

Bakalářská práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra telekomunikační techniky

Demonstrace řízení teploty v rodinném domě na základě měření teploty s využitím přípravku Nexys 4

Ondřej Janovský

Vedoucí: Ing. Pavel Lafata, Ph.D.
Obor: Elektronika a komunikace
Studijní program: Elektronika a komunikace
Květen 2020

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Janovský** Jméno: **Ondřej** Osobní číslo: **465922**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra telekomunikační techniky**
Studijní program: **Elektronika a komunikace**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Demonstrace řízení teploty v rodinném domě na základě měření teploty s využitím přípravku Nexys4

Název bakalářské práce anglicky:

Using Nexys4 Kit to Measure the Temperature and to Control the Heating System in a House

Pokyny pro vypracování:

Seznamte se s přípravkem Digilent Nexys4 a jeho obsluhou pomocí jazyka VHDL. K přípravku připojte pomocí standardních digitálních rozhraní dvojici teplotních čidel typu DS18B20, případně doplňte využití vestavného čidla ADT7420. Navrhněte a realizujte VHDL kódy pro periodické měření teploty z těchto čidel a její zobrazení na dvojici segmentových displejů na přípravku Nexys4. Demonstrujte využití aktuálně naměřených teplot formou modelového řízení teploty rodinného domu. Uvažujte různé stavy (teploty) čidel a simulujte pomocí dostupných jednoduchých prvků (reproduktor, displej, LED diody, apod.) možnost ovládání a řízení teploty v modelovém příkladu.

Seznam doporučené literatury:

- [1] Manuál a dokumentace k přípravku Digilent Nexys4. Dostupné na: <https://store.digilentinc.com/nexys-4-artix-7-fpga-trainer-board-limited-time-see-nexys4-ddr/> [on-line]
- [2] Manuál a datasheet teplotního čidla DS18B20. Dostupné na: <https://www.gme.cz/ds18b20> [on-line]

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Pavel Lafata, Ph.D., katedra telekomunikační techniky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **08.01.2020**

Termín odevzdání bakalářské práce: **22.05.2020**

Platnost zadání bakalářské práce: **30.09.2021**

Ing. Pavel Lafata, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studenta

Poděkování

Rád bych poděkoval svému vedoucímu práce Ing. Pavlu Lafatovi, Ph.D. za odborné vedení, připomínky, licenci k softwaru a věnovaný čas. Také bych rád poděkoval své rodině a přátelům za morální podporu.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze, 22. května 2020

Abstrakt

Tato bakalářská práce se věnuje modelovému návrhu automatického zařízení pro energeticky efektivní řízení teploty v rodinném domě pomocí rekuperační jednotky. Jako platforma je využit kit Nexys 4™ FPGA Board s programovatelným hradlovým polem naprogramovaným v jazyce VHDL. Práce nastiňuje princip funkce systému, způsob získávání údajů o teplotách pomocí čidla DS18B20. Dále je popsán způsob zobrazení naměřených teplot na sedmissegmentový displej a uživatelské ovládání systému.

Klíčová slova: VHDL, FPGA, Nexys 4, DS18B20, rekuperační jednotka, ventilace, topení, energetická efektivita, sedmissegmentový displej, letní režim, zimní režim

Vedoucí: Ing. Pavel Lafata, Ph.D.
České vysoké učení technické v Praze,
Fakulta elektrotechnická,
Katedra telekomunikační techniky,
Technická 2,
16627 Praha 6

Abstract

This bachelor thesis describes the model design of an automatic device for energy-efficient temperature control in a family house using a recuperation unit. As a platform, the Nexys 4 FPGA Board with a field programmable gate array programmed in VHDL. The thesis outlines the main functions of the system, the method of obtaining temperature data using the DS18B20 temperature sensor. Furthermore, the method of displaying measured temperatures on a seven-segment display and efficient control of the system is described.

Keywords: VHDL, FPGA, Nexys 4, DS18B20, recuperation unit, ventilation, heating, energy efficiency, seven-segment display, summer mode, winter mode

Title translation: Using Nexys 4 Kit to Measure the Temperature and to Control the Heating System in a House

Obsah

1 Úvod	1	3.2 Vlastní řešení - popis jednotlivých bloků systému	29
2 Teoretická část	3	3.2.1 Bloky systému	29
2.1 Rozbor zadání	3	3.3 Vlastní řešení - typické výstupy .	41
2.2 Použité nástroje	5	3.3.1 Zimní stavy	42
2.2.1 Jazyk VHDL	5	3.3.2 Letní stavy	46
2.2.2 Xilinx ISE Design Suite	5	3.3.3 Další možné výstupy	48
2.2.3 PC Osciloskop	6	3.3.4 Celkový pohled na zařízení . .	49
2.2.4 Nexys 4	6	3.4 Kabeláž čidel	50
2.2.5 FPGA	7	3.5 Uvedení do provozu	50
2.2.6 Zdroj hodinového signálu	8	3.6 Bezpečnost	51
2.2.7 Sedmsegmentový displej	9	4 Závěr	53
2.2.8 PMOD Porty	9	5 Citace	55
2.2.9 Digitální čidlo DS18B20	10	.1 Ukázky VHDL kódu	58
2.2.10 Protokol komunikace s čidlem	13		
3 Praktická část	17		
3.1 Vlastní řešení - návrh	17		

Obrázky

2.1 PC osciloskop	6	2.14 Resetovací a ověřovací pulz (1, str. 15, upraveno)	15
2.2 Nexys 4 - součásti	7	2.15 Read a Write time sloty (1, str. 16, upraveno)	16
2.3 Porty - pinout	8	3.1 Hlavní část	24
2.4 Pinout jednoho segmentu	9	3.2 Blok A	25
2.5 Nexys 4 - Port PMOD piny	10	3.3 Blok B	25
2.6 Nexys 4 - Namapování portů PMOD	10	3.4 Blok C	26
2.7 Blokové schéma čidla DS18B20 (1, str 2., upraveno)	11	3.5 Blok D	26
2.8 Paměť Scratchpad - rozdělení (1, str. 7, upraveno)	11	3.6 Blok E	27
2.9 Scratchpad: formát teploty (1, str. 4, upraveno)	12	3.7 Blokové schéma systému	31
2.10 Scratchpad: Ukázka reprezentace teploty (1, str. 4, upraveno)	12	3.8 Stavový diagram vyčítání teploty	37
2.11 Bitů nastavení rozlišení teploty (1, str. 8, upraveno)	12	3.9 Vstupy a výstupy	42
2.12 Konfigurace bitů nastavení rozlišení teploty (1, str.8, upraveno)	13	3.10 Zima - 1	42
2.13 Algoritmus výpočtu CRC kódu (1, str. 9, upraveno)	13	3.11 Zima - 1	43
		3.12 Zima - 2	43
		3.13 Zima - 3	43
		3.14 Zima - 4	44
		3.15 Zima - 5	44

3.16 Zima - 6	45
3.17 Zima - 7	45
3.18 Zima - 8	45
3.19 Zima - 9	46
3.20 Léto - 1	46
3.21 Léto - 2	47
3.22 Léto - 3	47
3.23 Léto - 4	47
3.24 Léto - 5	48
3.25 Odpojené výstupy	48
3.26 Odpojené výstupy	49
3.27 Odpojené výstupy	49
3.28 Celkový pohled	50

Tabulky

3.1 Zimní stavy - část 1	19
3.2 Zimní stavy - část 2	20
3.3 Zimní stavy - část 3	21
3.4 Letní stavy - část 1	22
3.5 Letní stavy - část 2	23



Kapitola 1

Úvod

Tato bakalářská práce obsahuje návrh a modelovou realizaci řídicího zařízení rekuperační jednotky za použití přípravku Digilent Nexys 4. Jedná se o model celoročního řízení teploty moderního nízkoenergetického domu založeného na efektivním způsobu větrání pomocí rekuperační jednotky s výměníkem tepla, který minimalizuje použití dalších zdrojů tepla (teplovzdušné vytápění pomocí rekuperační jednotky, či externí topení nebo klimatizace).

Zařízení měří pomocí dvojice čidel DS18B20 venkovní a vnitřní teplotu, průměruje naměřenou venkovní teplotu za účelem zjištění teplotních extrémů během dne a dle těchto údajů spíná rekuperační jednotku. Konkrétně jde o její následující funkce: větrání s a bez tepelného výměníku a teplovzdušné vytápění. Měřené teploty jsou zobrazovány na sedmissegmentovém displeji a referenční teploty jsou uživatelsky nastavitelné.

V práci je nejprve detailně popsána požadovaná funkčnost, následně použité hardwarové prostředky, konkrétní implementace ve VHDL kódu, a nakonec samotná praktická realizace.

Kapitola 2

Teoretická část

2.1 Rozbor zadání

Teplotu rodinného domu lze regulovat několika způsoby. Mezi jinými vytápěním, větráním a klimatizací. V našem modelovém případě použijeme rekuperační jednotkou s teplovzdušným vytápěním a tepelným výměníkem, kde si vzájemně předává teplo čerstvý vzduch vpouštěný dovnitř a vypouštěný vydýchaný vzduch. Příkladem takové jednotky mohla být např. ATREA DUPLEX RB 730 / 440. [10] [11] [12]

Úlohu lze rozdělit do dvou základních částí podle ročního období. V zimě by se úloha dala popsat následovně: Ochlazovat dům větráním co nejméně a zbývající teplotní rozdíl upravit vytápěním. Větrání je optimální v nejteplejší denní době, tj. okolo 12. – 14. hodiny. V této době je rozdíl mezi vyšší teplotou v domě a nižší venkovní teplotou nejmenší. V optimálním případě je venku dokonce tepleji. To je v zimě řídká situace, může však nastat. Je nezbytné uživatelsky nastavit požadovanou teplotu v domě, a sice zvlášť pro denní a zvlášť pro noční hodiny. V případě, že je venku chladněji, a v domě je teplota nižší než žádoucí, spustí se větrání rekuperační jednotkou a zbývající teplotní rozdíl se upraví přitápěním vpouštěného vzduchu.

Je-li však venku chladněji o více než 30°C, jednotka přestane větrat zcela a vyčkává se na příhodnější meteorologické podmínky z hlediska energetické účinnosti.

Není-li zrovna optimální doba pro větrání, opět se vyhodnotí, zda je teplota v domě pod nebo nad žádoucí hodnotou a je-li pod ní, spustí se vnitřní recirkulace vzduchu po domě a teplovzdušné vytápění.

V létě je úloha v podstatě opačná. Dům chceme udržet pokud možno v

chladu jako útočiště před vnějšími teplotami. Moderní rodinné domy s dobrou tepelnou izolací klimatizaci nutně nepotřebují (mohu potvrdit z vlastní zkušenosti) a vhodná teplota se v nich dá udržet za pomoci prostého větrání.

Opět uživatelsky nastavíme dvě teploty. Tentokrát však nikoli denní a noční, ale minimální žádaná teplota v domě, pod kterou se nemá vzduch ochlazovat a pak maximální venkovní teplota, při které se má větrat s výměníkem z důvodu jeho maximální účinnosti [11].

Dále je obvyklé, že jednotka umožňuje vyvětrat na toaletě a v kuchyni při vaření. Zavedme tedy na toaletě tlačítko, po jehož stisku se bude větrat 3 minuty za všech meteorologických podmínek. S výměníkem či bez, ale bez omezení v případě teplotních extrémů. Do kuchyně zavedme tentokrát spínač. Ten, je-li v příslušné poloze, indikuje větrání až do jeho opětovného přepnutí do původní polohy. Není tedy automaticky vypnut jako tlačítko na toaletě, musí být vypnut manuálně.

■ 2.2 Použité nástroje

■ 2.2.1 Jazyk VHDL

VHDL je jazyk pro popis funkce digitálních zařízení. [13] Kódem se popíše, jak má zařízení fungovat a na příslušném kompilátoru je pak syntéza popsaného zařízení z dostupných hardwarových prostředků, v mém případě hradlového pole. Možnou alternativou k tomuto jazyku je jazyk Verilog.

Ve VHDL je kód členěný do bloků s definovanými vstupy a výstupy. K portům celého zařízení je nutno přiřadit fyzické porty zařízení, pro než je kód určen. Kód je tedy přenositelný mezi různými architekturami, kdežto namapování portů již nikoliv.

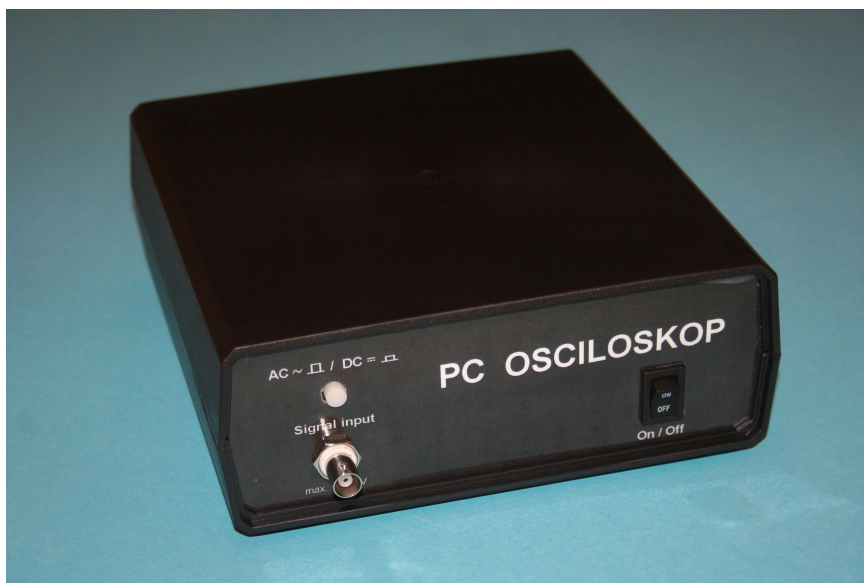
Více informací naleznete například v knize *The VHDL Handbook* [15].

■ 2.2.2 Xilinx ISE Design Suite

Xilinx ISE Design Suite je vývojové prostředí od firmy Xilinx umožňující vývoj VHDL kódu, jeho kompilaci a testování v prostředí ISim. Dále umožňuje návrh na nižší úrovni abstrakce pomocí RTL popisu a mnohé další funkce. [14]

2.2.3 PC Osciloskop

K ověření funkce byl použit PC osciloskop vlastní výroby (Obrázek 2.1). Dokumentaci s technickými podrobnostmi naleznete na přiloženém CD.



Obrázek 2.1: PC osciloskop

2.2.4 Nexys 4

Nexys 4™ FPGA Board [2] je kit pro vývoj digitálních elektrických obvodů s programovatelným hradlovým polem FPGA Artix-7. Součástí kitu jsou i vestavěné periferie jako akcelerometr, teplotní senzor a MEMS digitální mikrofon. Kit je podporován vývojovými prostředími Xilinx ISE a Xilinx Vivado [4]. Součásti desky jsou na obrázku 2.2.

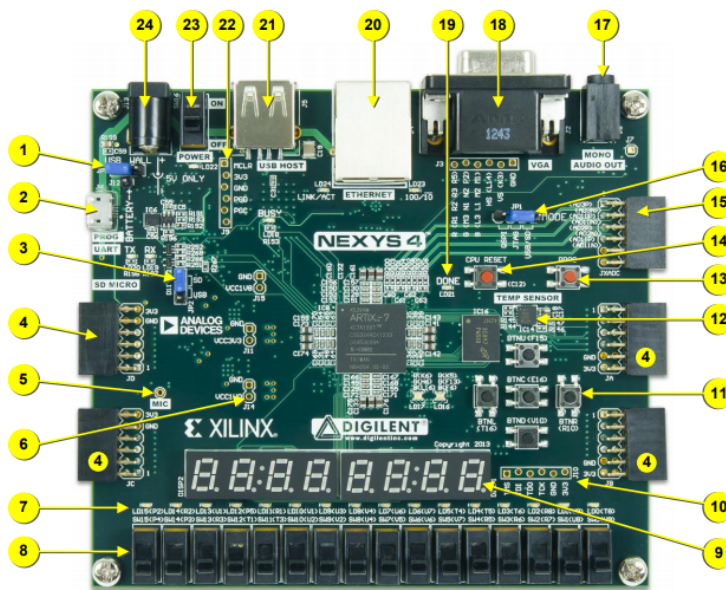


Figure 1. Nexys 4 board features

Callout	Component Description	Callout	Component Description
1	Power select jumper and battery header	13	FPGA configuration reset button
2	Shared UART/ JTAG USB port	14	CPU reset button (for soft cores)
3	External configuration jumper (SD / USB)	15	Analog signal Pmod port (XADC)
4	Pmod port(s)	16	Programming mode jumper
5	Microphone	17	Audio connector
6	Power supply test point(s)	18	VGA connector
7	LEDs (16)	19	FPGA programming done LED
8	Slide switches	20	Ethernet connector
9	Eight digit 7-seg display	21	USB host connector
10	JTAG port for (optional) external cable	22	PIC24 programming port (factory use)
11	Five pushbuttons	23	Power switch
12	Temperature sensor	24	Power jack

Obrázek 2.2: Nexys 4 - součásti

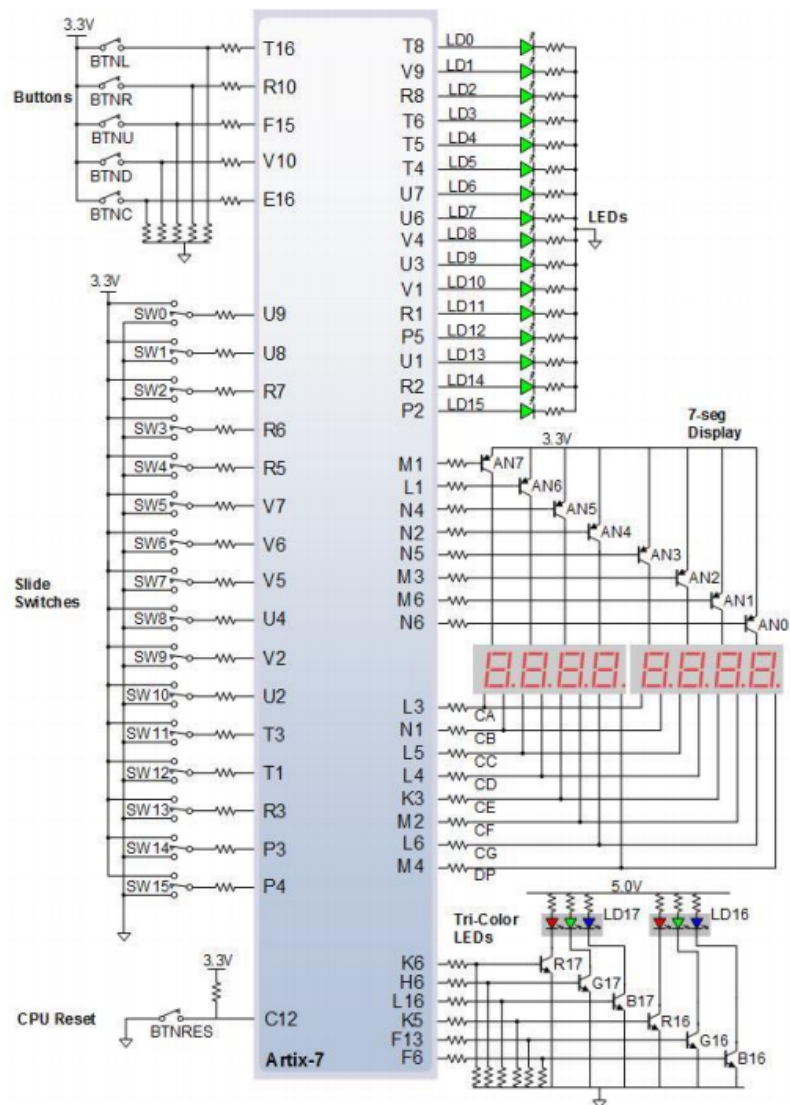
2.2.5 FPGA

Deska Nexys 4 obsahuje také FPGA. FPGA je programovatelné hradlové pole sestávající z logických bloků. Každý blok má několik hlavních částí. Vyhledávací tabulka LUT slouží k realizaci samotných logických funkcí. Multiplexor pak může posílat dále výstup z této tabulky či z logického bloku. A nakonec je zde registr. Ten může být nastaven jako klopný obvod flip-flop nebo latch, podle toho, zda reaguje na hranu či úroveň. Dále do každého bloku vstupuje ještě povolovací vstup, hodinový signál a reset. Výstupy jsou dva. Prvním je výstup z vyhledávací tabulky LUT a druhým je výstup z registru. Z těchto stavebních bloků se pak skládá celé pole FPGA, potažmo realizované zařízení. [16] Syntetizátor jazyka VHDL pak propojí tyto logické

bloky automaticky dle zdrojového kódu a v této práci s nimi nebudeme přímo pracovat.

