



Středoškolská technika 2018

Setkání a prezentace prací středoškolských studentů na ČVUT

Zavlažovací systém skleníku

Lukáš Holeka

Střední průmyslová škola elektrotechnická a Vyšší odborná škola Pardubice

Karla IV. 13, 530 02 Pardubice

Čestné prohlášení

„Prohlašuji, že jsem závěrečnou maturitní práci vypracoval samostatně a použil jsem literárních pramenů a informací, které cituji a uvádím v seznamu použité literatury a zdrojů informací.“

V Pardubicích dne

.....

Lukáš Holeka

Anotace

Práce se zabývá konstrukcí a návrhem zavlažovacího systému skleníku pomocí jednočipového mikroprocesoru Arduino. Skleník informuje o vlhkosti půdy v jednotlivých sektorech, vlhkosti a teploty vzduchu, datumu a času. Tyto hodnoty jsou zobrazeny na LCD displeji, připojeného k Arduino. V závislosti na aktuálních hodnotách ve skleníku, jsou jednotlivé sektory zalévány.

Klíčová slova

Arduino Mega2560 R3, Automatický skleník, DHT22 senzor, LCD displej, Půdní vlhkoměr

Annotation

The work deals with construction and design of the irrigation system Greenhouse Using the Arduino free microprocessor. The greenhouse informs about soil moisture in individual sectors, humidity and air temperature, date and time. These values are displayed on the LCD screen, attached to the Arduino. In Depending on current Values in the Greenhouse Each, sector is watered.

Keywords

Arduino Mega2560 R3, Automatic Greenhouse, DHT22 sensor, LCD display, Soil hygrometer

Seznam použitých značek a symbolů

Značka	Jednotka v soustavě SI	Název fyzikální veličiny
V	Volt	Napětí
A	Ampér	Elektrický proud
°C	Stupeň Celsia	Teplota
Ω	Ohm	Elektrický odpor

Seznam použitých zkratek

Zkratka	Vysvětlení
DPS	Deska plošného spoje
I2C	Sběrnice, která zajišťuje komunikaci mezi procesory. Je tvořena dvojicí signálových vodičů. Těmi jsou SDA neboli serial data a SCL což je serial clock.
PWM	Pulzně šířková modulace.

Obsah

1. Úvod	8
2. Skleník	9
2.1. Základní vlastnosti	9
2.2. Funkce skleníku	9
3. Arduino	10
3.1. Programovací prostředí a jazyk.....	11
3.2. Popis programovacího prostředí	11
3.3. Jazyk Wiring	13
3.3.1. Knihovny	13
3.3.2. Komentáře	14
3.3.3. Proměnné.....	14
3.3.4. Datové typy	15
3.3.5. Podmínka if().....	16
3.3.6. Cyklus while()	17
3.3.7. Cyklus for()	17
3.3.8. Delay().....	17
4. Použité komponenty	18
4.1. Řídící jednotka	18
4.2. Vstupní periferie.....	19
4.2.1. DHT 22 – Senzor teploty a vlhkosti okolí.....	19
4.2.2. RTC DS3231	19
4.2.3. Půdní vlhkoměr	20
4.2.4. Ultrazvukový měřič vzdálenosti – Měření hladiny vody	21
4.3. Výstupní periferie.....	22
4.3.1. Čerpadlo	22
4.3.2. Servomotor – Pootevírání okna	22
4.3.3. LCD displej	23
5. Praktická část	24
5.1. Blokové schéma	24
5.2. Popis displeje	24
5.3. Mechanická konstrukce.....	25
5.4. Zapojení pinů	28
5.4.1. Vstupní periferie	28

5.4.2. Výstupní periferie	28
5.5. Návod k provozu a jeho průběh	29
5.6. Elektronická rozpiska součástek	32
5.7. Mechanická rozpiska součástek	32
6. Software.....	33
6.1. Struktura programu	33
6.2. Knihovny.....	34
6.3. LCD displej	34
6.4. Zalévání.....	34
6.5. Pootevírání okna.....	35
6.6. Hladina vody v nádržích	37
7. Oživení výrobku.....	38
8. Závěr.....	39
9. Přílohy	40
9.1. Seznam obrázků	40
9.2. Seznam použité literatury	42
9.2.1. Seznam použité literatury – obrázky	43
9.3. Seznam grafických příloh	44
9.4. Seznam souborů na CD	45

1. Úvod

Automatický skleník slouží pro lidi, kteří mají málo volného času a nemůžou se plně věnovat své úrodě. Takovéto zařízení, může využít každý člověk, který má málo času. Nebo je na delší dobu pryč a nemá možnost zalévat a starat se o svou úrodu. Zavlažovací systém skleníku je plně automatizovaný a bez lidské pomoci se sám dokáže dlouhodobě starat o úrodu. Toto zařízení nám sleduje a informuje o vlhkosti půdy v jednotlivých sektorech dále vlhkost a teplotu vzduchu uvnitř skleníku a také datumu a času, pomocí RTC modulu. Jelikož RTC modul obsahuje 3 V baterii, díky které datum a čas stále běží i přes odpojení Arduina od napájení. Všechny tyto informace, budou pro uživatele zobrazeny na LCD displeji, který je připojen k Arduinu. Skleník nás bude také informovat o hladině vody v obou nádržích, které budou indikovat tři různobarevné diody. Pomocí servomotorů je možné pootočit okno skleníku v závislosti na teplotě ovzduší. Jednotlivé sektory, budou zalévány pomocí čerpadel, v závislosti na hodnotách vlhkosti půdy. Celý tento systém bude řízen pomocí jednodeskového počítače Arduino Mega 2560 R3 s čipem ATmega2560. Zavlažovací systém skleníku jsem si vybral, jelikož mi to přišel velmi zajímavý a užitečný projekt, který ulehčí každodenní práci a ušetří spoustu času.

2. Skleníky

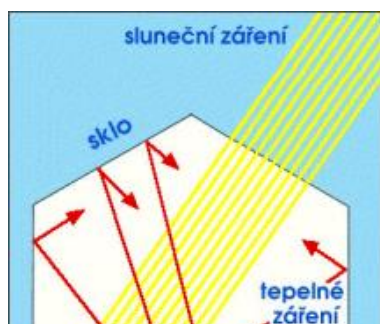
Tato kapitola se zabývá důvodem používání skleníku a způsobem, jakým skleník funguje.

2.1. Základní vlastnosti

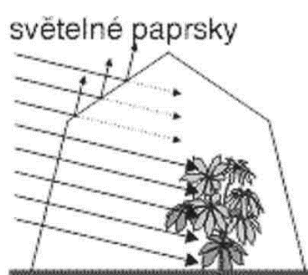
Skleníky se využívají k pěstování rostlin, které potřebují specifické klimatické podmínky. Díky skleníku jsou uvnitř mnohem vyšší teploty než mimo skleník. Proto nám uvnitř skleníku rostou rostliny rychleji. Nejenže, rostliny ve skleníku rostou rychleji, ale také jsou chráněny před nepříznivými podmínkami počasí jako je déšť, vítr a mnoho dalších. Pokud chceme zvolit stěny skleníku, můžeme volit buď sklo nebo polykarbonát. Je uváděno, že pro květiny je lepší použít sklo, ale pro zeleninu a vytápěné skleníky je výhodnější polykarbonát. Polykarbonát je dnes více využívaný, jelikož propouští více světla než sklo a také je odolnější. [1]

2.2. Funkce skleníku

Princip fungování skleníku je založen na takzvaném skleníkovém efektu. Před průchodem světelného záření přes sklo (polykarbonát), má světlo krátkou vlnovou délku. Pokud světlo dopadne na sklo (polykarbonát) skleníku, část paprsků se odrazí od stěny skleníku a část projde přímo dovnitř skleníku. Dále se energie světelného záření uvnitř skleníku přeměňuje na energii tepelného záření a ta se odrazí od jeho stěn. Tím se zvyšuje teplota uvnitř skleníku. Aby se zajistil maximální průnik slunečních světelných paprsků, musí se zvolit co nejlepší úhel, který nám zaručí kolmé dopadání slunečního záření. [1] [2]



Obr. 1 Skleníkový efekt [1]



Obr. 2 Odras dopadajícího záření [2]



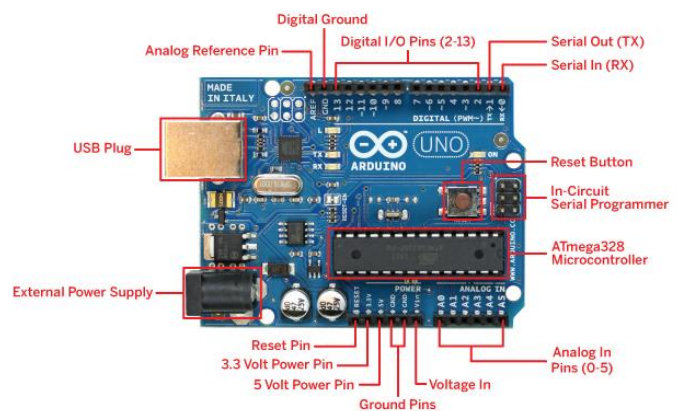
3. Arduino

Arduino je jednodeskový počítač, který využívá programovatelný mikrokontroler od firmy Atmel s názvem ATmega. Arduino vzniklo roku 2005 v severozápadní Itálii ve městě Ivrea. Jeho zakladatelé jsou Massimo Banzi a David Cuartielles, kteří pojmenovali celý projekt po Arduinovi Ivrejském. Jednalo se o významnou historickou postavu města. Ty se snažili vytvořit prototypovací platformu pro studenty a tím jim pomoci k rychlému vývoji. Nejenže, si můžeme koupit různé desky od této platformy, ale také je možnost, díky různým schémátům, vytvořit si vlastní Arduino doma.

Arduino je rozděleno na hardwarovou část a softwarovou část. Hardwarovou částí jsou myšleny všechny vytvořené desky, různých druhů s mikročipem ATmega. Softwarovou částí je program, vytvořený pro programování Arduina, který má název Arduino IDE.

Arduino desky jsou rozděleny do několika typů. Dělíme je podle velikostí, podle velikosti paměti, dále rozdílný počet pinů u různých desek a mnoho dalších různých parametrů. Jedno z nejpoužívanějších typů je Arduino UNO. Také se často používá Arduino Mega2560, která má větší paměť a poskytuje také více pinů. Mezi další patří například Arduino Nano, Leonardo, Due a mnoho dalších typů desek. K připojení desky k počítači využíváme USB kabel, přes který nahráváme program do Arduina. Desky se k počítači připojují pomocí USB kabelu, který slouží jak pro nahrání kódu, tak pro napájení v době programování. Napájecí napětí desky je od 5 V–12 V.

Jak už bylo zmíněno, nejpoužívanějším typem desky je Arduino Uno, u kterého je možno vyjmout mikrokontroler a nahradit ho novým. Jeho mikročip je uváděn pod názvem ATmega328. Jeho Flash paměť má 32 KB. Obsahuje 14 digitálních a 6 analogových vstupů/výstupů. Má 6 PWM. Výrobce je uváděno, že maximální odebraný proud z jednoho pinu je 40 mA. Výrobce, ale doporučuje hodnotu odebraného proudu pouze 20 mA, aby nedošlo k zničení desky. Arduino může řídit malé, ale i velké, vymyšlené projekty. [3]



Obr. 3 Arduino Uno [3]

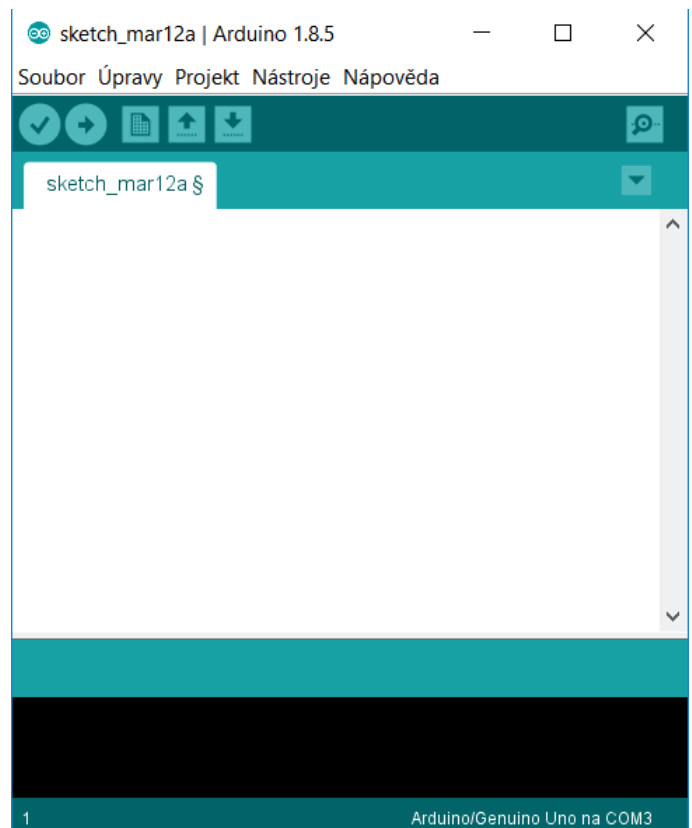
3.1. Programovací prostředí a jazyk

Programovací prostředí pro Arduino se nazývá Arduino IDE (integrated development environment = integrované vývojové prostředí). Celé prostředí je napsáno v jazyce Java. Programuje se v jazyce C nebo C++. Jeho nejrozšířenější knihovna se nazývá Wiring a většina ji považuje za samotný jazyk tohoto programu. [4]

3.2. Popis programovacího prostředí

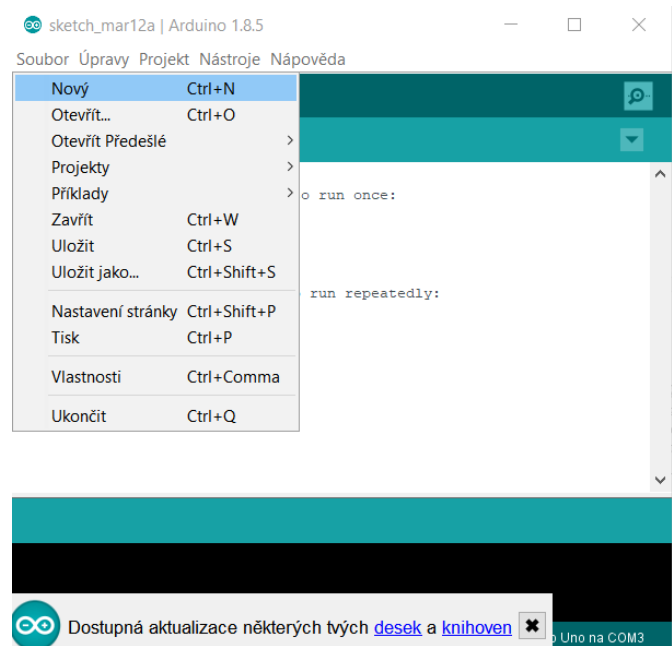
V této části se zaměříme na obecný popis vývojového prostředí.

