



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV MIKROELEKTRONIKY

DEPARTMENT OF MICROELECTRONICS

## PŘENOSNÉ ZAŘÍZENÍ PRO DETEKCI VŮNĚ V POTRAVINÁŘSTVÍ

THE PORTABLE DEVICE FOR AROMA DETECTING IN FOOD INDUSTRY

### BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

### AUTOR PRÁCE

AUTHOR

David Beneš

### VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Martin Adámek, Ph.D.

BRNO 2016



# Bakalářská práce

bakalářský studijní obor **Mikroelektronika a technologie**  
Ústav mikroelektroniky

**Student:** David Beneš

**ID:** 164242

**Ročník:** 3

**Akademický rok:** 2015/16

## NÁZEV TÉMATU:

### Přenosné zařízení pro detekci vůně v potravinářství

#### POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s problematikou monitorování vůně v potravinářství a s vývojovou deskou řady Arduino. Seznamte se s vývojovým prostředím pro tuto platformu.

Navrhnete jednoduché přenosné zařízení pro detekci vůně v potravinářství řízené modulem řady Arduino. Jednotku navrhnete jako autonomní systém s možností sdílení dat pomocí počítačové sítě. Vytvořte jednoducho databázi vůní a detekovanou vůni porovnávejte s touto databází. Přístroj otestujte na vybraných potravnářských vůních. Zhodnoťte váš návrh a jeho možnosti.

#### DOPORUČENÁ LITERATURA:

Podle pokynů vedoucího práce.

**Termín zadání:** 8.2.2016

**Termín odevzdání:** 2.6.2016

**Vedoucí práce:** Ing. Martin Adámek, Ph.D.

**Konzultant bakalářské práce:**

**doc. Ing. Jiří Háze, Ph.D., předseda oborové rady**

#### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

## **ABSTRAKT:**

Tato práce se zabývá konstrukcí jednoduchého levného přenosného zařízení, které umožňuje měření a rozpoznávání vůní v potravinářství pomocí polovodičových senzorů plynů s možností sdílet naměřená data se vzdáleným serverem. Zařízení je řízeného vývojovou platformou Arduino, obohacenou o dva moduly. Práce stručně seznamuje čtenáře s problematikou monitorování vůní v potravinářství a vývojovou deskou Arduino osazeného mikrokontrolérem řady ATmega společně s modulem pro síťovou komunikaci. Součástí práce je i příklad měření a jeho následné vyhodnocení při použití navrženého zapojení.

## **ABSTRACT:**

This work deals with the construction of a simple cheap portable device that allows the measurement and recognition of aromas in food industry using semiconductor gas sensors with the ability to share measurement data to a remote server. The device is controlled by Arduino development platform, enriched by two modules. Work shortly apprises readers with aroma monitoring in food industry and development board Arduino which contains ATmega series microcontroller and with module for network communication. This work also includes a measurement sample and review using the proposed circuit.

## **KLÍČOVÁ SLOVA:**

Arduino, mikrokontrolér, plynové senzory, elektronický nos, vůně

## **KEYWORDS:**

Arduino, microcontroller, gas sensors, e-nose, aroma

BENEŠ, D. *Přenosné zařízení pro detekci vůně v potravinářství*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav mikroelektroniky, 2016. 63 s. Bakalářská práce. Vedoucí práce: Ing. Martin Adámek, Ph.D.

## **PROHLÁŠENÍ**

Prohlašuji, že jsem tuto vysokoškolskou kvalifikační práci vypracoval samostatně pod vedením vedoucího semestrální práce, s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury. Jako autor uvedené semestrální práce dále prohlašuji, že v souvislosti s vytvořením této semestrální práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení.

V Brně dne .....

.....  
(podpis autora)

## **PODĚKOVÁNÍ**

Tímto bych rád poděkoval vedoucímu bakalářské práce Ing. Martinu Adámkovi, Ph.D. za odborné vedení, za pomoc a rady při zpracování této práce.

V Brně dne .....

.....  
(podpis autora)

# OBSAH

<b>ÚVOD</b> .....	<b>8</b>
<b>1 VNÍMÁNÍ VŮNĚ A JEJÍ MONITOROVÁNÍ</b> .....	<b>9</b>
1.1 DEFINICE PACHU.....	9
1.2 VNÍMÁNÍ VŮNĚ ČLOVĚKEM.....	10
1.3 ZAŘÍZENÍ PRO SNÍMÁNÍ PACHŮ.....	10
1.3.1 <i>Laboratorní metody</i> .....	11
1.3.1.1 Hmotnostní spektrometrie.....	11
1.3.1.2 Plynová chromatografie.....	11
1.3.2 <i>Polovodičové senzory</i> .....	12
1.3.2.1 Chemorezistory.....	12
1.3.2.2 Chemokondenzátory.....	14
1.3.2.3 Chemodiody.....	15
1.3.2.4 Chemotranzistory.....	15
1.3.2.5 Teplotní chemické senzory.....	15
1.3.2.6 Gravimetrické chemické senzory.....	16
1.3.2.7 Optické chemické senzory.....	17
1.4 MONITOROVÁNÍ VŮNĚ V POTRAVINÁŘSTVÍ.....	17
1.4.1 <i>Olfaktometrické měření</i> .....	19
<b>2 VÝVOJOVÁ PLATFORMA ARDUINO</b> .....	<b>20</b>
2.1 VÝVOJOVÉ PROSTŘEDÍ ARDUINA.....	21
2.2 HARDWAROVÉ USPOŘÁDÁNÍ.....	22
2.2.1 <i>AVR</i> .....	22
2.2.2 <i>ATmega</i> .....	22
2.2.3 <i>Čítač / časovač</i> .....	23
2.2.4 <i>Přerušení</i> .....	23
2.2.5 <i>Pulzně šířková modulace</i> .....	23
2.2.6 <i>A/D převodník</i> .....	23
2.2.7 <i>Serial Peripheral Interface</i> .....	24
2.3 ETHERNET SHIELD.....	26
2.3.1 <i>Síťová komunikace</i> .....	26
2.3.2 <i>Protokol DHCP</i> .....	27
2.3.3 <i>Protokol HTTP verze 1.1</i> .....	29
2.3.4 <i>Externí mikroSD FLASH paměť</i> .....	30
2.4 LCD SHIELD.....	32
<b>3 HYPER TEXT MARKUP LANGUAGE</b> .....	<b>33</b>
3.1 KASKÁDOVÉ STYLY.....	34
3.2 JAVASCRIPT.....	34
<b>4 NÁVRH A REALIZACE ZAŘÍZENÍ PRO MONITOROVÁNÍ VŮNÍ</b> .....	<b>35</b>
4.1 KONSTRUKCE ZAŘÍZENÍ.....	35
4.2 NÁVRH PROGRAMOVÉHO VYBAVENÍ.....	37
4.3 MĚŘENÍ A DISKUSE.....	40
4.3.1 <i>Databáze vůní</i> .....	41

<b>5 ZÁVĚR.....</b>	<b>44</b>
<b>LITERATURA .....</b>	<b>45</b>
<b>SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK .....</b>	<b>48</b>
<b>SEZNAM OBRÁZKŮ .....</b>	<b>49</b>
<b>SEZNAM TABULEK.....</b>	<b>51</b>
<b>SEZNAM PŘÍLOH .....</b>	<b>52</b>

# ÚVOD

Člověk vnímá svými smysly okolní svět již od pradávna. Čich je dokonce vývojově nejstarším smyslem. Evoluce však naše způsoby a schopnosti vnímání vůní negativně ovlivnila a čich se stal nejméně schopným smyslem člověka. Nicméně stále je člověk schopen od sebe rozlišit na 4000 různých látek a každý den tuto schopnost používá, především při příjmu potravin, jelikož čich je úzce svázán s chutí. Pach může obecně navodit pozitivní i negativní pocity nebo může člověka upozornit na nebezpečné prostředí. Jako nejjednodušší elektronické indikátory pachů se používají elektronické nosy s polovodičovými senzory. Použití elektronických nosů postupně nachází uplatnění v různých oborech lidské činnosti. Obecně mají společný cíl a to chránit lidské zdraví nebo majetek. Využití nachází i v kriminalistice při detekci drog a jiných návykových látek, ve vojenství pro odhalování nebezpečných oblastí, v dopravě pro zjišťování alkoholu v krvi řidičů, v ekologii pro stanovení kvality ovzduší apod.

Identifikace nejrůznějších vůní se stává důležitou součástí zejména v potravinářském průmyslu. Sensorická analýza vůně u potravin je jednou z nejjednodušších zkoušek hodnocení potravin, proto bývá jednou z nejčastěji prováděných zkoušek v laických testech. Pro sensorickou analýzu vůně využívají potravinářské podniky znalostí a zkušeností profesionálních hodnotitelů, kteří jsou trénováni na určitý sortiment potravin. Při hodnocení se nejčastěji využívá párového nebo trojúhelníkového testu. Hodnotící metoda vychází ze schopností lidského čichového ústrojí - olfaktometrická metoda. Náročnější rozборы mohou využít přesnější laboratorní metody - spektrální analýza nebo plynová chromatografie. Méně přesná zařízení využívají např. polovodičových senzorů a jsou často nazývána tzv. elektronickými nosy. V potravinářství se jich především využívá pro detekci nedostatečně skladovaných potravin, které mohou způsobit při požití zdravotní potíže. Každá z metod detekce vůně má své klady i zápory a je třeba volit tu nejvhodnější.

Cílem práce je návrh a realizace levného, jednoduchého přenosného zařízení schopného detekovat změnou vůni na základě porovnání se svými ostatními vzorky, které jsou uloženy v databázi na jeho úložišti. Cílem je rovněž i vytvoření webového rozhraní, umožňujícího sdílet data zařízení prostřednictvím počítačové sítě nebo i zařízení vzdáleně ovládat. Zařízení pro rychlejší a flexibilnější vývoj je navrhováno na platformě Arduino s použitím dalších hardwarových prostředků. Zařízení je připojováno do počítačové sítě, která je používána ke sdílení a zpracování naměřených dat ukládaných do jednoduché webové databáze *Lighttpd*. Návrh je koncipován tak, že provoz serveru bude zatěžovat minimálně, od toho se odvíjí i velmi jednoduché ovládací rozhraní.

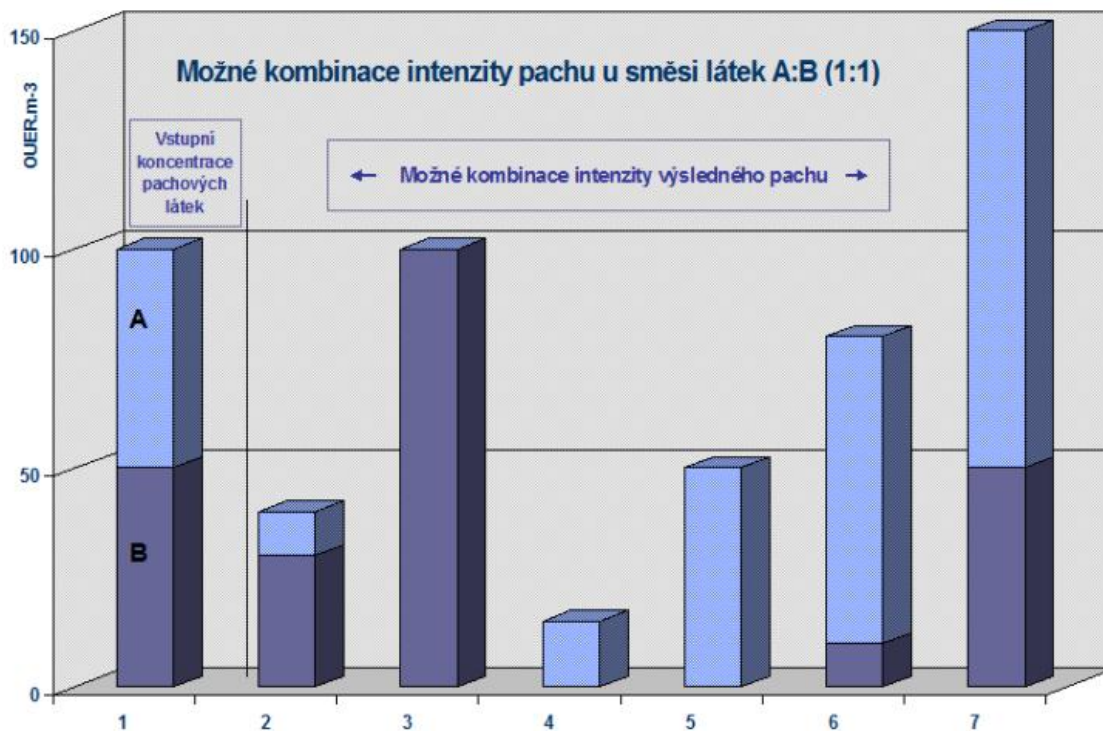


# 1 VNÍMÁNÍ VŮNĚ A JEJÍ MONITOROVÁNÍ

## 1.1 Definice pachu

Norma ČSN EN ISO 5492 (Senzorická analýza - Slovník) definuje pach následovně: "Pach je organoleptická (smyslová) vlastnost, která je vnímána čichovým orgánem po vdechnutí určitého objemu látky." [1]

Pachy jsou obvykle tvořeny směsí nejrůznějších chemických látek, jež jsou vzájemně ovlivňovány. Zpravidla se jedná o stovky takových látek o rozdílných koncentracích. Například vůni kávy tvoří asi 800 různých chemických látek. Koncentraci jednotlivých látek lze vhodnými metodami určit, ovšem ani taková analýza nedokáže určit přesný charakter pachu nebo jeho intenzitu. Nepřesnosti vznikají právě kvůli vzájemným interakcím mezi jednotlivými látkami, kde některé se navzájem maskují, jiné sčítají apod. Příklad různých kombinací intenzity pachu dvousložkové směsi o stejných koncentracích je znázorněn na obr. 1. Z toho plyne, že dva naprosto rozdílné pachy mohou obsahovat stejné chemické látky o stejných koncentracích. [2]



Obr. 1: Různé kombinace intenzity pachu dvou látek o stejné koncentraci

## 1.2 Vnímání vůně člověkem

Přesto, že je čich vývojově jedním z nejstarších smyslů, stále se jedná o doposud nejméně prozkoumaný smysl. Může za to především fakt, že se jedná spíše o subjektivní vnímání. Čich je navíc u člověka jen málo vyvinutý oproti zvířatům. Princip vnímání vůně je založen na řadě chemických reakcí, ke kterým dochází při kontaktu molekuly plynu s receptory vůně, jež vyvolají proud elektrických signálů do neuronů, které mozek zpracovává. U člověka tvoří povrch čichové sliznice kolem  $5 \text{ cm}^2$ , kde je rozseto na 1000 různých typů čichových receptorů. Zatímco čichová sliznice psa zabírá plochu až  $170 \text{ cm}^2$ , což dokazuje jejich lepší čichové vnímání. Člověk je schopen rozlišit maximálně 4000 různých látek, přičemž aby byl schopen odlišit rozdíly v intenzitě pachu, je třeba aby se tato intenzita zvýšila alespoň o 30 %. Rovněž pachy mají na psychický stav člověka nejsilnější účinky ze všech jeho smyslů. Ve vyšších koncentracích může dokonce vyvolávat i různé zdravotní potíže (nevolnost, bolest hlavy) nebo naopak pocity příjemné. [3, 4]

## 1.3 Zařízení pro snímání pachů

I přes to, že člověk je schopen odlišit řadu vůní, v některých případech jsou jeho čichové smysly nedostačující a přicházejí na řadu jiné metody určování pachu, kde kromě pomoci zvířat lze využít především tzv. "elektronické nosy". Jedná se o zařízení, které je schopné monitorovat koncentraci určitých látek v plynu, nejedná se tedy na rozdíl od lidského čichu o subjektivní vnímání. Tato zařízení bývají obvykle vybavena hned několika polovodičovými plynovými senzory. Každý je citlivý jen na určitý typ plynu nebo skupinu plynů. Zařízení rovněž obsahuje databázi referenčních vzorků, která slouží k vyhodnocování měřených vzorků. Nejprve je tedy nutné elektronický nos vůně "naučit". Přesnější zařízení navíc mohou kombinovat různé metody měření koncentrací látek v plynu. Elektronické nosy pak mohou najít uplatnění u bezpečnostních systémů (detekce hořlavých či jinak nebezpečných látek pro člověka), při ochraně životního prostředí (určení znečištění ovzduší), ve zdravotnictví a především v potravinářství. [3, 5]

Přítomnost plynů v určitém objemu lze měřit různými metodami. Jednou z možností je použití přesných analyzátorů, pracujících na principu plynové chromatografie, hmotnostní spektrometrie nebo dokonce lze i obě tyto metody zkombinovat. Takové analyzátory jsou ovšem drahé, jen těžko přenositelné a náročné na obsluhu. Přichází proto na řadu polovodičové chemické senzory, jež se vyznačují svojí jednoduchostí, nízkou cenou, nižší spotřebou a vyšší rychlostí měření. Na druhou stranu ale jsou obvykle méně přesné, protože bývají citlivé jen na určitou skupinu plynů. [6]

## 1.3.1 Laboratorní metody

### 1.3.1.1 Hmotnostní spektrometrie

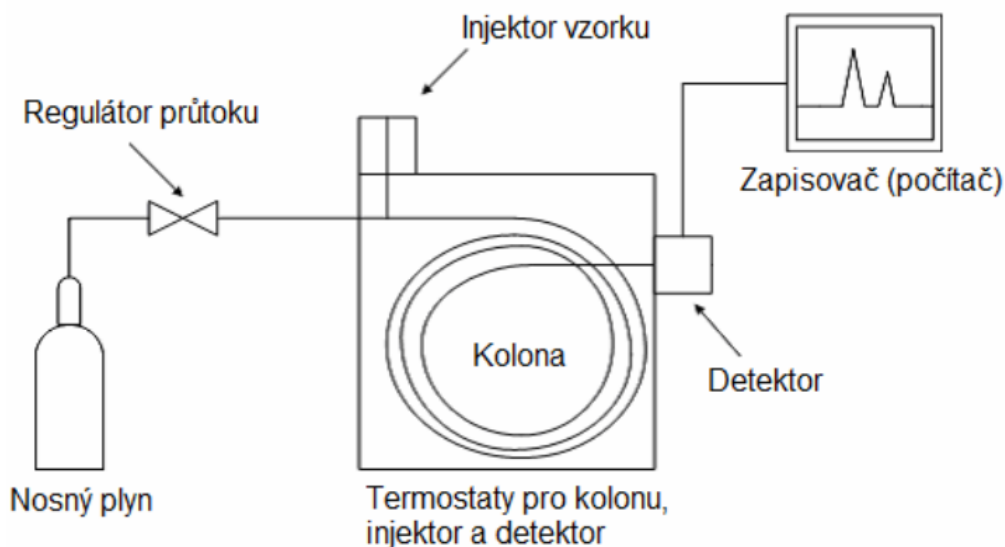
Hmotnostní spektrometrie je analytickou chemickou metodou, která je používána především ke zkoumání organických látek. Lze ji použít při kvantitativní i kvalitativní analýze, například k určování složení vzorku nebo ke stanovení hmotnosti částic. V praxi se hmotnostní spektrometrie používá k detailnímu rozboru různých sloučenin v analytických laboratořích.

Metoda je založena na ionizaci molekul zkoumané látky a jejich následném rozlišení podle poměru hmotnosti ku velikosti náboje díky interakci s elektrickými nebo magnetickými poli. Hmotnostní spektrometr se skládá ze tří hlavních částí. První částí je zdroj iontů, který ionizuje molekuly zkoumané látky. Druhou částí je hmotnostní analyzátor, jenž pomocí elektromagnetického pole rozděljuje ionty podle jejich hmotnosti. Třetí částí je detektor, který měří četnosti iontů dle podílu jejich hmotnosti a velikosti náboje, čímž poskytuje data pro výpočet hojnosti iontu. [7]

### 1.3.1.2 Plynová chromatografie

Plynová chromatografie je separační metodou zkoumání plynů, kapalin nebo i pevných látek s bodem varu do 400 °C. Hlavními výhodami této metody je především její rychlost, jednoduchost a účinnost separace zkoumaných látek, jelikož k analýze postačí i relativně malé množství vzorku. Plynovou chromatografií lze dále kombinovat s ostatními analytickými metodami (např. s hmotnostní spektrometrií) a dosáhnout lepších výsledků.

Princip funkce je založen na rozdělení složek plynu do dvou fází - pohyblivé (mobilní) a nepohyblivé (stacionární). Uspořádání plynového chromatografu je znázorněno na obr. 2. Nepohyblivou fází je pevná látka (silikagel, oxid hlinitý), která je umístěna v chromatografické koloně. Pohyblivá fáze zde představuje nosný plyn, který neustále prochází kolonou. Následně je do injektoru nanesen vzorek, kde je odpařen a dále unášen nosným plynem. Vytvořená směs je postupně v koloně pohlcována stacionární fází a dochází k dělení jednotlivých složek. Rychlost šíření každé složky vzorku kolonou je závislá na distribuční konstantě složky  $K_D = c_s / c_m$ , kde  $c_s$  a  $c_m$  jsou rovnovážné koncentrace složky ve stacionární a mobilní fázi. Jednotlivé látky vycházejí z kolony postupně, dle jejich distribučních konstant a vstupují do detektoru, který určí koncentrace separovaných látek v nosném plynu. [8, 9]



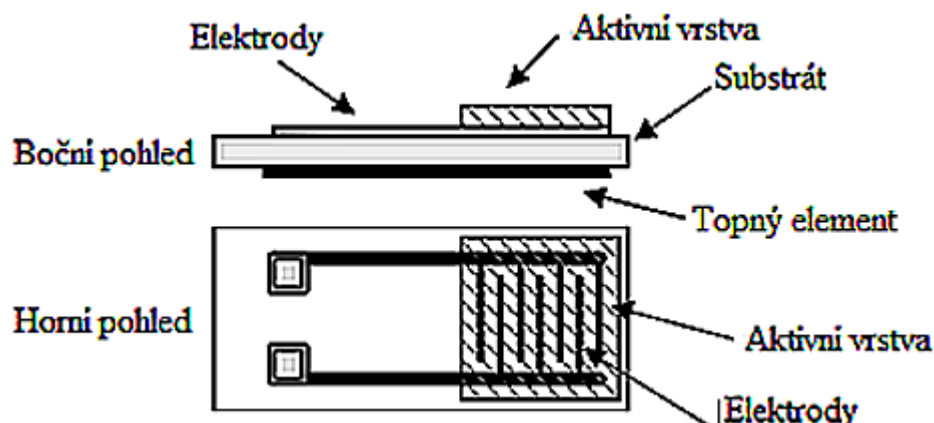
Obr. 2: Schéma uspořádání plynového chromatografu [9]

## 1.3.2 Polovodičové senzory

### 1.3.2.1 Chemorezistory

Chemorezistory nebo-li chemické vodivostní senzory patří mezi nejjednodušší a nejvíce rozšířené senzory. To je dáno především jejich snadným zapojením, snadnou výrobou, nízkou cenou, dlouhodobou stabilitou a životností. Chemorezistory se vyznačují malou selektivitou, která může být v některých aplikacích vhodná, jestliže jsou detekovatelné plyny podobného charakteru (výbušné, toxické), což lze využít např. u bezpečnostních hlásičů apod. Naopak, jestliže je vyžadována vyšší selektivita senzorů, lze senzory vybavit požadovanými filtry, které mohou alespoň částečně pohltnout plyny, které nemají být měřeny.

