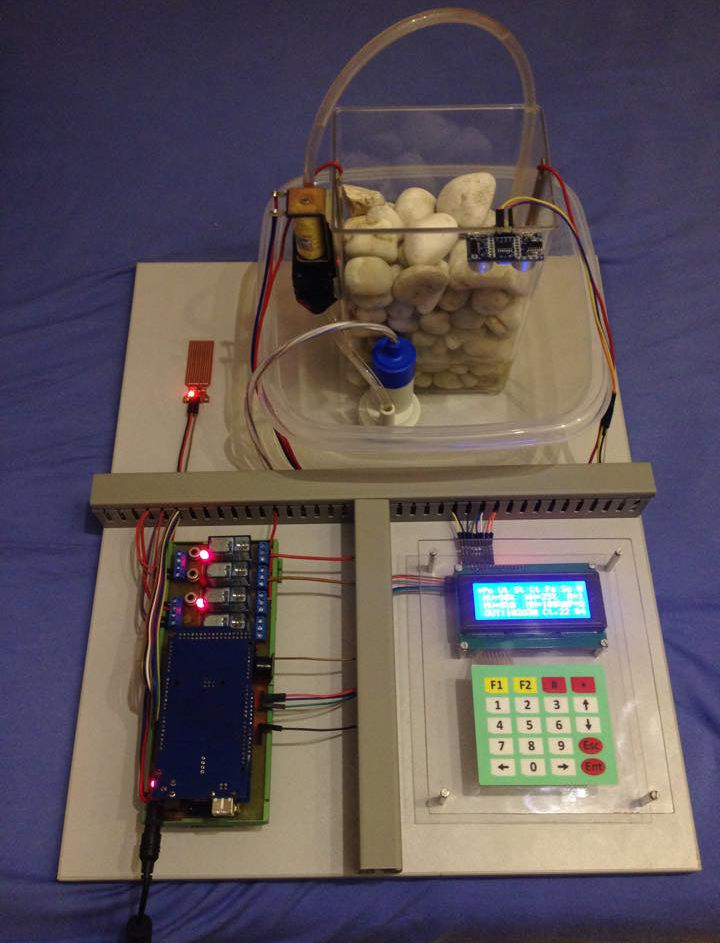
**STŘEDNÍ Průmyslová škola a obchodní akademie Uherský Brod**

**Automatický Zavlažovací Systém**

****

**Kočica Filip ME4**

Uherský Brod 2015

**Prohlášení**

*Prohlašuji, že jsem svou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v seznamu vloženém v práci.*

**Anotace**

Toto téma jsem si vybral hlavně z jednoho důvodu, a to, že jsem hledal nějakou průmyslovou aplikaci s velkým potenciálním využitím, kterou automatické zavlažovací systémy rozhodně jsou. V současné době se stávají již samozřejmou součástí většiny okrasných zahrad a trávníků. Zahradu, jako místo pro zábavu a odpočinek, si bez stále zeleného a svěžího trávníku nelze představit. Stejně tak, jako trávník bez vody. Zavlažovací systém je velkou předností hlavně u domu s velkou zahradou a ušetří spoustu práce i času, takže z mého pohledu je jeho využití v dnešní době velké a mnohdy i nezbytné, například na velkých komerčních prostranstvích (golfová a fotbalová hřiště, parky, …).

Závěrečná práce pojednává o automatických zavlažovacích systémech, jejich účelu, běžném provedení, o porovnání reálného systému s touto prací, zejména pak o řídícím programu, možnosti dálkového přístupu a o fyzické realizaci. Cílem práce bylo zkonstruovat právě takovou řídicí jednotku i se simulací běhu zavlažovacího systému. V práci je popsán návrh i realizace tohoto zařízení od stanovení výchozích požadavků, přes návrh obvodového zapojení a fyzické konstrukce, až po návrh struktury řídicího programu a zprovoznění zařízení. Výsledkem práce je funkční řídicí jednotka pro dvouokruhový zavlažovací systém s možností měření vlhkosti zavlažovaného prostředí, hladiny vody v nádrži a dešťového senzoru jako poruchy. Jednotka poskytuje množství různých funkcí a je možno ji ovládat jak přímo z místa umístění modulu, tak i na dálku přes ethernetovou přípojku.

**Klíčová slova**

Automatický zavlažovací systém, řídicí jednotka Arduino Mega 2560, webové rozhraní, komerční, porovnání, ethernet, ethernetová přípojka, vzdálený přístup, program, LCD, požadavky, návrh, realizace, submit, odeslat.

**OBSAH**

**Strana**

1. [**Úvod**](#uvod) **5**
2. [**Požadované parametry a vlastnosti systému**](#pozadavky) **6**

**3.** [**Automatické zavlažovací systémy**](#autm_zavlaha) **7**

**3.1** Popis automatického zavlažovacího systému 7

**3.2** Účel a uplatnění v průmyslu 7

**3.3** Instalace a obsluha 7

**3.4** Porovnání reálné komerční jednotky s touto prací 8

**4.** [**Ethernetové rozhraní**](#ethernet) **9**

**4.1** Rozložení a popis webové stránky 9

**4.2**. Funkce stránky 10

**4.3** Průběh nastavování 10

**4.4** Popis zpracování dat načtených do vyrovnávací paměti 11

**5.** [**Řídící program**](#řídící_program) **12**

**5.1** Popis, vysvětlení a ukázka všech funkcí řídícího programu 12

**5.2** Základní struktura programu 14

**5.3** Primární části programu 17

**6.** [**Fyzická realizace**](#fyz_real) **19**

**6.1** Blokové schéma 20

**6.2** Plošný spoj 22

**6.3** Osazovací plán22

**7.** [**Závěr**](#zaver) **23**

**8.** [**Soupis použité literatury**](#pouz_lit) **24**

1. **Úvod**

Řídící jednotku je možno spravovat ať už z místa umístění jednotky pomocí LCD 20\*4 s I2C sběrnicí a membránové klávesnice viz [5.1](#řídící_program), nebo vzdáleně pomocí ethernetového rozhraní viz [4](#ethernet), kterým drtivá většina podobných komerčních systémů nedisponuje.

Jednotka je schopna ovládat čerpadlo a elektromagnetický ventil, kterým dokáže čerpat vodu z vodní nádrže do zavlažovaného prostředí v přesně stanovený čas, po přesně stanovenou dobu a den, které se nahrávají do paměti EEPROM nebo ORC. Je možná také simulace výstupů.

Dále také jednotka dokáže kalibrovat elektrody i ultrazvukové čidlo, měřit jak vlhkost pomocí dvou elektrod v zavlažovaném prostředí a hladinu pomocí ultrazvukového čidla, tak i poruchu pomocí tzv. „dešťového senzoru“, který indikuje, že se voda vylila z nádrže a hrozí nebezpečí ať už jen poruchy zařízení, nebo i újmy na zdraví osoby, která by přišla do kontaktu s vodou a elektronikou. Všechny hodnoty se zobrazují na LCD, kde jsou přehledně uspořádány.

V práci je také popsán zavlažovací systém jako takový a jeho využití v průmyslu. [Porovnání reálného komerčního systému s našim vytvořeným systémem](#porovnani) a jeho možnost uplatnění a úspěchu na trhu.

Cílem této závěrečné práce bylo vytvořit program pro elektronickou řídicí jednotku arduino mega 2560, zrealizovat možnost dálkového přístupu alias ethernetové rozhraní a také navrhnout a zrealizovat fyzické řešení plošného spoje, simulaci zavlažování, měření vlhkosti, hladiny a poruchy.

1. **Požadované parametry a vlastnosti systému**

Požadavkem bylo vytvoření simulace plně automatizovaného zavlažovacího systému, který by byl schopen konkurovat obdobným komerčním řešením poskytovaným na trhu, s co nejnižšími náklady na výrobu.

1. Požadavek – Dokázat na řídící jednotce nastavovat – Dny pondělí až neděle, v každém ze sedmi dní dokázat nastavit dobu, kdy se sepnou, po jakou budou sepnuty okruhy 1 a 2, zapnutí/vypnutí funkce současný běh výstupů. Taktéž dokázat nastavit parametry Vlhkost, Hladinu, Hysterezi, Limit Displaye a Režim. Možnost nastavení aktuálního dne v týdnu, hodiny a minuty, podle kterého bude fungovat celý systém, do ORC.
2. Požadavek – Dokázat s maximální přesností snímat pomocí analogových vstupů - vlhkost zavlažovaného prostředí pomocí dvou elektrod umístěných na krajích nádoby, výšku hladiny pomocí ultrazvukového čidla umístěného na hraně nádoby a obě hodnoty dokázat kalibrovat.
3. Požadavek – Vytvořit poruchu a pozastavení poruchy. Porucha by měla při zavodnění tzv. „dešťového senzoru“ začat hlásit poruchu hlasitým pískáním piezo-bzučáku, blikáním displaye a vypnutím výstupů. Při pozastavení poruchy, by měl přestat pískat piezo bzučák, blikání podsvícení displaye a možnost přístupu ke všem funkcím systému, kromě sepnutí výstupů.
4. Požadavek – Při sepnutí relé sepnout pomocí DC 12V/2A čerpadlo a elektromagnetický ventil, které začnou přelévat vodu z nádrže do zavlažovaného prostředí, a pak aby se voda s nějakou časovou prodlevou dokázala výlet zpět do nádrže a tím simulovat vypaření se vody z prostředí a fyzické doplnění vody do nádrže.
5. Požadavek – Možnost vzdáleného přístupu se všemi funkcemi, jaké jsou obsaženy v bodech 1,2 a 3, pomocí jazyků značkovacího HTML a skriptovacího Javascriptu.
6. Požadavek – Navržení a vytvoření plošného spoje (jednoduchého PLC) pro snadnější realizaci a větší přehlednost celé simulace. S možností sepnutí výstupů pomocí relé, indikací pomocí LED, možností donastavení citlivosti vstupů a také možností resetu.
7. Požadavek – Náklady do 1500Kč.

