

Mikropočítačový systém pro monitorování terária

Jana Svozilová

Bakalářská práce
2016



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Jana Svozilová**

Osobní číslo: **A14937**

Studijní program: **B3902 Inženýrská informatika**

Studijní obor: **Informační a řídicí technologie**

Forma studia: **kombinovaná**

Téma práce: **Mikropočítačový systém pro monitorování terária**

Téma anglicky: **A Microcontroller System for Monitoring Terrariums**

Zásady pro vypracování:

1. Zvolte vhodné komponenty pro snímání veličin a kontrolu životního prostředí terarijního živočicha.
2. Navrhněte zapojení a funkce ovládacího programu pro systém monitorování terária.
3. Realizujte navržený systém, včetně programového vybavení pro použitý mikropočítač.
4. Ověřte funkci zařízení a zhodnoťte dosažené výsledky.

Rozsah bakalářské práce: -
Rozsah příloh: -
Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **BUMBA, Jiří. Programování mikroprocesorů. Brno: Computer Press, 2011. ISBN 9788025128381**
2. **BARR, Michael a Anthony J MASSA. Programming embedded systems. 2nd ed. Sebastopol: O'Reilly, 2006, xxi, 301 s. ISBN 978-0-596-00983-0.**
3. **MANN, Burkhard. C pro mikrokontroléry: ANSI-C, kompilátory C, spojovací programy – linkery, práce s ATMELE AVR a MSC-51, příklady programování v jazyce C, nástroje pro programování, tipy a triky. Praha: BEN, 2003. ISBN 80-730-0077-6.**
4. **MARGOLIS, Michael. Arduino cookbook. 2nd ed. Sebastopol, Calif.: O'Reilly, 2012, xx, 699 p. ISBN 1449313876.**
5. **MASSIMO BANZI. Getting started with Arduino. 2nd ed. Farnham: O'Reilly, 2011. ISBN 9781449309879.**
6. **MONK, Simon. Programming Arduino. New York: McGraw-Hill, 2012. ISBN 9780071784221.**

Vedoucí bakalářské práce: **Ing. Jan Dolinay, Ph.D.**
Ústav automatizace a řídicí techniky
Datum zadání bakalářské práce: **19. února 2016**
Termín odevzdání bakalářské práce: **27. května 2016**

Ve Zlíně dne 19. února 2016

doc. Mgr. Milan Adámek, Ph.D.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

Prohlašuji, že

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

Prohlašuji,

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

Ve Zlíně, dne 26.5.2016



.....
podpis diplomanta

ABSTRAKT

Cílem této bakalářské práce je vytvořit jednoduché zařízení zajišťující ideální životní podmínky terarijního živočicha. Zařízení sleduje za pomoci mikropočítače Arduino teplotu spolu s vlhkostí a zajišťuje přirozené denní osvětlení. Vlhkost a teplota jsou zobrazovány na LCD displeji. Pokud teplota není v žádaných hodnotách, je termostaticky upravena. V teoretické části je popsána historie vzniku mikropočítačů, jejich současná podoba a popis jednotlivých druhů mikropočítačů Arduino. V praktické části jsou popsány součástky, postup návrhu a programové řešení zařízení.

Klíčová slova: mikropočítač, Arduino, terárium, teplota

ABSTRACT

The aim of this work is to develop a simple device providing ideal living conditions of animal in terrarium. The equipment monitors temperature along with humidity and provides natural daylight with the help of Arduino microcontroller. Humidity and temperature are displayed on the LCD. If the temperature is not at the desired levels, it is thermostatically regulated. The theoretical part describes the history of microcomputers, their current appearance and description of each type Arduino microcontrollers. The practical part describes the components, process design and device programming solutions.

Keywords: microcontroller, Arduino, terrarium, temperature

Poděkování a motto

Tímto bych chtěla poděkovat vedoucímu bakalářské práce panu Ing. Janu Dolinayovi, Ph.D. za odborné vedení, cenné rady a pomoc při řešení problémů, kterou mi poskytl při jejím vypracování. Také bych chtěla poděkovat rodině a zaměstnavateli za podporu v průběhu studia.

Motto: „Je lepší položit otázku a vypadat hloupě pět minut, než se vůbec nezeptat a být hloupý celý život.“

Konfucius

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

OBSAH

OBSAH.....	6
ÚVOD	9
I. TEORETICKÁ ČÁST.....	10
1 HISTORIE MIKROPOČÍTAČŮ	11
1.1 PRVNÍ MIKROPOČÍTAČ	11
1.2 VÝVOJ.....	12
2 SOUČASNÁ PODOBA MIKROPOČÍTAČŮ.....	13
2.1 ROZDĚLENÍ MIKROPOČÍTAČŮ DLE ARCHITEKTURY	13
2.1.1 VON NEUMANNOVA ARCHITEKTURA	13
2.1.2 HARVARDSKÁ ARCHITEKTURA	14
2.2 MIKROKONTROLERY	14
2.2.1 VÝVOJOVÝ MIKROKONTROLER PICAXE.....	15
3 VÝVOJOVÁ PLATFORMA ARDUINO	16
3.1 VZNIK ARDUINA	16
3.2 DRUHY ARDUINO PLATFORM	16
3.2.1 ARDUINO MINI	17
3.2.2 ARDUINO LILYPAD.....	17
3.2.3 ARDUINO UNO	17
3.2.4 ARDUINO YÚN	18
3.2.5 ARDUINO MEGA 2560.....	18
3.2.6 ARDUINO ESPORA.....	19
3.2.7 ARDUINO ROBOT	19
3.2.8 ARDUINO INTEL GALILEO.....	20
3.2.9 DALŠÍ TYPY ARDUINO PLATFORM	20
II. PRAKTICKÁ ČÁST	21
4 VLASTNOSTI SYSTÉMU PRO ŘÍZENÍ TERÁRIA	22
5 POUŽITÉ KOMPONENTY	24
5.1 NEPÁJIVÉ KONTAKTNÍ POLE.....	24
5.2 MIKROPOČÍTAČ DCCDUINO MEGA 2560	24
5.2.1 KLONY ARDUINA.....	24
5.2.2 VÝVOJOVÉ PROSTŘEDÍ ARDUINO IDE	25
5.2.3 POPIS DCCDUINO MEGA 2560	25
5.3 DVOUŘÁDKOVÝ LCD DISPLAY QAPASS 1602A	26
5.4 AKTIVNÍ PIEZOREPRODUKTOR TMB12A05	26
5.5 SENZOR TEPLoty A VLHKOSTI DHT22	27
5.6 MODUL REÁLNÉHO ČASU DS3231	27
5.7 MODUL ČTYŘ RELÉ S GALVANICKÝM ODDĚLENÍM	28
5.8 PASIVNÍ SOUČÁSTKY	28
5.8.1 POTENCIOMETR 50K OHM.....	28
5.8.2 MIKROSPÍNAČE	29
6 ZAPOJENÍ JEDNOTLIVÝCH SOUČÁSTEK	30
6.1 DVOUŘÁDKOVÝ LCD DISPLAY QAPASS 1602A	30

6.2	POTENCIOMETR 50K OHM	32
6.3	AKTIVNÍ PIEZO REPRODUKTOR TMB12A05	32
6.4	SENZOR TEPLoty A VLHKOSTI DHT22	33
6.5	MODUL REÁLNÉHO ČASU DS3231	34
6.6	MODUL ČTYŘ RELÉ S GALVANICKÝM ODDĚLENÍM	35
6.7	MIKROSPÍNAČE	36
7	PROGRAMOVÉ VYBAVENÍ	38
7.1	NASTAVENÍ ČASU DO RTC MODULU	38
7.2	NAČTENÍ KNIHOVEN	38
7.3	NASTAVENÍ PINŮ A DEFINICE PROMĚNNÝCH	39
7.4	POPIS FUNKCE VOID SETUP()	39
7.5	POPIS FUNKCE VOID LOOP()	40
8	OVĚŘENÍ FUNKČNOSTI ZAŘÍZENÍ	43
8.1	INSTALACE ZAŘÍZENÍ DO TERÁRIA	43
8.2	TESTOVÁNÍ FUNKČNOSTI PERIFERIÍ	44
8.3	VYHODNOCENÍ	44
	ZÁVĚR	45
	SEZNAM POUŽITÉ LITERATURY	46
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	48
	SEZNAM OBRÁZKŮ	50
	SEZNAM PŘÍLOH	ERROR! BOOKMARK NOT DEFINED.

ÚVOD

Dnešní mikropočítače usnadňují práci ve spoustě odvětvých a nejrůznějších aplikacích. Najdeme je také ve většině elektronických zařízeních v běžném životě. Podoba mikrokontrolerů závisí na daném použití a očekávání z hlediska funkcí.

Mikropočítače jsou elektronické součástky v podobě integrovaného obvodu a mají uspořádání umožňující provádění logických a aritmetických operací podle řídicího programu za účelem získání výstupních hodnot.

Nejrozšířenější vývojové mikrokontrolery jsou Arduino. Ve své podstatě se jedná o velmi inteligentní a interaktivní elektronické stavebnice, které vznikly v Itálii jako levné řešení pro studenty technologií a v současné době se velmi rozšiřují hlavně mezi domácími kutily. Existuje nespočet přídavných senzorů a modulů, které umožňují velkou variabilitu v použití a práce s nimi je naučná z hlediska elektronických obvodů a programování.

Hlavním tématem této bakalářské práce je systém monitorování terária realizovaný právě na platformě Arduino. Teoretická část se zabývá historií mikropočítačů již od počátku jejich vzniku. Dále je jejich problematika rozvedena k vývojovým platformám Arduino a popisu jednotlivých výrobních typů.

Praktická část bakalářské práce je soustředěna na konkrétní využití Arduina v problematice zajištění ideálního prostředí terarijního živočicha. Součástí je návrh řešení, popis součástí a programového vybavení, zajišťujícího chod všech připojených periférií.

I. TEORETICKÁ ČÁST

1 HISTORIE MIKROPOČÍTAČŮ

Bez moderních počítačů si ve své podstatě dnes již nedokážeme svůj život vůbec představit. Ať už se jedná o stolní počítač, notebook, tablet a v neposlední řadě také chytrý telefon. Ona počítačová revoluce, jak je dnes známá, započala již v 60. letech minulého století a to pouze prostou myšlenkou.

Laická veřejnost v této době neměla ani tušení, k čemu se takový počítač používá. Ten kdo alespoň tušil co počítač je, věděl, že tyto velké výpočetní stroje byly používány primárně pro vědecké účely, výzkumy a vládu.

S první myšlenkou použitelnosti počítače pro laickou veřejnost přišel v roce 1960 vizionář J. C. R. Licklider, který s faktickou přesností předpověděl budoucnost výpočetní techniky. Ve svém tehdy nepochopeném článku psal o grafické podobě, ovládání klikáním na obrazovku, elektronických obchodech, sdílení souborů a spoustě dalších ve své době velmi sci-fi scénářích. Jeho předpověď však byla velmi přesná.[1]

1.1 První mikropočítač

První mikropočítač jenž byl plně schopný samostatného provozu byl Commodore PET 2001 (PET = Personal Electronic Transactor) vyvinutý čipovým inženýrem Chuckem Peddlem a firmou Commodore. Názory na vznik prvního all-in-one počítače jsou však různé, ale jistotou je, že zde svou roli sehrála i legendární a úspěšná firma Apple.

