

Česká zemědělská univerzita v Praze
Technická fakulta



Vytvoření virtuální laboratoře s programem Scilab, Scilab – Xcos

Bakalářská práce

Vedoucí práce:

Ing. Miloslav Linda, Ph.D

Autor:

Kedrovskii Nikita

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Nikita Kedrovskii

Technika a technologie v dopravě a spojích
Silniční a městská automobilová doprava

Název práce

Vytvoření virtuální laboratoře s programem Scilab, Scilab – Xcos

Název anglicky

Creation of a virtual laboratory with the program Scilab, Scilab – Xcos

Cíle práce

Cílem práce je vytvoření virtuální laboratoře s programem Scilab, ve spojení např. s platformou Arduino. Tvorba uživatelských rozhraní a automatických ladících skriptů v automatizaci.

Metodika

Prostudování hardwarových a softwarových možností řešení. Navržení několika variant provedení úlohy. Výběr nejvhodnější varianty s kritickým hodnocením návrhu. Realizace modelu a vytvoření obslužné aplikace na platformě Arduino a Scilab. Specifikace funkcí modelu podle cílů práce.

Doporučený rozsah práce

50stran, bez příloh

Klíčová slova

Scilab, Arduino, řízení, snímače, regulace

Doporučené zdroje informací

BELL, C A. *Beginning sensor networks with Arduino and Raspberry Pi*. [New York, New York]: Apress, 2013. ISBN 1430258241.

HERODEK, M. *Android : jednoduše*. Brno: Computer Press, 2014. ISBN 978-80-251-4298-1.

JAVŮREK, J. *Regulace moderních elektrických pohonů*. Praha: Grada, 2003. ISBN 80-247-0507-9.

KAINKA, B. *Měření, řízení a regulace pomocí PC : [vývoj hw a sw pro praxi]*. Praha: BEN – technická literatura, 2005. ISBN 80-7300-089-.

KAINKA, B. *Využití rozhraní PC : měření, řízení a regulace pomocí standardních portů PC*. Ostrava: HEL, 1996. ISBN 80-902059-3-3.

VALTER, J. *Regulace v praxi, aneb, Jak to dělám já*. Praha: BEN – technická literatura, 2010. ISBN 978-80-7300-256-5.

Předběžný termín obhajoby

2020/2021 LS – TF

Vedoucí práce

doc. Ing. Miloslav Linda, Ph.D.

Garantující pracoviště

Katedra elektrotechniky a automatizace

Elektronicky schváleno dne 7. 12. 2020

doc. Ing. Miloslav Linda, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 10. 2. 2021

doc. Ing. Jiří Mašek, Ph.D.

Děkan

V Praze dne 10. 05. 2021

Prohlášení

Prohlašuji, že jsem bakalářskou práci na téma „vytvoření virtuální laboratoře s programem Scilab, Scilab Xcos“ vypracoval samostatně a použil jen prameny, které cituji a uvádím v seznamu použitých zdrojů. Jsem si vědom, že odevzdáním bakalářské práce souhlasím s jejím zveřejněním dle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů, ve znění pozdějších předpisů, a to i bez ohledu na výsledek její obhajoby. Jsem si vědom, že moje bakalářská práce bude uložena v elektronické podobě v univerzitní databázi a bude veřejně přístupná k nahlédnutí. Jsem si vědom že, na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů, ve znění pozdějších předpisů, především ustanovení § 35 odst. 3 tohoto zákona, tj. o užití tohoto díla.

V Praze:.....

Podpis:.....

Poděkování:

Děkuji vedoucímu bakalářské práce Ing. Miloslavovi Lindovi, Ph.D. za vedení a cenné rady při zpracování bakalářské práce.

Abstrakt:

V této práci je popsáno vytvoření automatického systému řízení skleníku pomocí vizuálních programovacích nástrojů Scilab-Xcos a modulu Arduino UNO. Systém se skládá ze tří částí - řídicí jednotky, virtuálního modelu Scilab-Xcos a webové aplikace.

Teoretická část představuje použité komponenty, jejich parametry a vlastnosti.

V praktické části je představen návrh automatizovaného skleníku. Řídicí jednotka je vytvořena pomocí platformy Arduino UNO a požadovaných snímače. Pracovní logika je implementována v programu Scilab-xcos. Grafické rozhraní představuje webová aplikace napsaná v jazyce Java pomocí frameworku SpringBoot.

Klíčová slova: Scilab, Arduino, řízení, snímače, regulace.

Abstract in English:

This work describes the creation of an automatic greenhouse control system using the Scilab-Xcos visual programming tools and the Arduino UNO module. The system consists of three parts - a control unit, a Scilab-Xcos virtual model and a web application.

The theoretical part presents the components used, their parameters and features.

In the practical part, the project of an automated greenhouse is presented. The control unit is implemented by the Arduino platform. The work logic is implemented in the Scilab-xcos program. The graphical interface is represented by a web application written in Java using the SpringBoot framework.

Keywords: Scilab, Arduino, control, sensors, regulation

Obsah	
1 Úvod	1
2 Cíl práce	2
3 Technické prostředky	3
3.1 Scilab	3
3.2 Arduino.....	4
3.3 Webový server	5
4 Návrh systému.....	6
4.1 Dálkové ovládání přes server	6
4.2 Používání funkcionality Scilab Xcos	6
4.3 Arduino.....	7
5 Hardwarový návrh.....	9
5.1 Servomotor	9
5.2 Analogový snímač teploty	10
5.3 Analogový snímač vlhkosti.....	11
5.4 Topení.....	12
5.5 Ultrazvukové zvlhčovače vzduchu	13
5.6 Skleník.....	15
6 Softwarový návrh	16
6.1 Scilab	16
6.2 Webový server	23
6.3 Arduino.....	26
7 Výsledky a diskuse.....	27
8 Závěr.....	28
9 Seznam použitých zdrojů	29
10 Seznam obrázků	30
11 Seznam tabulek	32
12 Seznam příloh	33

1. Úvod

Účelem této práce je analýza a metody používání programu Scilab a platformy Arduino. Interakce softwarových a fyzických komponent je realizována vytvořením skleníku s webovým rozhraním a automatickou údržbou parametrů pomocí Scilab.

Řídící částí v práci je modul Arduino Uno. Softwarová část je založena na interakci modulu Xcos programu Scilab a webové aplikace napsané v jazyce Java. Interakce se serverem probíhá pomocí protokolů HTTP. Data se ukládají lokálně do databáze PostgreSQL.

Následující kapitoly popisují instalaci, její software a interakci s ní. Bude shrnuto prostřednictvím finančního posouzení zařízení a jeho možných budoucích vylepšení.