2.2.6 Zdroj hodinového signálu

Nexys 4 disponuje zdrojem hodinového signálu o frekvenci 100 MHz. [2]

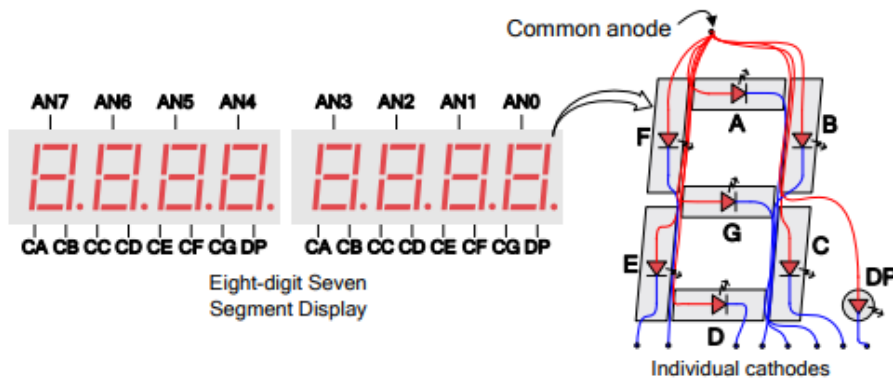


Obrázek 2.3: Porty - pinout

2.2.7 Sedmisegmentový displej

Nexys 4 je vybaven displejem sedmisegmentového typu s desetinnou tečkou. Má celkem osm digitů. Celá soustava digitů je zapojena maticově. Segmenty na jednom digitu se jsou zapojeny paralelně, avšak při použití více digitů je pro jejich současné rozsvícení nutné použít časový multiplex. Doporučená frekvence je 60Hz až 1kHz. Jednotlivé segmenty mají propojené anody. Segmenty se řídí negativní logikou, na rozsvícení je potřeba na porty CA – CF i na porty L1, M, M3, M6, N2, N4, N5, N6 přivést logickou nulu. Obrázek 2.3 ukazuje namapování displeje a dalších periférií na porty Nexysu. [2]

Obrázek 2.4 ukazuje rozpis namapování jednotlivých segmentů displeje pro každý digit [2].



Obrázek 2.4: Pinout jednoho segmentu

Kromě zmíněného displeje Nexys disponuje ještě dvěma tříbarevnými LED diodami. Dále patnácti jednobarevnými signalizačními LED diodami a stejným počtem přepínačů. Dále šest tlačítek, z toho jedno funguje jako reset, další jsou uživatelská. Tlačítka a přepínače jsou chráněny pull-up rezistory proti omylu způsobenému zkratu. [2]

2.2.8 PMOD Porty

Nexys disponuje čtyřmi PMOD porty plus jeden pro DAC převodník. Piny 6 a 12 poskytují napájení 3,3V o proudu 1A. Zbýlé piny jsou logické datové. [2]



Obrázek 2.5: Nexys 4 - Port PMOD piny

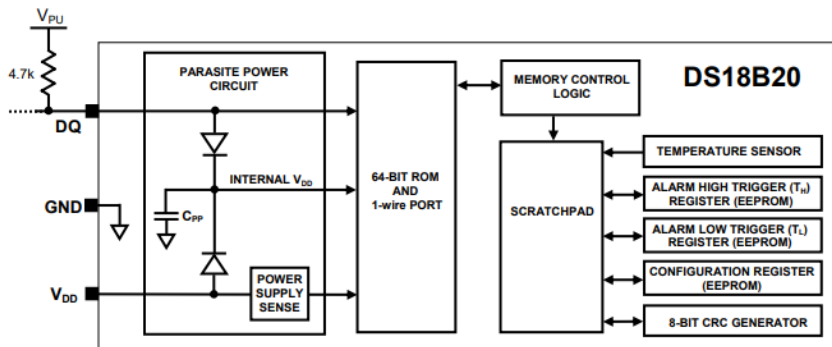
Pmod JA	Pmod JB	Pmod JC	Pmod JD	Pmod XDAC
JA1: B13	JP1: G14	JC1: K2	JD1: H4	JXADC1: A13
JA2: F14	JB2: P15	JC2: E7	JD2: H1	JXADC2: A15
JA3: D17	JB3: V11	JC3: J3	JD3: G1	JXADC3: B16
JA4: E17	JB4: V15	JC4: J4	JD4: G3	JXADC4: B18
JA7: G13	JB7: K16	JC7: K1	JD7: H2	JXADC7: A14
JA8: C17	JB8: R16	JC8: E6	JD8: G4	JXADC8: A16
JA9: D18	JB9: T9	JC9: J2	JD9: G2	JXADC9: B17
JA10: E18	JB10: U11	JC10: G6	JD10: F3	JXADC10: A18

Obrázek 2.6: Nexys 4 - Namapování portů PMOD

2.2.9 Digitální čidlo DS18B20

DS18B20 je digitální čidlo původně od firmy Dallas Semiconductor, dnes již Integrated Products. Disponuje programovatelným rozlišením 9 až 12 bitů, kde rozsah je $-55\text{ }^{\circ}\text{C}$ až $+125\text{ }^{\circ}\text{C}$, kde celočíselný výsledek je v Celsiově stupnici. V této práci bylo pužito rozlišení 12 bitů. Komunikace probíhá po sběrnici One-Wire umožňující provoz čidla s napájením po datové sběrnici – „Parazitické napájení“. [1] Čidlo má již z výroby svůj unikátní 64-bitový kód, umožňující provoz více čidel na jedné sběrnici. Tato čidla budeme v dalším textu nazývat „slave“ a řídicímu zařízení sběrnice - desce Nexys 4 - budeme říkat „master“.

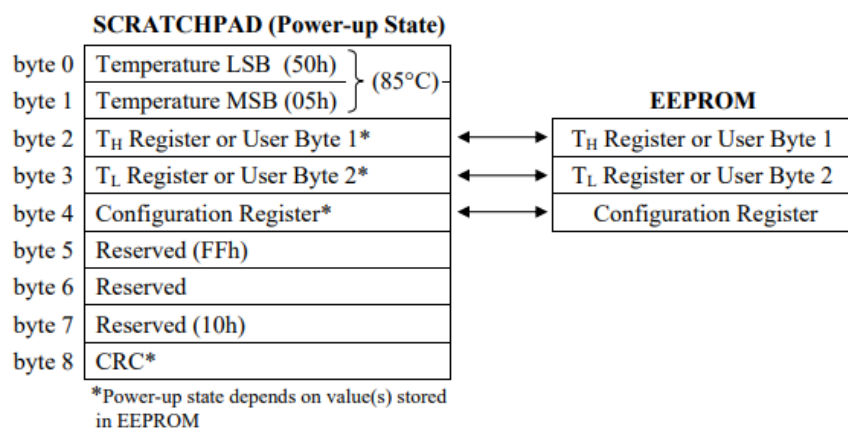
Jak vidíme, čidlo má 3 piny. (Obrázek 2.7) Datový pin DQ slouží zároveň pro parazitické napájení, je-li čidlo provozováno v tomto módu. Pin VDD slouží jako napájecí v módu s externím napájením. Pin GND je pak zemnicí jak pro napájení, tak pro datový vodič. Přípustné napájecí napětí se pohybuje v rozsahu 3,3V až 5V.



Obrázek 2.7: Blokové schéma čidla DS18B20 [1], str 2., upraveno

Na vstupu do čidla je blok pro zajištění parazitického napájení. Je-li sběrnice ve stavu plného napájecího napětí VPU, pak toto napětí nabíjí přes pull-up rezistor $4,7k\Omega$ kondenzátor CPP, který zajišťuje napájení čidla ve stavu, je-li na sběrnici nula. Následuje I/O buffer odesílající a přijímající data do a ze senzoru. Taktéž je zde 64-bitová paměť ROM, ve které je uložena stejně dlouhá unikátní 64-bitová adresa čidla. Za ní následuje řídicí obvod celého čidla a pak již následuje devítibajtová paměť „SCRATCHPAD“ (Obrázek 2.8) . Bajty 0 a 1 slouží pro uložení naměřené teploty. Bajt 1 je MSB, bajt 0 je pak LSB. Jednotlivé bity budeme číslovat od MSB k LSB sestupně 15 až 0 (Obrázek 2.9).

Bits 15 až 11 reprezentují znaménko teploty (a doplňují tak zbytek bajtu), kde logická 1 reprezentuje teplotu pod bodem mrazu, kdežto logická 0 teplotu nad bodem mrazu. Bits 10 až 4 reprezentují celočíselnou část teploty v dvojkovém doplňku pro zápornou část. [5] Bits 3 až 0 pak reprezentují neceločíselnou část teploty, bit 3 reprezentuje poloviny stupně, bit 2 čtvrtiny, bit 1 osminy a konečně bit 0 šestnáctiny stupně, což je zároveň největší rozlišení čidla. Příklady takto reprezentovaných teplot viz Obrázek 2.10 .



Obrázek 2.8: Paměť Scratchpad - rozdělení [1], str. 7, upraveno

	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
LS Byte	2 ³	2 ²	2 ¹	2 ⁰	2 ⁻¹	2 ⁻²	2 ⁻³	2 ⁻⁴
	bit 15	bit 14	bit 13	bit 12	bit 11	bit 10	bit 9	bit 8
MS Byte	S	S	S	S	S	2 ⁶	2 ⁵	2 ⁴

Obrázek 2.9: Scratchpad: formát teploty [1], str. 4, upraveno

TEMPERATURE	DIGITAL OUTPUT (Binary)	DIGITAL OUTPUT (Hex)
+125°C	0000 0111 1101 0000	07D0h
+85°C*	0000 0101 0101 0000	0550h
+25.0625°C	0000 0001 1001 0001	0191h
+10.125°C	0000 0000 1010 0010	00A2h
+0.5°C	0000 0000 0000 1000	0008h
0°C	0000 0000 0000 0000	0000h
-0.5°C	1111 1111 1111 1000	FFF8h
-10.125°C	1111 1111 0101 1110	FF5Eh
-25.0625°C	1111 1110 0110 1111	FE6Fh
-55°C	1111 1100 1001 0000	FC90h

*The power-on reset value of the temperature register is +85°C

Obrázek 2.10: Scratchpad: Ukázka reprezentace teploty [1], str. 4, upraveno

Po dvou bajtech reprezentujících teplotu následují bajty 2 a 3 – TH a TL. Tyto bajty slouží jako komparační registry. Pokud je naměřená teplota vyšší než TH nebo naopak nižší než TL, dojde k nastavení příznaků v čidle, které lze vyčíst příkazem Alarm Search [ECh] (viz. dále). Bajt 4 je pak konfigurační registr pro nastavení rozlišení měřené teploty (Obrázek 2.11). Bity 7 a 4-0 jsou pouze na doplnění do bajtu a žádnou konfigurační funkci nemají.

bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0
0	R1	R0	1	1	1	1	1

Obrázek 2.11: Bity nastavení rozlišení teploty [1], str. 8, upraveno

Jinak je tomu u bitů 6 a 5. Dle jejich nastavení se rozhoduje mezi čtyřmi možnými teplotními rozlišeními 9, 10, 11 nebo 12 bitů (Obrázek 2.12). Toto nastavení hraje klíčovou roli i kvůli minimální době měření, kterou je nutné zohlednit v komunikačním protokolu čidla (viz. dále).

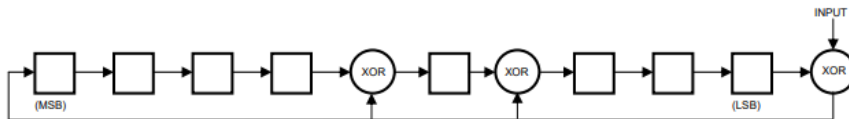
R1	R0	Resolution	Max Conversion Time	
0	0	9-bit	93.75 ms	($t_{CONV}/8$)
0	1	10-bit	187.5 ms	($t_{CONV}/4$)
1	0	11-bit	375 ms	($t_{CONV}/2$)
1	1	12-bit	750 ms	(t_{CONV})

Obrázek 2.12: Konfigurace bitů nastavení rozlišení teploty [1], str.8, upraveno

Bajty 5 až 7 opět žádnou funkci nekonfigurují, avšak jsou součástí výpočtu CRC kódu jako konstanty. V osmém bajtu je uložen CRC kód paměti. CRC kód slouží pro kontrolu správnosti přenosu dat. Dle vztahu (viz. dále) se vypočítá CRC kód v čidle a po přenesení do masteru je CRC kód vypočítán z přenesených dat znova. Jestliže je přenesený a vypočtený CRC kód shodný, data se přenesla v pořádku. V opačném případě došlo k chybě a je nutno vyčtení dat zopakovat. CRC kód se vypočítává [1] z bitů 0 až 56 dle vztahu:

$$CRC = X^8 + X^5 + X^4 + 1$$

Všechny bity mimo vstup jsou pro začátek nastaveny do nuly, a pak se do registru vsunují bity ze vstupu. V každé iteraci jsou na tyto data na příslušných bitech vyplývající z zmíněného vztahu aplikovány operace XOR s krajním bitem. Grafické znázornění vzorce (výše) vidíme zde:



Obrázek 2.13: Algoritmus výpočtu CRC kódu [1], str. 9, upraveno

2.2.10 Protokol komunikace s čidlem

Komunikační protokol na úrovni bitů-instrukcí

Komunikační protokol čidla DS18B20 sestává ze tří kroků:

1. Inicializace
2. ROM příkazy

3. Funkční příkazy

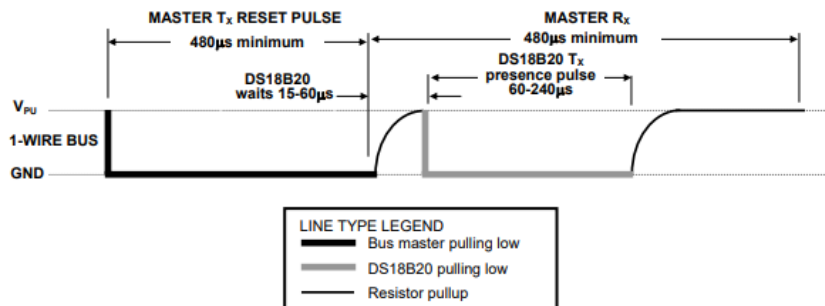
Ve fázi inicializace master vyšle resetovací pulz. Následuje prezenční pulz, na který slave zařízení odpoví svým vlastním prezenčním pulzem. Po této fázi se přistupuje k ROM příkazům, jejichž účelem je získat unikátní adresy všech zařízení na sběrnici. Když je toho dosaženo, slouží tato sada příkazů k jejich adresaci:

- SEARCH ROM [F0h] – V případě, že se Master připojuje ke sběrnici poprvé, je použit příkaz SEARCH ROM. Jedná se o poměrně složitý algoritmus [6], jehož výsledkem je získání 64-bitových adres všech zařízení na sběrnici.
- READ ROM [33h] – Obdoba předchozího pro případ, že je na sběrnici jen jedno zařízení. Master získá adresu slave zařízení jednoduchým mechanismem.
- MATCH ROM [55h] – V případě, že již adresy máme, pak můžeme tímto příkazem adresovat konkrétní slave zařízení a vyslat další příkazy.
- SKIP ROM [CCh] – Obdoba předchozího pro 1 zařízení, vůbec neobsahuje adresu, po obdržení je slave okamžitě připraven k přijetí dalších příkazů.
- ALARM SEARCH [ECh] – Odpoví všechny slavy s nastaveným alarm set příznakem. Tedy ty, u nichž je teplota vyšší než TH, nebo nižší než TL a ty je pak třeba obsloužit.

A konečně následuje třetí fáze, funkční příkazy, kdy již adresovanému zařízení vysíláme konkrétní požadavek. Příkazy z této sady jsou následující:

- CONVERT T [44h] – Čidlo provede konverzi teploty a zapíše ji do paměti SCRATCHPAD. Používám-li parazitický mód, musím nastavit sběrnici po dobu konverze do stavu 1. Když je převod dokončen, čidlo se ohlásí pulzem.
- WRITE SCRATCHPAD [4Eh] – přenesení do paměti SCRATCHPAD 3 konfigurovatelné bajty – TH, TL a bajt nastavení rozlišení.
- READ SCRATCHPAD [BEh] – vyčtení celé paměti SCRATCHPAD.
- COPY SCRATCHPAD [48h] – uložení bajtů paměti 2,3 a 4 do paměti EEPROM, kde tyto zůstanou uchovány i po přerušování napájení.