**3.** **Automatické zavlažovací systémy**

V této kapitole naleznete základní informace o tom, co vlastně automatický zavlažovací systém je, jaké je jeho uplatnění v průmyslu a porovnání funkcí komerčního zavlažovacího systému s touto prací.

**3.1 Popis automatického zavlažovacího systému**

Automatický zavlažovací systém bývá zpravidla složen z řídící jednotky, elektroventilů, snímačů, zásobníku vody (zásobárna dešťové vody, studna, vodovodní řád), podzemního trubního vedení, na kterém jsou připojen výsuvné postřikovače, které pokrývají celou plochu zavlažovaného prostředí. Automatický znamená, že ke svému chodu vůbec nepotřebuje lidský zásah, pouze pro počáteční nastavení, popřípadě opravu poruchy. Systém je také možno doplnit o různá čidla podnebních podmínek, aby tak řídicí jednotka mohla přizpůsobit program zavlažování aktuálnímu počasí.

**3.2 Účel a uplatnění v průmyslu**

Automatické zavlažovací systémy se v současné době stávají již samozřejmou součástí ne jen soukromých okrasných zahrad a trávníků, ale i komerčních prostranství jako jsou golfová hřiště, fotbalová hřiště, parky apod.

Za účel a největší výhodu lze považovat ušetření velkého množství času a práce, které by muselo být investováno pro pěkný trávník a zahradu. Další, ačkoli na první pohled ne úplně zřejmou výhodou je velmi kvalitní závlaha a v neposlední řadě je také systém několika postřikovačů estetičtější, než roztahané vodovodní hadice přes celou zahradu.

**3.3 Instalace a obsluha**

V dnešní době stačí kontaktovat příslušnou firmu, která potřebuje vědět pouze, jaký zdroj vody bude systém používat, (zásobárnu dešťové vody, studnu, vodovodní řád) typ čerpadla, rozměry a terén zavlažované plochy. Firma sama udělá návrh, pokud se zákazníkovi bude líbit, pak ho může zrealizovat zcela bez zásahu zákazníka. K dispozici je pak záruční a havarijní servis. Pokud se zákazník rozhodne si zavlažovací systém zapojit zcela sám, výrazně se pak snižuje celková pořizovací cena, ale zvyšuje se obtížnost a při malých zkušenostech i možnost špatného zapojení rozvodného systému či postřikovačů samotných.

Po nainstalování, jak už bylo psáno výše, zákazník nemusí vůbec do systému zasahovat a zalévání probíhá samo podle nastavení, pokud ovšem chce přenastavit řídící jednotku, tak nepotřebuje žádné znalosti a jednoduše dokáže řídící jednotku přenastavit podle přiloženého manuálu. Při větší poruše je třeba provést výkopové práce a tudíž i zavolat firmu, která systém nainstalovala.

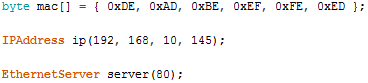
**3.4** **Porovnání reálné komerční jednotky s touto prací**

|  |  |  |
| --- | --- | --- |
|  | **Běžný komerční zavlažovač** | **Tato práce** |
| **Možnost vzdáleného přístupu** |  |  |
| **Indikace probíhající závlahy** |  |  |
| **Nastavení délky závlahy** |  |  |
| **Až 4 denní starty** |  |  |
| **Sedmidenní zavlažovací kalendář s volbou zavlažovacích dní** |  |  |
| **Volba zavlažování sudé/liché dny** |  |  |
| **Manuální spouštění sekcí** |  |  |
| **Manuální blokování programu** |  |  |
| **Manuální zadání přestávky v závlaze 1 – 31 dní** |  |  |

**4.** **Ethernetové rozhraní**

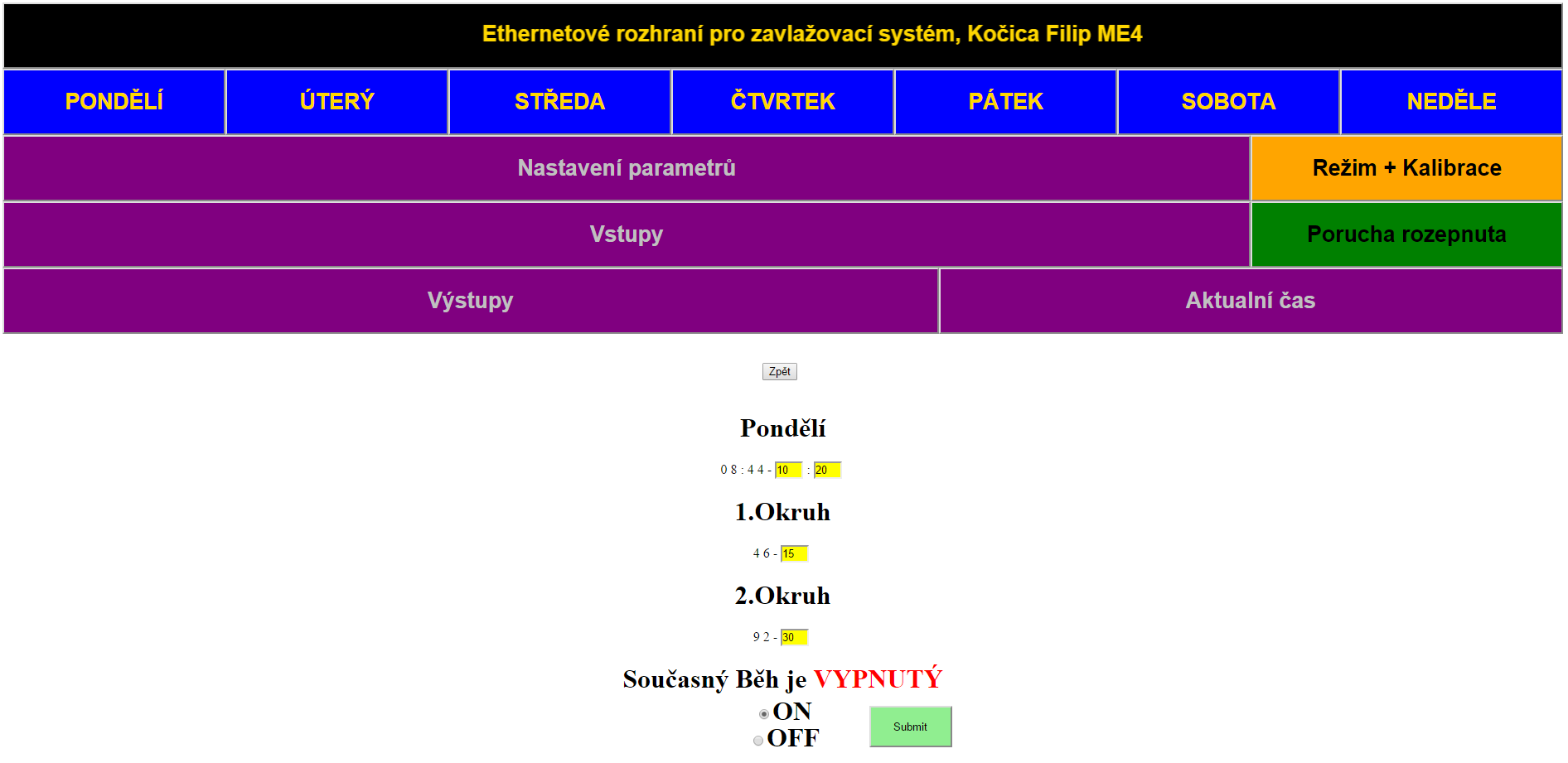
Možnost dálkového přístupu byla vytvořena za účelem předčení levných komerčních systémů nabízených v dnešní době na trhu. U drtivé většiny drahých systémů jakýchkoli druhů, už je dálkový přístup samozřejmostí. Při vytváření rozhraní bylo hlavním cílem zajistit maximální podobnost způsobu výpisu i zápisu dat jako je tomu přímo na jednotce (LCD). V této kapitole je popsáno celé ethernetové rozhraní s názornými ukázkami na webové stránce v prohlížeči.

Na začátku se v programu při vytváření instancí tříd uživatelsky definovaných typů IPAddress a EthernetServer zadává pomocí explicitních konstruktorů volná IPv4 adresa ve vnitřní síti, MAC adresa ethernet shieldu v šestnáctkové soustavě a port 80, bez kterých by komunikace po síti nebyla možná.



**4.1 Rozložení a popis webové stránky**

Celé menu je vytvořeno v hlavičce stránky a vypadá takto: Úplně nahoře jezdí zprava doleva text: „Ethernetové rozhraní pro zavlažovací systém, Kočica Filip ME4“. Jinak je celá stránka navržena tak, aby se co nejvíce podobala způsobu výpisu dat na LCD 20\*4. Takže nahoře jsou dny „Pondělí“ až „Neděle“, pod nimi „Nastavení parametrů“ a „Režim“ + „Kalibrace“, pod nimi „Vstupy“ + „Porucha sepnuta“ / „Porucha rozepnuta“ a pod nimi „Výstupy“ + „Aktuální čas“. Všechny rozkliknuté položky se zobrazují pod tímto menu, viz:

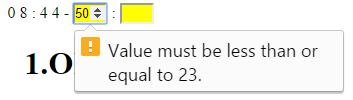
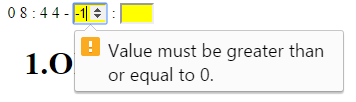


**4.2 Funkce stránky**

Na ethernetu se dají provádět veškerá nastavování, zobrazovat všechna data z paměti řídící jednotky, zobrazovat aktuální hodnoty vstupů stejně tak, jako je tomu na přímo na displayi. Stránka má v hlavičce nastavenou češtinu „UTF-8“ a také obnovení stránky (refresh) jedenkrát za minutu.