2. Cíl práce

Cílem práce je:

- a) Vytvoření fyzického modelu skleníku s připojenými senzory.
- b) Vytvoření virtuálního modelu v modulu Scilab Xcos pro automatické řízení nastavených parametrů.
- c) Vytvoření webového rozhraní pro nastavení požadovaných parametrů.
- d) Analýza získaných výsledků a hodnocení budoucí práce.

3. Technické prostředky

Před zahájením práce je třeba se seznámit s použitými zařízeními a softwarem.

Hlavní použitý software je:

- 1) Scilab.
- 2) Arduino.
- 3) Webový server.

3.1 Scilab

Scilab je matematický softwarový balíček, který poskytuje otevřené prostředí pro inženýrské (technické) a vědecké výpočty. Jedná se o nejkompletnější alternativu open source MATLAB. Scilab obsahuje stovky matematických funkcí a je možné přidat nové funkce napsané v různých jazycích (C, C++, Fortran atd.). Existuje také řada datových struktur (seznamy, polynomy, racionální funkce, lineární systémy), tlumočnický a jazyk na vysoké úrovni [10].

Schopnosti:

1. 2D a 3D grafika, animace.
2. Lineární algebra, řídké matice.
3. Polynomiální a racionální funkce.
4. Interpolace, aproximace.
5. Scicos: Hybrid modelování a simulace dynamických systémů.
6. Zpracování signálu.
7. Paralelní práce.
8. Práce s počítačovou algebrou.

Program je k dispozici pro různé operační systémy, včetně Linux, Microsoft Windows a Mac OS X. Schopnosti Scilab lze rozšířit o externí programy a moduly napsané v různých programovacích jazycích. Program je otevřeným zdrojem, který umožňuje bezplatné komerční použití a distribuci nemodifikovaných verzí a nekomerční distribuci upravených verzí, které musí obsahovat zdrojový kód. Pro komerční distribuci upravených verzí je vyžadován souhlas INRIA [10].

3.2 Arduino

Deska Arduino Uno je hlavní deskou produktů Arduino, nejpobulárnějším a nejdostupnějším zařízením. Je založen na čipu ATmega. Nejnovejší revizí Arduino Uno R3 je ATmega328 (i když na trhu stále ještě najdete varianty desky UNO s ATmega168) [11].



Obr 1. Arduino UNO R3

Tab. 1 Specifikace Arduino Uno

Mikrokontrolér	ATmega328
Pracovní napětí	5 V
Napájecí napětí (doporučeno)	7–12 V
Napájecí napětí (limit)	6–20 V
Digitální vstupy/výstupy	14 (z toho 6 lze použít jako PWM výstupy)
Analogové vstupy	6
Maximální proud jednoho výstupu	40 mA
Maximální výstupní proud na výstupu 3,3 V	50 mA
Flash paměť	32 kB (ATmega328), z čehož 0,5 kB používá bootloader
SRAM	2 kB (ATmega328)
EEPROM	1 kB (ATmega328)
Frekvence	16 MHz

Chcete-li zapnout desku, musí být napájena buď z portu USB, přímo z počítače, nebo z externího zdroje napájení – od 7 do 15 voltů. Deska má lineární regulátor, jako je L7805 nebo LDO. Je nutné, aby bylo do mikrokontroléru dodáváno stabilizované napětí 5 V.

Pro programování mikrokontrolérů Arduino se používá zjednodušená verze nízkourovňového jazyka C++. Existuje také mnoho funkcí, tříd, metod a knihoven pro zjednodušení vývoje firmwaru. Díky tomu je práce s těmito mikrokontroléry velmi pohodlná a jednoduchá.

Chcete-li pracovat s knihovny, stačí je nainstalovat pomocí funkce Arduino IDE nebo stáhnout z otevřených zdrojů, například github.com.

3.3 Webový server

Webový server je server, který přijímá požadavky HTTP od klientů, obvykle webových prohlížečů, a vydává na ně odpovědi HTTP, obvykle spolu se stránkou HTML, obrázkem, souborem, mediálním streamem nebo jinými daty [9].

Webový server označuje jak software, který vykonává funkce webového serveru, tak i samotný počítač, na kterém tento software běží.

Klient, kterým je obvykle webový prohlížeč, odesílá na webový server požadavky na prostředky označené adresami URL. Prostředky jsou stránky HTML, obrázky, soubory, mediální streamy nebo jiná data, která klient potřebuje. V reakci na to webový server pošle požadovaná data klientu. Tato výměna probíhá přes protokol HTTP [9].

4. Návrh systému

System se skládá ze skleníku řízeného webovým serverem Arduino Uno R3 s uživatelským rozhraním. Sběr a přenos dat se provádí pomocí Scilab Xcos a odesílá se na webový server. Pomocí webového serveru může uživatel ovládat a měnit parametry skleníku. Například: změna teploty a vlhkosti, ovládání skleníkových klapků atd. Data se ukládají lokálně do databáze.

4.1 Dálkové ovládání přes server

K ovládání a monitorování skleníku se používá webový server, na který se odesílají data ze senzorů a servomotorů. Data jsou čtena pomocí Arduino, přenesena do počítače a odeslána na server pomocí Scilab Xcos.

K odesílání a přijímání dat se používá protokol HTTP. HTTP je protokol pro přenos dat na úrovni aplikace jako hypertextové dokumenty ve formátu HTML, ale nyní se používá k přenosu libovolných dat. Klient, kterým je obvykle webový prohlížeč, odesílá na webový server požadavky na prostředky určené pomocí adres URL. Zdroji jsou stránky HTML, obrázky, soubory, mediální streamy nebo jiná data, která klient potřebuje. V reakci na to webový server odešle požadovaná data klientu. Tato výměna probíhá přes protokol HTTP. Informace v této práci budou přenášeny pomocí metody GET [9].

Webová aplikace, která je zodpovědná za správu prvků skleníku, odesílání a přijímání dat, je napsána v Javě pomocí rámce SpringBoot. Pro uživatelské rozhraní byly použity HTML, JavaScript, CSS. Data se ukládají do místní databáze PostgreSQL. Webový server lze provozovat místně nebo pomocí bezplatného hostování z důvodu hospodárnosti a dostupnosti.

4.2 Používání funkcionality Scilab Xcos

V souvislosti s rozsáhlým vývojem moderních počítačových technologií se významně změnily přístupy k řešení problémů navrhování složitých technických systémů. Jako jeden z nejslibnějších aplikačních balíčků je především třeba zmínit systém matematického modelování Scilab, jehož integrace s balíčkem Xcos otevírá

nové možnosti využití nejmodernějších matematických metod pro řešení problémů dynamického a situačního modelování složitých systémů, procesů, zařízení, počínaje strukturálním znázorněním systému a konče prototypováním systému v reálném čase [10]. Pomocí Xcos můžete vytvářet řídicí systémy, systémy pro zpracování signálu, komunikační systémy, modely libovolných dynamických systémů. Xcos byl vytvořen vývojáři společnosti Scilab a je samostatným balíčkem i platformou pro design založený na modelech.