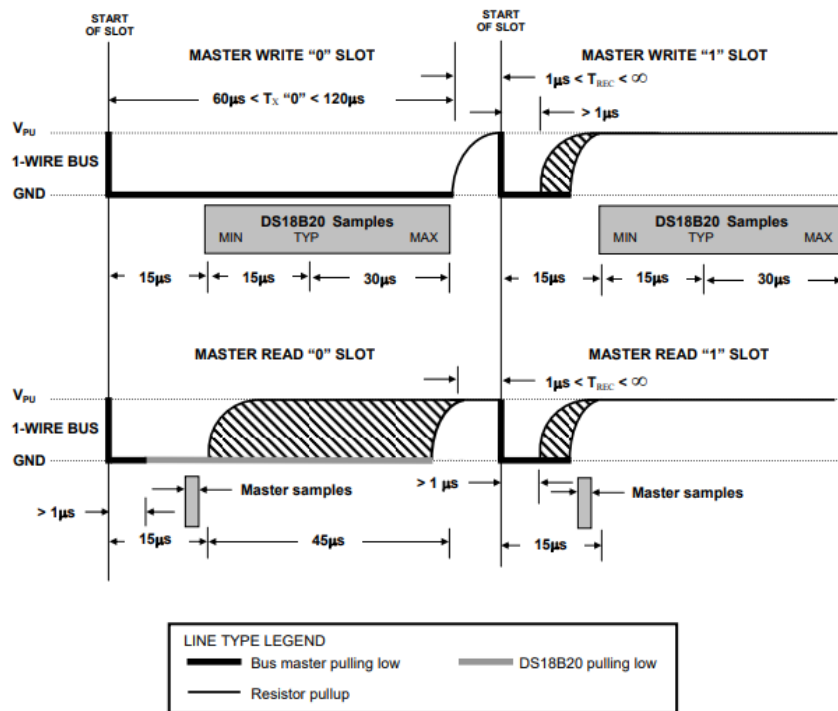
- **RECALL E2 [B8h]** – Vymazání bajtů 2,3 a 4 z paměti SCRATCHPAD a EEPROM v čidle.
- **READ POWER SUPPLY [B4h]** – Zjištění, jestli jsou čidla napájena parazitně či externě.



Obrázek 2.14: Resetovací a ověřovací pulz [1], str. 15, upraveno

■ Komunikační protokol na úrovni napěťových signálů

Master každou komunikaci iniciuje následujícím způsobem. Master vyšle úvodní resetovací pulz a poté vyšle prezenční pulz. Pokud jej slave zachytí, vyšle stejný pulz masteru nazpět a potvrdí tak svou přítomnost na sběrnici a připravenost komunikovat. Pak již probíhá komunikace v rámci instrukcí, tedy v bitech. V bitové komunikaci na 1-Wire sběrnici rozeznáváme 4 komunikační sloty. Jsou to Master Write „0“ Slot, Master Write „1“ Slot, MasterRead „0“ Slot a Master Read „1“ Slot. Na 1-Wire sběrnici řídí veškerou komunikaci master a to do té míry, že slave může vysílat pouze tehdy, pokud mu k tomu master nastaví read time slot. Slave jej pak může pouze nastavit do jedničky nebo nuly. Komunikační iniciativa ze strany slavu není technicky vůbec možná. To v praxi znamená, že master tyto sloty musí nastavit vždy tehdy, když mohou nastat, tj. po odeslání instrukcí Convert T nebo Recall E2. Jinak by slave neměl možnost data předat. Časování slotů (viz. Obrázek 2.14 a Obrázek 2.15) je bezpodmínečně nutné dodržet, je to opět zejména záležitost masteru.



Obrázek 2.15: Read a Write time slots [1], str. 16, upraveno

Kapitola 3

Praktická část

3.1 Vlastní řešení - návrh

Nyní se dostáváme k podrobnému návrhu. Úkolem zařízení je regulovat teplotu a výměnu vzduchu v rodinném domě následujícími způsoby:

- Větráním rekuperační jednotkou v optimálním čase.
- Větráním rekuperační jednotkou s použitím tepelného výměníku. Ten umožňuje v zimě ohřívat vyfukovaným teplým vzduchem zevnitř nasávaný chladný vzduch zvenčí, kdy účinnost výměníku je až 90%. [10] [12]
- Ohřevem vyměňovaného vzduchu při větrání, je-li teplota v domě pod nastaveným minimem.
- Vytápěním za pomoci recirkulace ohřívaného vzduchu po domě v módu vnitřní recirkulace zajišťované rekuperační jednotkou - teplovzdušným vytápěním
- Pro případ budoucí potřeby je k dispozici i další externí signál např. externí vytápění.

Toto zařízení je navrženo s premisou, že letní a zimní mód nejsou určeny pouze pro roční období z jejich názvů plynoucí. Tyto módy reprezentují hlavně požadavek, zda se má dům udržovat chladný, nebo se má naopak

udržovat co nejteplejší s co nejmenšími nároky na vytápění.

Je-li v létě chladné období a požaduje se naopak topit, po přepnutí do zimního módu se tak bez jakékoliv újmy na funkčnosti stane. Toto platí samozřejmě i pro zbylá roční období.

Dospěl jsem k názoru, že toto základní rozhodování by mělo být ponecháno na člověku. Neboť zařízení nezná aktuální předpověď počasí a nezná změny individuálních teplotních preferencí členů domácnosti.

K vytápění je ještě vhodné podotknout, že topné těleso je v rekuperační jednotce pouze jedno. Dva stavy vytápění se liší přepnutím režimu rekuperační jednotky mezi větráním vzduchu z venčí a recirkulací v rámci budovy. [10] Ve spojitosti s topným tělesem tedy realizujeme funkci vytápění bez větrání z vnějšku (dále topení a recirkulace), nebo ohřevu vzduchu z vnějšku (dále větrání a topení). Dále však budeme tyto stavy rozlišovat podle jejich funkce, nikoliv prvku, jež je realizuje.

V následujících tabulkách je uveden přehled uvažovaných stavů systému a odpovídajících výstupů, které v takovém stavu požadujeme. Zde bych rád podotkl, že stavy systému se v tomto případě myslí množina možných vstupů a výstupů. Stavový automat v tomto bloku totiž nenalezneme, jedná se o sekvenční logiku. V těchto tabulkách vycházejme z předpokladu, že přepínač výstupů systému je v poloze zapnuto. Tedy že uživatel nastavil, že chce, aby systém nejen aktualizoval svá data a zobrazoval aktuálně měřené teploty, ale aktivně řídil rekuperační jednotku.

První možností je, že přepínač výstupů systému je v poloze vypnuto, pak je požadovaným výstupem zobrazování aktuálně měřených teplot a aktualizování záznamů průměrných teplot bez jejich okamžitého dalšího použití.

Nyní se zaměříme na případ, že přepínač výstupů systémů je v poloze zapnuto. První základní možností je případ, kdy uživatel nastavil přepínač volby ročního období na zimu. Zde je základní filozofií systému vyhřát dům co nejvíce bez nutnosti topení.

Tabulka 3.1, 3.2 a 3.3 by se dala shrnout následujícím způsobem. Je-li požadavek na vyvětrání toalety nebo větrání v kuchyni, toto větrání musí za každého počasí proběhnout. S výměníkem či bez, ale nelze nevyvětrat z důvodu příliš extrémních teplot venku. Mezi těmito dvěma spínači je rozdíl v tom, že větrání WC je vždy na 3 minuty, kdežto větrání kuchyně probíhá dokud uživatel nepřepne spínač zpět do polohy log. 0.

Dále záleží zejména na tom, jestli nastalo časové okno, které bylo vyhodnoceno jako nejvhodnější pro větrání – v zimě se jedná o nejteplejší část dne. Nastane-li vzácný případ, že je venku tepleji než uvnitř, větrá se bez výměníku, aby se dům bez nutnosti vytápění prohřál. Dále se vyhodnocuje, zda je teplotní rozdíl mezi vnitřkem a vnějškem domu větší či menší než 30°C. Je-li menší, větrá se s výměníkem. Je-li větší, pravidelné větrání neproběhne. Zde je důležité si uvědomit, že se jedná o maximální teplotu dne. Ačkoliv se teplotní rozdíl nezdá velký, tak v případě požadované vnitřní teploty 22°C by denní maximum muselo být -8°C a menší. Jako referenční požadovaná teplota je vstupem denní či noční (horní/dolní) uživatelsky nastavená teplota. Jak již z požadavků vyplývá, maximum nastává ve dne a porovnání s noční

teplotou se tak jeví nepotřebné. Nicméně, z důvodů zajištění stability systému a možných výjimek bylo ponecháno.

Vstupní podmínky	Požadovaný výstup
Byl zapnut spínač větrání v kuchyni, venku je tepleji než uvnitř.	Větrá se bez výměníku.
Byl zapnut spínač větrání v kuchyni. Je den, venku je chladněji o libovolný počet °C, a uvnitř je více než nastavená denní teplota.	Větrá se s výměníkem.
Byl zapnut spínač větrání v kuchyni. Je den, venku je chladněji o libovolný počet °C a uvnitř je méně než nastavená denní teplota.	Větrá se s výměníkem a topí se.
Byl zapnut spínač větrání v kuchyni. Je noc, venku je chladněji o libovolný počet °C a uvnitř je více, než nastavená noční teplota.	Větrá se s výměníkem.
Byl zapnut spínač větrání v kuchyni. Je noc, venku je chladněji o libovolný počet °C a uvnitř je méně než nastavená noční teplota.	Větrá se s výměníkem a topí se.
Bylo stisknuto tlačítko větrání na toaletě. Ještě neuběhly 3 minuty, venku je tepleji než uvnitř, větrej bez výměníku.	Větrá se bez výměníku.
Bylo stisknuto tlačítko větrání na toaletě. Ještě neuběhly 3 minuty, je den, venku je chladněji o libovolný počet °C a uvnitř je více, než nastavená denní teplota.	Větrá se s výměníkem.

Tabulka 3.1: Zimní stavy - část 1

Vstupní podmínky	Požadovaný výstup
Bylo stisknuto tlačítko větrání na toaletě. Ještě neuběhly 3 minuty, je den, a když je venku chladněji o libovolný počet °C a uvnitř je méně, než nastavená denní teplota.	Větrá se s výměníkem a topí se.
Bylo stisknuto tlačítko větrání na toaletě. Ještě neuběhly 3 minuty, je noc, a když je venku chladněji o libovolný počet °C a uvnitř je více, než nastavená noční teplota.	Větrá se s výměníkem.
Bylo stisknuto tlačítko větrání na toaletě. Ještě neuběhly 3 minuty, je noc, a když je venku chladněji o libovolný počet °C a uvnitř je méně, než nastavená noční teplota.	Větrá se s výměníkem a topí se.
Nebylo ručně spuštěno větrání, venku je tepleji než uvnitř.	Větrá se bez výměníku.
Nastala denní doba s nejvyšší průměrnou teplotou. Venku je chladněji než uvnitř, teplotní rozdíl je menší než 30°C, uvnitř je více, než nastavená denní/noční teplota.	Větrá se s výměníkem.
Nastala denní doba s nejvyšší průměrnou teplotou. Venku je chladněji než uvnitř, teplotní rozdíl je menší než 30°C, uvnitř je méně, než nastavená denní/noční teplota.	Větrá se s výměníkem a topí se.

Tabulka 3.2: Zimní stavy - část 2

Vstupní podmínky	Požadovaný výstup
Nastala denní doba s nejvyšší průměrnou teplotou. Venku je chladněji než uvnitř, teplotní rozdíl je větší než 30°C, uvnitř je více, než nastavená denní/noční teplota.	Nevětrá se a netopí se.
Nastala denní doba s nejvyšší průměrnou teplotou. Venku je chladněji než uvnitř, teplotní rozdíl je větší než 30°C, uvnitř je méně, než nastavená denní/noční teplota.	Vnitřní recirkulace a topí se.
Je den, nenastala denní doba s nejvyšší průměrnou teplotou. uvnitř je více, než nastavená denní teplota.	Nevětrá se a netopí se.
Je den, nenastala denní doba s nejvyšší průměrnou teplotou. uvnitř je méně, než nastavená denní teplota.	Vnitřní recirkulace a topí se.
Je noc, nenastala denní doba s nejvyšší průměrnou teplotou. uvnitř je více, než nastavená noční teplota.	Nevětrá se a netopí se.
Je noc, nenastala denní doba s nejvyšší průměrnou teplotou. uvnitř je méně, než nastavená noční teplota.	Vnitřní recirkulace a topí se.

Tabulka 3.3: Zimní stavy - část 3

Nyní analogická tabulka pro přepínač letního období v poloze léto. Zde je základní filozofií udržet dům co nejvíce v chladu v parném létě. Větrání toalety a kuchyně funguje zcela analogickým způsobem jako v zimě. Naopak odlišností je, že optimální dobou pro větrání je doba nad ránem, kdy je teplota obvykle nejnižší. Pro letní režim se změnila funkce uživatelsky nastavovaných teplot. Již nejsou odlišné teploty pro den a noc. Na rozdíl od zimy totiž nemáme možnost hýbat s teplotou v kteroukoliv denní dobu oběma směry. Tato funkce přísluší klimatizaci, kterou v tomto modelu neuvažujeme. Máme tedy pouze jednu teplotu, která reprezentuje hodnotu, pod níž není žádoucí, aby teplota v domě poklesla. Druhá teplota je místo toho určena k nastavení venkovní teploty, při níž se již větrání – byť s výměníkem – považuje za neefektivní.

Je-li venkovní teplota níže, než nastavená nejnižší akceptovatelná, větrá se s výměníkem. Je-li však v domě tepleji, než tato nejnižší hodnota a venku je chladno, využije se tato situace k ochlazení interiéru. Je-li však venku v nejchladnější denní době tepleji, větrá se s výměníkem. Stoupne-li však teplota příliš nad uživatelsky nastavitelné maximum, větrání neproběhne. Rozsahy nastavení maxima je, jako u všech uživatelsky nastavitelných teplot, 5 až 30°C. Je tedy více než dostačující, uvědomíme-li si, že oněch 30°C by musela být nejnižší teplota během 24h, a je tedy poměrně extrémní.

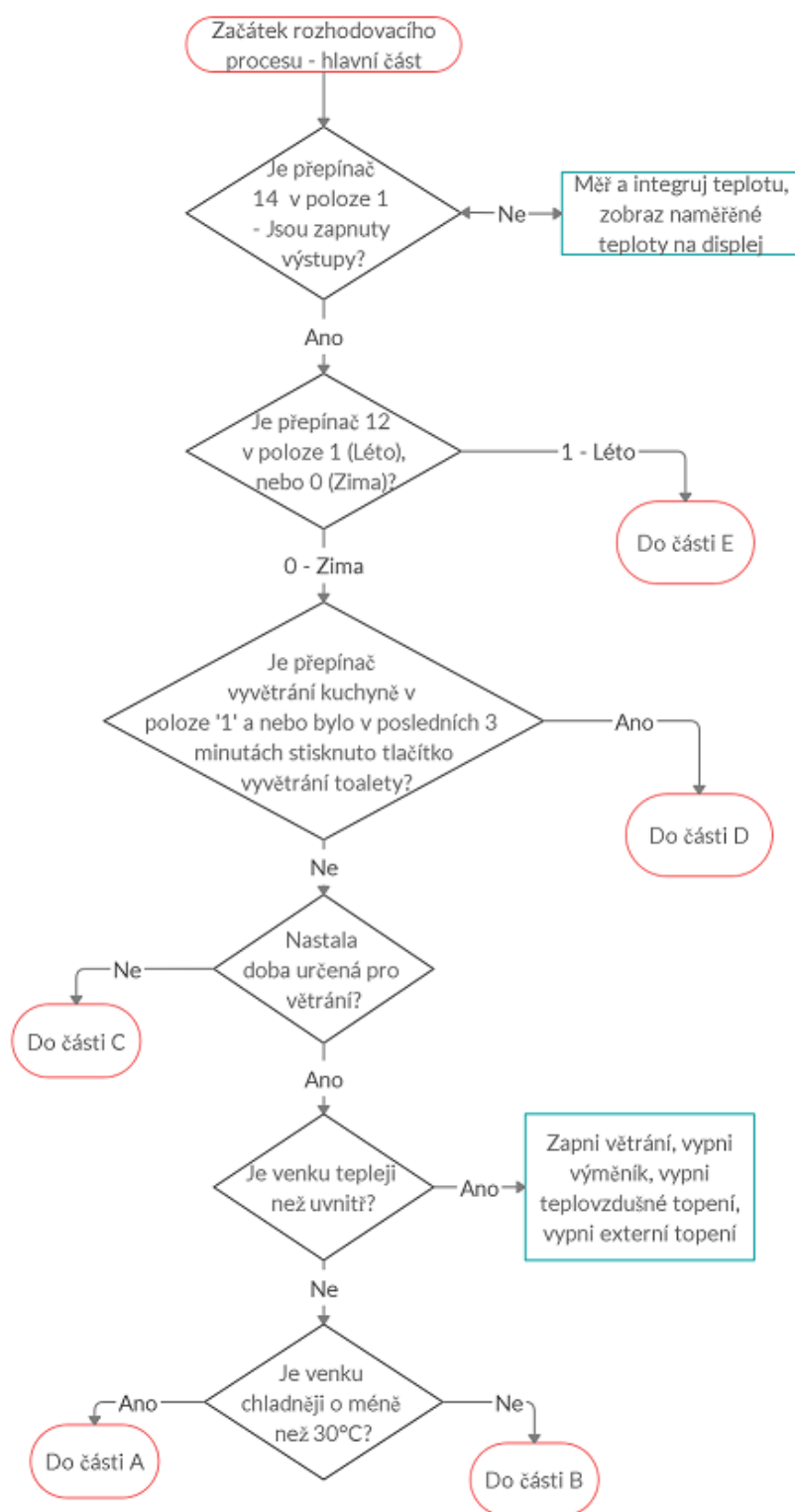
Vstupní podmínky	Požadovaný výstup
Byl zapnut spínač větrání v kuchyni, venku je tepleji než uvnitř.	Větrá se bez výměníku.
Byl zapnut spínač větrání v kuchyni. Je den, venku je chladněji a uvnitř je více, než nastavená denní teplota.	Větrá se s výměníkem.
Byl zapnut spínač větrání v kuchyni. Je den, venku je chladněji a uvnitř je méně, než nastavená denní teplota.	Větrá se s výměníkem.
Byl zapnut spínač větrání v kuchyni. Je noc, venku je chladněji a uvnitř je více, než nastavená noční teplota.	Větrá se s výměníkem.
Byl zapnut spínač větrání v kuchyni. Je noc, venku je chladněji a uvnitř je méně, než nastavená noční teplota.	Větrá se s výměníkem.
Bylo stisknuto tlačítko větrání na toaletě. Ještě neuběhly 3 minuty, venku je tepleji než uvnitř, větrej bez výměníku.	Větrá se bez výměníku.
Bylo stisknuto tlačítko větrání na toaletě. Ještě neuběhly 3 minuty, je den, venku je chladněji, uvnitř je více, než nastavená denní teplota.	Větrá se s výměníkem.
Bylo stisknuto tlačítko větrání na toaletě. Ještě neuběhly 3 minuty, je den, venku je chladněji, uvnitř je méně, než nastavená denní teplota.	Větrá se s výměníkem.
Bylo stisknuto tlačítko větrání na toaletě. Ještě neuběhly 3 minuty, je noc, venku je chladněji, uvnitř je více, než nastavená noční teplota.	Větrá se s výměníkem.