**4.3 Průběh nastavování**

Nastavení se realizuje zapsáním čísel do žlutých rámečků, které jsou ošetřeny, aby do nich nebylo možno zapsat menší nebo větší číslo než je u daného prvku možné. V případě zadání takovéto hodnoty se vypíše chybové hlášení:



Celý proces se potvrdí zmáčknutím tlačítka Odeslat. Kód vracený stránkou řídící jednotce vypadá takto: (A provádí se s ním další operace viz [4.4](#popiszpracdat))

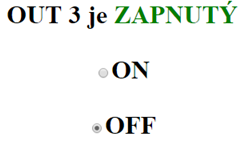


A nahrává se do instance třídy String jménem buffer (vyrovnávací paměť) dokud nenarazí na znak konec řádku ‚\n‘. A pomocí tohoto objektu buffer se provádí veškeré operace a zjišťují se z něj veškeré informace, které jsme zadali na naší stránce. Hned v ten moment se na LCD vypíše, co bylo na ethernetu nastaveno.

Výběr z více možností jako zapnout/vypnout je potom realizován přes vstup typu ‚radio‘.



A vypadá takto:

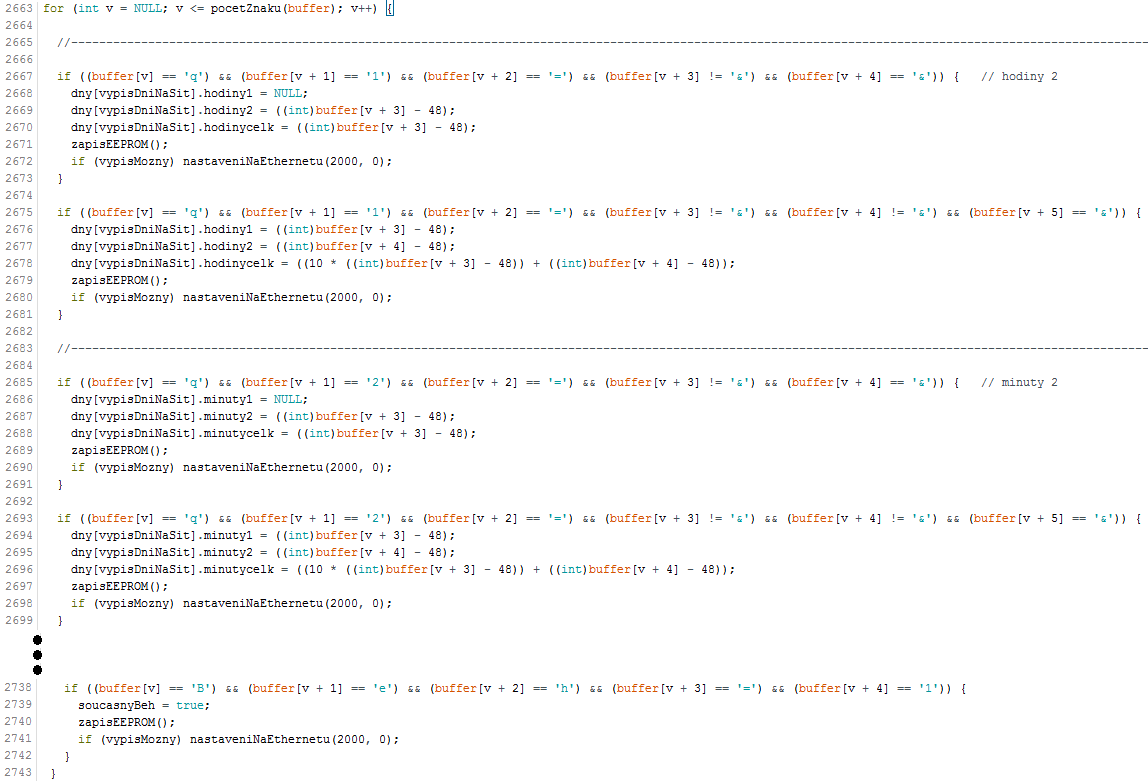


Celý proces se potvrdí tlačítkem submit (odeslat), které odešle kód, viz [kód vrácený stránkou](#kod_vraceny_strankou), řídící jednotce, ta si jej načte do vyrovnávací paměti a provádí s ním další operace, viz níže.

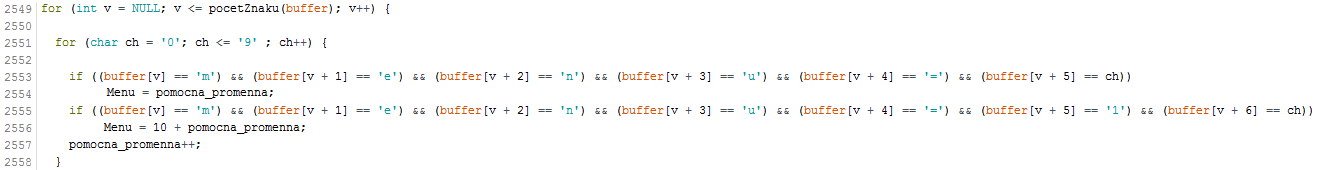


**4.4** **Popis zpracování dat načtených do vyrovnávací paměti**

První se definuje automatická proměnná „v“ typu int jako „NULL“ a cyklus for se provádí do té doby, dokud nedojde na konec řetězce, jehož velikost nám vrací funkce typu int, která vrací celé číslo s počtem znaků ve vyrovnávací paměti. V případě uvedeném níže hledáme řetězec „q1=číslo&“ nebo „q1=čísločíslo&”. Až narazí na řetězec, který hledáme, provede tělo podmínky - pomocí ascii tabulky zjistí číslo/čísla, načte je do příslušné proměnné, provede proceduru zápisu do EEPROM a proceduru, která vypíše na LCD, že byly uloženy data do EEPROM.



Princip zjištění, které z tlačítek v menu na ethernetu bylo zmáčknuto

****

**5.** **Řídící program**

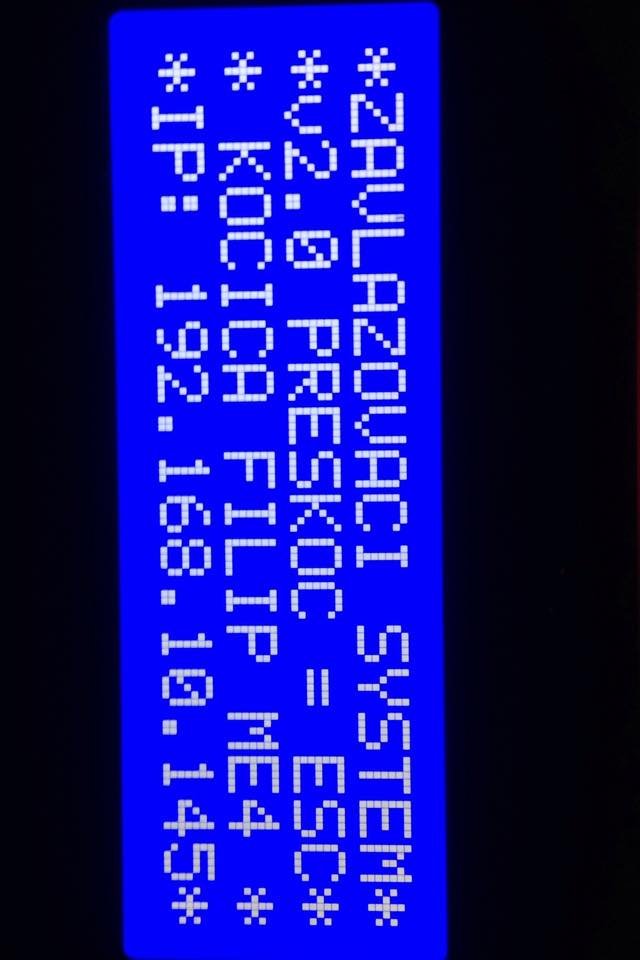
V této kapitole jsou uváděny nejdůležitější úryvky ze zdrojového kódu řídicího programu doplněny o popis a názornou ukázku na LCD. Tyto části kódu byly upraveny (zkráceny) pro lepší názornost ukázek. Program byl vytvořen ve vývojovém prostředí Arduino 1.6.6 a celkem čítá přes 3700 řádků.