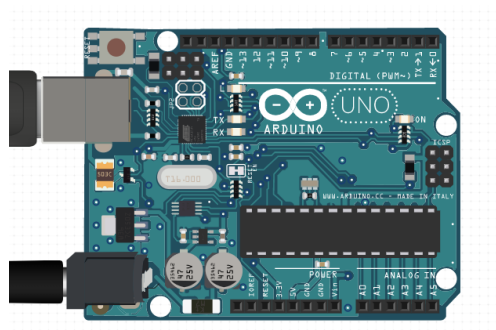
Téměř u všech bloků existuje možnost individuálního nastavení: můžete změnit jak vnitřní parametry bloků (počet vstupů), tak jejich vzhled (velikost). Bloky obsažené ve vytvořeném modelu mohou být navzájem související, a to jak z hlediska informací, tak kontroly. Data se přenášejí přes datová připojení a aktivační signály se přenášejí přes řídicí. Bloky mohou mít také informační a řídicí vstupy a výstupy. Informační vstupy a výstupy bloků jsou zpravidla umístěny nalevo a napravo od obrazu bloku a kontrolní jsou umístěny nad a pod nimi. Typ připojení závisí na bloku a logice modelu. Data vyměňovaná mezi bloky mohou být skaláry, vektory nebo matice libovolných rozměrů [10].

4.3 Arduino

Hlavním prvkem pro čtení parametrů bude Arduino UNO. K ovládání senzorů a motorů je nutné je připojit k ovládacím pinům. Z technických specifikací je známo, že Arduino Uno má 6 analogových pinů a 14 digitálních pinů. V naší práci použijeme několik z nich:

- a. D3 – Relé pro regulaci teploty.
- b. D4 – Relé pro regulaci vlhkosti.
- c. D9 – Servomotor №1.
- d. D10 – Servomotor №2.
- e. A0 – Teplotní snímač TMP36.
- f. A1 – Snímač vlhkosti AMT 1001.

Díky nízkým nákladům na komponenty je stavba skleníku hospodárná a praktická.



Obr. 2 Arduino

5. Hardwarový návrh

5.1 Servomotor

Servo je typ aktuátoru, který dokáže přesně řídit parametry pohybu. Jinými slovy, jedná se o motor, který může otáčet hřídelem v určitém úhlu nebo udržovat nepřetržité otáčení s přesnou periodou. Obvod servopohonu je založen na použití zpětné vazby (smyčka s uzavřeným obvodem, ve které se neshoduje signál na vstupu a výstupu). Jakýkoli mechanický pohon může fungovat jako servopohon, který zahrnuje senzor a řídicí jednotku, která automaticky udržuje všechny nastavené parametry senzoru. Struktura serva se skládá z motoru, snímače polohy a řídicího systému. Hlavním úkolem těchto zařízení je implementace v oblasti servomechanismů. Servopohony se také často používají v takových oblastech, jako je manipulace s materiálem, výroba dopravních zařízení, zpracování dřeva, výroba plechů, výroba stavebních materiálů a další.



Obr. 3 Servomotor

V projektech robotiky Arduino se servo často používá pro nejjednodušší mechanické akce:

- Sledování vzdálenosti otáčením dálkoměru.
- Tvorba kontrolovatelných končetin v robotu.
- Vytváření robotických manipulátorů.
- K implementaci mechanismu řízení.
- Ovládání mechanismu ventilu.

Při práci bude použit motor SG9x2 [7]:

Tab. 2 SG9

Rozsah otáčení	180 °
Napájecí napětí	3 ... 7,2 V
Točivý moment:	0.17 N.m
Rychlost otáčení	60 ° za 0,12 s
Interní rozhraní	Analogové
Směr	CCW (proti směru hodinových ručiček)
Materiál převodovky	Nylon
Materiál pouzdra	Plast
Délka drátu	150 mm
Rozměry	22 × 11.5 × 27 mm
Hmotnost	14 g

5.2 Analogový snímač teploty

Analogový teplotní snímač lze použít v různých podmínkách. Snímač je určen k měření teploty ve stupních Celsia a převodu na napětí. Teplotní snímač je vhodné pro práci v běžných průmyslových oblastech a na otevřených prostranstvích.



Obr. 4 Snímač teploty

Fyzikální princip činnosti polovodičového teploměru je založen na teplotní závislosti poklesu napětí na PN přechodu. K provádění měření musí proudit snímacím prvkem stabilní proud. Výstupním signálem je pokles napětí na senzoru.

Ke zpracování teploty uvnitř skleníku se používá snímač TMP36GT9Z [5]:

- Nízké napájecí napětí 2,7 V – 5,5 V.
- Kalibrováno ve stupních Celsia.
- Faktor měřítka 10 mV/°C.

- Přesnost měření není horší než $\pm 2 \text{ }^\circ\text{C}$.
- Linearita v rozmezí $\pm 0,5 \text{ }^\circ\text{C}$.
- Umožňuje provoz s kapacitní zátěží až 10 nF.
- Parametry jsou garantovány v rozsahu $-40 \text{ }^\circ\text{C} - +125 \text{ }^\circ\text{C}$, maximální teplota je $+150 \text{ }^\circ\text{C}$.
- Odběr proudu není větší než $50 \text{ } \mu\text{A}$, v neaktivním režimu $0,5 \text{ } \mu\text{A}$.
- Nízká úroveň vlastního ohřevu.

5.3 Analogový snímač vlhkosti

Odporové vlhkoměry jsou konstrukcí dvojice elektrod uložených na vodivém polymerním substrátu. Horní část je pokryta vrstvou vodivého materiálu. Oxid hlinitý se ukázal jako velmi citlivý materiál. Když absorbuje vlhkost z atmosféry, jeho odpor se začíná měnit. Jinými slovy, odpor vlhkoměru přímo souvisí s vlhkostí.

Základním principem činnosti je měření změn elektrického odporu v hygroskopickém prostředí. Při příjmu napětí je možné vypočítat hodnotu vlhkosti. Takové senzory jsou účinné a mají nízkou cenu.



Obr. 5 Snímač vlhkosti

Výhody:

- Senzory mají krátkou dobu odezvy na změny vlhkosti, od 11 do 35 sekund.
- Pracují v poměrně širokém teplotním rozsahu, od -40 do $+100 \text{ }^\circ\text{C}$.
- Malé rozměry a nízké náklady.