Tabulka 3.4: Letní stavy - část 1

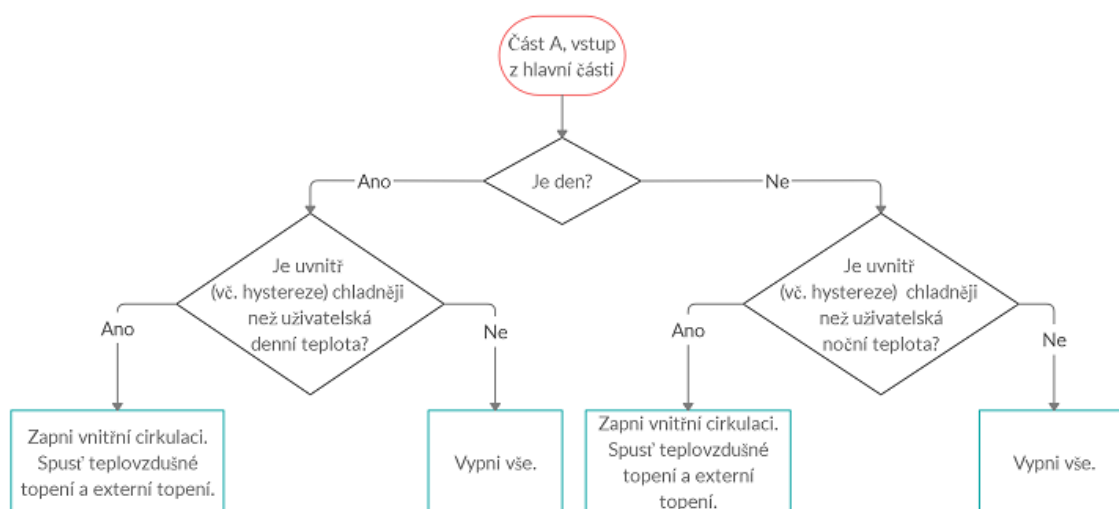
Vstupní podmínky	Požadovaný výstup
Bylo stisknuto tlačítko větrání na toaletě. Ještě neuběhly 3 minuty, je noc, venku je chladněji, uvnitř je méně, než nastavená noční teplota.	Větrá se s výměníkem.
Nastala denní doba s nejnižší průměrnou teplotou. Venku je chladněji než uvnitř a uvnitř je více, než nastavená minimální teplota.	Větrá se bez výměníku.
Nastala denní doba s nejnižší průměrnou teplotou. Venku je chladněji než uvnitř a uvnitř je méně, než nastavená minimální teplota.	Větrá se s výměníkem.
Nastala denní doba s nejnižší průměrnou teplotou. Venku je tepleji než uvnitř, venku je méně, než horní nastavená teplota a uvnitř je více, než dolní nastavená teplota.	Větrá se s výměníkem.
Nastala denní doba s nejnižší průměrnou teplotou. Venku je tepleji než uvnitř, venku je méně, než horní nastavená teplota a uvnitř je méně, než dolní nastavená teplota.	Větrá se s výměníkem.
Nastala denní doba s nejnižší průměrnou teplotou. Venku je tepleji než uvnitř, venku je více, než horní nastavená teplota a uvnitř je více, než dolní nastavená teplota.	Nevětrá se.
Nastala denní doba s nejnižší průměrnou teplotou. Venku je tepleji než uvnitř, venku je více, než horní nastavená teplota a uvnitř je méně, než dolní nastavená teplota.	Nevětrá se.

Tabulka 3.5: Letní stavy - část 2

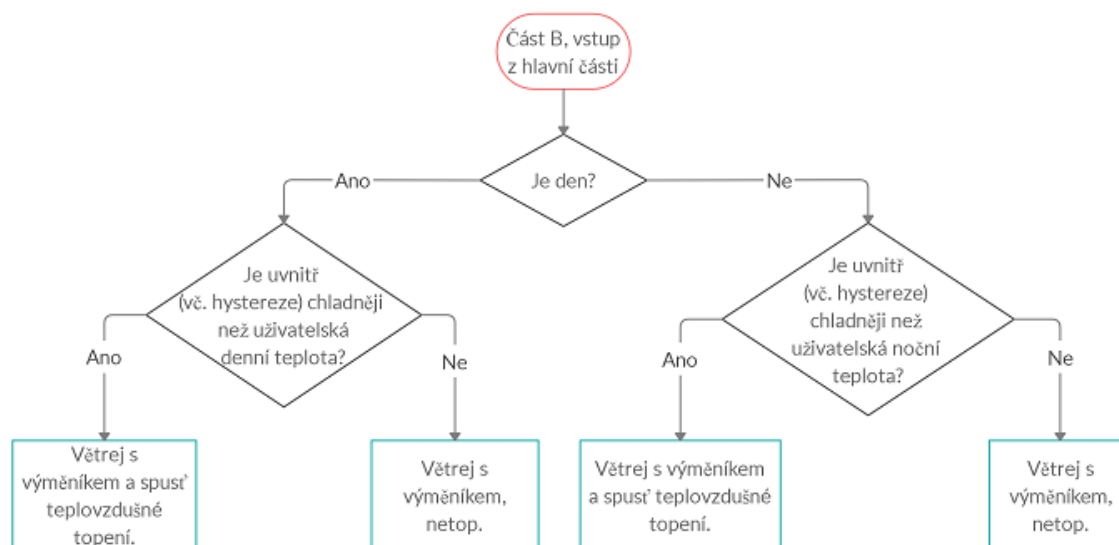
Nyní si toto vše osvětlíme ještě graficky. Diagram vyhodnocovací logiky je pro svou rozsáhlost rozdělen do šesti částí: Hlavní část (obrázek 3.1), blok A (obrázek 3.2), blok B (obrázek 3.3), blok C (obrázek 3.4), blok D (obrázek 3.5) a blok E (obrázek 3.6). Toto označení je pouze pro účely rozdělení grafu do více obrázků, nemá konsekvence v dalším textu.



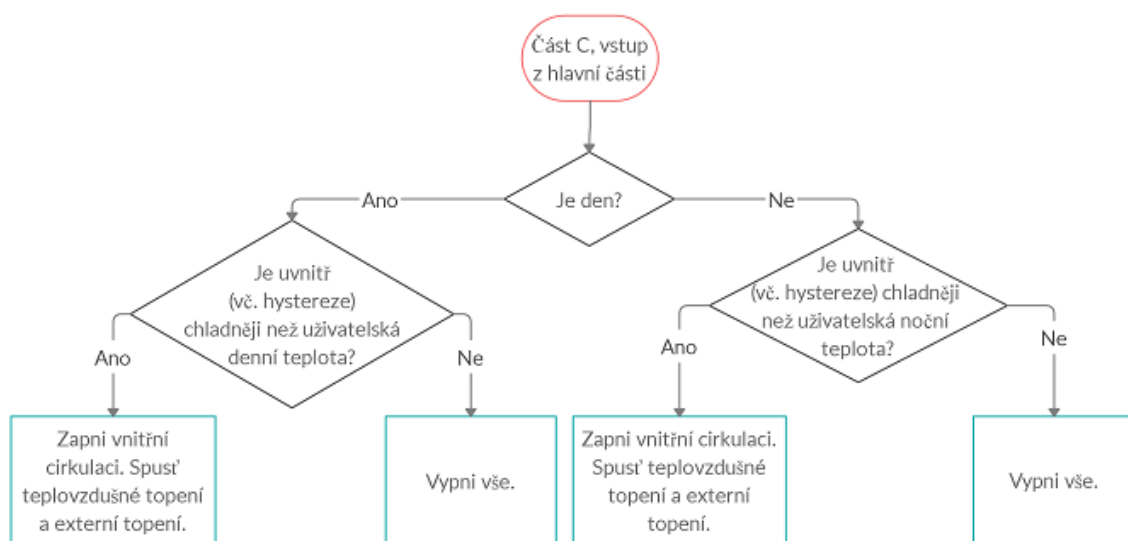
Obrázek 3.1: Hlavní část



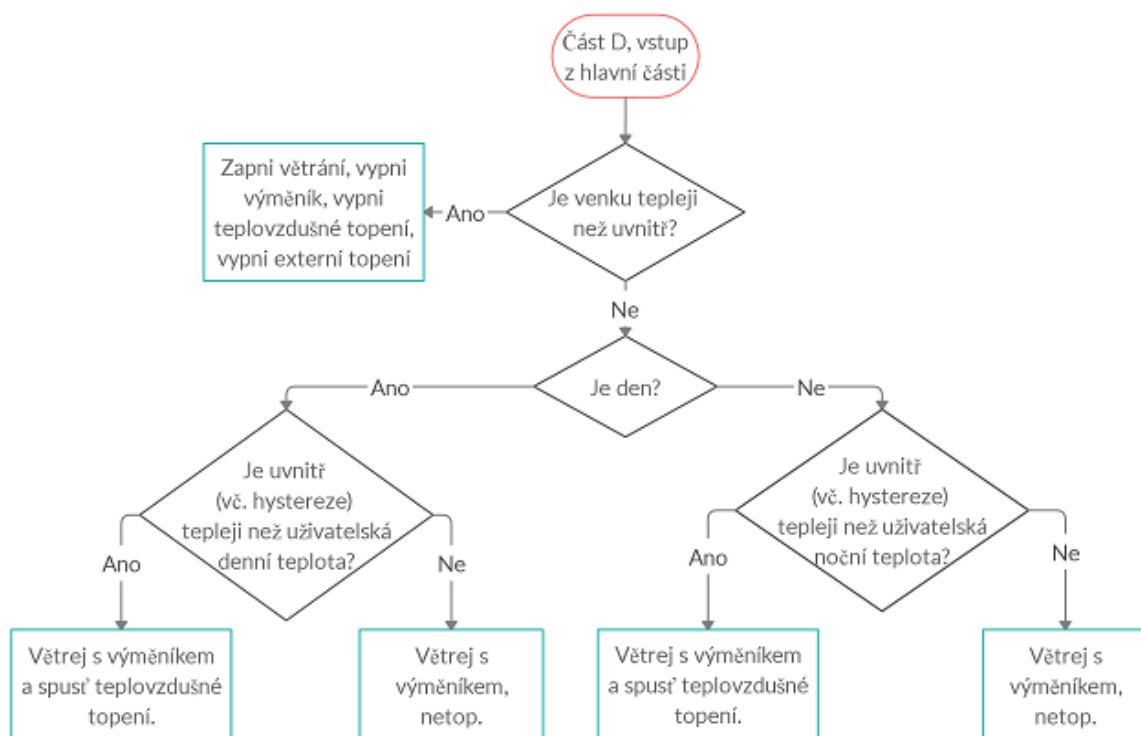
Obrázek 3.2: Blok A



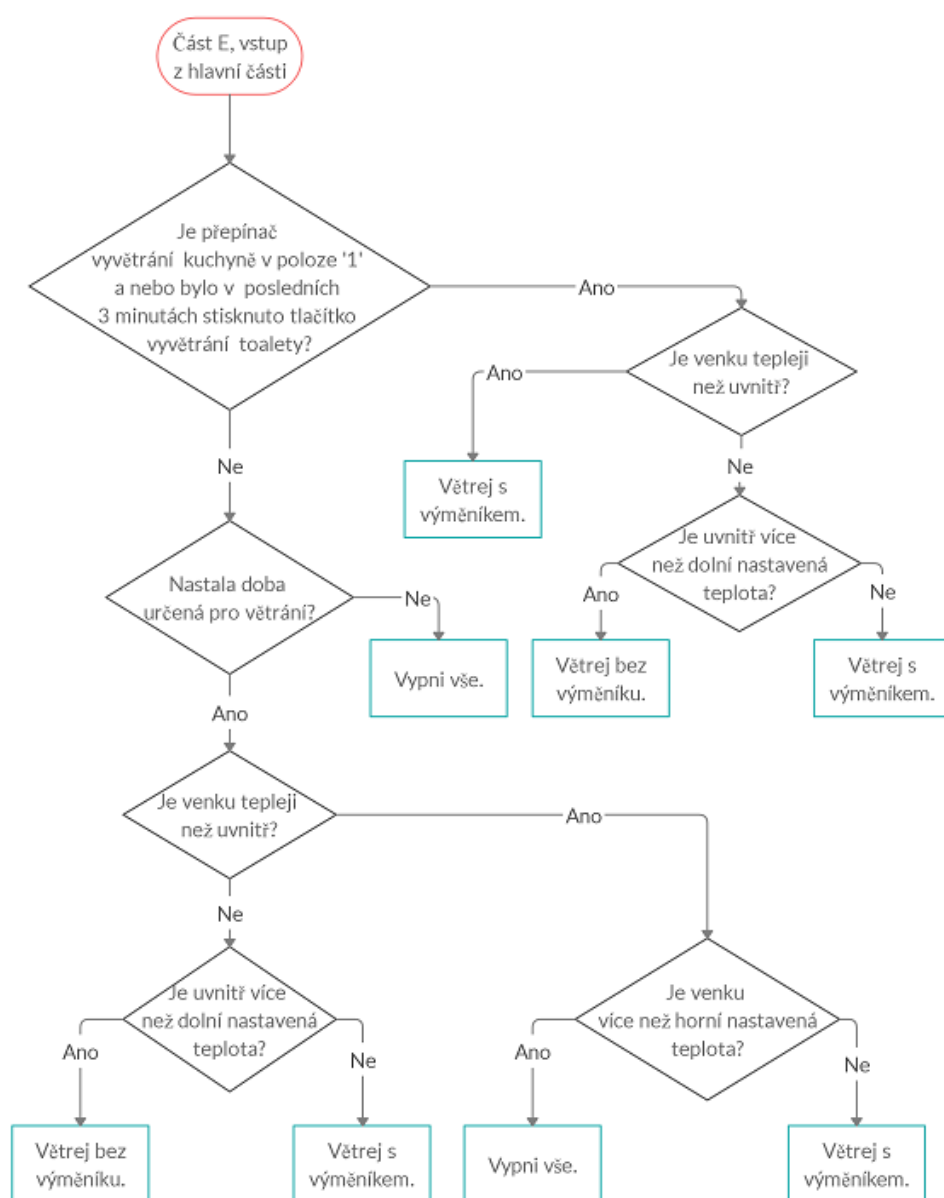
Obrázek 3.3: Blok B



Obrázek 3.4: Blok C



Obrázek 3.5: Blok D



Obrázek 3.6: Blok E

V uvedených stavech (letních i zimních) jsme pracovali s maximální a minimální teplotou během dne. Jak ji však získáme? Základní úloha je zjevná. Typicky v zimě nestačí jen porovnávat teploty uvnitř a vně domu. Zařízení musí rozpoznat, kdy je venku nejvyšší teplota nebo alespoň časové okolo ní. Zařízení nemůže pracovat jen s okamžitými teplotami, musí průměrovat naměřené venkovní teploty za určité časové období. Tento průměr teplot musí porovnat s okamžitou hodnotou venkovní a vnitřní teploty. Teprve se znalostí těchto tří údajů, dvou přímých a jednoho statistického, může sekvenční logika spínat rekuperační jednotku.

Další úlohu k vyřešení představují náhlé změny venkovních meteorologických podmínek, kdy obvykle nejteplejší část dne nemusí být aktuálně ta nejteplejší. I s tímto by se mělo průměrování teplot vyrovnat, avšak pouze za předpokladu, že bude probíhat za dostatečně dlouhé časové období, např. jednoho týdne. Uvedená úloha byla řešena následujícím způsobem: Jako perioda pro výpočet průměru byla zvolena doba jednoho týdne. Ten byl rozdělen na 672 čtvrt hodin. Časové rozlišení o délce jedné hodiny by bylo vzhledem k délce větracího intervalu (2 hodiny) příliš málo a na efektivitě větrání by se již podepsalo. Délka jedné čtvrt hodiny se jeví jako rozumný kompromis mezi tímto požadavkem a hardwarovými nároky.

Navzorkovaná teplota z každé čtvrt hodiny je výsledkem čtyř teplot změřených v rozmezí 3 minut a následně zprůměrovaných, aby se omezily náhodné výkyvy a teplota byla v rámci možností odpovídající.

Po uplynutí doby jednoho týdne se spustí druhý proces průměrování. Ten zprůměruje vždy sedm teplot dle vzorce (t je teplota čtvrt hodiny ($95 * n$), postupně procházíme celý týden):

$$\sum_{n=0}^6 t(95 * n)$$

Výsledkem je tedy průměr pro každou čtvrt hodinu dne. Do nového datového pole o 96 položkách tyto průměry uložíme a až se tak stane, spustí se proces hledání nejvyšší a nejnižší teploty. Výstupem jsou čtvrt hodiny s nejvyšší a nejnižší teplotou reprezentované integerem, nabývajícím hodnot 0 až 95. Zároveň je výstupem aktuální čtvrt hodina dne, v níž se nacházíme. Tyto údaje jsou již dostatečné pro to, abychom věděli, za jak dlouho požadovaná extrémní teplota dne nastane. Vypočítáme tedy dvouhodinové časové okno kolem této teploty a budeme již jen hlídat, zda se aktuální čtvrt hodina shoduje s počáteční či koncovou čtvrt hodinou tohoto okna. Jakmile zjistíme, že jsme v časovém okně pro provedení větracích úkonů, spustí se příslušná sekvenční logika. Ta podle výše uvedených tabulek vyhodnotí, jaké zařízení a v jakém režimu se mají spustit

Druhý stavový automat, a zároveň druhá velká funkční část, bude potřeba pro ošetření komunikačního protokolu s čidlem DS18B20. Tato část je zvláště citlivá na přesné načasování požadovaných délek trvání napěťových úrovní v protokolu. Mimo to tento automat musí zabezpečit sled instrukcí na vyčtení dat z čidla (viz Komunikační protokol na úrovni bitů–instrukcí). Tato část také zabezpečuje 9 bajtový vstupní a výstupní buffer pro odesílání dat do čidla a příjem dat z něj. Tato velikost je dána velikostí paměti SCRATCHPAD v čidle [1].

Třetí částí je zobrazení vnitřní a venkovní teploty na sedmissegmentový displej. Tato část zahrnuje převedení naměřené teploty na BCD kód [3]. Následuje převedení čísl v BCD kódu na jí odpovídající znak na displeji, načtení do výstupního bufferu a nakonec zmultiplexování všech cifer a znaků na digity displeje.

Poslední, ale neméně důležitou částí je ošetření vstupních tlačítek a spínačů proti zákmitům. Daný ovládací prvek se vzorkuje každých 50ms, tři poslední shodné stavy odlišné od aktuálně platné hodnoty se chápou jako změna stavu ovládacího prvku.