**5.1 Popis, vysvětlení a ukázka všech funkcí a možností řídícího programu**

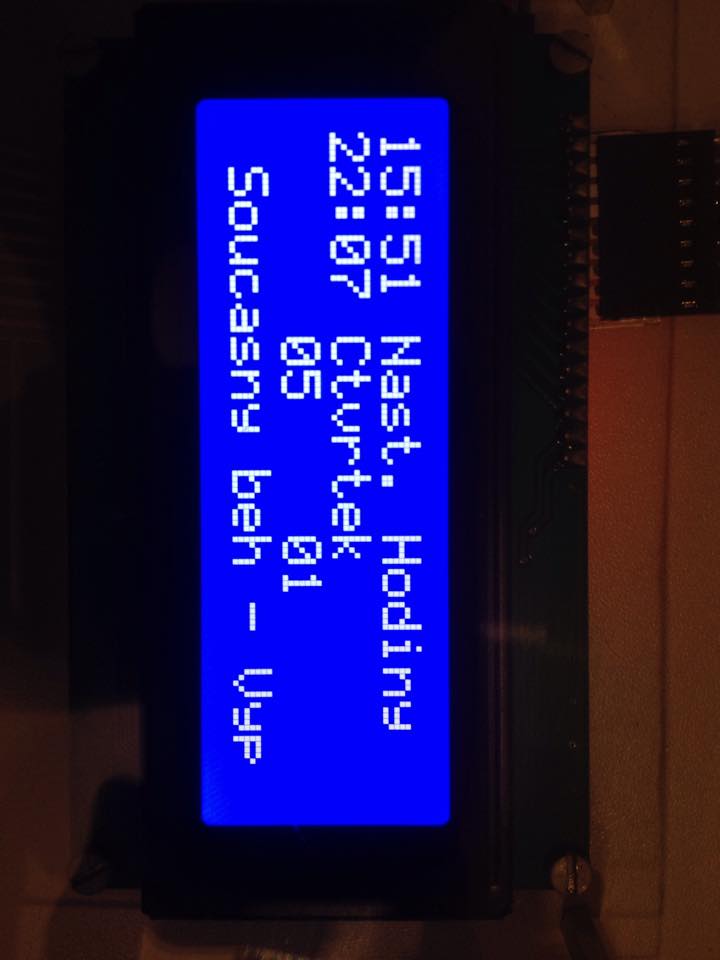
*Ruční (lokální) nastavování pomocí membránové klávesnice a LCD:*

****

Na začátku programu se vypíší na 5 sekund základní informace o systému. Na displayi (20\*4) se lze pohybovat kurzorem ve tvaru šipky pomocí tlačítek nahoru, dolu, doleva a doprava na membránové klávesnici. Výběr a uložení dat se provedete tlačítkem Enter a výstup z funkce tlačítkem Escape, nastavuje se pomocí čísel 0-9. Tlačítka ‚F2‘, ‚#‘ a ‚\*‘ se nevyužívají v žádných částech programu. Při zmáčknutí ‚F1‘ kdekoli v programu se systém zablokuje. Pokud je u kteréhokoli vstupu/výstupu plný čtverec, znamená to, že je sepnutý (log. 1), pokud je u něj prázdný čtverec, znamená to, že je vypnutý (log. 0). Po každém nastavení je nutno uložit data do EEPROM / ORC stisknutím tlačítka Enter!,



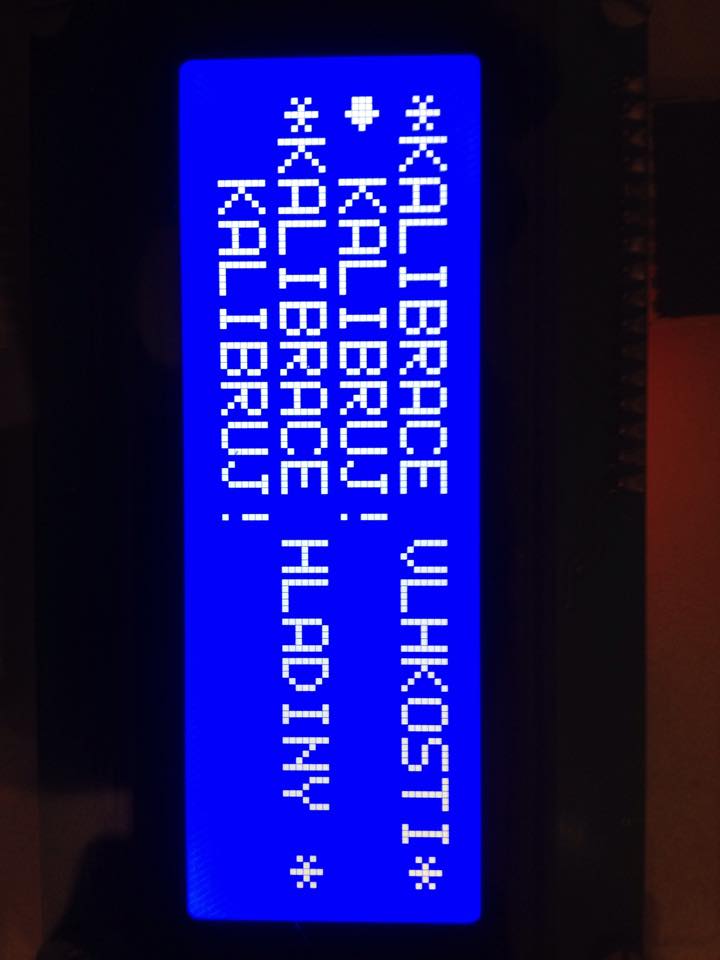
Na prvním řádku LCD se nachází dny Pondělí až Neděle, kde můžete ve funkci „den“ nastavit který den, v jakou hodinu, minutu a na jak dlouho budou sepnuty okruhy (výstupy) 1, 2 a buď zapnout, nebo vypnout funkci současný běh výstupů. Také se tam zobrazuje aktuální čas pro usnadnění nastavování. Při zmáčknutí ‚F1‘ kdekoli v programu se systém zablokuje!



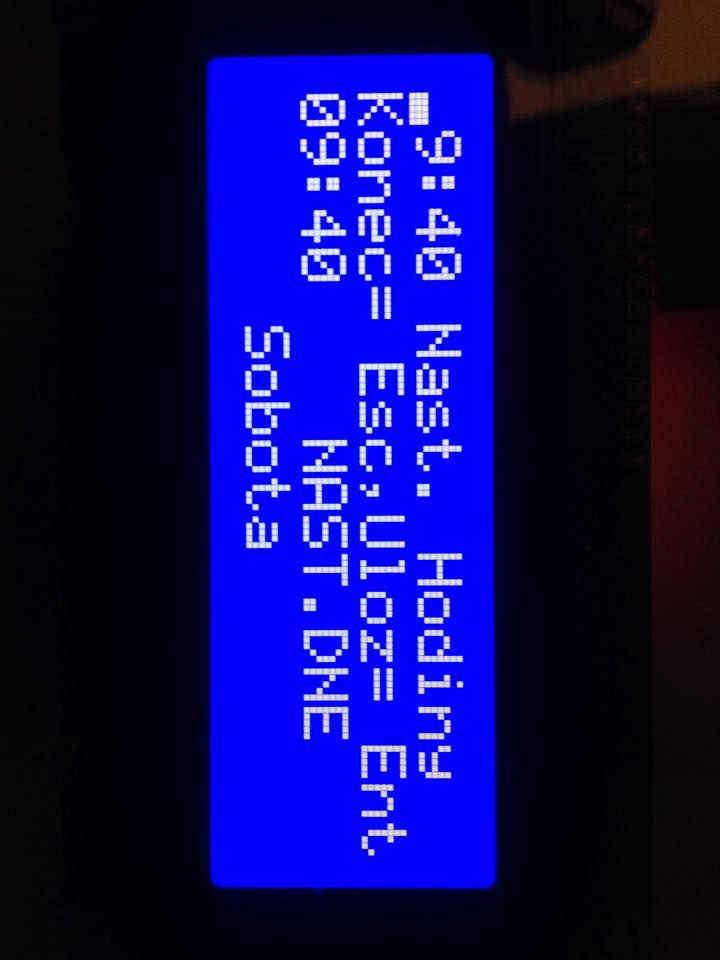
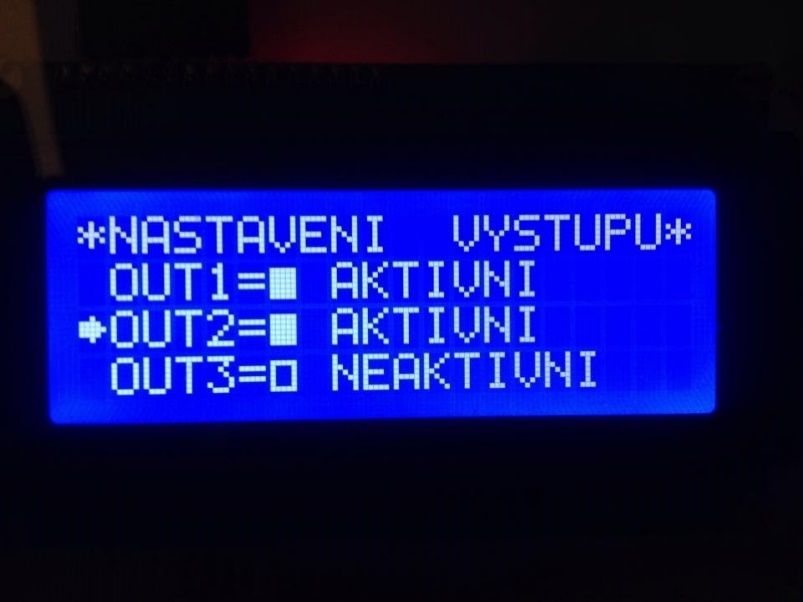
Na druhém řádku nastavujeme vlhkost (0-100%), hladinu (0-100%), hysterezi (0-10%), limit displaye (0-60 Minut) a režim (Normal – Vlhko – Sucho). Nastavení vlhkosti a hladiny bude porovnáváno s hodnotami ze vstupů (při porovnávání je nutno vzít v potaz i hysterezi) a na základě toho se bude program dále chovat, pokud NV>MV (nastavená vlhkost bude větší než vlhkost naměřená na elektrodách) a zároveň NH<MH (nastavená hladina bude menší než hladina snímaná ultrazvukovým senzorem), bude možno výstupy sepnout, v opačném případě pokud NV<MV nebo NH>MH výstupy nebude možno sepnout a pokud už sepnuty byly, neprodleně se rozepnou. Nastavení hystereze je ošetření proti náhlému vypínání a spínaní relé. Limit displaye znamená, po jaké době se vypne podsvícení displaye, opětovné zapnutí podsvícení displaye je možno provést pomocí libovolného tlačítka. Výběr režimu usnadňuje práci se systémem, aby se nemusel přenastavovat celý při náhlé změně počasí, ale stačilo jen přepnout režim (Normal – Vlhko – Sucho).