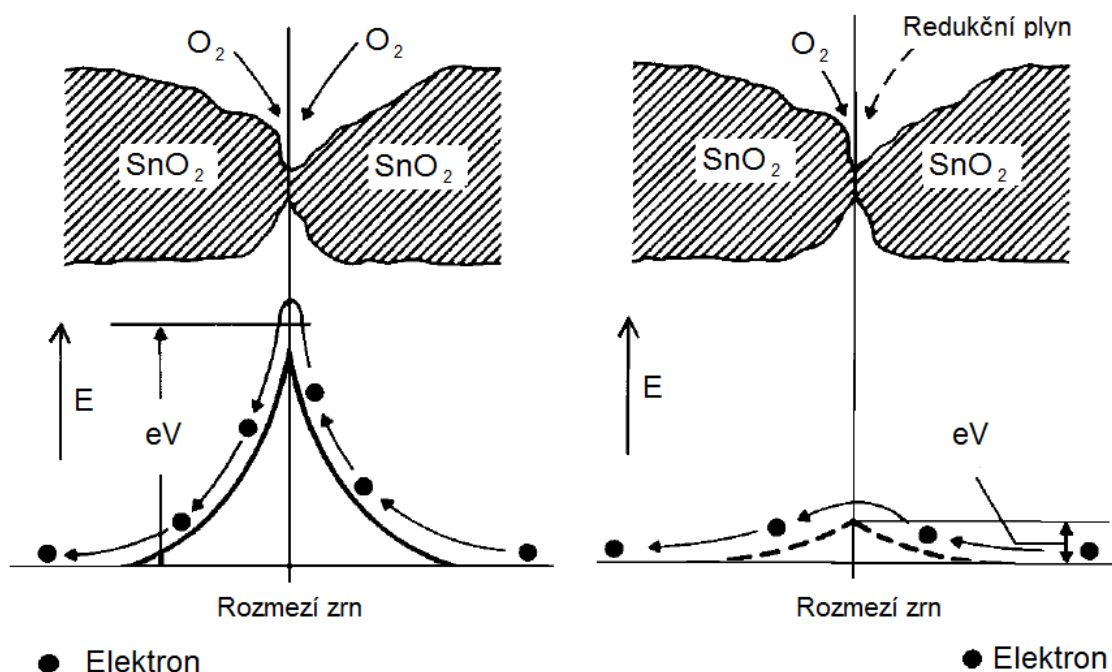
Senzory lze zhotovovat pomocí tenkovrstvé i tlustovrstvé technologie (obr. 3), kde substrát bývá zhotoven z korundové keramiky ( $\text{Al}_2\text{O}_3$ ), zatímco u tenkovrstvé je použit  $\text{SiO}_2$  nebo  $\text{Si}_3\text{N}_4$ . Jelikož substrát musí být dostatečně tepelně odolný a při svém zahřátí nesmí téměř ovlivňovat složení okolního plynu. Aktivní vrstva chemorezistorů bývá tvořena různými oxidy kovů ( $\text{SnO}_2$ ,  $\text{ZnO}$ ,  $\text{WO}_3$ ,  $\text{G}_2\text{O}_3$ , apod.).



Obr. 3: Struktura tlustovrstvého chemorezistoru [11]

Tyto vrstvy mají být chemicky stálé, bez možnosti oxidace. Vytvořením pórovitého povrchu lze zvýšit kontaktní plochu s plynem a tím i citlivost senzoru. Dále také vhodnou dotací této vrstvy lze upravit koncentrace volných nosičů a změnit tak výslednou citlivost. Podobného efektu lze dosáhnout i nanesením katalyzátorů na povrch aktivní vrstvy, což sníží potřebnou aktivační energii a dovolí rychlejší průběh chemických reakcí, tím se nejen zvýší citlivost ale i zkrátí doba odezvy. Další nezbytnou součástí senzoru je jeho topné tělísko (Pt meandrovitá vrstva), které slouží k zahřátí aktivní vrstvy (u některých senzorů až 1000 °C), čímž je do procesu dodána energie a atomy daného plynu tak snadněji překonají aktivační energii chemických reakcí.

Chemorezistory tak při kontaktu s detekovaným plynem mění vodivost aktivní vrstvy v závislosti na jeho koncentraci. Jedná-li se o N-typ polovodičové vrstvy, kyslík vodivost snižuje, jelikož na rozhraní plynu a aktivní vrstvy vznikají kyslíkové ionty, které odsávají elektrony z aktivní vrstvy, čímž se snižuje množství nosičů náboje a vzniká potenciálová bariéra (viz obr. 4). Ta postupně roste a brání další adsorpci a přenosu náboje, rezistivita vrstvy se zvýší. Tyto změny vzniklé při adsorpci (přichycení na povrch), absorpci (vnik do objemu) nebo chemisorpci (vznik chemických vazeb) částic plynů s aktivní vrstvou vedou ke změně povrchové či vnitřní vodivosti (rezistivity), která je následně měřena a zpracovávána. Zatímco u P-typu aktivní vrstvy probíhá mechanismus obráceně. [6, 10, 11]

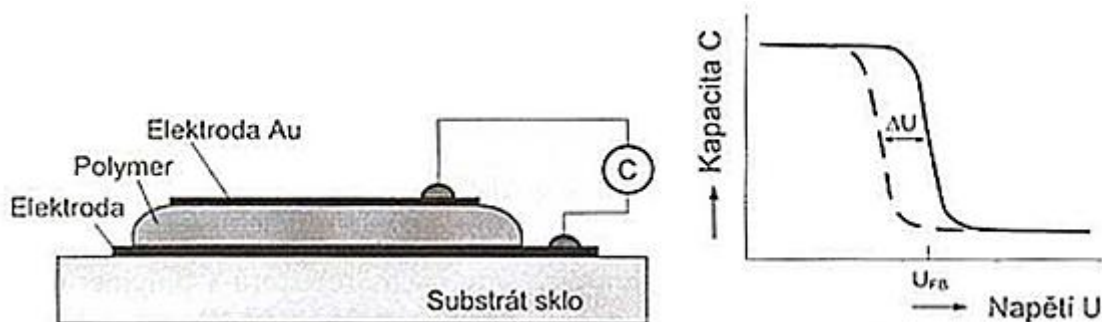


Obr. 4: Zobrazení potenciálové bariéry na vzduchu a redukčním plynu [6]

### 1.3.2.2 Chemokondenzátory

Tyto senzory jsou tvořeny pórovitou, chemicky citlivou vrstvou, jež je umístěna mezi dvě elektrody. Citlivá vrstva je obvykle tvořena polymery, na který je napařena polopropustná zlatá elektroda. (viz obr. 5)

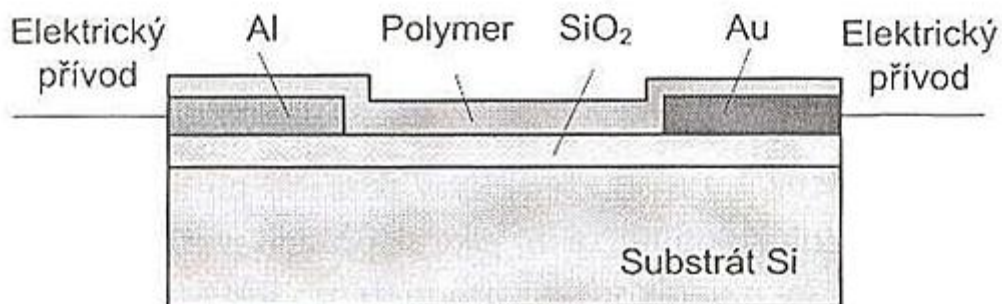
Jestliže dojde ke styku chemicky citlivé vrstvy s požadovaným plynem, dojde ke změně její relativní permitivity  $\epsilon$ , což se projeví na změně kapacity (v řádech pF). Tyto změny jsou velmi závislé na vlhkosti, teplotě prostředí nebo i na pracovní frekvenci měřicího obvodu, což lze považovat za nevýhody těchto senzorů. Na druhou stranu mají rychlejší odezvu a jsou mechanicky odolnější. [28, 29]



Obr. 5: Struktura chemokondenzátoru a jeho převodní charakteristika [6]

### 1.3.2.3 Chemodiody

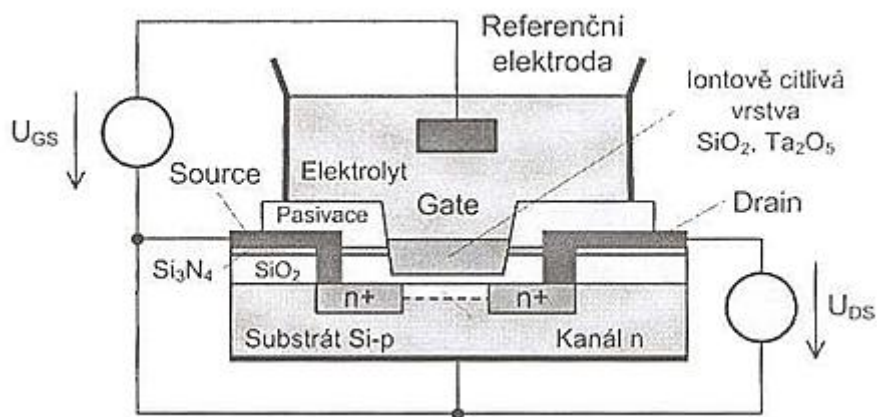
Chemodiody využívají vlastností Shottkyho PN přechodu, kde jedna elektroda je kovová, zatímco druhou tvoří polymerový polovodič P-typu (obr. 6). Vlastnosti polymeru se tak mění podle přítomnosti daného plynu. Tím se mění VA charakteristika přechodu diody. [10, 11]



Obr. 6: Struktura chemodiody [6]

### 1.3.2.4 Chemotranzistory

Princip činnosti chemotranzistorů je velmi podobný principu chemodiody. Tyto senzory jsou založeny na MOS struktuře, kde chemicky citlivá vrstva nahrazuje hradlo gate (obr. 7). Díky tomu při působení plynu na senzor dochází ke změně některých charakteristik tranzistoru. Výhodou těchto senzorů je jejich nižší šum výstupního signálu. [29, 30]



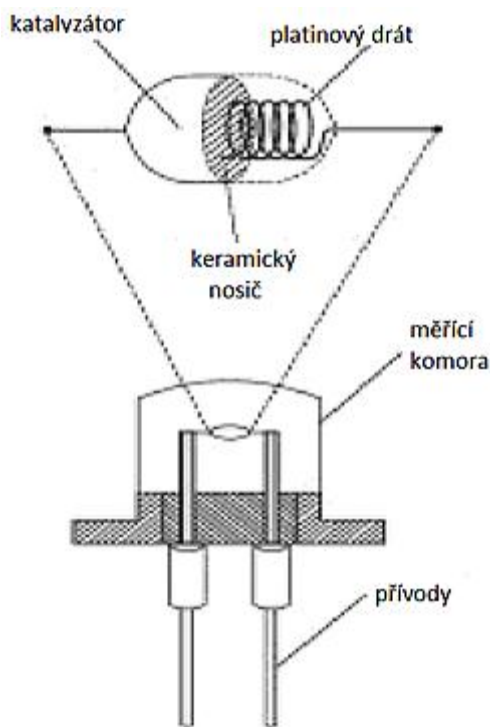
Obr. 7: Příklad senzoru pH - struktura ISFET a jeho převodní charakteristika [6]

### 1.3.2.5 Teplotní chemické senzory

Teplotní chemické senzory (pelistory) patří k nejstarším typům chemických senzorů, které slouží především k měření koncentrace hořlavých látek. Tyto senzory pracují na principu změny tepla uvolněného či absorbovaného při chemických reakcích s měřeným plynem. Základem tohoto senzoru je platinový drátek, který slouží jako



teploměr a topné tělísko zároveň. Tento vinutý platinový drátek je zapouzdřen uvnitř keramického tělíska na jehož povrch je nanesen katalyzátor (obr. 8). [10]



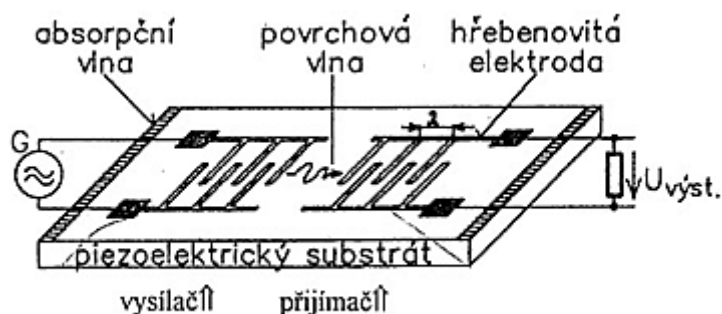
Obr. 8: Konstrukční uspořádání pelistoru [10]

Pro měření existují dva pracovní režimy, neizotermní a izotermní. U neizotermního za přítomnosti hořlavé látky probíhá na aktivním povrchu spalovací reakce, čímž roste teplota a mění se odpor platinového drátku. Měřící senzor pak bývá zhotoven ze dvou pelistorů, kde jeden je měřící (s aktivní vrstvou) a druhý referenční (bez aktivní vrstvy). S výhodou pak lze využít různých můstků k vyhodnocení výsledku (Wheatstoneův můstek apod.). Izotermním režimem se myslí takové zapojení, kterým je udržována stálá teplota aktivní vrstvy. Vyhodnocován je příkon, jež je potřeba k udržení této teploty, ten je i přímo úměrný koncentraci plynu. [6, 10]

### 1.3.2.6 Gravimetrické chemické senzory

Tyto senzory tvoří dva rezonátory interdigitální (hřebínkové) struktury, jež jsou propojeny aktivní vrstvou, která je chemicky citlivá. Po této vrstvě se zároveň šíří povrchová akustická vlna, která je vybuzena jedním z rezonátorů (obr. 9). Obvykle se jedná o ultrazvukové vlny, kde kmitočet může dosahovat jednotek GHz. Druhý rezonátor pak tuto vlnu přijímá a na základě jejího útlumu lze určit přítomnost určitého plynu. Pro tyto senzory se také používá označení SAW (surface acoustic wave). [10, 12]

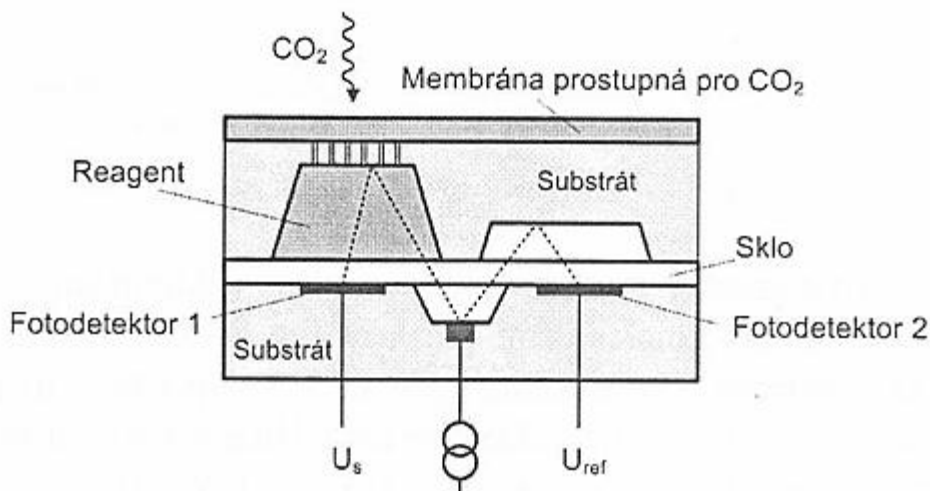




Obr. 9: Uspořádání senzoru SAW [11]

### 1.3.2.7 Optické chemické senzory

Optické chemické senzory pracují na principu porovnávání některých vlastností (polarizace, intenzita, atd.) dvou světelných paprsků vycházejících z jednoho světelného zdroje (LED dioda nebo laser). Kde jeden z paprsků prochází referenčním prostředím a je tedy časově stálý, zatímco druhý paprsek prochází látkou, která reaguje s detekovaným plynem (reagent) a tím ovlivňuje některé z vlastností procházejícího paprsku (obr. 10). Jako detektory světelných paprsků se využívá fotodiod nebo fototranzistorů. [6, 11]



Obr. 10: Konstrukční uspořádání optického chemického senzoru [6]

## 1.4 Monitorování vůně v potravinářství

V potravinářském průmyslu lze k posuzování kvality potravin použít nejen chuť, ale i čich. Správné a průběžné monitorování pachu je důležité již při skladování a v neposlední řadě i před samotnou konzumací potravin. Každým rokem je v Americe hospitalizováno na 325000 lidí v důsledku přiotrávení špatným jídlem a 5000 lidí podlehne otrávení úplně. Nebo například Američané promrhají každým rokem téměř

40% veškerých jedlých potravin, což je tak přijde zhruba na 240 miliard dolarů. Takové plýtvání je pak problémem, jelikož v některých částech světa lidé naopak trpí nedostatkem potravin. Jak již bylo zmíněno, důležité je monitorování stavu potravin při jejich skladování. Například při dozrávání ovoce je uvolňován ethen, který vede k dalším chemickým reakcím, jež dodají ovoci sladkou chuť. Ovšem pokud ovoce přezrává, uvolňuje vyšší množství ethenu, které při skladování ve větším množství vede k rychlejšímu dozrávání okolního ovoce, což má za následek lavinový efekt, kde veškeré ovoce začne rychle přezrávat až se zkaží.

Podobně je tomu i u masa, které uvolňuje aminy a jestliže není dále správně zpracováno, může to vést k jeho zkažení. Pomocí monitorování koncentrace těchto uvolněných látek lze tak předejít následnému znehodnocení potravin a případnému ohrožení zdraví člověka. [13, 14]

Elektronické nosy mohou jen těžko plně nahradit lidské smysly při určování jakosti potravin, jelikož lidský čich je s chutí úzce svázán. Dalším faktorem mohou být měnící se požadavky spotřebitelů potravin, tudíž lidské smysly jsou v tomto ohledu nepostradatelné. Na druhou stranu, ale lze elektronické nosy v potravinářském průmyslu využít v případech, kde je pevně nastavena hranice, která se nemění. Především jde o již zmíněné monitorování skladování potravin, které může vést i k záchraně lidských životů. Některá zařízení již byla pro podobné účely vyvinuta. Např. Peres, zařízení určené k identifikaci čerstvosti masa (obr. 11). Zařízení mohou využít např. kuchaři v restauracích a detailněji prozkoumat stav masa bez nutnosti jeho konzumace. Nebo zjistit přibližnou zbývající dobu trvanlivosti potraviny.



Obr. 11: Peres - Přenosný elektrický nos určující čerstvost masa [14]

### 1.4.1 Olfaktometrické měření

I přes to, že je lidský čich jen málo vyvinutým smyslem, je využíván při olfaktometrických měřeních, která často nacházejí uplatnění při měření emisí továren nebo u posuzování kvality potravin. Především jde o vyhodnocování kvality různých druhů masa, která musí splňovat neustále proměnlivé požadavky spotřebitele. Jedná se o měření, která provádí vybraná skupina trénovaných posuzovatelů vlastním čichem. Měření jsou velmi náročná na podmínky, při kterých se provádí. Posuzovatel musí být naprosto zdravý, nekuřák apod. Rovněž je opakovaně testován na správnou identifikaci evropské referenční pachové látky n-butanolu v určité koncentraci, dle definice. Optimálně tvoří skupinu šest až osm posuzovatelů a měření jsou porovnávány s ostatními laboratořemi, přičemž jejich výsledky se nesmí od sebe lišit o více jak 5%. Celkový výsledek měření je vypočten jako průměr celé skupiny posuzovatelů. Těmito přísnými podmínkami je zajištěna dostatečná reprodukovatelnost, srovnatelná s ostatními metodami měření pachů. Olfaktometr je zobrazen na obr. 12.

Pro vyjádření výsledků měření se používá evropská pachová jednotka  $ou_E$  (odour unit) dle definice:

"Evropskou pachovou jednotkou se rozumí množství odorantu, které, pokud je rozptýleno v  $1\text{ m}^3$  neutrálního plynu za standardních podmínek, vyvolá fyziologickou reakci respondentů odpovídající evropské referenční pachové jednotce (*EROM*).

Evropskou referenční pachovou jednotkou se rozumí fyziologická reakce respondentů vyvolaná dávkou 123 mg n-butanolu rozptýleného v  $1\text{ m}^3$  neutrálního plynu za standardních podmínek. To je množství, které odpovídá 0,040 mmol/mol n-butanolu." [15]



Obr. 12: Ukázka olfaktometru [16]

## 2 VÝVOJOVÁ PLATFORMA ARDUINO

Arduino je vývojová platforma, která je založena především na mikrokontrolérech ATmega od společnosti Atmel. Jedná se o kompletní elektronickou stavebnici ve formátu malého plošného spoje, ke kterému je možné připojovat další elektroniku. [17]

Výhodnou Arduina je jeho otevřenost a podrobná dokumentace, kde plošné spoje i vývojové prostředí jsou dostupné pod svobodnou open-source licenci (Creative Commons). Vzniká tak několik vylepšení a různých variací vývojového kitu, mezi které patří i tzv. "štítý" (shield). Jedná se o kompatibilní moduly, realizované jako plošné spoje podobných rozměrů, které lze k Arduinu snadno připojit (např. relé, motor, ethernet, GSM nebo LCD štítý) a rozšířit tak jeho možnosti. Tímto způsobem lze tvořit různé roboty nebo i dokonce 3D tiskárny. Další výhodou této platformy je předprogramovaný bootloader tj. kód, který po spuštění provede nejzákladnější nastavení mikrokontroléru. Uživatelé se usnadňuje práce, protože se o inicializaci mikrokontroléru nemusí starat. Na desce Arduino se nachází i několik dalších obvodů jako jsou stabilizátory napětí nebo oscilátor. Nejdůležitějším je ale převodník, zajišťující komunikaci s PC přes USB standart na místo dnes, méně používaného RS-232 portu. Odpadá tak nutnost použití externího programátoru. [18, 19]

Popis několika verzí vývojových desek Arduino je zobrazen v tab. 1. Při návrhu této aplikace byla použita verze *Arduino Mega* s mikrokontrolérem ATmega1280, která je v současnosti již zastaralá, byla nahrazena novější verzí *Arduino Mega 2560* s mikrokontrolérem ATmega2560, který má především dvojnásobnou velikost programové paměti. Počet vstupně-výstupních portů zůstal stejný, stejně tak i pracovní kmitočet.