Úloha tedy sestává z následujících velkých funkčních celků:

- Ošetření vstupů
- Vyčtení teploty z čidel
- Zobrazení teplot na displej
- Průměrování teploty za dané časové období
- Hlavní řídicí stavový automat

■ 3.2 Vlastní řešení - popis jednotlivých bloků systému

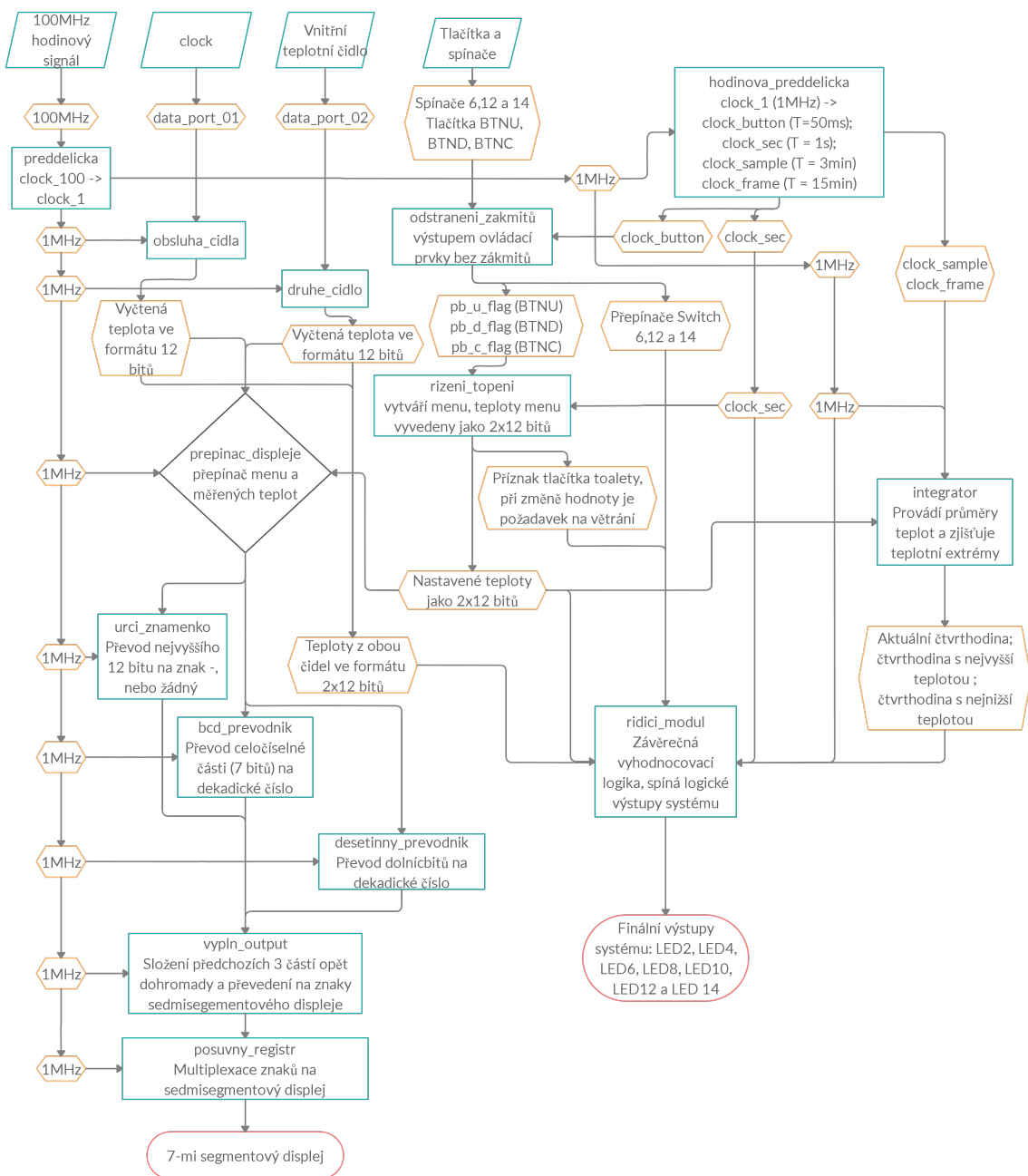
■ 3.2.1 Bloky systému

Zobrazovací část sestává z následujících dílčích částí:

- Předdělička hodinového signálu - `preddelicka`
- Blok vyčtení znaménka teploty v °C - `urci_znamenko`
- Převodník signálu z čidla do BCD formátu - `bcd_prevednik`
- Převodník neceločíselné části naměřené teploty na BCD formát - `desetinny_prevednik`
- Převodník BCD kódu na příslušný znak na displeji - `vypln_output`
- Posuvný registr pro multiplexování znaků - `posuvny_registr`
- Bloky obsluhy komunikace s čidlem po sběrnici One-Wire - `obsluha_cidla`, `druhe_cidlo`
- Blok hodinové předděličky - `hodinova_preddelicka`
- Blok integrátoru - `integrator`
- Blok odstranění zákmitů - `odstraneni_zakmitu`

- Blok řízení topení - `rizeni_topeni`
- Blok Přepínač displeje - `prepinac_displeje`
- Blok řídicího modulu - `ridici_modul`

Na obrázku 3.7 pak vidíme blokové schéma systému. Kosodélníkové zelené bloky jsou vstupy systému. Obdélníkové zelené bloky reprezentují bloky kódu (název vždy uveden). Žluté bloky jsou popisem, jaká data se v daném propoji přenáší. Oblé červené bloky jsou výstupy systému.



Obrázek 3.7: Blokové schéma systému

■ Předdělička hodinového signálu

Úkolem této části je vydělit vstupní signál 100MHz z interního zdroje Nexysu 4 [2] koeficientem 100 pro získání frekvence 1MHz. Vstupní signálem je 100MHz v signálu `clock_100` typu `STD_LOGIC`. Výstupním je 1MHz, který je pak v hlavním bloku kódu (dále krátce jen `main`) uložen do signálu `clock_1` typu `STD_LOGIC`.

■ Blok vyčtení znaménka teploty

Začínáme první velkou funkční částí – zobrazovací jednotkou. Vstupem jsou proměnné `teplota_vydelena_01` a `teplota_vydelena_02` jako vstupy bitu s údaji o znaménku teploty z čidel. Z těchto vstupů jsou vyčtena znaménka teploty v celsiově stupnici. Toto je pak posláno na výstup do proměnné `teplota_vydelena_01` a `teplota_vydelena_02` v `mainu` ve formátu znaku mínus (-) sedmisegmentového displeje v případě teplot pod bodem mrazu, případně prázdného znaku, je-li teplota nad bodem mrazu. Tato je v `mainu` předána do posuvného registru na zobrazení.

■ Převodník signálu z čidla do BCD formátu

Úkolem této části je vyčíst ze signálu z čidla celočíselnou část a převést ji do 12 bitového BCD kódu po 3 čtveřících bitů, kde každá čtveřice nabývá binárních hodnot 0000 až 1001 (0 až 9), tedy výstupem je trojčíslné dekadické číslo reprezentující hodnoty 0 - 127°C, které pak předává přes 12bitový signál `cela_bcd_01` pro první čidlo (venkovní teplota) a `cela_bcd_02` pro druhé čidlo (vnitřní teplota).

Samotný převod probíhá Hornerovým schématem [3]. Tento algoritmus je založen na opakovaných bitových posunech doleva. Bitový posun doleva je fakticky násobení dvěma. Při každé iteraci, kdy se vydělená celočíselná část z čidla postupně posunuje zprava do požadovaných tří čtveřic bitů, je pro každou tuto čtveřici kontrolováno, zdali je číslo v ní větší rovno pěti. Jestliže ano, při posunu (a tedy násobení dvěma) by byl dekadický výsledek 10 či vyšší a přetekl by tak řád. K číslu se tedy přičte konstanta 3. Tato konstanta se po vynásobení dvěma při bitovém posunu zvětší na 6 a dojde tak k přeskočení zbylých šesti číslic v hexadecimální soustavě do vyššího řádu. Hexadecimální z toho důvodu, že každá cifra je reprezentována čtveřicí bitů, a její maximální hodnota tedy nabývá 15-ti, kde ale vyšší čísla než 9

nejsou pro BCD kód žádoucí. Po přičtení 3 (6 po posunu) se tedy číslo 5 a vyšší (10 a vyšší po posunu) dostane i v hexadecimálním tvaru o řád výše do levé sousední čtveřice a zbylé jednotky zůstanou v původní čtveřici. Tak je postupně dosaženo požadovaného stavu tří cifer a ty jsou posléze poslány do signálu `cela_bcd_01` pro vnější a `cela_bcd_02` pro vnitřní teplotu. A nakonec v bitech 23 – 16 bitů (vnější teplota) a 11 – 4 (vnitřní teplota) signálu `teplota_24bit` nachystány na převod na znaky.

■ Převodník neceločíselné části naměřené teploty na BCD formát

Úkolem této části je převzít neceločíselnou část vstupu z čidla a převést ji do BCD kódu. Převedenou hodnotu ve formě jednoho desetinného místa pak uložit do signálu `desetinna_bcd_01` (vnější teplota) a `desetinna_bcd_02` (vnitřní teplota) odkud pak putují do bloku Převodník BCD kódu na příslušný znak na displeji. Desetinná část vstupuje ve formátu 4 bitů, kde MSB reprezentuje poloviny stupně. Dále v sestupném pořadí čtvrtiny, osminy a šestnáctiny. Vzhledem k řešení úlože postačuje přesnost na desetiny stupně.

Samotný převod by bylo možné řešit pomocí násobení bitů vstupu jeden po druhém příslušným koeficientem a tyto dílčí výsledky sčítat. Avšak vzhledem k tomu, že možností vstupu je pouze 16, je méně výpočetně i hardwarově náročné získat výsledek převodní tabulkou.

Teploty pod bodem mrazu vyžadují samostatnou převodní tabulku z důvodu dvojkového doplňku. Tyto operace jsou zvlášť vykonány pro teplotu z obou dvou čidel.

■ Převodník BCD kódu na příslušný znak na displeji

Úkolem této části je použít výstupy výše zmíněných částí a převést je na formát znaku na displeji sedmisedimentového typu. Vstupem je z mainu signál `cela_bcd_01` a `cela_bcd_02` (venkovní/vnitřní teplota) s celočíselnou částí, dále `desetinna_bcd_01` a `desetinna_bcd_02` (venkovní/vnitřní teplota) s desetinnou částí a nakonec `znamenko_8digit` s již hotovým znakem na displej, tato poslední se jen přkopíruje na výstup. Je-li na vstupu signál reprezentující zápornou teplotu, je nutno tento signál znegovat a tím jej převést z dvojkového doplňku na kladné číslo odpovídající jeho absolutní hodnotě. Přiřazení znaménka mínus na displej v případě záporného čísla pak zajišťuje blok vyčtení znaménka teploty. Výsledek se uloží do 64 bitového signálu `bity_output` v mainu. V tomto signálu každá osmice bitů reprezentuje jednu osmicí segmentů na displeji, může tedy využít plný rozsah nabízený kitem.

Ten je také využit, 2x 4 digity pro teplotu ze dvou čidel, každá ve formátu 1 digit znaménko, 2 digity celočíselná část a 1 digit desetinná.

■ Posuvný registr pro multiplexování znaků

Úkolem této části je znaky, vytvořené v převodníku BCD na znaky, zobrazit na displeji. Vzhledem k faktu, že Nexys umí zobrazit znaky pouze multiplexem [2], je nutné v této části multiplex zajistit.

Vstupem je tedy signál `bits_output` přejímající data z převodníku na znaky. Prvním výstupem je `zobraz_cislo_port`, který posílá na výstup znak, který se má zobrazit. Signál je osmibitový, vzhledem k faktu, že jeden digit má 8 LED diod [2]. Druhým je signál `vyber_digit_port` vybírající příslušný digit, na který se má znak zobrazit. Digitů je taktéž 8 [2], takže i tento signál je osmibitový.

Aby nedocházelo k zrcadlení znaku na sousední digit, je nutné ponechat po dobu 1 taktu hodinového signálu všechny digity zhasnuté. Střída pohasnutí je zvolna jako 1%, takže ztmavnutí displeje je naprosto marginální.

■ Bloky obsluhy komunikace s čidlem po sběrnici One-Wire

Nyní se dostáváme do druhé velké funkční části – měření teploty jako zdroje dat pro systém. Obsluha čidel samotných je řešena dvěma stejnými bloky z důvodu přehlednosti kódu, dále umožňuje dvojnásobně vyšší frekvenci vyčítání teplot (možnost dalšího budoucího vývoje) a v neposlední řadě z důvodu omezení jazyka VHDL, kdy ošetření přístupu k jedné sběrnice proměnné dvěma nezávislými procesy by bylo značně komplikované. VHDL totiž nepřipouští možnost ovlivňování jedné proměnné dvěma nezávislými stavy[7], je nutno řešit např. dalším procesem provádějícím OR těchto dvou vstupů. Neumožňuje tedy například i dvěma procesům nastavit a vynulovat stejný příznakový bit, atd. Tento blok je řešen jako stavový automat sestávající z následujících stavů:

- `ini_state` - Vykonává inicializační proceduru – Resetovací puls, případně puls detekce přítomnosti zařízení. [1] Tato procedura se vykonává po spuštění/restartu zařízení a po dokončení sekvence Inicializace – ROM Příkaz – Funkční příkaz [1], kde plní právě úlohu první části – inicializace. Spouští se tedy (krom mimořádných situací) po zápisu příkazu CONVERT TEMPERATURE a po vyčtení dat z čidla po příkazu READ SCRATCHPAD a ověření CRC kódu. Po úspěšném provedení inicializace

přechází vždy do stavu `skip_rom`, není-li inicializace úspěšná, vykoná se znova.

- `skip_rom` - Jediný zde použitý ROM příkaz adresující všechny zařízení na sběrnici. [1] Vzhledem k přítomnosti pouze jednoho zařízení na sběrnici je tento plně postačující. Stav tedy provede zápis adresy instrukce [CCh, 11001100] do signálu `instruction` předávající tato data k zápisu na sběrnici a provede změnu stavu na stav `branch_write`. Do signálu `after_write` uloží stav, na který se má přejít po vykonání zápisu instrukce.
- `branch_write` - Tento stav řídí proces zapisování instrukce na sběrnici. Prochází bit po bitu z předávacího signálu `instruction` a volá stavy `write_0_first`, je-li zapisovaným bitem logická nula a `write_1_first`, je-li jím logická 1. Jsou-li zapsány všechny bity, dojde ke změně stavu na stav, uložený v signálu `after_write`.
- `write_0_first` - Vykoná první část write slotu, stažení sběrnice do log. 0 po dobu 80 μ s. [1]
- `write_0_second` - Opětovné uvolnění sběrnice po danou dobu 10 μ s. [1]
- `write_1_first`, `write_1_second` - Analogicky k předešlým, pouze časové úseky se liší (4 μ s;80 μ s).
- `convert_temperature` - stav pro vyslání příkazu CONVERT TEMPERATURE [44h], opět vloží do předávacího signálu adresu a změní stav na `scratchpad_ready`.
- `scratchpad_ready` - Tento stav zajišťuje čekání po dobu 800 μ s. 750 μ s je dle datasheetu [1] typická doba změření a zpracování teploty čidla DS18B20 pro 12bitové rozlišení. K této době je ještě připočtena rezerva 50 μ s. Během této doby změní adresu instrukce v předávacím signálu na READ SCRATCHPAD [BEh] a po jejím uplynutí změní stav na `read_scratchpad`.
- `read_scratchpad` - Změní adresu instrukce v předávacím signálu na READ SCARTCHAD [BEh] a po uplynutí této doby změní stav na `branch_write` pro vykonání zápisu adresy instrukce na sběrnici. Zároveň nastaví do signálu `after_write` hodnotu `branch_read` coby příští stav, aby se po zápisu instrukce spustil Read-time sloty pro vyčtení dat z čidla. [1]
- `branch_read` - Tento stav řídí proces vyčítání všech 72 bitů paměti SCRATCHPAD [1] z čidla do desky Nexys. Postupně volá stav `read_data_first` pro každý jednotlivý bit a po vyčtení všech bitů nastaví jako příští stav `crc`. Volá stav `read_data_first`.

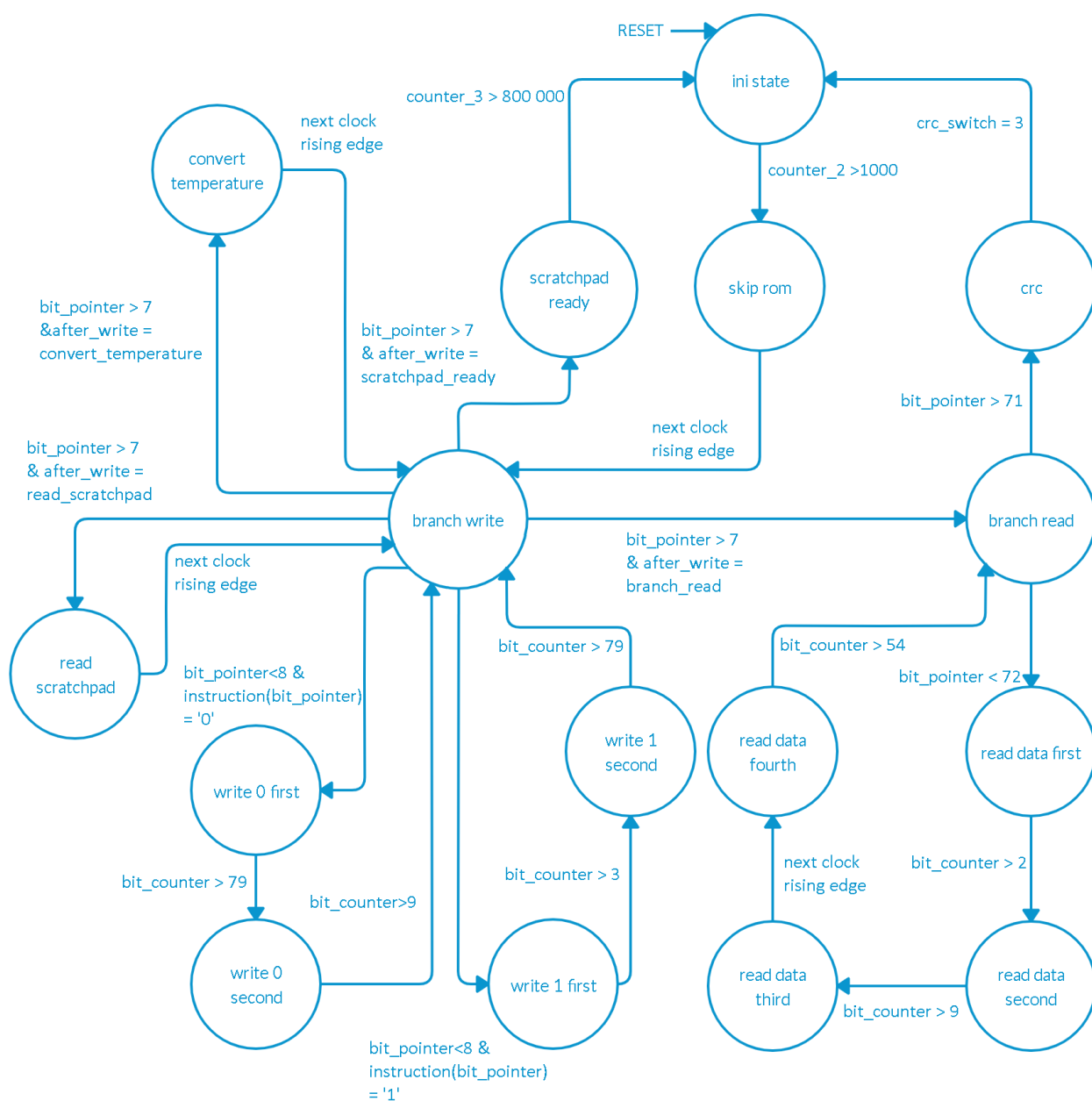
Ten tvoří spolu se stavy `read_data_second`, `read_data_third` a `read_data_fourth` jednu na sebe navazující sekvenci.