Na třetím řádku je zobrazena MV (měřená vlhkost), MH (měřená hladina) a P (porucha). Vstupní hodnoty se čtou pouze minutu před sepnutím výstupů, při stlačení tlačítka a při běhu výstupů, kvůli minimálnímu opotřebení elektrod a snímače. Pokud se na analogovém vstupu (A10) objeví více než 25% Ucc, vrátí funkce „prectiPort“ hodnotu 0 a sepne se hlášení poruchy, to se projeví pískáním piezo bzučáku, blikáním podsvícení displaye a nemožností manipulace se systémem. Při poruše nelze sepnout žádný z výstupů. Poruchu lze „pozastavit“ tím, že stlačíte jakékoli tlačítko na klávesnici, vypne se piezo bzučák, blikání podsvícení displaye a bude možný přístup ke všem funkcím programu, nebude ovšem možno v žádném případě sepnout výstupy a na displayi se bude pořád zobrazovat sepnutá porucha. Při výběru na třetím řádku se spustí funkce pro kalibraci vlhkosti a hladiny. Dolní mez je konstantní a kalibruje se pouze horní mez a pak se hodnota přepočítává podle tohoto rozsahu pomocí funkce map (vstup, 0, kalibrace, 0, 100); na rozsah 0 – 100%, který potom vidíme na displayi.

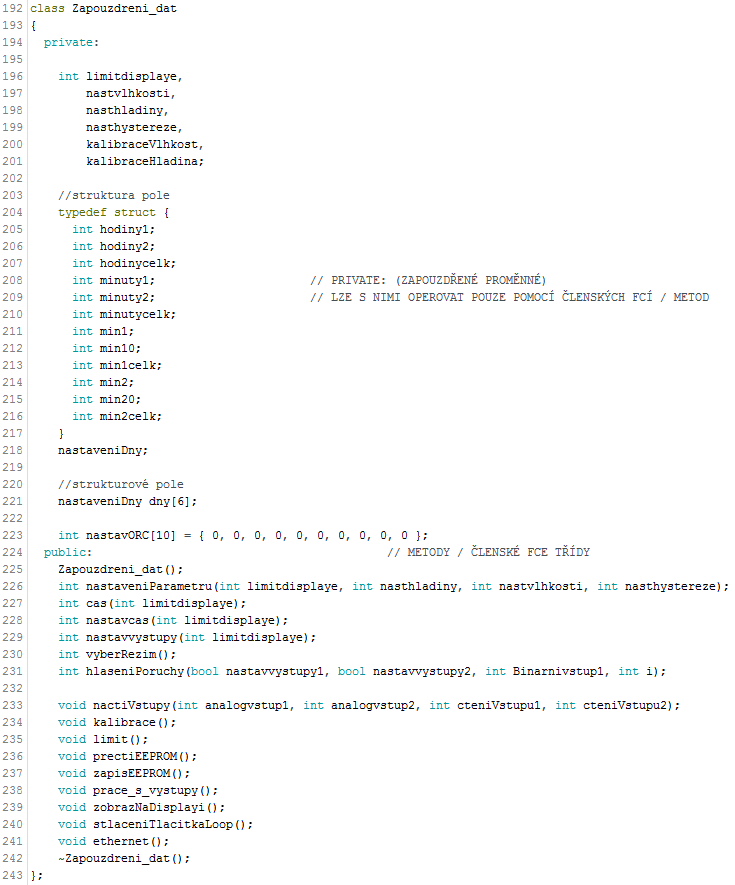


Na čtvrtém řádku jsou zobrazeny výstupy a aktuální den + čas. Výstupy lze simulovat výběrem v menu sepnutím požadovaného výstupu a vrácením se do hlavní nabídky (Ale pozor! Tato simulace není limitována podmínkami nastavených a měřených parametrů a výstupy se sami nikdy nevypnou (pouze při aktivní poruše, nebo blokaci systému)!). Taktéž nastavení obvodu reálného času lze nastavit výběrem v menu. Nastavování probíhá stejně jako u nastavování dnů Po-Ne Po.

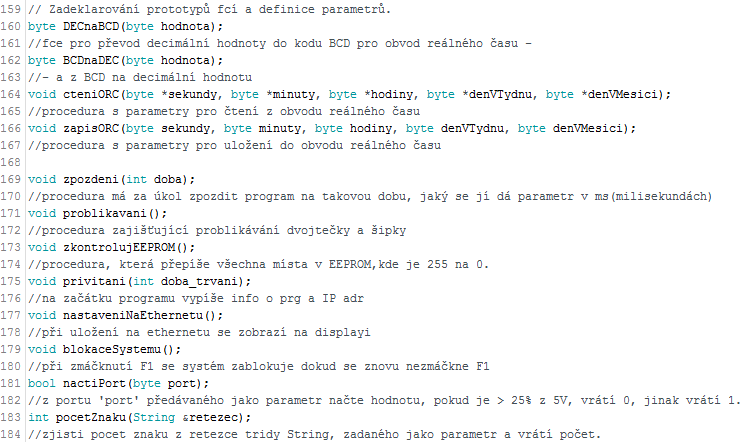


**5.2 Nejdůležitější části programu**

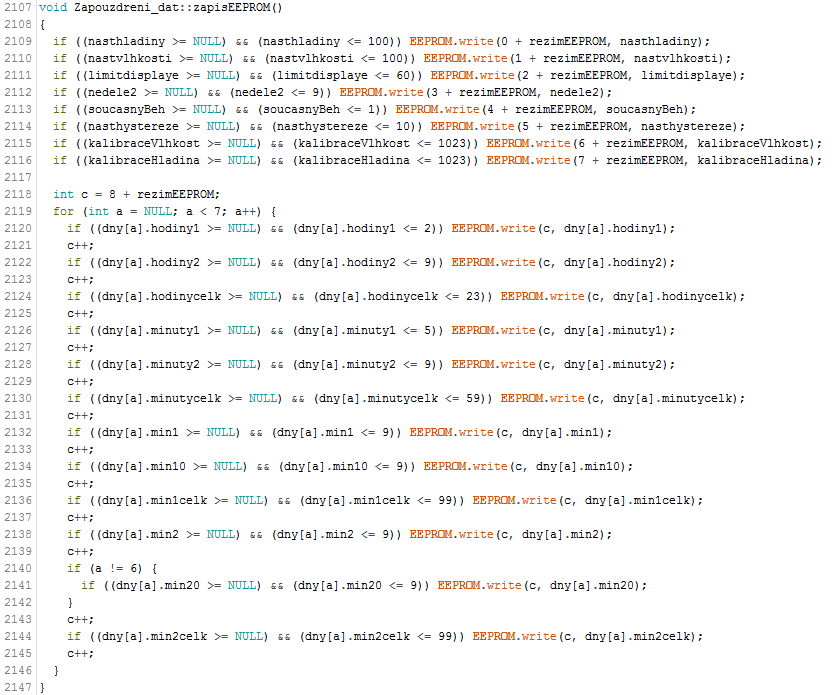
Srdcem programu je třída Zapouzdreni\_dat, kde jsou v privátní části zapouzdřeny důležité proměnné, se kterými lze operovat pouze pomocí členských funkcí třídy/metod. Tato třída není předkem ani potomkem. Obsahuje také pole struktur nastaveniDny, do kterého se ukládají všechny data z nastavení dnů Po-Ne. Ve veřejné části se nachází konstruktor Zapouzdreni\_dat, který vytváří objekty, potom funkce nastaveniParametru, pomocí které nastavujeme všechny čtyři parametry, jako je vlhkost, hladina, hystereze a limit displaye, pomocí funkce cas, se nastavují dny Po-Ne, pomocí nastavcas nastavujeme obvod reálného času, simulace výstupů se provádí funkcí nastavvystupy, vyberRezim umožňuje vybrat režim – Vlhko, Sucho nebo Normal, funkce hlaseniPoruchy se volá při sepnutí poruchy a blokuje celý systém, procedura nactiVstupy(); načte z analogových vstupů data, která podle kalibrace přepočítá na % a vypíše na displayi, kalibrace zajišťuje možnost nakalibrování elektrod i ultrazvukového senzoru, limit způsobuje vypnutí podsvícení displaye po uplynutí limitu displaye, prectiEEPROM a zapisEEPROM zajišťují čtení a zápis do EEPROM paměti, procedura zobrazNaDisplayi zajišťuje výpis všeho, co nakonec uvidíme na LCD, stlaceniTlacitkaLoop určuje, co se stane při zmáčknutí příslušného tlačítka, procedura ethernet zajišťuje síťovou komunikaci a nakonec destruktor objektů ~Zapouzdreni\_dat.



V programu dále nalezneme funkce:



Další neméně důležitou částí je zápis do EEPROM. Zapisování je ošetřeno proti tomu, aby se tam neuložily parazitní informace. Každá proměnná má své místo v paměti.

