mezeře mezi konci feritového jádra je umístěn Hallův senzor, který generuje napětí, jenž je přímo úměrné proudu, který protéká vodičem. [24], [25]



Obr. 25. Schéma měření pomocí Hallova sondy. [24]

Roku 1879 Edwin Hall objevil efekt, který byl později pojmenován po svém objeviteli, jedná se o Hallův jev. Princip Hallova jevu je založen na procesu, při kterém přes polovodičovou desku protéká proud. Na základě působení kolmého magnetického pole dochází k přeskupení náboje v destičce na jednu stranu. Vytvořené přeskupení definuje měřitelné napětí na obou okrajích. Získané napětí je přímo úměrné síle magnetického pole. [26]



Obr. 26. Princip Hallova jevu. [26]

Legenda:

- *Hall element* – Hallův senzor
- *Magnetic field* – magnetické pole
- *Constant current* – konstantní proud
- *V varies directly with B* – V (napětí) se mění přímo s B (indukcí)

Jako Hallův jev lze označit proces, kdy v polovodiči vzniká elektrické pole za stálého působení vnějšího elektrického i magnetického pole. Tím dochází k seskupení záporného náboje na jedné straně vodiče a kladného náboje na druhé straně vodiče. Jelikož oba póly mají různý potenciál, tak dochází k vzniku Hallova napětí. Výstupní Hallovo napětí je následně zesíleno zesilovačem. Poté je přivedeno napětí do voltmetru kalibrovaného v ampérové stupnici. Proud procházející vodičem, je proud, který indikuje uvedený měřicí přístroj. Za výhody této metody lze považovat přesnost měření a galvanické oddělení od měřeného obvodu. [24], [25]

3.1.9 Paměťová karta

Pro ukládání informací o IP kameře a měřené spotřebě posloužila microSD karta od výrobce SanDisk. Uvedená microSD karta poskytuje kapacitu 32 GB. Paměťová karta je dodávána společně s SD adaptérem, který lze jednoduše vložit do notebooku.



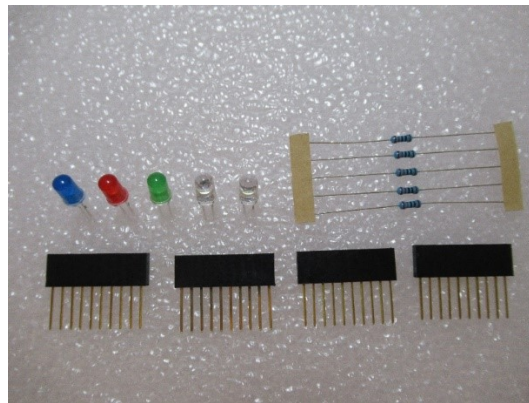
Obr. 27. MicroSD karta a adaptér.

3.1.10 Ostatní komponenty

Dále byly použity LED diody, modrá pro signalizaci zapnuté IP kamery, zelená pro signalizaci přiložené správné RFID karty a červená LED dioda pro signalizaci přiložení nesprávné karty. Pro osvětlení monitorovací místnosti, kde se nachází IR senzor byla využita bílá LED dioda, stejně jak pro osvětlení Hallova senzoru proudu. Tyto bílé LED diody svítí po celou dobu, když je diagnostická sestava v provozu. Anoda LED diody se připojí na GND, katoda LED diody se přes rezistor 220 R připojí na výstupní pin Arduina. Rezistor se u LED diod nachází z důvodu, aby nedošlo k odpálení LED diody. Pro celkový reset prototypu je využito externě připojené tlačítko. Jeden vývod tlačítka se připojí na GND a druhý kontakt na pin Arduina označený jako RESET. Pokud bude diagnostická sestava napájena pouze z bateriového boxu s 9 V baterií, tak pro zapnutí/vypnutí lze využít spínač, který se nachází na přední straně. Svorkovnice pro GND a 5 V budou řešeny s využitím stohovatelné 10 pinové dutinkové lišty, jejíž konce se navzájem proleťují. Sběrnice SDA a SCL také bude řešena s využitím stohovatelné 10 pinové dutinkové lišty. Pro celkové propojení mezi komponenty byly využily F-M a M-M Dupont kabely o délce 20 cm.

Tab. 6. Zapojení výstupů na piny Arduino.

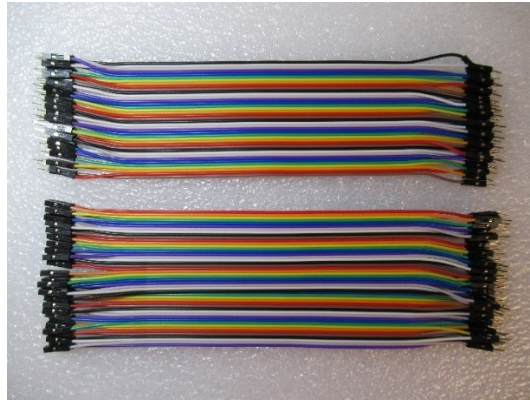
výstupy	piny na Arduino
bílá LED dioda – oblast IR	22
bílá LED dioda – oblast Halova senzoru	24
modrá LED dioda	26
červená LED dioda	28
zelená LED dioda	30
relé (IP kamera)	13



Obr. 28. LED diody, rezistory a stohovatelné dutinkové lišty.



Obr. 29. Kabelové průchodky, objímky na LED diody, tlačítko a spínač.



Obr. 30. Propojovací kabely.

3.2 Napájení prototypu

Pro napájení diagnostické sestavy lze využít dvojí řešení. Napájení pomocí bateriového boxu, ve kterém je umístěna baterie, nebo pomocí napájecího adaptéru. Bateriový box i napájecí adaptér mají stejný výstupní konektor, který lze připojit do desky Arduino Mega.

3.2.1 Napájení pomocí bateriového boxu

Napájení celé diagnostické sestavy je řešeno pomocí 1 x 9 V baterie, která je uložena v bateriovém boxu. Bateriový box je následně připojen kabelem do napájecího konektoru na desce Arduino Mega. Bateriový box se používá, pokud se diagnostická sestava často přemísťuje, nebo se v blízkosti nenachází zásuvka na 230 V



Obr. 31. Bateriový box.

3.2.2 Napájení adaptérem

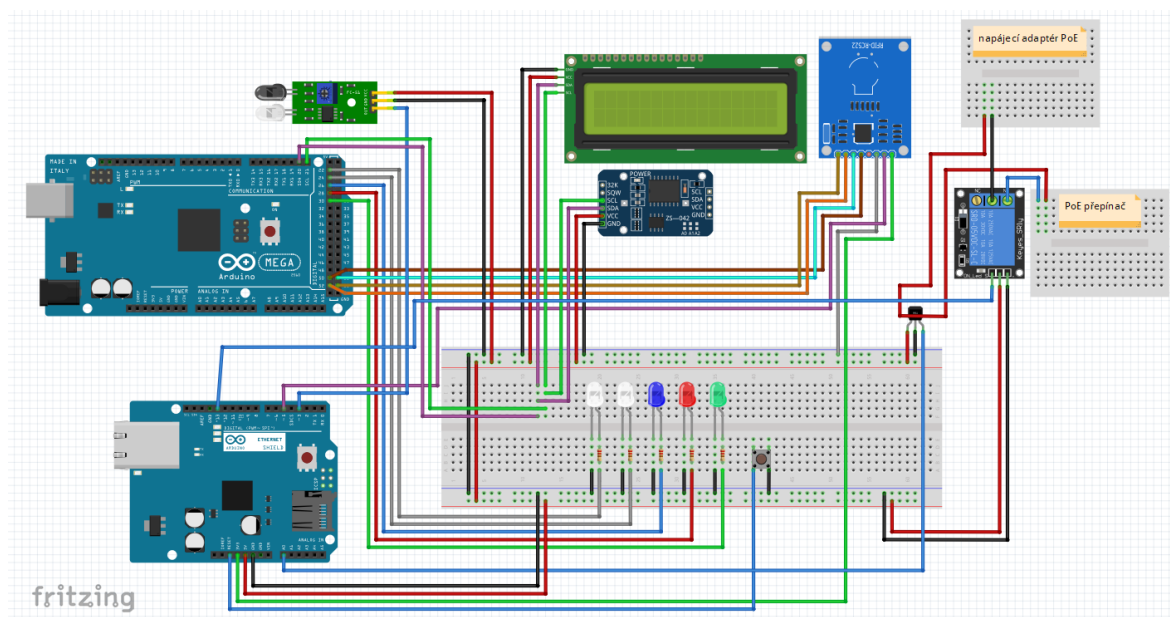
Pro napájení diagnostické sestavy lze využít také externí adaptér. Externí adaptér má stejný konektor pro připojení jako bateriový box, tudíž není problém připojit adaptér přímo do desky Arduino Mega. Adaptér lze využít pouze v případě, že diagnostická sestava bude umístěna na pevném místě v blízkosti zásuvky na 230 V.



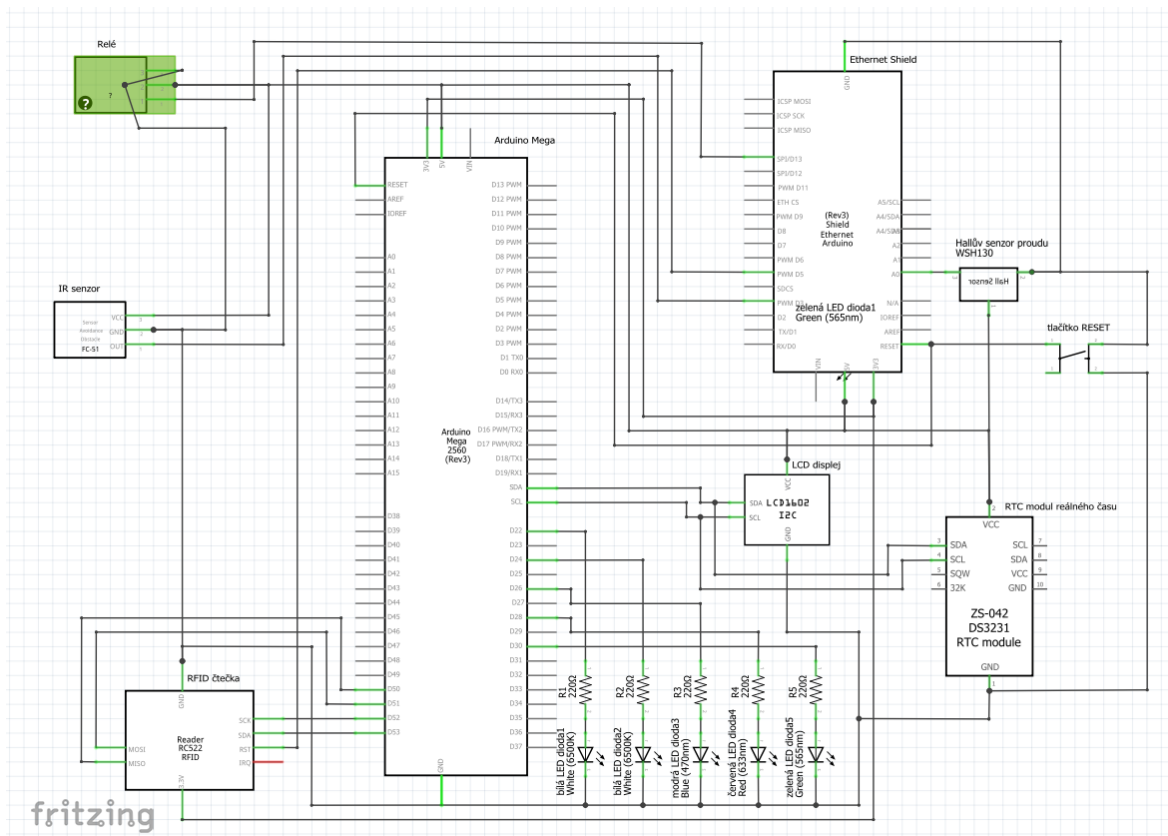
Obr. 32. Adaptér pro napájení.

3.3 Schéma prototypu

Schéma zapojení bylo vytvořeno za použití volně stažitelného software Fritzing [27]. V knihovně, která je obsažena v software se nachází pouze základní komponenty. Další potřebné komponenty bylo nutné vyhledat na internetu, stáhnout a importovat do knihovny.



Obr. 33. Schéma zapojení.



Obr. 34. Schéma zapojení.

3.4 Vytvořená diagnostická sestava

Prototyp diagnostické sestavy byl vytvořen převážně z kartonové proložky o tloušťce 4 mm. Ke spojování kartonové proložky posloužila tavná pistole, která dovede roztavit speciální tavné tyčinky, které zajistí pevné spojení kartonové proložky. Použitý materiál byl zvolen z důvodu snadné manipulace a poměrně nízké hmotnosti. Jelikož se jedná pouze o testovací prototyp, tak na použitý materiál nebyl brán velký zřetel.



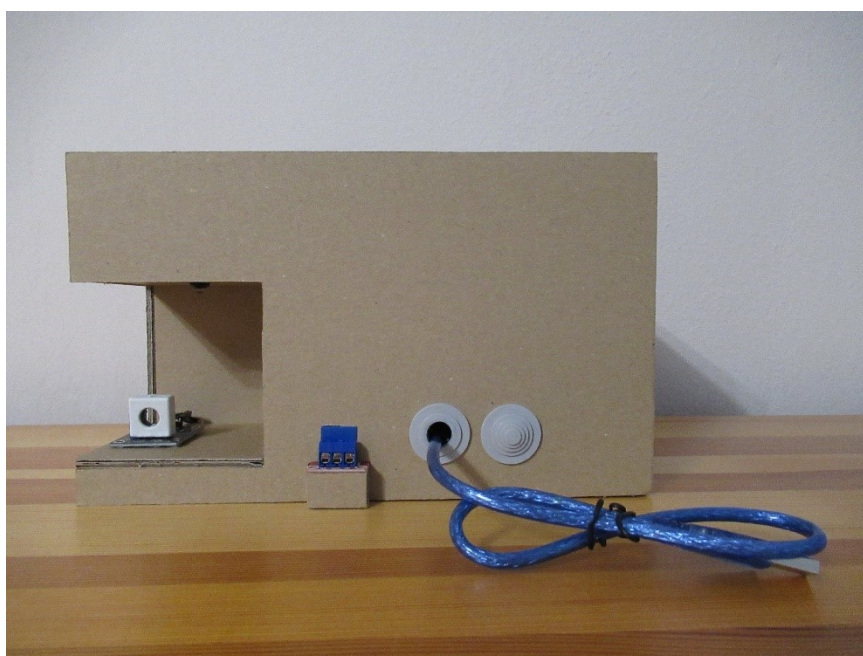
Obr. 35. Vytvořená diagnostická sestava – přední strana.

Na přední straně se nachází LCD displej pro zobrazení informací o zapnutí/vypnutí IP kamery. Pod LCD displejem jsou umístěny tři LED diody, modrá LED dioda signalizuje zapnutou IP kameru, zelená LED dioda signalizuje přiložení správné RFID karty a červená LED dioda signalizuje přiložení nesprávné RFID karty pro vypnutí IP kamery. V levé části se nachází tlačítko pro celkový reset diagnostické sestavy. Nad tlačítkem pro celkový reset je umístěn spínač, kterým je možné zapínat/vypínat diagnostickou sestavu při napájení z bateriového boxu. Pod LED diodami je umístěna RFID čtečka pro čtení karty. V pravé části je vytvořena monitorovací místnost, která je od vnitřní části oddělena kartonovou proložkou. Místnost je otevřena pouze ze přední strany, detekce je proto zajištěna pouze uvnitř místnosti.



Obr. 36. Umístění IR senzoru.