- `read_data_first` - Podrží po 3 μs sběrnici v nule jako zprávu čidlu, že read slot započal.
- `read_data_second` - Sběrnici uvolní a ponechá ji tak po následujících 10 μs .
- `read_data_third` - Vyčtený bit uloží do předávacího signálu `receive`.
- `read_data_fourth` - Čeká po dobu 55 μs na konec read slotu a vrací se zpět do stavu `branch_read`.
- `crc` - Poslední stav ve stavovém automatu. Došlo-li k úspěšnému vyčtení všech 71 bitů z paměti `SCRATCHPAD`, spustí se proces ověření CRC kódu, který je do čidla implementován výrobcem [1]. Pracovní registr `crc_buffer` se naplní nulami a pak už se provádí operace XOR mezi vstupem z daného bitu načtených dat a pozicemi `crc_buffer` danými vzorcem [1]:

$$CRC = X^8 + X^5 + X^4 + 1$$

Je-li konečný stav `crc_buffer` po provedení všech 72 iterací osm nul, pak data došly neporušené a aktualizuje se signál `teplota_24bit`. Není-li tomu tak, cyklus čtení teploty se spustí znovu bez aktualizace teploty. Vzhledem k vzorkování teplot pro průměrování s taktom 3 minuty (viz. blok integrátor), je počet provedených vzorků za 1 průměrovací takt více než dostatečný.

Na obrázku 3.8 je stavový diagram bloků vyčítání teplot. Pro každý z bloků je diagram, stejně jako stavy samotné, stejný.



Obrázek 3.8: Stavový diagram vyčítání teploty

■ Blok hodinové předděličky

Úkolem této části je vydělit 1MHz hodinový signál na delší časové úseky potřebné dále v systému. K taktování vzorkování teplot slouží signál `clock_sample` s periodou 50ms. Sekundový hodinový signál `clock_sec` zajišťuje počítání 180 sekund větrání toalety. Dále čtvrt hodinový signál `clock_frame` jako

spouštěč výpočtů průměrů v bloku integrátor. Poslední hodinovým signálem je `clock_button`, který v bloku `odstraneni_zakmitu` slouží jako taktovací signál při ošetření zákmitů při stisku tlačítek a přepínání přepínačů. Jeho perioda je 50ms.

■ Blok integrátoru

Integrátor zabezpečuje zpracování naměřených teplot. Prvním krokem je proces spouštěný hodinovým signálem `clock_sample` s periodou 3 minuty z části Blok hodinové předděličky `clock_sample`. Hned po načtení teplotního signálu je tento převeden z typu `std_logic_vector` na typ `integer` pomocí funkce `conv_integer` z knihovny `Std_Logic_Arith` [8][9]. Tyto čtyři teploty jsou následně zprůměrovány a uloženy do pole integerů `datalogger`. Toto pole má 672 dílčích proměnných pro každou čtvrt hodinu týdne a do každé z nich se ukládá průměrná teplota v dané čtvrt hodině navzorkovaná vždy po třech minutách. Tyto čtvrt hodiny se však nutně nekryjí se čtvrt hodinami dle skutečného času. A zařízení s teplotními čidly tak může pracovat jako systém nezávislý na okolí.

Je-li změřeno všech 672 čtvrt hodin týdne, následuje druhý průměrovací proces. Zde se zprůměruje každých 7 čtvrt hodin změřených ve stejnou denní dobu. Výsledkem je tedy 96 čtvrt hodin (dohromady jeden den).

Následuje další proces, který postupně prochází všech 96 zprůměrovaných čtvrt hodin dne a hledá čtvrt hodinu s nejvyšším a nejnižším průměrem. Po Prohledání celého pole si tyto teploty a jim odpovídající čtvrt hodiny zaznamená. Podstatné pro nás budou zejména čtvrt hodiny. Teploty samotné jsou ponechány pro další vývoj, pro daný účel nejsou nezbytné.

Smyslem celé této operace je zjistit denní dobu s průměrně nejvyšší teplotou během dne a denní dobu s průměrně nejnižší teplotou. Toto průměrování během celého týdne umožňuje vyloučit vliv náhodných změn počasí (v tomto případě teploty) a tím případného zkreslení výsledků. Výstupem jsou tedy čtvrt hodiny dne s průměrně nejvyšší a nejnižší teplotou a zároveň je vyvedeno počítadlo čtvrt hodin jako indikace aktuální čtvrt hodiny denní doby. Ta je v bloku integrátoru vypočítaná jako čtvrt hodina týdne modulu 96.

■ Blok odstranění zákmitů

Tento blok je vstupním blokem pro signály z tlačítek a přepínačů. Vstupují do něj tlačítka pro nastavování teplotního menu, tlačítka pro vyvětrání toalety, přepínač pro větrání kuchyně, přepínače pro letní/zimní mód a přepínač pro zapnutí termoregulační funkce přístroje. Není-li tento sepnut, přístroj slouží

pouze jako teploměr a integruje teplotu pro případ budoucího znovuspuštění.

Signál z ovládacích prvků se vzorkuje s periodou 50ms a změna stavu se objeví na výstupu po třech stejných stavech za sebou.

■ Blok řízení topení

Tento blok řídí uživatelské menu nastavení horní (tlačítko BTNU) a dolní mezní teploty (tlačítko BTND). V zimním módu je dolní teplota noční teplotou a horní teplota je denní referenční teplotou pro termostat pomocného i hlavního vytápění. V létě se funkce těchto teplot mění. Dolní teplota je minimální teplota pod kterou se dům již nemá ochlazovat a horní teplota maximální venkovní teplota, při které se ještě má větrat s výměníkem, a při které už ne. Toto rozlišení na letní a zimní mód však již není funkcí tohoto bloku, děje se až dále v bloku Řídicí modul. Rozsahy těchto teplot jsou od 5-ti do 30-ti °C, jsou nastavitelné s krokem 0,5°C. Při stisku tlačítka se teplota zvýší o 0,5°C, při dosažení 30°C a opětovném stisku se nastaví na 5°C a pak při dalších stiscích stoupá zase ke 30°C. Nastavení teplota se po dobu 5 sekund objeví na sedmi segmentovém displeji. Přepojení displeje do nastavovacího módu místo zobrazení měřených teplot zajišťuje Blok přepínač displeje (viz.). Dolní mezní teplota se objeví na čtyřech levých digitech, kdežto horní mezní teplota na 4 pravých digitech ve stejném formátu jako aktuálně měřené teploty: Dvojmístná celočíselná část, desetinná tečka a jedno desetinné místo.

■ Blok Přepínač displeje

Tento blok dle příznakového bitu (`temp_settings_flag`) vyslaného Blokem řízení topení přepíná sedmi segmentový displej ze zobrazení aktuálně měřených teplot na teploty nastavované v menu a naopak. Je-li příznak nastaven, je do výstupní proměnné `temp_display_24bit` vyslán signál z proměnné `temp_settings`, tedy nastavované teploty z menu. V opačném případě se na výstup zrcadlí proměnné `teplota_24bit`, tedy teploty aktuálně naměřené z čidel DS18B20.

■ Blok řídicího modulu

A nyní se dostáváme k poslední části celého zařízení, řídicímu modulu. Jeho úkolem je až dosud získaná data porovnat a dle výsledků spínat ventilaci, tepelný výměník, a topné těleso. Vstupními daty řídicího modulu jsou venkovní a vnitřní teplota, opět převedené na integer s pomocí funkce `conv_integer` z knihovny `Std_Logic_Arith`. [8][9] Dále jsou to výstupy bloku integrátoru, čtvrt hodina s průměrně nejvyšší a nejnižší teplotou dne a čtvrt hodina dne, v níž se zrovna nacházíme.

Vstupními daty jsou ještě uživatelsky nastavené teploty z bloku Řízení topení - dolní a horní uživatelská teplota. V zimním módu je dolní teplota noční teplotou a horní teplota je denní referenční teplotou pro řízení vytápění. V létě se funkce těchto teplot mění. Dolní teplota je minimální teplota pod kterou se dům již nemá ochlazovat a horní teplota maximální venkovní teplota, při které se ještě má větrat s výměníkem, a při které už ne.

Jako první krok je nutné určit, zda již nastala doba, kdy se má větrat. Ta byla určena jako dvouhodinový interval, tři čtvrt hodiny před nejteplejší/nejchladnější čtvrt hodinou, pak tato čtvrt hodina samotná a další čtyři za ní. Vypočítají se tedy počáteční a koncové čtvrt hodiny pro léto (okolo nejchladnější čtvrt hodiny) a pro zimu (okolo nejteplejší čtvrt hodiny).

Následně se hlídá, jestli se v tomto intervalu nacházíme. Jestliže ano, nastaví se příznak pro větrání do 1, když ne, pak se nastaví do nuly.

Pak se ve větrací době neustále porovnává vnější teplota s vnitřní. Pak se pro zimu zjišťuje, zda je venku v nejteplejší dvouhodině dne:

- Tepleji než v budově – větrá se bez výměníku.
- Chladněji než v budově ale o méně než 30°C – větrá se s výměníkem a topí se, jestliže teplota v domě je nižší než horní uživatelská teplota (jestliže je den), nebo dolní uživatelská teplota (jestliže je noc).
- A nebo chladněji o více než 30 °C – nevětrá se.

Jestliže zrovna není větrací časové okno a teplota v domě je vyšší než horní uživatelská teplota (jestliže je den), nebo dolní uživatelská teplota (jestliže je noc), pak se netopí a necirkuluje. Jsou-li nižší, pak se topí s recirkulací. Určení dne a noci probíhá obdobným způsobem jako u větracího časového okna. Nocí se rozumí 8h před a 2h po čtvrt hodině s průměrně nejnižší teplotou během dne.

Pro léto pak, zda je venku v nejchladnější dvouhodině dne:

- Chladněji než v budově a uvnitř je více, než dolní uživatelská teplota,

větrá se bez výměníku.

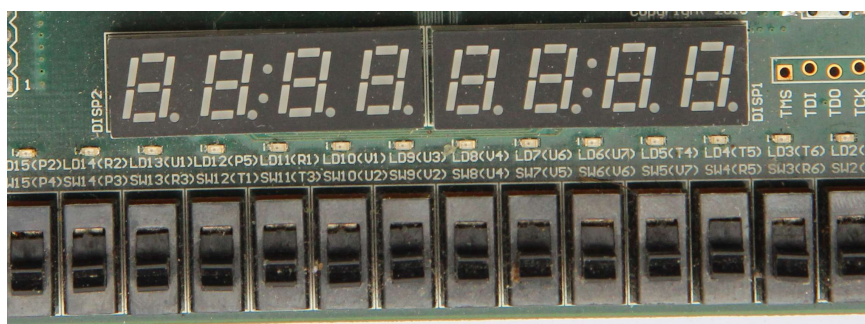
- Chladněji než v budově a uvnitř je méně, než dolní uživatelská teplota - větrá se s výměníkem.
- Tepleji než v budově a venku je méně, než horní uživatelská teplota, větrá se s výměníkem.
- Tepleji než v budově a venku je více, než horní uživatelská teplota a více, nevětrá se.

Větrací jednotka je simulačně vyvedena na LED6 - signál `led_sig_06`, tepelný výměník na LED4 - signál `led_sig_04`, externí vytápění (pro budoucí vývoj, nezapojeno) - LED10 - signál `led_sig_10`, teplovzdušné topení na LED8 - signál `led_sig_08`, recirkulace na LED2 - signál `led_sig_02`. Pokud jde o spínače, Větrání v kuchyni se spíná spínačem SW14 - signál `led_sig_14`. Letní/zimní mód se spíná spínačem SW12 - signál `led_sig_12`, kde poloha log. 0 reprezentuje zimu a poloha log. 1 reprezentuje léto. Výstupy se zapínají spínačem SW14 - signál `led_sig_14`, kde poloha log. 0 reprezentuje vypnuto a poloha log. 1 reprezentuje zapnuto. Nakonec, tlačítko BTNU slouží pro volbu horní uživatelské teploty v menu. Tlačítko BTND slouží pro volbu dolní uživatelské teploty v menu. Tlačítko BTNC slouží pro spouštění 3 minutového větrání na toaletě.

3.3 Vlastní řešení - typické výstupy

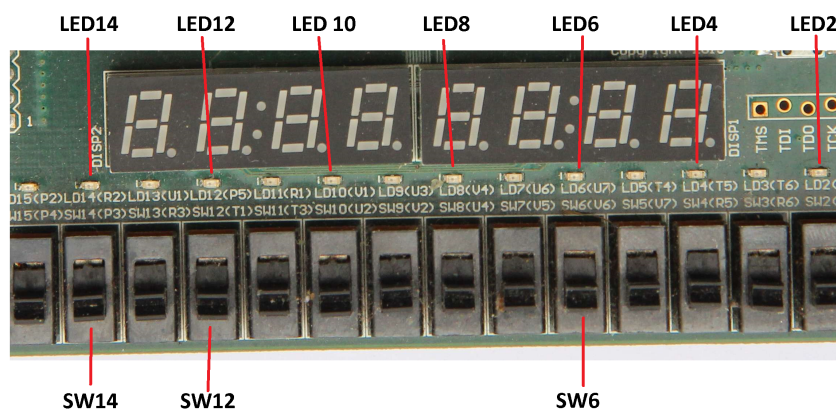
A nyní již k praktickým ukázkám. Stavů zde uvedené nejsou kompletní výčet těch, které mohou nastat. Stavů je celkem 38 a jejich výčet naleznete v tabulkách 3.1, 3.2, 3.3, 3.4 a 3.5. Zde si uvedeme stavy pokrývající všechny výstupní funkce, zbývající stavy jsou již jen kombinací vstupů a výstupů stavů zde uvedených.

Na obrázku 3.9 vidíme vstupy a výstupy systému, které jsou pro funkci potřeba. Jsou to přepínače Switch 06 (větrání kuchyně - v poloze log. 1), Switch 12 (přepínání mezi módem léta (log. 1) a zimy (log. 0) a Switch 14 - spínač termoregulační funkce. V poloze log. 0 systém pouze měří teplotu, ale nespíná rekuperační jednotku. V poloze log. 1 pracuje bez omezení. Dále jsou to LED diody indikující, že příslušné zařízení bylo uvedeno v činnost: LED 02 (recirkulace vzduchu po domě), LED 04 (tepelný výměník), LED 06 (větrání), LED 08 (teplovzdušné topení), LED 10 (externí topení), LED 12 (letní/zimní mód) a konečně LED 14 (termoregulační funkce). Dále jsou ještě použita tlačítka BTNU (horní uživatelská teplota), BTND (dolní uživatelská teplota) a BTNC (vyvětrání toalety). Ty jsou zde mimo výřez. Naleznete je však na obrázku 2.2.



Obrázek 3.9: Vstupy a výstupy

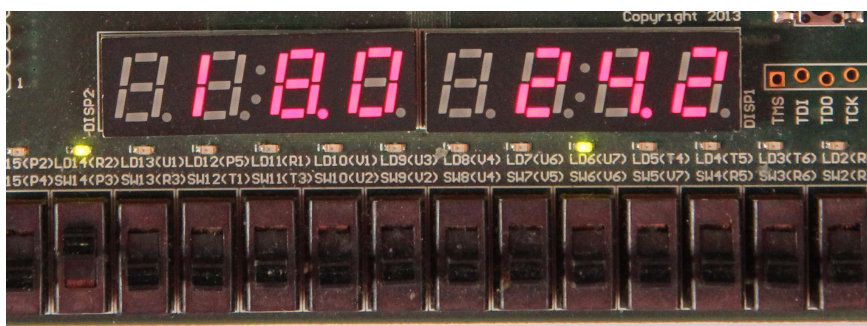
A nyní na obrázku 3.10 totéž se zvýrazněnými prvky, jež jsme si právě popsali:



Obrázek 3.10: Zima - 1

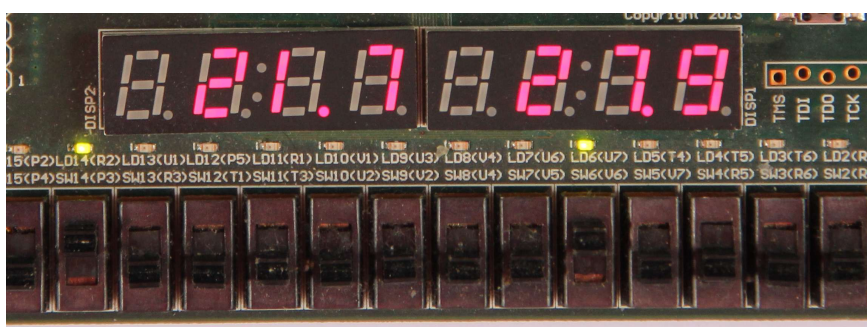
3.3.1 Zimní stavy

Na obrázku 3.11 vidíme stav, kdy byl spínač větrání v kuchyni přepnut do polohy log. 1. Zde je venku tepleji než uvnitř, větrá se tedy bez výměníku. Pro další kombinace teplot se větrání kuchyně chová analogicky jako větrání iniciované časovým větracím oknem s tím rozdílem, že se větrá za každého počasí. Tedy i tehdy, je-li teplotní rozdíl větší než 30°C . Aktivní je pouze výstup větrání LED 6.



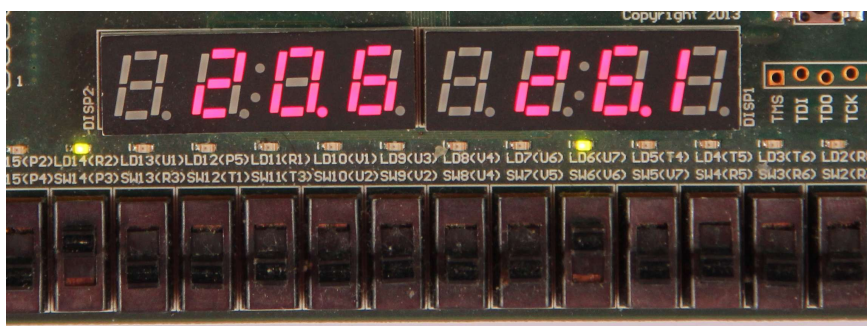
Obrázek 3.11: Zima - 1

Na obrázku 3.12 je podobná situace jako na 3.11. Zde se jedná o tlačítko 3 minutového větrání na toaletě. Aktivní je pouze výstup větrání LED 6, 3 minuty zde ještě neuběhly a větrá se.



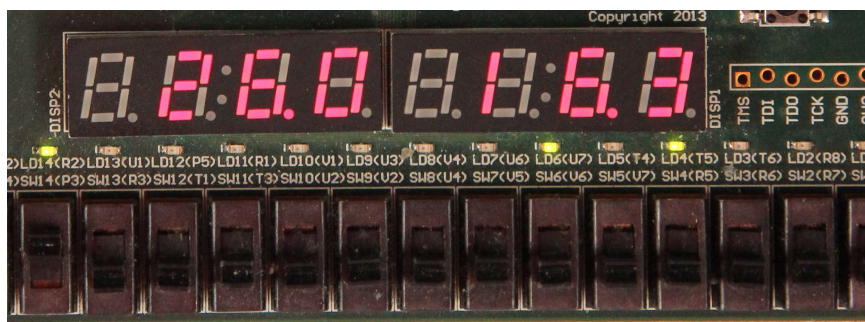
Obrázek 3.12: Zima - 2

Na obrázku 3.13 je situace, kdy se větrá automaticky. Venku je tepleji než uvnitř, takže probíhá větrání bez výměníku. Aktivní je pouze výstup větrání LED 6.