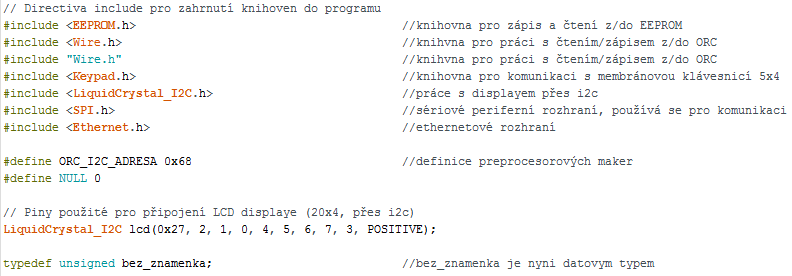


Část procedury pro zápis do EEPROM také ošetřený proti přečtení parazitních informací.

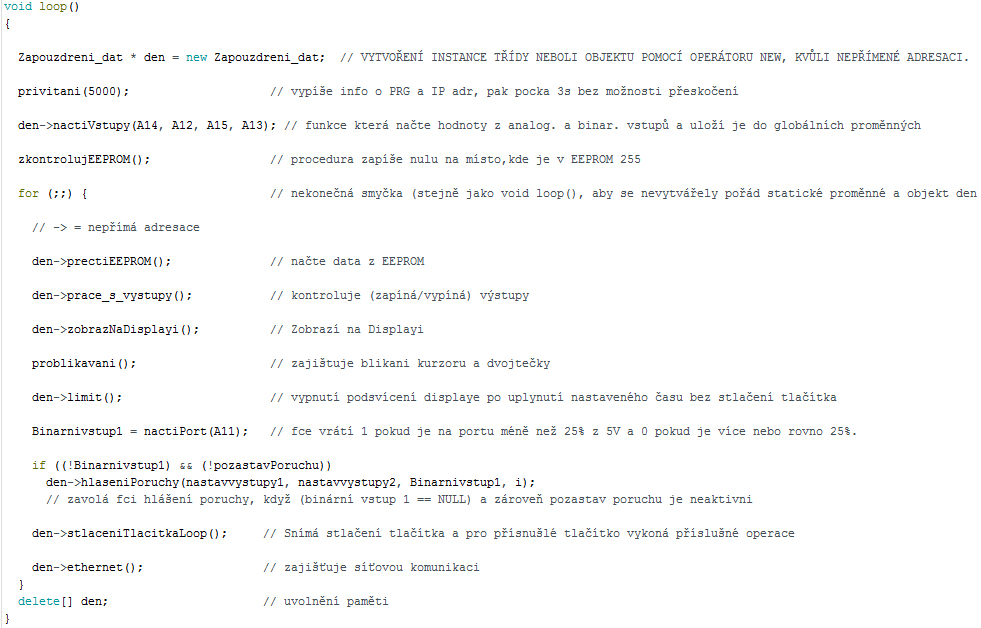
****

**5.3 Základní struktura programu**

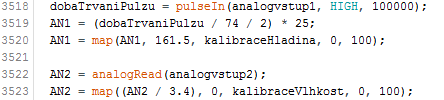
Struktura programu je vytvořena hlavně v závislosti na tom, že řídící jednotka má dvě na sobě nezávislá ovládací rozhraní (LCD + membránová klávesnice a erhernetová přípojka) a že je potřeba provádět naplánované procesy v reálném čase (spouštět výstupy v přesně nastavený čas, měřit všechny analogové vstupy, číst z klávesnice, vypisovat data) nezávisle na stavu ovládacích rozhraní. Proto je program rozdělen na hlavní funkce, které se starají o jednotlivé procesy. Na začátku zahrneme do programu zdrojové soubory pomocí direktivy include:

****

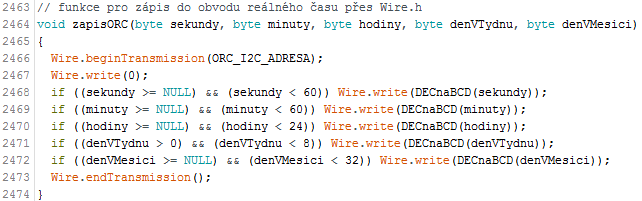
Hned na začátku smyčky loop program vytvoří ukazatel na objekt den pomocí operátoru ‚new‘, provede proceduru privitani, která vypíše na LCD důležité informace o programu, poté zavolá metodu nactiVstupy, která z pinů zadaných jako parametr načte data a uloží do proměnných. Poté program provede proceduru zkontrolujEEPROM, která přepíše všechna místa v paměti, kde jsou ještě z výroby nahrané hodnoty 255 v každé buňce. Nakonec spustí nekonečný cyklus, ve kterém jsou všechny potřebné funkce (popsány yýše).



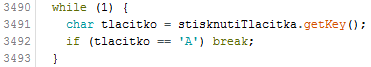
Měření vlhkosti a hladiny: funkce pulseIn vrací dobu, jakou trvalo, než se vyslaný impulz dostal do přijímače. Poté se převede na milimetry a pomocí funkce map rozdělí v rozsahu dolní a horní meze v milimetrech na 0-100%. Obdobně to funguje i u snímání elektrod, tam ale měříme velikost napětí přivedeného na pin analogvstup2.



Procedura pro zápis do ORC. Taktéž ošetřena proti parazitním informacím.

****

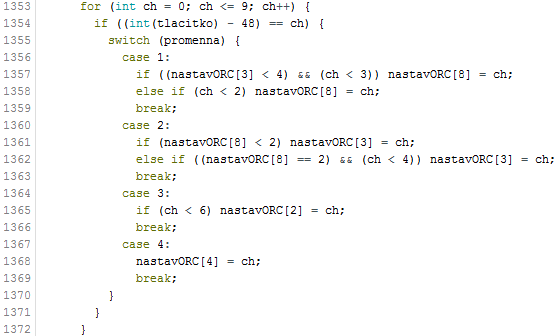
Nekonečný cyklus určený pro blokaci systému, bude se opakovat, dokud nezmáčkneme ‚F1‘

****

Zjištění délky řetězce ve vyrovnávací paměti

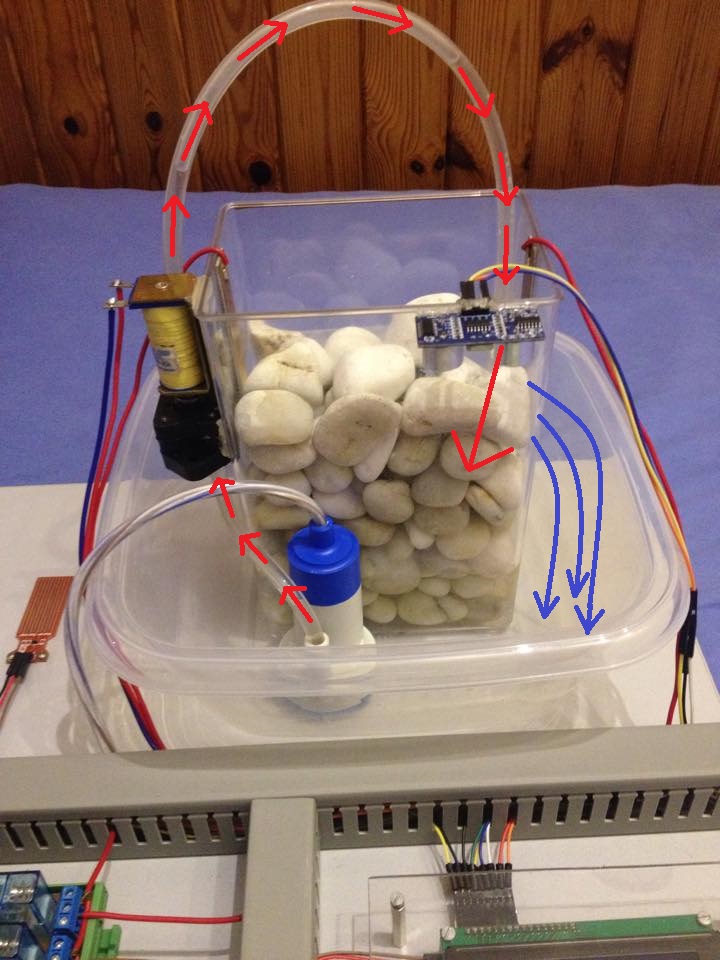


Cyklus, který zajišťuje, že se do proměnných nedají nahrát mylné informace (např. 27 hodin)

****

**6.** **Fyzická realizace**

Realizace simulace zavlažování prošla složitým vývojem. Zezačátku jsme měli dvě nádoby a jedno čerpadlo, kterým jsme proměňovali vodu v obou nádobách, první se načerpala voda do jedné nádoby a z ní jsme ji museli přelít zpět, ale vzhledem k neefektivnosti jsme od tohoto návrhu upustili a vymysleli automatický systém, kdy se voda napuštěná do zavlažovaného prostředí po čase sama vyleje vyvrtanými dírami zpět do zásobníku vody. Červené šipky určují směr vody vytlačované čerpadlem z nádrže do zavlažovaného prostředí, modré signalizují vodu vytékající dírami na straně ven.