Nevýhody:

- Provozní doba – 5 let. Překročení provozní doby vede k nepřesným měřením a poškození senzoru.
- Vysoká citlivost na chemické a olejové páry. Chemické výpary mohou zařízení brzy poškodit.
- Nepřesnost měření při práci v kondenzátu.

Použitý senzor: „Odporový senzor vlhkosti AMT1001“.

Tab. 3 ATM1001 [5]

Napájení	5 V
Pracovní teplota	0–60 °C
Rozsah měření	20–95 % RH
Přesnost	± 5% RH

5.4 Topení

K ohřevu skleníku se používá kabelový topný systém. Kabelový topný systém je topný systém, který přeměňuje elektrinu na teplo v důsledku tepelného účinku proudu v topných prvcích vyrobeného ve formě speciálních kabelů. Princip fungování topného kabelu popisuje Joule-Lenzův zákon, který stanovuje, že když elektrický proud protéká jakýmkoliv odporovým prvkem, bude z něj uvolněna tepelná energie. Tento proces je způsoben přítomností elektrického odporu ve vodivém materiálu, který vzniká interakcí nabitých částic. Tyto částice vytvářejí překážku směrového pohybu proudu a při srážce se uvolňuje teplo.



Obr. 6 Topný kabel

Výhody:

- Zvýšená odolnost proti mechanickému namáhání.
- Odolný proti vlhkosti.
- Hospodárnost.

Nevýhody:

- Možnost přehřátí kabelu a požáru.
- Nepřípustnost překrytí kabelu.

Použitý kabel: „Topný kabel 133 Ω 10 metrů 2 mm silikon“.

5.5 Ultrazvukový zvlhčovač

Hlavním úkolem zvlhčovače je zvýšit stupeň vlhkosti v objektu. Toho je dosaženo odpařováním vody a dispergováním částic páry ve vzduchu. Tradiční zvlhčovače odpařují vodu zvýšením teploty. Ultrazvuk k tomu využívá vysokofrekvenční vibrace, která rozptyluje vodní kapky a vytváří mlhu.

Typy zvlhčovačů vzduchu:

- Tradiční.
- Parní.
- Ultrazvukové.
- Vysokotlaké trysky.

Pro nejlepší produktivitu a výkon byl vybrán ultrazvukový zvlhčovač.



Obr. 7 Zvlhčovač vzduchu

Ultrazvukové zvlhčovače jsou považovány za neúčinnější zvlhčovače, které jsou k dispozici. Takové zvlhčovače vytváří mlhu srážením drobných kapiček vody z vodní hladiny pomocí ultrazvukových vibrací přijímaných piezoelektrickým emitorem. Mlha je přenášena objemem místnosti přirozenými proudy vzduchu nebo silou (například pomocí ventilátorů). Po nějaké době (v závislosti na relativní vlhkosti vzduchu v místnosti) se částice mlhy v procesu přirozeného odpařování změni na páru, což vede ke zvýšení relativní vlhkosti vzduchu.

Výhody:

- Schopnost zvyšovat vlhkost na vysoké hodnoty.
- Výstupní teplota páry nejvýše 20 °C.
- Nízká hladina hluku.
- Jsou možné vestavěné ohřívače vody.

Nevýhody:

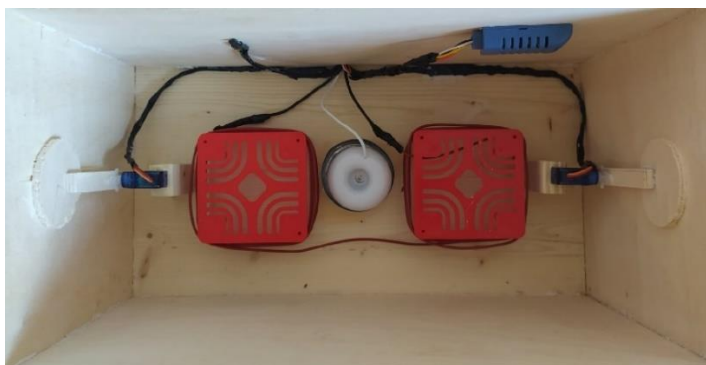
- Zvýšené požadavky na kvalitu a čistotu vody.
- Mlha ze zvlhčovače může obsahovat všechny složky kapaliny v nádrži, včetně solí tvrdosti, mikroorganismů a jejich spor atd.

5.6 Skleník

Model představuje skleník vyrobený ze dřeva o rozměrech 35 × 20 × 15 cm. Na protilehlých stěnách skleníku jsou dva otvory o průměru 5 cm, uzavřené klapkami. Elektronická část systému a napájecí zdroj jsou umístěny venku v samostatné krabici. Z bezpečnostních důvodů, a aby nedošlo k deformaci, byl vnitřek skleníku pokryt vodoodpudivým nátěrem. Skleník je připojen k počítači pomocí kabelu USB.



Obr. 8 Skleník



Obr. 9 Zařízení skleníku

6. Softwarový návrh


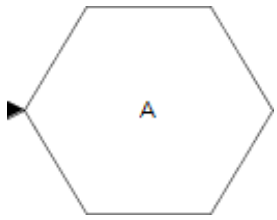

Jak bylo zmíněno v kapitole 2, základem pro správu bude webový server, Scilab a Arduino.

6.1 Scilab



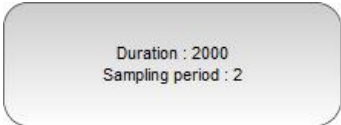


Pro správné ovládání systému pomocí Scilab byla nainstalována knihovna Arduino výběrem v „nástroje“ – „Správa modulů Atoms“ – „Arduino“ – „Instalace“ viz Příloha 3 . K sestavení programu se používá modul Xcos.