V roce 1976 Peddleho navštívil Steve Jobs a Steve Wozniak se svým prototypem počítače Apple II, který se snažili prodat firmě Commodore. Vlastník firmy Jack Tramier však shledával tento výrobek výrobně příliš nákladným a tak zadal Peddlemu úkol, sestrojít podobný, lépe vyhovující kus.[1,5]

V týmu o třech lidech vytvořili odolný obal s monitorem a kazetopáskovou pamětí. Mikroprocesor počítače byl taktován na 1MHz a řídil paměť, klávesnici, obraz a další komponenty, které se připojovaly přes rozhraní. Paměť RAM byla ve dvou verzích, a to buď 4KB nebo 8KB. Monitor měl 40x25 znaků a na svou dobu zobrazoval skvělou grafiku. Operačním systémem se stal BASIC, který napsal Bill Gates a Paul Allen. Tento systém se stal prvním krůčkem k rozsáhlé produkci operačních systémů firmy Microsoft.[1,5]



Obr. 1 Historicky první mikropočítač na světě –
Commodore PET

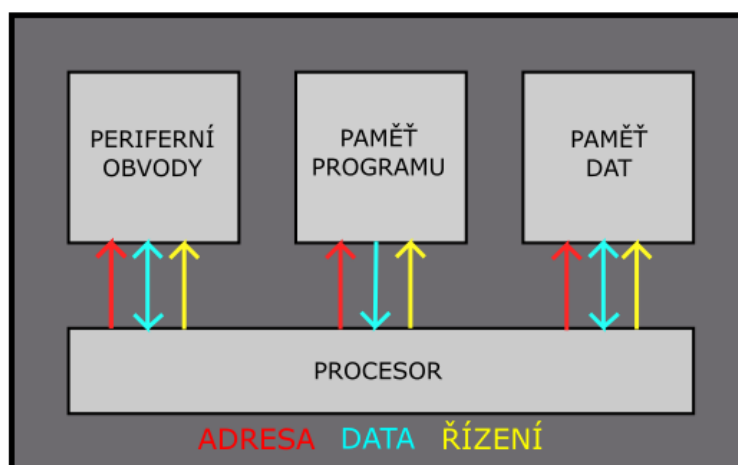
1.2 Vývoj

Po vytvoření Commodore PET se jeho vzhled a vybavenost neustále zlepšovala. Neustále se vydávali nové verze např. s lepší pamětí, obrazem a příslušenstvím. Vývoj se tedy nezastavil, v podstatě šlo o evoluci.[5]

Úžasně rychlý rozvoj těchto systémů byl způsoben velkým pokrokem v technologické výrobě elektronických součástek, integrovaných obvodů, výkonostních a optoelektornických součástek. Díky tomu vznikla tak malá počítačová zařízení, jako jsou mobilní telefony, smartwatch a spousta dalších. Pro jejich vyjmenování by nestačila celá kniha. Mikropočítače jsou všude kolem nás, moderní doba zkrátka vyžaduje moderní postupy.[4]

2 SOUČASNÁ PODOBA MIKROPOČÍTAČŮ

Mikropočítače jsou konstruovány jako integrované obvody. Mikroprocesor nemůže samostatně pracovat a proto je zapotřebí dalších součástí. To je programová paměť, datová paměť a periferní obvody. Propojení těchto částí zajišťuje soustava sběrnic (viz obr. 2). Samotná sběrnice také umožňuje rozšiřování počítače o další součásti. Architektura mikropočítačů je však v mnohých případech odlišná. Existují dva základní typy počítačové architektury.



Obr. 2 Schéma počítače

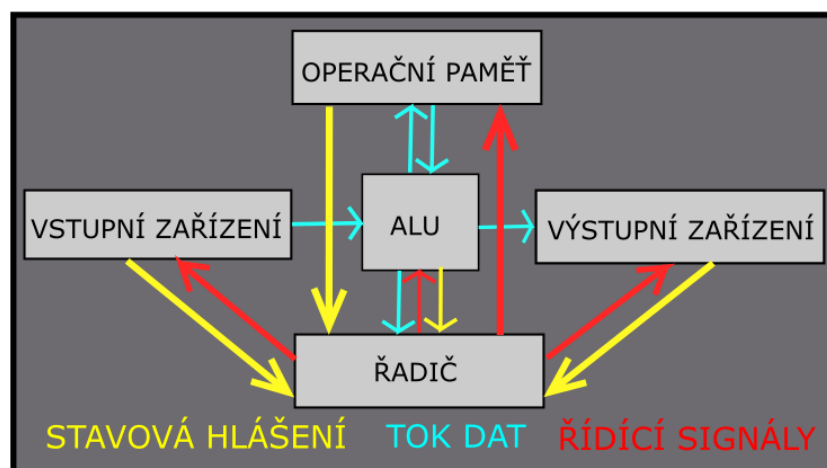
2.1 Rozdělení mikropočítačů dle architektury

Většina dnešních mikropočítačů je založena na architektuře Johna von Neumanna, která vznikla ve 40. letech a popisuje počítač jako samočinný. Dřívější stroje měli svůj ovládací program pevně stanovený a pokud by bylo nutno tento program změnit, znamenalo by to výměnu hardwaru. Tato Harvardská architektura se používá dodnes ve spoustě jednoduchých zařízení, nejznámějším příkladem jsou jednoduché kalkulačky.[6]

2.1.1 Von Neumannova architektura

Von Neumannova koncepce popisuje počítač se společným paměťovým prostorem pro data i instrukce, která jsou uložena do slov a slabik stejného formátu. Toto uspořádání má výhodu především v tom, že není potřeba rozlišovat instrukce, které jednotlivě přistupují k paměti dat a programu. Díky tomu může být čip jednodušší, avšak v tomto případě trpí jiná funkční část. Nevýhodou této architektury je pomalý přenos, který vzniká díky přenosu obou typů dat po jedné sběrnici.[7]

Operační paměť uchovává právě prováděný program, zpracovává data a výsledné výpočty. ALU modul – aritmetickologická jednotka provádí všechny aritmetické a logické operace. Řadič je řídicí jednotka veškeré činnosti částí počítače. K řízení používá řídicí signály, které jsou dopravovány k různým částem. Na tyto signály jsou vytvářeny reakce nebo taky stavy modulů, které se zasílají zpět řadiči ve formě stavových hlášení. Vstupní zařízení je takové, přes které vniká program a data. Naopak výstupní zařízení data ve formě zpracovaných výsledků odvádí. (viz. obr. 3)



Obr. 3 John Von Neumannova architektura

2.1.2 Harvardská architektura

Změnou Von Neumannovy architektury vznikla Harvardská architektura, která měla za úkol zrychlit přenos. Paměť určená pro program a data se rozdělila do dvou samostatných částí. Díky tomu je možné načítat současně instrukce z programové paměti a pracovat s datovou pamětí. Paměť pro program může být i jiného typu, například ROM, PROM, EPROM nebo FLASH. Použití této architektury je vhodné zejména pro aplikace, které nevyžadují změnu ovládacího programu a běží stále stejným způsobem.[6,7]

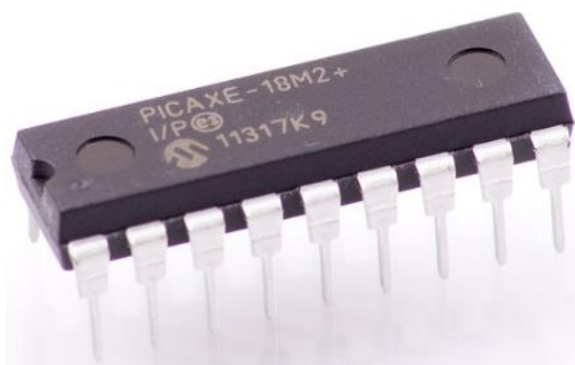
2.2 Mikrokontrolery

Mikrokontroler je jednočipový mikropočítač, který má ve svém jádru integrované všechny základní prvky nutné k běhu mikroprocesoru. Svou architekturou je určen zejména pro řízení různých mechanismů. Na jednom čipu je integrovaná paměť RAM, programová paměť (ROM, PROM nebo EPROM), vstupní a výstupní obvody a rozhraní pro řízení různých periférií (např. časovač). Jedná se tedy o řídicí systémy v reálném čase. Použití těchto mikrokontrolerů je nekonečné, od běžných domácích spotřebičů jako je pračka,

sporák nebo DVD přehrávač až po složité aplikace v leteckém průmyslu. Historicky nejznámějším výrobcem mikrokontrolerů je firma Intel, která v této oblasti zapříčinila velký rozvoj, ale od roku 2007 už jednočipové mikropočítače nevyrábí.[3,4]

2.2.1 Vývojový mikrokontroler PICAXe

Mezi vývojovými gadgets platformami existuje velmi oblíbený mikrokontroler PICAXe. Při jeho výrobě do něj byl nahrán speciální kód, který umožňuje programování skrze sériový třívodičový kabel. Tento mikrokontroler je postaven na programovacím jazyce BASIC a jeho architektura spadá do kategorie PIC. Jedná se o polovodičovou součástku, která je založena na harvardské architektuře a díky velikosti, ceně a spotřebě je její použití velmi univerzální. Nejlépe se však hodí pro využití v amatérské robotice. Roboti vybavení kontrolerem PICAXe vykonávají většinu obvyklých pohybů (řídí elektromotory a serva) a zpracovávají signály ze senzorů a čidel. [4]



Obr. 4 Mikrokontroler PICAXe 18M2

3 VÝVOJOVÁ PLATFORMA ARDUINO

Samostatnou rodinou mikropočítačů jsou také nejrozličnější vývojové platformy. Tyto desky s čipy vyrábí spousta firem a pravděpodobně nejznámější z nich je Arduino. Ve srovnání se samostatnými mikrokontrolery jsou open source vývojové platformy nenáročné na sestavení a následnou implementaci pro požadovaný projekt. Mikrokontroler je již napájen na základní desce, která obsahuje dle výrobního typu spousty různých vstupů a výstupů v libovolných počtech (např. USB, HDMI, Ethernet port). Samozřejmostí je nepřeberné množství dále připojitelných zařízení, které dále rozšiřují možnosti. Takové moduly můžou být například různé senzory, displeje, tlačítka, ovladače, Wifi, GPS a Bluetooth shieldy. Všechny tyto součásti dělají z Arduina interaktivní stavebnici pro velkou škálu využití.

3.1 Vznik Arduina

První vývojová platforma Arduino vznikla v roce 2005 s předpokladu vyvinout levný a jednoduchý set primárně pro studenty se zálibou v mikropočítačích. Od počátku vzniku se po celém světě prodalo již několik stotisíc desek a vývoj běží dál. Nedávno byl ohlášen vývoj nové výkonnější platformy Arduino Galileo vznikající za podpory firmy Intel. A protože je Arduino open source, vznikají také různé klony a neoficiální typy.



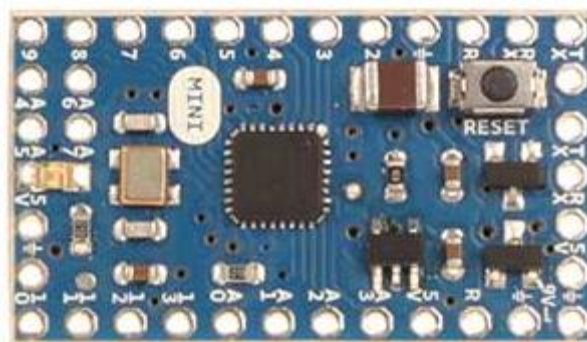
Obr. 5 Značka Arduino

3.2 Druhy Arduino platforem

Většina typů Arduino desek má společný mikroprocesor Atmel, který je spojen s dalšími komponenty na designově jednotné základní desce. Další součástí platforem je u většiny typů převodník, který zajišťuje připojení počítače k čipu. Některé typy jej však nemají z důvodu úspory místa nebo jej mají zabudovaný přímo v čipu.