**6.1 Blokové schéma**

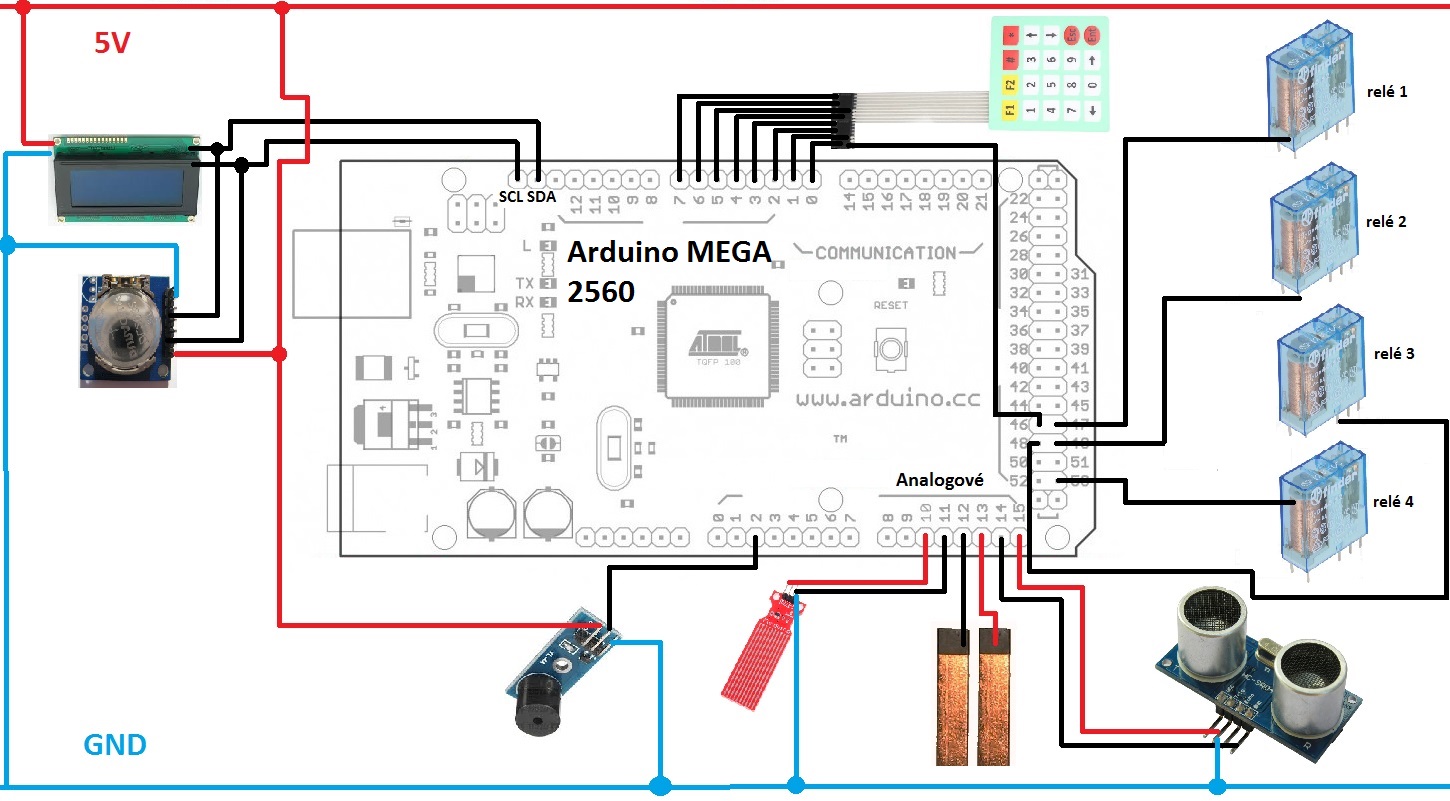
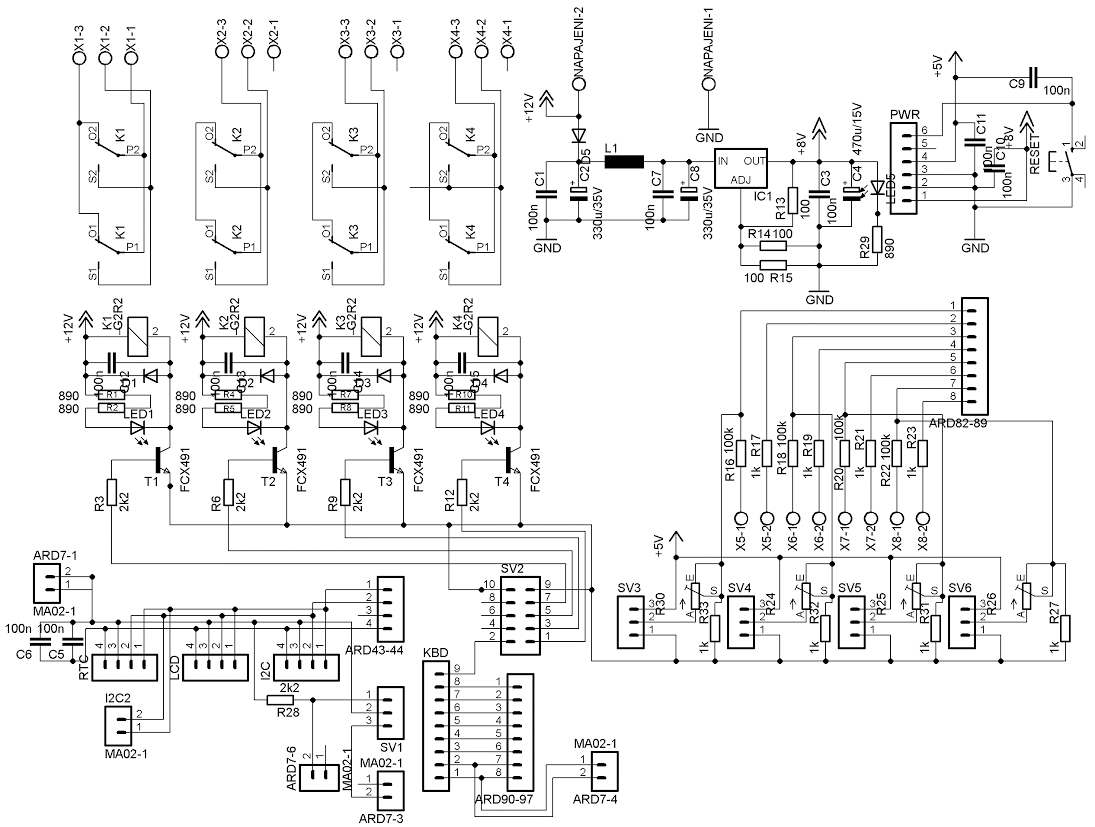
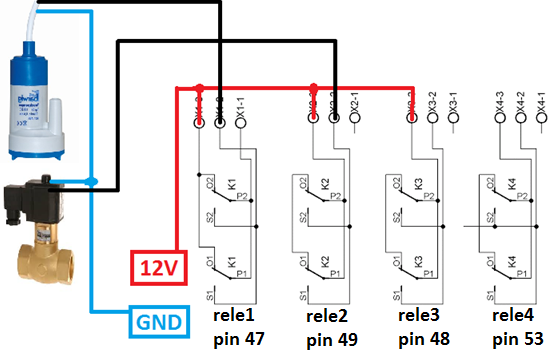


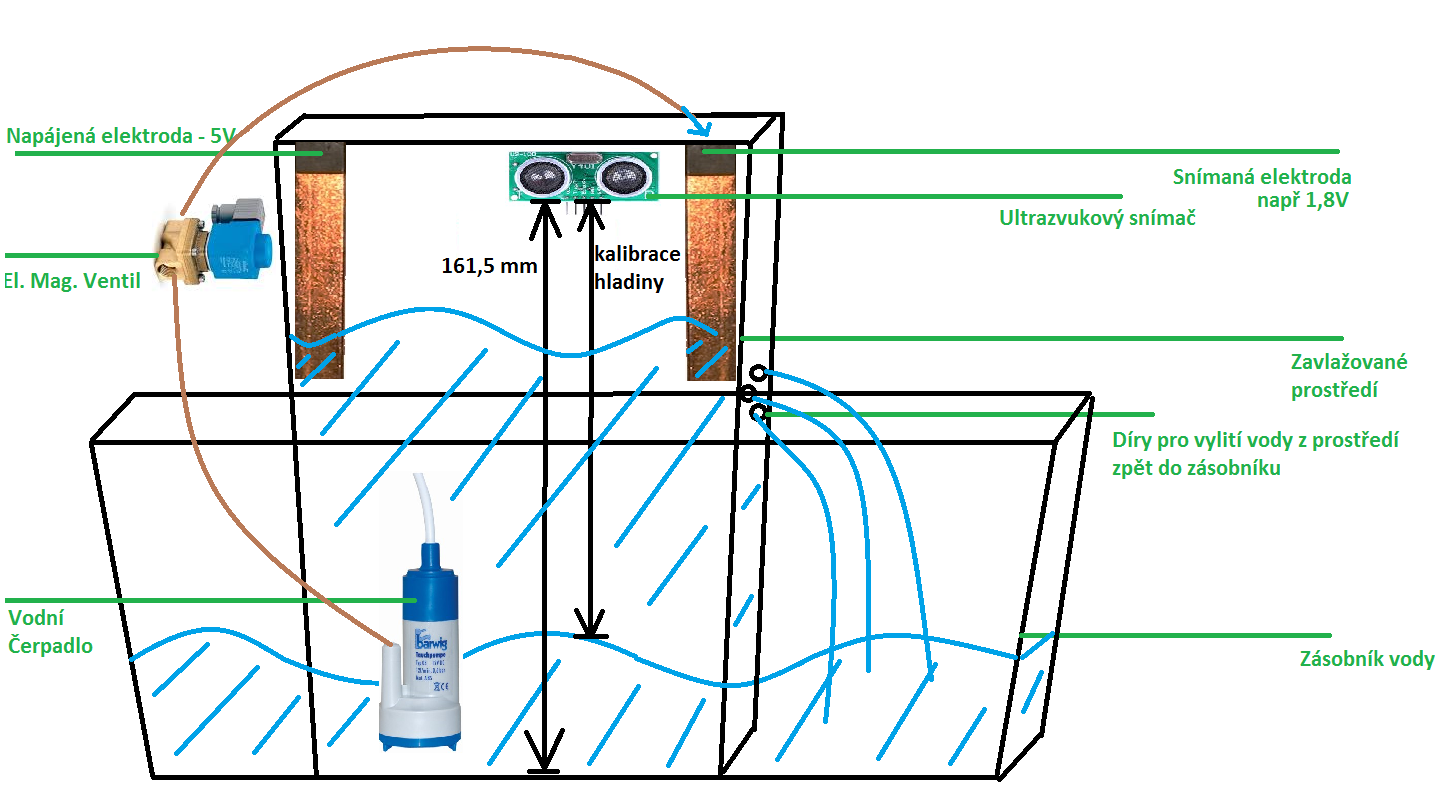
Schéma zapojení:

****

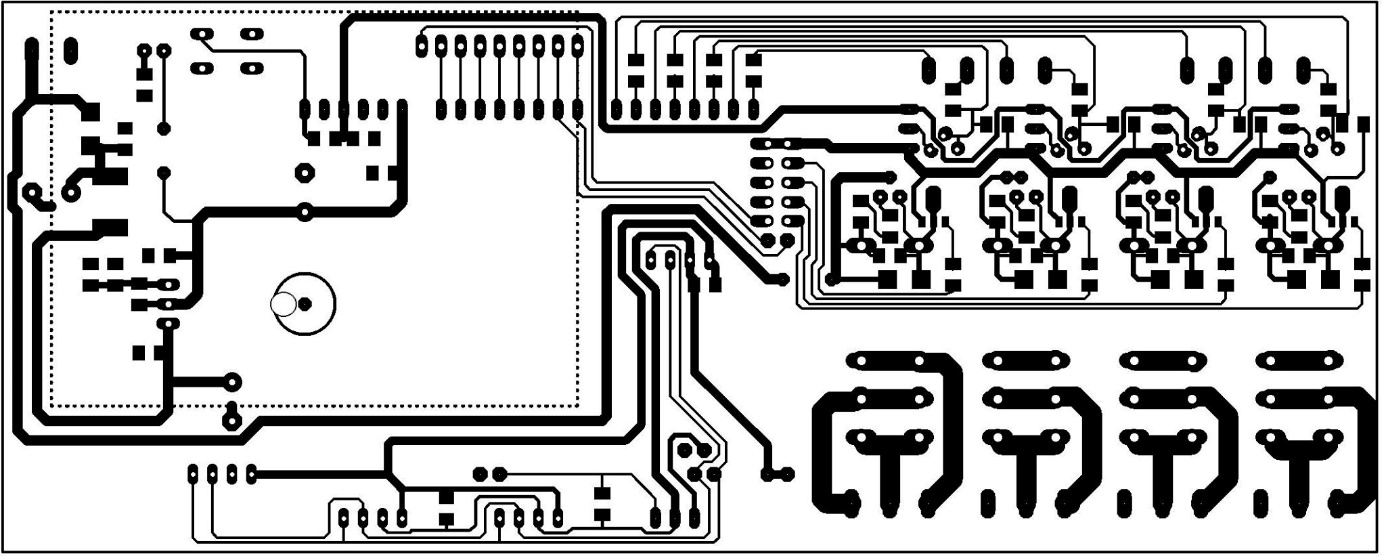
Zapojení elektromagnetických relé – výstupů:



Ukázka běhu simulace:

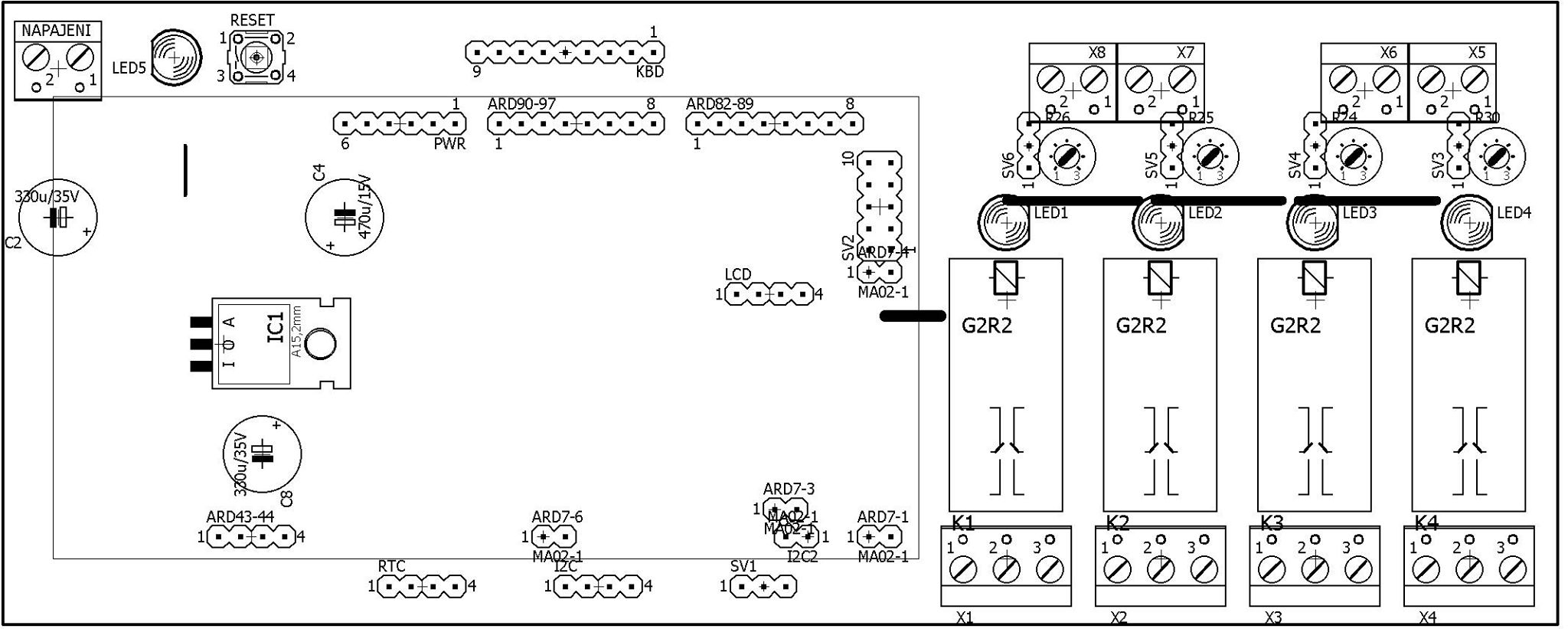
****

**6.2 Plošný spoj**

****

**6.3 Osazovací plán**

Pohled ze strany součástek (THT).

****

1. **Závěr**

Při návrhu i samotné realizaci jednotky byl kladen největší důraz zejména na univerzálnost a na následnou kompatibilitu řídící jednotky (po drobné úpravě by se dal tento systém připojit k už nainstalovanému vodnímu zavlažování). Důraz byl kladen také na dosažení všech sedmi požadavků. Nakonec se ze sedmi, na začátku práce stanovených požadavků, podařilo dosáhnout všech sedmi.

Výsledkem závěrečné práce je plně funkční dvouokruhový řídící systém pro automatické zavlažovací systémy i se simulací a indikací probíhající závlahy. Sepnutím výstupů neboli přivedením napětí na elektromagnetické relé, které spínají vodní čerpadlo (12V/2A) a elektromagnetický ventil, se začne vodou napouštět zavlažované prostředí a při správné kalibraci začne velikost napětí na měřící elektrodě nelineárně narůstat, zatímco se snižující se hladinou vody v zásobníku klesá výše hladiny měřená ultrazvukovým čidlem. Až vyprší čas, po který mají být výstupy sepnuty, měřená vlhkost bude větší než nastavená vlhkost, měřená hladina bude nižší než nastavená hladina, bude aktivní porucha, nebo se zablokuje systém pomocí ‚F1‘, se výstupy (relé) rozepnou a díky dírám vyvrtaným na straně zavlažované nádoby se voda začne vylévat zpět do nádoby reprezentující vodní zásobník, to simuluje vypařování se vody z půdy (poněkud zrychlené) a následným doplněním vody do nádrže simulujícím například déšť doplňující vodní zásobník.

Jedinou věcí, kterou bych na této celé práci vytkl, je nevhodně zvolený ultrazvukový senzor. Nevhodně zvolený je z důvodu pohybu hladiny (voda vytékající ze zavlažované nádoby vyvrtanými dírkami rozvlní hladinu v zásobníku) a následným špatným odrážením vyslaných ultrazvukových impulzů. V případě, že se do senzoru nedostane vyslaný impulz do času nastaveného ve funkci pulseIn(), do proměnné nahraje maximální hodnotu, neboli 100% (po převodu na %), ale většinou měří „pouze“ s odchylkou cca 5-10% a měření tak není úplně přesné. V případě měření ustálené hladiny by ultrazvukový senzor byl bezproblémový, protože by se vyslané impulzy ihned odrazily přímo do přijímače.

Myslím si, že tento systém je konkurence schopný vůči ostatním komerčním cenově dostupným systémům, viz [tabulka porovnání](#porovnani).

1. **Soupis použité literatury**

<http://www.azzahrada.cz/zavlazovaci-systemy/hunter.html>

<http://park.trebicsko.com/43830/zavlahy/>

<https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=41068>