Na Obr. 4 je vývojové prostředí. Na druhém řádku je několik ikon, kde první ikona zleva se nazývá Verify a spouští kontrolu celého programu. Pokud nalezne chybu, vypíše chybu v černém prostoru dole a syntaxi ji zvýrazní. Vedle je ikonka nazývající se Upload. Po kliknutí na tuto ikonku, se zkontroluje program a pokud bude bez chyby, program se nahraje do připojeného Arduina. Další ikonkou je ikonka New, která vytvoří nový soubor pro programování. Vedle nalezne ikonu Open. Ta slouží k otevření nabídky, z které si můžeme otevřít jakýkoliv vlastní program. Vedle ikony Open je ikona Save, která nám uloží aktuální program. Úplně vpravo nalezneme poslední ikonku, která označuje Seriál Monitor. Po kliknutí na tuto ikonu, se otevře sériový monitor. [4]



Obr. 4 vývojové prostředí Arduina IDE

Pokud chceme vytvořit nový program, klikneme v prvním řádku na ikonu Soubor a poté klikneme na Nový. Na začátku vidíme dva bloky. Prvním je blok void setup() a druhým blok loop(). To se ukazuje na Obr. 6. Do bloku void setup() píšeme kód u kterého chceme, aby se provedl pouze jednou na začátku programu. U bloku loop() píšeme kód, který se neustále opakuje. Pokud by jeden z těchto bloků byl smazán, došlo by při kompilaci k chybě viz Obr. 7. [4]



Obr. 5 Vytvoření nového programu

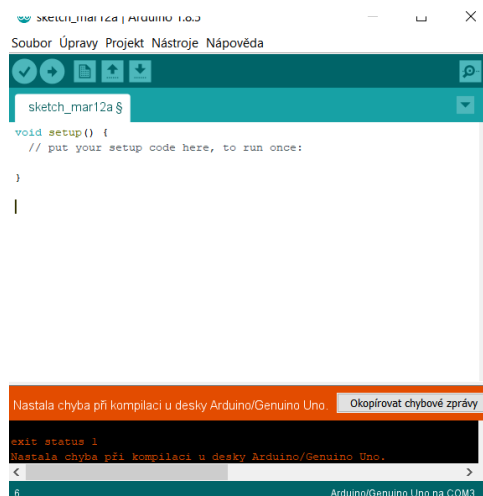
```

void setup() {
  // put your setup code here, to run once:
}

void loop() {
  // put your main code here, to run repeatedly:
}

```

Obr. 6 Ukázka hlavního kódu



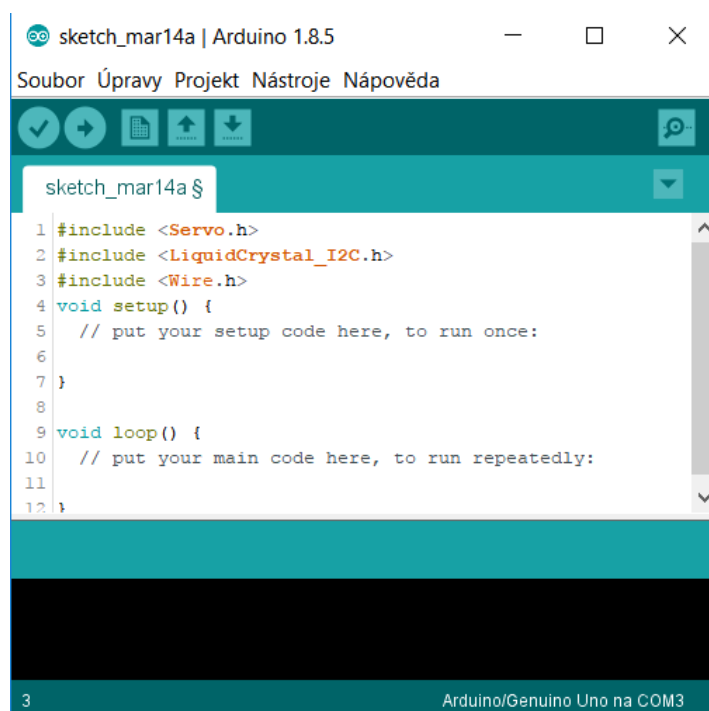
Obr. 7 Ukázka chyby při kompilaci desky

3.3. Jazyk Wiring

V této části si popíšeme základní informace, které byly potřeba k programování programu pro Zavlažovací systém skleníku. [4]

3.3.1. Knihovny

Knihovny jsou velmi důležité, pokud píšeme vlastní složitější kód. Sdružují více souborů se zdrojovými kódy, které nám zjednoduší a zpřehlední náš program, pokud budeme připojovat moduly, jako displej, servomotory nebo také různé senzory. Některé knihovny máme k dispozici už po stáhnutí programu Arduino IDE. Další knihovny si musíme stáhnout z internetu a importovat do programu. To uděláme po kliknutí na Projekt, dále Přidat knihovnu a zvolíme Přidat .ZIP Knihovnu. Pokud chceme přidat knihovnu do našeho programu, je třeba před void setup () napsat #include <název knihovny.h>. Na Obr.8 je vidět přidání knihoven pro LCD displej a pro servomotory. [4]



```
sketch_mar14a | Arduino 1.8.5
Soubor Úpravy Projekt Nástroje nápověda
sketch_mar14a §
1 #include <Servo.h>
2 #include <LiquidCrystal_I2C.h>
3 #include <Wire.h>
4 void setup() {
5     // put your setup code here, to run once:
6
7 }
8
9 void loop() {
10    // put your main code here, to run repeatedly:
11
12 }
```

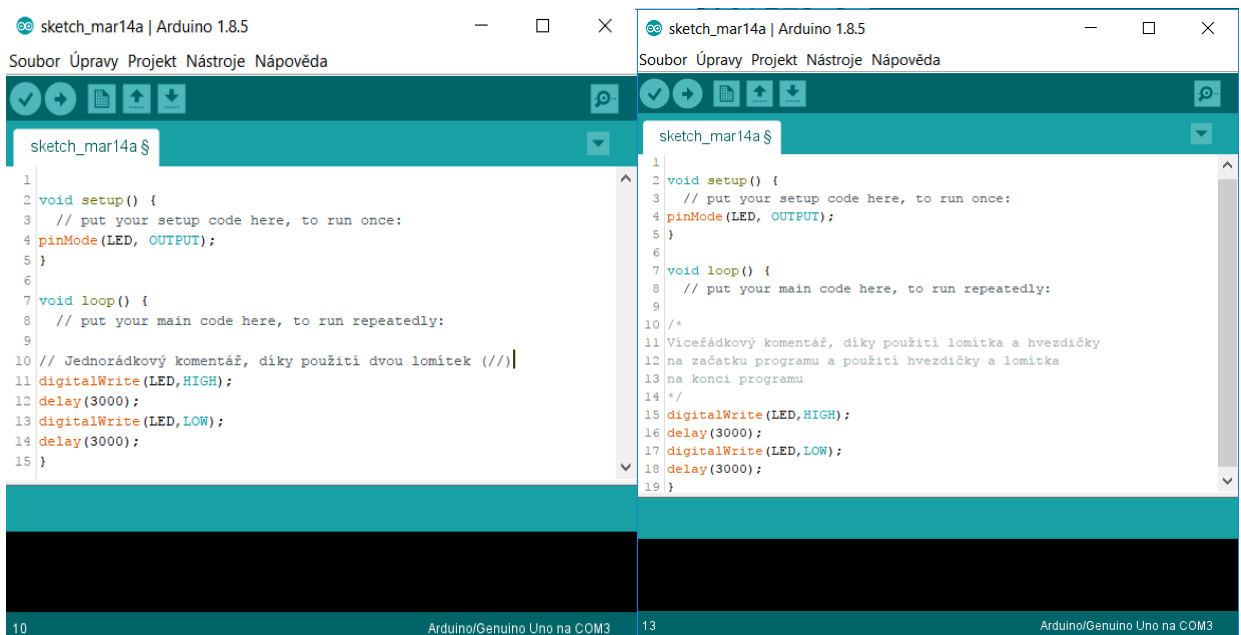
3 Arduino/Genuino Uno na COM3

Obr. 8 Ukázka přidávání knihoven do Arduina IDE

3.3.2. Komentáře

Komentáře slouží k různým poznámkám v programu, ale nejsou součástí kódu. Pokud by, jsme chtěli zapsat komentář kamkoliv do programu, musíme použít dvě lomítka (//), za kterou napíšeme vlastní poznámku, aniž by se stala součástí kódu.

Tento komentář je pouze jednořádkový. Můžeme, ale také zapsat komentáře víceřádkové. Ty se píše pomocí lomítka s hvězdičkou na začátku komentáře a hvězdičkou s lomítkem na konci komentáře viz Obr. 10. [4]

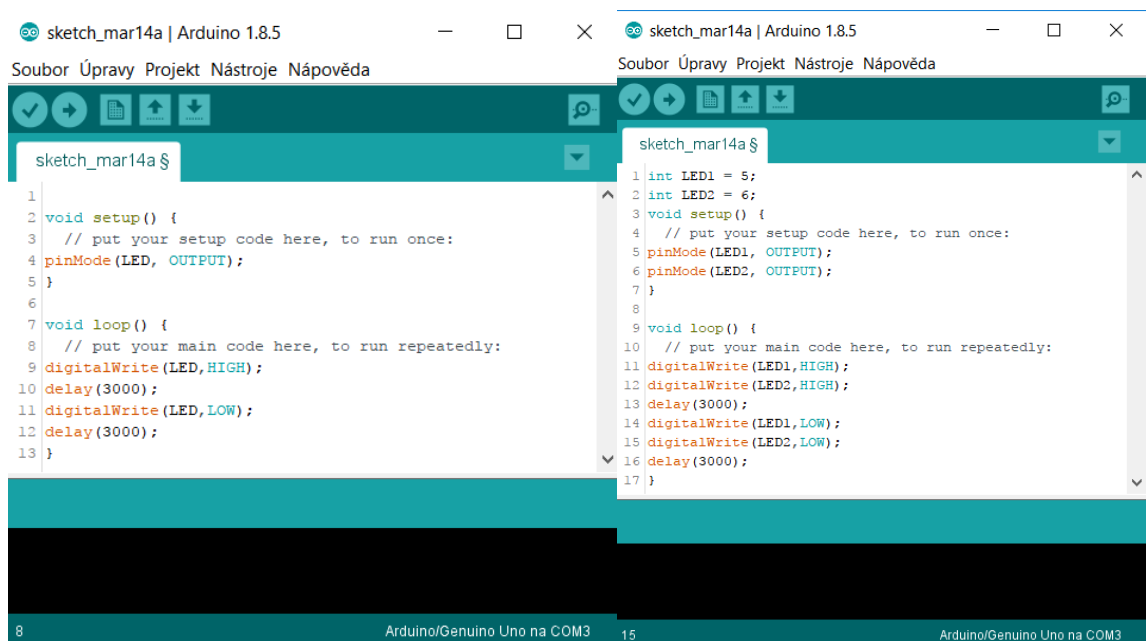


Obr. 9 Jednořádkový komentář

Obr. 10 Víceřádkový komentář

3.3.3. Proměnné

Proměnná je definována, jako místo v paměti, ve které jsou uloženy data. Proměnná má vlastní datový typ, jméno a hodnotu. Proměnný můžeme používat v programech, kde se nám opakuje jedna stálá hodnota dokola. Nejdříve je třeba proměnnou definovat, a to buď lokálně nebo globálně. Lokální způsob definice je, že proměnná platí pouze v části programu. Ale u globální definice, nám proměnná platí v celém programu. Deklarace proměnné má tvar integer, která se v programu píše ve zkratce `int`. Pokud chcete, aby proměnná se dala používat všude v programu, musíme ji vytvořit vně všech funkcí, jak `void setup()` tak i `loop()`. Na Obr. 11 vidíme, jak vypadá program bez proměnné hodnoty a na Obr. 12 vidíme přepsání toho samého kódu s použitím proměnné. Na Obr. 11 a Obr. 12 je příklad se zapínáním dvou LED diod. [4]



Obr. 11 Program bez použití proměnných

Obr. 12 Program s využitími proměnných

3.3.4. Datové typy

Jak bylo řečeno v minulé části, každá proměnná má vlastní datový typ. Datové typy slouží k tomu, abychom věděli, jaké data můžeme v proměnné najít. Níže uvedeme základní datové typy.

Nejdříve si vyjmenujeme číselné datové typy. Těmi jsou:

Byte – tento datový typ o velikosti 8 bitů uchovává celá čísla v rozsahu od 0 až po 255

Integer – jak bylo zmíněno, v programech se využívá zkratka int. Jedná se o velmi používaný datový typ, který ukládá celá čísla. Jeho rozsah závisí na použitém procesoru. U procesoru Atmega uchovává 16-bit hodnotu. Jeho rozsah je pak tedy od -32 768 až do 32 767.

Long – Jedná se o datový typ, který uchovává celá čísla 32 bitových hodnot. Rozsah tohoto datového typu je od -2 147 483 648 až do 2 147 483 647.

Float – Uchovává čísla s desetinnou čárkou a jeho velikost činí 32 bitů. Ukládány jsou hodnoty od $-3,402835 \times 10^{38}$ do $3,4028235 \times 10^{38}$.

Dalšími jsou logické datové typy. Zde uvedu pouze jeden a tím je datový typ s názvem boolean, kde proměnné s tímto datovým typem uchovávají pouze dvě hodnoty. Tím jsou hodnoty true a false neboli pravda a nepravda.