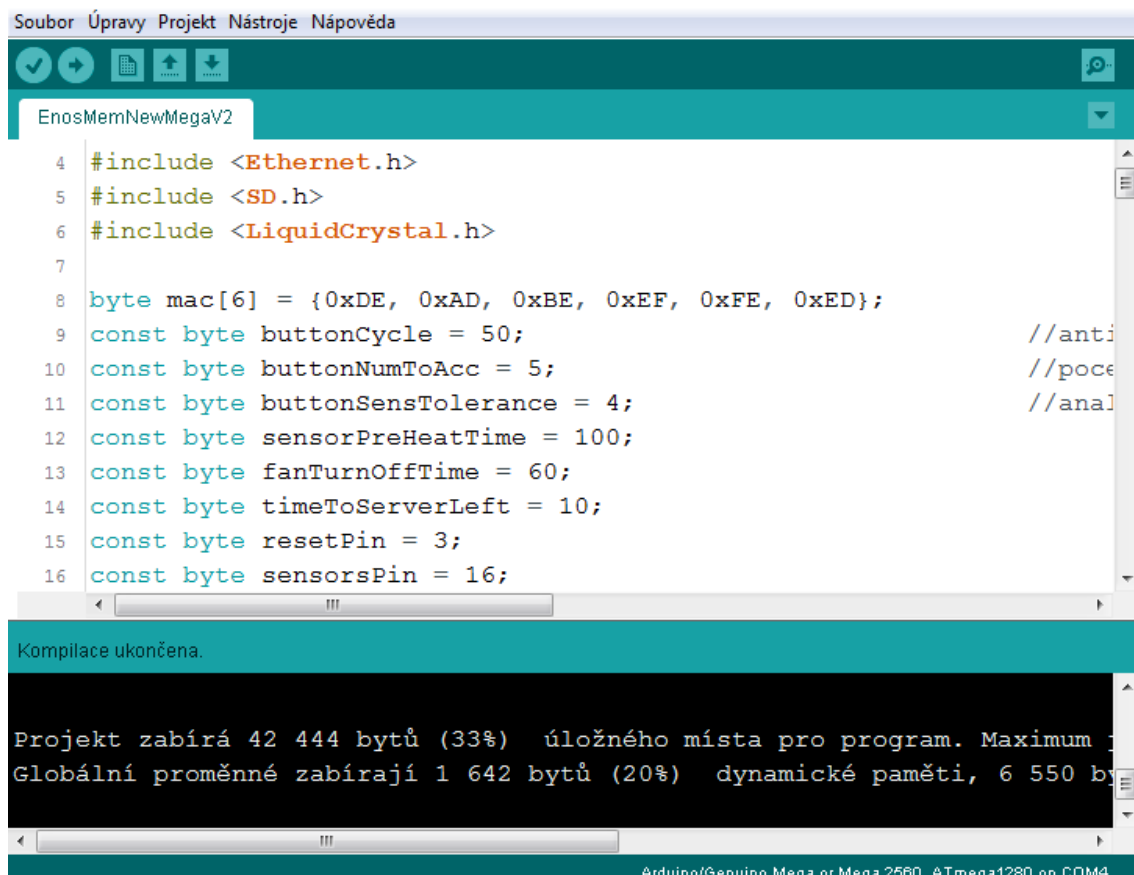
Tab. 1: Příklady vývojových platforem Arduino

Označení	Mikrokontrolér		Paměť		Vstupně-výstupní porty	
	Typ	Kmitočet	Data	Program	Digitální	Analogový
Nano	ATmega328	16 MHz	1 kB	16 kB	14	8
LilyPad	ATmega168V	8 MHz	1 kB	16 kB	14	6
Mega2560	ATmega2560	16 MHz	8 kB	256 kB	54	16
Uno	ATmega328P	16 MHz	2 kB	32 kB	14	6
Leonardo	Atmega32U4	16 MHz	2,5 kB	32 kB	20	12
Due	ATSAM3X8E	84 MHz	96 kB	512 kB	54	12
Zero	ATSAMD21G18A	48 MHz	32 kB	256 kB	14	6
101	Intel® Curie	32 MHz	24 kB	384 kB	14	6

## 2.1 Vývojové prostředí Arduina

Samotné programy pro Arduino lze psát všemi možnými způsoby, jednak klasickou metodou, pomocí originálních nástrojů jako je např. AVR Studio a to v jazyce C nebo Assembleru. Další, jednodušší možností je psaní programů v jazyce Arduino Programmable Language, který je pro programování této platformy záměrně vytvořen. Použitý jazyk je založen na jazyce Wiring, který je vzniklý z jazyka C a uživatelsky se jeví jako zjednodušený jazyk C.

Programy pro Arduino se proto nejčastěji píšou v jednoduchém vývojovém prostředí Arduino IDE (Integrated Development Environment), které je přizpůsobeno i začátečníkům (obr. 13). Velkou výhodou tohoto vývojového prostředí je jeho kompatibilita s ostatními platformami, což je způsobeno napsáním aplikace v jazyce Java. Tento vývojový prostředek se skládá z jednoduchého textového editoru a lišty se základním nastavením, kde uživatel volí především typ Arduina, které bude programovat a port na kterém je připojeno. Na základě toho program zvolí vhodný bootloader a všeobecné nastavení specifické pro danou desku. Dále má uživatel možnost zobrazit okno sériové komunikace mezi Arduinem a počítačem. Zde může zobrazit příchozí data z Arduina. Rovněž lze i data do Arduina podobnou cestou odesílat. Toto monitorování slouží jako základní nástroj pro odladění programu. [20]



```
4 #include <Ethernet.h>
5 #include <SD.h>
6 #include <LiquidCrystal.h>
7
8 byte mac[6] = {0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED};
9 const byte buttonCycle = 50; //anti
10 const byte buttonNumToAcc = 5; //pocet
11 const byte buttonSensTolerance = 4; //anal
12 const byte sensorPreHeatTime = 100;
13 const byte fanTurnOffTime = 60;
14 const byte timeToServerLeft = 10;
15 const byte resetPin = 3;
16 const byte sensorsPin = 16;
```

Kompilace ukončena.

Projekt zabírá 42 444 bytů (33%) úložného místa pro program. Maximum Globální proměnné zabírají 1 642 bytů (20%) dynamické paměti, 6 550 bytů

Arduino/Genuino Mega or Mega 2560, ATmega1280 on COM4

Obr. 13: Vývojové prostředí Arduina

Samotné programování v tomto prostředí je velmi jednoduché. Každý kód se skládá ze dvou základních funkcí, kterými jsou *setup* a *loop*. První slouží obecně k inicializaci, spouští se totiž jen jednou při startu mikrokontroléru. Zatímco do funkce *loop* se obvykle píše většina programu, neboť se jedná o nekonečnou smyčku a kód se zde neustále opakuje.

Jednoduchý program proto obsahuje jen málo řádků a může vypadat např. takto:

```
void setup() {                               //funkce setup - provádí se jednou po
                                             spuštění Arduina
  pinMode(13, OUTPUT);                       //nastavení pinu 13 jako výstupního
}
void loop() {                                //funkce loop - nekonečná smyčka, příkazy
                                             se neustále provádějí
  digitalWrite(13, HIGH);                   //nastavení pinu 13 na log. 1
  delay(1000);                              //čekání 1000ms
  digitalWrite(13, LOW);                    //nastavení pinu 13 na log. 0
  delay(1000);                              //čekání 1000ms
}
```

## 2.2 Hardwarové uspořádání

### 2.2.1 AVR

AVR především označuje architekturu osmibitových jednočipových mikropočítačů norské firmy Atmel. Tyto mikrokontroléry jsou založeny na harvardské architektuře s redukovanou instrukční sadou, která byla ovšem vyvíjena k použití s vyššími programovacími jazyky, jako je jazyk C. Obsahují 32 pracovních osmibitových registrů do kterých lze zapsat jak data, tak i adresy. Aritmeticko-logická jednotka je přímo propojena s těmito registry, což jí umožňuje zpracovávat instrukce v jediném hodinovém cyklu. Dnes se vyrábějí tři základní řady mikrokontrolérů a to s označením ATtiny, ATmega a ATXmega. [21]

### 2.2.2 ATmega

Jedná se o výkonné mikrokontroléry, které jsou považovány za jedny z nejznámějších a nejrozšířenějších od firmy Atmel. Jsou univerzálními, snadno dostupnými mikrokontroléry s dobrým poměrem výkonu a ceny. Zároveň se lépe programují a to díky rozhraní JTAG pomocí kterého lze provádět tzv. "on-chip debugging". Tato funkce umožňuje odladění a krokování jednotlivých instrukcí kódu přímo za běhu programu na reálném čipu. Oproti ATtiny mají tyto mikrokontroléry i větší paměť flash a RAM, zároveň disponují větším množstvím instrukcí a integrovaných periférií. Výpočetní výkon těchto mikrokontrolérů může dosahovat až několik desítek miliónů instrukcí za sekundu. [22]

### 2.2.3 Čítač / časovač

Arduino, resp. jeho mikrokontrolér obsahuje tzv. časovače a čítače. Jedná se o jeho samostatné bloky, které jsou naprosto nezávislé na programu a můžou tak pracovat souběžně s ním. Čítač obecně reaguje na impulzy z vnějšího zdroje, které neustále počítá, tento počet se ukládá do definovaného registru se kterým lze dále pracovat. Zatímco časovač pracuje již s nastaveným hodinovým signálem a počítá jejich impulzy, které jsou opět uloženy v určitém registru. Kmitočet tohoto signálu lze měnit pomocí nastavovacích registrů, které zvolí příslušnou děličku kmitočtu. [23]

### 2.2.4 Přerušení

Přerušením lze asynchronně obsloužit určité události, které nastanou v průběhu zpracovávání instrukcí v hlavním těle programu. Po výskytu události je tedy běh hlavního programu pozastaven a veškerá data, se kterými program aktuálně pracuje uložena do příslušných registrů a to proto, aby bylo možné se k rozpracovaným výpočtům po obsluze přerušení opět vrátit a správně pokračovat v pozastavených výpočtech. Obsluhou přerušení jsou tedy instrukce, které se ihned provedou po spuštění dané události. Těmito událostmi mohou být vnější nebo vnitřní impulzy, kterým může být např. přetečení časovače. [24]

### 2.2.5 Pulzně šířková modulace

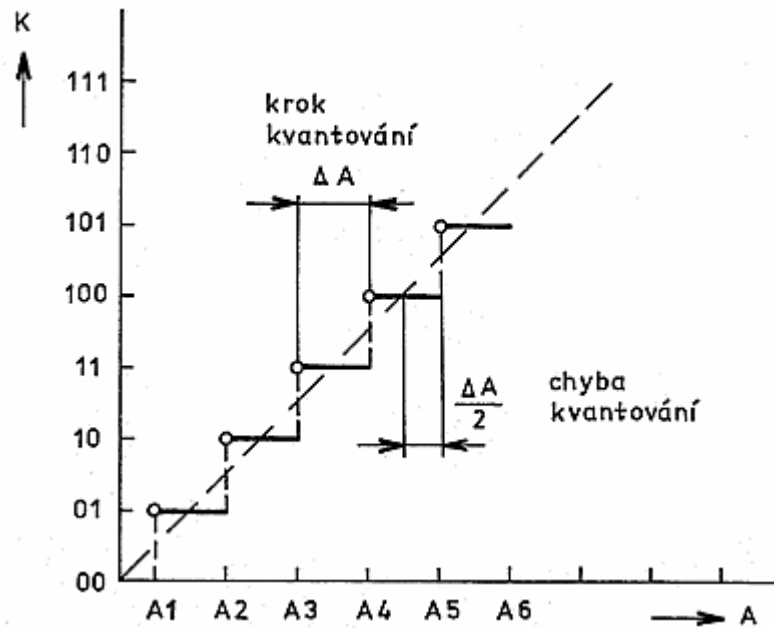
Pulzně šířková modulace (PWM - Pulse Width Modulation) je způsob generování pulzního signálu o daném kmitočtu s proměnnou střídou. Tento signál tedy může nabývat pouze hodnot log. 0 a log. 1, obvykle tedy nulového napětí a napětí napájecího. Kmitočet tohoto signálu zůstává zachován, ovšem změna střídy dvou použitých úrovní mění hodnotu tohoto signálu. Tento digitální signál lze snadno převést na analogový s malými zámkami pomocí RC filtru. Těchto signálů pak lze využít pro spínání a regulaci intenzity či rychlosti různých periférií. [25]

### 2.2.6 A/D převodník

Analogově digitální převodník je elektronická součástka, která slouží k převodu analogového signálu na digitální. A to z toho důvodu, aby mohl mikrokontrolér se signálem dále pracovat. Tyto analogové signály lze nalézt u většiny senzorů, kde se převádí měřená fyzikální veličina na veličinu elektrickou, např. napětí.

Při převodu je nejdříve provedeno vzorkování spojitého analogového signálu, kdy se v určitých okamžicích měří aktuální amplituda. Následně jsou tyto hodnoty kvantováním převedeny na digitální tvar, tedy zaokrouhleny na hodnotu, která odpovídá nejbližší kvantovací úrovni, čímž se vytvoří výstupní datové slovo požadované hodnoty (obr. 14). Ty nabývají hodnot buď od 0 V do  $U_{\max\text{ref}}$  (maximálního referenčního napětí, které lze obvykle nastavit) u unipolárních převodníků a nebo od  $-U_{\min\text{ref}}$  do  $U_{\max\text{ref}}$

u bipolárních. Velikost napětí mezi dvěma sousedícími úrovněmi se nazývá kvantovací krok, který pak odpovídá nejméně významnému bitu výstupu A/D převodníku.



Obr. 14: Převodní charakteristika A/D převodníku [26]

Mezi základní parametry patří rychlost vzorkování analogového signálu, která je dána Shanon-Kotelníkovým vzorkovacím teorémem, který říká, že nemá li docházet ke zkreslení při převodu analogového signálu na číslicový, pak frekvence vzorkování musí být větší než dvojnásobek největší frekvence spektra analogového signálu. Dále rychlost převodu, která bývá shodná s rychlostí vzorkování. Např. 200 převodů/s zvládne převést spektrum o frekvenci 100 Hz. Dále rozlišovací schopnost, která je určena počtem diskretních úrovní výstupního datového slova. [26]

Nejjednodušším typem je paralelní A/D převodník, který je i nejrychlejší, při převodu používá soustavu  $m$  napěťových komparátorů. Pro  $n$ -bitový výstup je potom  $m=2^n-1$ . Větší rozlišovací schopnosti jsou pak poměrně nákladné, na druhou stranu ale rychlost převodu je vysoká, dosahuje tak až jednotek GHz. Tyto převodníky lze skládat do kaskády a snížit tak výrobní náklady za cenu zhoršení parametrů.

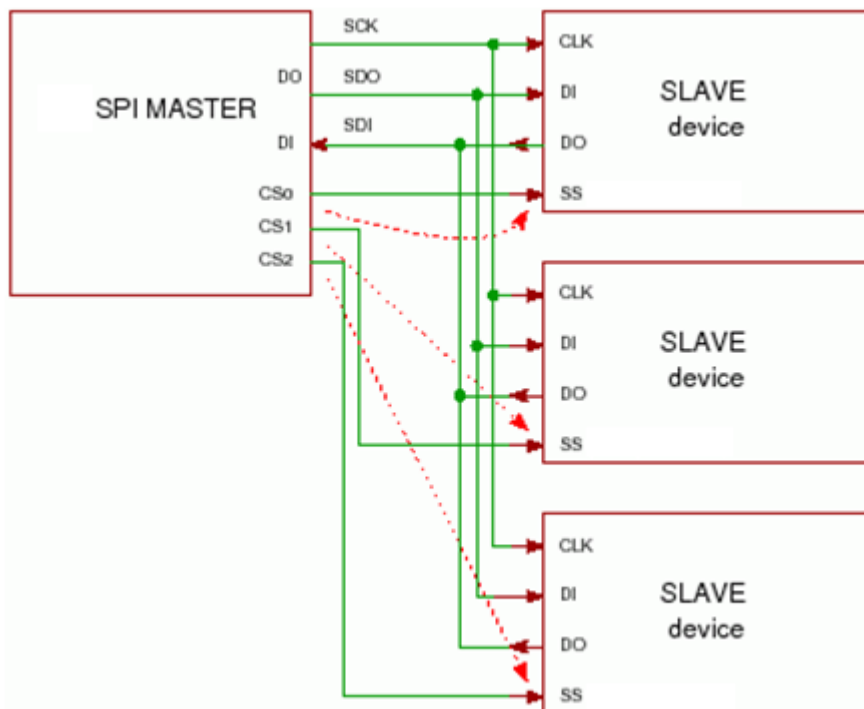
### 2.2.7 Serial Peripheral Interface

Serial Peripheral Interface (SPI) je sériová sběrnice, která je v současnosti hojně používaná, především u mikrokontrolérů pro jejich snadnou komunikaci s ostatními perifériemi. Jedná se o sériový synchronní přenos dat mezi několika zařízeními, obvykle pouze na krátké vzdálenosti. Na tuto sběrnici může být připojeno hned několik periférií, které spolu mohou komunikovat. Ze všech připojených zařízení je právě jedno připojeno jako "master", zatímco všechna ostatní jako "slave". "Mastrem" bývá obvykle



právě mikrokontrolér, řídící veškerou komunikaci mezi ostatními periferiemi, kterými mohou být různé senzory, převodníky nebo i jiný mikrokontrolér.

Sběrnice je tvořena čtyřmi vodiči a to obvykle značenými SCK (hodinový signál), MISO (Master In, Slave Out), MOSI (Master Out, Slave In), SSEL (Slave Select). Nejdříve je tedy potřeba jednotlivá zařízení správně propojit (obr. 15) a nastavit příslušný režim v kterém budou pracovat, resp. zvolit zařízení pracující jako "master", což je obvykle provedeno nastavením příslušného registru daného zařízení. V těchto registrech lze obvykle nastavovat i další parametry např. zda se bude reagovat na sestupnou či náběžnou hranu hodinového signálu apod. Samotná komunikace je pak velmi jednoduchá, prakticky se totiž jedná o soustavu synchronních posuvných registrů, které si navzájem vyměňují data, kde hodinový signál generuje zařízení označeno jako "master". V případě paralelního zapojení více zařízení na sběrnici je potřeba zajistit přepínání mezi jednotlivými zařízeními označenými jako "slave". To se provede nastavením pinu SSEL příslušného zařízení. Další variantou je zapojení všech zařízení do série, kde je vždy výstup zařízení připojen do vstupu toho následujícího, tak aby tvořily kruh. V takovém zapojení musí být všechna zařízení aktivní nastavením příslušné hodnoty pinu SSEL. Zároveň je také řízení tohoto přenosu složitější a přenos pomalejší. [27]



Obr. 15: Zapojení pro komunikaci s více zařízeními přes SPI sběrnici [27]

## 2.3 Ethernet SHIELD

Pomocí Ethernet SHIELDU je možné připojit Arduino do sítě sdílené s počítači a to jak přímo do internetu tak třeba jen do lokální sítě. Tento modul totiž obsahuje Ethernetový kontroler Wiznet W5100, jež je srdcem tohoto modulu a tak lze s Arduinem pracovat jako s každým jiným síťovým zařízením. Rychlost síťové komunikace 10/100 Mbit/s je pro většinu aplikací dostačující. Pracuje se základními síťovými protokoly jako TCP, UDP, ICMP, IPv4, ARP, IGMP, PPPoE. Zvládne také současně řídit až čtyři spojení. Tato deska obsahuje také slot pro micro-SD paměťové karty, se kterými Arduino umí pracovat. Zvládá řízení základního souborového systému FAT32, jehož možnosti jsou opět pro většinu aplikací více než dostačující. Moduly spolu komunikují po SPI sběrnici. Samozřejmostí jsou veřejně dostupné knihovny (*Ethernet.h*) s dokumentací k programovému ovládní tohoto modulu včetně paměťového slotu (*SD.h*). [18, 28]

### 2.3.1 Síťová komunikace

Počítačovou sítí se rozumí spojení několika zařízení, které jsou schopny se sebou komunikovat a sdílet svoje prostředky. Komunikaci mezi těmito zařízeními lze pak rozdělit do tří základních částí a to: otevření přenosu, přenos a uzavření přenosu. Proto byl v roce 1984 organizací ISO vytvořen tzv. OSI referenční model, jež se stal standardem pro komunikaci veškerých síťových zařízení. Tento model definuje celkem sedm vrstev, kde každá má při komunikaci důležitou roli. Kde platí, že vrstvy na jednom zařízení mohou komunikovat vždy jen se sousedící vrstvou (obr. 16). Postupně, od nejvyšší (od softwarové po hardwarovou tj. počínaje aplikací a ukončeno bity na přenosovém médiu), jsou tyto vrstvy definovány následovně:

**Aplikační vrstva:** Jedná se o softwarovou vrstvu, která tvoří hranici mezi aplikacemi a samotnou komunikací. Zajišťuje tak přímou spolupráci se síťovými aplikacemi. Patří sem například protokoly HTTP, FTP apod.

**Prezentační vrstva:** Softwarová vrstva, která upravuje formátování dat ve smyslu jejich kódování, šifrování nebo kompresi. Zajišťuje jejich stejný význam na obou stranách komunikace.

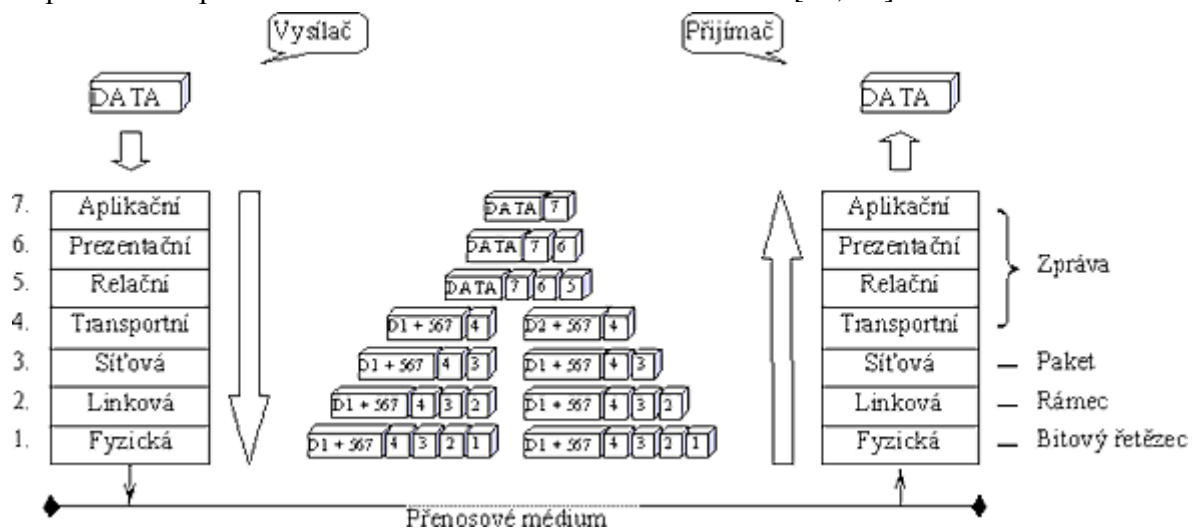
**Relační vrstva:** Softwarově řídí relace. Tedy stará se o synchronizace a obsluhu spojení jednotlivých aplikací formou stavových zpráv.

**Transportní vrstva:** Poskytuje softwarové zabezpečení přenosu (kontrola chyb) a správné rozdělení dat do segmentů. Rozlišují se zde dva hlavní protokoly a to TCP, který je spojovaný a zaručuje tak, že veškerá data dorazí do cíle bez porušení ve správném pořadí, na rozdíl od protokolu UDP, který je nespojovaný a není zde tak zaručeno správné přijetí všech dat v cílové stanici.

**Síťová vrstva:** Zajišťuje přenos dat na síti zvolením nejvhodnějšího cílového a zdrojového zařízení, stará se tedy o směrování paketů a jejich adresaci. V současnosti nejrozšířenějším protokolem je IPv4.

**Linková vrstva:** Hardwarová vrstva, zabezpečující přenos jednotlivých bloků dat (rámců) mezi přímo propojenými (sousedícími) zařízeními. Zároveň přidává data o kontrolním součtu (CRC), čímž je zabezpečuje proti chybám.