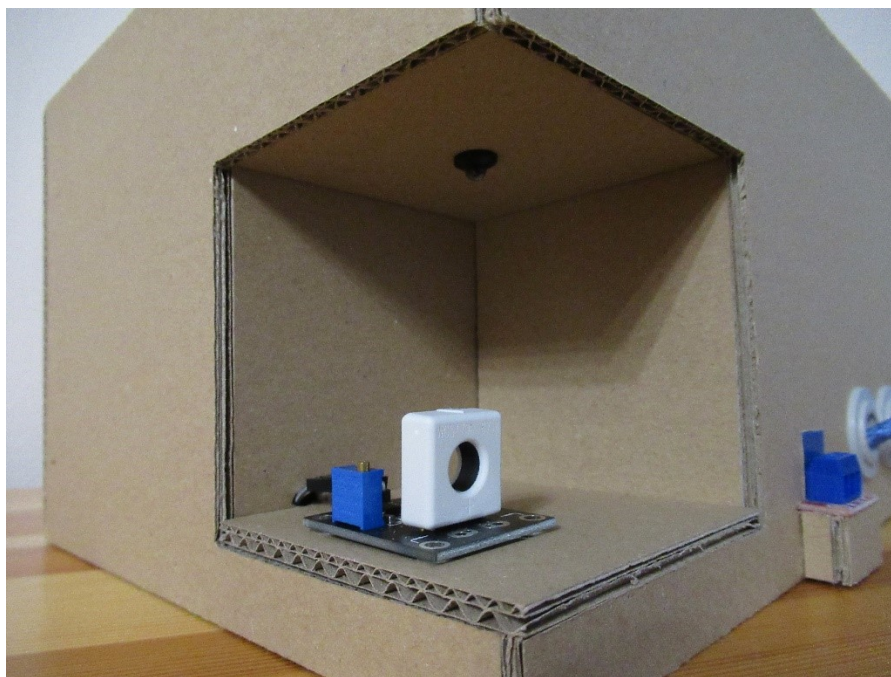
IR senzor je umístěn uprostřed horní části v monitorovací místnosti, aby bylo zajištěno co největší pokrytí spodní části. Za IR senzorem je v horní části umístěna bílá LED dioda, která zajišťuje osvětlení monitorovací místnosti, tato LED dioda svítí nepřetržitě, pokud je diagnostická sestava v provozu.



Obr. 37. Vytvořená diagnostická sestava – zadní strana.

Na levé straně je vytvořen prostor pro měření pomocí Hallova senzoru. Nad Hallovým senzorem je umístěna bílá LED dioda, která zajišťuje osvětlení Hallova senzoru, LED dioda

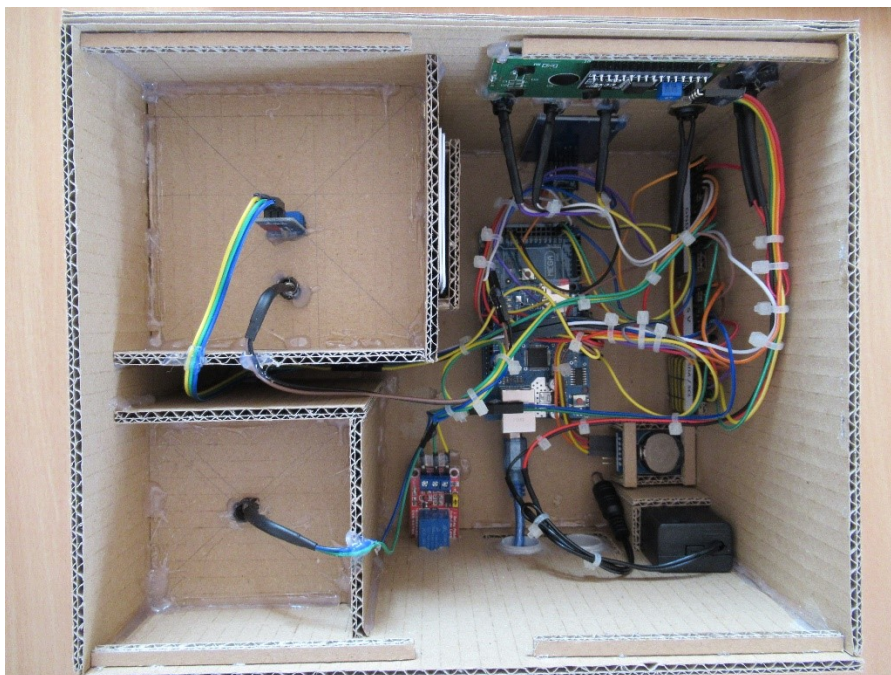
svítí nepřetržitě po dobu provozu diagnostické sestavy. Uprostřed zadní strany se nachází relé s vyvedenými vstupy. Na tyto vstupy je možné připojit IP kameru nebo jiné zařízení, které bude aktivováno, jakmile IR senzor detekuje pohyb. Přes kabelovou průchodku byl vyveden modrý USB kabel pro připojení diagnostické sestavy k notebooku.



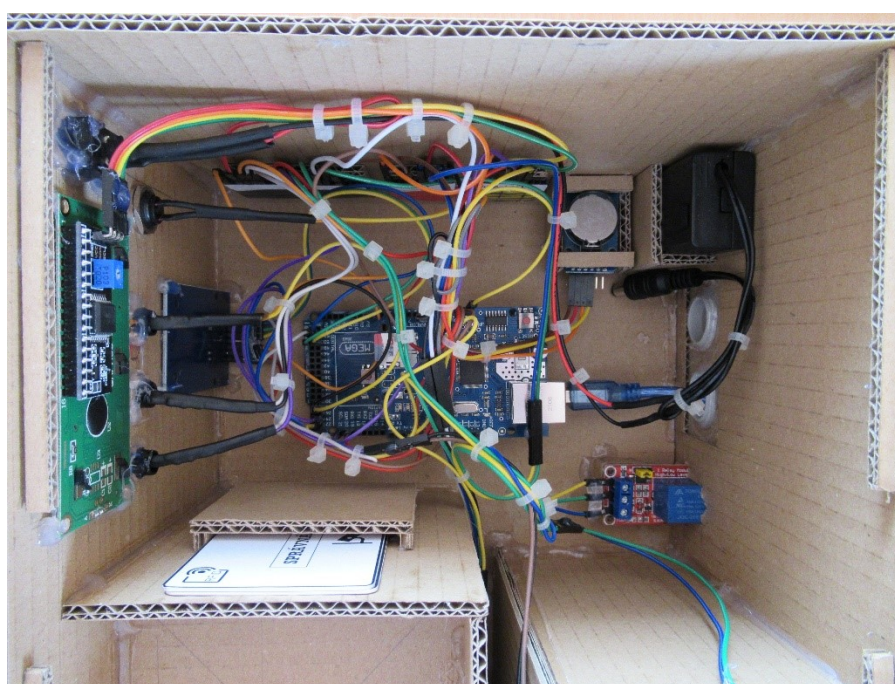
Obr. 38. Umístění Hallova senzoru.

Umístění Hallova senzoru bylo zvoleno tak, aby nebyla potřeba při měření spotřeby zasahovat do vnitřní části diagnostické sestavy, kde by mohlo omylem dojít ke zkratu. Prostor, kde je umístěn Hallův senzor je oddělen od vnitřního prostoru kartonovou proložkou tloušťky 4 mm.

Na Obr. 39 a Obr. 40 lze vidět oddělenou část pro detekci pohybu a další oddělenou část pro měření spotřeby s využitím Hallova senzoru. V pravé spodní části se nachází bateriový box, do kterého je vložena 9 V baterie, která zajistí napájení diagnostické sestavy. Nad baterií jsou umístěny RTC hodiny reálného času k zajištění přesného data a času kdy nastala detekce pohybu. Na pravé straně jsou umístěny svorkovnice. Na přední straně se nachází LCD displej, pod kterým jsou umístěny LED diody, spínač, tlačítko pro reset a RFID čtečka. V levé spodní části se nachází relé pro spínání IP kamery.



Obr. 39. Uložení komponent uvnitř sestavy.



Obr. 40. Uložení komponent uvnitř sestavy.



Obr. 41. Vytvořené svorkovnice.

Svorkovnice pro GND je složena celkem ze dvou deseti pinových stohovatelných dutinkových lišt, které jsou navzájem proleťovány a poté překryty izolační nevodivou páskou.

Pro svorkovnici 5 V posloužila jedna deseti pinová stohovatelná dutinková lišta. Vývody jsou taktéž proleťovány a následně obaleny izolační nevodivou páskou.

Svorkovnice pro SDA a SCL je vytvořena z jedné deseti pinové dutinkové lišty, jelikož pro SDA i SCL jsou využity pouze tři vývody. Vývody pro SDA i SCL jsou samostatně proleťovány a odděleny izolační nevodivou páskou, aby nedošlo ke zkratu.

Pro ovládání diagnostické sestavy posloužily RFID karty pracující na frekvenci 13,56 MHz



Obr. 42. RFID karty k ovládání diagnostické sestavy.

3.5 Použitá IP kamera

Pro testování diagnostické sestavy posloužily dvě IP kamery. IP kamera Vivotek FD8136B-F3 1Mpix, jedná se o provedení dome IP kamera. Jako další posloužila IP kamera Axis M1125, jedná se o fixní IP kameru. Aby bylo možné s IP kamera pracovat, tak bylo nezbytné zjistit jejich IP adresy.

Vivotek – IP adresa: **10.53.26.13**

Axis – IP adresu: **10.53.25.201**



Obr. 43. IP kamera Vivotek.



Obr. 44. IP kamera Axis.

3.6 PoE přepínač

Pro napájení IP kamery byl použit PoE přepínač značky tp-link, samotné napájení je zajištěno prostřednictvím UTP kabelu. Výhodou použití PoE přepínače je snížení počtu kabelů, jelikož přenos dat i napájení IP kamery bude zajištěno prostřednictvím UTP kabelu.

Na levé straně se nachází konektor RJ45 pro připojení UTP kabelu k IP kameře. Na pravé straně se nachází opět konektor RJ45, ale pro připojení na switch a také konektor pro připojení napájecího adaptéru.



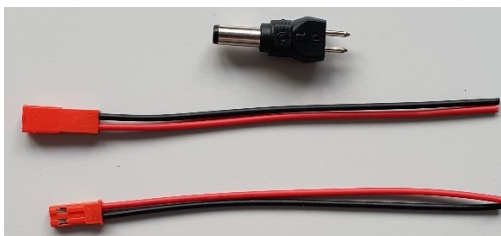
Obr. 45. PoE přepínač.

Původní konektor na adaptéru byl odstraněn a místo něj byly na konce vodičů naletovány drátky o délce 10 cm s JTS konektorem (samec) pro snadné připojení vodičů.

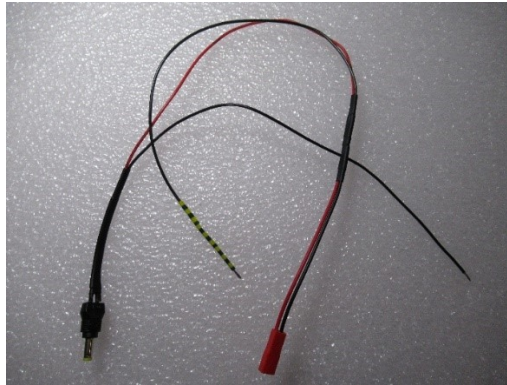


Obr. 46. Adaptér pro napájení PoE přepínače.

Jelikož byl původní konektor z adaptéru odstraněn, tak bylo potřeba vytvořit vhodnou kabeláž, které zajistí propojení napájecího adaptéru, PoE přepínače a relé. Na Obr. 47 se nachází JTS konektory (samec a samice) a konektor pro napájení PoE přepínače.



Obr. 47. Konektory pro kabeláž k PoE.



Obr. 48. Kabeláž pro připojení k PoE přepínači.

Konektor v levé části je použit pro připojení do PoE přepínače. Uprostřed se nachází JTS konektor (samice), který je propojen s JTS konektorem (samec) na kabelu od adaptéru. Vodič uprostřed, jehož konec je zelenožlutý, bude připojen na pin COM na relé. Zbývající černý vodič v pravé části je použit pro připojení na pin NO na relé. Měření Hallovým senzorem bude probíhat na přívodním vodiči od adaptéru.

Cílem takového zapojení bude zapínání a vypínání PoE přepínače pomocí relé. Jakmile nastane zapnutí PoE přepínače, tak se automaticky uvede do provozu IP kamera. IP kamera je propojena s PoE přepínačem pomocí UTP kabelu zakončeného koncovkou RJ45.

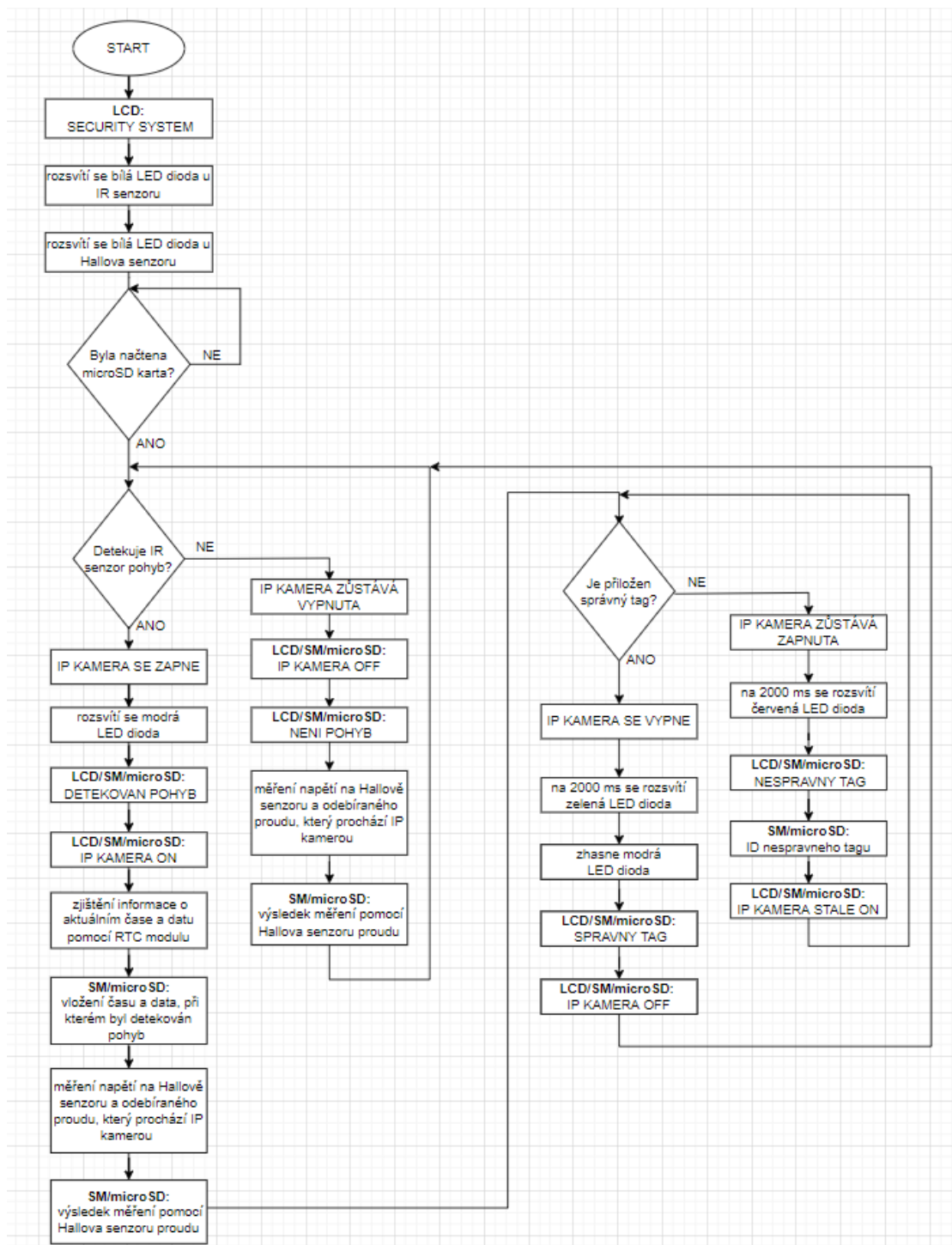
4 PROGRAMOVÉ VYBAVENÍ

Uvedená kapitola se věnuje vývojovému diagramu pro řízení diagnostické sestavy a vývojovému diagramu pro sledování obrazu z IP kamery. V závěru kapitoly budou popsány vybrané části zdrojových kódů v Arduinu IDE a Visual Studiu. Kompletní zdrojové kódy jsou uvedeny v elektronické příloze.