Použité bloky [10]:

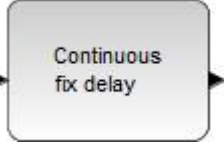
Tab. 4 Směrování signálu:

<p>FROM – Hlavní rolí bloků FROM je přenášet signály z bloku do jiného bloku bez jejich fyzického připojení. Blok FROM přenáší přijatá data (z odpovídajícího GOTO) na svůj výstup.</p>	 <p>Obr. 10 From</p>
<p>GOTO – Hlavní rolí bloků GOTO je přenášet signály z bloku do jiného bloku bez jejich fyzického připojení. Blok GOTO přenáší svá vstupní data do odpovídajícího bloku FROM.</p>	 <p>Obr. 11 Goto</p>
<p>SELECT_m - Blok zodpovědný za řízení změny teploty v závislosti na přijatém signálu z bloku IFTHEL_f.</p>	 <p>Obr. 12 Select_m</p>


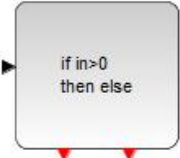
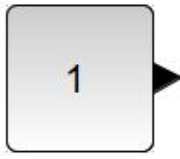
Tab. 5 Arduino blok:

<p>ARDUINO_SETUP – Blok pro připojení desky Arduino a programu scilab.</p>	 <p>Obr. 13 Arduino_setup</p>
<p>ANALOG_READ - Jednotka pro odečítání analogového signálu z pinu Arduino.</p>	 <p>Obr. 14 Analog_read</p>
<p>TIME_SAMPLE – Blok odpovědný za trvání modelu Scilab.</p>	 <p>Obr. 15 Time_sample</p>
<p>SERVO_WRITE - Blok zodpovědný za změnu úhlu otáčení servomotoru</p>	 <p>Obr. 16 Servo_write</p>
<p>DIGITAL_WRITE - Blok mdesílá digitální signál na arduino.</p>	 <p>Obr. 17 Digital_write</p>


Tab. 6 Systémy s nepřetržitým časem:

<p>TIME_DELAY - Blok TIME_Delay zpožďuje vstup o zadanou dobu. Lze jej použít k simulaci časového zpoždění.</p>	 <p>Obr. 18 Time_delay</p>
---	---


Tab. 7 Zpracování událostí:

<p>CLOCK_c – Jediněčný výstup tohoto bloku generuje pravidelný sled událostí, které jsou naplánovány parametrem Period v sekundách.</p>	 <p>Obr. 19 Clock_c</p>
<p>IFTHEL_f – V závislosti na splnění podmínek vstupního signálu bude generována akce na prvním nebo druhém výstupu.</p>	 <p>Obr. 20 Ifthel_f</p>
<p>CONST_m - Tento blok je generátor konstantní hodnoty.</p>	 <p>Obr. 21 Const_m</p>

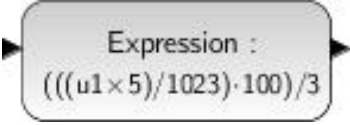

Tab. 8 Matematické operace:

<p>BIGSOM_f – Tento blok provádí sčítání nebo soustružení na svých skalárních nebo vektorových vstupech.</p>	 <p>Obr. 22 Bigsom_f</p>
--	---

Tab. 9 Registrační zařízení:

<p>CMSCOPE – Blok CMSCOPE zobrazuje svůj vstup s ohledem na čas simulace. Blok CMSCOPE může mít více os (jednu na port).</p>	 <p>Obr. 23 Cmscope</p>
--	--

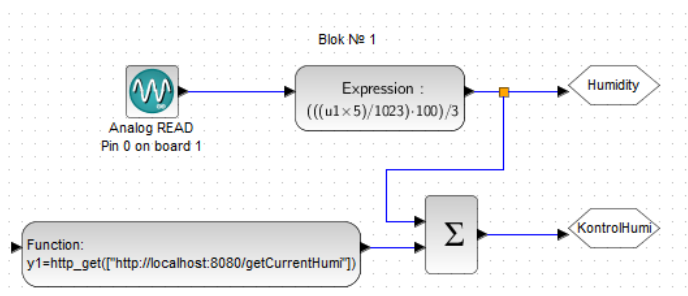
Tab. 10 Vlastní funkce:

<p>EXPRESSION - Blok výrazů použije na svůj vstup zadané výrazy Scilab.</p>	 <p>Obr. 24 Expression</p>
<p>SCIFUNC_BLOCK_M - Blok Xcos Scufunc_block_m nebo funkční blok Scilab, blok, který umožňuje uživateli vytvořit svůj vlastní blok definovaný Xcos pomocí jeho vlastní funkce Scilab.</p>	 <p>Obr. 25 scifunc_block_m</p>

Model Xcos se skládá ze sedmi hlavních částí:

1) První blok(Obr. 11):

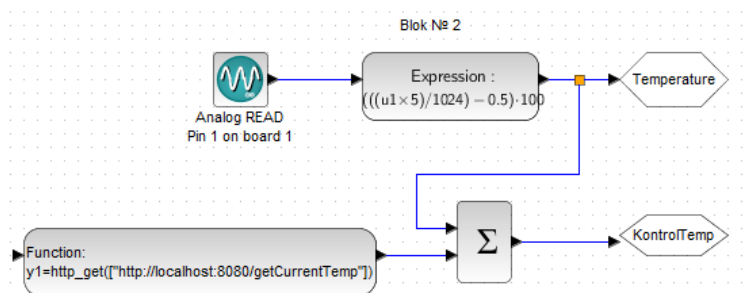
- Příjem dat ze snímače vlhkosti.
- Získání nastavené uživatelem hodnoty vlhkosti ze serveru pomocí GET dotazu „http_get(['http://localhost:8080/getCurrentHum'])*(-1)“.



Obr. 26 Scilab-blok 1

2) Druhý blok (Obr. 12):

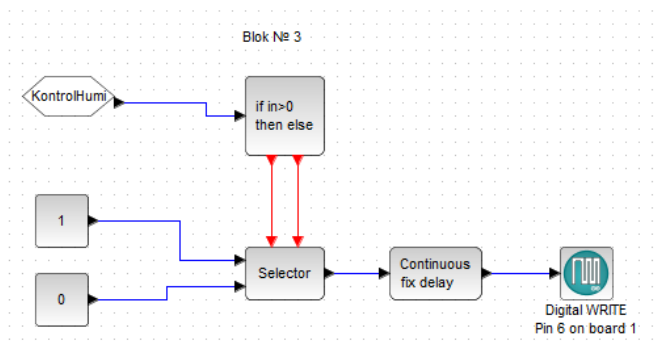
- Příjem dat ze snímače teploty.
- Získání uživatelem nastavené hodnoty teploty ze serveru pomocí GET dotazu
„`http_get([,http://localhost:8080/getCurrentTemp‘])*(-1)“`“.



Obr. 27 Scilab-blok 2

3) Třetí blok (Obr. 13):

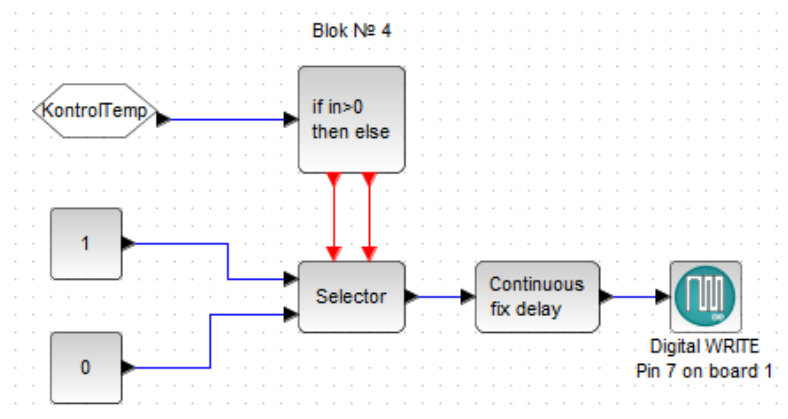
- Udržování vlhkosti pomocí relé. 1 – zapnout, 2 – vypnout.