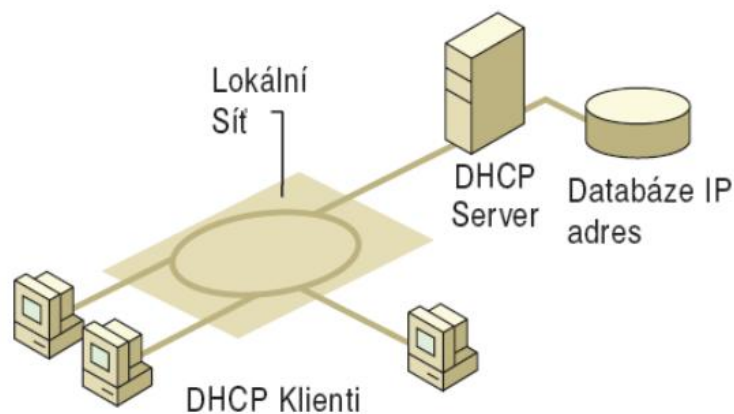
**Fyzická vrstva:** Obstarává převedení dat z vyšších vrstev do formy jednotlivých bitů, které jsou následně vyslány po daném přenosovém médiu (kabelu 10BaseT apod.). Např. určité napětěvé úrovně s danou modulací u Ethernetu. [29, 30]



Obr. 16: Grafické znázornění průchodu dat v OSI modelu mezi vysílací a cílovou stanicí [29]

### 2.3.2 Protokol DHCP

Protokol DHCP (Dynamic Host Configuration Protocol) je založen na modelu server-klient, využívající ke komunikaci spojení UDP s portovými čísly 67 a 68 pro server a klienta. Tento protokol zajišťuje automatickou správu síťové konfigurace stanic v dané počítačové síti (obr. 17). Jedná se především o IP adresu stanice, masku sítě, výchozí bránu a adresu serveru DNS. Tyto informace jsou nutné ke správnému připojení a fungování stanice v nové počítačové síti, díky tomuto protokolu se nově připojující uživatel (klient) nemusí starat o svá síťová nastavení, jelikož ve výchozím nastavení je obvykle DHCP protokol povolen. [31, 32]



Obr. 17: Zobrazení základní struktury počítačové sítě s DHCP serverem [31]

Veškerá síťová nastavení jednotlivých stanic jsou tak uložena na DHCP serveru dané sítě, kde mohou být spravovány administrátorem sítě. Při konfiguraci DHCP serveru lze rozlišit dva režimy přidělování adres - dynamickou a statickou alokaci.

Při dynamické alokaci je na DHCP serveru nastaven pracovní rozsah adres, které budou přidělovány novým stanicím. Zároveň je nastaven časový limit platnosti (pronájem) adres, což umožní uvolnit již nepoužívané adresy pro jiná, nová zařízení a smazat záznamy těch starých. Jestliže ale dojde k vypršení platnosti adresy zařízení, které je stále k síti připojeno, dojde pouze k obnovení časového limitu záznamu a adresa stanice zůstává stejná.

Statická alokace adres představuje přímé přiřazení logických adres k fyzickým MAC adresám. Správce tak sestavuje tabulku IP adres a k nim odpovídající zařízení, označené MAC adresou. Následně pouze zařízení v této tabulce obdrží od DHCP serveru požadovanou adresu. Statická alokace adres je administrativně náročnější, za to ale může být bezpečnější, jelikož správce má vždy kontrolu nad všemi zařízeními.

V některých případech lze rozlišit i hybridní model, který je kombinací statické a dynamické alokace, kde statická alokace má vyšší prioritu. Tedy jestliže již správce nastavil dané MAC adrese její IP adresu, je upřednostněna a klientovi nabídnuta. V případě, že se MAC adresa zařízení ve statické tabulce nenachází, je mu přiřazena automaticky z pracovního rozsahu dynamických adres. [31, 33]

Ke komunikaci používá DHCP protokol předdefinované zprávy, které mohou být vysílány pouze prostřednictvím druhé (linkové) vrstvy OSI modelu (tab. 2). Díky tomu se zprávy mohou šířit pouze v dané broadcastové (všesměrové) doméně (segment sítě, který je oddělen zařízeními vyšších vrstev např. směrovači).

Základní komunikace pak vypadá následovně:

- 1.) Klient po připojení do sítě vyšle všesměrovou zprávu o žádost konfigurace prostřednictvím zprávy DHCPDiscover.
- 2.) Server odpoví na žádost klienta nabídkou volné adresy zprávou DHCPOffer.
- 3.) Klient přijme jednu z nabízených adres (např. v případě většího počtu serverů) a vyšle všesměrovou zprávu DHCPRequest. Stejnou zprávu klient vysílá i v případě vypršení platnosti jeho adresy, tedy při žádosti o prodloužení doby zapůjčení.
- 4.) Server, který adresu nabízel vyšle zprávu DHCPAck, potvrzující platnost klientovy konfigurace. [31, 32]

Tab. 2: Zprávy používané při komunikaci protokolu DHCP [31]

Zpráva	Popis
DHCPDiscover	Všesměrová žádost klienta o přidělení IP adresy.
DHCPOffer	Nabídka volné adresy od DHCP serveru.
DHCPRequest	Potvrzení nabídky adresy klientem.
DHCPAck	DHCP server potvrzuje doručení předchozí zprávy a přidává zbytek konfigurace.
DHCPSNak	Oznámení o neplatné adrese ze strany DHCP serveru.
DHCPDecline	Zamítnutí platnosti adresy obdržené od DHCP serveru.
DHCPRelease	Zrušení zápůjčky adresy klienta.

V případě, že se je třeba zprávy protokolu DHCP posílat mimo tuto doménu (např. server DHCP se nachází za směrovačem), musí být na hraničním směrovači nakonfigurováno předávání těchto zpráv tzv. DHCP relay.

### 2.3.3 Protokol HTTP verze 1.1

HTTP protokol verze 1.1, který je dnes běžně používán a dostatečně zoptimalizován definuje komunikaci mezi klientem (webovým prohlížečem) a serverem (webovou stránkou). Komunikace probíhá formou stavových zpráv tak, že klient se spojí se serverem a pošle mu požadavek na danou stránku. Server v zápětí odesílá odpověď s hlavičkou se stavovou informací a požadovaným dokumentem, přičemž spojení zůstává navázáno, dokud jedna ze stran spojení neuzavře patřičnou odpověď. To je výhodné zejména pro složitější stránky, kde je třeba odesílat více dat z rozdílných míst (např. obrázky), snižuje se tak doba odezvy. Parametry komunikace nebo dokumentu mohou být v průběhu různě měněny a to v hlavičce, kterou vyšle server společně s požadovaným dokumentem. Klient ve své hlavičce může také posílat doplňující informace o svých možnostech na které může server patřičně reagovat. Přehled základních stavových kódů a požadavků je uveden v tab. 3. [34]

Nejjednodušší příklad komunikace se základními stavovými zprávami:

**Klient posílá požadavek serveru:**

```
GET /mereni/graf.html HTTP/1.1
HOST: 192.168.2.128
prázdný řádek, značící konec požadavku (hlavičky)
```

**Odpověď serveru na požadavek:**

```
HTTP/1.1 200 OK
Content-type: text/html
prázdný řádek, značící konec odpovědi (hlavičky)
samotný html dokument
```

Tab. 3: Základní požadavky, stavová hlášení a hlavičky [35]

Metody požadavku	
GET	žádost o stránku
POST	odeslání dat z formuláře
HEAD	žádost pouze o hlavičku
PUT	uložení objektu
DELETE	smazání objektu
Stavové kódy	
100 - 199	informativní kód
200 - 299	úspěšné vyřízení požadavku
300 - 399	přesměrování
400 - 499	chyba klienta
500 - 599	chyba na straně serveru
Pole v hlavičce	
Date	Datum a čas požadavku/odpovědi
Content-Type	Typ odesílaných dat
Host	Adresa serveru
Accept-Charset	Kódování znaků klienta
If-Modified-Since	Datem podmíněné načtení stránky
User-Agent	Identifikace klienta
Server	Identifikace serveru
Refresh	Doba, mezi automatickým obnovením stránky

### 2.3.4 Externí mikroSD FLASH paměť

Arduino ethernet shield rovněž disponuje slotem pro připojení externí microSD paměťové karty a umožňuje tak rozšířit paměť dat. Komunikace s pamětí a mikrokontrolérem je provedena pomocí SPI sběrnice. Arduino tak může pracovat se základními souborovými systémy jako jsou FAT16 a FAT32.

Funkčnost těchto souborových systémů je založena na dvou kopiích alokační tabulky souborů, jež jsou uloženy na začátku média. V nichž jsou uloženy informace

o jednotlivých clusterech (nejmenší logická datová jednotka paměti, alokační jednotka). Jsou zde uloženy odkazy na pokračování či konce daného souboru nebo informace zda je dané místo volné nebo poškozené. Souborový systém FAT16 je v současné době již zastaralý a je určen především pro kapacitně malá paměťová média. Maximální velikost úložiště je pak dána počty bitů jenž jsou určeny na adresaci clusterů a velikostí samotného clusteru. Systémy FAT16 používají 16bitů na adresaci clusterů při jejich maximální velikosti 64 KB, čímž lze dosáhnout maximální velikosti média 4 GB. Zatímco FAT32 zvládne vytvořit souborový systém o velikosti až 2 TB. Původní verze těchto souborových systémů podporovali pouze 8.3 formát názvů souborů, tedy osm znaků před příponou a tři za příponou. Později byl tento nedostatek rozšířen tak aby byla zachována zpětná kompatibilita. [36]

Princip funkce externího paměťového média (microSD karty), paměti typu FLASH je prakticky stejný jako u paměti typu EEPROM, kde je možné elektricky programovat a mazat paměť, vždy ovšem jako celek. Oproti tomu FLASH paměti lze programovat a mazat po blocích, do nichž je celá paměť rozdělena. To nejen umožňuje mazání a programování pouze určitých částí paměti ale i zvyšuje její životnost, jelikož řadič paměti pak může jednotlivé zápisy rozdělit do celé paměti a nepřepisovat tak stále stejné buňky. Životnost paměťové buňky je totiž dána množstvím jejích přepisů a klesá s každým dalším přepisem.

Každá buňka je tvořena unipolárním tranzistorem se dvěma hradly, které jsou těsně nad sebou a izolována oxidovou vrstvou. Horní hradlo je řídicí, zatímco spodní je volné, plovoucí, na kterém se v případě programování pomocí řídicího hradla protuneluje náboj, čímž je zajištěno sepnutí tranzistoru, výstupem je pak log. 0. Zatímco mazání lze provést opačnou operací, tedy přivedením záporného řídicího napětí, čímž je náboj z plovoucího hradla odveden a tranzistor uzavřen, výstupem je pak log. 1. [37]

## 2.4 LCD SHIELD

Modul LCD SHIELD obsahuje dvouřádkový, alfanumerický, podsvícený displej o délce 16 znaků na řádek (obr. 18). Displej je řízen řadičem HD44780, který komunikuje s Arduinem ve 4-bitovém režimu na datových digitálních portech 4 (D0), 5 (D1), 6 (D2) a 7 (D3), dále porty 8 (RS) a 9 (EN) jsou řídicí. Arduino je plně kompatibilní s tímto řadičem, což umožnilo použít vestavěnou knihovnu *LiquidCrystal.h*. Stejně jako ostatní moduly, je i tento napájen přímo z +5 V sítě Arduina. Komunikace s řadičem probíhá formou jednoduchých, předdefinovaných instrukcí, kterými je nutné nejprve provést inicializaci displeje nastavením příslušného režimu (4-bitový) nebo pozici kurzoru, data jsou následně odesílána ve formě standardizovaného ASCII kódu.

Modul LCD SHIELD je osazen i pěti tlačítky, kterými lze jednoduše zařízení ovládat. Všechna tlačítka jsou připojena na vstupní analogový port A0 a zapojena jako různé děliče napětí. Dle úrovně napětí na vstupu A/D převodníku lze detekovat příslušné tlačítko. Nevýhodou této konfigurace je nemožnost detekce více současných stisků. [18]



Obr. 18: LCD Modul



### 3 HYPER TEXT MARKUP LANGUAGE

Hyper Text Markup Language (HTML) je značkovací jazyk, který je základem při tvorbě webových stránek. Slouží k přesnému a stručnému vyjádření toho, co má cílová uživatelská aplikace, zvaná jako prohlížeč, zobrazit. Tento jazyk je interpretován přímo v textovém formátu, nikdy neprobíhá jeho kompilace do jiné podoby. Syntaxi jazyka definuje standard HTML, který tím definuje množinu značek (tagů) a jejich atributy. Původní syntaxe vychází z jazyka SGML (Standard Generalized Markup Language), která byla postupem času upravována podle aktuálních trendů internetu. [38]

První definice HTML jazyka byla vytvořena v roce 1991, která byla součástí projektu WWW. Ten měl umožnit vědcům z CERNu (Evropské centrum jaderného výzkumu) sdílení výsledků výzkumu po celém světě, zde celý projekt vznikl. Za vývojem těchto základů jazyka HTML (verze 0.9 - 1.2) stojí Tim Berners-Lee a Daniel Connolly. Postupem času byl tento jazyk zdokonalován, v listopadu 1995 organizace IETF (Internet Engineering Task Force) vydala verzi 2.0, která přidává především práce s formuláři. Další zásadní vylepšení přišlo v lednu 1997 od organizace W3C (World Wide Web Consortium), která vydala verzi 3.2, jenž přidala možnost vkládání tabulek, zarovnávání textu, podporu Java appletů a definování vzhledových stylů dokumentů. Další vylepšení přišlo opět od organizace W3C v prosinci 1997, kde byly vylepšeny prvky tvorby tabulek a formulářů, přibýly rámy, skripty a vkládání objektů. Po dvou letech byla vydána verze 4.01, která jen opravuje některé chyby předchozí verze, poté se vývoj HTML na dlouhou dobu pozastavil. Následovníkem HTML měl být totiž jazyk XHTML, který měl být založen na univerzálním jazyku XML. Tento směr vývoje se ovšem nezamlouval předním tvůrcům webových prohlížečů, tím byl v roce 2010 jazyk XHTML zavržen. V říjnu 2014 byla vydána verze 5.0, která je naprosto nezávislá na SGML, přidává další užitečné funkce, ty zastaralé odstraňuje a opravuje spoustu chyb předešlé verze. Verze HTML 5.0 je v současnosti aktuální verzí tohoto jazyka. [39]

HTML soubor, který popisuje vzhled a obsah dané stránky je pouhým textem a lze jej tedy tvořit v libovolném textovém editoru. Při tvorbě je jen potřeba dodržovat pravidla určená HTML standardem, ten popisuje obsah pomocí značek (tagů) a jejich atributů, které následují za jejich názvem. Tyto tagy jsou reprezentovány příslušným názvem, jenž je uzavřen do závorek ("**název** atribut"), lze je rozdělit na párové a nepárové tagy. Párové jsou definovány dvěma značkami, počátečním a ukončovacím, mezi kterými je samotný obsah, ten je patřičným způsobem tímto tagem ovlivněn. Zatímco nepárový tag definuje pouze jeden element, jehož parametry jsou definovány přímo v něm, v jeho attributech. [40]

Příklad HTML kódu:

```
<b>Tučný text v párovém tagu</b>  
<hr>  
<br>  
<u>Podtržený text pod čarou</u>
```

### 3.1 Kaskádové styly

Kaskádové styly CSS (Cascade Style Sheet) umožňují jednoduchým způsobem definovat jednotný vzhled webových stránek a oddělit jej tím od samotného obsahu stránek. Nejčastěji se jedná o samostatné externí soubory se kterými je HTML dokument provázán. Syntaxe je zároveň odlišná od HTML, skládá se ze selektoru, který určuje element stránky, jenž bude modifikován příslušnými pravidly. Tato pravidla jsou dále definována hned za selektorem v bloku, jenž je uzavřen složenými závorkami. [41]

Příklad CSS kódu:

```
body {  
    font-style: italic;  
    font-decoration: blink;  
}
```

### 3.2 JavaScript

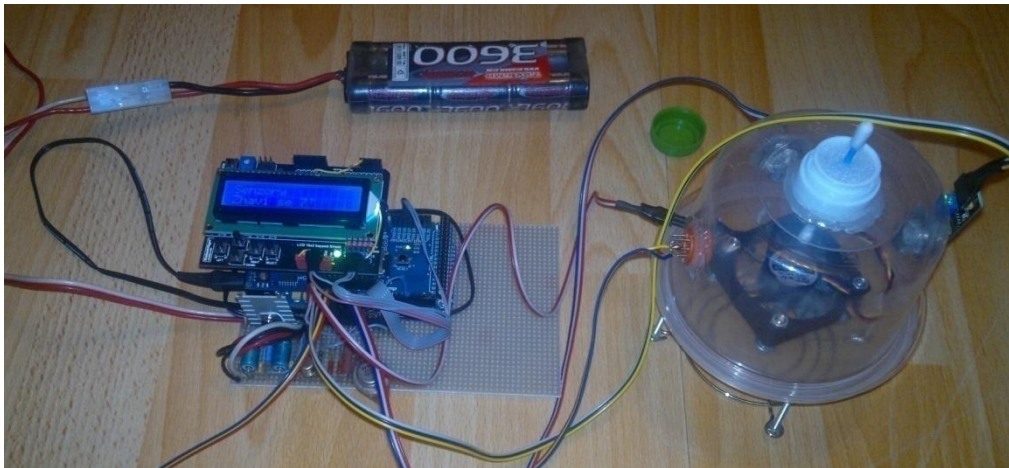
JavaScript je objektově orientovaný skriptovací jazyk, který je určený pro řešení dynamiky stránek. Zdrojový kód se vkládá přímo do HTML dokumentu a není potřeba kompilovat, je zpracováván na straně klienta. Syntaxí je podobný jazykům C, C++ nebo Java. Obecně se JavaScript používá k řízení chování klientského prohlížeče jako je dynamická práce s dialogovými okny, formuláři, obrázky nebo pro práci s cookies. [42]

Příklad kódu v JavaScriptu:

```
<script>  
var cislo, text;  
cislo = ( 4 + 5 ) * 7;  
text = "Hodnota cisla je " + cislo;  
document.write(text);  
</script>
```

## 4 NÁVRH A REALIZACE ZAŘÍZENÍ PRO MONITOROVÁNÍ VŮNÍ

Cílem této práce bylo vytvoření prototypu jednoduchého, nízkorozpočtového, přenosného zařízení, které je schopno od sebe rozlišit vybrané vůně. Princip funkce konstruovaného zařízení se nějak výrazně neliší od funkčnosti běžně dostupných elektronických nosů. Zařízení potřebuje nejprve znát referenční vůně, mezi kterými bude při dalších měřeních vybírat a vyhodnocovat, o jakou vůni se jedná. Přičemž veškerá data má uložena na paměťovém úložišti typu microSD. Kromě toho je schopno komunikovat se vzdáleným serverem pomocí technologie ethernet a sdílet tak svá data. Stejným způsobem lze zařízení nepřímo řídit přes tento vzdálený server pomocí jednoduchých webových formulářů, které umožňují jeho snadné ovládání nebo rozšířené zpracovávání dat. Nicméně zařízení disponuje i displejem a tlačítky, kterými jej lze rovněž ovládat.

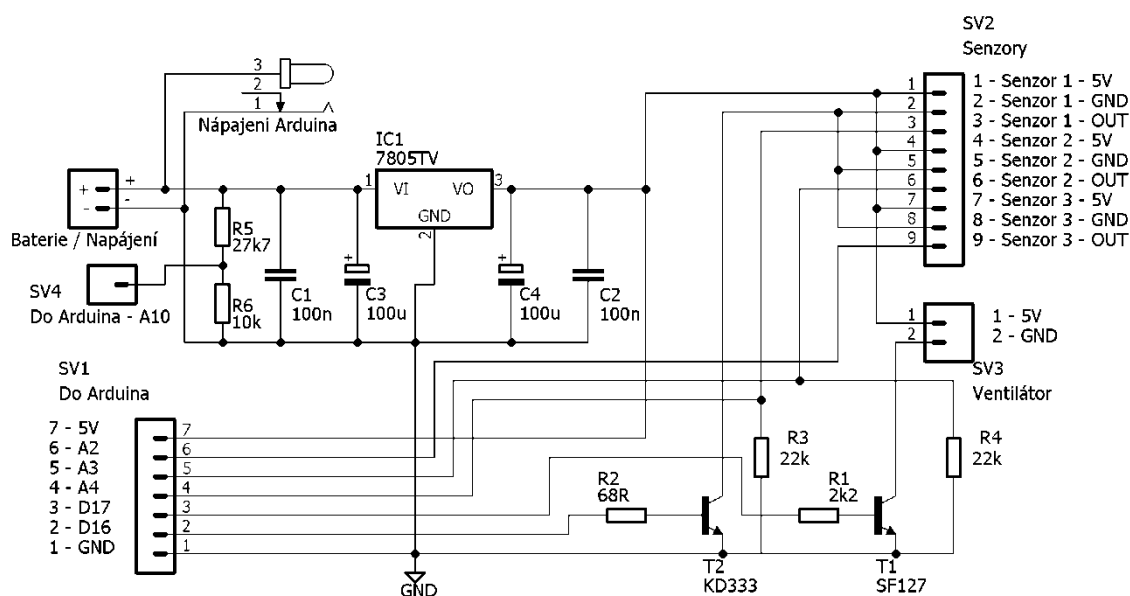


Obr. 19: Konečné uspořádání konstruovaného měřicího zařízení

### 4.1 Konstrukce zařízení

Celé zařízení je ovládáno mikrokontrolérem ATmega1280 na vývojové desce Arduino Mega. Tento typ platformy byl zvolen především kvůli vyšším požadavkům této aplikace na paměť programu i dat a pro vymezení dostatečného prostoru při dalším rozšiřování (např. dalšími senzory). Základní desku Arduina dále rozšiřují dva moduly. Jedná se o modul "*Ethernet Shield*" zajišťující nejen komunikaci s ostatními zařízeními v počítačové síti pomocí čipu W5100 firmy WIZnet, ale obsahuje i slot a řadič umožňující připojení microSD karty jako přídatného externího úložiště. Druhým modulem je dvouřádkový displej s pěti tlačítky, značen jako "*LCD display BLUE*

*1602A Shield*", který slouží jako prostředek ke komunikaci s uživatelem. Detekci vůně zprostředkovávají tři plynové senzory, které jsou uzavřeny v oddělitelné měřicí komoře. Do ní je zasunut vzorek na vatové tyčince. Kromě senzorů se v komůrce nachází také ventilátor, kterým lze před měřením do komory vehnat čistý vzduch nebo urychlit rozptýření měřeného plynu po celém objemu komory. Původní model zařízení počítal s možností plynulé regulace otáček ventilátoru pomocí modulace obdélníkových signálů (PWM), ovšem později se ukázalo, že nejefektivnější je ponechat ventilátor běžet na nejvyšších otáčkách. Sensory byly voleny levné a snadno dostupné a to pokud možno takové, aby reagovali na odlišné skupiny plynů. Jedná se o senzor MQ-6, který je nejcitlivější především na propan nebo isobutan (300 - 10000 ppm) a méně citlivý na alkohol. Dále senzor MQ-3, který je velmi citlivý na alkohol (25 - 500 ppm) a senzor MQ-8, určen pro detekci vodíku (100 - 1000 ppm). Zároveň je třeba zmínit, že při vyhodnocování dat ze senzorů není třeba znát přesné množství měřené látky v plynu (např. v jednotkách ppm). Protože zařízení při konečném vyhodnocování dat pouze porovnává svá dřívější měření mezi sebou. Z tohoto důvodu bylo použito doporučené zapojení výrobce, bez dodatečné kalibrace nastavovacích odporů a číselné výstupy senzorů jsou reprezentovány jako čistá data z vnitřního 10-ti bitového A/D převodníku mikrokontroléru. Dále byl vytvořen zvláštní napájecí a spínací obvod pro tyto senzory a ventilátor, jelikož je nelze napájet přímo z desky Arduina kvůli jejich relativně velké spotřebě energie. Jelikož se jedná o přenosné zařízení, zdrojem napájení celého systému je šesti-člávková modelářská baterie o jmenovitém napětí 7,2 V. Zařízení je ovšem možné napájet v napěťovém rozsahu od 6 V do 13 V.