3.2.1 Arduino Mini

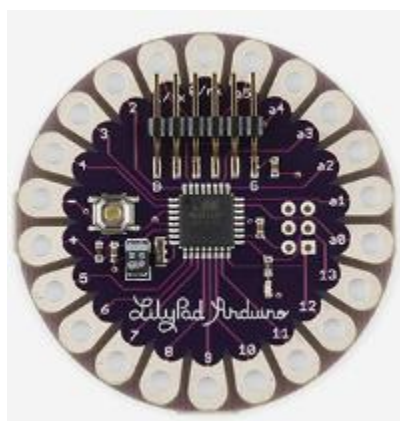
Nejmenší verze Arduina byla konstruována pro úsporu místa a kvůli tomu zde chybí USB port. Programovat lze tedy pouze za pomoci externího USB převodníku. Díky mikroprocesoru Atmega328 s taktem 16 MHz tento typ nezaostává za ostatními většími deskami. Jeho použití je možné například v různých vypínačích a ovladačích. Velikost je pouhých 3 x 1,8 cm.[8]



Obr. 6 Arduino Mini

3.2.2 Arduino LilyPad

Zajímavou verzí je bezesporu LilyPad. Již na první pohled je jasně zřetelná jeho netypičnost. Tento typ je určen k všití přímo do textilu, jeho spoje jsou tvořeny vodivostními nití a je vyráběn v několika verzích a také bez USB slotu. Díky tomu lze vyrobit například různě reflektivní a blikající bunda.[9]



Obr. 7 Arduino LilyPad

3.2.3 Arduino Uno

Patrně nejpoužívanější a nejrozšířenější typ Arduino platformy je Uno. Z této základní verze se dále vyvinulo několik dalších. Naprosto stejný model je Arduino Geniuno, které je

určeno k prodeji mimo USA. Dalším již mírně odlišným typem vycházejícím z Uno je Arduino Ethernet, místo USB portu zde nalezneme Ethernet port a navíc také slot na microSD karty. Praktickou odnoží je i Arduino Bluetooth, jehož komunikace probíhá, jak název napovídá, za pomoci bluetooth technologie a přímo zabudovaného modulu.[10]



Obr. 8 Arduino Uno/Arduino Genuino

3.2.4 Arduino Yún

Model tohoto Arduina je na pohled téměř stejný jako Uno, ale ve skutečnosti se jedná o mnohem výkonnější model. Na desce je čip Atheros, který zvládne rozběhnout méně náročnou linuxovou distribuci Linino a je určen k bezdrátové komunikaci.[11]



Obr. 9 Arduino Yún

3.2.5 Arduino Mega 2560

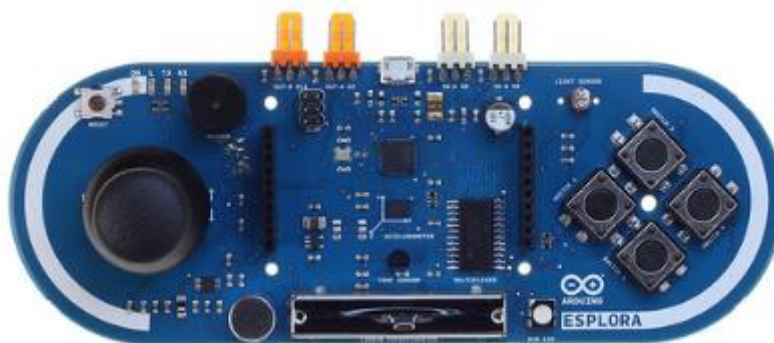
Arduino Mega 2560 je v podstatě větší a vybavenější verzí Uno. Pro připojení k počítači nevyužívá FTDI převodník, ale místo něj obsahuje druhý mikročip, který je naprogramovaný jako USB převodník na sériový port. Deska obsahuje 54 digitálních a 16 analogových vstupů/výstupů a také LED diodu. Pro distribuci mimo USA se obdobně jako u Uno používá značka Arduino Geniuno MEGA 2560.[12]



Obr. 10 Arduino MEGA 2560/Arduino Geniuno
MEGA 2560

3.2.6 Arduino Esplora

Hybridní deska Arduino Esplora je primárně určena pro výrobu vlastního ovladače k herní konzoli. Na základní desce vzhledu gamepadu obsahuje joystick, základní tlačítka, potenciometr, bzučák, teploměr, akcelerometr a piny pro připojení TFT LCD displeje.[13]



Obr. 11 Arduino Esplora

3.2.7 Arduino Robot

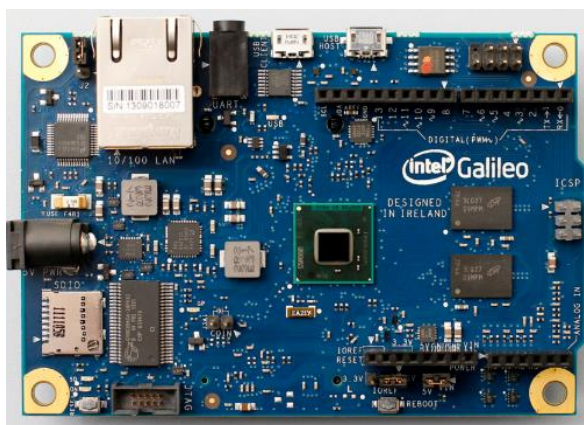
Jak už název napovídá, tento druh Arduino platformy je určen pro robotické sestavy. Ve své podstatě by se dalo říct, že už robotem je. Sestava obsahuje dva čipy, jeden zajišťuje ovládání motorů a druhý snímá data ze senzorů a podle toho samostatně řídí robota. Cena tohoto modelu je už nepatrně vyšší, u nás se pohybuje okolo 5500 Kč.[14]



Obr. 12 Arduino Robot

3.2.8 Arduino Intel Galileo

Arduino Intel Galileo je první model vyvinutý ve spolupráci s firmou Intel. Tento typ obsluhuje 32 bitový mikročip Intel Quark taktovaný na 400MHz. Na základní desce nechybí USB, dokonce ve dvou kusech, microSD slot, Ethernetový port a mini-PCI Express slot.[15]



Obr. 13 Arduino Intel Galileo

3.2.9 Další typy Arduino platforem

Existuje velká spousta dalších typů a jejich různých modifikací. Před nákupem je vhodné všechny tyto typy projít a zvolit pro vlastní projekt nejvhodnější variantu. Univerzální platformy jsou Uno a Mega 2560. Jedno ale mají všechny typy společné, jednoduchost programování, rozmanitost a nízkonákladovost.

II. PRAKTICKÁ ČÁST

4 VLASTNOSTI SYSTÉMU PRO ŘÍZENÍ TERÁRIA

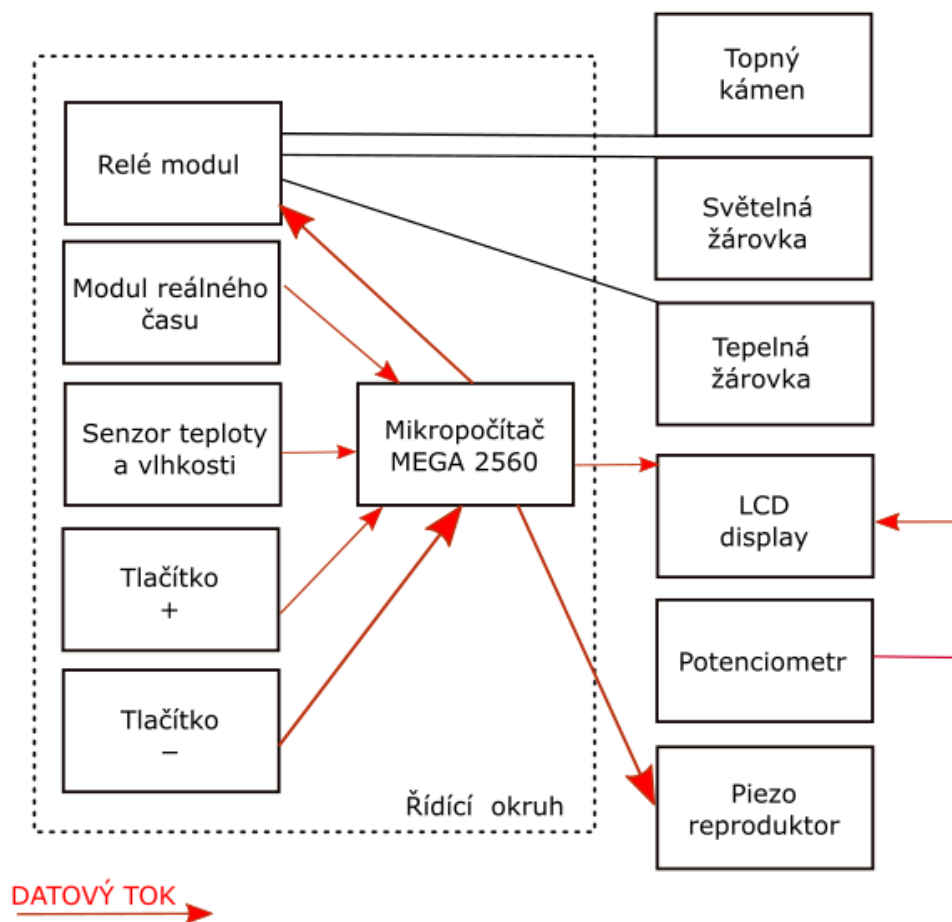
Systém je sestaven pro zajištění ideálního životního prostředí terarijního živočicha, v tomto případě hada. Užovka červená potřebuje stabilní teplotu. Vlhkost je v tomto případě vcelku zanedbatelná, avšak neměla by klesnout pod 50%. Tuto veličinu hlídá senzor vlhkosti, který je zabudován v jednom modulu spolu se senzorem teploty. Při poklesu vlhkosti pod 50% se na displeji zobrazí upozornění, které zmizí po detekování vyšší vlhkosti. Tato je zajištěna ručním mechanickým rozprašovačem vody aplikací přímo do prostoru terária.

Teplota je v teráriu řízena principem termostatu a je nastavena na stabilních 25 °C. Přítápění je realizováno výhřevnou terarijní žárovkou, která neprodukuje světlo. Pokud se chce had více zahřát, využije topný kámen, který je vyhřátý na 40 °C po dobu 8 hodin v letním období od března do října a podobu 10 hodin v období zimním.

Denní světlo v teráriu je zajišťováno pomocí LED žárovky, která svítí ve stejných intervalech, jako je spuštěn topný kámen.

V některých případech je na krátkou dobu nutno zvýšit teplotu. Ihned po krmení se doporučuje zvýšení teploty cca o 3°C. Vytvořený systém s touto možností počítá a je vybaven tlačítky + a -, které umožňují krátkodobé zvýšení teploty a následné snížení po krmení. Stisk tlačítek je doprovázen pípnutím aktivního piezo reproduktoru v odlišných zvucích.

Teplota a vlhkost je po celou dobu zobrazována na dvouřádkovém displeji, jehož jas lze nastavit potenciometrem.



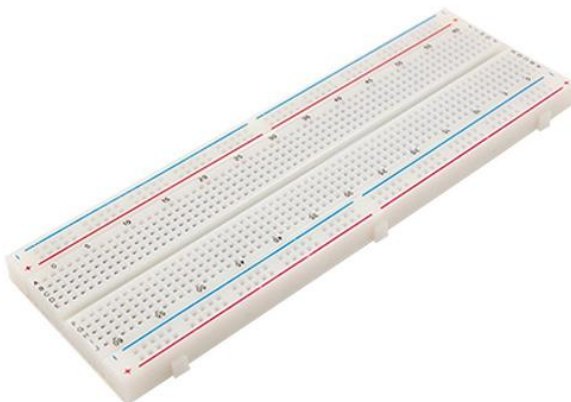
Obr. 14 Datový přenos mezi součástkami

5 POUŽITÉ KOMPONENTY

Řídící obvod terária je realizován za pomoci nejnutnějších součástek v minimalizovaném množství. Srdcem celé sestavy je vývojový kit Arduino MEGA 2560, který je pro toto použití nadmíru dostačující a poskytuje variabilitu při změně nebo přidání dalších komponentů.

5.1 Nepájivé kontaktní pole

Jako základ konstrukce jsem zvolila větší vodivostní pole o velikosti 164x54x9 mm a označením modelu MB-120. Obsahuje 830 pinů a dvě napájecí větve. Tyto větve jsou praktické z důvodu malého množství napájecích pinů, které jsou integrovány přímo na Arduino. Díky jednomu kabelu tedy propojíme celou linku, do které pak můžeme zapojit větší množství součástek. Stejná linka je zde i z důvodu vyvedení pinu GND (zem).



Obr. 15 Nepájivé vodivostní pole

5.2 Mikropočítač DCcduino MEGA 2560

Nejdůležitější částí sestavy je mikropočítač DCcduino MEGA 2560. Jedná se o klon typu Arduino MEGA 2560.