4.1 Vývojové diagramy

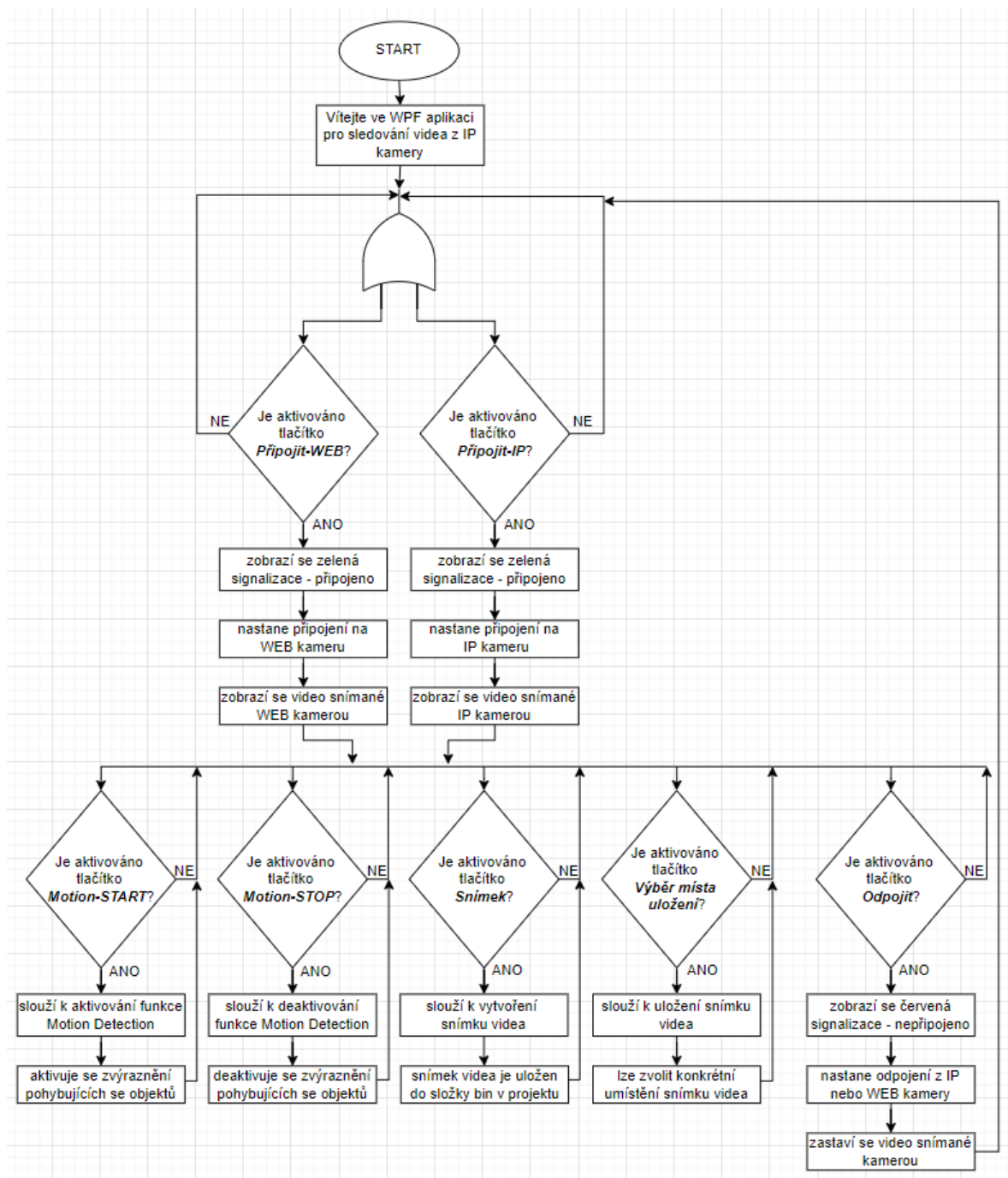
Před vytvářením obslužného programu byl sestaven vývojový diagram, podle kterého mohl být následně vytvořen obslužný program pro řízení diagnostické sestavy. Pro vytvoření vývojového diagramu posloužila webová stránka Diagrams.net [28]. Na uvedené webové stránce lze sestavit různé vývojové diagramy a následně provést jejich export do počítače pro možnost dalších úprav.

První vývojový diagram popisuje funkci diagnostické sestavy v programu Arduino IDE. Nejdříve se ověřuje, zda byla načtena microSD karta, pokud proběhlo načtení, tak se pokračuje dál, v opačném případě se program vrací na začátek a je proveden výpis o nesprávném načtení microSD karty. Dále se kontroluje, zda IR senzor detekuje pohyb, pokud není detekován pohyb, tak se provede patřičný výpis a program se vrací na začátek. Pokud je detekován pohyb, tak se zobrazí informace, které odpovídají detekci pohybu. Následně se ověřuje, zda je přiložena správná RFID karta. Pokud není přiložena správná karta, tak systém vyčká, dokud není přiložena odpovídající karta. Po přiložení správné karty na RFID čtečku, nastane vypnutí IP kamery. Během celého cyklu probíhá výpis na LCD displej, sériový monitor a ukládání záznamu na microSD kartu.



Obr. 49. Vývojový diagram – Arduino IDE.

Druhý vývojový diagram popisuje činnost aplikace, která je využívána pro sledování videa z IP kamery. Aplikace byla vytvořená v programu Visual Studio 2019, WPF aplikace. Program lze volně stáhnout z oficiálních stránek Visual Studio [29]. Dále bylo potřeba stáhnout a importovat knihovnu pro IP kameru, knihovnu lze volně stáhnout z webu OZEKI Camera SDK. [30]



Obr. 50. Vývojový diagram Visual Studio – aplikace WPF.

4.2 Popis zdrojového kódu v Arduino IDE

Před vytvořením smyčky void setup je provedeno úvodní nastavení, jedná se především o importování knihoven, definování vstupních a výstupních pinů, přiřazení pinů pro LCD displej a RFID čtečku. Důležitou částí je nastavení RFID karty určené pro vypínání IP kamery. Uvnitř smyčky void setup probíhá nastavování výstupních pinů pro relé, LED diody a vstupního pinu pro IR senzor. Dále je využit boolean stav, který slouží pro zajištění, že při vypnutí

IP kameře nebude detekována RFID karta. Na začátku smyčky void setup je boolean *stav_1* nastaven na logickou hodnotu „false“, což označuje vypnutou IP kameru a není možné číst data z RFID karty. Dále se po zapnutí diagnostické sestavy kontroluje připojení microSD karty, na sériový monitor proběhne výpis o správném nebo nesprávném načtení karty. Pokud je microSD karta načtena v pořádku, tak je na kartě je vytvořen soubor *data.txt*, do kterého se budou následně ukládat informace o činnosti diagnostické sestavy.

Na začátku smyčky void loop je provedeno rozsvícení dvou bílých LED diod, které svítí po celou dobu činnosti diagnostické sestavy. Tyto diody jsou využívány pouze k osvětlení oblasti u Hallova senzoru proudu a pro osvětlení monitorovací místnosti s IR senzorem. Jelikož měření pomocí Hallova senzoru probíhá, i když je IP kamera vypnuta, tak další část kódu se zabývá právě získáním naměřených údajů z Hallova senzoru a následně jejich převedením do podoby, ve které jsou uloženy na microSD kartu.

Následuje podmínka, která ověřuje, zda je aktivován IR senzor. Ta je řešena pomocí funkce IF. Pokud je IR senzor aktivován (nastane přerušování paprsku), proběhne sepnutí relé, které aktivuje IP kameru a zároveň se také rozsvítí modrá LED dioda. Tato činnost vede k zapnutí IP kamery a změně booleovské proměnné *stav_1* z „false“ na „true“.

Další podmínka ověřuje, zda *stav_1* je roven hodnotě „true“, pokud je podmínka splněna, pak se provedou postupně dvě části. První část funkce zahrnuje výpis informací o detekování pohybu, zobrazení data a času, kdy byl pohyb detekován spolu s výpisem odebíraného proudu na vodiči a naměřeného vnitřního napětí na Hallově senzoru. Druhá část funkce je použita pro čtení dat z RFID karty, která je přiložena na RFID čtečku. Proběhne porovnání přiložené RFID karty s kartou v databázi, pokud se RFID karta shoduje s kartou v databázi, tak proběhne vypnutí IP kamery a zhasnutí modré LED diody. Následně se na 2000 ms rozsvítí zelená LED dioda, která signalizuje přiložení správné RFID karty, zároveň se boolean *stav_1* nastaví na „false“.

Pokud by byla přiložena nesprávná RFID karta, tak se na 2000 ms rozsvítí červená LED dioda, IP kamera zůstává stále v provozu a boolean *stav_1* zůstává na hodnotě „true“.

4.2.1 Importování knihoven

Na Obr. 51 probíhá importování knihoven pro RFID čtečku, LCD displej, komunikační rozhraní SPI, RTC hodiny reálného času a čtečku microSD karet. Nejdříve bylo potřeba

uvedené knihovny stáhnout ze stránek Arduina u jednotlivých komponent. Následně mohly být knihovny importovány a použity pro tvorbu programu.

```
// importování knihoven
#include <MFRC522.h>           // knihovna pro RFID čtečku
#include <MFRC522Extended.h>  // knihovna pro RFID čtečku
#include <LiquidCrystal_I2C.h> // knihovna pro LCD displej
#include <SPI.h>              // knihovna pro komunikační rozhraní SPI

#include <Wire.h>             // knihovna pro RTC hodiny reálného času
#include <DS3231.h>          // knihovna pro RTC hodiny reálného času
#include <Ethernet.h>        // knihovna pro Ethernet Shield W5100

#include <SD.h>              // knihovna pro čtečku micro-SD
#include "RTClib.h"
#include <deprecated.h>
#include <require_cpp11.h>
```

Obr. 51. IDE – importování knihoven.

4.2.2 Nastavení pro vybrané komponenty

Pro microSD kartu bylo potřeba nastavit vývod CS na pin 4. Proměnná *zaznam* náleží pro ukládání dat na microSD kartu.

```
const int pinCS = 4;
File zaznam;
```

Obr. 52. IDE – Pin pro microSD kartu a proměnná pro uložení dat.

Pro datum a čas bylo potřeba nadefinovat novou proměnnou s označením *DatCas*. Kvůli snadnějšímu použití časové prodlevy byla nadefinována proměnná *PRODLEVA_2*, která se používá při rozsvícení zelené nebo červené LED diody na 2000 ms.

```
RTCDateTime DatCas;
# define PRODLEVA_2 2000
```

Obr. 53. IDE – proměnná pro datum a čas a časová prodleva.

Text je zobrazován na LCD displeji s označením 0x27 s dvěma řádky po šestnácti znacích. Vývod SDA u RFID čtečky byl nastaven na pin 53 a vývod RESET na pin s označením 5.

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
MFRC522 mfrc522(53, 5);
```

Obr. 54. IDE – nastavení LCD displeje a RFID čtečky.

4.2.3 Nastavení ID karty, která bude sloužit pro vypnutí IP kamery

Součástí diagnostické sestavy jsou dvě RFID karty pracující na frekvenci 13,56 MHz. Jedna karta je použita pro vypnutí IP kamery a druhá karta plní funkci nesprávné RFID karty.

```
# define karta_UI "86 99 C6 1F"
```

Obr. 55. IDE – nastavení povolené RFID karty.

4.2.4 Kontrola microSD karty

Po zapnutí diagnostické sestavy nastane kontrola vložené microSD karty. Zjišťuje se tak, zda je možné na microSD kartu provádět záznam, nebo jestli byla microSD karta správně načtena. Na sériový monitor se provede výpis, zda načtení microSD karty bylo úspěšné nebo neúspěšné. Pokud načtení proběhlo úspěšně, tak je na microSD kartě vytvořen soubor *data.txt*, který bude sloužit pro ukládání informací o činnosti diagnostické sestavy.

```
Serial.println("//-----//");  
Serial.println("probíhá načítání microSD karty");  
  
if (!SD.begin(pinCS))  
{  
  Serial.println("načtení microSD karty - neúspěšné");  
  Serial.println("//-----//");  
  while (1);  
}  
Serial.println("načtení microSD karty - úspěšné");  
Serial.println("//-----//");  
  
zaznam = SD.open("data.txt", FILE_WRITE);
```

Obr. 56. IDE – kontrola microSD karty.

4.2.5 Načtení hodnot z Hallova senzoru proudu

Následující část se zabývá měřením odebíraného proudu na vodiči a vnitřního napětí na Hallově senzoru. Jedná se především o načítání hodnot z analogového pinu do proměnné *hodnota*. Data, která byla získána se následně přepočtou na napětí v milivoltech. Pro přepočet je použita funkce *map* s předem stanovenými limitními hodnotami, následně je odečtena kalibrační konstanta, kterou bylo nutné na začátku programu nadefinovat. Na základě

dostupných informací o Hallově senzoru proudu posloužil analogový rozsah: 0 až 1023 s napájecím rozsahem 0 až 5000 mV.

Pro přepočítání na Ampéry posloužil datový typ float, který dovede pracovat s desetinnými čísly. Pro zjištění skutečné hodnoty odebíraného proudu vodičem, bylo nutné vypočtenou hodnotu vnitřního napětí na Hallově senzoru podělit rozlišením uváděným výrobcem. Výrobce u senzoru WCS170 uvádí hodnotu 32 mV na 1 A.

```
int hodnota = analogRead(AgPin);

int napeti = map(hodnota, 0, 1023, 0, 5000) - konst_kalibrace;

float proud = napeti / 32.0;
```

Obr. 57. IDE – výpočet napětí a odebíraného proudu.

4.2.6 Detekování pohybu pomocí IR senzoru

Ověření probíhá prostřednictvím funkce IF, která ověřuje, zda se na pinu, který odpovídá IR senzoru nachází logická hodnota LOW nebo HIGH. Pokud se na pinu nachází logická hodnota LOW, tak nastane sepnutí relé, rozsvícení modré LED diody a nastavení boolean stav_1 na „true“.

```
if (digitalRead(PIN_IR) == LOW)
{
    digitalWrite(PIN_RELE, HIGH);
    digitalWrite(PIN_LED_M, HIGH);
    stav_1 = true;
}
```

Obr. 58. IDE – detekování pohybu.

4.2.7 Ukládání data a času na microSD kartu

Jakmile je detekován pohyb pomocí IR senzoru, tak se pomocí RTC modulu reálného času zaznamená aktuální datum a čas detekování pohybu. Tento údaj je následně uložen na microSD kartu a souběžně zobrazen na sériovém monitoru.