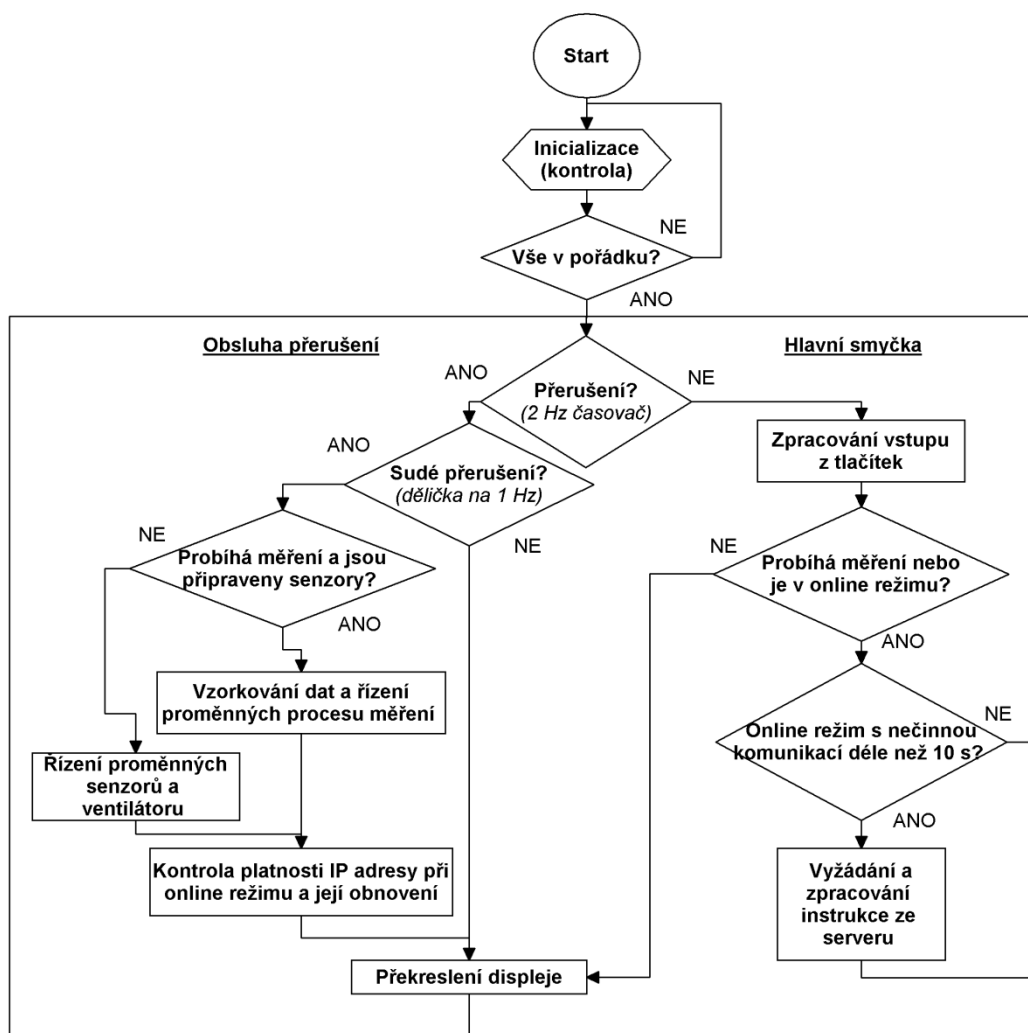
Obr. 20: Schéma zapojení napájecího a spínacího obvodu senzorů

## 4.2 Návrh programového vybavení

Funkčnost celého systému lze rozdělit do tří zdrojových kódů.

První je napsán v programovacím jazyce Arduina, uložen v mikrokontroléru a řídí jeho chod včetně všech jeho modulů, především se jedná o zprostředkování komunikace mezi uživatelem prostřednictvím displeje a tlačítek. K tomuto účelu bylo vytvořeno jednoduché menu s požadovanými příkazy a nastaveními, které uživateli umožní provádět různé operace. Do této skupiny akcí spadá i samotný proces měření. Než je však uživateli umožněno se zařízením pracovat, je ihned po spuštění provedena kontrola veškerých komponent, potřebných pro správnou funkci zařízení. Dokud neproběhne kontrola bez chyb, není možné zařízení používat. Dále tento program také řídí svoji síťovou komunikaci se vzdáleným serverem - vysílá požadavky, kterými získává úlohy, které následně zpracovává a provádí. Tento kód se proto neobešel bez systému přerušení, kterého je s výhodou využíváno především při procesu měření. Tímto způsobem byla přiřazena procesu nejvyšší priorita. Síťová komunikace, společně se zápisem a čtením z externí paměti se řadí prioritně na střední úroveň. Zatímco obsluha displeje a vstupu z tlačítek mají prioritu nejnižší. Celý tento kód zabírá necelých 34% programové paměti FLASH. Paměť SRAM je již při startu programu zaplněna z 20% globálními proměnnými a v průběhu programu může tato hodnota dosáhnout až na 70% celkové velikosti a to především při dlouhých měřeních, kdy se do této paměti naměřená data průběžně ukládají a až při skončení měření mohou být uložena na externí úložiště. Maximální možná délka měření byla omezena na deset minut, aby nedošlo k přetečení paměti. Tento zdrojový kód lze nalézt v příloze a jeho základní blokové schéma na obr. 21.

Druhý program byl napsán v HTML a rozšířen o PHP [48]. Leží na vzdáleném webovém serveru a zajišťuje řízení síťové komunikace mezi Arduinem a samotným serverem, za kterým se skrývá uživatel, jež vzdáleně ovládá zařízení přidáváním akcí do fronty úloh. Tato komunikace je založena na principech protokolu HTTP. Konkrétně se jedná o běžné *POST* metody, kde Arduino se chová jako jednoúčelový a jednoduchý webový prohlížeč, jež reaguje pouze na určité webové stránky (zprávy). Zjednodušeně si lze výstup tohoto kódu představit jako generátor webové stránky, jejíž výsledná data s jejich uspořádáním závisí na konkrétním požadavku ze strany zařízení, tedy prvního programu v mikrokontroléru. Tento kód tak zajišťuje i správu vyhrazeného paměťového úložiště na webovém serveru pro každé zařízení zvlášť a připravuje data k dalšímu zpracování. Zdrojový kód lze nalézt v příloze pod názvem "*arduino.php*".

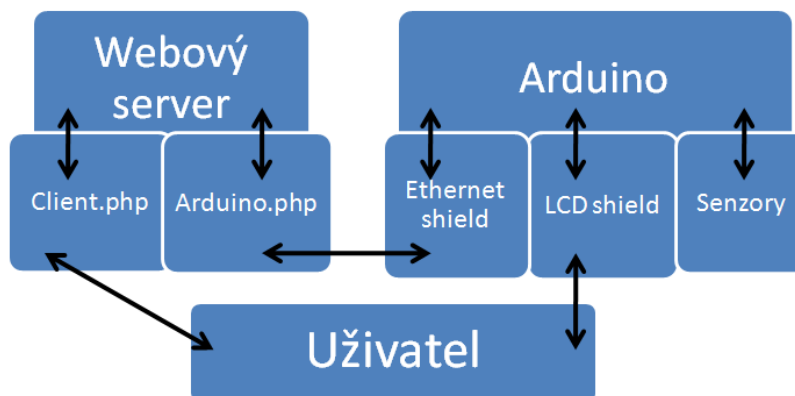


Obr. 21: Základní blokové schéma kódu Arduina

Třetí kód se opět nachází na vzdáleném webovém serveru a vytváří uživatelské rozhraní pro připojující se uživatele na tomto serveru formou jednoduchých webových stránek s formuláři, kterými uživatel může zařízení pohodlněji ovládat nebo procházet a zpracovávat naměřená data v grafické podobě nebo je i graficky porovnávat (obr. 23). Dále zde uživatel nalezne informace o posledním připojení zařízení nebo jeho aktuálním stavu. Jedná se především o akce obousměrné výměny naměřených dat nebo konfigurace mezi Arduinem a serverem, dále vzdálené spuštění automatického měření nebo senzorů. Tento kód byl napsán v HTML, rozšířen o PHP a Javascript. Prakticky se jedná o kód převádějící data do formy snadno čitelné pro uživatele. Tento kód byl pojmenován "*client.php*" a lze jej nalézt v příloze.

K vývoji výše uvedených programů bylo za potřebí softwaru třetích stran. Mimo samotného softwarového kitu Arduina, *Massimo Banzi* a *David Cuartielles, Itálie* [43] i Notepad++, *Don Ho, Francie* [44] pro snazší psaní HTML kódů. Dále aplikace Wireshark, *Geralda Combse* [45], která umožňuje detailně monitorovat provoz sítě.

Aplikací Tiny DHCP Server, *SoftCab* [46] bylo možno simulovat chod serveru DHCP a testovat tak funkčnost tohoto protokolu v zařízeních.



Obr. 22: Znárodnění komunikace mezi hlavními komponenty

Seznam provedených měření na zařízení *DEADBEEFFEEED*:

Skryt smazana

Cislo	Vyber	Nazev	Delka [s]	Typ
1	<input type="radio"/>	Vzduch1	60	Bezny vzorek
2	<input type="radio"/>	Vzduch2	60	Kalibracni vzorek (Sensor1<-40;40> Sensor2<-60;60> Sensor3<-20;20>)
3	<input type="radio"/>	Rumova1	60	Bezny vzorek
4	<input type="radio"/>	Rumova2	60	Kalibracni vzorek (Sensor1<-32;32> Sensor2<-32;32> Sensor3<-16;16>)
5	<input type="radio"/>	Vzduch3	60	Bezny vzorek
6	<input type="radio"/>	Kokos1	60	Kalibracni vzorek (Sensor1<-160;8> Sensor2<-64;8> Sensor3<-64;8>)
7	<input checked="" type="radio"/>	Kokos2	60	Bezny vzorek
8	<input type="radio"/>	Vanilka1	60	Kalibracni vzorek (Sensor1<-100;8> Sensor2<-16;8> Sensor3<-40;8>)
9	<input type="radio"/>	Vanilka2	60	Bezny vzorek
10	<input type="radio"/>	Citron1	60	Kalibracni vzorek (Sensor1<-120;8> Sensor2<-16;16> Sensor3<-50;8>)
11	<input type="radio"/>	Citron2	60	Bezny vzorek
12	<input type="radio"/>	Vzduch4	60	Bezny vzorek
13	<input type="radio"/>	Jablko1	60	Kalibracni vzorek (Sensor1<-20;20> Sensor2<-20;20> Sensor3<-8;8>)
14	<input type="radio"/>	Jablko2	60	Bezny vzorek
15	<input type="radio"/>	VanilkaOil1	60	Kalibracni vzorek (Sensor1<-20;20> Sensor2<-20;20> Sensor3<-8;8>)
16	<input type="radio"/>	VanilkaOil2	60	Bezny vzorek
17	<input type="radio"/>	Pomeranc1	60	Kalibracni vzorek (Sensor1<-16;16> Sensor2<-16;16> Sensor3<-8;8>)
18	<input type="radio"/>	Pomeranc2	60	Bezny vzorek
19	<input type="radio"/>	JablkoSeSkorici1	60	Kalibracni vzorek (Sensor1<-16;16> Sensor2<-16;16> Sensor3<-8;8>)
20	<input type="radio"/>	JablkoSeSkorici2	60	Bezny vzorek
21	<input type="radio"/>	ZimniSvet1	60	Kalibracni vzorek (Sensor1<-10;10> Sensor2<-10;10> Sensor3<-10;10>)
22	<input type="radio"/>	ZimniSvet2	60	Bezny vzorek
23	<input type="radio"/>	SantaloveDrevo1	60	Kalibracni vzorek (Sensor1<-20;20> Sensor2<-20;20> Sensor3<-8;8>)
24	<input type="radio"/>	SantaloveDrevo2	60	Bezny vzorek
26	<input type="radio"/>	Vzduch6	60	Bezny vzorek
27	<input type="radio"/>	Vzduch7	60	Bezny vzorek

Akce:

Zobrazit data

[Domu](#)

[Zpet](#)

Obr. 23: Ukázka webového uživatelského rozhraní aplikace

### 4.3 Měření a diskuse

Při měření je nutné dodržet předem dané postupy, kterými se zajistí podobné podmínky pro všechna měření. Při měření je třeba zachovávat stejné koncentrace vůní (množství vonného roztoku naneseného na vatovou tyčinku) a brát v potaz také znečištění okolního vzduchu především detekovatelnými látkami a nebo vlhkost vzduchu, která rovněž ovlivňuje charakteristiky senzorů. Samotné měření pak probíhá následovně: Nejdříve se uživatel ujistí, zda se v měřící komoře nenacházejí zbytky nechtěných plynů z minulých měření, které by mohly zapříčinit zkreslení výsledků. V případě potřeby lze tyto plyny odstranit vyfouknutím pomocí ventilátoru zabudovaného v měřící komoře. Jakmile je komora připravena, do otvoru těsnícího víka je zasunuta vatová tyčinka s testovaným vzorkem. Zároveň je komora sejmuta z podstavce, zajišťujícího dostatečný přísun okolního vzduchu při čištění komory. Následně jsou uživatelem spuštěny senzory, které ovšem musí být dostatečně dlouhou dobu předehřívány na svoji pracovní teplotu, což pohlídá mikrokontrolér. Během této doby je i pomocí ventilátoru urychlen proces odpařování a rozptylování vzorku po celém objemu komory. Jakmile jsou senzory připraveny, může začít samotné měření, kde každou vteřinu probíhá vzorkování všech senzorů najednou. Maximální délka měření je omezena deseti minutami. Po skončení měření jsou senzory automaticky vypnuty a spuštěno čištění komory, kdy je vhodné komoru postavit na podstavec pro lepší proudění vzduchu. Naměřená data lze dále zpracovávat dle uvážení uživatele a sestavovat tak například databázi známých vůní. Výsledky měření lze ukládat buď jako kalibrační (referenční) a nebo jako běžné. Kde kalibrační vzorky oproti běžným navíc obsahují informaci o velikosti mezi, ve kterých se hodnota výstupu senzoru má pohybovat pro takto změřenou vůni. I když je snahou zachovat stejné podmínky pro všechna měření, prakticky lze jen velmi obtížně naměřit naprosto stejné výsledky při opakovaném měření jednoho vzorku. Následné vyhodnocování vůně probíhá tak, že je ze všech naměřených hodnot každého senzoru vypočtena střední hodnota, která je porovnávána se všemi ostatními kalibračními vzorky. Referenční rozsah se řídí střední hodnotou každého z kalibračních vzorků, tedy pouze vzorek s nejmenším absolutním rozdílem je chápán jako výsledek vyhodnocovacího algoritmu. Limity kalibračních vzorků lze určit pouze experimentálně opakovaným měřením. Výše uvedený vyhodnocovací algoritmus je vhodný především pro data s velkou časovou stálostí. Ovšem variabilita naměřených dat v čase závisí na rychlosti odezvy daného senzoru. Z výsledků měření lze vidět, že senzor MQ-3 (alkoholový) má nejpomalejší odezvu. Takové výsledky nemusejí mít vždy požadovanou výpovědischopnost, protože naprosto odlišné charakteristiky mohou mít stejnou střední hodnotu. Detailnější popis možností zařízení lze nalézt v uživatelské příručce.



### 4.3.1 Databáze vůní

Prvotní testovací měření byla prováděna na starém uspořádání zařízení, které bylo pouze esteticky hůře provedeno, funkčnost zařízení avšak zůstala zachována. Bylo použito celkem deset různých vzorků k měření a testování funkčnosti. Jedná se především o různá potravinářská aroma nebo vonné oleje (tab. 4 a obr. 24). Tyto vůně byly několikrát proměřovány a ze získaných dat vytvořeny kalibrační vzorky, definující charakter dané vůně. Rovněž jako referenční vzorek byl použit pouhý okolní vzduch, kdy komora byla prázdná a otevřená. Vzorky vzduchu slouží především jako ukazatele čistoty měřící komory a rychlosti odezvy vzorků, protože měření čistého okolního vzduchu bylo prováděno ihned po doměření vybraných aromat. Bohužel ne vždy byl změřený vzorek hned správně vyhodnocen a rozpoznán. Některé vůně jsou si svým charakterem velmi podobné a docházelo tak k jejich záměně při rozpoznávání. Například velmi podobný charakter byl naměřen u jablečné vůně a vanilkového vonného oleje (obr. 25). K těmto nespolehlivým výsledkům lehce přispívá variabilita vnějších podmínek mezi měřeními, ovšem hlavní příčinou nepřesností je především nízký počet použitých senzorů a jejich citlivosti na méně vhodné skupiny plynů. Z měření lze vidět, že především naměřené hodnoty senzoru MQ-3, který je velmi citlivý na alkohol, mají obvykle největší váhu při určování charakteru dané vůně, což bylo u této aplikace i očekáváno, protože potravinářská aroma jsou často vyráběna rozpouštěním silic a extraktů v rafinovaném lihu. [47] Rovněž se jedná o prvotní výsledky, které doposud nebyly z časových důvodů opakovaně proměřovány. Proto nelze určit přesný zdroj nepřesností.

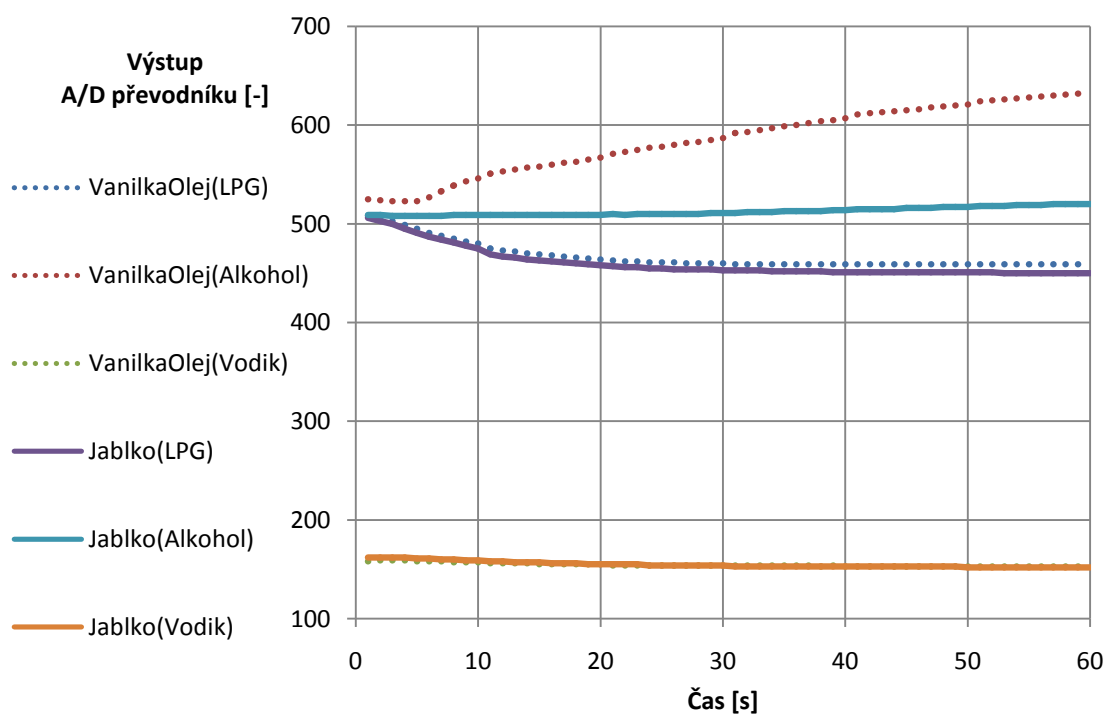
Příklad referenčního a obyčejného vzorku rumového aromatu, vykresleného ve webovém uživatelském rozhraní je na obr. 26. Veškerá uvedená měření, která se nacházejí v příloze této práce, byla provedena na konečném, vylepšeném uspořádání (obr. 19) a za běžných pokojových podmínek - běžné pokojové teploty 23 °C a relativní vlhkosti kolem 58 %. Při měření bylo respektováno veškerých požadavků na správné měření, přesně jak je popsáno v uživatelské příručce, která se nachází v příloze této práce.



Obr. 24: Měřené vzorky

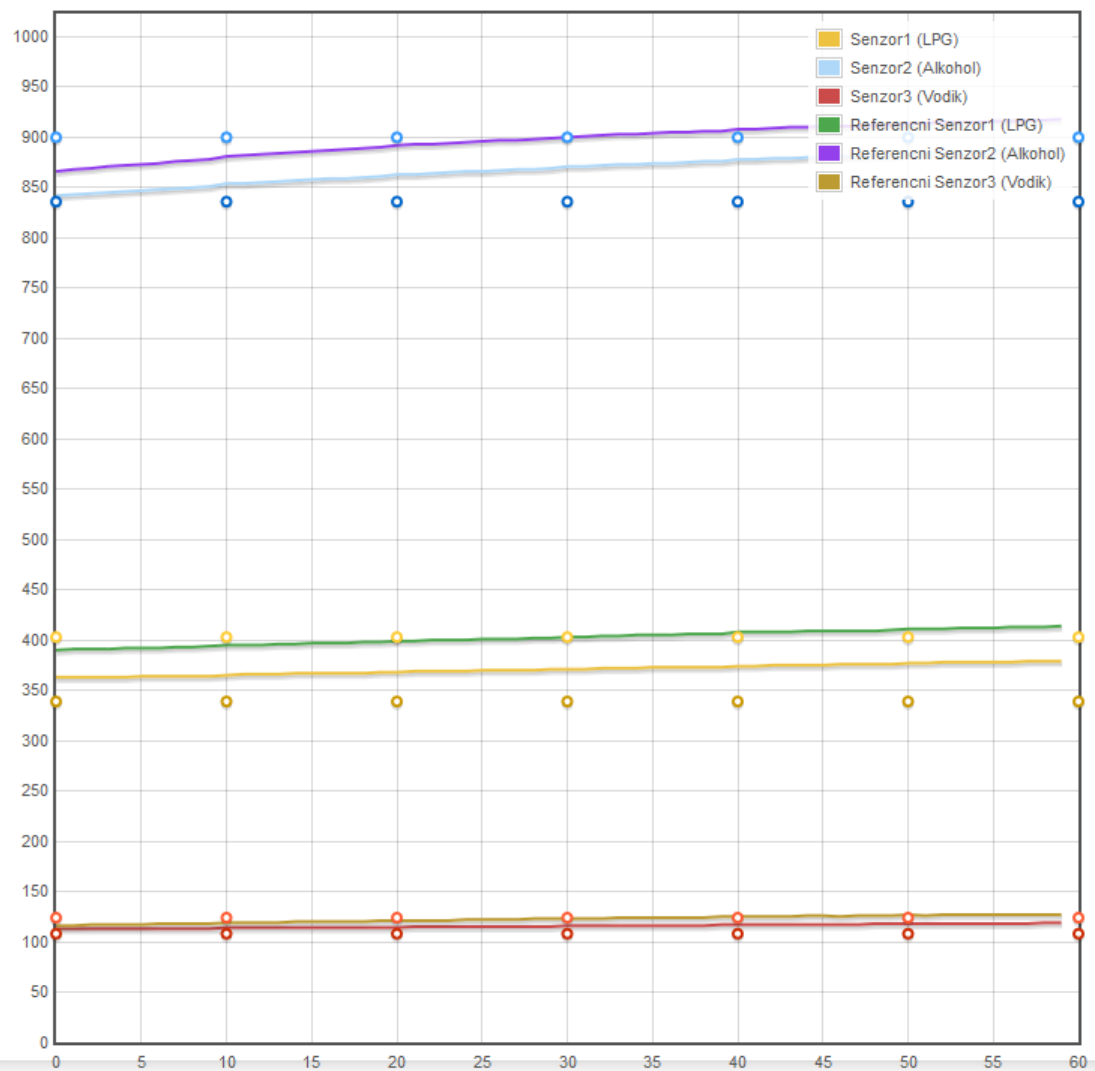
Tab. 4: Tabulka použitých vzorků

Název	Výrobce / distributor
Kokosové	Aroco s.r.o.
Vanilkové	Aroco s.r.o.
Citronové	Aroco s.r.o.
Rumové	Dr.Oetker s.r.o.
Pomeranč	Lidl ČR v.o.s.
Jablko	Kozák Svitavy s.r.o.
Vanilka	Lidl ČR v.o.s.
Santalové dřevo	Kozák Svitavy s.r.o.
Zimní svět	Lidl ČR v.o.s.
Jablko se skořicí	Lidl ČR v.o.s.