5.2.1 Klony Arduina

Díky open-source je na světě dostupná velká spousta takových klonů, jenž jsou mnohem levnější, než originální mikropočítač nesoucí značku Arduino. Ne všechny klony jsou však spolehlivé. Před nákupem je vhodné prověřit shodnost použitých součástek s originálním modelem. V případě odlišných částí se vystavujeme problémům s instalací ovladačů a dalším potížím, například v řízení součástek. Tento model jsem zakoupila přes internetový obchod www.ebay.com a v porovnání s originálem jsem ušetřila okolo tisíce korun.

Ovládání je standardním a originálním programovacím prostředím Arduino IDE, které je volně ke stažení na oficiálních stránkách výrobce v několika verzích pro různé operační systémy.

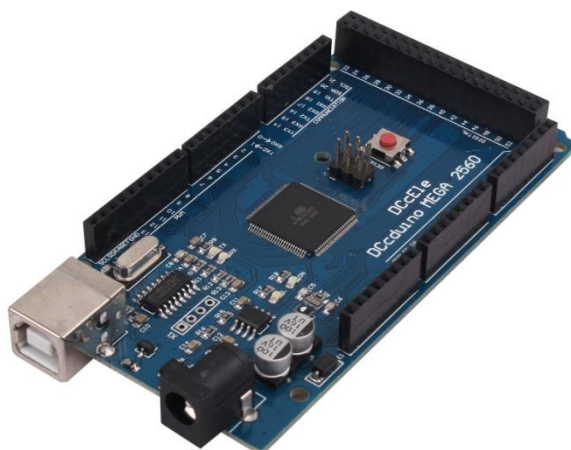
5.2.2 Vývojové prostředí Arduino IDE

Vývojové prostředí pro Arduino je skutečně zdařilé. Obsahuje velmi jednoduché ovládání, které zvládnout i začátečníci. V podstatě vypadá téměř jako textový editor. Software byl vytvořen v programovacím jazyku Java a vznikl z výukového prostředí Processing. Byli přidány různé funkce a také podpora jazyka Wiring.

Podle autorů Wiring jde o jazyk velmi podobný C++. Jedná se však jeho speciální verzi, která usnadňuje práci s s Arduinem. Z objektové syntaxe zbyl pouze jeden operátor a to tečka.

5.2.3 Popis DCcduino MEGA 2560

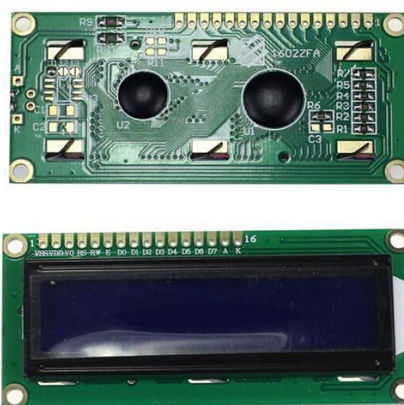
Mikropočítač je založen na mikroprocesoru ATmega2560, který je taktován na 16 MHz. Základní deska obsahuje 54 digitálních vstupních a výstupních pinů. Jako PWM výstupy lze použít 14 pinů, analogových máme 16 a hardware sériových portů UARTs jsou zde 4 kusy. Na desce najdeme také tlačítko reset, napájecí konektor, vstupní konektor a spoustu dalších malých součástek, které zajišťují bezproblémový běh celého mikropočítače. Provozní napětí je 5V, vstupní napětí se pohybuje mezi 7 až 12 volty a krajní limity jsou 6 až 20 voltů. Flash paměť je o velikosti 256 KB, z kterých je 8 KB použito pro zavaděč. Paměť EEPROM má 4 KB a SRAM 8 KB.[12]



Obr. 16 DCcduino MEGA 2560

5.3 Dvouřádkový LCD display QAPASS 1602A

Pro zobrazení měřených hodnot je díky nízké spotřebě vhodný dvouřádkový LCD display z tekutých krystalů, s 16 znaky v každém řádku a integrovaným řadičem. Pracuje na principu paralelního zápisu. Jeho jas lze regulovat za pomoci potenciometru.



Obr. 17 LCD display

5.4 Aktivní piezoreproduktor TMB12A05

Aktivní piezo měnič, zkráceně také bzučák, je použit pro hlasovou indikaci poklesu vlhkosti. Tento model má malé a jednoduché provedení se zapojením na dva piny. Napětí se pohybuje mezi 4 až 7V a frekvence zvuku dosahuje 2300 ± 500 Hz.[16]



Obr. 18 Bzučák

5.5 Senzor teploty a vlhkosti DHT22

Digitální senzor DHT22 patří do kategorie přesnějších typů senzorů pro měření teploty a vlhkosti. Je vyroben z kapacitního čidla vlhkosti, termistoru a malého mikrokontroleru, který zajišťuje převod analogových hodnot na digitální. Spolu s převodem také vykonává výpočet teploty ve °C a vlhkosti v %. Zajímavostí je, že přenos digitálního signálu probíhá pouze po jednom vodiči, což je zajišťováno za pomoci protokolu 1-WIRE-TTL 5V. Rozsah měření vlhkosti je od 0 až do 100% s přesností 2 až 5%. Rozsah měření teploty zvládá v rozmezí od -40 do 125 °C s přesností $\pm 0,5$ °C.[17]



Obr. 19 Senzor teploty
a vlhkosti DHT22

5.6 Modul reálného času DS3231

Modul (hodiny) reálného času je v projektu přidán z důvodu nastavení požadovaného času spínání obou žárovek a topného kamene. RTC modul obsahuje velmi přesný čip DS3231 a poskytuje sekundy, minuty, hodiny, den v týdnu, datum, měsíc a rok. Další funkce jsou korekce pro přestupný rok, datum ukončení, AM/PM nastavení, dva konfigurovatelné budíky a kalendář. Přesný čas se nastavuje pomocí řídicího programu přepsáním dat těsně před nahráním do mikroprocesoru. Modul je zálohován pomocí lithiové baterie CR2032, která je instalována přímo na základní desce a poskytuje zajištění pojistky v případě

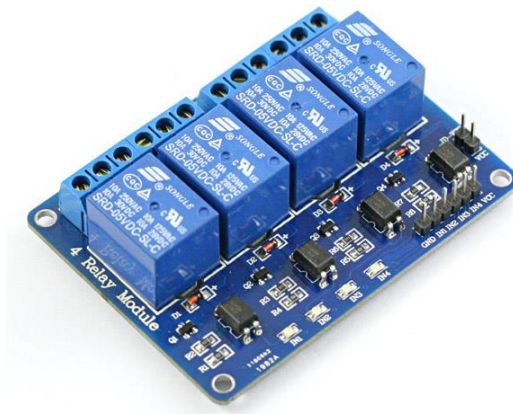
výpadku napájení. Přesnost tohoto typu je ± 5 vteřin za měsíc v rozmezí pracovní teploty 0 až 40 °C, což je v tomto případě zanedbatelná hodnota.[18]



Obr. 20 Modul reálného času DS3231

5.7 Modul čtyř relé s galvanickým oddělením

Pro spínání dvou žárovek a topného kamene byl zvolen modul čtyř relé z důvodu umístění všech relé na jedné základní desce, která zajišťuje jednotné uložení a přehlednost. Relé jsou odděleny galvanicky optočleny, což zajišťuje větší bezpečnost pro Arduino. Při průchodu proudu se navíc rozsvítí jedna ze čtyř diod, které signalizují sepnutí relé.



Obr. 21 Relé modul s optočleny

5.8 Pasivní součástky

5.8.1 Potenciometr 50k Ohm

Pro nastavení jasu displeje je v projektu použit lineární potenciometr 50k Ohm s úhlem rotace 300°. Díky otáčení dochází ke změně napětí (analogové hodnoty), díky čemuž se

mění jas displeje. Odpor potenciometru není příliš podstatný, byl zvolen univerzální lineární s odporem 50k Ohm. U lineárního potenciometru se mění odpor stejnoměrně s natočením ovládací osy.



Obr. 22

Potenciometr

5.8.2 Mikrospínače

Pro okamžitou regulaci teploty v projektu jsou použita dvě malá tlačítka. Jedno tlačítko reaguje jako plus pro zvýšení teploty, a druhé jako minus pro snížení. Rozměry mikrospínače jsou 6 x 6 x 6 mm

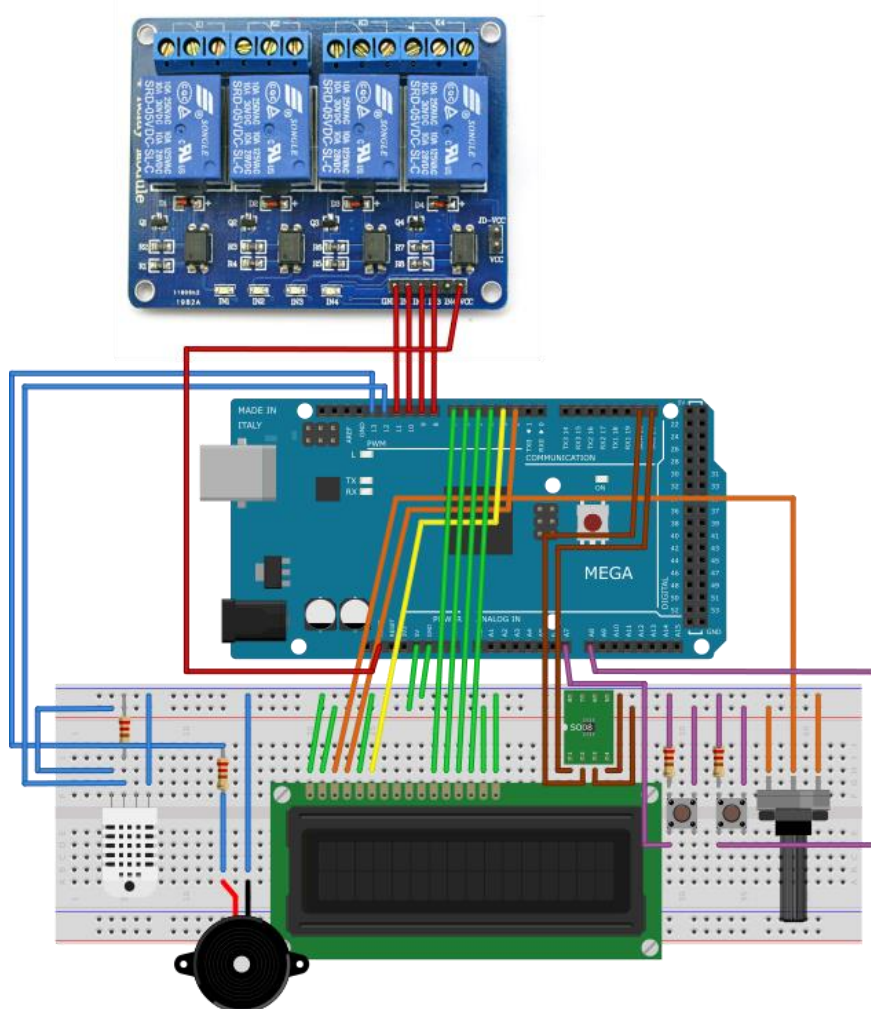


Obr. 23

Mikrospínač

6 ZAPOJENÍ JEDNOTLIVÝCH SOUČÁSTEK

Použité součástky jsou k mikropočítači připojeny za pomoci vodivostního kontaktního pole a vodičů s konektory, které jsou kompatibilní s 2,54 mm distančními kolíky. Tento způsob propojení je výhodný z hlediska jednoduchosti, časové nenáročnosti a variability při případných změnách v projektu.