Obr. 28 Scilab-blok 3

4) Čtvrtý blok (Obr. 14):

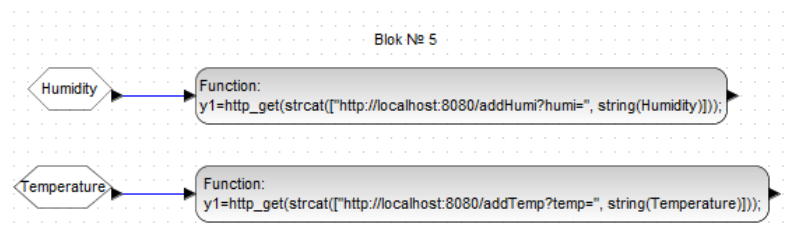
- Udržování teploty pomocí relé. 1 – zapnout, 2 – vypnout.



Obr. 29 Scilab-blok 4

5) Pátý blok (Obr. 15):

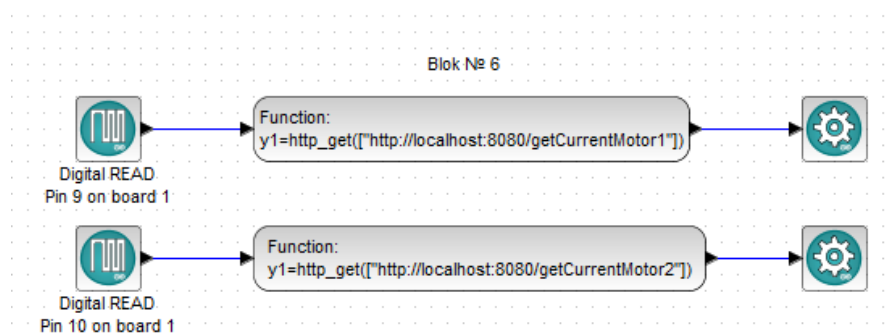
- Odesílání získaných dat ze snímače vlhkosti na server pomocí GET dotazu
„http_get(strcat([,http://localhost:8080/addHumi?humi=',
string(Humidity)]))“.
- Odesílání získaných dat ze snímače teploty na server pomocí GET dotazu „http_get(strcat([,http://localhost:8080/addTemp?temp=',
string(Temperature)]))“.



Obr. 30 Scilab-blok 5

6) Šestý blok (Obr. 16):

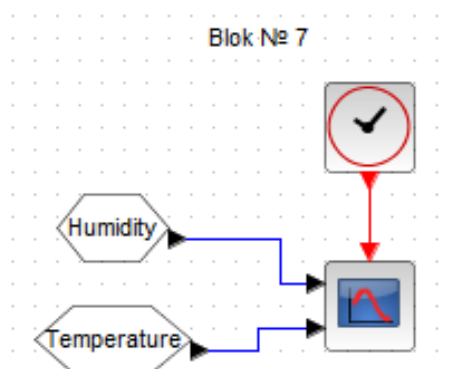
- Získání informací o poloze motoru №1 ze serveru a jeho otočení do vybrané polohy pomocí GET dotazu „`http_get([,http://localhost:8080/getCurrentMotor1 '])`“.
- Získání informací o poloze motoru №2 ze serveru a jeho otočení do vybrané polohy pomocí GET dotazu „`http_get([,http://localhost:8080/getCurrentMotor2 '])`“.



Obr. 31 Scilab-blok 6

7) Sedmý blok (Obr. 17):

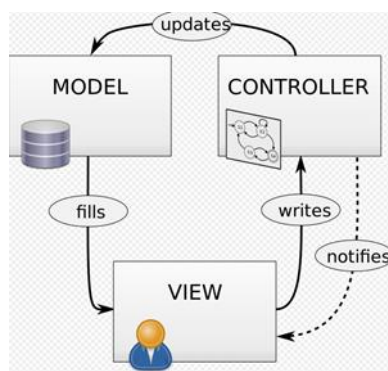
- Zobrazení hodnot teploty a vlhkosti v grafu.



Obr. 32 – Scilab-blok 7

6.2 Webový server

Skleník je řízen pomocí webového serveru postaveného na bázi systému MVC v jazyce Java. Pro zjednodušení vytváření webového serveru se používá framework Springboot viz Příloha 1.



Obr. 33 MVC Systém

Hlavní třídy modelů:

- 1) Humidity – Popisuje vlastnosti snímače vlhkosti.
- 2) Temperature – Popisuje vlastnosti snímače teploty.
- 3) HumidityCurrent – Popisuje vlastnosti snímače vlhkosti v reálném čase.
- 4) TemperatureCurrent – Popisuje vlastnosti snímače teploty v reálném čase.
- 5) MotorFirst – Popisuje vlastnosti prvního motoru.
- 6) MotorFirstCurrent – Popisuje vlastnosti prvního motoru v reálném čase.
- 7) MotorSecond – Popisuje vlastnosti druhého motoru.
- 8) MotorSecondCurrent – Popisuje vlastnosti druhého motoru v reálném čase.

Hlavní třídy řadičů:

- 1) MainController – Vývod systému řízení skleníku.
- 2) TemperatureController – Uložení hodnoty teploty ze snímače.
- 3) HumidityController – Uložení hodnoty vlhkosti ze snímače.
- 4) MotorController – Změna parametru prvního a druhého servomotoru (Otevřeno/ Zavřeno).

Hlavní pohled:

[Zobrazit všechna data](#)

Řídicí prvky	Aktuální	Změna
Teplota	27	<input type="text"/> Změnit
Vlhkost	76 %	<input type="text"/> Změnit
Motor 1	Zavřeno	<input type="text"/> Změnit
Motor 2	Otevřeno	<input type="text"/> Změnit

Obr. 34 Hlavní pohled

Maximální povolené hodnoty:

- Teplota – 50 °C.
- Vlhkost – 85 %.
- Motor 1 – Otevřeno/Zavřeno.
- Motor 2 – Otevřeno/Zavřeno.