Obr. 25: Porovnání dvou podobných charakteristik rozdílných aromat

### Zavislost velkosti vystupniho signalu senzoru na case [s] vzorku *Rumova2*



#### Porovnat data s:

Rumova1

Potvrdit

#### Prumer vychoziho vzorku (Rumova2):

senzor 1: 371

senzor 2: 868

senzor 3: 116

#### Prumer referencniho vzorku (Rumova1):

senzor 1: 402 (+31)

senzor 2: 897 (+29)

senzor 3: 122 (+6)

Obr. 26: Grafické zobrazení z webového rozhraní - porovnání kalibračního vzorku s odpovídající vůní

## 5 ZÁVĚR

Tato práce se věnuje realizaci levného, jednoduchého, přenosného zařízení, určeného k měření vůní a následnému zpracovávání výsledných dat. Toto zařízení je rovněž schopné komunikovat s webovým serverem a přes toto rozhraní zařízení vzdáleně ovládat. V teoretické části se práce zabývá základními principy nutnými pro činnost takového zařízení, tedy především samotným Arduinem, síťovými komunikačními mechanismy a práci s jazykem popisující webové stránky. Dále také možnými senzory, kterými lze vůně měřit, mechanismy detekce vůně a využití elektronických nosů v potravinářství. Závěr práce obsahuje samotnou realizaci zařízení, společně s výsledky zkušebního měření a vyhodnocením. Dále příloha obsahuje uživatelskou příručku, která detailně popisuje možnosti zařízení a postupy při jeho používání.

Praktické navržené zapojení společně s řídicí jednotkou pracují dostatečně přesně na to, aby bylo možné měřit charakteristiky vůní. Rovněž navržený systém vyhodnocování a porovnávání dat jsou funkční a celkově se osvědčily. Mezi charakteristikami vůní jsou někdy jen nepatrné rozdíly, což v těchto případech nedovoluje přesně identifikovat měřenou vůni a dochází k záměně vzorků a nesprávným výsledkům. Jedná se především o vzorky na olejové bázi s netypickými vůněmi jako jsou *jablko se skořicí* nebo *zimní svět*.

Vytvořené webové uživatelské rozhraní pro vzdálené řízení tohoto zařízení je graficky velmi jednoduše provedeno, ale pro svoji funkci je dostatečné a přehledné. Uživatel má možnost ukládat data pod libovolným názvem, mazat je nebo i měnit konfiguraci zařízení. Naměřená data lze následně graficky porovnávat. Toto rozhraní umožňuje také práci s více zařízeními.

Nevýhodou celého návrhu je především nízký počet senzorů, omezující detekční rozsah látek. Dále senzory použité v tomto návrhu nejsou nejvhodnější pro detekci vůní v potravinářství. Dalším, pravděpodobně nejzávažnějším problémem je samotná měřicí komora, která je zhotovena z tenkého plastu, který snadno absorbuje měřené vůně a následné čištění komory je mnohem náročnější. Důkazem je průběh charakteristik čistého vzduchu po skončení měření ostatních vzorků.

Mnohem lepších výsledků může být dosaženo přidáním dalších, méně dostupných senzorů, citlivých na látky používané v potravinářství (např. Amoniak). Dále záměnou měřicí komory na skleněnou lze vyloučit problémy způsobené absorpcí vůní. Takové vylepšení ovšem přináší řadu technologických problémů a podstatně zvyšuje finanční náklady na zhotovení zařízení.

# LITERATURA

- [1] *STUDIE EMISE PACHOVÝCH LÁTEK ZE ZEMĚDĚLSKÝCH ZDROJŮ* [online]. Praha, 2004 [cit. 2016-04-17]. Dostupné z: [http://www.cschi.cz/odour/files/czech/studie\\_zem.pdf](http://www.cschi.cz/odour/files/czech/studie_zem.pdf)
- [2] *Proč nelze pachy měřit analytickými metodami* [online]. 2016 [cit. 2016-04-17]. Dostupné z: <http://nehcemezitvesmradu.cz/proc-nelze-pachy-merit-analytickymi-metodami/>
- [3] CARLSSON, Mikael A. a Blanka KALINOVÁ. Tajemství čichu poodhaleno. *Vesmír*. 2005, (84), 148 - 155.
- [4] *ENose Technology* [online]. [cit. 2016-04-17]. Dostupné z: <http://www.enose.nl/rd/technology/>
- [5] Electronic Nose 'Smells' Rotting Food!. GOPAL, Deepa. *Youngzine* [online]. 2015 [cit. 2016-04-17]. Dostupné z: <http://www.youngzine.org/article/electronic-nose-smells-rotting-food>
- [6] JELÍNEK, Tomáš. DETECE KYSLÍKU POMOCÍ SENZORU PLYNŮ. Diplomová práce. Brno: FEKT VUT v Brně, 2012.
- [7] FRIEDECKÝ, D. a K. LEMR. *Úvod do hmotnostní spektrometrie* [online]. 2012 [cit. 2016-04-17]. Dostupné z: <http://www.cskb.cz/res/file/KBM-pdf/2012/2012-3/KBM12-3-Friedecky-152.pdf>
- [8] ZACHAŘ, Pavel a David SÝKORA. *Plynová chromatografie* [online]. [cit. 2016-04-17]. Dostupné z: <http://old.vscht.cz/anl/lach2/GC.pdf>
- [9] *06. Plynová chromatografie (GC)* [online]. [cit. 2016-04-17]. Dostupné z: [http://cheminfo.chemi.muni.cz/chem\\_sekce/predmety/C7300/GC/uvod.pdf](http://cheminfo.chemi.muni.cz/chem_sekce/predmety/C7300/GC/uvod.pdf)
- [10] PYTLÍČEK, Zbyněk. SENZORY PLYNŮ, JEJICH KONSTRUKCE A TESTOVÁNÍ. Diplomová práce. Brno: FEKT VUT v Brně, 2010.
- [11] GABLECH, Imrich. POLOVODIČOVÉ SENZORY PLYNŮ NA BÁZI OXIDU CINIČITÉHO. Diplomová práce. Brno: FEKT VUT v Brně, 2014.
- [12] OPEKAR, František. *SENZORY PLYNNÝCH LÁTEK* [online]. [cit. 2016-04-11]. Dostupné z: <http://old.vscht.cz/anl/paci/PAC/prezentace/senzory.pdf>
- [13] Electronic Nose 'Smells' Rotting Food!. GOPAL, Deepa. *Youngzine* [online]. 2015 [cit. 2016-04-17]. Dostupné z: <http://www.youngzine.org/article/electronic-nose-smells-rotting-food>
- [14] Portable "electronic nose" smells your meat for you and identifies food poisoning risk. MARKHAM, Derek. *Treehugger* [online]. 2014 [cit. 2016-04-17]. Dostupné z: <http://www.treehugger.com/gadgets/portable-electronic-nose-smells-your-meat-identifies-food-poisoning-risk.html>
- [15] *Jak se měří pachové látky* [online]. 2016 [cit. 2016-04-17]. Dostupné z: <http://nehcemezitvesmradu.cz/jak-se-meri-pachove-latky/>
- [16] KRAUS, Ronald a Ladislav CHLÁDEK. *OLFAKTOMETRIE A MOŽNOSTI JEJÍHO VYUŽITÍ PRO STANOVENÍ EMISÍ V PIVOVARSTVÍ* [online]. 2008 [cit. 2016-04-17]. Dostupné z: <http://www.kvasnyprumysl.cz/download.php?clanek=126>
- [17] *Arduino - introduction* [online]. [cit. 2015-11-6]. Dostupné z: <https://www.arduino.cc/en/Guide/Introduction>

- [18] *Arduino Shield List* [online]. [cit. 2015-11-6]. Dostupné z: <http://shieldlist.org/>
- [19] *Arduino* [online]. [cit. 2015-11-6]. Dostupné z: <http://www.root.cz/n/arduino/>
- [20] SLINTÁK, Vlastimil. *Vývojové prostředí a programování Arduina* [online]. 2011 [cit. 2015-11-8]. Dostupné z: <http://uart.cz/90/ide-a-programovani-arduino/>
- [21] BABČANÍK, Jan. *Začínáme s mikroprocesory Atmel AVR* [online]. 2006 [cit. 2015-11-8]. Dostupné z: <http://vyvoj.hw.cz/teorie-a-praxe/zaciname-s-mikroprocesory-atmel-avr.html>
- [22] FÜHRER, Ondřej. *Architektura mikroprocesoru AVR ATMega ( Pokročilé architektury počítačů )* [online]. Vysoká škola báňská – Technická univerzita Ostrava Fakulta elektrotechniky a informatiky [cit. 2015-11-8]. Dostupné z: [http://wh.cs.vsb.cz/mil051/images/c/ca/PAP\\_Architektura](http://wh.cs.vsb.cz/mil051/images/c/ca/PAP_Architektura)
- [23] KARAS, Ondřej. *AVR - Časovače a čítače* [online]. 2006 [cit. 2015-11-14]. Dostupné z: <http://bastleni.edev.cz/clanky/avr/42-citace-a-casovace.html>
- [24] SLINTÁK, Vlastimil. *Arduino a přerušení* [online]. 2011 [cit. 2015-11-14]. Dostupné z: <http://uart.cz/271/arduino-a-preruseni/>
- [25] TŮMA, Ondřej. *Arduino - pulsně šířková modulace* [online]. 2011 [cit. 2015-11-14]. Dostupné z: [http://www.linuxsoft.cz/article.php?id\\_article=1878](http://www.linuxsoft.cz/article.php?id_article=1878)
- [26] VLČEK. *Digitální zpracovávání naměřených signálů* [online]. [cit. 2015-11-21]. Dostupné z: [http://elektro.tzb-info.cz/download.py?file=docu/texty/0001/000102\\_cislicove\\_zpracovani\\_namerenych\\_signalu.pdf](http://elektro.tzb-info.cz/download.py?file=docu/texty/0001/000102_cislicove_zpracovani_namerenych_signalu.pdf)
- [27] TIŠNOVSKÝ, Pavel. *Externí sériové sběrnice SPI a I<sup>2</sup>C* [online]. 2008 [cit. 2015-11-16]. Dostupné z: <http://www.root.cz/clanky/externi-seriove-sbernice-spi-a-i2c/>
- [28] *Arduino Ethernet Shield* [online]. [cit. 2015-11-20]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoEthernetShield>
- [29] *Síťové modely a architektury* [online]. [cit. 2015-11-20]. Dostupné z: <http://site.borec.cz/02%20Architektura%20iso%20osi.htm>
- [30] *Počítačové sítě - Model ISO/OSI* [online]. [cit. 2015-11-20]. Dostupné z: <http://site.the.cz/index.php?id=4>
- [31] KLEMENT, Milan. *Technologie počítačových sítí* [online]. [cit. 2016-04-10]. Dostupné z: [http://www.kteiv.upol.cz/uploads/soubory/klement/web1/TPS\\_2014/prednasky/p%C5%99ed%2011.pdf](http://www.kteiv.upol.cz/uploads/soubory/klement/web1/TPS_2014/prednasky/p%C5%99ed%2011.pdf)
- [32] DROMS, Ralph. *Dynamic Host Configuration Protocol* [online]. [cit. 2016-04-10]. Dostupné z: <https://tools.ietf.org/html/rfc2131>
- [33] HORALEK, Josef. *Počítačové sítě 1* [online]. [cit. 2016-04-10]. Dostupné z: <http://www.horalek.org/site/Module10.pdf>
- [34] KOSEK, Jiří. *Základy protokolu HTTP* [online]. 1999 [cit. 2015-11-20]. Dostupné z: <http://www.kosek.cz/clanky/iweb/05.html>
- [35] *Počítačové sítě - Stavové kódy protokolu HTTP* [online]. [cit. 2015-11-20]. Dostupné z: <http://site.the.cz/index.php?id=29>
- [36] *FAT File System* [online]. [cit. 2015-11-28]. Dostupné z: <https://technet.microsoft.com/en-us/library/cc938438.aspx>

- [37] *Princip paměti FLASH* [online]. [cit. 2015-11-28]. Dostupné z: <http://noel.feld.cvut.cz/vyu/scs/prezentace2003/Flash-Intel/>
- [38] KOSEK, Jiří. *HTML5 Úvod* [online]. 2008 [cit. 2015-11-28]. Dostupné z: <http://htmlguru.cz/uvod.html>
- [39] KOSEK, Jiří. *Historie a vývoj HTML* [online]. 2008 [cit. 2015-11-28]. Dostupné z: <http://htmlguru.cz/uvod-historie.html>
- [40] KOSEK, Jiří. *HTML5 Základy HTML* [online]. 2008 [cit. 2015-11-28]. Dostupné z: <http://htmlguru.cz/zaklady-cleneni-dokumentu.html>
- [41] KOSEK, Jiří. *Kaskádové styly* [online]. 2014-10-02 [cit. 2015-11-28]. Dostupné z: <http://www.kosek.cz/vyuka/4iz228/prednasky/css.pdf>
- [42] KOČIČKA, Petr. *Úvod do jazyka JavaScript* [online]. 1996 [cit. 2015-11-29]. Dostupné z: <http://www.javascript.cz/?chap=4>
- [43] *Arduino software kit* [online]. [cit. 2016-04-17]. Dostupné z: <https://www.arduino.cc/en/Main/Software>
- [44] *Notepad++* [online]. [cit. 2016-04-17]. Dostupné z: <https://notepad-plus-plus.org/>
- [45] *Wireshark* [online]. [cit. 2016-04-17]. Dostupné z: <https://www.wireshark.org/>
- [46] *Tiny DHCP Server* [online]. [cit. 2016-04-17]. Dostupné z: <http://www.softcab.com/dhcp-server/>
- [47] *Aromex* [online]. [cit. 2016-04-17]. Dostupné z: <http://www.aromka.cz/aromata>
- [48] PONKRÁC, Miloslav. *PHP a MySQL: bez předchozích znalostí*. Brno: Computer Press, 2007. ISBN 978-80-251-1758-3.

# SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

MOS	Metal Oxide Semiconductor
ISFET	Ion Sensitive Field Effect Transistor
SAW	Surface Acoustic Wave
LED	Light Emitting Diode
OU	Odour Unit
EROM	European Reference Odour Unit
USB	Universal Serial Bus
GSM	Global System for Mobile communications
LCD	Liquid Crystal Display
IDE	Integrated Development Environment
SRAM	Static Random Access Memory
PWM	Pulse Width Modulation
SPI	Serial Peripheral Interface
TCP	Transmission Control Protocol
UCP	User Datagram Protocol
ICMP	Internet Control Message Protocol
IPv4	Internet Protocol version 4
ARP	Address Resolution Protocol
IGMP	Internet Group Management Protocol
PPPoE	Point to Point Protocol over Ethernet
FAT	File Allocation Table
ISO	International Organization for Standardization
OSI	Open Systems Interconnection
HTTP	Hyper Text Transfer Protocol
FTP	File Transfer Protocol
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
HTML	Hyper Text Markup Language
SGML	Standard Generalized Markup Language
WWW	World Wide Web
IETF	Internet Engineering Task Force
XML	Extensible Markup Language
CSS	Cascade Style Sheet



## SEZNAM OBRÁZKŮ

Obr. 1: Různé kombinace intenzity pachu dvou látek o stejné koncentraci .....	9
Obr. 2: Schéma uspořádání plynového chromatografu [9].....	12
Obr. 3: Struktura tlustovrstvého chemorezistoru [11] .....	13
Obr. 4: Zobrazení potenciálové bariéry na vzduchu a redukčním plynu [6] .....	14
Obr. 5: Struktura chemokondenzátoru a jeho převodní charakteristika [6].....	14
Obr. 6: Struktura chemodiody [6].....	15
Obr. 7: Příklad senzoru pH - struktura ISFET a jeho převodní charakteristika [6].....	15
Obr. 8: Konstrukční uspořádání pelistoru [10] .....	16
Obr. 9: Uspořádání senzoru SAW [11].....	17
Obr. 10: Konstrukční uspořádání optického chemického senzoru [6].....	17
Obr. 11: Peres - Přenosný elektrický nos určující čerstvost masa [14] .....	18
Obr. 12: Ukázka olfaktometru [16].....	19
Obr. 13: Vývojové prostředí Arduina .....	21
Obr. 14: Převodní charakteristika A/D převodníku [26] .....	24
Obr. 15: Zapojení pro komunikaci s více zařízeními přes SPI sběrnici [27].....	25
Obr. 16: Grafické znázornění průchodu dat v OSI modelu mezi vysílací a cílovou stanicí [29] .....	27
Obr. 17: Zobrazení základní struktury počítačové sítě s DHCP serverem [31] .....	28
Obr. 18: LCD Modul .....	32
Obr. 19: Konečné uspořádání konstruovaného měřicího zařízení .....	35
Obr. 20: Schéma zapojení napájecího a spínacího obvodu senzorů .....	36
Obr. 21: Základní blokové schéma kódu Arduina .....	38
Obr. 22: Znázornění komunikace mezi hlavními komponenty .....	39
Obr. 23: Ukázka webového uživatelského rozhraní aplikace .....	39
Obr. 24: Měřené vzorky .....	41
Obr. 25: Porovnání dvou podobných charakteristik rozdílných aromat .....	42
Obr. 26: Grafické zobrazení z webového rozhraní - porovnání kalibračního vzorku s odpovídající vůní .....	43
Obr. 27: Detail zapojení hlavních propojek .....	54
Obr. 28: Detail tlačítek a propojek .....	54
Obr. 29: Ovládací webové rozhraní .....	57

Obr. 30: Charakteristiky vzduchu, naměřené před měřením ostatních vzorků .....	59
Obr. 31: Charakteristiky potravinářského rumového aromatu .....	59
Obr. 32: Charakteristiky potravinářského kokosového aromatu .....	60
Obr. 33: Charakteristiky potravinářského vanilkového aromatu .....	60
Obr. 34: Charakteristiky potravinářského citronového aromatu .....	61
Obr. 35: Charakteristiky vzduchu, naměřené v průběhu měření ostatních vzorků.....	61
Obr. 36: Charakteristiky vůně jablečného vonného oleje.....	62
Obr. 37: Charakteristiky vůně vanilkového vonného oleje .....	62
Obr. 38: Charakteristiky vůně pomerančového vonného oleje.....	63
Obr. 39: Charakteristiky vůně jablečně-skořicového vonného oleje.....	63
Obr. 40: Charakteristiky vonného oleje s vůní zimního světa .....	64
Obr. 41: Charakteristiky vůně vonného oleje s vůní santalového dřeva .....	64
Obr. 42: Charakteristiky vzduchu, naměřené po měření ostatních vzorků.....	65

# SEZNAM TABULEK

Tab. 1: Příklady vývojových platforem Arduino.....	20
Tab. 2: Zprávy používané při komunikaci protokolu DHCP [31].....	29
Tab. 3: Základní požadavky, stavová hlášení a hlavičky [35].....	30
Tab. 4: Tabulka použitých vzorků .....	42

# SEZNAM PŘÍLOH

<b>A</b>	<b>UŽIVATELSKÁ PŘÍRUČKA .....</b>	<b>53</b>
A.1	PŘÍPRAVA ZAŘÍZENÍ.....	53
A.2	OFFLINE REŽIM .....	54
A.3	ONLINE REŽIM .....	56
<b>B</b>	<b>VÝSLEDKY MĚŘENÍ.....</b>	<b>59</b>

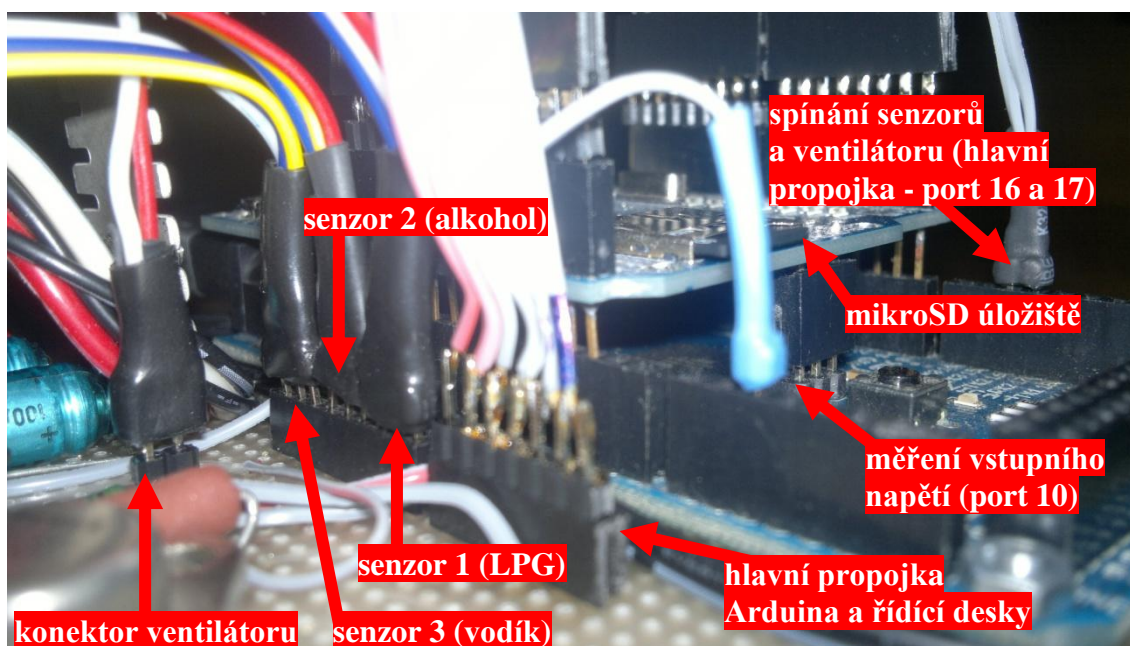
# A UŽIVATELSKÁ PŘÍRUČKA

Toto zařízení slouží k měření a následnému vyhodnocování charakteristik vůní. Jedná se o prototyp jednoduchého elektronického nosu, který lze v budoucnu rozšířit o další senzory a vylepšit tak jeho schopnosti detekce.