Poslední typ, který zmíním, bude znakový datový typ s názvem char. Tento datový typ nám uschovává hodnoty, kterými nejsou čísla ale znaky. K tomu se nejčastěji používá kód ASCII. [4]

3.3.5. Podmínka if()

Podmínky jsou velmi důležité, pokud potřebujeme, aby se nějaká akce (část kódu), provedla v určitém případě. Příkladem může být, například rozsvícení žárovky v určitém čase nebo zalévání hlíny, když nám procentuální hodnota klesne pod danou úroveň. Tyto akce, nám může zajistit podmínka if, která říká, že pokud něco nastane, vykoná se část kódu. Při použití else se vykoná jiná akce, jestli neplatí podmínka if. Je také možné, použít else if (nebo pokud), tedy pokud bude platit jiná podmínka. Na Obr. 13 vidíme příklad, použití podmínky if. Je zde zapnutí LED diod, jestli hodnota bude menší jak 20. [4]



```
sketch_mar14a | Arduino 1.8.5
Soubor Úpravy Projekt Nástroje nápověda
sketch_mar14a §
1 int LED1 = 5;
2 int LED2 = 6;
3 int x = 18;
4 void setup() {
5   // put your setup code here, to run once:
6   pinMode(LED1, OUTPUT);
7   pinMode(LED2, OUTPUT);
8 }
9
10 void loop() {
11   // put your main code here, to run repeatedly:
12   if(x < 20) {digitalWrite(LED1,HIGH);
13   digitalWrite(LED2,HIGH);
14   delay(3000);
15   digitalWrite(LED1,LOW);
16   digitalWrite(LED2,LOW);
17   delay(3000);}

```

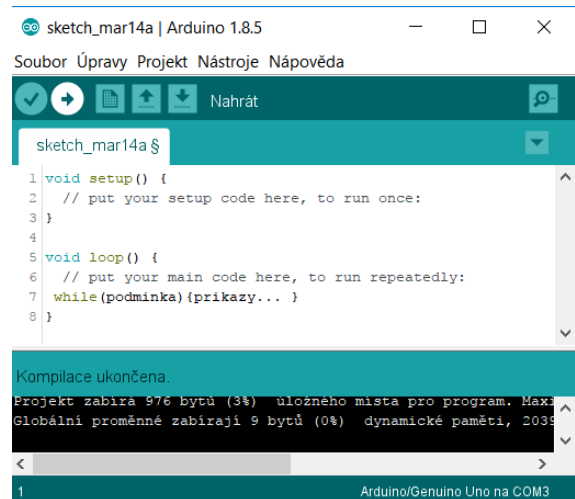
Kompilace ukončena.
Projekt zabírá 976 bytů (3%) uložného místa pro program. Ma
Globální proměnné zabírají 9 bytů (0%) dynamické paměti, 20

3 Arduino/Genuino Uno na COM3

Obr. 13 Příklad podmínky if

3.3.6. Cyklus while()

Cyklus while se provádí, dokud je podmínka pravdivá. Pokud podmínka pravdivá není cyklus while, se neprovede a skončí. [4]



```
sketch_mar14a | Arduino 1.8.5
Soubor Úpravy Projekt Nástroje Nápověda
sketch_mar14a $
1 void setup() {
2 // put your setup code here, to run once:
3 }
4
5 void loop() {
6 // put your main code here, to run repeatedly:
7 while(podminka){prikazy... }
8 }
Kompilace ukončena.
Projekt zabírá 976 bytů (3%) uložného místa pro program. Max:
Globální proměnné zabírají 9 bytů (0%) dynamické paměti, 2039
1 Arduino/Genuino Uno na COM3
```

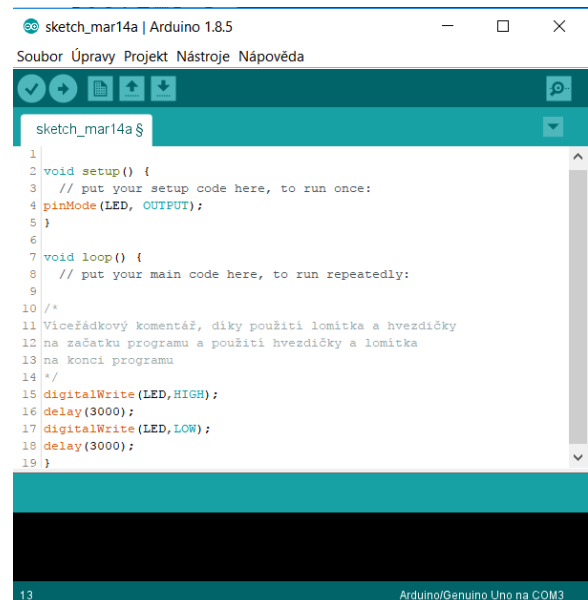
Obr. 14 ukázka cyklu while

3.3.7. Cyklus for()

Cyklus for využíváme, když známe počet opakování cyklu. Cyklus for, musí mít kvůli hlídání počtu opakování, vlastní proměnnou. [4]

3.3.8. Delay()

Funkci delay využíváme, když potřebujeme pozastavit program, na určitou dobu. Delay zastaví veškerou činnost, jako čtení ze senzoru atd. Delay obsahuje jeden parametr, který je udáván v milisekundách. Na Obr. 15 vidíme využití funkce delay, díky který nám led dioda bliká. [4]



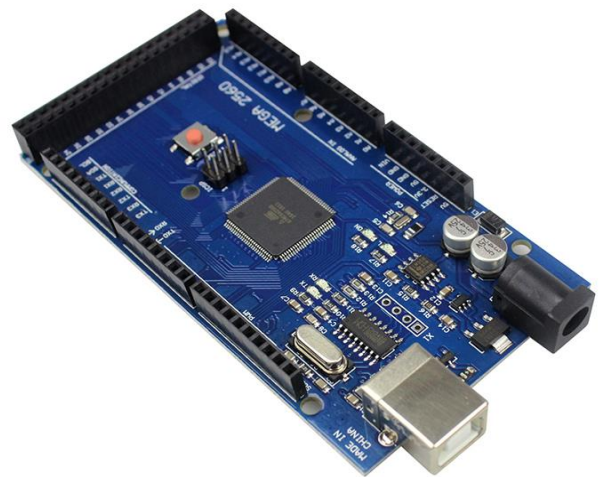
```
sketch_mar14a | Arduino 1.8.5
Soubor Úpravy Projekt Nástroje Nápověda
sketch_mar14a $
1
2 void setup() {
3 // put your setup code here, to run once:
4 pinMode(LED, OUTPUT);
5 }
6
7 void loop() {
8 // put your main code here, to run repeatedly:
9
10 /*
11 Víceřádkový komentář, díky použití lomítka a hvězdičky
12 na začátku programu a použití hvězdičky a lomítka
13 na konci programu
14 */
15 digitalWrite(LED,HIGH);
16 delay(3000);
17 digitalWrite(LED,LOW);
18 delay(3000);
19 }
13 Arduino/Genuino Uno na COM3
```

Obr. 15 Program s použitím funkce delay

4. Použité komponenty

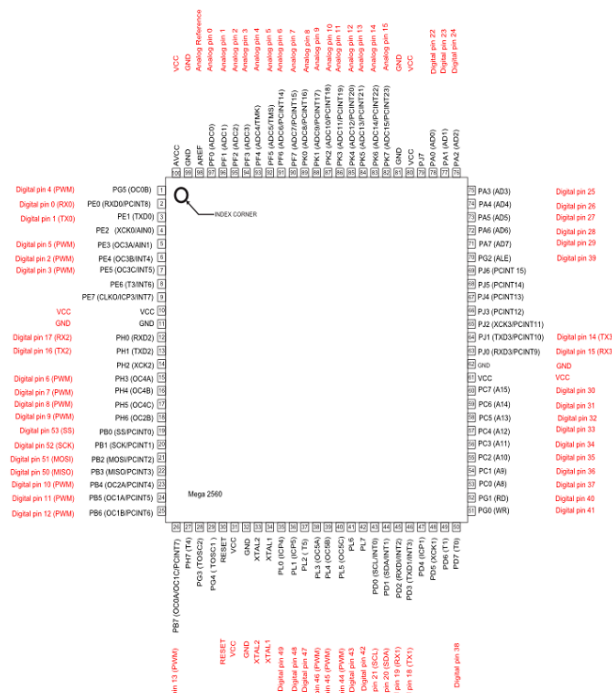
4.1. Řídící jednotka

Použitá řídicí jednotka je jednodeskový počítač Arduino Mega2560 R3, který využívá mikroprocesor ATmega2560. Zařízení obsahuje 54 digitálních vývodů, 4 UART (hardwarové sériové porty) a 16 analogových vývodů. U digitálních vývodů je možné, 15 z těchto vývodu použít jako výstupy PWM. Na desce nalezneme také resetovací tlačítko. Napájecí napětí desky je doporučeno, v rozmezí 7–12 V. Arduino Mega2560 používá ke komunikaci s počítačem USB rozhraní. Maximální proudové omezení na pin, je uváděno 40 mA. Jenže doporučená hodnota, se pohybuje okolo 20 mA, aby se předešlo, k poničení desky. Je uvedeno, že mikroprocesor ATmega2560 má 256 kB flash paměti pro uložení programu a z toho 8 kB si vezme bootleader mikroprocesoru. Dále SRAM je 8 kB paměti a EEPROM 4 kB paměti. Délka celé desky je 101,52 mm, šířka 53,3 mm a celková hmotnost je 37 g.



Obr. 16 Arduino Mega2560 R3 [4]

[5]



Obr. 17 Spojení pinů na Atmega2560 [5]

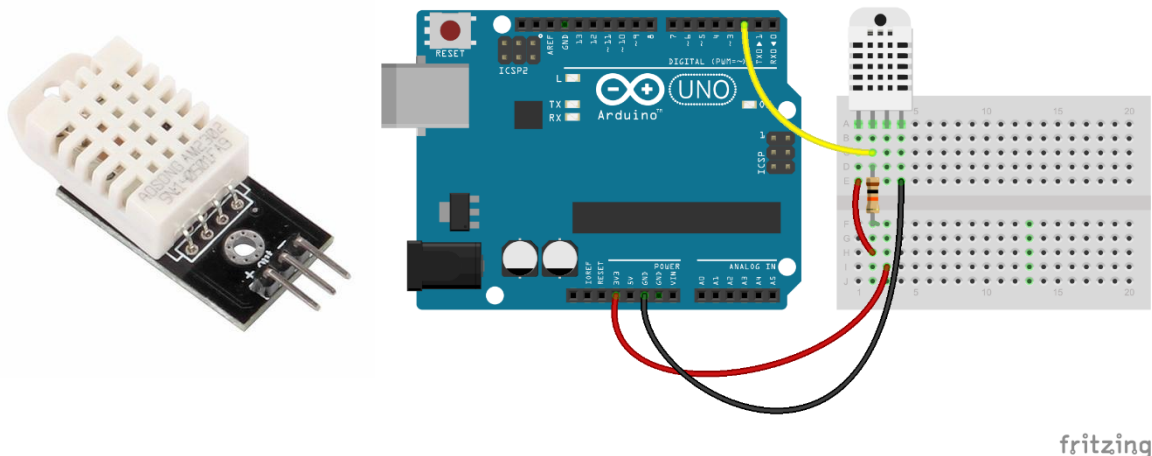
4.2. Vstupní periferie

4.2.1. DHT 22 – Senzor teploty a vlhkosti okolí

Senzor DHT22 se používají jak pro měření vlhkosti, tak i pro měření teploty vzduchu. To zajišťuje kapacitní vlhkosní čidlo. Napájecí napětí udáváno výrobcem je 3–5 V. Senzor má celkem 4 piny: +Vcc, Data dále nezapojený pin a GND.

DHT22 snímá v rozsahu od 0 do 100 % relativní vlhkosti a -40 až +80 °C provozní teploty Proto je možné použít DHT22 i pod 0 °C. Výhodou tohoto senzoru je vysoká přesnost, která je pro vlhkost $\pm 2 \%$ a u teploty $\pm 0,5 \text{ }^\circ\text{C}$. Jeho maximální doba vzorkování je 2 sekundy.

Na Obr. 8 je vidět pull-up rezistor 10 k Ω , který nám zajišťuje klidový stav pinu. Tento rezistor se zapojuje mezi pin pro napájení a data. Vzhledem k tomu, že samotné čidlo by mělo obsahovat integrované pull-up rezistory, by tento rezistor teoreticky nemusel být zapojen. Integrované rezistory se dají účinně použít, jen tehdy pokud je snímač zapojen bez vlivu přívodních kabelů (20 cm a více). [6]

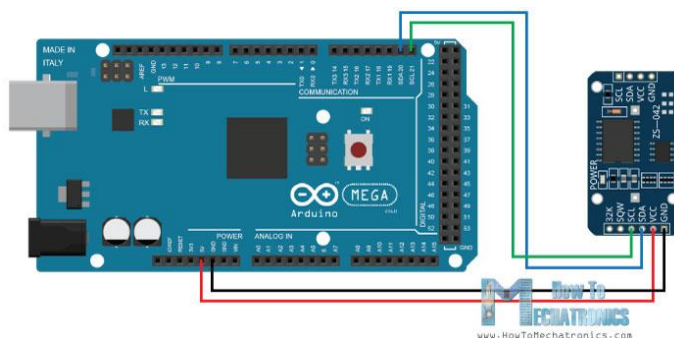


Obr. 18 DHT 22 senzor [6]

Obr. 19 Zapojení senzoru do Arduina Uno

4.2.2. RTC DS3231

Modul RTC, neboli real-time-clock slouží k indikaci reálného času. To nám zajišťuje vnitřní mikročip AT24C32, který má kapacita úložiště 32 KB. Modul využívá sběrnici I2C, což je speciální způsob komunikace Arduina. Piny k tomu určené, jsou označeny SDA a SCL. Modul poskytuje sekundy, minuty, hodiny, den, datum, měsíc a rok. I přesto, že modul bude odpojen od externího zdroje napájení, bude modul udržovat v provozu 3 V baterie. Jeho výhodou je zabudovaný vnitřní oscilátor, který neovlivňuje vnější faktory, a proto jeho odchylka je nanejvýš několik minut ročně. [7]



Obr. 20 Modul RTC DS3231 [7] Obr. 21 Zapojení modulu do Arduino Mega 2560 [7]

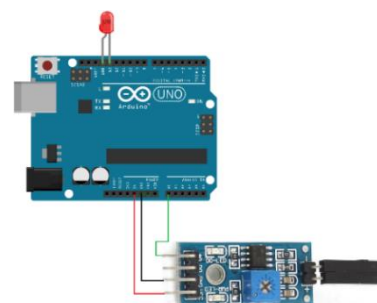
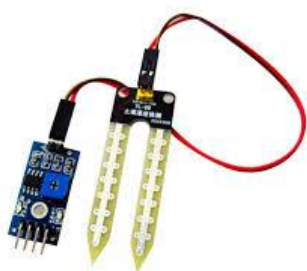
4.2.3. Půdní vlhkoměr

Tento modul slouží k měření vlhkosti půdy. Obsahuje dvě elektrody, který měří odpor mezi nimi a dle jeho velikosti přepočítává na množství vlhkosti v půdě.

Nevýhodou tohoto senzoru je dlouhodobé používání. Po čase tento senzor může být nespolehlivý. Může dojít k reznutí elektrod, což povede k nefunkčnosti senzoru nebo k jeho nepřesnosti při měření vlhkosti půdy.

Obsahuje potenciometr, kterým se nastavuje prahová hodnota. Prahová hodnota je dále porovnána s komparátorem LM393.