Zobrazit všechny data:

[Hlavní pohled](#)

ČAS	VLHKOST	TEPLOTA	Motor 1	Motor 2
2021-02-20 20:28:35.352	70	23	Otevřeno	Otevřeno
2021-02-20 20:28:37.566	71	23	Zavřeno	Zavřeno
2021-02-20 20:28:39.634	74	23	Zavřeno	Zavřeno
2021-02-20 20:28:41.722	76	24	Zavřeno	Otevřeno

Obr. 35 Všechna data

Data pocházející ze Scilabu se načítají z adresy URL a jsou vkládána do modelu v závislosti na volaném řadiči a jeho typu. Pro odeslání údajů o teplotě se vytvoří požadavek GET typu (`http://localhost:8080/addTemp?temp=", string (u1)`),

kde u1 je hodnota teploty, která byla přijata ze snímače. Po přijetí dat ze senzorů je server uloží do místní databáze.

PostgreSQL. Databáze se konfiguruje pomocí souboru „application.properties“:

```
spring.datasource.url=jdbc:postgresql://localhost/groov
spring.datasource.username=postgres
spring.datasource.password=Disciples1
spring.jpa.generate-ddl=true
```

Obr. 36 Konfigurace spring

Server je spuštěn pomocí IDE IntelliJ Idea. Spuštění se provádí pomocí kombinace kláves „Ctrl + Shift + F10“ ve třídě „Application“ [1].

```
package org.example.groov;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class Application {
    public static void main(String[] args) {
        SpringApplication.run(Application.class, args);
    }
}
```

Obr. 37 Vstupní třída

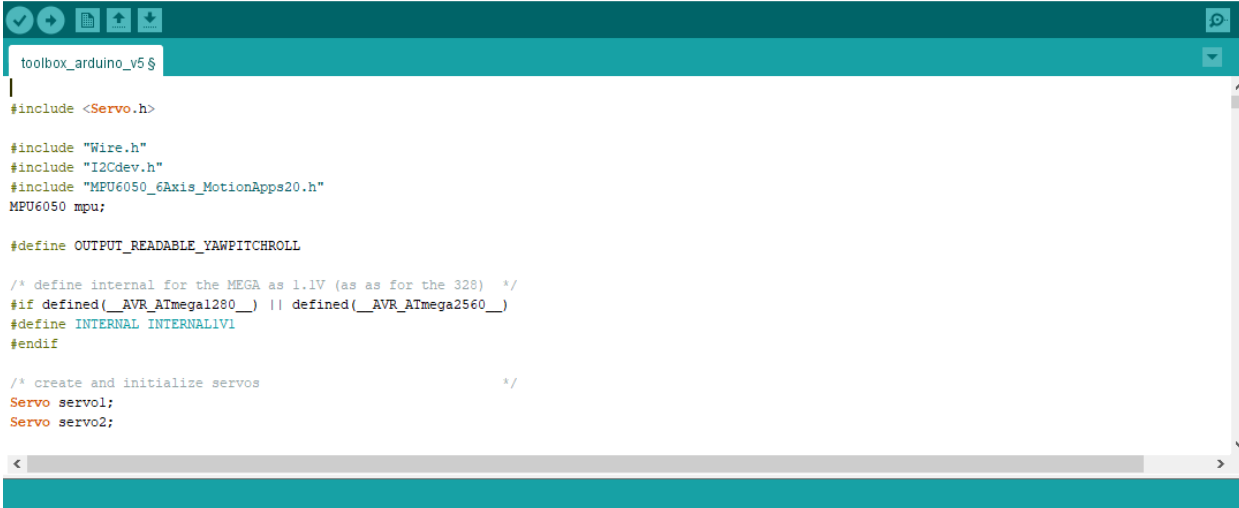
Knihovny potřebné pro fungování aplikace jsou propojeny pomocí souboru pom.xml.

Závislosti:

- 1) Spring-boot-starter-web.
- 2) Postgresql.
- 3) Spring-boot-starter-mustache.
- 4) Spring-boot-starter-data-jpa.

6.3 Arduino

Aby deska Arduino Uno správně fungovala se Scilab Xcos, musíte si na desku stáhnout knihovnu „toolbox_arduino_v5.ino“ prezentovanou na webu vývojáře viz Příloha 2. Knihovna se načte na platformu Arduino Uno připojením USB-kabelu a kombinace kláves „Ctrl + c“ a „Ctrl + U“.



```
toolbox_arduino_v5 $
|
#include <Servo.h>

#include "Wire.h"
#include "I2Cdev.h"
#include "MPU6050_6Axis_MotionApps20.h"
MPU6050 mpu;

#define OUTPUT_READABLE_YAWPITCHROLL

/* define internal for the MEGA as 1.1V (as as for the 328) */
#if defined(__AVR_ATmega1280__) || defined(__AVR_ATmega2560__)
#define INTERNAL INTERNAL1V1
#endif

/* create and initialize servos */
Servo servo1;
Servo servo2;
```

Obr. 38 Toolbox Arduino

Aby nedocházelo ke konfliktům verzí, používá se verze doporučená vývojářem knihovny. Pro správnou práci jsou vyžadovány následující knihovny:

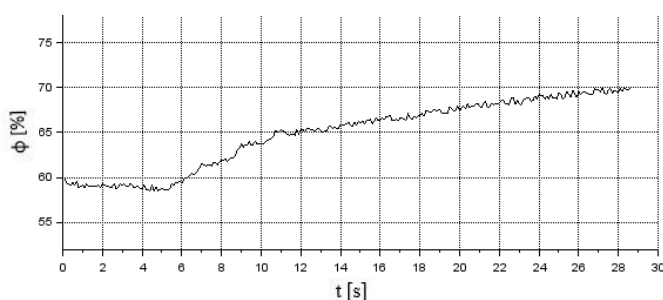
- 1) Servo.h.
- 2) Wire.h.
- 3) I2Cdev.h.
- 4) MPU6050_6Axis_MotionApps20.h.

7. Výsledky a diskuse

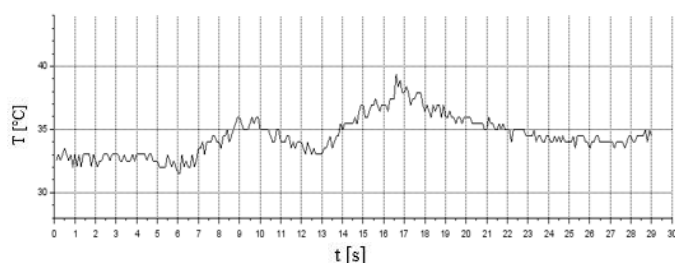
Vytvořený systém umožňuje měnit a řídit údaje o teplotě, vlhkosti, ventilech uvnitř skleníku. Bylo použito několik analogových a digitálních vstupů:

- 1) Senzor vlhkosti.
- 2) Teplotní čidlo.
- 3) Ovládání zvlhčovače relé.
- 4) Řízení topení pomocí relé.
- 5) Motor 1.
- 6) Motor 2.