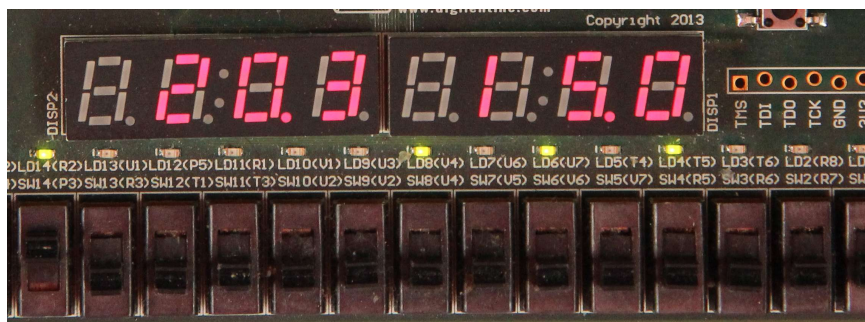
Obrázek 3.13: Zima - 3

Na obrázku 3.14 nastala denní doba s nejvyšší průměrnou teplotou. Venku je chladněji než uvnitř, teplotní rozdíl je menší než 30 °C. Uvnitř je více, než nastavená denní/noční teplota. Aktivní je výstup větrání LED 6 a tepelný výměník LED 4.



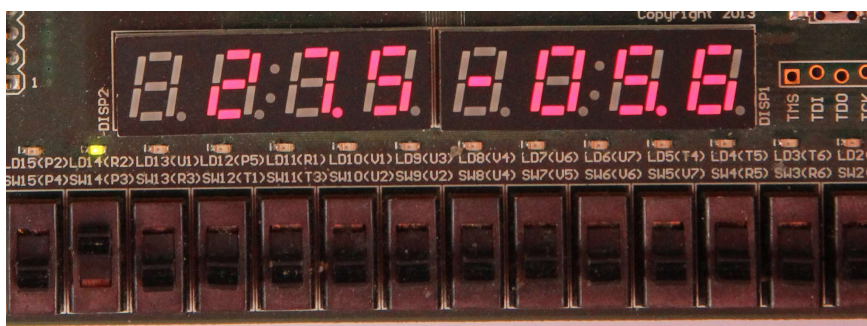
Obrázek 3.14: Zima - 4

Na obrázku 3.15 nastala denní doba s nejvyšší průměrnou teplotou. Venku je chladněji než uvnitř, teplotní rozdíl je menší než 30 °C. Uvnitř je méně, než nastavená denní/noční teplota. Aktivní je výstup větrání LED 6, tepelný výměník LED 4 a přitápění LED 8.



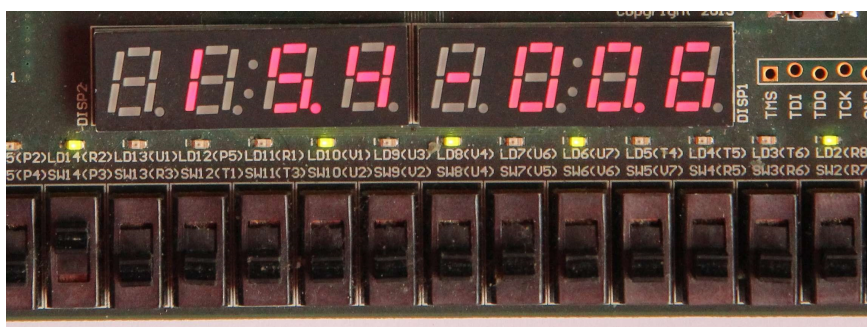
Obrázek 3.15: Zima - 5

Na obrázku 3.16 nastala denní doba s nejvyšší průměrnou teplotou. Venku je chladněji než uvnitř, teplotní rozdíl je větší než 30°C. Uvnitř je více, než nastavená denní/noční teplota. Nevětrá se a netopí, všechny výstupy mají hodnotu log. 0.



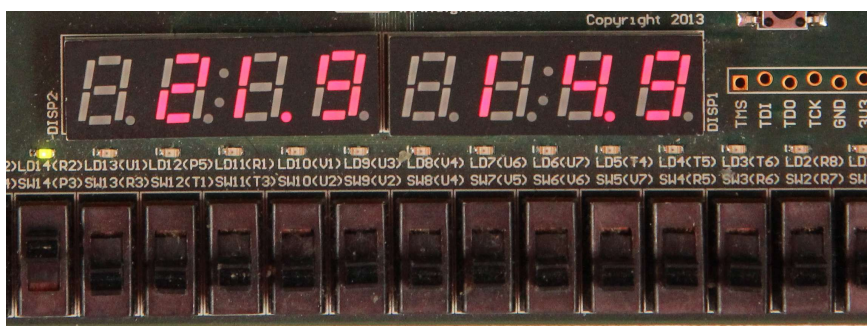
Obrázek 3.16: Zima - 6

Na obrázku 3.17 nastala denní doba s nejvyšší průměrnou teplotou. Venku je chladněji než uvnitř, teplotní rozdíl je větší než 30°C. Uvnitř je méně, než nastavená denní/noční teplota. Aktivní je výstup větrání LED 6, teplovzdušné topení LED 8 a recirkulace LED 2.



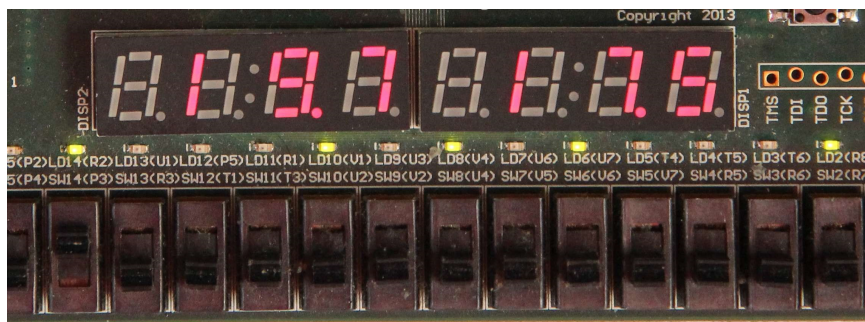
Obrázek 3.17: Zima - 7

Na obrázku 3.18 byla určena denní doba jako noc. Nenastala denní doba s nejvyšší průměrnou teplotou. Nevětrá se a netopí, žádný výstup tedy není aktivní.



Obrázek 3.18: Zima - 8

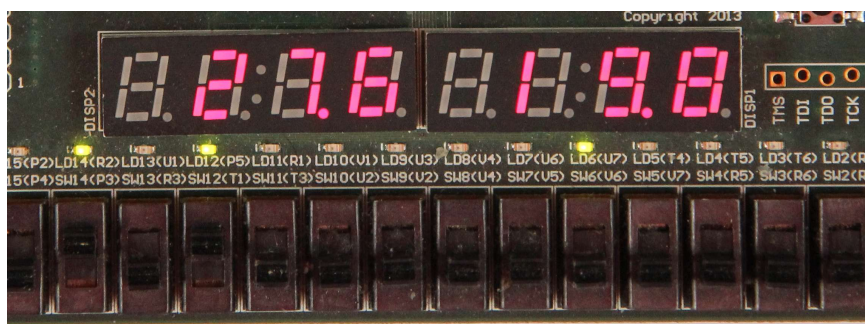
Na obrázku 3.19 byla určena denní doba jako noc. Nenastala denní doba s nejvyšší průměrnou teplotou. uvnitř je méně, než nastavená noční teplota. Aktivní je výstup větrání LED 6, teplotovzdušné topení LED 8, externí topení LED 10 a recirkulace LED 2.



Obrázek 3.19: Zima - 9

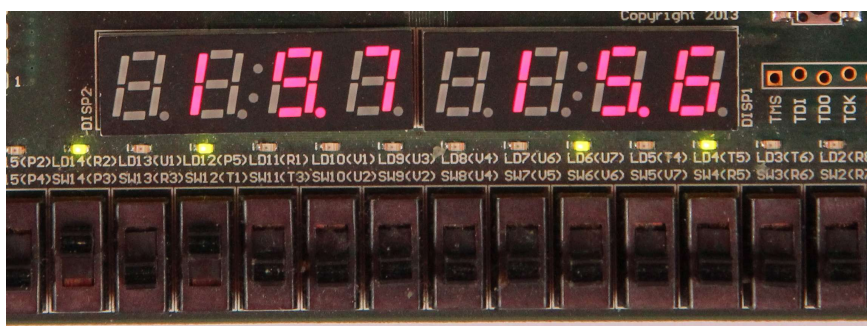
3.3.2 Letní stavy

Na obrázku 3.20 nastala denní doba s nejnižší průměrnou teplotou. Venku je chladněji než uvnitř a uvnitř je více, než nastavená minimální teplota. Aktivní je pouze výstup větrání LED 6.



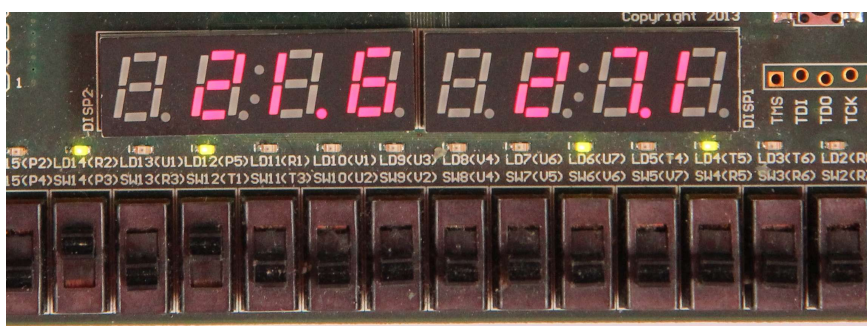
Obrázek 3.20: Léto - 1

Na obrázku 3.21 nastala denní doba s nejnižší průměrnou teplotou. Venku je chladněji než uvnitř a uvnitř je méně, než nastavená minimální teplota. Aktivní je výstup větrání LED 6 a tepelný výměník LED 4.



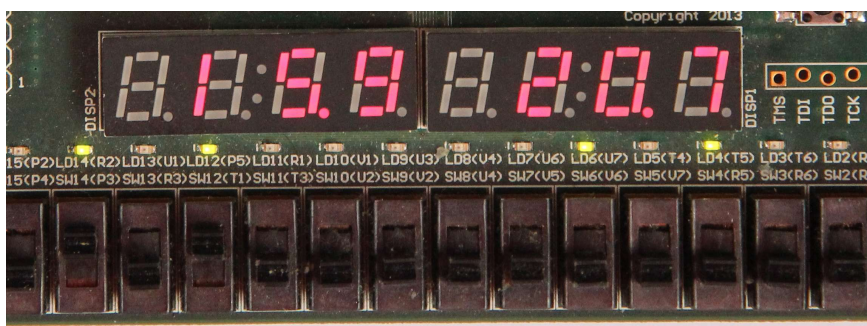
Obrázek 3.21: Léto - 2

Na obrázku 3.22 nastala denní doba s nejnižší průměrnou teplotou. Venku je tepleji než uvnitř, venku je méně, než horní nastavená teplota a uvnitř je více, než dolní nastavená teplota. Aktivní je výstup větrání LED 6 a tepelný výměník LED 4.



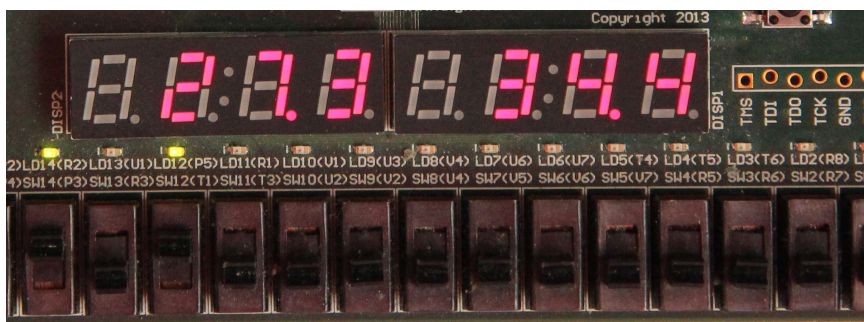
Obrázek 3.22: Léto - 3

Na obrázku 3.23 nastala denní doba s nejnižší průměrnou teplotou. Venku je tepleji než uvnitř, venku je méně, než horní nastavená teplota a uvnitř je méně, než dolní nastavená teplota. Aktivní je výstup větrání LED 6 a tepelný výměník LED 4.



Obrázek 3.23: Léto - 4

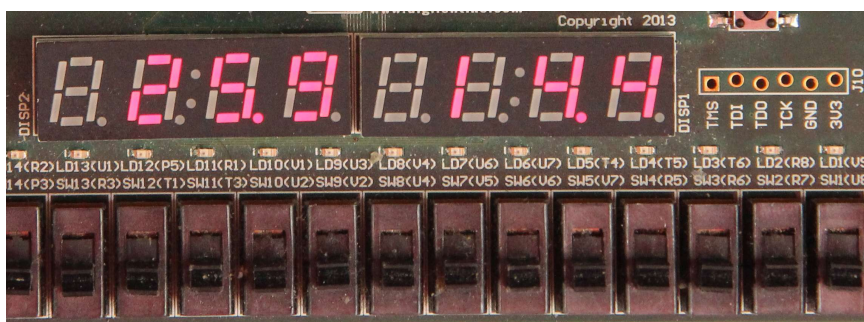
Na obrázku 3.24 nastala denní doba s nejnižší průměrnou teplotou. Venku je tepleji než uvnitř, venku je více, než horní nastavená teplota a uvnitř je více, než dolní nastavená teplota. Nemá se tedy větrat ani topit a žádný výstup není aktivní



Obrázek 3.24: Létá - 5

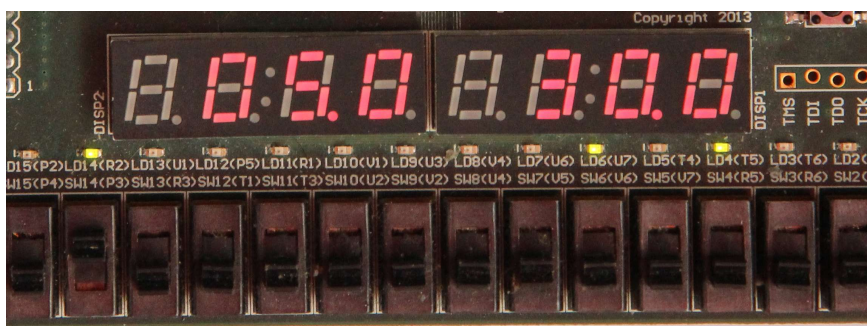
3.3.3 Další možné výstupy

Na obrázku 3.25 byl přepnut spínač termoregulační funkce do polohy log. 0, zařízení jen měří, zobrazuje a integruje teplotu. Žádný z logických výstupů není aktivní.



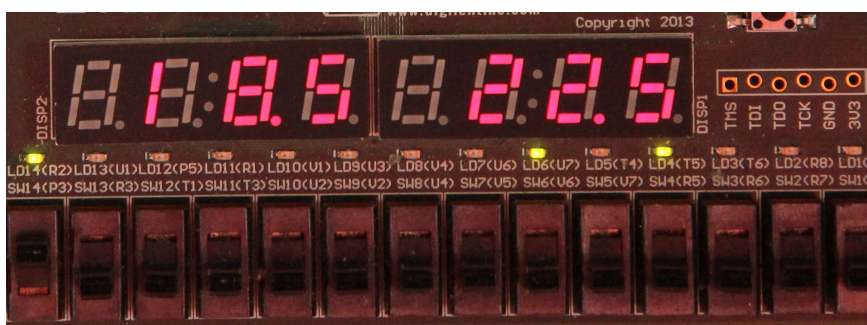
Obrázek 3.25: Odpojené výstupy

Na obrázku 3.27 vidíme uživatelské menu pro nastavení dolní (vlevo) a horní (vpravo) uživatelské teploty. Tyto teploty zde vidíme v jejich krajních polohách: 5°C a 30°C.



Obrázek 3.26: Uživatelské menu

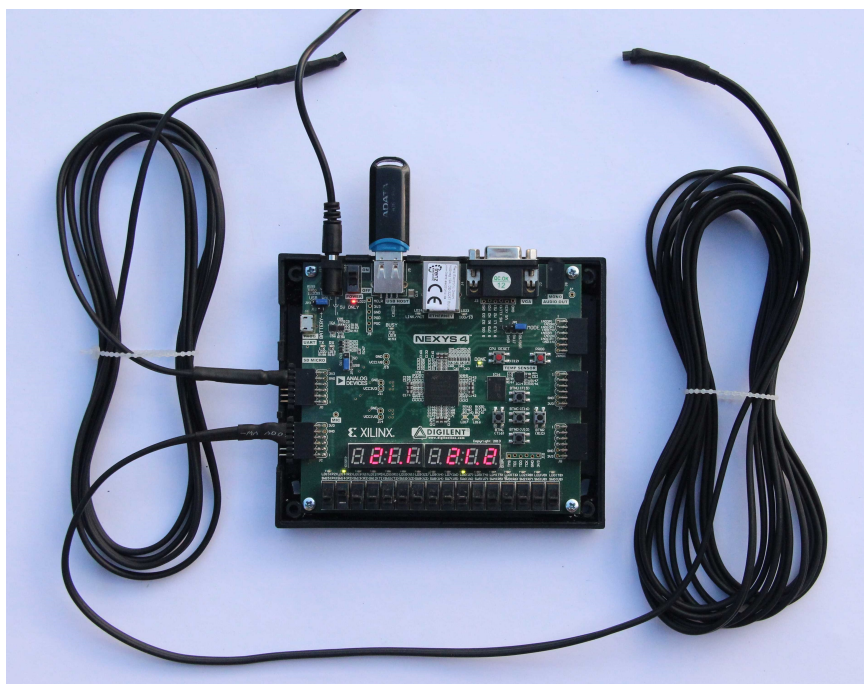
A nyní na obrázku 3.27 obdobný pohled, tentokrát pro pokojové teploty - obvyklé nastavení.



Obrázek 3.27: Uživatelské menu

3.3.4 Celkový pohled na zařízení

Na obrázku 3.28 vidíme pracující zařízení vestavěné do krabičky. Měří teplotu dvěma čidly připojenými dvoumetrovým a pětimetrovým stíněným kabelem. Binární kód je načítán z flash disku, napájení je realizováno externím zdrojem.



Obrázek 3.28: Celkový pohled

3.4 Kabeláž čidel

Při délkách kabelů k čidlům okolo 5m je již rušení poměrně výrazné. Pro delší vedení se doporučuje stíněný nebo alespoň kroucený UTP kabel s přídatným pull-up rezistorem (zde $3,9k\Omega$, mezi datovým a napájecím vodičem) a kondezátorem mezi napájecím napětím a zemí. Jeden pár UTP kabelu byl použit pro pár data - zem a druhý pro pár +5V - zem.