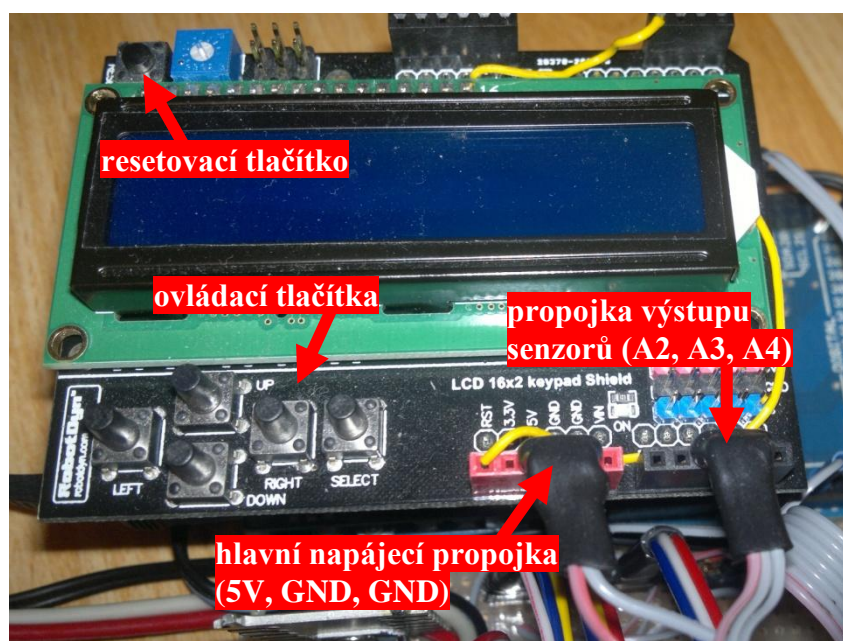
## A.1 Příprava zařízení

Před použitím zařízení je třeba se ujistit, zda jsou veškeré komponenty řádně připojeny. Nutno dodat, že zařízení využívá k ukládání dat externího paměťového úložiště - mikroSD karet, které je nutno před prvním použitím formátovat na souborový systém typu FAT32 a následně do takto připraveného úložiště zkopírovat systémové soubory zařízení, které lze nalézt v datové příloze této práce. Rovněž se nedoporučuje použití kapacity úložiště vyššího než 16 GB nebo vytváření více logických oddílů na médiu. Detail zapojení včetně slotu pro mikroSD karty je zobrazen na obr. 27 a obr. 28. Po připojení všech komponent lze zařízení bezpečně zapnout, připojením napájení. Jako zdroj napájení je nejvhodnější použití běžného šesti-článekového modelářského akumulátoru s konektorem Tamya. V případě potřeby může být použit libovolný zdroj, jehož napětí se pohybuje mezi sedmi až dvanácti volty. Proudová zátěž zařízení může dosahovat až jednoho ampéru, doporučuje se proto použít zdroj, který je schopen dlouhodobě poskytnout tento proud. Zařízení je ovšem funkční i v rozsahu napětí od šesti do třinácti voltů, ovšem jedná se o kritické limity, které nejsou vhodné pro dlouhodobé používání, především vyšší hodnoty napětí mohou zařízení poškodit. Dlouhodobým vystavením kritickým limitům napětí nebo při neposkytnutí dostatečného proudu může docházet ke značnému zkreslení výsledků. Je-li připojeno napětí přesahující kritické limity, zařízení nelze plně spustit a používat.

Ihned po zapnutí zařízení (připojení napájení) dochází k jeho automatické kontrole, kde se zjišťuje především stav napájení, funkčnost a zapojení senzorů nebo i dostupnost externího paměťového úložiště mikroSD. Veškeré informace o průběhu této kontroly uživatel vidí na displeji zařízení. V případě selhání některé části kontroly, je třeba tuto chybu ihned opravit (např. připojením chybějícího senzoru) a to lze v některých případech i za běhu zařízení (nedostatečné napájení, nepřipojený senzor), někdy je ovšem potřeba zařízení restartovat (chybějící mikroSD karta). Proběhne-li kontrola bez problémů (obvykle do deseti vteřin), je zařízení plně připraveno k použití a na displeji se zobrazí jednoduché menu, zařízení čeká na vstup uživatele. Zařízení lze kdykoli vypnout odpojením napájení, stane-li se tak v průběhu ukládání dat, je pravděpodobná jejich ztráta.



Obr. 27: Detail zapojení hlavních propojek



Obr. 28: Detail tlačítek a propojek

## A.2 Offline režim

Offline režimem lze označit veškerou činnost na zařízení až do chvíle, kdy uživatel spustí příkaz připojení k serveru. Offline režim je tedy zároveň výchozím režimem po spuštění zařízení. V tomto režimu uživatel zařízení ovládá prostřednictvím pěti tlačítek (obr. 28), kde tlačítka nahoru/dolů (up/down) se pohybuje v menu. Tlačítko zvolit

(select) slouží k potvrzení daného příkazu či volby a tlačítka vlevo/vpravo (left/right) lze volit parametr dané položky. Celé menu je stavěno následujícím způsobem: První řádek zobrazuje o jakou položku nebo příkaz se jedná, zatímco druhý řádek zobrazuje parametry nebo jiné volby dané položky. Menu obsahuje celkem dvanáct položek, které plní následující funkce:

**Senzory** - Zobrazuje aktuální stav senzorů, kde mohou nastat tři možné stavy: "Vypnutý", "Žhaví se", "Připravený". Zde může uživatel stiskem tlačítka select zapnout nebo vypnout senzory. Při stavu "Vypnutý" nelze provádět měření a automaticky je zapnut ventilátor, čímž čistí komoru od plynů ze starého měření, zařízení má tak minimální spotřebu energie. Stav "Žhaví se" signalizuje zapnuté senzory, ovšem bez možnosti měření, jelikož je třeba dosáhnout pracovní teploty senzorů, tento stav trvá sto vteřin, kde během této doby dojde i k vypnutí ventilátoru. Je vhodné již na začátku žhavicího procesu vložit měřený vzorek do komory. Uživateli se rovněž zobrazuje zbývající čas do konce žhavení senzorů. Po uplynutí této doby se stav senzorů změní na "Připravený" a je možné spustit měření pomocí další položky v menu.

**Měření** - Zobrazuje stav měření s uplynulou dobou od jejího začátku nebo v případě, že měření nemůže být spuštěno, zobrazí příčinu této skutečnosti. Stiskem tlačítka select je možné zapnout nebo vypnout proces měření. Ke spuštění měření je nutné, aby byly senzory ve stavu "Připravený". Data měření jsou ukládána do dočasné paměti a nedojde-li k uložení těchto dat prostřednictvím dané položky v menu, data jsou spuštěním nového měření nebo vypnutím zařízení ztracena. Vypnutím měření dojde automaticky i k vypnutí senzorů a spuštění čistícího procesu. Proces měření může trvat maximálně deset minut, poté dojde k jeho automatickému vypnutí.

**Senzor 1 až Senzor 3 (třetí až pátá položka menu)** - Slouží k zobrazení naposledy vzorkovaných dat daného senzoru. Rovněž lze každému ze senzorů nastavit maximální dovolenou odchylku od střední hodnoty, která se uplatní při porovnávání vzorků a identifikaci vůně. Nastavením alespoň jednoho z těchto limitů se vzorek automaticky změní na kalibrační, tedy na vzorový, pevně určující charakter dané vůně. Nastavení těchto limitů lze provést v editačním režimu, který se zapíná a vypíná stiskem tlačítka select. V tomto režimu uživatel tlačítka left/right přepíná mezi spodním a horním limitem, zatímco tlačítka up/down mění hodnoty těchto limitů.

**Vyhodnotit data** - Stiskem tlačítka select je spuštěn proces identifikace aktuálního vzorku, který se začne porovnávat se všemi ostatními kalibračními vzorky, je-li nalezena shoda, vypíše se název shodujícího se vzorku na druhý řádek, v opačném případě je zobrazena hláška "nenalezen".

**Uložit data** - Slouží k uložení aktuálních dat v dočasné paměti na externí mikroSD paměťové úložiště. Stiskem tlačítka select je spuštěn editační režim, kde uživatel zadává název, pod kterým má být vzorek uložen. Tlačítka up/down lze měnit znak na pozici kurzoru, zatímco tlačítka left/right lze kurzor posouvat. Vypnutím

editačního režimu se data uloží pod zadaným názvem. Jsou povoleny stejnojmenné názvy měření a nelze tak přepisovat stará data.

**Načíst data** - Tlačítka left/right lze procházet mezi dostupnými vzorky. Potvrzením pomocí tlačítka select je do dočasné paměti načten zvolený vzorek, se kterým lze dále manipulovat. Načtením nových dat jsou nenávratně ztracena předchozí data v dočasné paměti.

**Smazat data** - Tlačítka left/right lze procházet mezi dostupnými vzorky. Potvrzením pomocí tlačítka select je zvolený vzorek smazán. Reálně data v souboru stále existují a lze je v online režimu pomocí webového rozhraní obnovit nebo naopak smazat úplně.

**Připojit k webu** - Stiskem tlačítka select dojde k připojení ke vzdálenému serveru a přepnutí do online režimu. Jestliže zařízení zjistí neplatné nastavení IP adresy, pokusí se použít DHCP protokol. Druhý řádek uživatele informuje o aktuálním stavu připojení. V online režimu se rovněž nedoporučuje provádění veškerých činností na zařízení, jelikož zařízení lze plně ovládat z webového rozhraní.

**Nastavení IP** - Dovoluje uživateli nastavit pevnou IP adresu zařízení, která bude použita při online režimu. Stiskem tlačítka select je spuštěn editační režim, kde uživatel zadává adresu. Tlačítka up/down lze měnit číselnou hodnotu na pozici kurzoru, zatímco tlačítka left/right lze kurzor posouvat. Vypnutím editačního režimu se zadaná adresa uloží na externí úložiště.

**Adresa serveru** - Zde uživatel v případě potřeby mění adresu vzdáleného serveru, kterou lze zadávat jak v číselné tak i písemné formě (příklad DNS serverem bude automaticky vyžádán). Stiskem tlačítka select je spuštěn editační režim, kde uživatel zadává název serveru, ke kterému se má zařízení v online režimu připojit (bez podadresáře "enose"). Tlačítka up/down lze měnit znak na pozici kurzoru, zatímco tlačítka left/right lze kurzor posouvat. Vypnutím editačního režimu se adresa serveru uloží na externí úložiště. Maximální počet znaků je omezen na 64.

### A.3 Online režim

Zařízení je v online režimu, jestliže došlo k úspěšnému připojení ke vzdálenému serveru, tedy jestliže výsledkem akce "Připojit k webu" je "Připojen". V tomto režimu zařízení přijímá akce ze serveru a následně je provádí, proto se nedoporučuje zařízení ovládat jinak, než ze serveru - z webového rozhraní. Veškeré soubory, nutné ke správné funkci webového rozhraní a zprostředkování komunikace mezi zařízením jsou uloženy v datové příloze této práce. Funkčnost těchto kódů byla testována na minimalistickém webovém serveru "Lighttpd" verze 1.4.32 s PHP modulem verze 5.2.17.



**Seznam veskerých zarizeni:**

Cislo	Vyber	MAC adresa	Posledni pripojeni	Stav
1	<input checked="" type="radio"/>	DEADBEEFFEED	18-05-2016 - 23:32:26	Offline - Senzory vypnuty, nemeri se
2	<input type="radio"/>	DEADBEEFBEEF	17-05-2016 - 19:56:11	Offline - Senzory vypnuty, nemeri se

**Provest / naplanovat akci:**Odeslat data na server  **Obnovit**

Vygenerovano: 22-05-2016 - 20:22:19

Obr. 29: Ovládací webové rozhraní

Po připojení uživatele na adresu serveru (<http://dazzles.no-ip.org/enose>), je vyžádáno přístupové heslo, které může správce serveru změnit v proměnné "srvPass" kódu "client.php", kde je toto heslo uloženo ve formě MD5 šifry. Po přihlášení do tohoto systému je uživatel po hodinové nečinnosti automaticky odhlášen. Ve výchozím nastavení je tímto heslem řetězec: "qwertz". Po zadání správného hesla je zobrazena tabulka s veškerými zařízeními, které jsou nebo v minulosti byly k serveru připojeny (obr. 29). Jednotlivá zařízení lze od sebe odlišit pomocí jejich MAC adres. Dále zde uživatel nalezne čas posledního připojení zařízení a poslední známý stav zařízení. Výběrem některého ze zařízení je uživateli umožněno naplánovat pro toto zařízení určitou akci nebo provést jiné akce na straně serveru. Uživatel může provádět celkem jedenáct různých akcí, které jsou přidávány do fronty úloh, které zařízení po připojení k serveru postupně vykoná. Následuje popis jednotlivých akcí:

**Odeslat data na server** - Zařízení odešle veškerá naměřená data na server, kde jsou následně uložena a připravena pro další operace.

**Odeslat konfiguraci na server** - Zařízení odešle soubor své konfigurace na server, kde je následně uložen a připraven pro další operace. Tento soubor obsahuje nastavení IP adresy a adresy vzdáleného serveru.

**Zapnout/vypnout senzory** - Zapíná nebo vypíná senzory. Funkčnost stejná jako v offline režimu (viz. offline režim - senzory).

**Spustit/zastavit měření** - Zapíná nebo vypíná automatické měření. Uživatel zadává parametry tohoto měření jako je čas, název měření nebo limity jednotlivých senzorů, má-li se jednat o kalibrační vzorek. Vzorek lze rovněž uložit pod přesným časem spuštění měření, zvolením pole "Název podle data spuštění měření". Tuto položku je vhodné použít především při plánování několika úloh pro zařízení, které má být v budoucnu připojeno.

**Zobrazit naplánované akce** - Zobrazí frontu naplánovaných akcí pro dané zařízení s možností smazání některé z nich.

**Zobrazit data** - Zobrazí tabulku veškerých naměřených dat daného zařízení, které jsou uloženy na straně serveru. I měření, která byla zařízeními v minulosti smazána jsou zde zobrazena a lze s nimi dále pracovat. Uživatel může data obnovovat, mazat,

měnit jejich název či limity nebo data zobrazovat a následně porovnávat nejen s kalibračními, ale i všemi ostatními vzorky. Zobrazení dat je provedeno graficky, kde jsou tečkami vyznačeny i limity jednotlivých senzorů u kalibračních vzorků nebo střední hodnoty u běžných vzorků. Při volbě porovnání se všemi kalibračními, je spuštěn stejný algoritmus detekce zvoleného vzorku, jako při vyhodnocování dat v offline režimu (viz. offline režim - vyhodnotit data).

**Restart** - Naplánuje restart zařízení. Zařízení se tímto rovněž odpojí ze serveru.

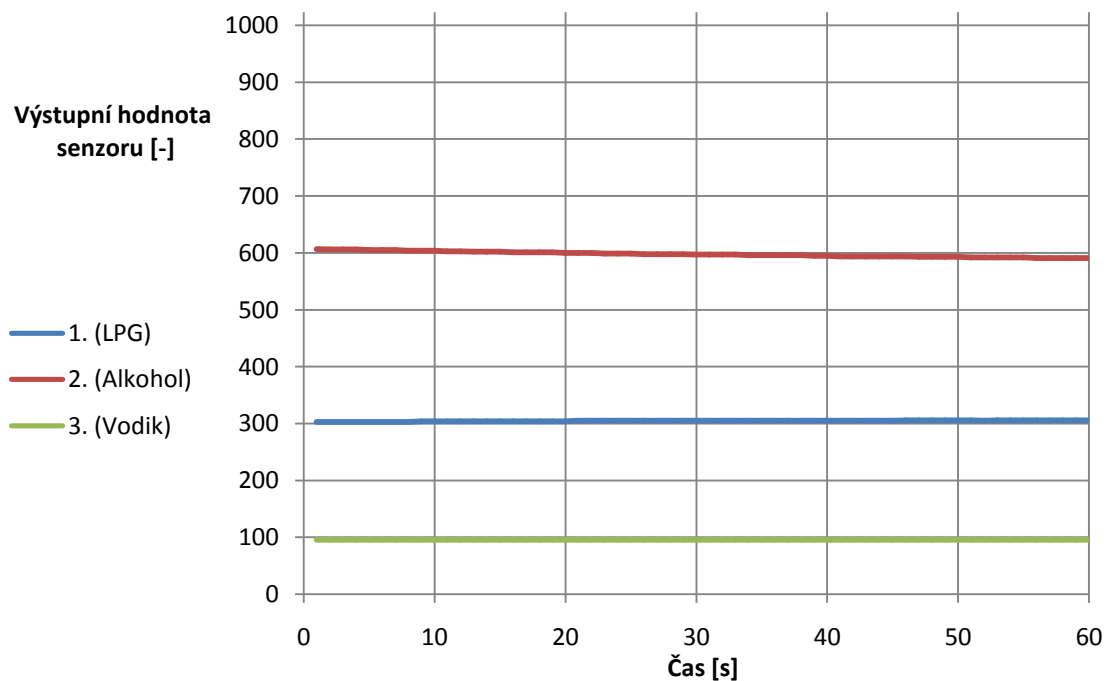
**Stáhnout data (.xls)** - Stáhne veškerá data, která jsou uložena na serveru pro dané zařízení ve formátu .xls.

**Synchronizovat data ze serveru** - Odešle data ze serveru do zvoleného zařízení, kde jsou stará data automaticky přepsána.

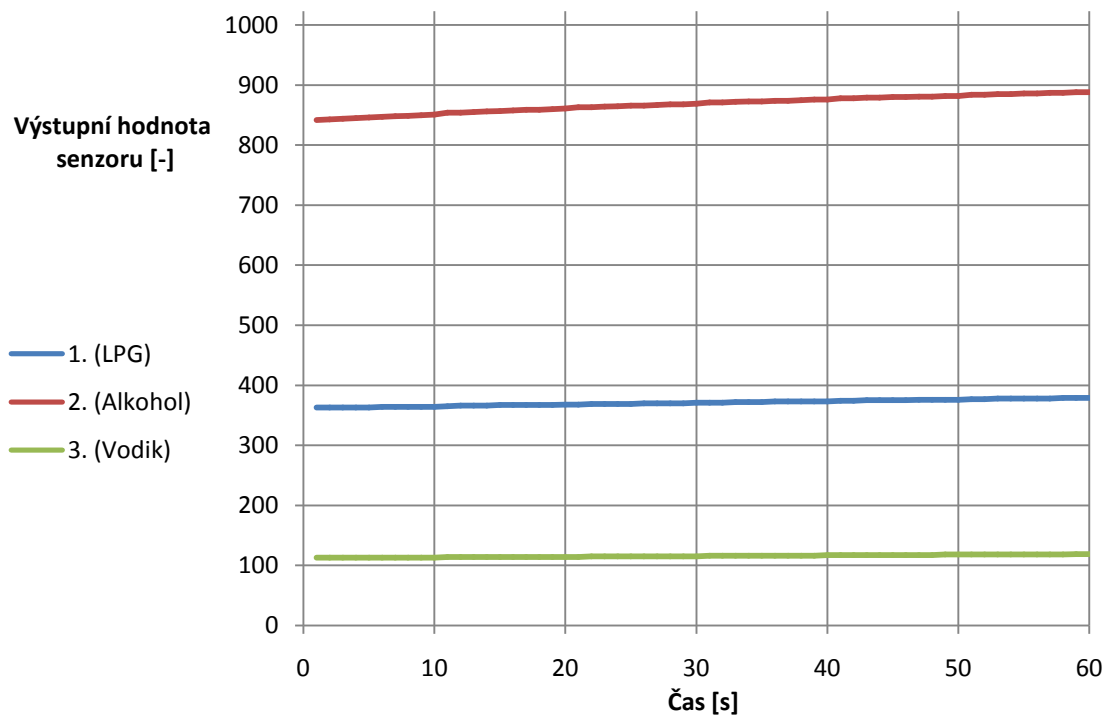
**Synchronizovat konfiguraci ze serveru** - Odešle konfiguraci uloženou na serveru do zvoleného zařízení, čímž je přepsána stará konfigurace.

**Upravit konfiguraci** - Upravuje konfigurační soubor zvoleného zařízení, který je uložen na serveru.

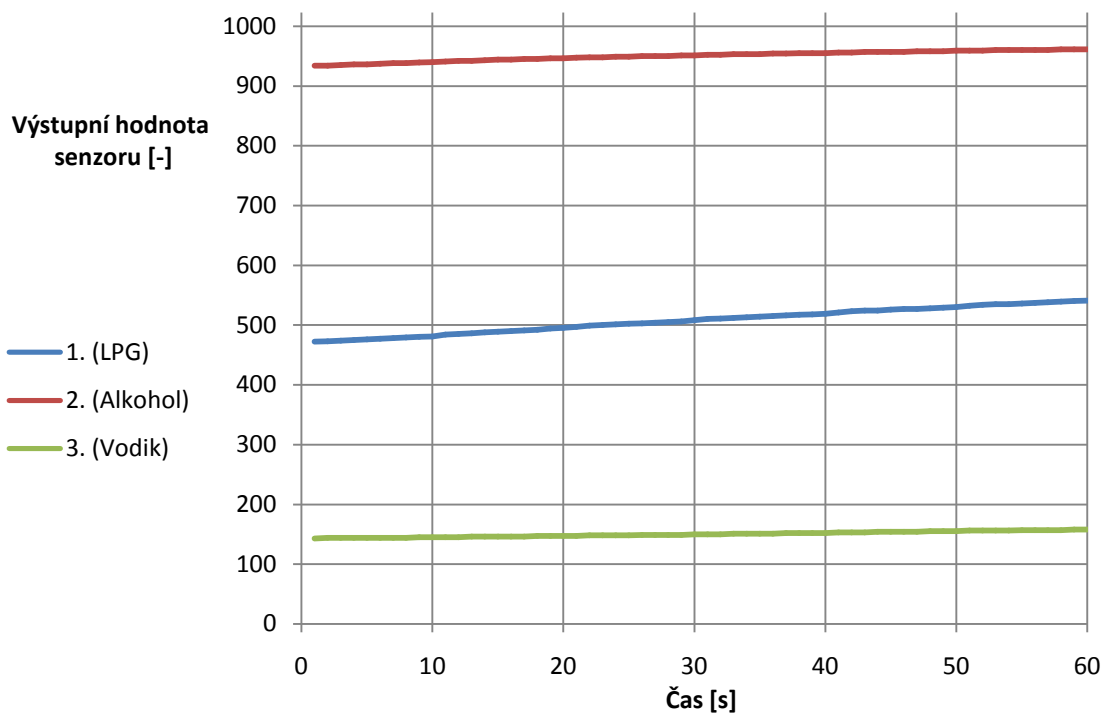
## B VÝSLEDKY MĚŘENÍ



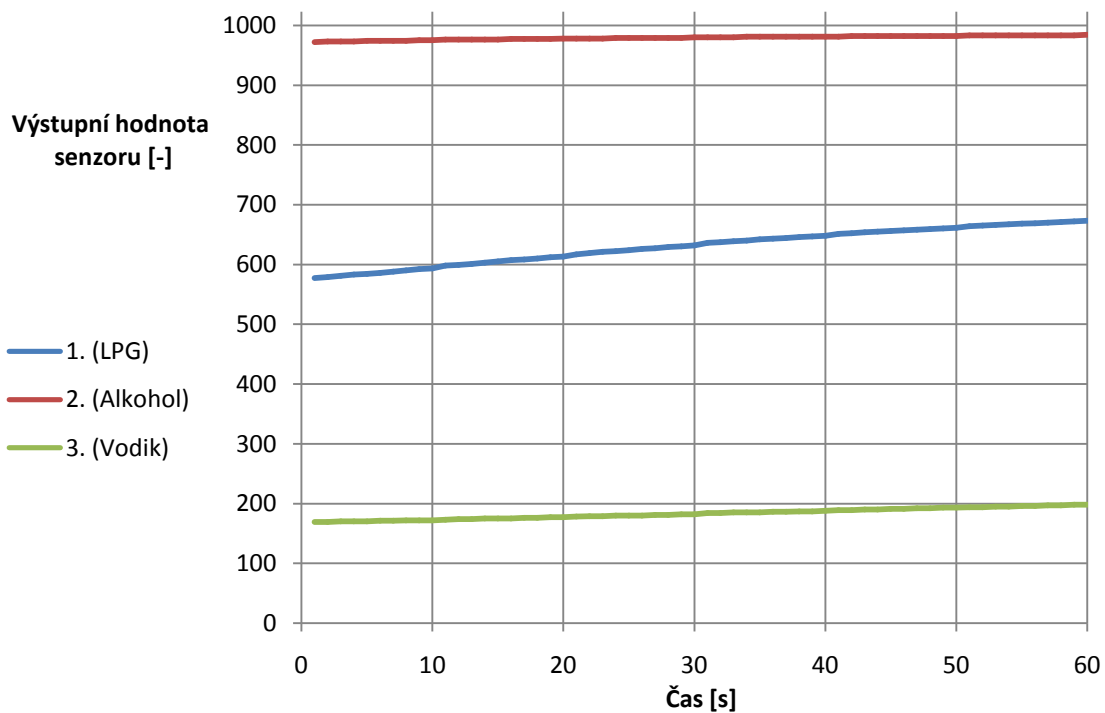
Obr. 30: Charakteristiky vzduchu, naměřené před měřením ostatních vzorků