Obr. 24 Zapojení všech součástek

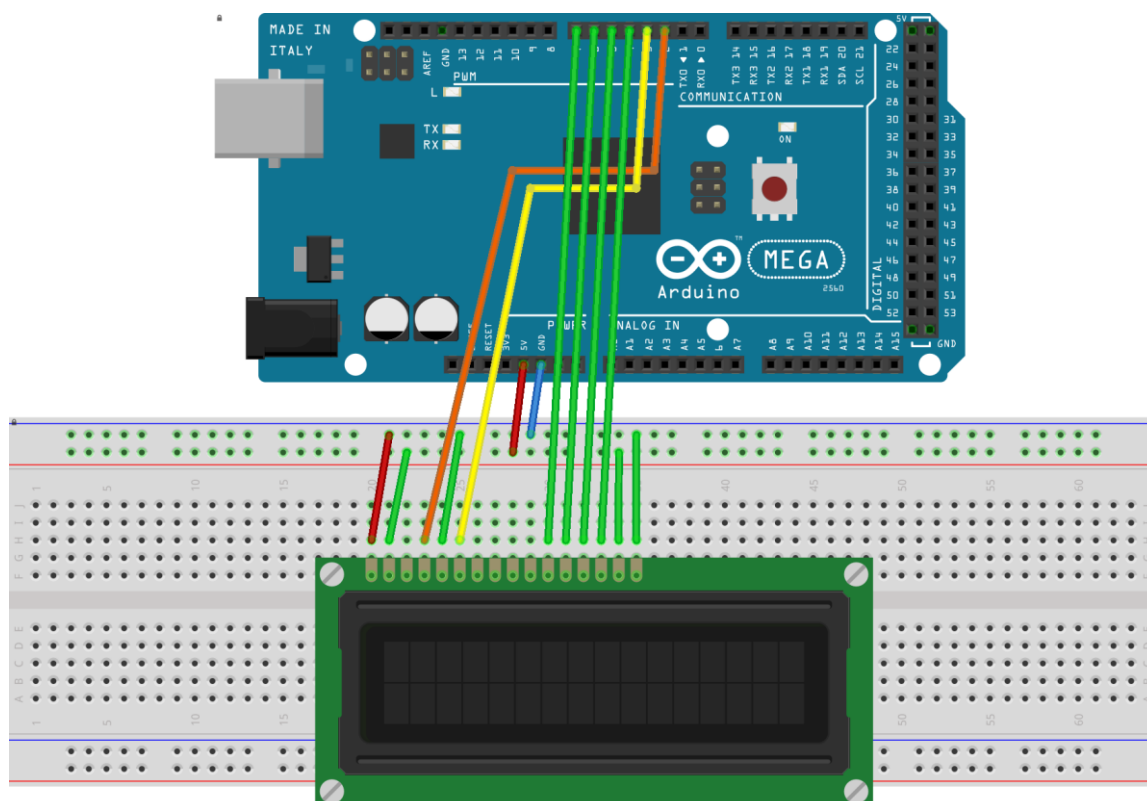
6.1 Dvouřádkový LCD display QAPASS 1602A

Dvouřádkový LCD display má šestnáct pinů. Pro zapojení do mikropočítače bylo použito pouze 12, což je plně dostačující. Z výroby má display na pinech tvrdé drátky, pomocí kterých jej zapojíme přímo do kontaktního pole bez nutnosti jakéhokoliv pájení. Tento model má piny předepsány přímo na základní desce, není tedy nutné zjišťovat přiřazení z datasheetu. Prvním krokem je připojení proudové větve 5V za pomoci červeného vodiče a

dále připojení GND větve pomocí modrého vodiče (viz. obr. 23). Tyto větve následně použijeme pro připojení dalších součástek.

Zapojení:

- 1 VSS – na větev GND
- 2 VDD – na větev 5V
- 4 RS – do pinu 2
- 5 RW – na větev GND
- 11 D4 – do pinu 7
- 12 D5 – do pinu 6
- 13 D6 – do pinu 5
- 14 D7 – do pinu 4
- 6 E – do pinu 3
- 15 A – anoda podsvícení na větev 5V
- 16 K – katoda podsvícení na větev GND



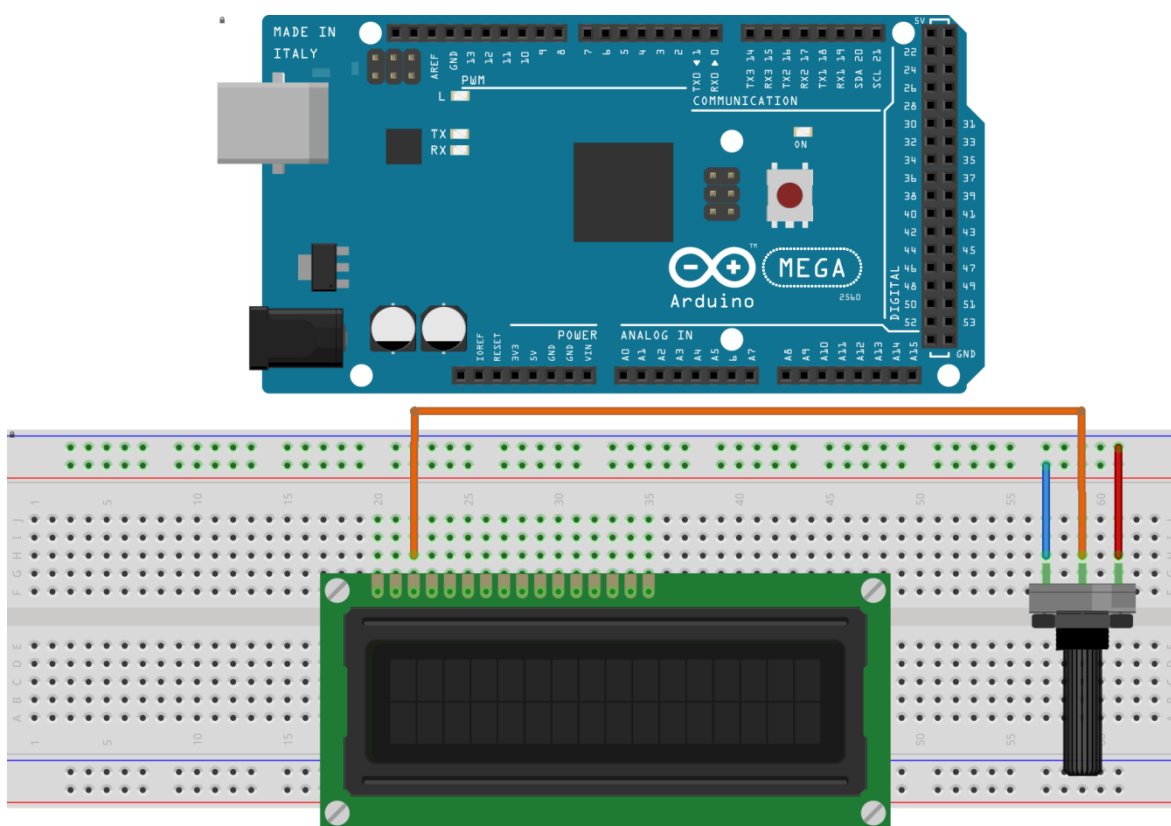
Obr. 25 Připojení LCD displeje

6.2 Potenciometr 50k Ohm

Potenciometr má pouze tři výstupní piny. Krajní piny slouží k připojení GND a napájení 5V, prostřední k LCD displeji (viz. obr. 24).

Zapojení:

- 1 – na větev 5V
- 2 – do LCD pinu 3
- 3 – na větev GND



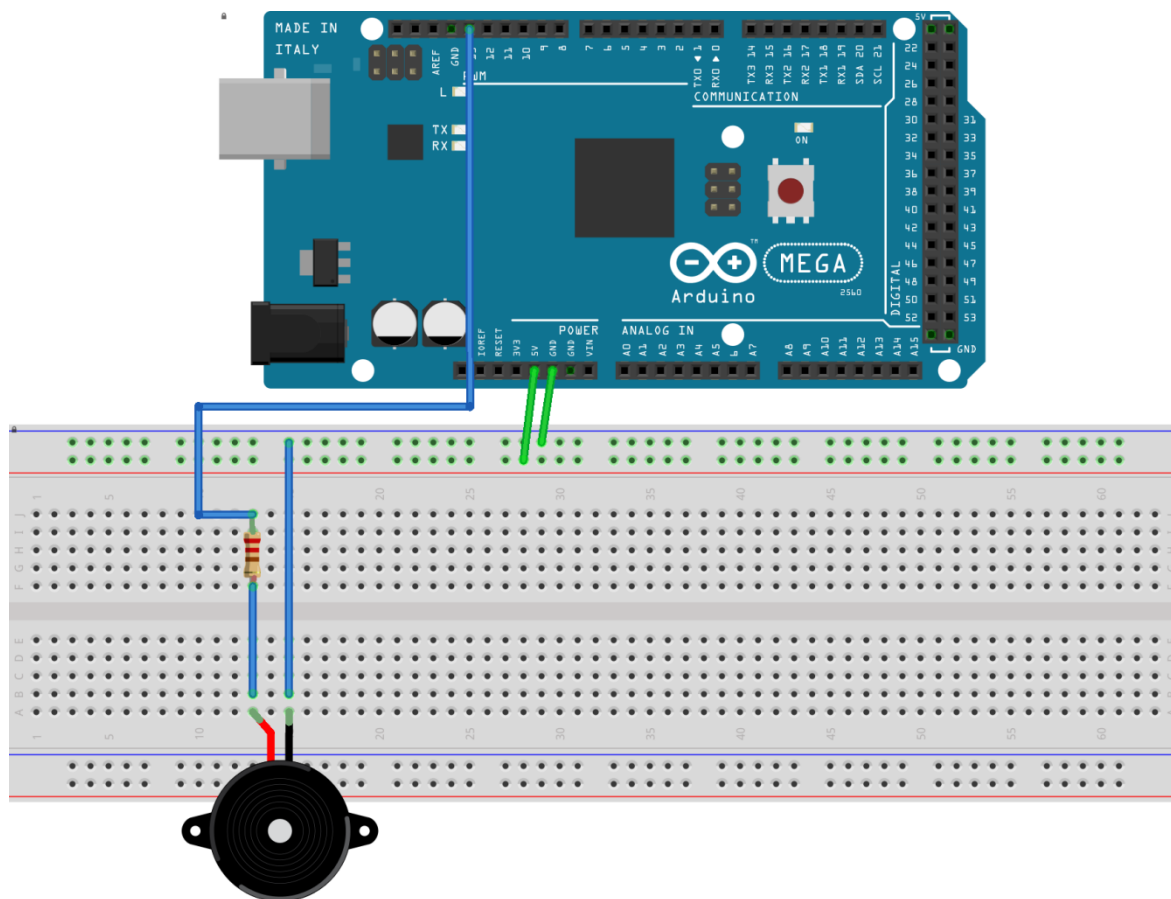
Obr. 26 Připojení potenciometru

6.3 Aktivní piezo reproduktor TMB12A05

Aktivní piezo reproduktor má pouze dva piny. Jeden je připojen skrz 10k Ohm rezistor do mikropočítače a druhý do 5V větve (viz. obr. 25).