Modul má dva výstupy. Tím je analogový a digitální výstup. Digitální výstup můžeme použít jako spínač, kde úroveň sepnutí nastavuje trimrem na DPS zesilovače. Analogový využíváme pro přesné měření hodnoty. Napájecí napětí udávané výrobcem je 3,3 – 5 V. [8]



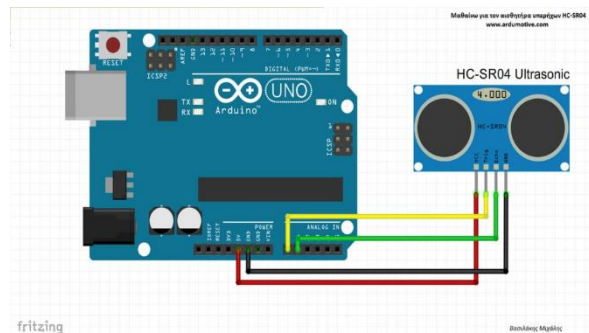
Obr. 22 Půdní vlhkoměr [8]

Obr. 23 Zapojení modulu do Arduino UNO [9]

4.2.4. Ultrazvukový měřič vzdálenosti – Měření hladiny vody

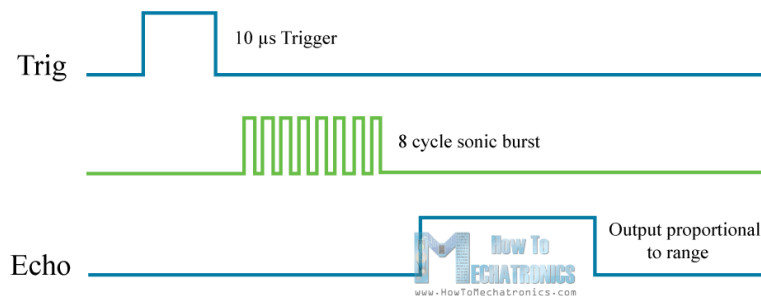
Ultrazvukový snímač HC-SR04 je využíván, jako snímač vzdálenosti. Měří od 2 cm do 400 cm. Výhoda tohoto modulu je, že není ovlivňována slunečním zářením nebo černým materiálem. Ultrazvukový modul HC-SR04 má 4 kolíky, kterými jsou uzemnění, VCC, Trig a Echo.

Abychom mohli generovat ultrazvuk, musíte nejdříve nastavit Trig na vysokém stavu po dobu 10 μ s. Díky tomu se vyšle 8 cyklický zvukový impulz, který se pohybuje rychlostí zvuku. Dále bude přijat v kolíku Echo. Z konektoru Echo bude výstupní čas v mikrosekundách. [9]

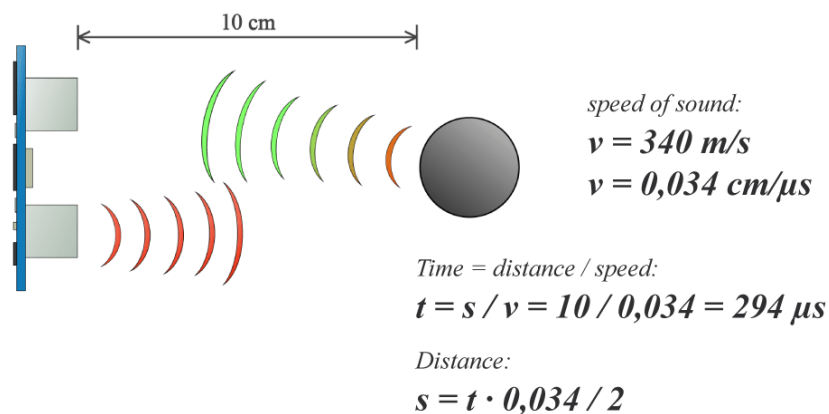


Obr. 24 Ultrazvukový měřič vzdálenosti [10]

Obr. 25 Zapojení modulu do Arduino UNO [10]



Obr. 26 Ukázka generování ultrazvuku [10]



Obr. 27 Ukázka měření vzdálenosti objektu a vypočítání hodnot [10]

4.3. Výstupní periferie

4.3.1. Čerpadlo

Jedná se o odstředivé čerpadlo, které využívá odstředivé síly působící na kapalinu vznikajícím při jejím rotujícím pohybu. Toto tiché a ponorné čerpadlo má zdvih 40 až 110 centimetrů s průtokem vody 80 až 120 litrů za hodinu. Jeho napětí, udávané výrobcem, je 2,5 – 6 V. Životnost čerpadla v neustálém provozu je udávána, až 500 hodin. Je velmi malé, jeho délka je pouhá 45 mm a výška 33 mm. [10] [13]



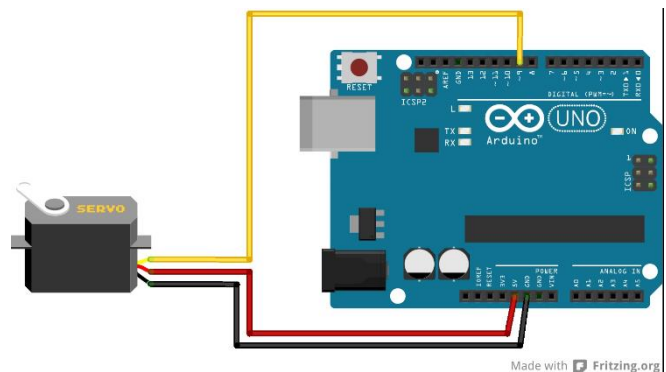
Obr. 28 Čerpadlo [11]

4.3.2. Servomotor – Pootevívání okna

Servomotor TOWERPROSG-90 využíváme, k přesnému ovládní úhlové polohy, rychlosti a také zrychlení. Tento servomotor má tři vývody. Jeden z nich, je pro napájení další pro uzemnění a poslední pro příjem ovládacích impulsů. Jeho hmotnost je poměrně malá váží pouhých 14 g. Jeho pracovní teplotní se pohybuje od -30 až +60 ° C. [11]



Obr. 29 Servomotor [12]

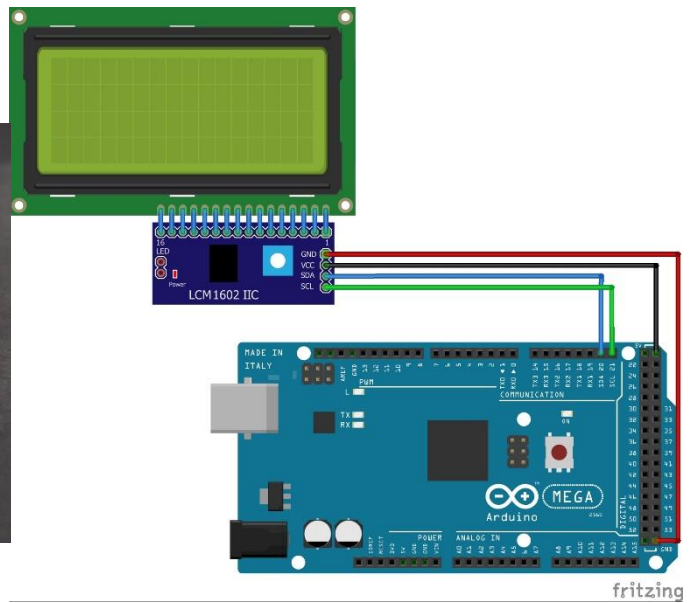


Obr. 30 Zapojení servomotoru do Arduino UNO

4.3.3. LCD displej

LCD displej se žlutým podsvícením má 20 znaků, neboli sloupců a 4 řádky. Jeho napájecí napětí je 5 V. Díky I2C sběrnici bude možné displej připojit pouze 4 vodiči, kterými jsou Vcc, SDA, SCL a GND.

Ladění kontrastu můžeme provést potenciometrem, který najdeme na I2C převodníku. Také zde najdeme jumper, který slouží k ladění podsvícení. [12]

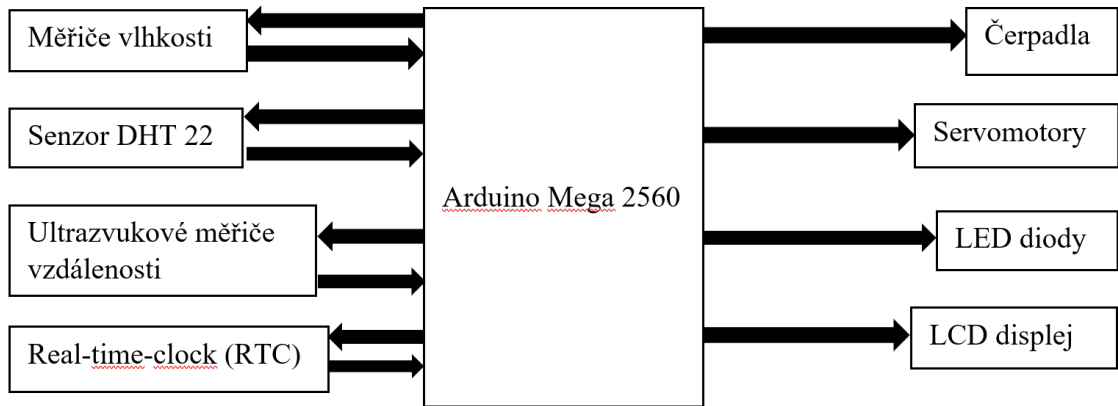


Obr. 31 Displej žlutý 20x4 [13]

Obr. 32 Zapojení přes I2C sběrnici do Arduino Mega

5. Praktická část

5.1. Blokové schéma



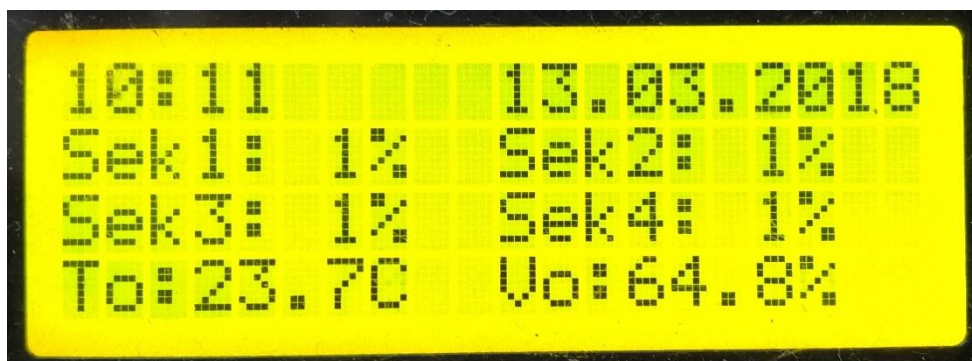
Obr. 35 Blokové schéma

5.2. Popis displeje

Na obrazu LCD displeje o velikosti 20 sloupců a 4 řádků se nám v horní části zobrazuje aktuální čas a také aktuální datum, pomocí RTC modulu (reálného modulu času).

Na dalších dvou řádcích se vypisuje jak vlevo, tak i vpravo aktuální procentuální hodnoty vlhkosti půdy ve všech čtyř sektorech. To zajišťují čtyři senzory vlhkosti půdy.

Na posledním řádku se vyobrazuje vlevo hodnota teploty okolí ve stupních Celsia a vpravo procentuální hodnota vlhkosti okolí, díky senzoru DHT22.



Obr. 36 Ukázka výpisů z čidel na LCD displej

5.3. Mechanická konstrukce

Řídící jednotka, displej a některé další periferie jsou uloženy do šedé krabičky z tvrzeného polyesteru. Její rozměry jsou 190×140×70 mm a má krytí IP65.



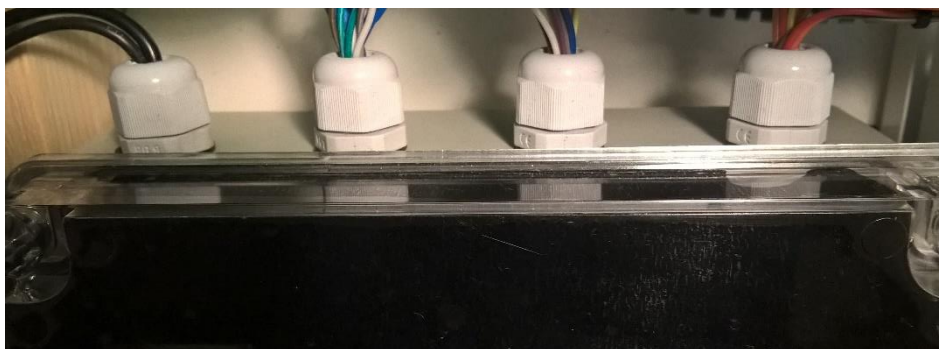
Obr. 37 Krabice

Jelikož je víčko průhledné, bylo zespodu nalepena černá fólie, do které byly posléze vyřezány dvě okénka. Jedno pro LCD displej a druhý pro LED diody.



Obr. 38 Víko krabice

Do této krabice byly vyvrtány 4 díry, do které se namontovaly 4 vodotěsné průchodky.



Obr. 39 Ukázka průchodek přidělaných v na boku krabice

Do dvou plastových nádržek na vodu se do první vyvrtalo 5 děr a do druhé 2 díry. I do těchto děr byly namontovány vodotěsné průchodky.



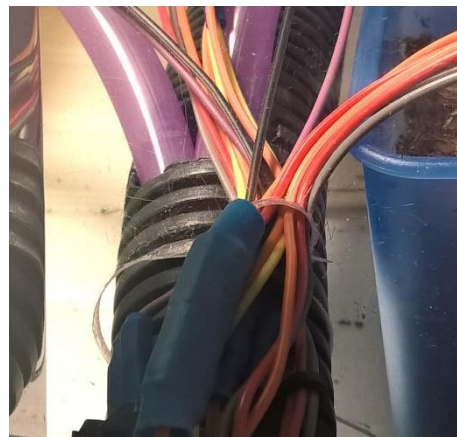
Obr. 40 Nádržky na vodu

Koupený skleník z velkoobchodu Ikea se skládá z polyesterového materiálu. Jeho velikost je 490×360×80 mm.



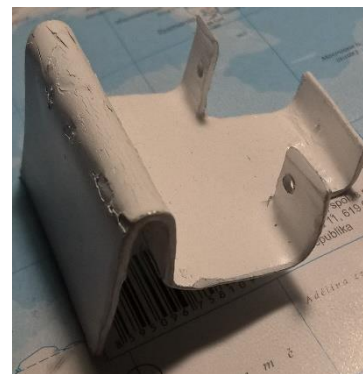
Obr. 41 Skleník

Do tohoto skleníku byla vyvrtána jedna větší a jedna malá díra. Do větší byl protáhnut „husí krk“, v kterém byly všechny 4 hadice. Do druhé menší díry se protáhly kabely k senzorům a servomotorům.



Obr. 42 Průchod kabelů a hadic skrz skleník

K uchycení servomotorů, byly vyrobeny dvě kovové úchytky, které jsou vyfoceny na Obr. 43. Všechny tyto komponenty byly dány a přidělány na dřevěnou desku o rozměrech 490×510. Výsledný projekt vidíme na Obr. 44.