```
zaznam.print("Čas detekování pohybu: ");
zaznam.print(DatCas.hour); zaznam.print(":");
zaznam.print(DatCas.minute); zaznam.print(":");
zaznam.print(DatCas.second); zaznam.println(" ");
zaznam.print("datum detekování pohybu: ");
zaznam.print(DatCas.day); zaznam.print(".");
zaznam.print(DatCas.month); zaznam.print(".");
zaznam.print(DatCas.year); zaznam.println("");
```

Obr. 59. IDE – uložení data a času.

4.2.8 Ukládání naměřených dat pomocí Hallova senzoru na microSD kartu

Před každým záznamem na microSD kartu je potřeba uvést příkaz pro otevření karty. Pokud by záznam na úložiště probíhal pouze v jedné části kódu, stačí příkaz uvést pouze jednou, ale obslužný program pro diagnostickou sestavu touto možností nedisponuje. Záznam probíhá v několika částech kódu, proto je nutné vždy před každým záznamem uvést právě tento příkaz.

```
zaznam = SD.open("data.txt", FILE_WRITE);
```

Obr. 60. IDE – otevření microSD karty.

Na posledním řádku se nachází příkaz pro uzavření microSD karty. Tento příkaz je nutné použít vždy, když záznam na microSD kartu je na nějakou dobu ukončen a bude probíhat třeba jiný záznam nebo jiná část kódu. Na pozici proměnné *napětí* je uložena skutečná hodnota naměřeného vnitřního napětí na Hallově senzoru, stejný postup je zopakován i u proměnné *proud*, pro uložení odebíraného proudu.

```
zaznam.print("Naměřené napětí: ");
zaznam.print(napeti);
zaznam.print(" mV , Odebíraný proud: ");
zaznam.print(proud);
zaznam.println(" A");
zaznam.println("//-----//");
zaznam.close();
```

Obr. 61. IDE – uložení naměřených hodnot.

4.2.9 Kontrola přiložené RFID karty

Jedná se celkem o dvě funkce IF. První funkce zajišťuje nalezení RFID karty, která je přiložena ke čtečce. Pokud je podmínka pravdivá, tak následuje druhá funkce pro přečtení identifikačního čísla z karty. Proměnná *tag* v sobě obsahuje jedinečné ID karty přiložené na

čtečku. V dalším kroku probíhá ověření, jestliže přiložená karta splňuje předepsaný formát. Formát jedinečného ID každé karty musí být hexadecimální a zároveň složen z osmi znaků, které jsou rozděleny po dvojicích. Jedinečné ID karty je v podstatě posloupnost znaků 0-9 a A-F, proto se jedná o číslo v hexadecimálním tvaru s délkou nejméně 24 bitů a nejvíc 128 bitů. Nicméně u různých typů karet může docházet k rozdílným délkám ID.

```
// slouží pro hledání RFID karty, která bude sloužit pro vypnutí LED diody
if ( ! mfrc522.PICC_IsNewCardPresent() )
{
    return;
}

// slouží pro přečtení nalezené RFID karty
if ( ! mfrc522.PICC_ReadCardSerial() )
{
    return;
}

//zde probíhá čtení z RFID karty
String tag = "";

for (byte k = 0; k < mfrc522.uid.size; k++)
{
    tag.concat(String(mfrc522.uid.uidByte[k] < 0x10 ? " 0" : " "));
    tag.concat(String(mfrc522.uid.uidByte[k], HEX));
}
tag.toUpperCase();
```

Obr. 62. IDE – přiložení RFID karty.

Jakmile jsou všechny znaky jedinečného ID karty převedeny na velká písmena, tak se porovnává, zda se ID přiložené karty shoduje s ID karty uložené v databázi. Pokud RFID karta odpovídá, tak zhasne modrá LED dioda a na 2000 ms se rozsvítí zelená LED dioda a IP kamera se vypne. Jestliže RFID karta neodpovídá, tak se na 2000 ms rozsvítí červená LED dioda, modrá LED dioda zůstává stále svítit a IP kamera je stále aktivní.

```
if (tag.substring(1) == (karta_UI))
```

Obr. 63. IDE – porovnání RFID karty.

4.2.10 Přiložení správné RFID karty

Nastavení logické hodnoty LOW na pinu relé a na modré LED diodě, tím je zajištěno vypnutí IP kamery a zhasnutí modré LED diody.

```
digitalWrite (PIN_RELE, LOW);  
digitalWrite (PIN_LED_M, LOW);
```

Obr. 64. IDE – vypnutí IP kamery a zhasnutí modré LED diody.

Zde probíhá výpis na LCD displej a rozsvícení zelené LED diody na 2000 ms.

```
lcd.clear();  
lcd.setCursor(0, 0);  
lcd.print("SPRAVNA KARTA");  
lcd.setCursor(0, 1);  
lcd.print("IP KAMERA OFF");  
digitalWrite (PIN_LED_Z, HIGH);
```

Obr. 65. IDE – výpis na LCD displej a rozsvícení zelené LED diody.

V následujícím kódu probíhá otevření microSD karty a uložení informací o přiložené kartě a vypnutí IP kamery.

```
zaznam = SD.open("data.txt", FILE_WRITE);  
zaznam.println("SPRAVNA KARTA");  
zaznam.println("IP KAMERA OFF");  
zaznam.println("//-----//");  
zaznam.close();
```

Obr. 66. IDE – uložení informací na microSD kartu.

Po uplynutí časové prodlevy nastane zhasnutí zelené LED diody. Jakmile zhasne zelená LED dioda, tak nastane vypnutí IP kamery a boolean *stav_1* se nastaví na hodnotu „false“

```
delay (PRODLEVA_2);  
digitalWrite (PIN_LED_Z, LOW);  
lcd.clear();  
stav_1 = false;
```

Obr. 67. IDE – zhasnutí zelené LED diody.

4.2.11 Přiložení nesprávné RFID karty

Následující řádky popisují zdrojový kód reagující na nesprávnou kartu. Kód uvedený na Obr. 68 ilustruje výpis na LCD displej a rozsvícení červené LED diody na časovou prodlevu 2000 ms.

```
lcd.clear();  
lcd.setCursor(0, 0);  
lcd.print("NESPRAVNA KARTA");  
lcd.setCursor(0, 1);  
lcd.print("IP KAMERA ON");  
digitalWrite(PIN_LED_C, HIGH);
```

Obr. 68. IDE – výpis na LCD displej a rozsvícení červené LED diody.

Při přiložení nesprávné RFID karty se provede uložení informací o činnosti společně s ID nesprávné RFID karty na microSD kartu.

```
zaznam = SD.open("data.txt", FILE_WRITE);  
zaznam.println("NESPRAVNA KARTA");  
zaznam.print("ID NESPRAVNE KARTY: ");  
zaznam.println(tag.substring(1));  
zaznam.println("IP KAMERA STALE ON");  
zaznam.println("//-----//");  
zaznam.close();
```

Obr. 69. IDE – uložení informací na microSD kartu.

Jakmile uplyne časová prodleva 2000 ms, tak nastane zhasnutí červené LED diody, která signalizovala přiložení nesprávné RFID karty.

```
delay(PRODLEVA_2);  
digitalWrite(PIN_LED_C, LOW);  
lcd.clear();  
stav_1 = true;
```

Obr. 70. IDE – zhasnutí červené LED diody.

4.2.12 IR senzor nedetekuje pohyb

Pokud není detekován pohyb pomocí IR senzoru, tak boolean *stav_1* zůstává nastaven na logickou hodnotu „false“. Proběhne výpis na LCD displej, sériový monitor a nastane uložení naměřených hodnot vnitřního napětí na Hallově senzoru a proudu na vodiči při vypnuté IP kameře.

```
lcd.clear();  
lcd.setCursor(0, 0);  
lcd.print("SECURITY SYSTEM");  
lcd.setCursor(0, 1);  
lcd.print("IP KAMERA OFF");  
  
stav_1 = false;  
  
Serial.println("IP KAMERA OFF");  
Serial.println("NENI POHYB");
```

Obr. 71. IDE – není detekován pohyb.

4.3 Popis zdrojového kódu WPF aplikace v C#

Při vytváření WPF aplikace na základě publikací [31], [32], [33] probíhá vytváření kódu ve dvou samostatných oknech. Jedná se o: *MainWindows.xaml* a *MainWindows.xaml.cs*.

MainWindows.xaml je použita pro vytvoření Graphical User Interface (dále jen GUI). Jedná se o umístění tlačítek, textových polí a pole pro zobrazení záznamu z IP kamery. Lze tedy říct, že tato část se zabývá pouze vizuální stránkou WPF aplikace.

MainWindows.xaml.cs se používá pro definování funkcí jednotlivých tlačítek. Pro každé tlačítko se musí vytvořit funkce, která bude vykonána při stisknutí tlačítka.

Prvním krokem je definování vhodných knihoven, v tomto případě těch pro připojení a správu kamer. Následně je nutné vytvořit proměnné a přiřadit proměnné k předdefinovaným funkcím. Poté lze vytvářet samostatné funkce pro jednotlivé tlačítka.

4.3.1 Návrh WPF aplikace

Celkem se ve WPF aplikaci nachází sedm tlačítek pro ovládání. Tři tlačítka souvisí s připojením a odpojením kamery. Tlačítko pro připojení IP kamery, tlačítko pro připojení WEB kamery, tlačítko pro odpojení IP kamery nebo WEB kamery. Další dvě tlačítka ovládají zapnutí a vypnutí Motion Detection. Ukládání snímku je zajištěno přes tlačítko *Snímek* a pro výběr konkrétní cesty je využito tlačítko *Výběr místa uložení*.

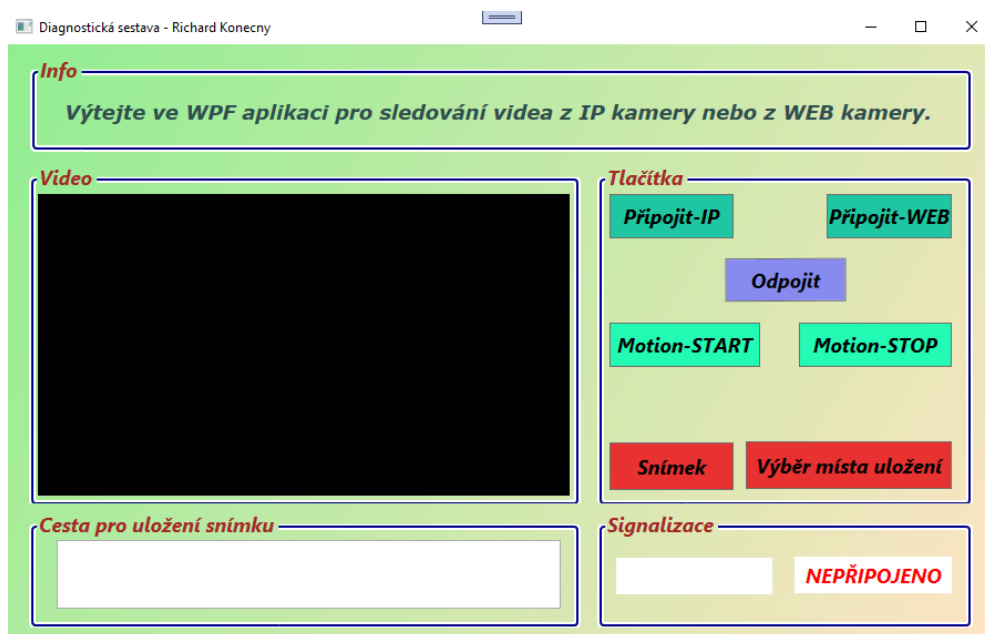
Signalizace připojení nebo odpojení kamery jsou provedeny pomocí zobrazení nápisu „připojeno“ nebo „odpojeno“. Tyto nápisy jsou přednastaveny v GUI, kde probíhá pouze změna barvy textu tak, aby byl text viditelný. Pokud je IP kamera připojena, tak se zobrazí zelený nápis „připojeno“, jestliže IP kamera není připojena, tak se zobrazí červený nápis „nepřipojeno“. Pro výběr konkrétní cesty pro uložení snímku slouží textBox s názvem *textBoxulozit*,

jedná se o bílý obdélník. Jakmile uživatel klikne na tlačítko *Výběr místa uložení* a provede výběr cesty, pak se vybraná cesta zobrazí v textBoxu, následně je možné cestu i ručně upravit.

Pro zobrazení videa je využit *VideoViewer*, který je vhodný k zobrazení videa z IP kamery při použití knihoven *Ozeki*. Často se při vytváření aplikace lze setkat s *ImageViewer*, který je využit taktéž pro sledování videa, např. při použití knihovny *Emgu*.

4.3.2 WPF aplikace pro sledování videa

Postupně zde budou vysvětleny funkce jednotlivých tlačítek a dalších polí. Snímané video se bude zobrazovat ve *VideoVieweru*, jedná se o černý obdélník na Obr. 72.



Obr. 72. WPF aplikace pro sledování videa z IP kamery.

4.3.3 Vytváření GUI

Při vytváření GUI byly definovány samostatné skupiny pro tlačítka, signalizaci, video atd. Pro vytvoření skupin byl použit příkaz *GroupBox*, příkaz zajišťuje ohraničení např. tlačítek nebo signalizace a zároveň popisek každé samostatné skupiny. Tím je zvýšena přehlednost jak pro uživatele, který aplikaci ovládá, tak i pro programátora, který by v kódu prováděl další úpravy. Také se nabízí možnost tloušťky ohraničení, umístění *GroupBoxu* a další parametry.

```
<GroupBox Header="Video" Height="212" Width="292" VerticalAlignment="Top"
HorizontalAlignment="Center" BorderThickness="2" BorderBrush="■"DarkBlue"
Foreground="■"Brown" Margin="20,138,288,0" Grid.RowSpan="3" Grid.ColumnSpan="2"
FontWeight="Bold" FontStyle="Italic" FontSize="18">
<control:VideoViewerWPF Name="videoViewer" Height="170" Width="275"
HorizontalAlignment="Stretch" VerticalAlignment="Stretch" Background="■"Black"/>
</GroupBox>
```

Obr. 73. C# – definování videoVieweru pro zobrazení videa.

Následující část se zabývá vytvořením tlačítek, jejich umístěním a vytvoření funkce *Click* pro každé tlačítko. Tato funkce je zároveň vytvořena v okně *MainWindows.xaml.cs*, zpočátku se jedná pouze o tělo funkce, které musí uživatel správně doplnit, tak aby byla zajištěna správná funkce tlačítka. Po kliknutí na tlačítko *Připojit-IP* ve WPF aplikaci, dochází k aktivaci funkce *Click* a provedení události, kterou tato funkce spouští. Tato funkce musí být u každého tlačítka, aby byla zajištěna kontinuita.

```
<Button Name="pripojit_IP" Content="Připojit-IP" Click="pripojit_IP_Click" Margin="10,3,37,-27"
RenderTransformOrigin="0.5,0.5" FontWeight="Bold" Background="■"#FF1EC7A1" Grid.ColumnSpan="2"/>
```

Obr. 74. C# – definování tlačítek v GUI.

4.3.4 Importování knihoven

Importování knihoven probíhá pouze v *MainWindows.xaml.cs*, tyto knihovny je potřeba nadefinovat před samotným začátkem vytváření zdrojového kódu.

```
using Ozeki.Media;
using Ozeki.Camera;
using System.Windows.Forms;
using Image = System.Drawing.Image;
using Ozeki.Vision;
```

Obr. 75. C# – importování knihoven.

4.3.5 Vytvoření proměnných

Základem každého programování je vytvoření proměnných, které se budou postupně v celém zdrojovém kódu používat.

```
private IIPCamera IP_kamera; //proměnná pro IP kameru
private WebCamera WEB_kamera; //proměnná pro WEB kameru
private MediaConnector P_propojeni_web; //proměnná, která zajišťuje připojení WEB kamery
private MediaConnector P_propojeni; //proměnná, která zajišťuje připojení IP kamery
private DrawingImageProvider P_supplier; //zajišťuje vizualizaci zpracovaného obrazu
private FolderBrowserDialog P_cesta; //zajišťuje ukládání obrázku
private MotionDetector P_detekce; //proměnná pro MotionDetection
private System.Drawing.Image image; //pořízení snímku obrazovky
private SnapshotHandler P_snap; //vytvoření obrazu
```

Obr. 76. C# – vytvoření proměnných.

4.3.6 Přiřazení proměnných k předdefinovaným funkcím

Jakmile jsou vytvořeny proměnné, tak je provedeno přiřazení přednastavených funkcí k novým proměnným. Funkce nejčastěji bývají součástí knihoven, které je potřeba nejdříve definovat.