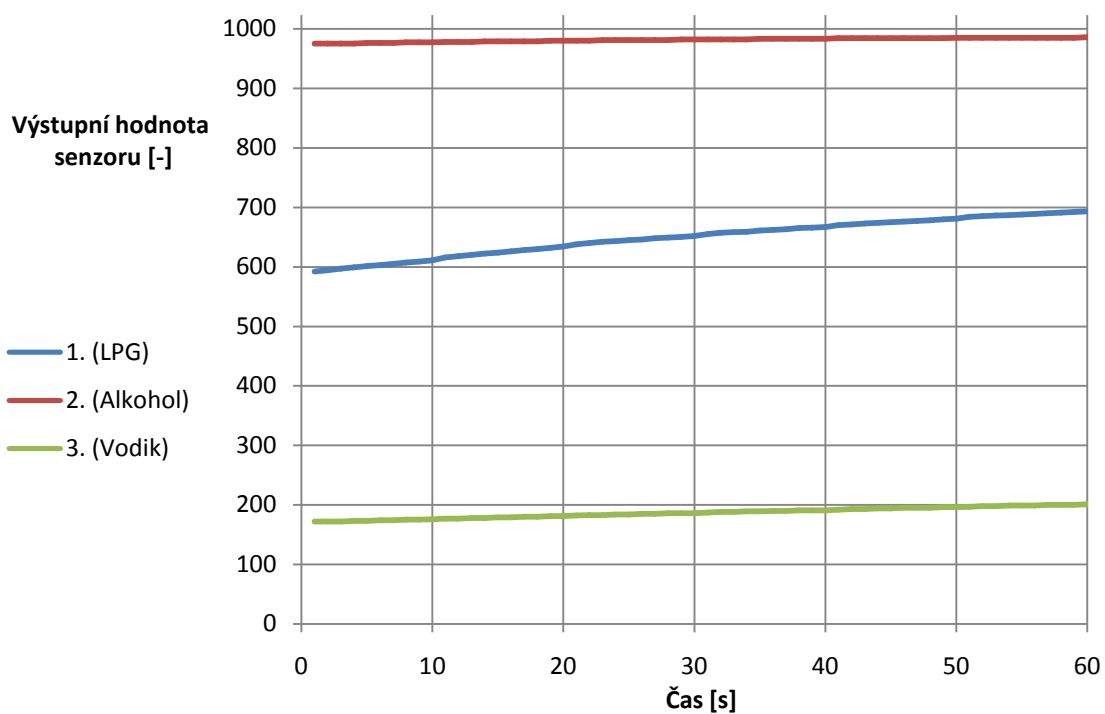
Obr. 31: Charakteristiky potravinářského rumového aromatu



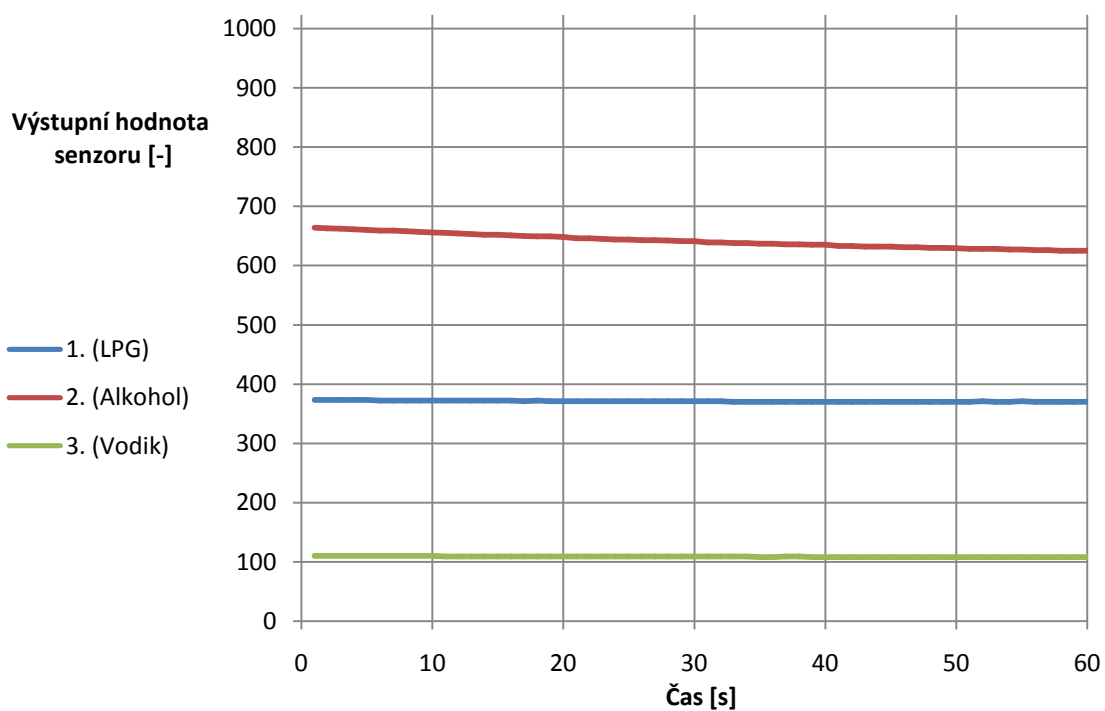
Obr. 32: Charakteristiky potravinářského kokosového aromatu



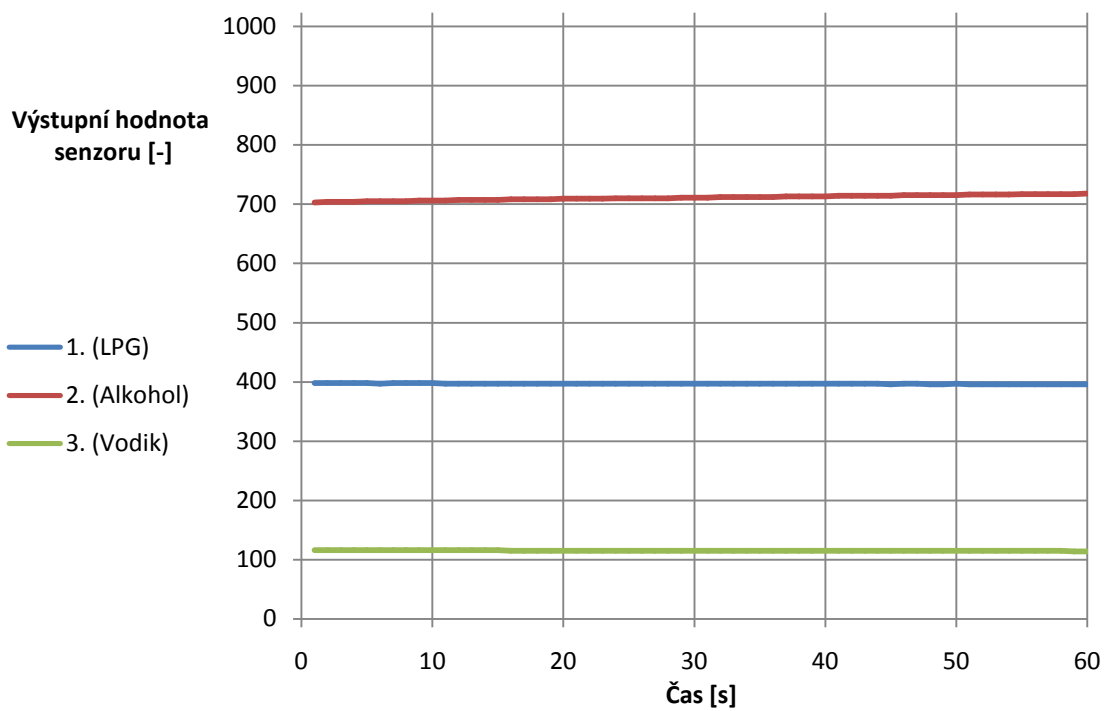
Obr. 33: Charakteristiky potravinářského vanilkového aromatu



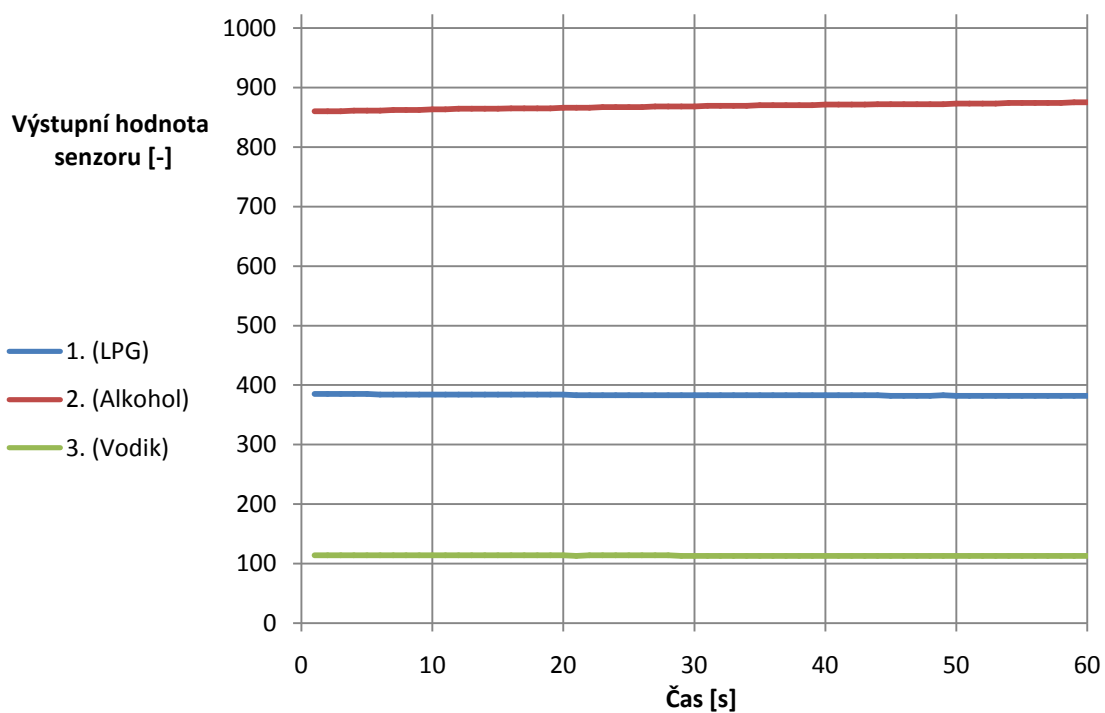
Obr. 34: Charakteristiky potravinářského citronového aromatu



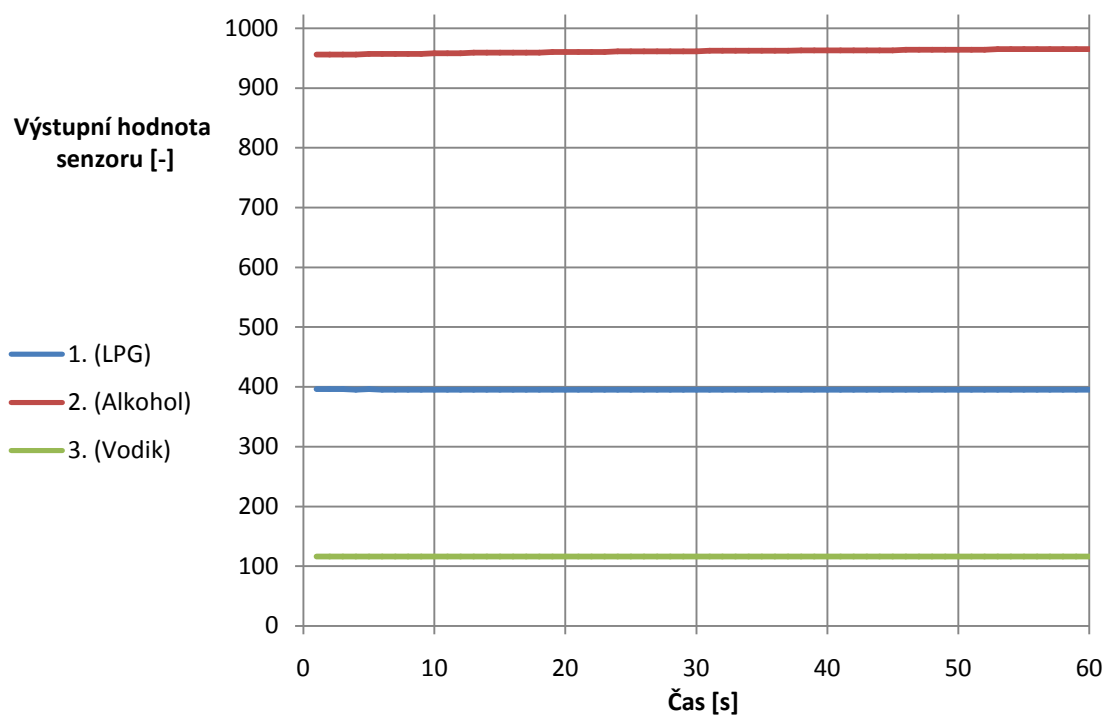
Obr. 35: Charakteristiky vzduchu, naměřené v průběhu měření ostatních vzorků



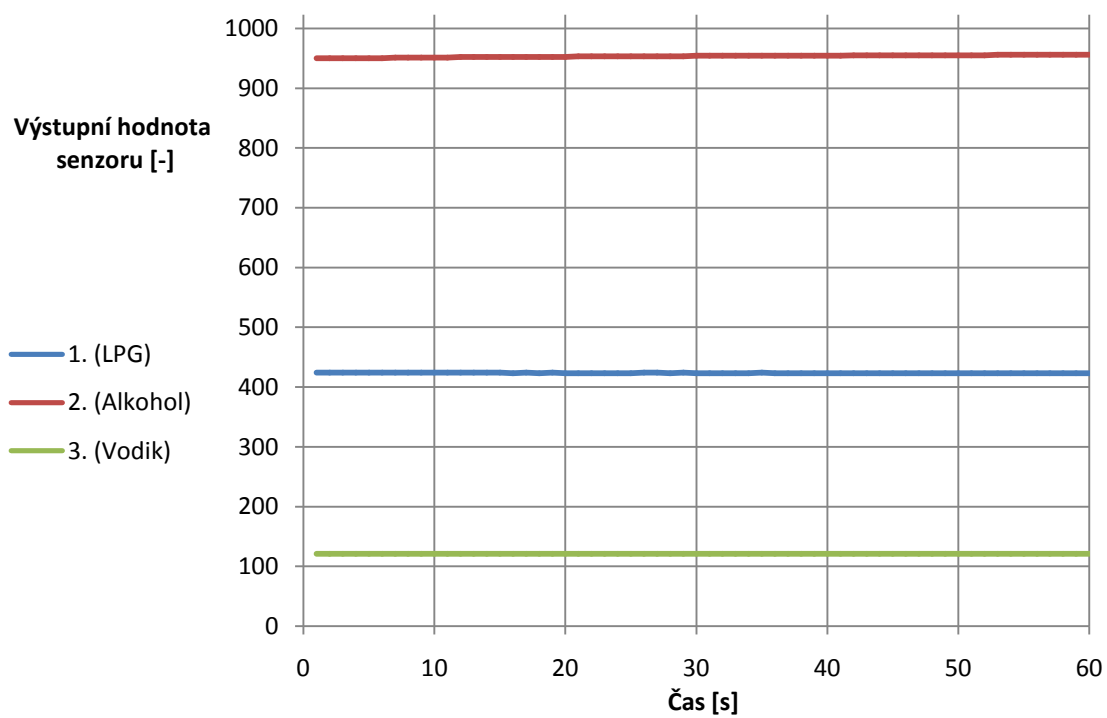
Obr. 36: Charakteristiky vůně jablečného vonného oleje



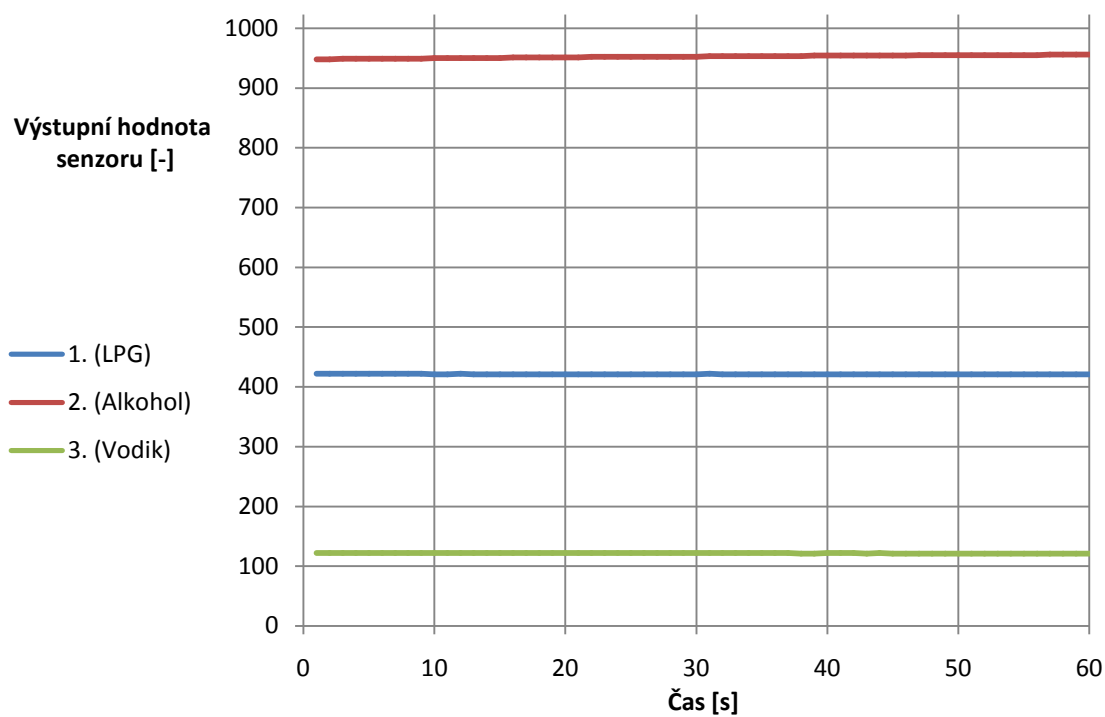
Obr. 37: Charakteristiky vůně vanilkového vonného oleje



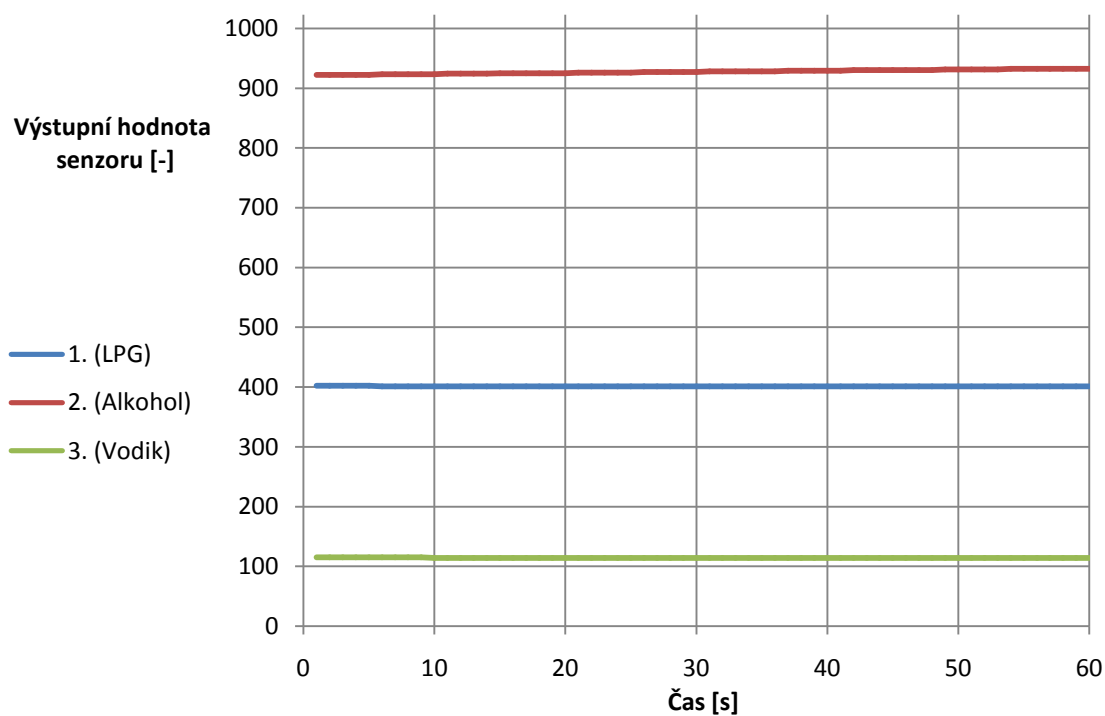
Obr. 38: Charakteristiky vůně pomerančového vonného oleje



Obr. 39: Charakteristiky vůně jablečně-skořicového vonného oleje

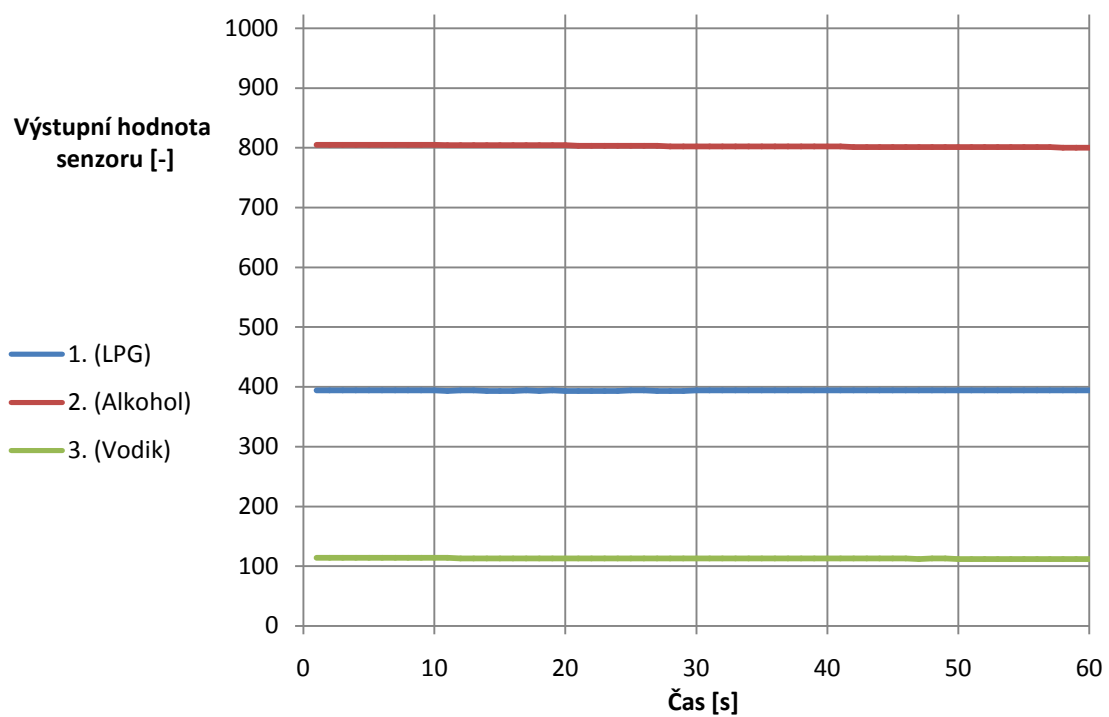


Obr. 40: Charakteristiky vonného oleje s vůní zimního světa



Obr. 41: Charakteristiky vůně vonného oleje s vůní santalového dřeva





Obr. 42: Charakteristiky vzduchu, naměřené po měření ostatních vzorků