Obr. 43 Úchytky pro servomotory



Obr. 44 Výsledný projekt

5.4. Zapojení pinů

5.4.1. Vstupní periferie

DHT22

Vcc -> 5 V

GND -> GND

DATA -> pin 36

NZ -> nezapojený pin

RTC DS3231

Vcc -> 5 V

GND -> GND

SDA -> SDA

SCL -> SCL

Půdní vlhkoměry

Vcc -> 5 V

GND -> GND

D -> nezapojený pin

A -> piny A0, A1, A2, A3

Ultrazvukové měřiče vzdálenosti

Vcc -> 5 V

GND -> GND

TRIG -> piny 40, 38

ECHO -> piny 6, 5

5.4.2. Výstupní periferie

Čerpadla

Vcc -> 5 V

GND -> piny 11, 10, 9, 8, 7

Servomotory

Vcc -> 5 V

GND -> GND

D -> piny 3, 4

Diody

Vcc -> piny 52, 50, 48, 46, 44, 42

GND -> GND

LCD displej

Vcc -> 5 V

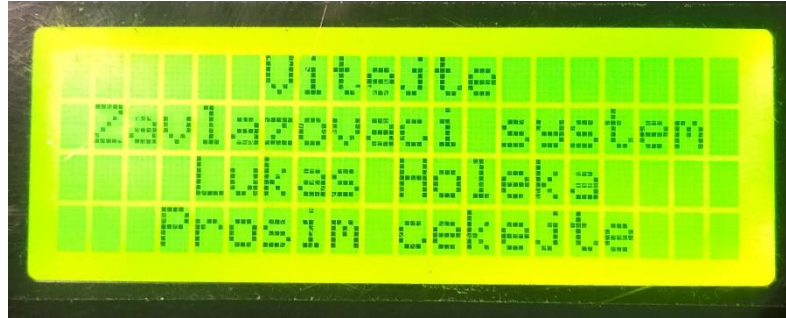
GND -> GND

SDA -> SDA

SCL -> SCL

5.5. Návod k provozu a jeho průběh

Nejdříve musíme připojit adaptér do sítě 230 V. Na prvním řádku LCD displeje, se zobrazí vítejte, pak název mého projektu, dále mé jméno a na posledním řádku je vypsáno Prosím čekejte.



Obr. 45 Počáteční vypsání displeje po zapnutí projektu

Po chvíli čekání, nám LCD displej vypíše hodnoty ze senzorů, kde aktuální čas s datumem je na prvním řádku na dalších dvou řádcích se vypisují hodnoty vlhkosti půdy ve všech čtyř sektorech, díky senzoru měřiče vlhkosti půdy. Poslední řádek nám vypisuje aktuální teplotu a vlhkost okolí, což zajišťuje senzor DHT22.



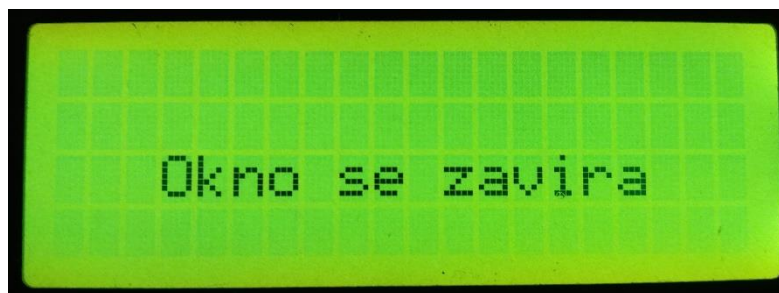
Obr. 46 Vypisování všech hodnot na displej

Pokud bude aktuální teplota ve skleníku rovna nebo nad 29 °C, spustí se servomotory a pootevřou okno. Zároveň se na LCD displeji vypíše Okno se otevírá viz Obr. 47.



Obr. 47 Vypsání informace na displej při otevírání okna

Jestli bude teplota ve skleníku nižší jak 29 °C, znovu se spustí servomotory, ale tentokrát okno zavřou. Na LCD displeji se tedy vypíše Okno se zavírá viz Obr. 48.



Obr. 48 Vypsání informace na displej při zavírání okna

Pokud nastane, že v jednom ze čtyřech sektorů, klesne požadovaná hodnota, spustí se jedno z čerpadel, které zaleje daný sektor. Díky LCD displeji zjistíme, jaké čerpadlo je spuštěné.



Obr. 49 Vypsání informace na displej při zapnutí daného čerpadla

Ultrazvukové měřiče vzdálenosti měří hladinu vody v nádržích. Tu nám indikují tři různobarevné diody ke každé nádrži. Pokud nádrže jsou plné, svítí zelená Led dioda, pokud hladina klesne na polovinu, rozsvítí se oranžová Led dioda a jestli nádrž bude prázdná, svítí červená Led dioda.



Obr. 50 Rozsvícení zelené LED diody



Obr. 51 Rozsvícení žluté LED diody



Obr. 52 Rozsvícení červené LED diody

Když dojde voda v hlavní nádrži, rozsvítí se červená LED dioda a zároveň se zapne páté čerpadlo, které také vypíše displej.



Obr. 53 Vypsání informace na displej při zapnutí čerpadla ze záložní nádrže

5.6. Elektronická rozpiska součástek

Pol.	Množství	Název	Typ	Hodnota
1	5	Rezistor	metalizovaný	1 k Ω / 0,6W
2	6	Rezistor	metalizovaný	330 Ω / 0,6 W
3	5	Tranzistor	bipolární	2N5088
4	1	Dioda	usměrňovací	1N4004

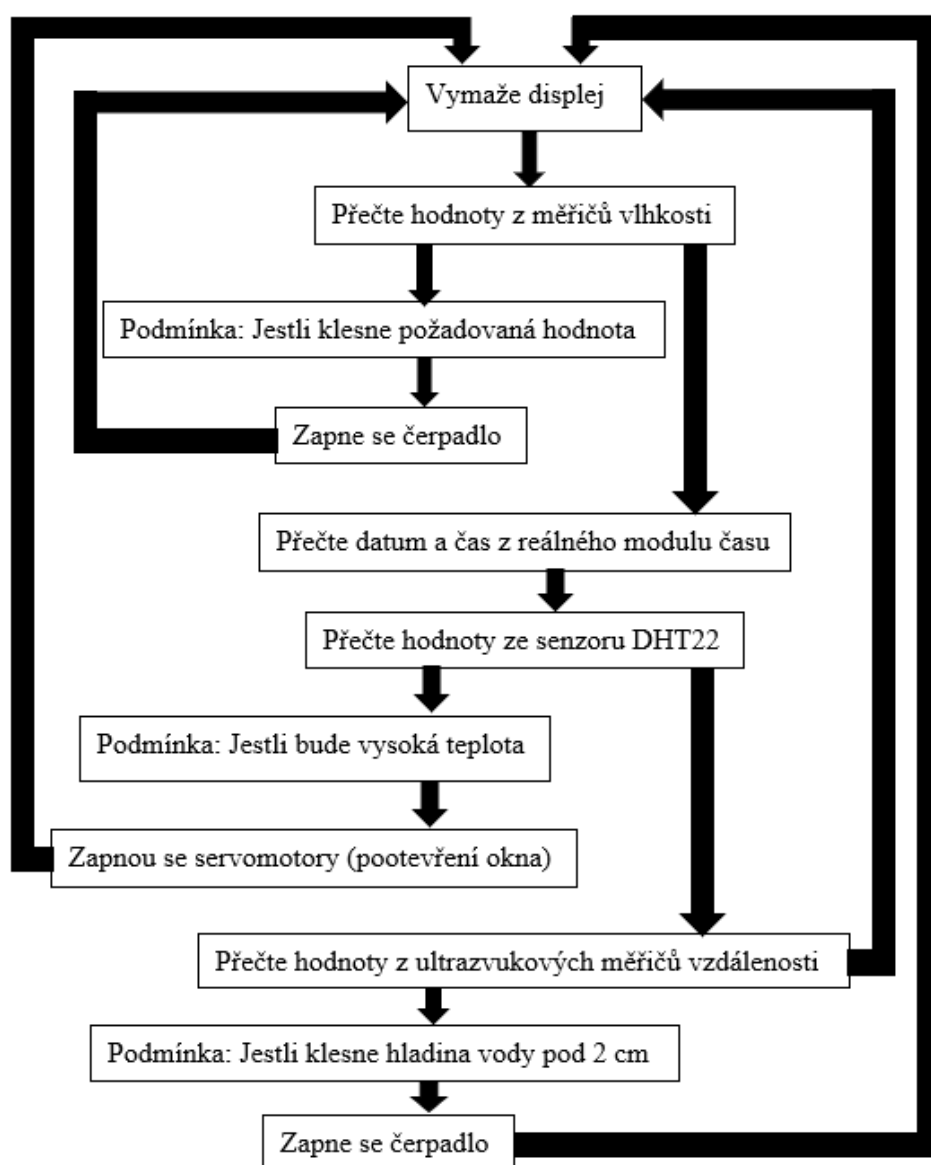
5.7. Mechanická rozpiska součástek

Pol.	Množství	Název	Typ	Hodnota
1	9	Šroub	M3x10	-
2	11	Šroub	M4x10	-
3	4	Distanční sloupek	M3x10	-
4	6	Kabelová vývodka	PG9	-
5	6	Kabelová vývodka	KSS EG-16	-
6	1	Krabička dvoudílná	S-BOX 416-P	-

6. Software

Software, který je nahrán do řídicí jednotky, byl vytvořen ve vývojovém prostředí Arduino IDE. Toto vývojové prostředí, jsem popisoval v kapitole 3.1. Programovací prostředí a jazyk. Zde popíšu několik naprogramovaných funkcí v mém programu.

6.1. Struktura programu



Obr. 54 Blokové schéma programu

6.2. Knihovny

Nejdříve v mém programu, bylo potřeba definovat knihovny, jak se definují knihovny můžeme najít v kapitole 3.3.1. Knihovny. V tomto programu, jsme si museli nadefinovat knihovnu pro servomotory, pak pro LCD displej, senzor DHT22 a knihovnu pro modul RTC.

```
1 //Zahrnutí všech knihoven:
2 #include <Servo.h>           //knihovna pro ovládání servo motoru
3 #include <LiquidCrystal_I2C.h> //knihovna pro Displej
4 #include <DS3231.h>         //knihovna pro RTC
5 #include <DHT.h>           //knihovna pro DHT
6 #include <Wire.h>
```

Obr. 55 Definované knihovny

6.3. LCD displej

LCD displej je naprogramován, aby po zapnutí Zavlažovacího systému skleníku nás několika řádky přivítal a dále vypsál hodnoty z modulů. Vypisuje aktuální čas a datum, vlhkost půdy ve všech čtyř sektorech, vlhkost a teplotu okolí. Také nás informuje o dalších věcech jako otevírání, zavírání okna a také které čerpadlo je zapnuté. To vše jsme mohli vidět na obrázkách v kapitole 5.5. Návod k provozu a jeho průběh.

6.4. Zalévání

Pokud jde o zalévání, tak pomocí senzorů vlhkosti půdy, měříme vlhkost půdy v každém sektoru. Pokud nám klesne požadovaná hodnota, zapne se čerpadlo. Program této funkce vidíme na Obr. 55. Je nadefinována `vystupni_hodnotu1` a také `meric_vlhkosti1`. Pomocí `lcd.setCursor`, nastavíme požadované vypisování hodnot na určitém místě na displeji. A pomocí `lcd.print` můžeme vypsát na displej co potřebujeme.

```

//Meric vlhkosti 1:
//mapa a nastavení
vystupni_hodnotal = map(analogRead(meric_vlhkostil), 1030, 100, 0, 100);
lcd.setCursor(0, 1);
lcd.print("Sek1: ");
lcd.print(vystupni_hodnotal);
lcd.print("%");
if (vystupni_hodnotal < 10) {
  lcd.print(" ");
}
if (vystupni_hodnotal < 10) {
  lcd.clear();
  lcd.setCursor(3, 1);
  lcd.print(sys_zpravy[6] + "1");
  lcd.setCursor(3, 2);
  lcd.print(sys_zpravy[7]);
  digitalWrite(cerpl, HIGH);
  delay(cas);
  digitalWrite(cerpl, LOW);
  lcd.clear();
}

```

Obr. 56 Ukázka programu pro zalévání

6.5. Pootevírání okna

Pokud potřebujeme pootevřít nebo zavřít okno podle teploty, je třeba, naprogramovat senzor DHT22 viz Obr. 57. Pokud chceme, například pootevřít okno při vysoké teplotě, musí se pomocí podmínky if napsat podmínka. Jestli teplota bude rovna nebo větší jak 29 °C okno se pootevře. K pootevírání okna využijeme funkci for viz Obr. 58. Zde nastavíme počáteční pozici, což je 90° a odečítáme stále 1, dokud se pozice nebude rovnat 20°. U zavírání okna je princip stejný, jen zde je nastavená výchozí hodnota 20° a stále se přičítá 1, dokud servomotory nebudou mít úhel 90° viz Obr. 59.

```

//DHT22:
float tep = mojeDHT.readTemperature();
float vlh = mojeDHT.readHumidity();
if (isnan(tep) || isnan(vlh)) {
  lcd.println("Chyba DHT senzoru!"); //Pokud hodnoty nejsou v pořádku vypíše se chyba
} else {
  // pokud jsou hodnoty v pořádku
  lcd.setCursor(0, 4);
  lcd.print("To:");
  lcd.print(tep);
  lcd.setCursor(7, 4);
  lcd.print((char)223);
  lcd.print("C");
  lcd.setCursor(10, 4);
  lcd.print("Vl:");
  lcd.print(vlh);
  lcd.setCursor(17, 4);
  lcd.print("%");
}

```

Obr. 57 Program pro DHT22

```

for (pos = 90; pos >= 20; pos -= 1) //je od úhlu 90 do úhlu 20
{
  servol.write(pos); //natočení motoru 1 na aktuální úhel
  servo2.write(pos); //natočení motoru 2 na aktuální úhel
  delay(50);
}

```

Obr. 58 Program pro Servomotory pro otevírání okna

```

for (pos = 20; pos >= 90; pos += 1) //je od úhlu 90 do úhlu 20
{
  servol.write(pos); //natočení motoru 1 na aktuální úhel
  servo2.write(pos); //natočení motoru 2 na aktuální úhel
  delay(50);
}

```

Obr. 59 Program pro Servomotory pro zavírání okna

6.6. Hladina vody v nádržích

Jelikož potřebujeme hlídat hladinu v obou nádržích, použijeme ultrazvukový měřič vzdálenosti. Nejdříve nadefinujeme pTrig1, pak pEcho1, vzdalenost1 a odezva1. Díky funkci pulseIn, získáme délku pulzu v mikrosekundách. Poté musíme vypočítat vzdálenost, což je získaný čas, tedy odezva1 / 58,31. Pomocí podmínky if určíme, která LED dioda se má rozsvítit, nastavíme tedy, například podmínku, jestli vzdalenost1 bude menší nebo rovna 6 bude svítit jen zelená LED dioda. Když bude větší jak 6 rozsvítí se oranžová dioda a jestli bude větší jak 10 bude svítit červená.

```
361 //Měřič vzdálenosti v první nádrži:
362 // nastavíme na 2 mikrosekundy výstup na GND (pro jistotu)
363 // poté nastavíme na 5 mikrosekund výstup rovný napájení
364 // a poté opět na GND
365 digitalWrite(pTrig1, LOW);
366 delayMicroseconds(2);
367 digitalWrite(pTrig1, HIGH);
368 delayMicroseconds(5);
369 digitalWrite(pTrig1, LOW);
370 // pomocí funkce pulseIn získáme následně
371 // délku pulzu v mikrosekundách (us)
372 odezva1 = pulseIn(pEcho1, HIGH);
373 // přepočítání získaného času na vzdálenost v cm
374 vzdalenost1 = odezva1 / 58.31;
375 Serial.print("Vzdálenost je ");
376 Serial.print(vzdalenost1);
377 Serial.println(" cm.");
378 if (vzdalenost1 <= 6 ) {
379     digitalWrite(zell, HIGH);
380     digitalWrite(zlul, LOW);
381     digitalWrite(cerl, LOW);
382 }
383 if (vzdalenost1 > 6 ) {
384     digitalWrite(zell, LOW);
385     digitalWrite(zlul, HIGH);
386     digitalWrite(cerl, LOW);
387 }
388 if (vzdalenost1 > 10) {
389     digitalWrite(zell, LOW);
390     digitalWrite(zlul, LOW);
391     digitalWrite(cerl, HIGH);
392     lcd.clear();
393     lcd.setCursor(3, 1);
394     lcd.print(sys_zpravy[6] + "5");
395     lcd.setCursor(3, 2);
396     lcd.print(sys_zpravy[7]);
397     digitalWrite(cerp5, HIGH);
398     delay(5000);
399     digitalWrite(cerp5, LOW);
400     lcd.clear();
401 }
```