```

WEB_kamera = new WebCamera();           //konfigurace WEB kamery
P_propojeni_web = new MediaConnector();  //konfigurace připojení pro WEB kameru
P_supplier = new DrawingImageProvider(); //vizualizace obrazu
IP_kamera = IPCameraFactory.GetCamera("rtsp://10.53.25.201:554/axis-media/media.amp", "root", "root");
//definování IP adresy pro kameru, jméno a heslo
P_propojeni = new MediaConnector();      //konfigurace pro IP kameru
P_cesta = new FolderBrowserDialog();     //slouží k výběru cesty pro uložení obrázku
P_detekce = new MotionDetector();        //slouží pro detekování pohybu
P_snap = new SnapshotHandler();          //zpracování obrazu

```

Obr. 77. C# – přiřazení proměnných k předdefinovaným funkcím.

4.3.7 Tlačítko pro připojení IP kamery

Na začátku každé funkce je potřeba provést propojení mezi IP kamerou a jednotlivými proměnnými s přiřazenými funkcemi. Poslední řádek definuje propojení se zobrazovacím polem. Rozdíl nastává u funkce pro odpojení IP kamery, kde se místo klíčového slova *Connect* používá klíčové slovo *Disconnenct*.

```

P_propojeni.Connect(IP_kamera.VideoChannel, P_detekce);
P_propojeni.Connect(P_detekce, P_supplier);
P_propojeni.Connect(IP_kamera.VideoChannel, P_snap);
videoViewer.SetImageProvider(P_supplier);...

```

Obr. 78. C# – propojení IP kamery s jednotlivými proměnnými.

Následující příkazy jsou použity pro zapnutí IP kamery a zapnutí videoVieweru. U funkce pro připojení se požaduje připojení IP kamery, pro odpojení kamery místo slova *Start* je použito slovo *Stop*.

```

IP_kamera.Start();
videoViewer.Start();

```

Obr. 79. C# – zapnutí IP kamery a videoVieweru.

Následující sada příkazů je využita jako efekt zmáčknutého tlačítka. Stisknutím tlačítka se deaktivuje událost opětovného stisknutí, která trvá, dokud není spuštěna jiná operace s tímto efektem spojená. Obr. 80 ilustruje deaktivaci tlačítka pro připojení IP kamery. Proto se doporučuje u každého tlačítka zároveň nastavit efekt vypnutého tlačítka.

Stejná sada příkazů se nachází i u tlačítka pro připojení a odpojení WEB kamery, s tím rozdílem, že logická hodnota „false“ se nachází právě u toho tlačítka, kterému odpovídá funkce, v níž jsou příkazy uvedeny.

```
odpojit.IsEnabled = true;  
pripojit_WEB.IsEnabled = true;  
pripojit_IP.IsEnabled = false;
```

Obr. 80. C# – efekt zmáčknutého tlačítka.

4.3.8 Tlačítko pro zapnutí Motion Detection

Po aktivování tlačítka pro Motion Detection jsou červeně zvýrazněny obrysy pohyblivých objektů ve snímané scéně. Nejčastěji se může jednat třeba o obličeje.

```
P_detekce.HighlightMotion = HighlightMotion.Highlight;  
P_detekce.MotionColor = MotionColor.Red;
```

Obr. 81. C# – barevné zvýraznění pohybu v obraze.

Pro Motion Detection byla vytvořena samostatná událost, která se ve funkci pro připojení IP kamery přidává a aktivuje. Ve funkci pro vypnutí Motion Detection je tato událost odebrána a deaktivuje se.

```
P_detekce.MotionDetection += MDetection;  
P_detekce.Start();
```

Obr. 82. C# – přidání události pro Motion Detection.

4.3.9 Vytvoření události pro Motion Detection

Tato událost se aktivuje při stisknutí tlačítka připojit. Jakmile Motion Detection rozpozná pohyb objektu, tak je vytvořen snímek videa. Problematika funkce Motion Detection je detailně rozebrána v publikaci [34]. Rozpoznání probíhá při změně pixelů neboli obrazových bodů v monitorované scéně, jinými slovy řečeno, když nastane odchýlení sledovaného předmětu z počátečního bodu. Dříve se objevovaly falešné poplachy způsobené pohybujícími se stromy, nicméně postupem času byly objeveny a aplikovány neuronové sítě, které učí a zdokonalují hodnotící algoritmus a chyby identifikují. Jako opatření proti těmto falešným poplachům přišla např. společnost VIVOTEK s novou generací detekce pohybu, označovaného jako Smart Motion Detection neboli inteligentní detekce pohybu. Nové detekční systémy pracují na základě neuronových sítí pro rozpoznávání lidských tvarů. S využitím databází

siluet a dovedností rychlého strojového učení, dovede algoritmus detekce pohybu rozpoznat postavu člověka v rychlém časovém intervalu. Sledování pohybu lidí se řadí mezi dnes spíše už základní požadavky u aplikací bezpečnostních kamerových systémů. Detekce pohybu osob je prováděna jen na základě událostí, tedy pohybu, resp. změny hodnot pixelů. Naopak křoviny, automobily, zvířata nebo jiné předměty nacházející se ve snímané scéně nepatří mezi vyhodnocované události. Tím nastane snížení falešných poplachů. Inteligentní detekce pohybu dokáže výborně pracovat i za zhoršených světelných podmínek.

Jako název vytvořeného snímku bude aktuální datum a čas. Následně proběhne spojení data a času s příponou .jpg, tak aby mohl být jednoduše snímek zobrazen. Vytvoření a uložení snímku zajišťuje funkce *snapshotImage*. Nejdříve proběhne vytvoření snímku a následně je snímek uložen do složky bin v daném projektu.

```
Image snapshotImage = (Image)P_snap.TakeSnapshot().ToImage();
snapshotImage.Save(N_cesta, System.Drawing.Imaging.ImageFormat.Jpeg);
```

Obr. 83. C# – vytvoření a uložení snímku.

4.3.10 Uložení snímku pomocí tlačítka Snímek

Při stisknutí tlačítka *Snímek* vytvoříme snímek aktuálního videa. Tento snímek se ukládá do složky bin v daném projektu. Uložení snímku na jiné místo je zajištěno přes tlačítko *Výběr místa uložení*, snímek tak lze uložit do požadované složky na disku. Cesta pro uložení se poté zobrazí v definovaném poli – *Cesta pro uložení snímku*. Jakmile proběhne vybrání cesty a aktivování tlačítka *Snímek*, tak je snímek uložen do požadované složky. Na Obr. 84 je uvedena varianta pro výběr cesty pro uložení snímku pomocí tlačítka *Výběr místa uložení*. Funkce *FolderBrowserDialog* je funkce, která se primárně používá k vyzvání uživatele, aby vybral cestu (koncovou složku) pro ukládání souboru. Pro tuto funkci je vytvořena proměnná *P_cesta*, se kterou se dále pracuje. Absolutní cesta se následně zobrazí v textboxu s názvem *textBoxulozit*. V podstatě se jedná o třídu, jež nabízí uživateli výběr složky, tato třída je využita pouze pro výběr složky, není využita pro výběr souborů.

```
var vysl_cesta = P_cesta.ShowDialog();
if (vysl_cesta == System.Windows.Forms.DialogResult.OK) ;
textBoxulozit.Text = P_cesta.SelectedPath;
```

Obr. 84. C# – vybrání cesty pro uložení snímku.

Pro uložení snímku z IP kamery byl vytvořen podprogram s názvem *CreateSnapshot*.

```
var cesta = textBoxulozit.Text;
CreateSnapshot(cesta);
```

Obr. 85. C# – uložení snímku z IP kamery.

Obr. 86 popisuje vytvoření názvu snímku, který bude obsahovat aktuální datum a čas vytvoření snímku, jestliže se v textovém poli nenachází žádná cesta, tak je snímek uložen do složky bin. Pokud uživatel vybere cestu pro uložení, která se zobrazí v textovém poli (Cesta pro uložení snímku), tak je snímek uložen na konkrétní místo. Základ funkce je obdobný, jako u vytvoření a uložení snímku při Motion Detection, ale navíc se zde nachází možnost uložení snímku na vybrané místo. Nejdříve je potřeba vytvořit aktuální datum a čas, se kterým se dále pracuje v proměnné *DatumCas*.

```
var DatumCas = "Čas " + DateTime.Now.Hour + "-" + DateTime.Now.Minute + "-" +
+ DateTime.Now.Second + " Datum " + DateTime.Now.Day + "-" + DateTime.Now.Month
+ "-" + DateTime.Now.Year;
```

Obr. 86. C# – vytvoření aktuálního data a času.

Obr. 87 popisuje možnosti zjištění, zda se v textovém poli nachází cesta pro uložení snímku. Pokud je podmínka splněna, tak cesta zadaná není a snímek je uložen do složky bin > Debug. Tato složka se nachází ve složce vytvořeného projektu. V opačném případě proběhne uložení snímku na definované místo. Poslední dva řádky zajišťují vytvoření a uložení snímku.

```
string N_cesta;
if (String.IsNullOrEmpty(cesta))
{
    N_cesta = DatumCas + ".jpg";
}
else
{
    N_cesta = cesta + "\\ " + DatumCas + ".jpg";
}
Image snapShotImage = (Image)P_snap.TakeSnapshot().ToImage();
snapShotImage.Save(N_cesta, System.Drawing.Imaging.ImageFormat.Jpeg);
```

Obr. 87. C# – uložení snímku.

5 OVĚŘENÍ FUNKČNOSTI

Následující kapitola je věnována ověření funkčnosti navržené diagnostické sestavy. Postupně jsou demonstrovány stavy, které se očekávají při provozu sestavy. Tyto údaje jsou ilustrovány formou výpisů z LCD displeje a signalizace pomocí LED diod. Každá ukázka popsaná v této kapitole je doplněna výstupem ze souboru *data.txt* vytvořeného na microSD kartě. Tento soubor obsahuje informace získané během provozu diagnostické sestavy. Tedy údaje o odebíraném proudu, vnitřním napětí na Hallově senzoru, informace o činnosti IP kamery, datum a čas, při kterém byl detekován pohyb pomocí IR senzoru a další informace. Informace jsou souběžně s ukládáním na kartu vypisovány na sériový monitor, tím je zajištěna aktuální kontrola dat v definovaném čase.

5.1 Kontrola microSD karty

Před uvedením diagnostické sestavy do provozu proběhne kontrola microSD karty, za účelem identifikace jejího stavu. Výsledek procesu je vypsán do sériového monitoru. Při zjištění nesprávného načtení karty proběhne výpis, který ilustruje Obr. 88 a provoz diagnostické sestavy je pozastaven, dokud neproběhne další kontrola.

```
//-----//  
probíhá načítání microSD karty  
načtení microSD karty - neúspěšné  
//-----//
```

Obr. 88. Sériový monitor – neúspěšné načtení microSD karty.

Jakmile je provedena kontrola nebo vložení nové microSD karty, tak proběhne následující výpis na sériový monitor a diagnostická sestava je úspěšně uvedena do provozu.

```
//-----//  
probíhá načítání microSD karty  
načtení microSD karty - úspěšné  
//-----//
```

Obr. 89. Sériový monitor – úspěšné načtení microSD karty.

5.2 Nedetekován pohyb

Pokud není detekován pohyb pomocí IR senzoru, pak je IP kamera vypnuta a žádná LED dioda není aktivní. Pouze Hallův senzor je aktivní nezávisle na činnosti IP kamery, měření odebíraného proudu na vodiči a vnitřního napětí na Hallově senzoru tedy probíhá, jak při

vypnuté IP kameře, tak i při zapnuté IP kameře. Následně probíhá výpis naměřených hodnot na sériový monitor a zároveň jsou hodnoty ukládány na microSD kartu.



Obr. 90. LCD displej – IP kamera OFF.

První řádek informuje uživatele o vypnutí IP kamery. Druhý řádek podává informaci, zda IR senzor detekuje pohyb. Na třetím řádku se nachází informace o naměřeném vnitřním napětí na Hallově senzoru a odebíraném proudu na vodiči, který prochází skrz Hallův senzor. Hodnota napětí a proudu byla v uvedeném případě nulová.

```
//-----//  
IP KAMERA OFF  
NENI POHYB  
Naměřené napětí: 0 mV , Odebíraný proud: 0.00 A  
//-----//
```

Obr. 91. Záznam z microSD karty – není pohyb.

5.3 Detekován pohyb, správná RFID karta

Jakmile proběhne detekce pohybu pomocí IR senzoru, tak nastane rozsvícení modré LED diody a IP kamera je uvedena do režimu zapnuto.



Obr. 92. LCD displej – IP kamera ON.

První řádek informuje obsluhu o detekování pohybu pomocí IR senzoru. Z druhého řádku je patrný provoz IP kamery. Na třetím a čtvrtém řádku se nachází čas a poté datum, kdy byl detekován pohyb pomocí IR senzoru. Na posledním řádku jsou informace o naměřeném vnitřním napětí na Hallově senzoru a odebíraném proudu na vodiči.

```
//-----//  
DETEKOVAN POHYB  
IP KAMERA ON  
čas detekování pohybu: 8:26:27  
datum detekování pohybu: 22.2.2022  
Naměřené napětí: -19 mV , Odebíraný proud: -0.59 A  
//-----//
```

Obr. 93. Záznam z microSD karty – detekován pohyb.

Dále následuje přiložení správné RFID karty. Po přiložení karty se na 2000 ms rozsvítila zelená LED dioda, následně zhasla modrá LED dioda a IP kamera se vypnula. Během celého cyklu probíhalo vypisování hlášek na LCD displej, sériový monitor a ukládání záznamu na microSD kartu. Výpis na sériovém monitoru byl totožný se záznamem uloženým na kartě.



Obr. 94. LCD displej – správná RFID karta.

Na Obr. 95 probíhá vypisování informace o přiložení správné RFID karty a hláška o vypnutí IP kamery.

```
//-----//  
SPRAVNA KARTA  
IP KAMERA OFF  
//-----//
```

Obr. 95. Záznam z microSD karty – správná RFID karta.

5.4 Detekován pohyb, nesprávná RFID karta

IR senzor detekoval pohyb, nastalo rozsvícení modré LED diody a zapnutí IP kamery. Následovalo přiložení nesprávné RFID karty. Po přiložení nesprávné karty proběhlo na 2000 ms rozsvícení červené LED diody, modrá LED dioda zůstává stále svítit a činnost IP kamery zůstává nezměněna. I zde po celou dobu probíhal výpis na LCD displej, sériový monitor a ukládání záznamů na kartu.



Obr. 96. LCD displej – nesprávná RFID karta.

Pokud je přiložena nesprávná RFID karta, tak proběhne záznam ID nesprávné karty, který je vypsán na sériový monitor a zároveň uložen na microSD kartu. Na Obr. 97 je uvedeno ID nesprávné karty, která je dodávána k diagnostické sestavě. Pokud je přiložena nesprávná karta, tak IP kamera zůstává stále v provozu, tomu odpovídá třetí řádek na uvedeném snímku.