Zapojení:

- 1 DATA – do pinu 13
- 2 VCC – na větev 5V



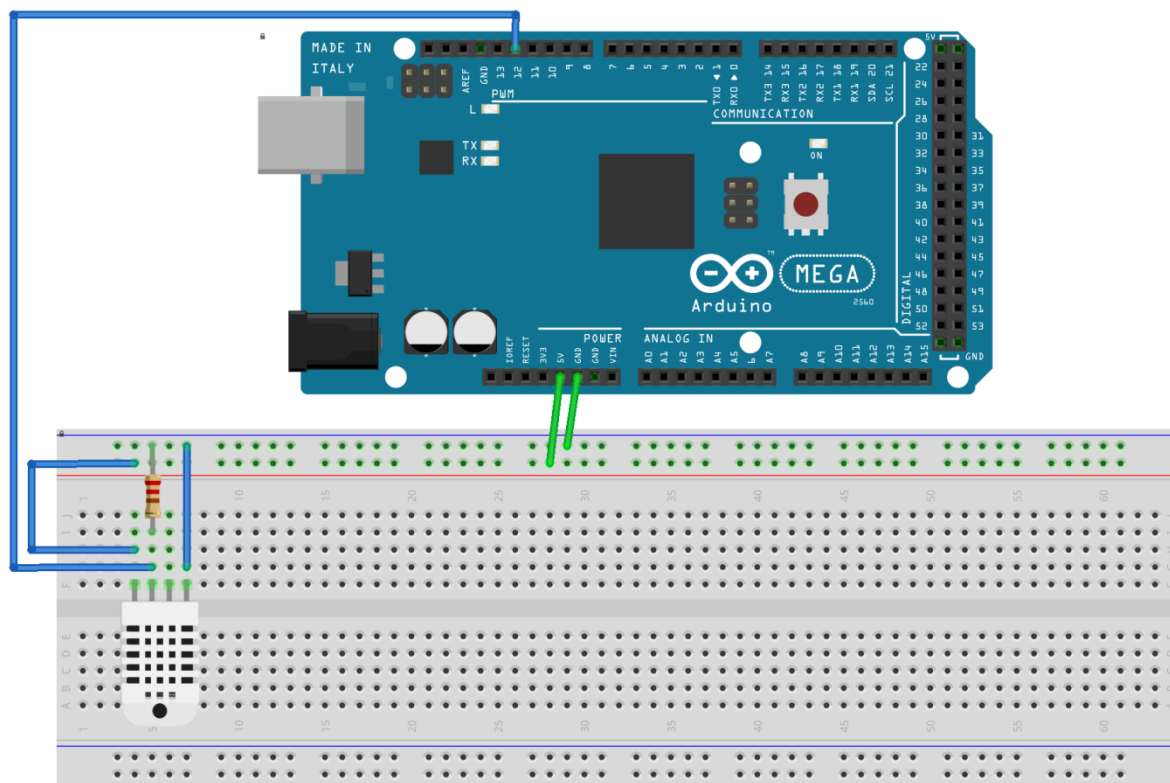
Obr. 27 Připojení piezo reproduktoru

6.4 Senzor teploty a vlhkosti DHT22

Senzor teploty a vlhkosti má čtyři vývody. První je pro napájení, druhý zajišťuje přenos dat, třetí není použit a čtvrtý je zem (viz. obr. 26).

Zapojení:

- 1 VCC - na větev 5V
- 2 DATA – do pinu 12
- 4 GND – na větev GND



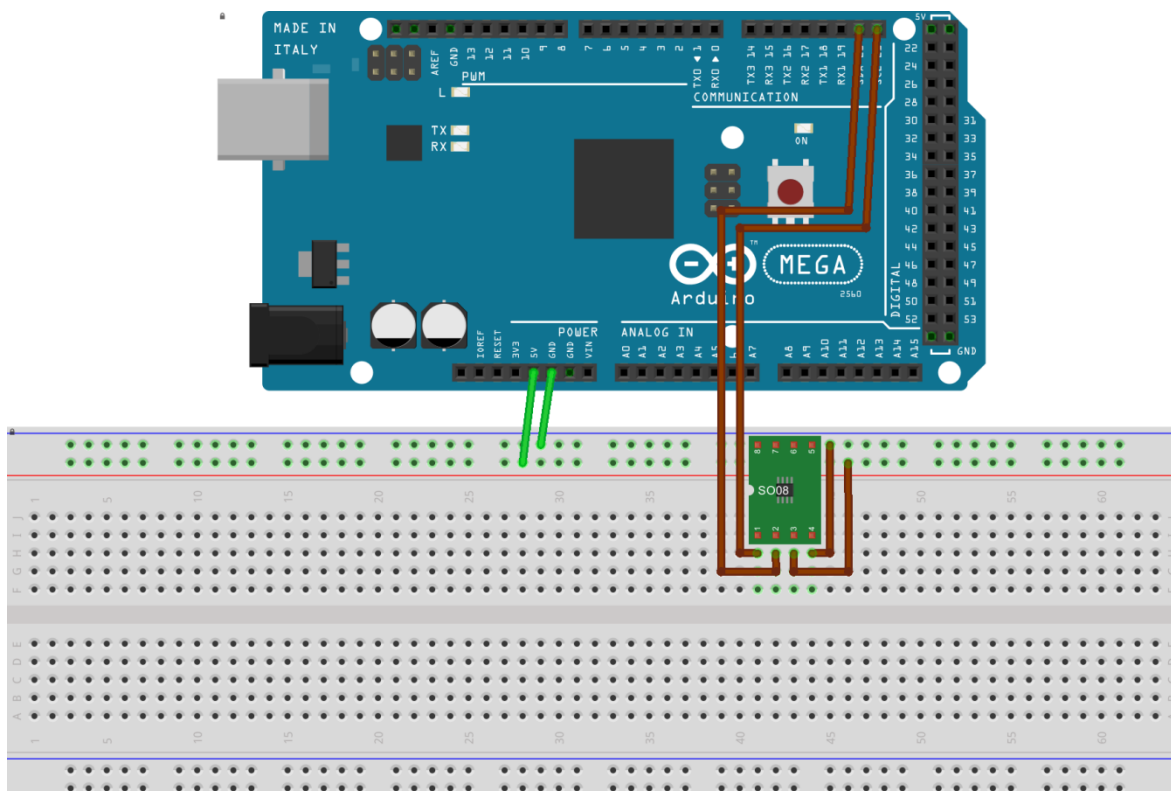
Obr. 28 Připojení senzoru teploty a vlhkosti

6.5 Modul reálného času DS3231

Na modulu reálného času jsou čtyři kontakty. SCL a SDA jsou vedeny do mikropočítače a zbylé dva na napájení a zem. Datový vývod je opět opatřen 10k Ohm rezistorem (viz. obr. 27).

Zapojení:

- 1 SCL – do pinu 21
- 2 SDA – do pinu 20
- 3 GND – na větev GND
- 4 VCC – na větev 5V



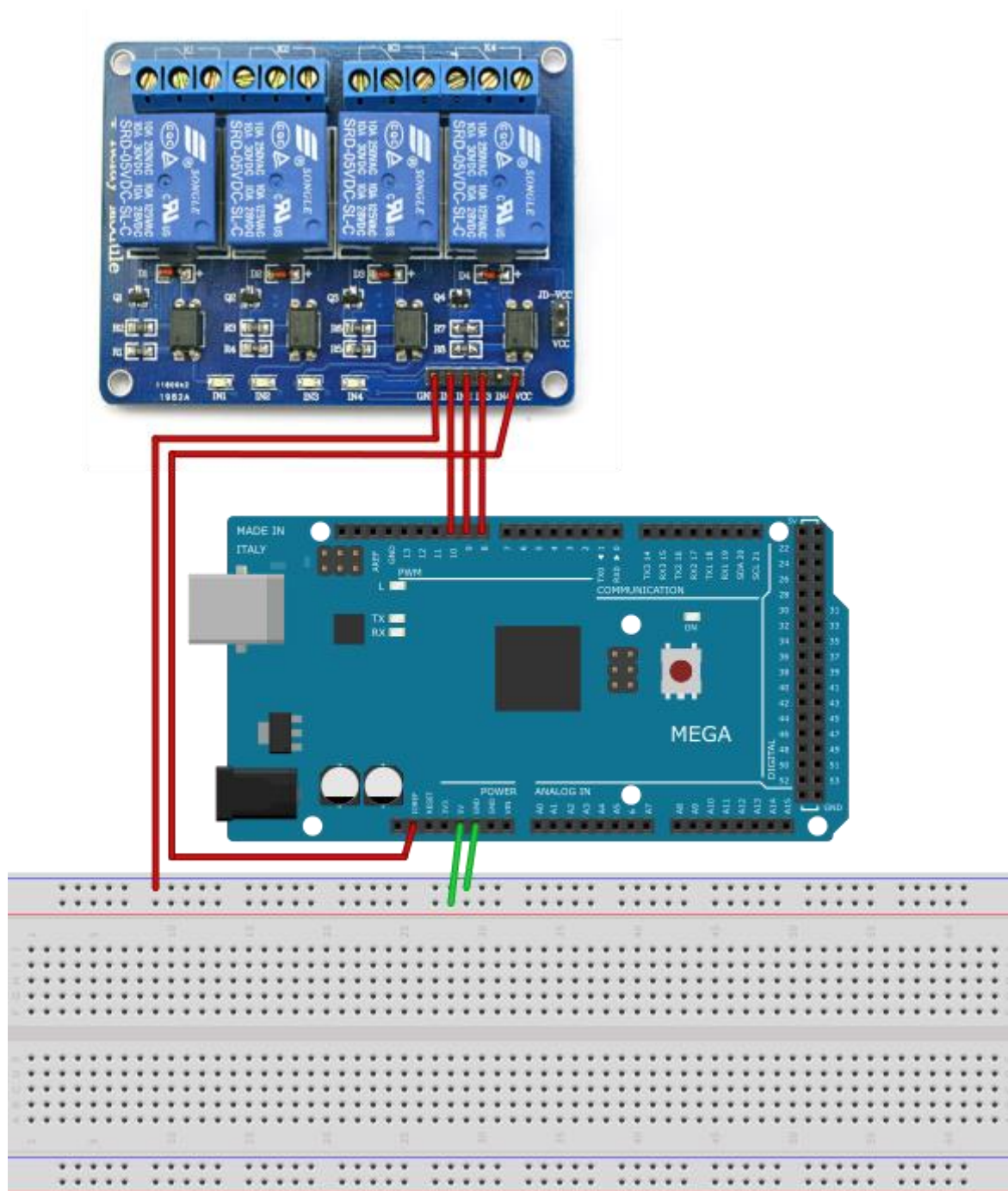
Obr. 29 Připojení RTC modulu

6.6 Modul čtyř relé s galvanickým oddělením

Připojení relé je realizováno za pomoci pěti vývodů. Na základní desce jsou popsány. První vývod je GND, druhý až čtvrtý slouží k připojení jednotlivých relé k mikropočítači a poslední šestý vývod je určen k napájení (viz. obr. 28).

Zapojení:

- 1 GND – na větev GND
- 2 IN1 – do pinu 10
- 3 IN2 – do pinu 9
- 4 IN3 – do pinu 8
- 5 VCC – na větev 5V



Obr. 30 Připojení relé modulu

6.7 Mikrospínače

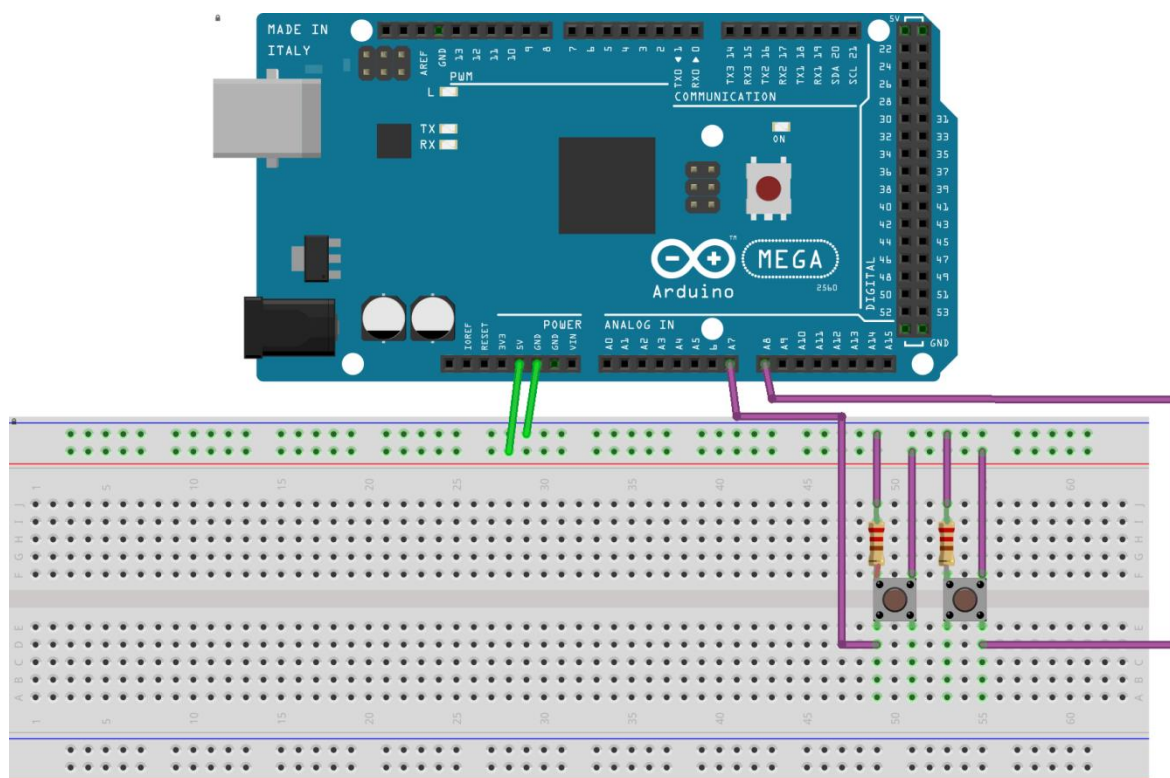
Zapojení obou mikrosčítačů je shodné. Mikrosčítač má vyvedené čtyři výstupy. Jeden slouží k připojení mikropočítače, protilehlý je navíc připojen přes 10k Ohm rezistor na 5V, kdy je při rozpojení kontaktu na pinu logická 1. Třetí vývod slouží k připojení na zem a čtvrtý (druhý konec vývodu) je nepoužit (viz. obr. 30).