Obr. 60 Program pro rozsvícení různobarevných diod podle hladiny vody

7. Oživení výrobku

Při konstrukci Zavlažovacího systému skleníku bylo nutné nejdříve vymyslet rozmístění skleníku s nádržemi a také krabici, kam měla být uložena řídicí jednotka. Všechny tyto komponenty, byly rozmístěny na dřevěnou desku o velikosti 490×510. Do hlavní nádrže, bylo vyvrtáno celkem 6 děr a do nich byly upevněny vodotěsné průchodky. U záložní nádrže byly vyvrtány pouze 2 díry. Jedna díra u každé nádrže, sloužila k protáhnutí kabelů a ostatními se protáhly hadice. Do krabice byli vyvrtány 4 díry a jako u nádrží, i zde byly namontovány vodotěsné průchodky, kterými vedli vodiče a 2 napájecí konektory pro Arduino a čerpadla. Do skleníku, byly z přední části vyvrtány 2 díry, kde jedna sloužila k protáhnutí vodičů a druhá pro všechny 4 hadice, vycházející z hlavní nádrže. Dalším krokem, bylo samotné rozmístění senzorů a ostatních komponentů. Čerpadla uvnitř nádrží a servomotory na přídatných kovových úchytech, byly uchyceny oboustrannou lepicí páskou. Senzor DHT22, byl umístěn a uchycen šroubem uvnitř skleníku. Dalším krokem bylo rozmístění a vyvrtání montážních otvorů displeje, reálného modulu času, řídicí jednotky a potenciometru od vlhkosti půdy dovnitř krabice. Pro RTC, byla vytvořena speciální podložka, která byla přivrtána dovnitř krabice a poté se přivrtalo RTC, pomocí šroubů k podložce. I pro LCD displej byl vytvořen speciální stojan z plastu, na kterém byl LCD displej uchycen též šrouby. Po připevnění všech komponentů šrouby dovnitř krabice, bylo potřeba navrhnout vlastní shield pro Arduino mega 2560 R3 v aplikaci Eagle. Po vytvoření DPS a osazení všemi součástky, byla DPS zasazena pomocí nožů do Arduina. Dále bylo potřeba zapojit napájecí a datové vodiče. Poslední úkol bylo vytvoření malé krabičky, která má sloužit jako úschovna pro napájecí adaptéry při převozu. Ta byla pomocí oboustranné pásky uchycena k dřevěné desce

8. Závěr

Cílem této maturitní práce bylo navrhnout a sestrojít Zavlažovací systém skleníku, jehož řídicí jednotka bude jednodeskový mikročip Arduino.

Když byly koupeny všechny součástky, potřebné k tomuto projektu, došlo nejdříve k odzkoušení každého senzoru a modulu zvlášť. Po otestování a seznámení s funkcemi jednotlivých senzorů došlo k napsání programu pro Arduino, návrhu a konstrukci výrobku.

U psaní programu, jsem se setkal s některými problémy. Například s vypisování hodnot na displej nebo otevření okna v závislosti na teplotě.

Ke konstrukci bylo třeba koupit modul skleníku a zvolit si nádrže pro vodu. Po koupení skleníku a zvolení správných nádrží, bylo třeba koupit krabici, kam se měla umístit řídicí jednotka a další součástky, moduly. Zvolil se správný rozměr krabice. Sestrojila se podložka pro komponenty a přešlo se k úpravám komponentů. Po vyvrtání potřebných děr a upevnění vodotěsných průchodek, byly komponenty připevněny k vyrobené podložce. Do krabice pro řídicí jednotku a ostatní komponenty se vyvrtaly montážní otvory. Všechny části se upevnily. Rozmístily se všechny senzory, moduly a čerpadla. Větším problémem bylo promyslet, co nejefektivněji rozmístit řídicí jednotku a ostatních komponenty uvnitř krabice. V posledním kroku práce bylo připevňování kabelů, aby výrobek vypadal co nejlépe.

Po všech kosmetických úpravách, bylo možné nahrát předem napsaný program, do Arduina a sledovat, jak funguje Zavlažovací systém skleníku. Jelikož každá rostlina potřebuje něco jiného, už je jen na mě, jaké hodnoty zadat, aby se o rostlinu pečovalo co, nejlépe bez přítomnosti člověka.

Po splnění všech těchto bodů, se podařilo sestrojít automatický Zavlažovací systém skleníku bez pomoci lidské ruky.

V budoucnu by se zavlažovací systém dal určitě vylepšit. Skleník by mohl být rozšířen o topení, které by celý skleník vytápělo. Také by se mohlo vyřešit doplňování vody do záložní nádrže, například připojení přídatné hadice z kohoutku ovládané pomocí ventilu.

9. Přílohy

9.1. Seznam obrázků

Obr. 1 Skleníkový efekt	12
Obr. 2 Odraz dopadajícího záření	12
Obr. 3 Arduino Uno	13
Obr. 4 vývojové prostředí Arduina IDE	14
Obr. 5 Vytvoření nového programu.....	15
Obr. 6 Ukázka hlavního kódu	15
Obr. 7 Ukázka chyby při kompilaci desky	15
Obr. 8 Ukázka přidávání knihoven do Arduina IDE	16
Obr. 9 Jednořádkový komentář.....	17
Obr. 10 Víceřádkový komentář	17
Obr. 11 Program bez použití proměnných.....	18
Obr. 12 Program s využitími proměnných.....	18
Obr. 13 Příklad podmínky if.....	19
Obr. 14 ukázka cyklu while	20
Obr. 15 Program s použitím funkce delay	20
Obr. 16 Arduino Mega2560 R3	21
Obr. 17 Spojení pinů na Atmega2560	21
Obr. 18 DHT 22 senzor.....	22
Obr. 19 Zapojení senzoru do Arduina Uno.....	22
Obr. 20 Modul RTC DS3231.....	23
Obr. 21 Zapojení modulu do Arduino Mega 2560	23
Obr. 22 Půdní vlhkoměr.....	23
Obr. 23 Zapojení modulu do Arduino UNO.....	23
Obr. 24 Ultrazvukový měřič vzdálenosti	24
Obr. 25 Zapojení modulu do Arduino UNO.....	24
Obr. 26 Ukázka generování ultrazvuku	24
Obr. 27 Ukázka měření vzdálenosti objektu a vypočítání hodnot.....	24
Obr. 28 Čerpadlo.....	25
Obr. 29 Servomotor	25
Obr. 30 Zapojení servomotoru do Arduino UNO.....	25

Obr. 31 Displaj žlutý 20x4.....	26
Obr. 32 Zapojení přes I2C sběrnici do Arduino Mega	26
Obr. 33 Schématická značka LED	26
Obr. 34 LED dioda.....	26
Obr. 35 Blokové schéma.....	27
Obr. 36 Ukázka výpisů z čidel na LCD displej	27
Obr. 37 Krabice.....	28
Obr. 38 Víko krabice	28
Obr. 39 Ukázka průchodek přidělaných v na boku krabice.....	28
Obr. 40 Nádržky na vodu.....	29
Obr. 41 Skleník	29
Obr. 42 Průchod kabelů a hadic skrz skleník.....	30
Obr. 43 Úchytky pro servomotory	30
Obr. 44 Výsledný projekt.....	30
Obr. 45 Počáteční vypsání displeje po zapnutí projektu.....	32
Obr. 46 Vypisování všech hodnot na displej	32
Obr. 47 Vypsání informace na displej při otevírání okna.....	32
Obr. 48 Vypsání informace na displej při zavírání okna	33
Obr. 49 Vypsání informace na displej při zapnutí daného čerpadla.....	33
Obr. 50 Rozsvícení zelené LED diody	33
Obr. 51 Rozsvícení žluté LED diody.....	33
Obr. 52 Rozsvícení červené LED diody	34
Obr. 53 Vypsání informace na displej při zapnutí čerpadla ze záložní nádrže.....	34
Obr. 54 Blokové schéma programu	36
Obr. 55 Definované knihovny	37
Obr. 56 Ukázka programu pro zalévání.....	38
Obr. 57 Program pro DHT22.....	39
Obr. 58 Program pro Servomotory pro otevírání okna.....	39
Obr. 59 Program pro Servomotory pro zavírání okna	39
Obr. 60 Program pro rozsvícení různobarevných diod podle hladiny vody.....	40

9.2. Seznam použité literatury

- [1] *Polykarbonát nebo sklo? - Dobrý skleník .cz.* [online]. [cit. 10.03.2018]. Dostupné z: <http://dobrysklenik.cz/cs/content/19-polykarbonat-nebo-sklo>
- [2] [online]. [cit. 10.03.2018]. Dostupné z: <https://www.cez.cz/edee/content/microsites/solarni/f3.htm>
- [3] *Arduino* – Wikipedie. [online]. [cit. 13.03.2018]. Dostupné z: <https://cs.wikipedia.org/wiki/Arduino>
- [4] *Home | Robotika Brno* [online]. [cit. 13.03.2018]. Dostupné z: <http://www.robotikabrno.cz/docs/arduino/Průvodcem-světlem-Arduina-CZ.pdf>
- [5] *Arduino Mega 2560 Rev3.* Arduino [online]. [cit. 14.03.2018]. Dostupné z: <https://store.arduino.cc/arduino-mega-2560-rev3>
- [6] *Teploměr a vlhkoměr DHT11 a DHT22 | Arduino návody.* Webový magazín o ARDUINU | Arduino návody [online]. [cit. 11.03.2018]. Dostupné z: <http://navody.arduino-shop.cz/navody-k-produktum/teplotni-senzor-dht11.html>
- [7] *Tutoriál – užívání hodin reálného času DS1307 a DS3231 s Arduinem | Arduino.cz.* Arduino.cz - Webový magazín o Arduinu a elektronice [online]. [cit. 11.03.2018]. Dostupné z: <https://arduino.cz/tutorial-uzivani-hodin-realneho-casu-ds1307-a-ds3231-s-arduinem/>
- [8] *Arduino and Soil Moisture Sensor -Interfacing Tutorial - Electronic Circuits and Diagrams-Electronic Projects and Design.* Electronic Circuits and Diagrams-Electronic Projects and Design - An Authentic Resource on Electronics [online]. [cit. 11.03.2018]. Copyright © 2015 [cit. 23.03.2018]. Dostupné z: <http://www.circuitstoday.com/arduino-soil-moisture-sensor>
- [9] *Ultrasonic Sensor HC-SR04 and Arduino Tutorial.* How To Mechatronics [online]. Copyright © 2018 HowToMechatronics.com. All rights reserved. [cit. 11.03.2018]. Dostupné z: <https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/>
- [10] *301 Moved Permanently.* 301 Moved Permanently [online]. [cit. 11.03.2018]. Dostupné z: <https://www.arduino-shop.cz/docs/produkty/0/75/1460321123.pdf>
- [11] *Servo motor | Arduino návody.* Webový magazín o ARDUINU | Arduino návody [online]. [cit. 12.03.2018]. Dostupné z: <http://navody.arduino-shop.cz/arduino-projekty/servo-motor.html>

[12] 301 Moved Permanently. 301 Moved Permanently [online]. [cit. 12.03.2018]. Dostupné z: <https://www.arduino-shop.cz/docs/produkty/0/105/1474620659.pdf>

[13] [online]. [cit. 02.04.2018]. Dostupné z: <https://automatizace.hw.cz/principy-pumyslovych-cerpadel-4dil-odstrediva-cerpadla>

9.2.1. Seznam použité literatury – obrázky

[1] [online]. [cit. 10.03.2018]. Dostupné z: <https://www.cez.cz/edee/content/microsites/solarni/f3.htm>

[2] Znižujeme tepelné straty v skleníku | Urob si sám. Urob si sám | Online magazín pre domácich majstrov [online]. [cit. 10.03.2018]. Dostupné z: <https://urobsisam.zoznam.sk/zahrada/zelenina-a-ovocie/znizujeme-tepelne-straty-v-skleniku>

[3] FAQ: The Arduino UNO Anatomy – Arduino Arts. Arduino Arts - Cool Arduino Projects, Tutorials and More! [online]. Copyright © 2018 Arduino Arts. All rights reserved. [cit. 13.03.2018]. Dostupné z: <http://arduinoarts.com/2011/08/the-arduino-uno-anatomy/>

[4] Arduino Mega 2560 Rev3 Mainboard with USB Cable - ATmega2560-16AU - Blue - JakartaNotebook.com. JakartaNotebook.com : Toko Komputer Belanja Online Harga Murah [online]. Copyright © 2018 JakartaNotebook.com [cit. 14.03.2018]. Dostupné z: <https://www.jakartanotebook.com/arduino-mega-2560-rev3-mainboard-with-usb-cable-atmega2560-16au-blue>

[5] Arduino - PinMapping2560 . Arduino - Home [online]. [cit. 14.03.2018] Dostupné z: <https://www.arduino.cc/en/Hacking/PinMapping2560>

[6] Forbidden. Forbidden [online]. [cit. 11.03.2018] Dostupné z: <https://www.sainsmart.com/products/dht22-am2302-digital-temperature-humidity-measurement-sensor>

[7] Arduino and DS3231 Real Time Clock Tutorial - HowToMechatronics. How To Mechatronics [online]. Copyright © 2018 HowToMechatronics.com. All rights reserved. [cit. 11.03.2018]. Dostupné z: <https://howtomechatronics.com/tutorials/arduino/arduino-ds3231-real-time-clock-tutorial/>

[8] How to Build an Arduino Soil Moisture Sensor Circuit. Learning about Electronics [online]. [cit. 11.03.2018] Dostupné z:

<http://www.learningaboutelectronics.com/Articles/Arduino-soil-moisture-sensor-circuit.php>

[9] KitsGuru Soil Moisture Sensor Module, DC 5V for Arduino and Other MCU KitsGuru, KG003: Amazon.in: Industrial & Scientific . Online Shopping site in India: Shop Online for Mobiles, Books, Watches, Shoes and More - Amazon.in [online]. Copyright © 1996 [cit. 11.03.2018]. Dostupné z: <https://www.amazon.in/KitsGuru-Moisture-Sensor-Module-Arduino/dp/B00XU8MJ4E>