```
//-----//  
NESPRAVNA KARTA  
ID NESPRAVNE KARTY: 96 B5 87 C9  
IP KAMERA STALE ON  
//-----//
```

Obr. 97. Záznam z microSD karty – nesprávná RFID karta.

5.5 Sledování videa přes WPF aplikaci

Pro sledování videa z IP kamery byla vytvořena WPF aplikace. Prostřednictvím WPF aplikace se lze vzdáleně přes síť připojit na IP kameru, která je připojena do switchu. WPF aplikace nabízí uživateli možnost vytvoření snímku obrazovky a následně jeho uložení do počítače. Na Obr.98 se nachází ukázka z prostředí WPF aplikace při připojení na IP kameru.



Obr. 98. WPF aplikace – detekován pohyb, připojení na IP kameru.

Dále je provedeno zaměření na obrazovou funkci Motion Detection, která je demonstrována na Obr. 99. Tato funkce je využita pro zvýraznění obrysu pohyblivých částí. Funkce Motion Detection je vyznačena červeným zvýrazněním hran objektů, které vykonávají pohyb. Na Obr. 99 si lze všimnout zešednutí tlačítka Připojit-IP a Motion-START, což znamená aktivování daných tlačítek ve WPF aplikaci. V ohraničení s názvem signalizace je zobrazen zelený nápis připojeno, který podává informaci o úspěšném připojení na IP kameru.



Obr. 99. WPF aplikace – aktivování funkce Motion Detection.

Další možností sledování videa z IP kamery, která je připojena přes switch do sítě lze po zadání IP adresy v prohlížeči. Jakmile je zadána správná IP adresa, tak se zobrazí okno se záznamem z IP kamery, patřičnou nevýhodou tohoto sledování je absence některých funkcí. Nicméně názornější je vytvořit si vlastní WPF aplikaci s dalšími funkcemi pro ovládání IP kamery.

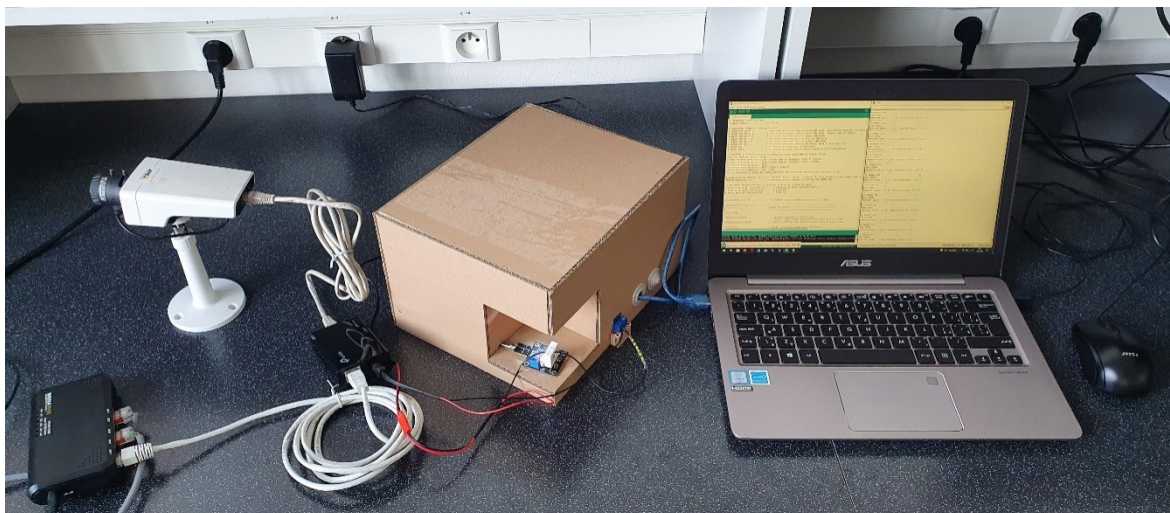
5.6 Zhodnocení měření

Měření Halloovým senzorem bylo ovlivněno řadou faktorů, které nebylo snadné úplně eliminovat. Jednalo se např. o různorodý průřez vodičů, připájených ke kabeláži napájecího adaptéru. Tento problém vznikl, z důvodu, že vodiče, jež byly součástí napájecího adaptéru se neshodovaly s průřezem vodičů, které byly součástí JTS konektorů. Dalším kritériem byl požadavek na kvalitní spoj v připojovací části relé, které zajišťuje zapínání IP kamery. Kde bylo potřeba využít vodič o větším průřezu, než byl přívodní vodič od adaptéru.

Ztráty byly způsobeny i modifikacemi napájecí kabeláže PoE, která se musela uzpůsobit pro účely výzkumu. Další ztráty mohly být způsobeny letovanými přechody mezi vodiči.

Hallův senzor je využíván pro měření proudu o velikosti až 70 A, při měření se očekávají hodnoty proudu do cca 1 A, případně o něco vyšší. Nízké hodnoty proudu zapříčiní, že při měření se nevyužívá celý měřicí rozsah, ale pouze jeho část. Např. při proudu do 1 A se využívá 1/70 měřicího rozsahu. Podmínkou měření je protažení pouze jednoho vodiče Halloovým senzorem, nikoliv celý přívodní kabel. Pokud nastane protažení obou vodičů přívodního kabelu, tak by naměřené hodnoty byly nepřesné.

Při porovnání většího množství výsledků získaných při měření pouze na jednom přívodním vodiči do PoE, byly získány celkem podobné hodnoty. Během měření se objevily i hodnoty, které byly mimo rozsah adaptérů, ale jelikož měření bylo prováděno vícekrát, bylo možné na tyto hodnoty nebrat velký zřetel. Dále zde budou uvedeny fotografie z měření a screeny uložených záznamů. Znaménko mínus určuje směr proudu. Samotné měření na přívodním vodiči od PoE adaptéru je uvedeno na Obr. 100 až Obr. 102.



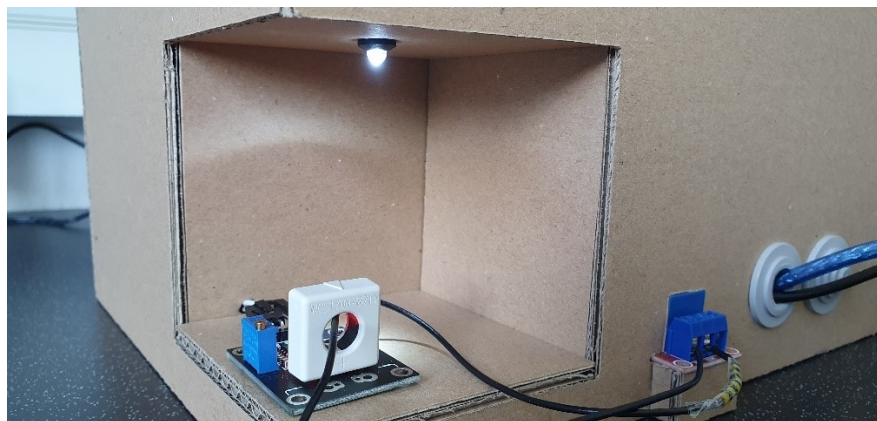
Obr. 100. Měření Hallovým senzorem, IP kamera Axis.

Z Obr. 100 je patrné, že diagnostická sestava je připojena pomocí USB kabelu do notebooku, USB kabel zároveň zajišťuje napájení diagnostické sestavy. V notebooku probíhá sledování aktuálního výpisu na sériový monitor. V levé spodní části se nachází switch, do kterého je připojen pomocí UTP kabelu PoE. Další UTP kabel je využit k propojení PoE a IP kamery. Měření Hallovým senzorem probíhá na přívodním kabelu od adaptéru, který přes relé pokračuje do PoE.

Diagnostickou sestavu lze dále napájet buď pomocí 9 V baterie, nebo adaptéru, jak je provedeno na Obr. 101 adaptér je přímo zapojen do desky Arduino Mega.



Obr. 101. Měření Hallovým senzorem, IP kamera Vivotek.



Obr. 102. Protážený vodič skrz Hallův senzor.