Zapojení prvního tlačítka plus:

- 1 VCC – na větev 5V
- 2 – do pinu A7
- 3 GND – na větev GND

Zapojení druhého tlačítka minus:

- 1 VCC – na větev 5V
- 2 – do pinu A8
- 3 GND – na větev GND



Obr. 31 Připojení mikropínačů

7 PROGRAMOVÉ VYBAVENÍ

Program, který obsluhuje ovládací soustavu terária je psán ve vývojovém prostředí Arduino verze 1.0.6. Kód, který je nahrán v mikroprocesoru je znázorněn obrázky a jednotlivé části (řádky) jsou popsány za těmito znaky //, což je syntaxe pro poznámku, která neovlivňuje běh programu.

Z jednotlivých součástek proudí data do mikrokontroleru, který zpracovává vstupy a na základě těchto následné výstupy.[2]

7.1 Nastavení času do RTC modulu

Před nahráním hotového ovládací programu je nutno nastavit aktuální čas do RTC modulu. Program byl přiložen při koupi modulu. Před prvním spuštěním upravíme kód RTC modulu a zapíšeme do něj aktuální čas.

Nastavení času proběhne v části void setup() přepsáním hodnot v závorce setDS3231time(00,17,18,28,5,16). První pozice značí sekundy, druhá minuty, další den v měsíci, měsíc a nakonec rok.

Zkompilujeme program a nahrajeme jej do paměti mikropočítače. Mikroprocesor se postará o zaslání aktuálního času do paměti RTC modulu, která je při výpadku proudu zálohována baterií přímo na plošném spoji modulu.

7.2 Načtení knihoven

Program pro každou součástku využívá specifickou knihovnu, která usnadňuje programování funkcí. Pro ovládání jednotlivých součástek a následné využití v kódu je zapotřebí načtení jejich knihoven hned na začátku programu. Načtou se přes položku Sketch v menu a následně zvolením jednotlivé knihovny v menu Import Library. Základní sada knihoven byla dodána spolu s mikrokontrolerem avšak pokud nějaká knihovna chybí, je potřeba ji stáhnout z webu výrobce dané součástky a importovat pomocí položky Add Library.

```
#include <EEPROM.h> // knihovna pro uložení teploty
#include <Time.h> // knihovna pro čas
#include <RTClib.h> // knihovna pro nastavení času
#include <Wire.h> // knihovna pro sériovou komunikaci
#include <DHT.h> // knihovna senzoru DHT22
#include <LiquidCrystal.h> // knihovna pro displej 1602A 16x2 znaků
```

Obr. 32 Použití knihoven v programu

7.3 Nastavení pinů a definice proměnných

Po načtení knihoven je potřeba nastavit piny. Na začátku souboru definujeme názvy a čísla pinů spolu s proměnnými, se kterými budeme později pracovat v těle programu. (viz. obr. 33)

```
#define PIN_RELAY_LIGHT 8 // relé 1 světelná žárovka
#define PIN_RELAY_THERMOSTAT 9 // relé 2 tepelná žárovka
#define PIN_RELAY_ROCK 10 // relé 3 topný kámen
#define PIN_BUZZER 13 // bzučák
#define pin 12 // pin DHT22
#define DHTTYPE DHT22 // definice senzoru DHT22
byte startHour1 = 8; // 8 hodin - 1 - letní cyklus
byte startMinute1 = 0; // 0 minut
byte endHour1 = 16; // 16 hodin
byte endMinute1 = 0; // 8:00 do 16:00
byte startHour2 = 8; // 8 hodin - 2 - zimní cyklus
byte startMinute2 = 0; // 0 minut
byte endHour2 = 18; // 18 hodin
byte endMinute2 = 0; // od 8:00 do 18:00
int DHT11_RETRY_DELAY=100; // pauza
int rs_pin = 2; // displej 1602A
int e_pin = 3; // displej 1602A
int d4_pin = 7; // displej 1602A
int d5_pin = 6; // displej 1602A
int d6_pin = 5; // displej 1602A
int d7_pin = 4; // displej 1602A
float settemp = 25.0; // variabilní teplota, která se následně přenastavuje tlačítky
int buttonplus= A7; // pin pro tlačítko plus
int buttonminus= A8; // pin pro tlačítko minus
DHT dht(pin, DHTTYPE); // definovaný senzor teploty a vlhkosti
RTC DS1307 RTC; // ovladač DS3231
LiquidCrystal LCD(rs_pin, e_pin, d4_pin, d5_pin, d6_pin, d7_pin); // nastavení pinů LCD
```

Obr. 33 Nastavení pinů a proměnných

7.4 Popis funkce void setup()

V horní části programu nalezneme funkci void setup(), do které zapíšeme kód, který se provede pouze jednou na začátku programu. Stane se tak při připojení napájení, zmáčknutí tlačítka restart nanebo po nahrání kódu do Arduina. Tato část je v programu nutná a i tehdy, pokud neobsahuje žádné příkazy. Při chybějící části void setup() by kompilace skončila chybou.

Prvně se provede nastavení výstupu tepelné a světelné žárovky společně se vstupem obou tlačítek. Dále si program zapíše hodnotu teploty do první buňky EEPROM paměti. V následujícím kroku se spustí displej, nastaví se kurzor na výchozí hodnotu a vypíše se text označující vlhkost a teplotu. V poslední části jsou definovány funkce pro zapnutí a

vypnutí světla společně se zapnutím a vypnutím tepelného kamene. S těmito inicializovanými údaji se nadále pracuje v další části programu void loop().

```
void setup()
{
  Wire.begin();
  dht.begin();
  pinMode(PIN_RELAY_THERMOSTAT,OUTPUT); // tepelná žárovka výstup
  pinMode(PIN_RELAY_LIGHT,OUTPUT); // světelná žárovka výstup
  pinMode(buttonplus,INPUT); // tlačítko plus vstup
  pinMode(buttonminus,INPUT); // tlačítko minus vstup
  EEPROM.write(1,25); // zapiše hodnotu 25 do první buňky EEPROM paměti
  LCD.begin(16,2); // spustí display s 16 znaky a 2 řádky
  LCD.setCursor(0,0); /*nastaví kurzor na první znak prvního řádku
  // (kam se nastaví sám, když se lcd inicializuje), čísluje se od nuly*/
  LCD.print("Vlhkost: "); // napíše text na první řádek
  LCD.setCursor(0,1); // nastaví kurzor na první znak druhého řádku
  LCD.print("Teplota: "); // napíše další text
}
void lightON () {digitalWrite(PIN_RELAY_LIGHT,HIGH); // zapne světlo
}
void lightOFF () {digitalWrite(PIN_RELAY_LIGHT,LOW); // vypne světlo
}
void rockON() {digitalWrite(PIN_RELAY_ROCK,HIGH); // zapne kámen
}
void rockOFF() {digitalWrite(PIN_RELAY_ROCK,LOW); // vypne kámen
}
```

Obr. 34 Funkce void setup()

7.5 Popis funkce void loop()

Funkce void loop() je taktéž další nutnou součástí fungujícího programu v jazyce Wiring. Obsahuje main kód, který se bude opakovat neustále dokola až do odpojení napájení.

Program si v první části načte data z RTC modulu, tedy čas. V celou sekundu program ověří aktuální měsíc a dle tohoto údaje spustí funkce, které byly definovány hned pod knihovnou mezi nastavením pinů. Pokud je jeden z měsíců od března do října (letní čas) spouští se kámen se světelnou žárovkou v 8 hodinovém denním cyklu od 8:00 do 16:00. Pokud není ani jeden z těchto měsíců, pak program skočí do části spuštění zimního cyklu, který volá funkce zajišťující osvětlení a spuštění topného kamene každá den po dobu 10 hodin od 8:00 do 18:00 hodin.(viz. obr. 35)


```

void loop(){
DateTime now = RTC.now(); // přečte čas z RTC
if (now.second() == 0) { // jestli je celá sekunda
    // pokud je 3, 4, 5, 6, 7, 8, 9 nebo 10 měsíc, je letní cyklus, čas 1
    if((now.month() == 3) || (now.month() == 4) || (now.month() == 5) || (now.month() == 6)
    || (now.month() == 7) || (now.month() == 8) || (now.month() == 9) || (now.month() == 10)) {
        if (now.hour() == startHour1 && now.minute() == startMinute1) {
            lightON();
            rockON();
        } else {
            if (now.hour() == endHour1 && now.minute() == endMinute1) {
                lightOFF();
                rockOFF();
            }
        }
    }
}
else { // pokud ne je zimní cyklus, čas 2
    if (now.hour() == startHour2 && now.minute() == startMinute2) {
        lightON();
        rockON();
    } else {
        if (now.hour() == endHour2 && now.minute() == endMinute2) {
            lightOFF();
            rockOFF();
        }
    }
}
}

```

Obr. 35 První část funkce void loop()

V druhé části funkce void loop() si program nejprve načte nastavenou teplotu do paměti EEPROM. Dále probíhá nastavení funkčnosti tlačítek. Pokud jsou připojena k napájení, tak se po jejich stisku přičte nebo odečte jeden stupeň Celsia za doprovodu odlišných zvuků pípáku pro každé tlačítko. Při přenastavení teploty nová hodnota přepíše stávající nastavení v EEPROM a termostat udržuje tuto teplotu.

```

settemp=EEPROM.read(1); // načítá do settemp výrobní teplotu
// načítáme si do proměnných čas
if (digitalRead(buttonplus)== HIGH )// pokud se detekuje 5V pak...
{
    ( settemp +=0.1 ) ;// při stisknutí tlačítka se připočte +1 stupeň k settemp
    tone(PIN_BUZZER, 560, 100); // tón tlačítka plus
    EEPROM.write (1,settemp); // uloží hodnotu do EEPROM
    delay(50);
}
if (digitalRead (buttonminus) == HIGH)// pokud se detekuje 5V pak...
{
    (settemp -=0.1);// při stisknutí tlačítka se odečte -1 stupeň od settemp
    tone(PIN_BUZZER, 420, 100); // tón tlačítka minus
    EEPROM.write (1,settemp); // uloží hodnotu do EEPROM
    delay(50);
}

```

Obr. 36 Druhá část funkce void loop()

Třetí část programu zajišťuje vypsání aktuálních hodnot teploty a vlhkosti na LCD displej. Prvně nastaví kurzor na správný řádek a pak čte hodnoty ze senzoru, které následně zapíše na určené místo LCD displeje.

```
float temperature = dht.readTemperature();
float humidity =dht.readHumidity();
LCD.clear();
LCD.setCursor(0,0);
LCD.print("T:"); // T - Teplota
LCD.print(temperature); // vypiš hodnotu ze senzoru
LCD.print((char)223); // znak stupeň Celsia
LCD.print("C");
LCD.print("H:"); // H - vlhkost
LCD.print(humidity); // vypiš hodnotu ze senzoru
LCD.print("%");
delay(DHT11_RETRY_DELAY); // zpoždění
```