[10] Ultrasonic Sensor HC-SR04 and Arduino Tutorial. How To Mechatronics [online]. Copyright © 2018 HowToMechatronics.com. All rights reserved. [cit. 11.03.2018]. Dostupné z: <https://howtomechatronics.com/tutorials/arduino/ultrasonic-sensor-hc-sr04/>

[11] Arduino-shop.cz: VELKOOBCHOD, MALOOBCHOD S ARDUINEM [online]. Copyright © [cit. 11.03.2018]. Dostupné z: <https://arduino-shop.cz/docs/produkty/0/75/1460321123.pdf>

[12] Arduino servo motor | arduino-shop.cz. Arduino-shop.cz: VELKOOBCHOD, MALOOBCHOD S ARDUINEM [online]. Copyright © Copyright ECLIPSE s.r.o. [cit. 12.03.2018]. Dostupné z: https://arduino-shop.cz/arduino/897-arduino-servo-motor-1420669476.html?gclid=Cj0KCQjwOLVBRCZARIsADPLtJ0ahNsKp8zx4Go4SwbeNn9h6-FiMZH5hVfjop6vdkwTddbphFmTV3QaAuitEALw_wcB

[13] LCD 2004 znakový displej žlutý s I2C // Arduino - Dexhal. Dexhal.cz – Internetový obchod - Dexhal [online]. [cit. 12.03.2018] Dostupné z: https://dexhal.cz/displeje/413-lcd-2004-znakovy-displej-zluty-s-i2c-arduino_el-at-ds-20x4yl.html

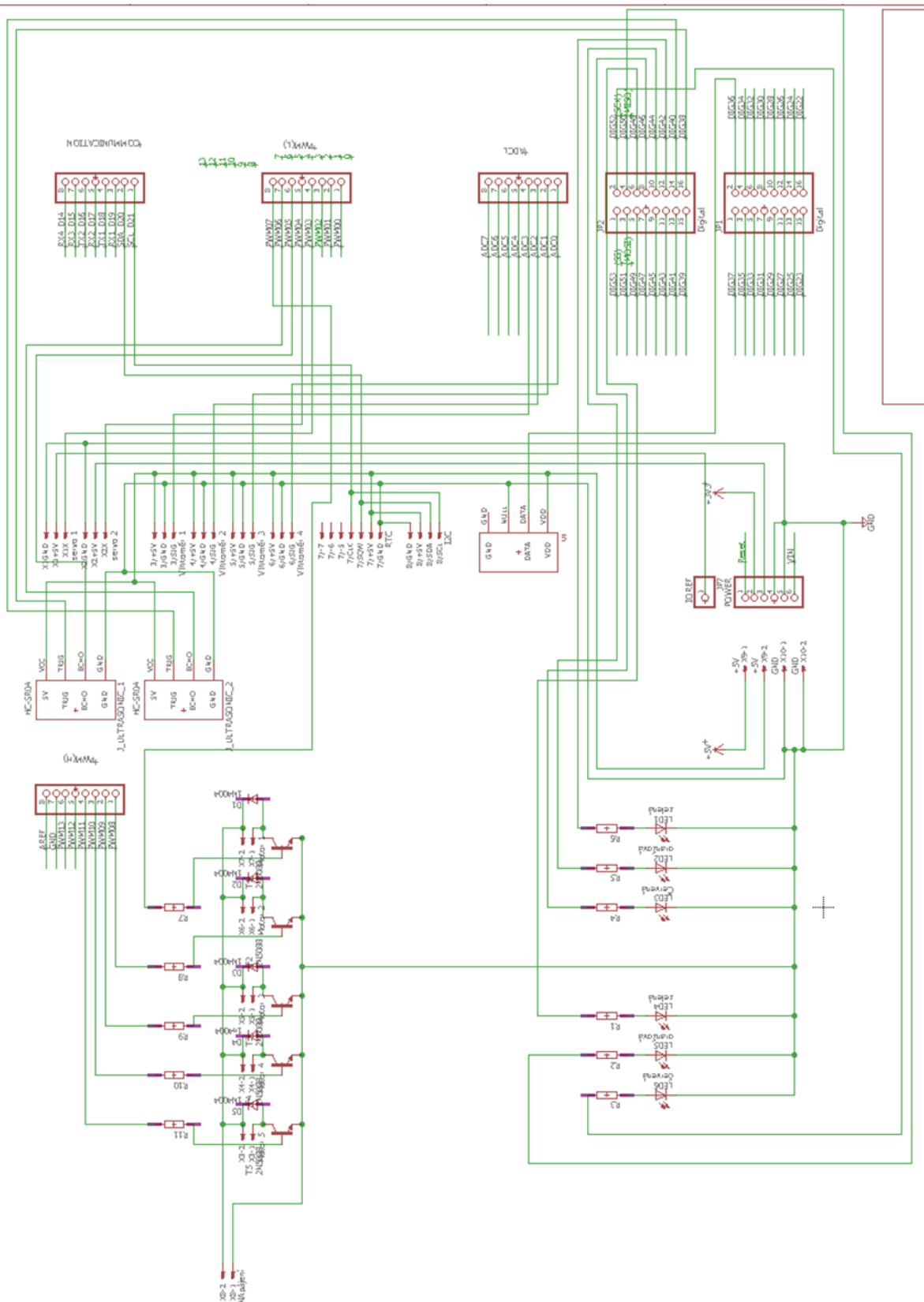
9.3. Seznam grafických příloh

- 1) Zapojení DPS shieldu pro Arduino mega 2560 R3 v EAGLE schematic
- 2) Schéma osazení DPS shieldu pro Arduino mega 2560 R3 v EAGLE board
- 3) Předloha pro DPS shieldu pro Arduino mega 2560 R3
- 4) Vývodové schéma Arduina Mega 2560 R3
- 5) Zapojení v aplikaci Fritzing
- 6) Zapojení a rozmístění komponentů uvnitř krabice
- 7) Vnitřek skleníku

9.4. Seznam souborů na CD

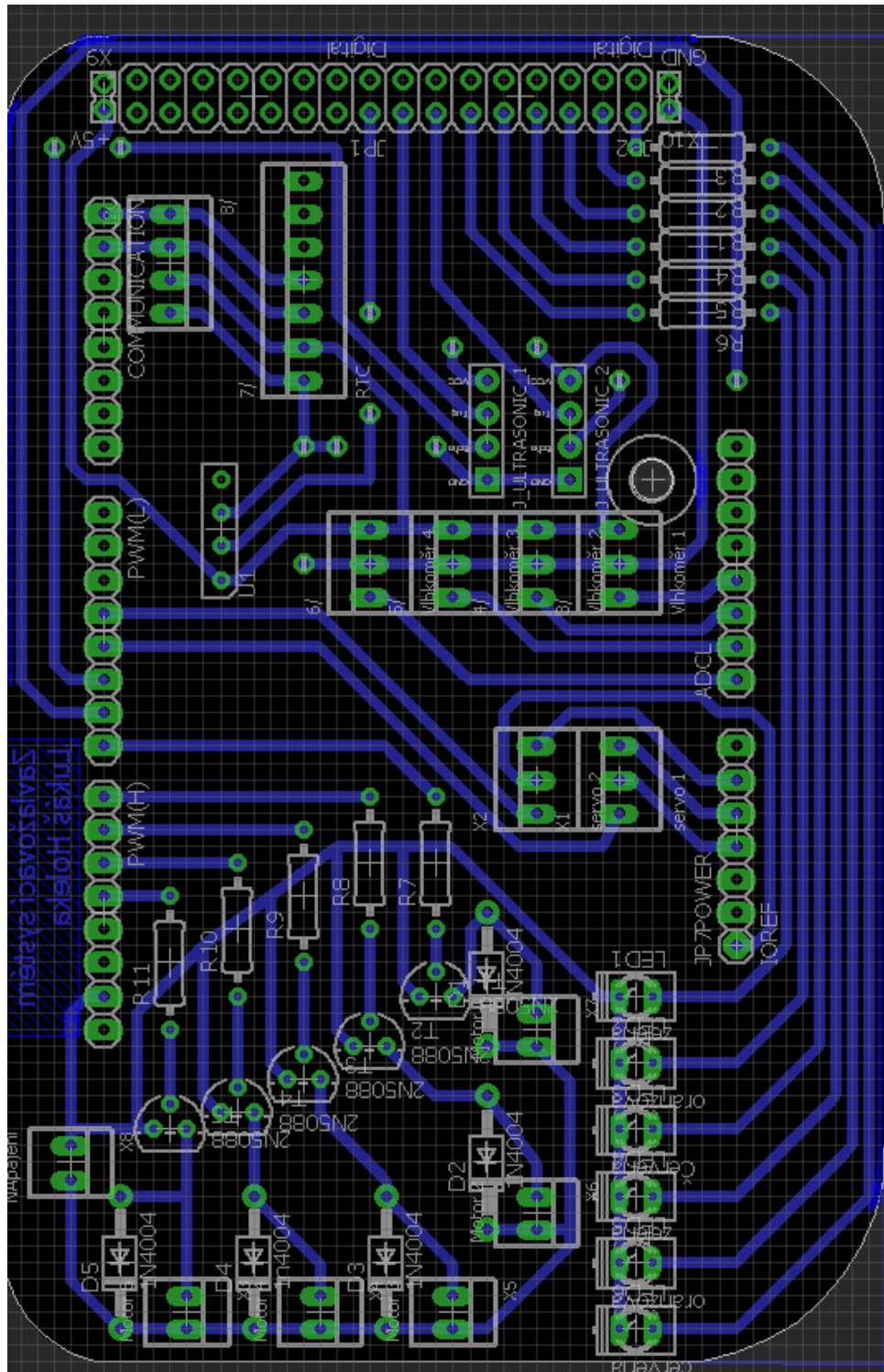
Složka	Název souboru	Popis obsahu souboru
Dokumentace	Dokumentace.pdf, Dokumentace.doc	Technická dokumentace zařízení
Přílohy	Přílohy.pdf, Přílohy.doc	Přílohy k dokumentaci
EAGLE	Arduino MEGA blank shield.sch, Arduino MEGA blank shield.brd	DPS EAGLE schematic, EAGLE board – Shield pro Arduino mega 2560
Zavla_ovac__syst_ m_Luk__Holeka	Zavla_ovac__syst_m_Luk_ __Holeka. ino	Program pro Zavlažovací systém skleníku

Příloha č. 1



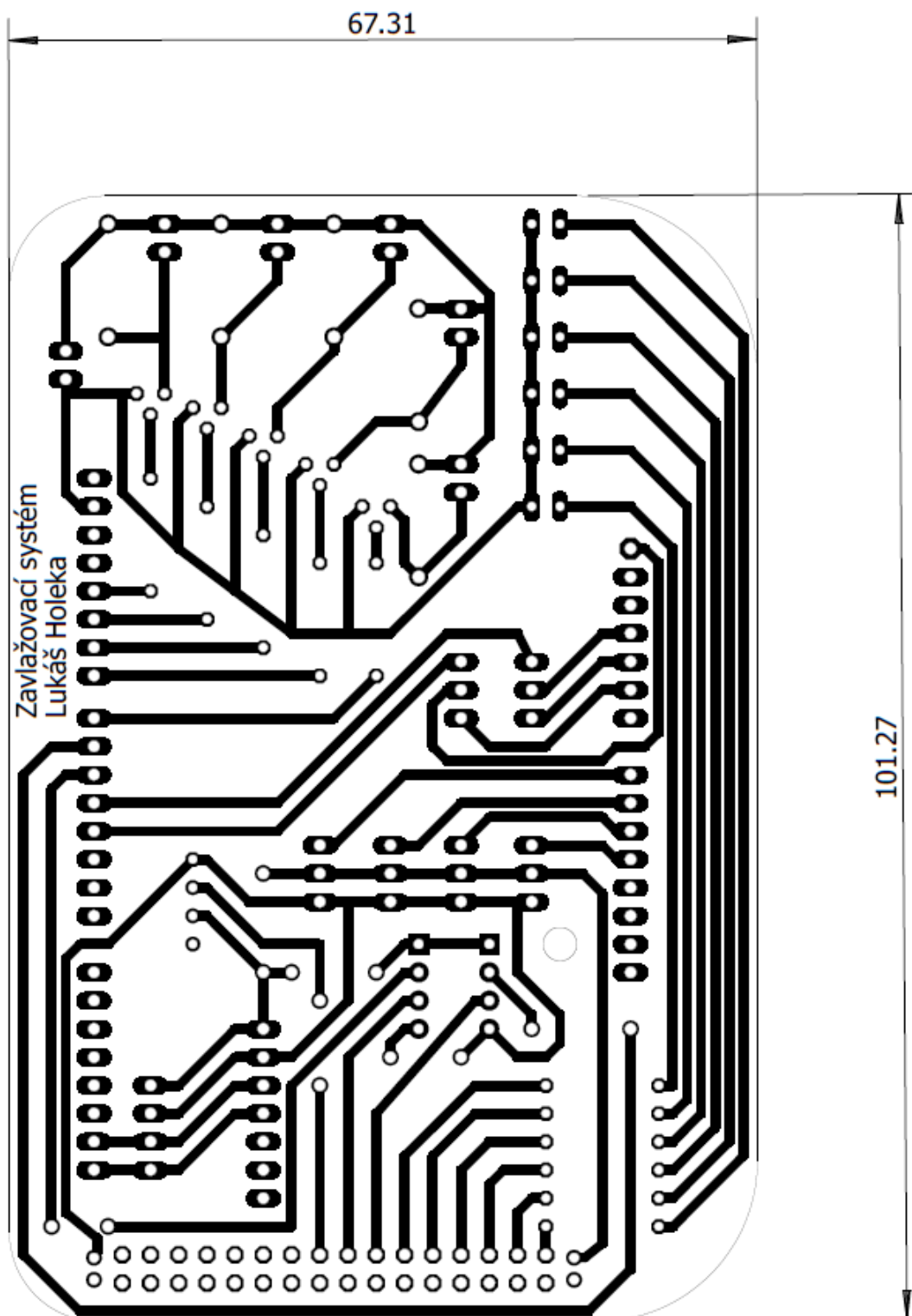
Obr. 1 Zapojení DPS shieldu pro Arduino mega 2560 R3 v EAGLE schematic