```
//-----//
DETEKOVAN POHYB
IP KAMERA ON
čas detekování pohybu: 8:26:32
datum detekování pohybu: 22.2.2022
Naměřené napětí: -15 mV , Odebíraný proud: -0.47 A
//-----//
```

Obr. 103. Vnitřní napětí na Hallově senzoru 15 mV, proud na vodiči 0.47 A.

5.7 Analyzování naměřených hodnot

Na základě naměřených hodnot Hallovým senzorem byl vytvořen dvojrozměrný bodový graf. Vytvořený graf popisuje průběh proudu po dobu zapnutí IP kamery. Z grafu je zřejmé, že dochází k mírným poklesům a nárůstům hodnot v celé šířce grafického průběhu, nicméně v celku je zachován konstantní průběh. Z naměřených hodnot byly vypočítány průměrné hodnoty vnitřního napětí na Hallově senzoru a proudu, který prochází vodičem skrz Hallův senzor. Výpočet byl proveden podle vzorce (5.3). Celkem byly sestrojeny dva grafy, jeden pro měření s IP kamerou Axis a druhý pro měření s IP kamerou Vivotek.

Dále byla vypočítána absolutní a relativní chyba pro měření proudu procházejícího vodičem. Výpočet absolutní chyby byl proveden jako rozdíl naměřené hodnoty a hodnoty skutečné. Pojem relativní chyba označuje podíl absolutní chyby a skutečné hodnoty proudu, následně je hodnota vyjádřena v procentech. Pro výpočet posloužily vzorce (5.1 a 5.2). Grafy a výpočty byly provedeny v prostředí Microsoft Excel.

Pro sestavení grafu u IP kamery Axis i IP kamery Vivotek bylo použito celkem 300 záznamů. Tyto záznamy poskytovaly hodnoty vnitřního napětí na Hallově senzoru a proudu procházející vodičem skrz Hallův senzor.

5.7.1 Měření s IP kamerou Axis

Tab. 7 prezentuje vybraných patnáct hodnot z celkové tabulky, kompletní tabulka je uvedena v elektronické příloze.

Tab. 7. Výpočet absolutní a relativní chyby – IP kamera Axis.

číslo měření	vnitřní napětí na Hallově senzoru [mV]	měřená hodnota proudu [A]	skutečná hodnota proudu [A]	absolutní chyba měření [-]	relativní chyba měření [%]
41.	19	0,59	0,5	0,09	18
42.	15	0,47	0,5	-0,03	-6
43.	15	0,47	0,5	-0,03	-6
44.	15	0,47	0,5	-0,03	-6
45.	15	0,47	0,5	-0,03	-6
46.	19	0,59	0,5	0,09	18
47.	19	0,59	0,5	0,09	18
48.	15	0,47	0,5	-0,03	-6
49.	15	0,47	0,5	-0,03	-6
50.	19	0,59	0,5	0,09	18
51.	15	0,47	0,5	-0,03	-6
52.	15	0,47	0,5	-0,03	-6
53.	19	0,59	0,5	0,09	18
54.	15	0,47	0,5	-0,03	-6
55.	19	0,59	0,5	0,09	18

Vzorec pro výpočet absolutní chyby:

$$\Delta = MH - SH \quad (5.1)$$

Vzorec pro výpočet relativní chyby:

$$\delta = \frac{\Delta}{SH} \cdot 100 \quad (5.2)$$

Δ – absolutní chyba [-]

MH – měřená hodnota proudu [A]

SH – skutečná hodnota proudu [A]

δ – relativní chyba [%]

Zápis naměřené hodnoty společně s vypočítanou absolutní chybou a relativní chybou je uveden v Tab. 8.

Tab. 8. Zápis hodnot pro první řádek tabulky.

zápis hodnot	
$I = (0,59 \pm 0,09) [A]$	$\delta = 18 \%$

V Tab. 9 jsou uvedeny průměrné hodnoty vnitřního napětí na Hallově senzoru a proudu procházejícího vodičem.

Tab. 9. Průměrné hodnoty napětí a proudu – IP kamera Axis.

průměrná hodnota napětí na Hallově senzoru [mV]	průměrná hodnota proudu [A]
14,882	0,466

Vzorec pro výpočet průměrné hodnoty napětí a proudu:

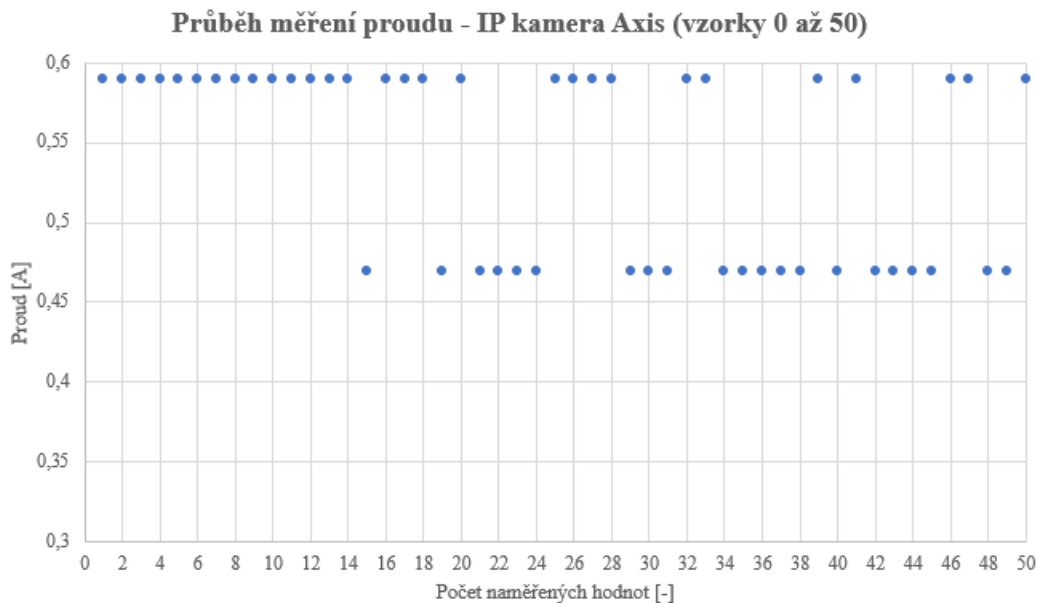
$$\bar{x} = \frac{x_1 + x_2 + \dots + x_n}{n} \quad (5.3)$$

\bar{x} – průměrná hodnota [A]; [mV]

$x_1 + x_2 + \dots + x_n$ – součet jednotlivých naměřených hodnot [A]; [mV]

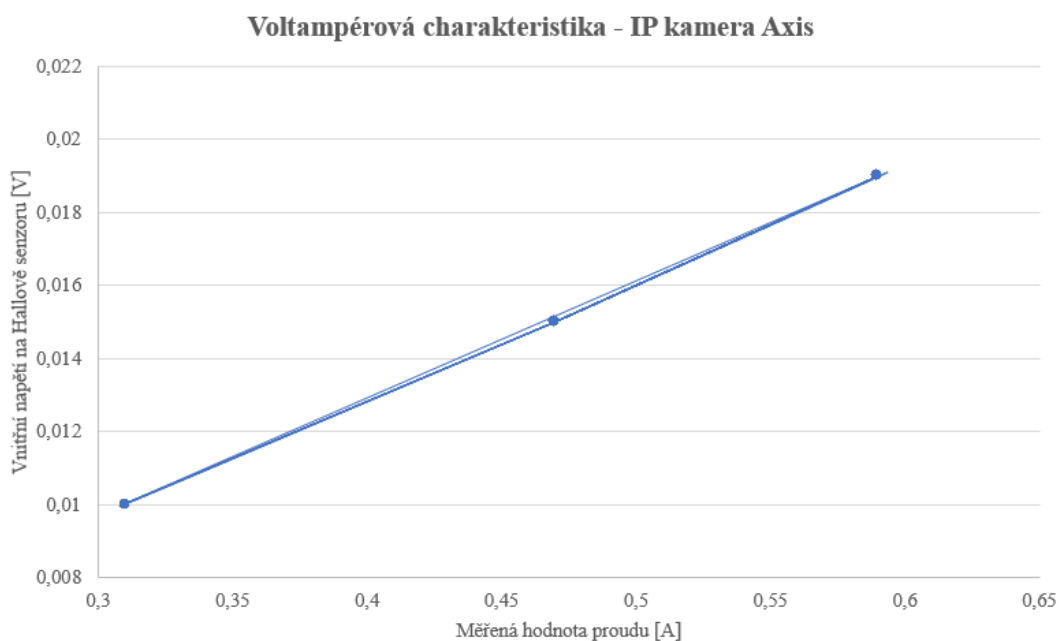
n – celkový počet měření [-]

Na Obr. 104 je vyobrazeno prvních padesát naměřených hodnot při připojení IP kamery Axis. Kompletní graf je uveden v příloze.



Obr. 104. Průběh měření proudu – IP kamera Axis (vzorky 0 až 50)

Závislost vnitřního napětí na Hallově senzoru a odebíraného proudu na IP kameře Axis popisuje Obr. 105. V podstatě se jedná o Voltampérovou charakteristiku, nicméně konstrukce uvedeného grafického průběhu není primárním výsledkem daného měření. Jedná se pouze o doplnění, které dává do souvislosti hodnoty naměřeného vnitřního napětí na Hallově senzoru a odebíraný proud na IP kameře Axis. Z grafického průběhu je patrné, že z celkového počtu 300 záznamů dochází k opakování celkem tří průsečíků po celou dobu pozorovaného měření.



Obr. 105. Voltampérová charakteristika – IP kamera Axis.

5.7.2 Měření s IP kamerou Vivotek

Tab. 10 prezentuje vybraných patnáct hodnot z celkové tabulky, kompletní tabulka je uvedena v elektronické příloze.

Tab. 10. Výpočet absolutní a relativní chyby – IP kamera Vivotek.

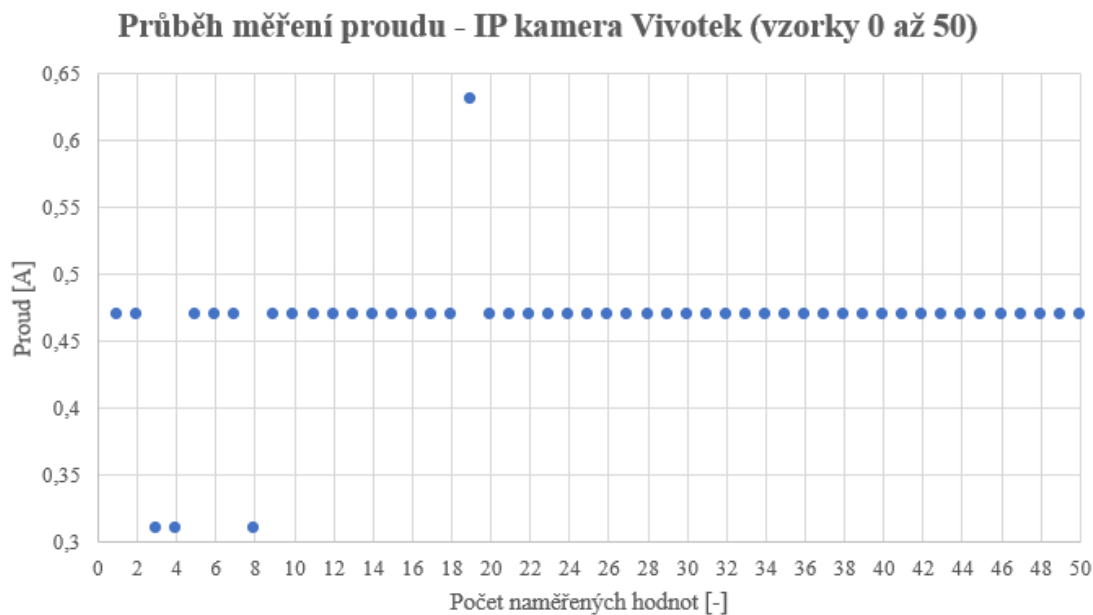
číslo měření	vnitřní napětí na Hallově senzoru [mV]	měřená hodnota proudu [A]	skutečná hodnota proudu [A]	absolutní chyba měření [-]	relativní chyba měření [%]
1.	15	0,47	0,5	-0,03	-6
2.	15	0,47	0,5	-0,03	-6
3.	10	0,31	0,5	-0,19	-38
4.	10	0,31	0,5	-0,19	-38
5.	15	0,47	0,5	-0,03	-6
6.	15	0,47	0,5	-0,03	-6
7.	15	0,47	0,5	-0,03	-6
8.	10	0,31	0,5	-0,19	-38
9.	15	0,47	0,5	-0,03	-6
10.	15	0,47	0,5	-0,03	-6
11.	15	0,47	0,5	-0,03	-6
12.	15	0,47	0,5	-0,03	-6
13.	15	0,47	0,5	-0,03	-6
14.	15	0,47	0,5	-0,03	-6
15.	15	0,47	0,5	-0,03	-6

Tab. 11 popisuje průměrné hodnoty vnitřního napětí na Hallově senzoru a proudu procházejícího vodičem, který je umístěn skrz Hallův senzor.

Tab. 11. Průměrné hodnoty napětí a proudu – IP kamera Vivotek.

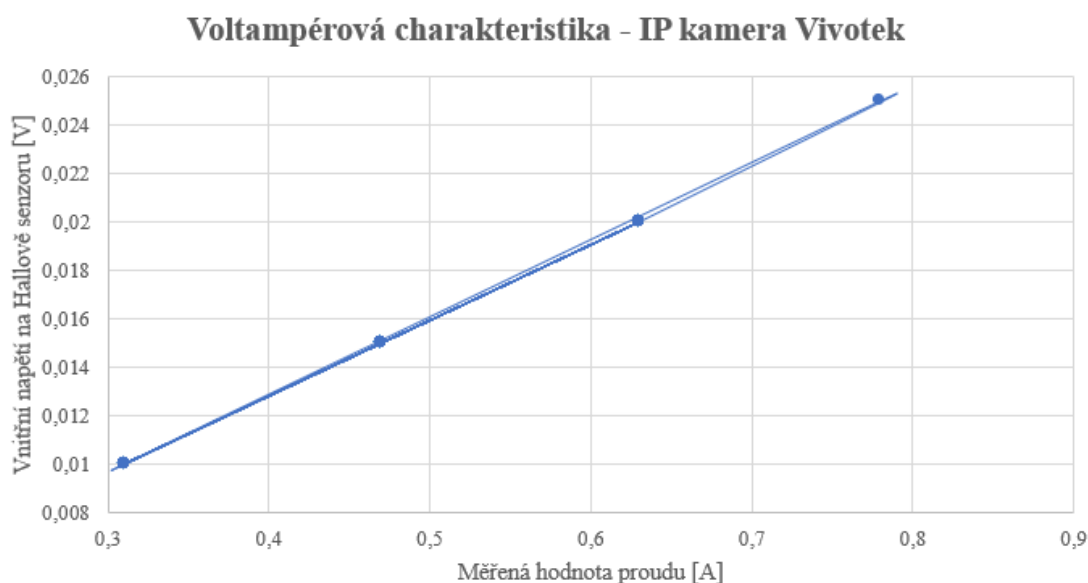
průměrná hodnota napětí na Hallově senzoru [mV]	průměrná hodnota proudu [A]
15,428	0,484

Na Obr. 106 je vyobrazeno prvních padesát naměřených hodnot při připojené IP kamere Vivotek. Kompletní graf je součástí elektronické přílohy.



Obr. 106. Průběh měření proudu – IP kamera Vivotek (vzorky 0 až 50).

Stejně jako u IP kamery Axis, i zde je na Obr. 107 vyobrazena grafická závislost vnitřního napětí na Hallově senzoru a odebíraného proudu na IP kameře Vivotek. Jak již bylo zmíněno výše, tak se jedná pouze o doplnění, jenž seskupuje hodnoty vnitřního napětí na Hallově senzoru a odebíraný proud na IP kameře Vivotek. Z grafického průběhu je patrné, že z celkového počtu 300 záznamů dochází k opakování celkem čtyř průsečíků, které jsou opakovány po dobu měření.



Obr. 107. Voltampérová charakteristika – IP kamera Vivotek.

ZÁVĚR

Cílem práce bylo navrhnout a realizovat diagnostickou sestavu pro IP kamery založenou na platformě Arduino. Nejdříve byla rozebrána problematika současného stavu a principů kamerových systémů společně s platformou Arduino. Poznatky a znalosti získané v teoretické části byly aplikovány v praktické části při návrhu a konstrukci diagnostické sestavy. Diagnostická sestava zajišťuje měření spotřeby IP kamery, která je aktivována při detekci pohybu zaznamenaného pomocí IR senzoru. Vypnutí IP kamery je provedeno s využitím RFID karty. Dále je popsána teorie kamerových systémů, jedná se především o rozdělení kamerových systémů, komunikace, konstrukce, integrace a další pojmy z oblasti kamerových systémů. Teoretická část se také v jisté míře věnuje platformě Arduino, která posloužila jako hlavní řídicí jednotka diagnostické sestavy.

Praktická část je zaměřena na popis jednotlivých komponent, které byly vybrány pro realizaci diagnostické sestavy. Obsahem praktické části je schéma zapojení a vytvořený prototyp diagnostické sestavy s podrobným popisem. Další část práce je zaměřena na vytvoření programového vybavení pro navrženou diagnostickou sestavu. Základem bylo vytvoření vývojových diagramů, které následně posloužily pro vytváření zdrojových kódů. Celkem se jedná o dva vývojové diagramy, jeden je použit pro obslužný program diagnostické sestavy a druhý vývojový diagram popisuje funkce WPF aplikace pro sledování videa z IP kamery. Zdrojové kódy diagnostické sestavy s využitím Arduina byly napsány v jazyce C, zatímco pro WPF aplikaci se využil programovací jazyk C#, resp. značkovací jazyk XAML. Pro správnou funkci bylo nezbytné stáhnout a importovat potřebné knihovny.

Závěrečná část diplomové práce se zabývá ověřením funkčnosti diagnostické sestavy při variabilních provozních stavech společně s navrženými metodami aktivace a deaktivace pro úsporu energie IP kamery. Při testování měření spotřeby IP kamery bylo provedeno několik různých měření při různých stavech. Na základě těchto měření byly získány hodnoty, které posloužily pro vytvoření grafického zobrazení.

SEZNAM POUŽITÉ LITERATURY

- [1] LUKÁŠ, Luděk. *Bezpečnostní technologie, systémy a management: [teorie a praxe ochrany majetku a fyzické bezpečnosti]*. Zlín: VeRBuM, 2011. ISBN 978-80-87500-05-7.
- [2] ADÁMEK, Milan. *Kamerové systémy*. 2021. Zlín: Fakulta aplikované informatiky – Univerzita Tomáše Bati ve Zlíně.
- [3] ČSN EN 62676-1-1 (334592): *Dohledové videosystémy pro použití v bezpečnostních aplikacích – Část 1-1: Systémové požadavky – Obecně*. 2014. Brno-Husovice: Technické normy, © 2003–2022. Dostupné také z: <https://shop.normy.biz/detail/95677>
- [4] JINDRA, Josef. Co znamená antivandal? *Abbas* [online]. 12.3.2015. © 2011-2022, [cit. 2022-05-02]. Dostupné z: <https://www.abbas.cz/clanky/recenze-technika/co-znamenava-antivandal/>
- [5] LOVEČEK, Tomáš a Peter NAGY. *Bezpečnostné systémy: kamerové bezpečnostné systémy*. Žilina: Žilinská univerzita, 2008. Vysokoškolské učebnice. ISBN 9788080708931.
- [6] *Nové technologické trendy v oblasti zabezpečení pro rok 2018: Nové technologie u kamerových systémů*. Security Blog [online]. 1.8.2018 [cit. 2022-01-17]. Dostupné z: <https://www.securityblog.cz/2018/08/01/nove-technologicke-trendy-v-oblasti-zabezpeceni-pro-rok-2018/>
- [7] VALOUCH, Jan. *Projektování integrovaných systémů*. Zlín: Fakulta aplikované informatiky – Univerzita Tomáše Bati ve Zlíně, 2015. ISBN 978-80-7454-557-3.
- [8] VODA, Zbyšek. *Průvodce světem Arduina*. Bučovice: Martin Stríž, 2015. ISBN 978-80-87106-93-8.
- [9] MANN, Burkhard. *C pro mikrokontroléry: ANSI-C, kompilátory C, spojovací programy – linkery, práce s ATMEL AVR a MSC-51, příklady programování v jazyce C, nástroje pro programování, tipy a triky ...* Praha: BEN – technická literatura, 2003. μ C & praxe. ISBN 80-7300-077-6.
- [10] Originál Arduino MEGA2560 REV3. *Drátek.cz* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://dratek.cz/arduino/3126-original-arduino-mega2560-rev3.html>

- [11] RFID čtečka s vestavěnou anténou. *Drátek.cz* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://dratek.cz/arduino/833-rfid-ctecka-s-vestavenou-antenou.html>
- [12] M., Luboš. RFID čtečka 13.56 MHz. *Drátek.cz: Drátek návody* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://navody.dratek.cz/navody-k-produktum/rfid-ctecka-13.56-mhz.html>
- [13] I2C 20x4 display pro jednodeskové počítače. *Drátek.cz* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://dratek.cz/arduino/1421-eses-i2c-20x4-display-pro-jednodeskove-pocitace.html>
- [14] M., Luboš. LCD Displej. *Drátek.cz: Drátek návody* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://navody.dratek.cz/zaciname-s-arduinem/lcd-displej.html>
- [15] Infračervený senzor překážek. *Drátek.cz* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://dratek.cz/arduino/3086-infracerveny-senzor-prekazek.html>
- [16] Modul relé 5V - 1 kanál s úrovnovým spínačem. *Drátek.cz* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://dratek.cz/arduino/2947-modul-rele-5v-1-kanal-s-urovnovym-spinacem.html>
- [17] FALTUS, Robert. Jednakanálový relé modul. *Drátek.cz: Drátek návody* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://navody.dratek.cz/zaciname-s-arduinem/spinaci-prvky-rele-tranzistory-a-tranzistorova-pole.html>