Obr. 37 Třetí část void loop()

Poslední část opakující se funkce void loop() je pokračování v regulaci teploty. Program přečte teplotu ze senzoru a pokud je nižší než nastavená hodnota zapne termostat, který zajistí sepnutí relé číslo 2, na které je napojená tepelná žárovka. Při teplotě vyšší se termostat vypne.

```
if (temperature < settemp) // teplota kdy se má zapnout topení pokud bude teplota nižší než nastavená
    digitalWrite(PIN_RELAY_THERMOSTAT,HIGH) ;// zapne termostat
    else
        digitalWrite(PIN_RELAY_THERMOSTAT,LOW) ;// vypne termostat
if(humidity < 50) { // při poklesu vlhkosti pod 50% vypíše text...
    LCD.setCursor(1,0);
    LCD.print("  Vlhkost  ");
    LCD.setCursor(1,1);
    LCD.print("    pod 50%    ");
    delay(2000); // varování ponechá 2 sekundy
}
```

Obr. 38 Poslední část funkce vodi loop()

8 OVĚŘENÍ FUNKČNOSTI ZAŘÍZENÍ

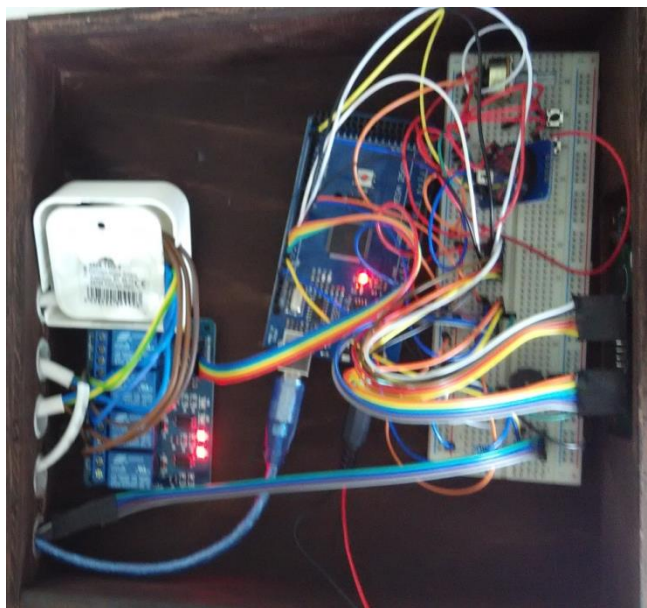
Pro funkční jistotu je třeba zařízení otestovat. Jednotlivé části byly podrobeny zkoušce. Testování nejprve probíhalo mimo terárium a po této zkoušce se zařízení přesunulo na delší testování přímo dovnitř terária.

8.1 Instalace zařízení do terária

Pro ochranu komponentů a celkového řídicího obvodu byla vytvořena dřevěná krabička s výřezy pro LCD displej a elektrické vedení, která bude umístěna nad teráriem. Elektrické vývody procházejí zadní stěnou terária dovnitř. Veškeré kabely jsou dostatečně izolovány, bezpečnost je tedy maximální.



Obr. 39 Zařízení v praxi



Obr. 40 Umístění obvodu v dřevěné krabičce

8.2 Testování funkčnosti periférií

Při stisku tlačítek se ozve zvukový signál z bzučáku, který značí, že tlačítko funguje a bylo stisknuto. Při stlačení tlačítka plus se teplota zvedne o stupeň, což je doprovázeno upozorněním na LCD displeji. Stejná situace nastala při stisku tlačítka mínus, které snižuje teplotu o jeden stupeň. Pokud je teplota nižší než nastavená sepne relé, které ovládá průchod proudu pro tepelnou žárovku. Díky velké výhřevnosti speciální terarijní žárovky trvalo přitápění asi 2 min. Poté se žárovka vypnula.

Díky funkčnosti hodin reálného času se světelná žárovka za pomoci relé spustila v 8:00 ráno a v 16:00 se opět vypla. Současně se světelnou žárovkou hřeje i topný kámen, což ověřuje funkčnost třetí části relé modulu. Toto také dokazuje správné programové nastavení letního období.

8.3 Vyhodnocení

Zařízení splnilo všechny požadavky a je plně funkční.

ZÁVĚR

Jako výstup práce bylo vytvořeno zařízení pro ovládání ideálního životního prostředí užovky červené. Specifika tohoto terarijního živočicha byla nastavena za pomoci ovládacího programu, který je nahrán v mikropočítači Arduino.

Tvorbě předcházelo seznámení s problematikou vývojových platforem a elektrických obvodů. Následně bylo spoustu času věnováno návrhu celého obvodu spolu s volbou součástek a funkcí. Programování mikrokontroleru probíhalo v jazyce Wiring za podpory Arduino IDE. Funkčnost součástek a jejich ovládacích knihoven byla nejprve ověřena přes sériový port programu a následně byly součástky zařazeny do obvodu. Programování probíhalo postupně. Součástka po součástce byla naprogramována a následně proběhlo doladění společné funkčnosti a návaznosti na vstupy a výstupy.

Systém je po otestování plně funkční a vhodný k okamžitému použití. Zařízení je v plánu do budoucna rozšiřovat. Díky spoustě zbylých volných pinů, které na mikropočítači zůstaly neosazené, je možno doplnit další funkce. První předběžný plán, který by zajistil plnou automatizaci terária, obsahuje připojení čerpadla a trysek, které budou automaticky zvlhčovat terarijní prostředí.

Během vypracovávání a praktického modelování této práce jsem se naučila spoustu nových dovedností. Především jsem získala více znalostí z oblasti elektrických obvodů , programování v jazyce Wiring a zručnost při zapojování součástí.

SEZNAM POUŽITÉ LITERATURY

- [1] TRONNER, Pavel. *Mikropočítačová revoluce*. Praha: Dokořán, 2014. ISBN 978-80-7363-617-3.
- [2] BUMBA, Jiří. *Programování mikroprocesorů*. Brno: Computer Press, 2011. ISBN 9788025128381
- [3] MANN, Burkhard. *C pro mikrokontroléry: ANSI-C, kompilátory C, spojovací programy - linkery, práce s ATMEL AVR a MSC-51, příklady programování v jazyce C, nástroje pro programování, tipy a triky ..* Praha: BEN - technická literatura, 2003. μ C & praxe. ISBN 80-7300-077-6.
- [4] PINKER, Jiří. *Mikroprocesory a mikropočítače*. Praha: BEN - technická literatura, 2004. ISBN 80-7300-110-1.
- [5] *Commodore64 historie* [online]. [cit. 2016-05-29]. Dostupné z: http://studium.chytrak.cz/nostalgia/commodorec64_historie.pdf
- [6] KUBÁTOVÁ, Hana. *Struktura a architektura počítačů s řešenými příklady*. V Praze: České vysoké učení technické, 2013. ISBN 978-80-01-05191-7.
- [7] LIČEV, Lačezar. *Architektura počítačů*. Ostrava: VŠB-Technická univerzita, 1999. ISBN 80-7078-632-9.
- [8] *Arduino Mini* [online]. [cit. 2016-05-29]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardMini>
- [9] *Arduino LilyPad* [online]. [cit. 2016-05-29]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardLilyPad>
- [10] *Arduino Uno* [online]. [cit. 2016-05-29]. Dostupné z: <https://www.arduino.cc/en/main/arduinoBoardUno>
- [11] *Arduino Yun* [online]. [cit. 2016-05-29]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardYun>
- [12] *Arduino Mega 2560* [online]. [cit. 2016-05-29]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardMega2560>
- [13] *Arduino Esplora* [online]. [cit. 2016-05-29]. Dostupné z: <https://www.arduino.cc/en/Main/ArduinoBoardEsplora>
- [14] *Arduino Robot* [online]. [cit. 2016-05-29]. Dostupné z: <https://www.arduino.cc/en/Main/Robot>
- [15] *Arduino Galileo* [online]. [cit. 2016-05-29]. Dostupné z: <https://www.arduino.cc/en/ArduinoCertified/IntelGalileo>
- [16] *Buzzer datasheet* [online]. WTW Media, 2016 [cit. 2016-05-29]. Dostupné z: <http://www.electro-tech-online.com/threads/tmb12a05-buzzer-datasheet.120953/>

[17] *DHT22 datasheet* [online]. Disclaimer, 2016 [cit. 2016-05-29]. Dostupné z: <http://www.electroschematics.com/11293/am2302-dht22-datasheet/>

[18] *RTC datasheet* [online]. Maxim Integrated, 2016 [cit. 2016-05-29]. Dostupné z: <https://www.maximintegrated.com/en/products/digital/real-time-clocks/DS3231.html>

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

V	Jednotka elektrického napětí Volt
°C	Jednotka teploty stupeň Celsia
SRAM	Static Random Access (statická paměť)
ROM	Read Only Memory (paměť, jejíž obsah je dán z výroby)
PROM	Programmable Read Only Memory (jednorázově programovatelná paměť)
EPROM	Erasable Programmable Read Only Memory (paměť mazatelná UV zářením)
EEPROM	Electrically Erasable Programmable Read Only Memory (el.mazatelná paměť)
UV	Ultra fialové
VCC	Napájecí pin
GND	Ground (uzemnění)
A	Analogový pin
Ohm	Jednotka odporu Ohm
IN	Input (vstup)
Hz	Jednotka frekvence
RTC	Real Time Clock (hodiny reálného času)
LCD	Liquid Crystal Display (displej z tekutých krystalů)
mm	Jednotka délky milimetr
AM	Ante Meridiem (dopoledne)
PM	Post Meridiem (odpoledne)
PWM	Pulzně šířková modulace
UART	Universal Asynchronous Receiver Transmitter (asynchronní komunikace)
FLASH	Elektricky programovatelná paměť s libovolným přístupem
IDE	Vývojové prostředí
USB	Universal Serial Bus (univerzální sériová sběrnice)
TFT	Thin Film Tranzistor (novodobý druh displeje)

FTDI	Future Technology Devices International (společnost)
USA	Spojené státy americké
HDMI	High Definition Multimedia Interface (přenos obrazu a zvuku)
GPS	Global Positioning System (globální polohovací systém)
ALU	Arithmetic Logic Unit (aritmetickologická jednotka)
Wifi	Wireless Ethernet Compatibilty Aliance (standart bezdrátové komunikace)

SEZNAM OBRÁZKŮ

Obr. 1 Historicky první mikropočítač na světě – Commodore PET.....	12
Obr. 2 Schéma počítače	13
Obr. 3 John Von Neumannova architektura	14
Obr. 4 Mikrokontroler PICAXe 18M2	15
Obr. 5 Značka Arduino	16
Obr. 6 Arduino Mini	17
Obr. 7 Arduino LilyPad	17
Obr. 8 Arduino Uno/Arduino Genuino.....	18
Obr. 9 Arduino Yún	18
Obr. 10 Arduino MEGA 2560/ Arduino Geniuno MEGA 2560	19
Obr. 11 Arduino Esplora.....	19
Obr. 12 Arduino Robot	20
Obr. 13 Arduino Intel Galileo.....	20
Obr. 14 Nepájivé vodivostní pole	24
Obr. 15 DCduino MEGA 2560	26
Obr. 16 LCD display.....	26
Obr. 17 Potenciometr	29
Obr. 18 Bzučák	27
Obr. 19 Senzor teploty a vlhkosti DHT22	27
Obr. 20 Modul reálného času DS3231	28
Obr. 21 Relé modul s optočleny	28
Obr. 22 Mikrospínač	29
Obr. 23 Zapojení všech součástek	30
Obr. 24 Připojení LCD displeje	31
Obr. 25 Připojení potenciometru	32
Obr. 26 Připojení piezo reproduktoru	33
Obr. 27 Připojení senzoru teploty a vlhkosti	34
Obr. 28 Připojení RTC modulu.....	35
Obr. 29 Připojení relé modulu	36
Obr. 30 Připojení mikrospínačů.....	37