Příloha č. 2



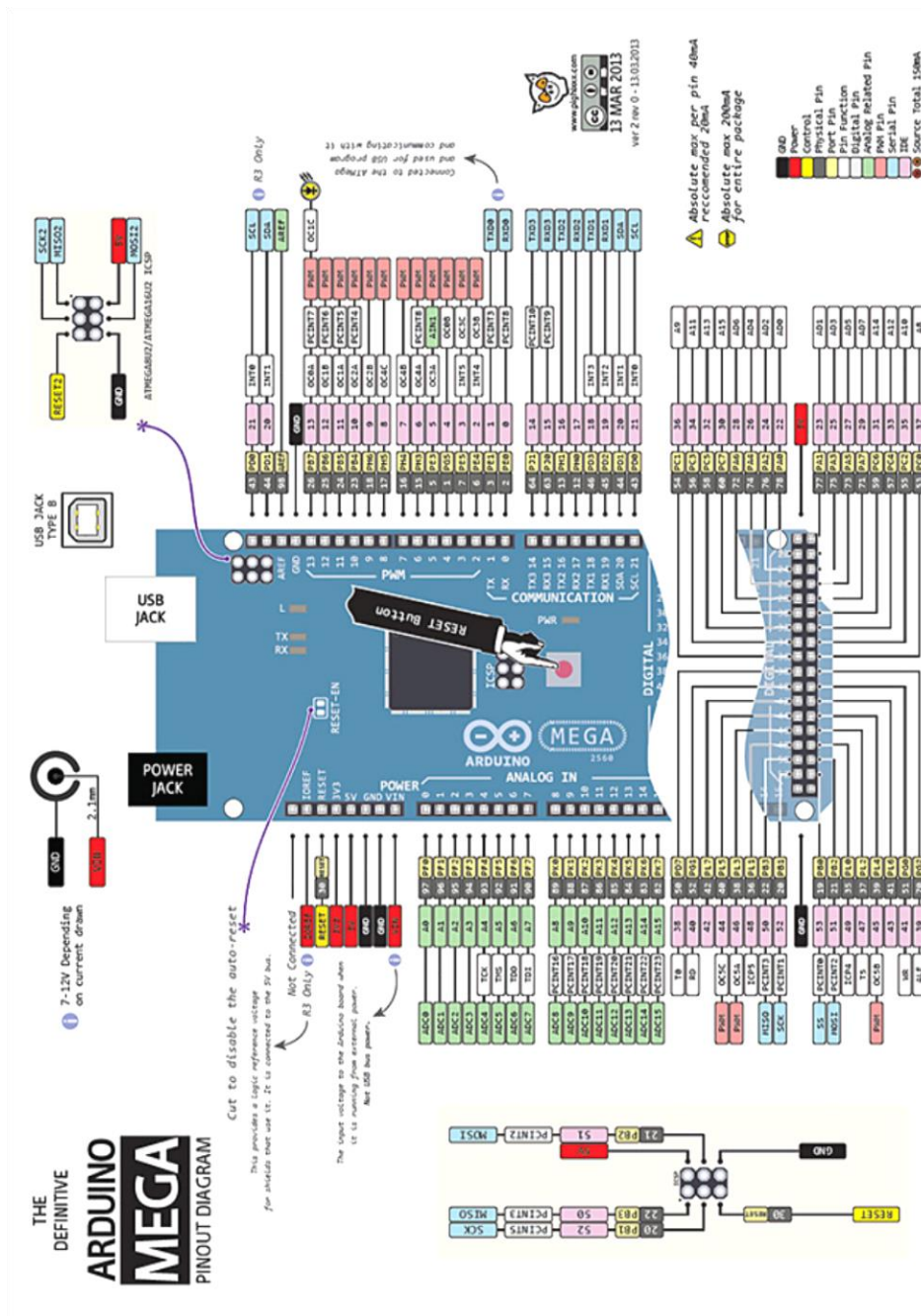
Obr. 2 – Schéma osazení DPS shieldu pro Arduino Mega 2560 R3 v EAGLE board

Příloha č. 3



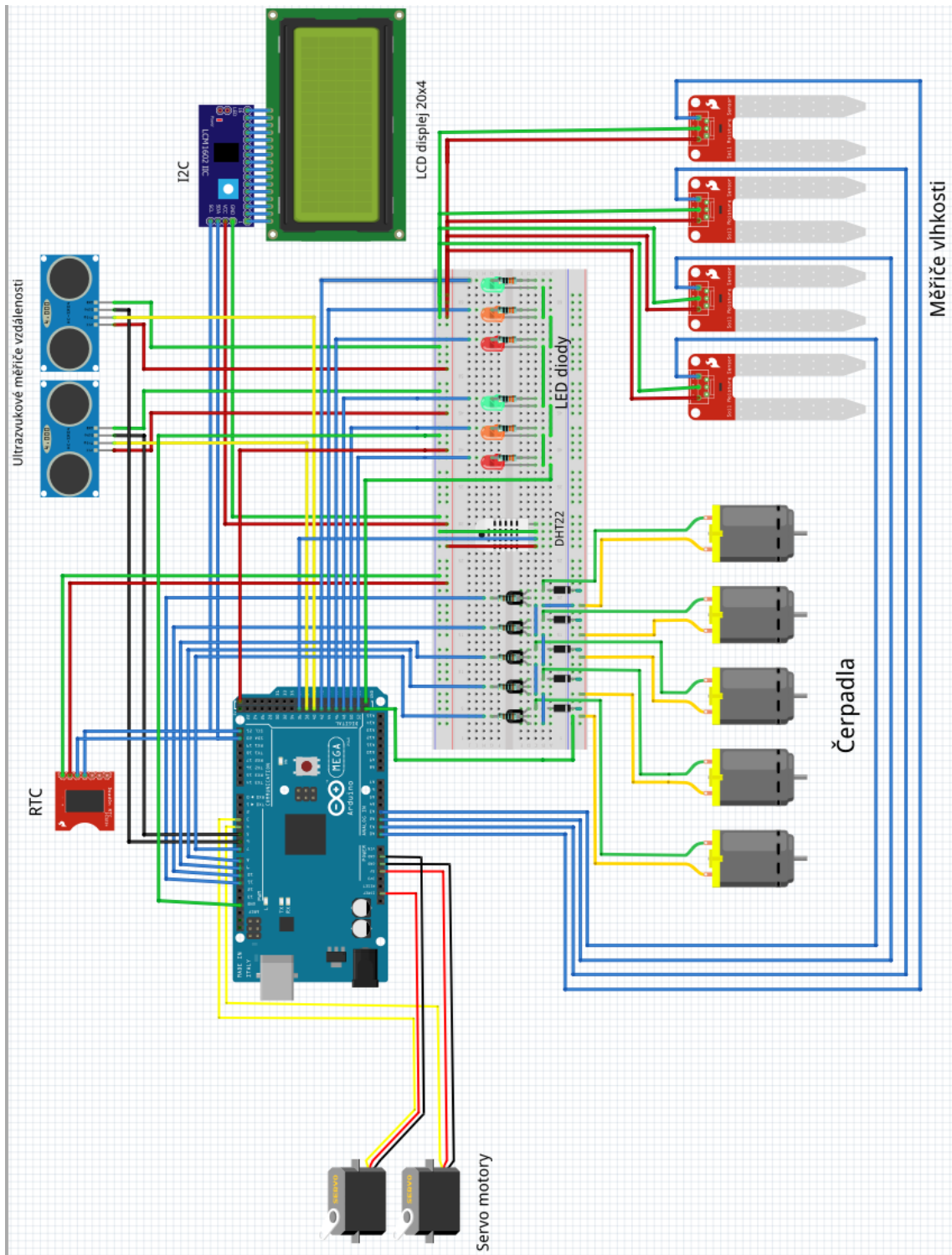
Obr. 3 Předloha pro DPS shieldu pro Arduino mega 2560 R3

Příloha č. 4



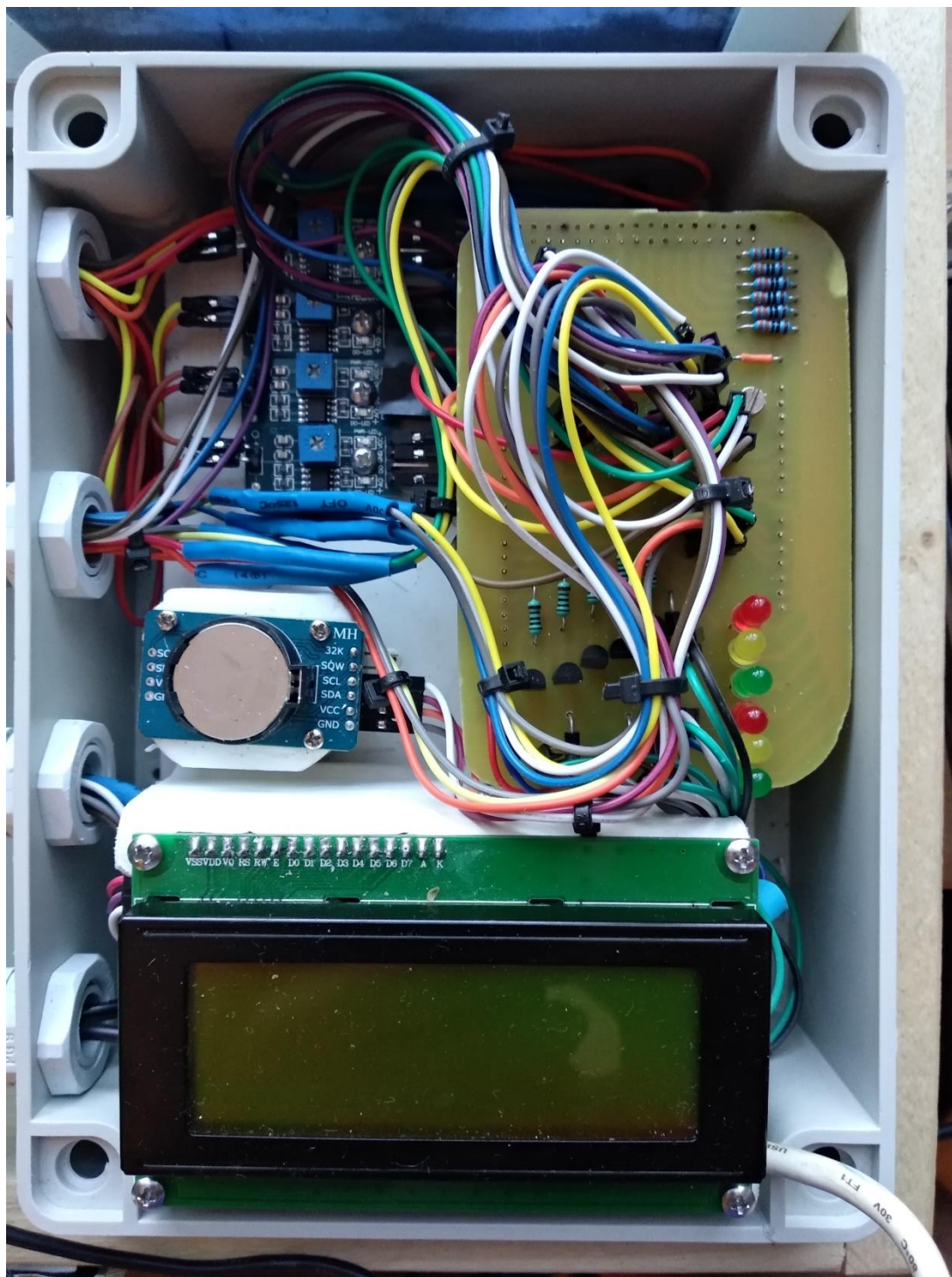
Obr. 4 – Vývodové schéma Arduino Mega 2560 R3

Příloha č. 5



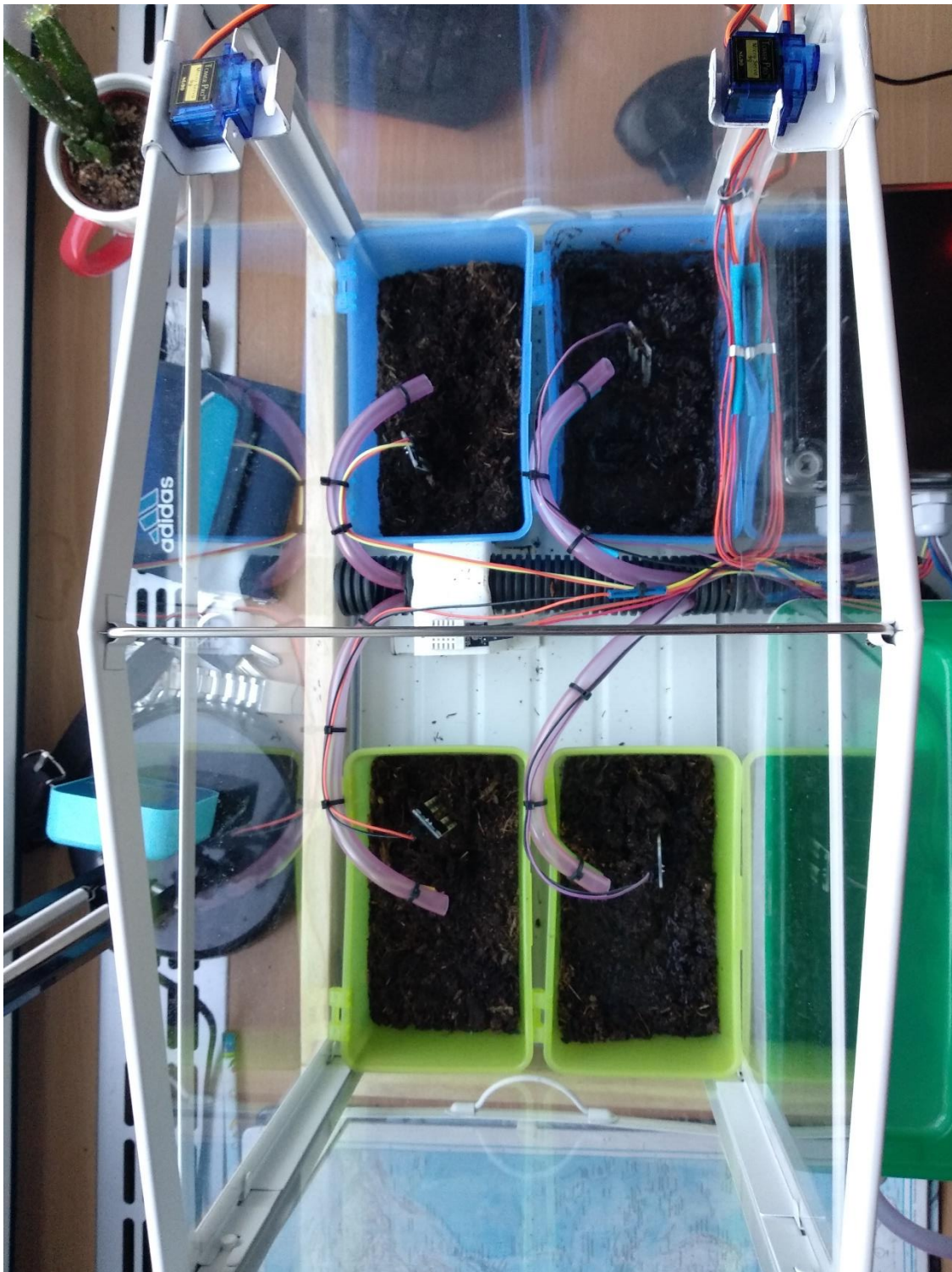
Obr. 5 Zapojení v aplikaci Fritzing

Příloha č. 6



Obr. 6 Zapojení a rozmístění komponentů uvnitř

Příloha č. 7



Obr. 7 Vnitřek skleníku