- [18] Ethernet Shield W5100 pro Arduino. *Drátek.cz* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://dratek.cz/arduino/917-ethernet-shield-w5100-pro-arduino.html>
- [19] M., Luboš. Arduino Ethernet Shield W5100. *Drátek.cz: Drátek návody* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://navody.dratek.cz/navody-k-produktum/arduino-ethernet-shield-w5100.html>
- [20] RTC hodiny reálného času DS3231 AT24C32 IIC paměťový modul pro Arduino. *Drátek.cz* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://dratek.cz/arduino/1261-rtc-hodiny-realneho-casu-ds3231-at24c32-iic-pametovy-modul-pro-arduino.html>
- [21] M., Luboš. TC Hodiny reálného času DS3231 + AT24C32 paměťový modul. *Drátek.cz: Drátek návody* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://navody.dratek.cz/navody-k-produktum/rtc-hodiny-realneho-casu-ds3231-at24c32-pametovy-modul.html>

- [22] Hallův senzor proudu WCS1700. *Drátek.cz* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://dratek.cz/arduino/7710-halluv-senzor-proudu-wcs1700.html>
- [23] M., Luboš. Hallův senzor proudu WCS1700. *Drátek.cz: Drátek návody* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://navody.dratek.cz/navody-k-produktum/halluv-senzor-proudu-wcs1700.html>
- [24] HAVLÍČEK, Martin a Orbit MERET. Měření kolem nás: Měření okrajových hodnot proudu a napětí. *Elektro: Časopis pro elektroniku*. FCC Public, 2020, **8-9**. ISSN 1210-0889. Dostupné také z: <http://www.odbornecasopisy.cz/elektro/casopis/tema/mereni-kolem-nas-23--17408>
- [25] HARTING S.R.O. Proudové senzory využívající Hallův jev. *Elektro Průmysl* [online]. © 2011–2022, 15.července 2019 [cit. 2022-02-18]. Dostupné z: <https://www.elektroprumysl.cz/merici-technika/proudove-senzory-vyuzivajici-halluv-jev>
- [26] What is the Hall-effect?. *Melexis* [online]. © 2021 [cit. 2022-02-18]. Dostupné z: <https://www.melexis.com/en/articles/hall-effect?fbclid=IwAR3Huv8Dh1LMn-LLD9krCy-WZYff-6W3iff3axnhXJXs879AD9IIM7Iixqtk>
- [27] Installing Fritzing. *Fritzing* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://fritzing.org/home/>
- [28] Diagram.drawio. *Diagrams.net* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://app.diagrams.net>
- [29] Visual Studio: Visual Studio 2019. *Microsoft* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://visualstudio.microsoft.com/cs/>
- [30] Download Ozeki Camera SDK. *Ozeki Camera SDK* [online]. © 2000–2022 [cit. 2022-02-18]. Dostupné z: https://camera-sdk.com/p_6513-download-onvif-ozeki-camera-sdk-for-c-sharp-.net.html
- [31] KAČMÁŘ, Dalibor. *Programujeme .NET aplikace ve Visual Studiu .NET*. Praha: Computer Press, 2001. Všechny cesty k informacím. ISBN 80-7226-569-5.
- [32] PETZOLD, Charles. *Mistrovství ve Windows Presentation Foundation: [aplikace = kód + markup]*. Brno: Computer Press, 2008. ISBN 9788025121412.

- [33] VIRIUS, Miroslav. *Cvičení k programování pro .NET*. Praha: České vysoké učení technické, 2019. ISBN 9788001065464.
- [34] Smart Motion Detection – Inteligentní detekce pohybu lidí. *Kamery-VIVOTEK* [online]. © 2022 [cit. 2022-02-18]. Dostupné z: <https://www.kamery-vivotek.cz/clanky/technicke-pojmy/vivotek-inteligentni-detekce-pohybu-lidi/>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

A	Ampér
ACS	Access Control System
AES	Automatic Electronic Shutter
AGC	Automatic Gain Control
Bd	Baud
CCD	Charge-coupled device
cm	Centimetr
CMOS	Complementary Metal-Oxide-Semiconductor
CPU	Central Processing Unit
DOF	Deep of field
DRAM	Dynamic Random Access Memory
DSP	Digital Signal Processor
DV	Dohledové videosystémy
DVR	Digital Video Recorder
EEPROM	Electrically Erasable Programmable Read-Only Memory
FTP	File Transfer Protocol
GB	Gigabajt
GND	Ground
GUI	Graphical User Interface
HIGH	Vysoká logická úroveň
I2C	Inter-Integrated Circuit
IP	Internet Protocol
IPS	Integrovaný poplachový systém
IDE	Integrated Development Environment

IPv4	Internet Protocol version 4
IR	Infrared radiation
IRQ	Interrupt ReQuest
KB	Kilobyte
kHz	Kilohertz
LAN	Local Area Network
LCD	Liquid Crystal Display
LED	Light Emitting Diode
LOW	Nízká logická úroveň
m	Metr
mA	Miliampér
Mb/s	Megabit za sekundu
MHz	Megahertz
MISO	Master In Slave Out
mm	Milimetr
MOSI	Master Out Slave In
MTF	Modulation Transfer Function
mV	Milivolt
např	Například
NC	Normally Closed
NIC	Network Interface Card
nm	Nanometr
NO	Normally Open
NVR	Network Video Recorder
PoE	Power over Ethernet
PZTS	Poplachové zabezpečovací a tísňové systémy

QR	Quick Response
RFID	Radio Frequency Identification
RTC	Real-time clock
RZ	Registrační značka
SCL	Serial Clock
SDA	Serial Data
SKV	Systemy kontroly vstupu
SPI	Serial Peripheral Interface
TWI	Two-Wire Interface
USB	Universal Serial Bus
UTP	Unshielded Twisted Pair
V	Volt
VCC	Voltage Common Collector
VMS	Video Management System
VRC	Videocassette recorder
WPF	Windows Presentation Foundation
°C	Stupeň Celsia

SEZNAM OBRÁZKŮ

Obr. 1. Blokové schéma funkce IP kamery. [1].....	12
Obr. 2. Objektivy s uchycením typu CS a C. [5].....	13
Obr. 3. Zobrazení optickým systémem. [5].....	14
Obr. 4. Připojení IP kamery přes PoE přepínač. [1].....	19
Obr. 5. Integrace s využitím síťové kamery. [7].....	23
Obr. 6. Vývojové rozhraní Arduino IDE.	26
Obr. 7. Deklarace proměnné a přiřazení hodnoty do proměnné.	26
Obr. 8. Nastavení časové prodlevy bez proměnné.....	27
Obr. 9. Nastavení časové prodlevy s proměnnou.	27
Obr. 10. Přehled datových typů.	28
Obr. 11. Informace o připojení na port COM7.	28
Obr. 12. Otevření sériového monitoru.	29
Obr. 13. Arduino Mega.....	33
Obr. 14. RFID čtečka RC522.....	34
Obr. 15. LCD displej.....	34
Obr. 16. LCD displej a I2C převodník.....	35
Obr. 17. IR senzor.....	35
Obr. 18. Modul relé 5 V.....	36
Obr. 19. Ethernet Shield W5100.....	37
Obr. 20. Ethernet Shield W5100 – vývody na propojení.....	38
Obr. 21. Propojení Shieldu s Arduinem.....	38
Obr. 22. RTC hodiny reálného času.	39
Obr. 23. RTC hodiny reálného času – zadní strana, patice pro vložení baterie.....	39
Obr. 24. Hallův senzor proudu.....	40
Obr. 25. Schéma měření pomocí Hallova sondy. [24].....	41
Obr. 26. Princip Hallova jevu. [26].....	42
Obr. 27. MicroSD karta a adaptér.....	43
Obr. 28. LED diody, rezistory a stohovatelné dutinkové lišty.	44
Obr. 29. Kabelové průchodky, objímky na LED diody, tlačítko a spínač.....	44
Obr. 30. Propojovací kabely.	45
Obr. 31. Bateriový box.	45
Obr. 32. Adaptér pro napájení.	46

Obr. 33. Schéma zapojení.	46
Obr. 34. Schéma zapojení.	47
Obr. 35. Vytvořená diagnostická sestava – přední strana.	48
Obr. 36. Umístění IR senzoru.	49
Obr. 37. Vytvořená diagnostická sestava – zadní strana.	49
Obr. 38. Umístění Hallova senzoru.	50
Obr. 39. Uložení komponent uvnitř sestavy.	51
Obr. 40. Uložení komponent uvnitř sestavy.	51
Obr. 41. Vytvořené svorkovnice.	52
Obr. 42. RFID karty k ovládání diagnostické sestavy.	52
Obr. 43. IP kamera Vivotek.	53
Obr. 44. IP kamera Axis.	53
Obr. 45. PoE přepínač.	54
Obr. 46. Adaptér pro napájení PoE přepínače.	54
Obr. 47. Konektory pro kabeláž k PoE.	54
Obr. 48. Kabeláž pro připojení k PoE přepínači.	55
Obr. 49. Vývojový diagram – Arduino IDE.	57
Obr. 50. Vývojový diagram Visual Studio – aplikace WPF.	58
Obr. 51. IDE – importování knihoven.	60
Obr. 52. IDE – Pin pro microSD kartu a proměnná pro uložení dat.	60
Obr. 53. IDE – proměnná pro datum a čas a časová prodleva.	60
Obr. 54. IDE – nastavení LCD displeje a RFID čtečky.	60
Obr. 55. IDE – nastavení povolené RFID karty.	61
Obr. 56. IDE – kontrola microSD karty.	61
Obr. 57. IDE – výpočet napětí a odebíraného proudu.	62
Obr. 58. IDE – detekován pohyb.	62
Obr. 59. IDE – uložení data a času.	63
Obr. 60. IDE – otevření microSD karty.	63
Obr. 61. IDE – uložení naměřených hodnot.	63
Obr. 62. IDE – přiložení RFID karty.	64
Obr. 63. IDE – porovnání RFID karty.	64
Obr. 64. IDE – vypnutí IP kamery a zhasnutí modré LED diody.	65
Obr. 65. IDE – výpis na LCD displej a rozsvícení zelené LED diody.	65

Obr. 66. IDE – uložení informací na microSD kartu.....	65
Obr. 67. IDE – zhasnutí zelené LED diody.	65
Obr. 68. IDE – výpis na LCD displej a rozsvícení červené LED diody.	66
Obr. 69. IDE – uložení informací na microSD kartu.....	66
Obr. 70. IDE – zhasnutí červené LED diody.	66
Obr. 71. IDE – není detekován pohyb.	67
Obr. 72. WPF aplikace pro sledování videa z IP kamery.	68
Obr. 73. C# – definování videoVieweru pro zobrazení videa.	69
Obr. 74. C# – definování tlačítek v GUI.....	69
Obr. 75. C# – importování knihoven.	69
Obr. 76. C# – vytvoření proměnných.	69
Obr. 77. C# – přiřazení proměnných k předdefinovaným funkcím.....	70
Obr. 78. C# – propojení IP kamery s jednotlivými proměnnými.	70
Obr. 79. C# – zapnutí IP kamery a videoVieweru.....	70
Obr. 80. C# – efekt zmáčknutého tlačítka.	71
Obr. 81. C# – barevné zvýraznění pohybu v obraze.....	71
Obr. 82. C# – přidání události pro Motion Detection.	71
Obr. 83. C# – vytvoření a uložení snímku.....	72
Obr. 84. C# – vybrání cesty pro uložení snímku.	72
Obr. 85. C# – uložení snímku z IP kamery.....	73
Obr. 86. C# – vytvoření aktuálního data a času.....	73
Obr. 87. C# – uložení snímku.	73
Obr. 88. Sériový monitor – neúspěšné načtení microSD karty.	74
Obr. 89. Sériový monitor – úspěšné načtení microSD karty.	74
Obr. 90. LCD displej – IP kamera OFF.....	75
Obr. 91. Záznam z microSD karty – není pohyb.	75
Obr. 92. LCD displej – IP kamera ON.	75
Obr. 93. Záznam z microSD karty – detekován pohyb.	76
Obr. 94. LCD displej – správná RFID karta.	76
Obr. 95. Záznam z microSD karty – správná RFID karta.	76
Obr. 96. LCD displej – nesprávná RFID karta.	77
Obr. 97. Záznam z microSD karty – nesprávná RFID karta.....	77
Obr. 98. WPF aplikace – detekován pohyb, připojení na IP kameru.	78

Obr. 99. WPF aplikace – aktivování funkce Motion Detection.....	78
Obr. 100. Měření Halloovým senzorem, IP kamera Axis.	80
Obr. 101. Měření Halloovým senzorem, IP kamera Vivotek.	80
Obr. 102. Protažený vodič skrz Hallův senzor.	81
Obr. 103. Vnitřní napětí na Hallově senzoru 15 mV, proud na vodiči 0.47 A.	81
Obr. 104. Průběh měření proudu – IP kamera Axis (vzorky 0 až 50)	84
Obr. 105. Voltampérová charakteristika – IP kamera Axis.....	84
Obr. 106. Průběh měření proudu – IP kamera Vivotek (vzorky 0 až 50).....	86
Obr. 107. Voltampérová charakteristika – IP kamera Vivotek.....	86

SEZNAM TABULEK

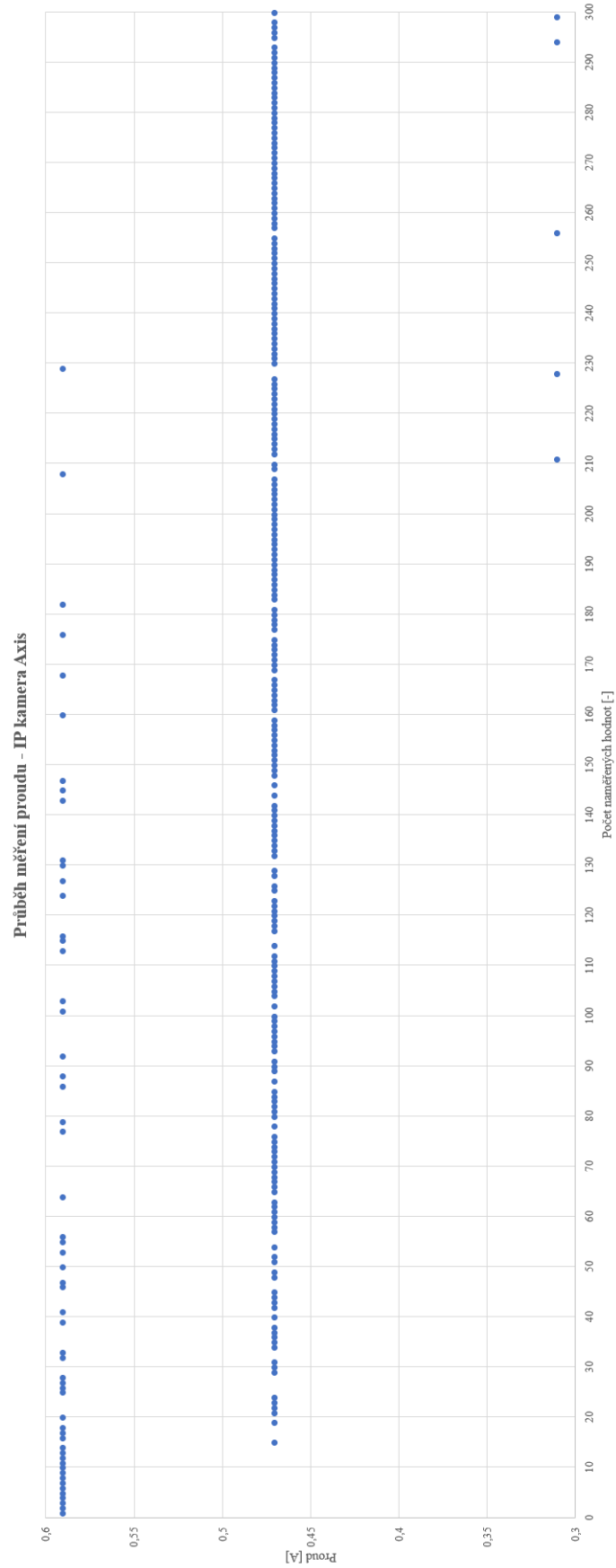
Tab. 1. Zapojení vývodů z RFID čtečky na piny Arduina.....	33
Tab. 2. Vstupní piny na relé.....	36
Tab. 3. Popis jednotlivých pinů, které využívá Ethernet Shield.....	37
Tab. 4. Zapojení vývodů z RTC modulu na piny Arduina.	39
Tab. 5. Zapojení vývodů z Hallova senzoru na piny Arduina.	40
Tab. 6. Zapojení výstupů na piny Arduina.	44
Tab. 7. Výpočet absolutní a relativní chyby – IP kamera Axis.	82
Tab. 8. Zápis hodnot pro první řádek tabulky.....	83
Tab. 9. Průměrné hodnoty napětí a proudu – IP kamera Axis.....	83
Tab. 10. Výpočet absolutní a relativní chyby – IP kamera Vivotek.....	85
Tab. 11. Průměrné hodnoty napětí a proudu – IP kamera Vivotek.	85

SEZNAM PŘÍLOH

Příloha P I: MĚŘENÍ PROUDU – IP KAMERA AXIS

Příloha P II: MĚŘENÍ PROUDU – IP KAMERA VIVOTEK

PŘÍLOHA P II: MĚŘENÍ PROUDU – IP KAMERA AXIS



PŘÍLOHA P IV: MĚŘENÍ PROUDU – IP KAMERA VIVOTEK