Grafy 1 a 2 ukazují změnu parametrů vlhkosti a teploty za určité časové období.



Obr. 39 Změna vlhkosti, pin A1



Obr. 40 Změna teploty, pin A0

Specifikace Arduino Uno také umožňuje připojit další senzory, což rozšiřuje funkčnost systému. Softwarová část bude v tomto případě vyžadovat drobné úpravy a změny grafického rozhraní.

Implementace systému byla založena na místní správě skleníku, ale je také možné zlepšit instalaci umístěním kódu Java na hosting, jako je „Heroku“.

V případě takového vylepšení musíte změnit adresu požadavků na přenos dat na nové, které nabízí hostování.

8. Závěr

V této práci byl vytvořen systém udržování klimatických podmínek ve skleníku. K jeho implementaci byly napsány dvě aplikace:

- 1) Virtuální systém v programu Scilab Xcos.
- 2) Webová aplikace pro správu systému, napsaná v Javě pomocí frameworku SpringBoot.

Data se ukládají lokálně do počítače v databázi PostgreSQL. Vytvořená webová aplikace umožňuje prohlížet data přijatá ze senzorů po celou dobu provozu. Pomocí aplikace můžete také podle potřeby změnit parametry skleníku (změna teploty, vlhkosti, otevírání a zavírání ventilů).

Teoretická část popisuje použité konstrukční detaily a jejich specifikaci. Praktická část obsahuje koncept popsaného projektu. Komponenty byly vybrány především na základě ceny, životnosti a dostupnosti. Přidání grafického rozhraní umožňuje uživateli snadnou navigaci v systému a analýzu výstupních dat. Cena všech komponent nepřesáhla 2 000 Kč, což činí systém ziskovým.

Pokud je k dispozici zařízení, je možné vylepšit systém nahrazením dřevěných komponentů polykarbonátem. Pokud jde o software Scilab, je možné ovládat systém přímo, bez dalšího grafického rozhraní, ale uživatel je zbaven možnosti dálkově ovládat zařízení. V tomto případě jsou data zapisována přímo do textového souboru .txt pomocí interních schopností Scilab Xcos.

Instalaci lze použít k analýze dlouhodobých účinků teploty a vlhkosti na různé materiály. Systém splňuje všechny požadavky uvedené v zadání.

9. Seznam použitých zdrojů

- [1] **Herbert Schildt.** Java The Complete Reference Eleventh Edition [Online]
Dostupné
z: https://vk.com/doc37932704_564352858?hash=ebd51dbf940baec9be&dl=520f4a236069becf0c.
- [2] **The PostgreSQL Global Development Group.** Postgres Pro Standard 13.2.1 Documentation [Online] Dostupné z: <https://postgrespro.com/docs>.
- [3] <https://www.arduino.cc/>.
- [4] **Craig Walls.** Spring in action [Online] Dostupné
z: https://vk.com/doc26879026_479528892?hash=c4dfdbc03d4058300f&dl=54b5379234ee4e30f4.
- [5] **Documentation TMP36GT9Z.** [Online] Dostupné
z: <https://www.gme.cz/data/attachments/dsh.530-093.1.pdf>.
- [6] **AMT1001** [Online] Dostupné z: <https://www.gme.cz/amt1001-odporovy-senzor-vlhkosti>.
- [7] **Documentation SG9.** [Online] Dostupné
z: <https://www.gme.cz/data/attachments/dsh.772-211.1.pdf>.
- [8] **Jonatahn Oxe, Hugh Blemings.** Practical Arduino [Online] Dostupné
z: <http://rc-master.ucoz.ru/books/practical-arduino.pdf>.
- [9] **HTTP.**[Online] Dostupné z: <https://www.w3big.com/ru/http/default.html>
- [10] **Scilab Group.** On-line Documentation [Online] Dostupné
z: http://www.eeng.dcu.ie/~ee317/Matlab_Clones/manual.pdf.
- [11] **Brock Craft.** Arduino Projects for Dummies, 2013 ISBN-13: 978-1118551479

10. Seznam obrázků

Obr. 1 Arduino UNO R3	4
Obr. 2 Arduino	8
Obr. 3 Servomotor.....	9
Obr. 4 Snímač teploty.....	10
Obr. 5 Snímač vlhkosti.....	11
Obr. 6 Topný kabel.....	12
Obr. 7 Zvlhčovač vzduchu	13
Obr. 8 Skleník	15
Obr. 9 Zařízení skleníku	15
Obr. 10 From	16
Obr. 11 Goto.....	16
Obr. 12 Select_m.....	16
Obr. 13 Arduino_setup.....	17
Obr. 14 Analog_read	17
Obr. 15 Time_sample.....	17
Obr. 16 Servo_write	17
Obr. 17 Digital_write	17
Obr. 18 Time_delay.....	17
Obr. 19 Clock_c	18
Obr. 20 Ifthel_f.....	18
Obr. 21 Const_m	18
Obr. 22 Bigsom_f.....	18
Obr. 23 Cmscope.....	18
Obr. 24 Expression.....	19
Obr. 25 Scifunc_block_m	19
Obr. 26 Scilab-blok 1	19
Obr. 27 Scilab-blok 2	20
Obr. 28 Scilab-blok 3	20
Obr. 29 Scilab-blok 4	21

Obr. 30 Scilab-blok 5	21
Obr. 31 Scilab-blok 6	22
Obr. 32 Scilab-blok 7	22
Obr. 33 MVC Systém.....	23
Obr. 34 Hlavní pohled.....	24
Obr. 35 Všechna data	24
Obr. 36 Konfigurace spring.....	25
Obr. 37 Vstupní třída.....	25
Obr. 38 Toolbox Arduino.....	26
Obr. 39 Změna vlhkosti, pin A1	27
Obr. 40 Změna teploty, pin A0	27

11. Seznam tabulek

Tabulka 1: Specifikace Arduino Uno.....	4
Tabulka 2: SG0.....	10
Tabulka 3: ATM1001	12
Tabulka 4: Směrování signálu.....	16
Tabulka 5: Arduino blok	17
Tabulka 6: Systémy s nepřetržitým časem.....	17
Tabulka 7: Zpracování událostí.....	18
Tabulka 8: Matematické operace	18
Tabulka 9: Registrační zařízení.....	18
Tabulka 10: Vlastní funkce	19

12. Seznam příloh

Příloha 1: Webová aplikace	1
Příloha 2: Program pro společnou práci Arduino a Scilab	1
Příloha 3: Knihovna Arduino v programu Scilab	2