3.5 Uvedení do provozu

Zařízení nevyžaduje složitější proceduru uvedení do provozu, vše se děje automaticky. Nicméně je důležité zdůraznit, že aby zařízení správně pracovalo, musí být týden nepřetržitě v činnosti a integrovat teploty. Teprve až se tak stane, zařízení má dost dat na to, aby bezpečně určilo extrémy teploty a správně řídilo rekuperační jednotku. Toto platí i v případě, že bylo přerušeno napájecí napájení. V takovém případě se totiž rozsynchronizuje počátek dne a opět je nutno ponechat týden na kalibraci. V praxi toto realizujeme tak, že

spínač zapnutí výstupů přístroje SW 14 přepneme do polohy '0', ale spínač napájecího napětí ponecháme zapnutý. Zařízení tak bude měřit a integrovat teploty, avšak výstupy ponechá vypnuté.

Zařízení po zapnutí napájení automaticky načte binární kód z připojeného flash disku nebo z SD karty dle nastavení přepínače JP1 a JP2.

■ 3.6 Bezpečnost

Zařízení je napájeno napětím 5V stejnosměrných z externího zdroje proudem až 1A. Není tudíž nutno činit zvláštní bezpečnostní opatření týkající se zařízení napájených přímo z elektrické sítě. Bezpečnostním regulím musí podléhat použitý napájecí zdroj.

Kapitola 4

Závěr

V této bakalářské práci bylo navrženo, zhotoveno a otestováno modelové řídicí zařízení pro rekuperační jednotku na bázi přípravku Digilent Nexys 4 v jazyce VHDL.

První dvě kapitoly této bakalářské práce byly věnovány teoretickým základům, seznámení se s použitým hardwarem a nástinem řešeného problému.

Ve třetí kapitole byl proveden podrobný návrh systému. Byla popsána funkce a rozhraní jednotlivých funkčních bloků VHDL kódu. Dále byly uvedeny fotografie význačných výstupů zařízení.

V příloze pak naleznete ukázkou zdrojového kódu.

Zařízení měří teplotu uvnitř a vně rodinného domu. K tomuto účelu používá dvě čidla DS18B20. Zjišťuje průměrně nejteplejší a průměrně nejchladnější denní dobu prostřednictvím týdenní integrace teplot pro každou čtvrt hodinu dne.

Poskytuje uživateli možnost nastavit dvě uživatelské teploty, jejichž funkce závisí na zvolené roční době. K zobrazení uživatelského menu, měřených a nastavovaných teplot slouží sedmissegmentový displej s desetinnou tečkou. Dále umožňuje vyvětrání toalety a kuchyně v libovolnou roční dobu.

V závislosti na těchto vstupech pak zařízení udržuje optimální teplotu v budově za maximálního využití energeticky efektivního větrání a teplotovzdušného vytápění v rekuperační jednotce.

Zařízení do budoucna ponechává v případě potřeby možnost připojení například externího vytápění, např. plynové kotle, a příslušné výstupy jsou již vyvedeny. Další možnosti rozšíření se nabízejí v podobě přidání ovládání klimatizace či implementace reálného času. Také lze navýšit počet senzorů a pomocí recirkulace rozvádět teplo po domě z bodového zdroje, např. krbu.

Zařízení funguje spolehlivě se všemi funkcemi požadovanými v zadání. Přesnost měření teplot je dostatečná. Zařízení poskytl funkce, jež řídicí jednotky nabízené výrobcem rekuperačních jednotek zpravidla nemají.

Zdrojový kód ve VHDL, projektové soubory a jiné doplňkové materiály jsou k dispozici na přiloženém CD.

Tato bakalářská práce velmi rozšířila mé znalosti práce s jazykem VHDL, simulace v něm navržených modulů, práce s přípravkem *Nexys 4* a znalosti o One-Wire datové sběrnici.



Kapitola 5

Citace

[1] DS18B20 Programmable Resolution 1-Wire Digital Thermometer. In: *All datasheet* [online]. Maxim Integrated Products, 120 San Gabriel Drive, Sunnyvale, CA 94086 408- 737- 7600: Maxim Integrated Products, 2008, 2008 [cit. 2019-12-28]. Dostupné z: <https://www.alldatasheet.com/datasheet-pdf/pdf/227472/DALLAS/DS18B20.html>

[2] DIGILENT, INC. Nexys 4™ FPGA Board Reference Manual. In: *Digilent* [online]. 1300 Henley Court Pullman, WA 99163: Digilent, 2006, 11 April 2006, 1 - 21 [cit. 2019-12-28]. Dostupné z: https://reference.digilentinc.com/_media/nexys:nexys4:nexys4_rm.pdf

[3] JIŘÍ, Hrbáček. Převod Bin do BCD kódu: Převod Bin do BCD pomocí Hornerova schématu. In: *Wrack Ped MUNI* [online]. Brno: Masarykova Univerzita v Brně [cit. 2019-12-28]. Dostupné z: <http://wrack.ped.muni.cz/hrbacek/ucebnice/PrevodBinDoBcd.pdf>

[4] Digilent Nexys 4 Artix-7 FPGA Trainer Board. *Digilent Nexys 4 Artix-7 FPGA Trainer Board* [online]. San Jose: Xilinx, 2020, 2020 [cit. 2020-01-10]. Dostupné z: <https://www.xilinx.com/products/boards-and-kits/1-3yznp5.html>

[5] ŠNOREK, Miroslav, Michal ŠTEPANOVSÝ a Pavel PÍŠA. Architek-

tura počítačů: Počítačová aritmetika. In: *Course Ware* [online]. Praha: ČVUT, 2013, 2013, s. 14-15 [cit. 2020-01-10]. Dostupné z: http://scw.fel.cvut.cz/b182/_media/courses/b35apo/lectures/01/a0b36apo_prednaska01-2013.pdf

[6] APPLICATION NOTE 187: 1-WIRE SEARCH ALGORITHM. *Maxim Integrated* [online]. San Jose: Maxim Integrated Products, 2002, 28 Mar 2002 [cit. 2020-01-10]. Dostupné z: <https://www.maximintegrated.com/en/design/technical-documents/app-notes/1/187.html>

[7] RICHICHI, Francesco. VHDL General Concept: VHDL Driver and Source concept. Surf VHDL [online]. Surf VHDL, 2018 [cit. 2020-04-29]. Dostupné z: <https://surf-vhdl.com/vhdl-syntax-web-course-surf-vhdl/vhdl-driver-and-source-concept/>

[8] BAKER, Greg. The `conv_integer` function. The `conv_integer` function [online]. Vancouver: Simon Fraser University, 2002, 2002 [cit. 2020-04-30]. Dostupné z: https://www.cs.sfu.ca/~ggbaker/reference/std_logic/arith/conv_integer.html

[9] MERRICK, Russel. Examples of VHDL Conversions: Using both `Numeric_Std` and `Std_Logic_Arith` Package Files. Examples of VHDL Conversions: Using both `Numeric_Std` and `Std_Logic_Arith` Package Files [online]. Boston, 2017, 2017 [cit. 2020-04-30]. Dostupné z: <https://www.nandland.com/vhdl/tips/tip-convert-numeric-std-logic-vector-to-integer.html>

[10] Duplex RB: Jednotka Duplex RB. In: *Yumpu.com* [online]. Jablonec nad Nisou: Atrea, 2007, Mar 2007 [cit. 2020-05-06]. Dostupné z: <https://www.yumpu.com/xx/document/read/42746653/duplex-rb-prima-klima-pardubice>

[11] DOSTRAŠIL, Pavel. Větrací a vytápěcí vzduchotechnická jednotka Atrea RB [online]. Liberec, 2008 [cit. 2020-05-06]. Dostupné z: https://dspace.tul.cz/bitstream/handle/15240/5800/mgr_15027.pdf?sequence=1&isAllowed=y. Diplomová práce. Technická Univerzita v Liberci, Fakulta mechatroniky a mezioborových inženýrských studií. Vedoucí práce Ing. Miloš Hernych.

[12] Ceník výrobků: část Větrání a vytápění rodinných domů, bytů a bazénů rodinných domů. In: *Docplayer*[online]. V Aleji 466 01 Jablonec n. N.: Atrea, 2009, Apr 2009 [cit.2020-05-06]. Dostupné z: <https://docplayer.cz/16310729-Cenik-vyrobku-p-l-a-t-n-o-s-t-cast-v-etrani-a-vytapeni-r-odinnych-d-omu-bytu-a-bazenu-r-odinnych-d-omu-4-6-6-0-1-j-a-b-l-o-n-e-c-n-n.html>

[13] JÁNEŠ, Vlastimil. Jazyk VHDL. In: <https://www.fd.cvut.cz/>: *Fakulta dopravní* [online]. Praha: FD, ČVUT [cit. 2020-05-22]. Dostupné z: https://www.fd.cvut.cz/personal/janes/HWpocitacu/Laboratornicviceni/Navod_VHDL.pdf

[14] ISE Design Suite. *Xilinx.com: ISE Design Suite* [online]. San Jose: Xilinx [cit. 2020-05-22]. Dostupné z: <https://www.xilinx.com/products/design-tools/ise-design-suite.html>

[15] *The VHDL Handbook*. New York: Springer-Verlag New York, 2012. ISBN 9781461289029.

[16] MAXFIELD, Clive. Fundamentals of FPGAs: What Are FPGAs and Why Are They Needed? *Digikey.com: Fundamentals of FPGAs: What Are FPGAs and Why Are They Needed?* [online]. 2019, 14 Nov 2019 [cit. 2020-05-22]. Dostupné z: <https://www.digikey.com/en/articles/fundamentals-of-fpgas-what-are>

■ .1 Ukázky VHDL kódu

A nyní již ukážka VHDL kódu z bloku Regulace teploty implementujícího menu pro uživatelské teploty.

```
1  library IEEE;
2  use IEEE.STD_LOGIC_1164.ALL;
3  use IEEE.STD_LOGIC_ARITH.ALL;
4  use IEEE.STD_LOGIC_SIGNED.ALL;
5  use IEEE.NUMERIC_STD.ALL;
6
7
8  entity ridici_modul is                                     -- tento modul
   obsahuje vyslednou ridici logiku majici primy vliv na funkci rekuperacni jednotky
9      Port (teplota_24bit: IN std_logic_vector(23 downto 0); -- vystup
   namerenych teplot z cidel, pul vektoru pro kazde cidlo
10         peak_temperature : IN INTEGER RANGE -8388608 TO 8388607; -- maximalni
   teplota ve formatu integeru - nepouzito, ponechano pro budouci vyvoj
11         peak_quarter : IN INTEGER RANGE 0 TO 1023; -- ctvrt hodina s
   nejvyssi namerenou teplotou
12         minimal_temperature : IN INTEGER RANGE -8388608 TO 8388607; -- minimalni
   teplota ve formatu integeru - nepouzito, ponechano pro budouci vyvoj
13         minimal_quarter : IN INTEGER RANGE 0 TO 1023; -- ctvrt hodina s
   nejnizssi namerenou teplotou
14         quarter_counter : IN INTEGER RANGE 0 TO 1023; -- aktualni
   ctvrt hodina
15         clock_1 : IN STD_LOGIC; -- hodinovy
   signal 1MHZ
16         clock_sec : in std_logic; -- sekundovy
   hodinovy signal
17         sw_06_flag : in STD_LOGIC; -- prepinač vetrání v kuchyni
18         sw_12_flag : in STD_LOGIC; -- prepinač leto/zima
19         sw_14_flag : in STD_LOGIC; -- prepinač zapnutí přístroje
20         led_sig_00 : out STD_LOGIC; -- neimplementováno
21         led_sig_01 : out STD_LOGIC; -- neimplementováno
22         led_sig_02 : out STD_LOGIC; -- kontrolka cirkulace
23         led_sig_03 : out STD_LOGIC; -- neimplementováno
24         led_sig_04 : out STD_LOGIC; -- kontrolka výměníku
25         led_sig_05 : out STD_LOGIC; -- neimplementováno
26         led_sig_06 : out STD_LOGIC; -- kontrolka vetrání
27         led_sig_07 : out STD_LOGIC; -- neimplementováno
28         led_sig_08 : out STD_LOGIC; -- kontrolka topení
29         led_sig_09 : out STD_LOGIC; -- neimplementováno
30         led_sig_10 : out STD_LOGIC; -- kontrolka externího topení
31         led_sig_11 : out STD_LOGIC; -- neimplementováno
32         led_sig_12 : out STD_LOGIC; -- kontrolka leto/zima
33         led_sig_13 : out STD_LOGIC; -- neimplementováno
34         led_sig_14 : out STD_LOGIC; -- kontrolka zapojení výstupu
35         led_sig_15 : out STD_LOGIC; -- neimplementováno
36         heat_temp_day : in integer range 0 TO 1023; -- horní uživatelská teplota
   z menu
37         heat_temp_night : in integer range 0 TO 1023; -- dolní uživatelská teplota
   z menu
38         pb_c_flag_out : in STD_LOGIC -- při změně stavu proměnné
   doslo k vzestupně hrane (stisku) tlačítka 3 minutového větrání toalety
39     );
40 end ridici_modul;
41
42 architecture Behavioral of ridici_modul is
43
44     signal temperature_outdoor: INTEGER RANGE -8388608 TO 8388607; -- teplota z
   venkovního čidla převedena na integer
```

```
45 signal temperature_indoor: INTEGER RANGE -8388608 TO 8388607; -- teplota z
vnitrniho cidla prevedena na integer
46 signal quarter_start_s: INTEGER RANGE 0 TO 1023; -- vypoctena
ctrthodina v lete, kdy se ma zacit vetrat
47 signal quarter_stop_s: INTEGER RANGE 0 TO 1023; -- vypoctena
ctrthodina v lete, kdy se ma prestat vetrat
48 signal quarter_start_w: INTEGER RANGE 0 TO 1023; -- vypoctena
ctrthodina v zime, kdy se ma zacit vetrat
49 signal quarter_stop_w: INTEGER RANGE 0 TO 1023; -- vypoctena
ctrthodina v zime, kdy se ma prestat vetrat
50 signal heat_start_n: INTEGER RANGE 0 TO 1023; -- vypoctena
ctrthodina v zime, kdy se ma zacit topit
51 signal heat_stop_n: INTEGER RANGE 0 TO 1023; -- vypoctena
ctrthodina v zime, kdy se ma prestat topit
52 signal ventilate_s : STD_LOGIC := '0'; -- priznak vetrani
v lete
53 signal ventilate_w : STD_LOGIC := '0'; -- priznak vetrani
v zime
54 signal ventilate_bathroom : STD_LOGIC := '0'; -- priznak vetrani
na toalete
55 signal heat : STD_LOGIC := '0'; -- priznak vytapeni
56 signal semiheat : STD_LOGIC := '0'; -- priznak topeni
57 signal counter_01: INTEGER RANGE 0 to 1023; -- citac
58 signal ventilate_3min_flag : STD_LOGIC := '0'; -- priznak z bloku
pocitani 3 minut, zda se jeste ma vetrat na toalete
59 signal pb_c_flag_mem : STD_LOGIC := '0'; -- pamet detekce
zmen ystavu tlacitka na toalete
60 signal counter: INTEGER RANGE 0 to 1023;
61 signal t_ref: INTEGER RANGE 0 to 65534; -- pamet s
referencni teplotou nastavenou na horni ci dolni uzivatelskou teplotu
62
63 begin
64 process(clock_1)
65 begin
66 if (rising_edge(clock_1)) then
67 led_sig_00 <= '0'; -- osetreni nepouzitych led
68 led_sig_01 <= '0';
69 led_sig_03 <= '0';
70 led_sig_05 <= '0';
71 led_sig_07 <= '0';
72 led_sig_09 <= '0';
73 led_sig_11 <= '0';
74 led_sig_13 <= '0';
75 led_sig_15 <= '0';
76
77 -- vymezeni dvouhodinoveho
intervalu, ve kterem se bude vetrat v zime
78 if(peak_quarter>91) then -- osetreni stavu, je-li
nejteplejsi ctvrtthodina blizko pod 95 a koncova ctvrtthodina pretece
79 quarter_start_w <= (peak_quarter - 3); -- zimni pocatecni
ctvrtthodina vetrani
80 quarter_stop_w <= (peak_quarter - 92); -- zimni koncova ctvrtthodina
vetrani
81 elsif(peak_quarter<3) then -- osetreni stavu, je-li
nejteplejsi ctvrtthodina blizko nad 0 a koncova ctvrtthodina pretece
82 quarter_start_w <= (peak_quarter +93); -- zimni pocatecni
ctvrtthodina vetrani
```

```
83     quarter_stop_w <= (peak_quarter + 4);           -- zimni koncova ctvrtthodina
vetrani
84     else                                           -- standardni vypocet
zacatku a konce mimo mezni hodnoty
85     quarter_start_w <= (peak_quarter - 3);       -- zimni pocatecni
ctvrtthodina vetrani
86     quarter_stop_w <= (peak_quarter + 4);       -- zimni koncova ctvrtthodina
vetrani
87     end if;
88
89     if(minimal_quarter>91) then                   -- vymezeni dvouhodinoveho
intervalu, ve kterem se bude vetrat v lete, analogicky k zime, jen vstupem je
nejchladnejsi ctvrtthodina
90     quarter_start_s <= (minimal_quarter - 3);
91     quarter_stop_s <= (minimal_quarter - 92);
92     elsif(minimal_quarter<3) then
93     quarter_start_s <= (minimal_quarter +93);
94     quarter_stop_s <= (minimal_quarter + 4);
95     else
96     quarter_start_s <= (minimal_quarter - 3);
97     quarter_stop_s <= (minimal_quarter + 4);
98     end if;
99
100    if(minimal_quarter>87) then                   -- vymezeni vytapeciho
intervalu v zime
101    heat_start_n <= (minimal_quarter - 31);       -- pozaduje se topeni 8 hodin
(31 ctvrtthodin ) pred a 2 (8 ctvrtthodin) hodiny po minimu
102    heat_stop_n <= (minimal_quarter - 88);       -- tudiz ciselen hodnoty jsou
jine, jinak ale analogicky k vetrani
103    elsif(minimal_quarter<31) then
104    heat_start_n <= (minimal_quarter + 65);
105    heat_stop_n <= (minimal_quarter + 8);
106    else
107    heat_start_n <= (minimal_quarter - 31);
108    heat_stop_n <= (minimal_quarter + 8);
109    end if;
110
111
112    if(quarter_counter=quarter_start_s) then     -- citac ctvrtthodin vetrani vlete
113    ventilate_s <= '1';                           -- shoduje-li se aktulani
ctvrtthodina s pocatecni, nastav priznak vetrani do "vetrat"
114    end if;
115    if(quarter_counter=(quarter_stop_s+1)) then -- kdyz uz ubehly 2 hodiny a
shoduje-li se aktulani ctvrtthodina s koncovou, nastav priznak vetrani do "prestan
vetrat"
116    ventilate_s <= '0';
117    end if;
118
119    if(quarter_counter=quarter_start_w) then     -- citac ctvrtthodin vetrani v
zime, analogicke k letu
120    ventilate_w <= '1';                           -- priznak do vetrat
121    end if;
122    if(quarter_counter=(quarter_stop_w+1)) then -- kdyz uz ubehly 2 hodiny,
priznak do nevetrat
123    ventilate_w <= '0';
124    end if;
125
```

```
126
127     if(quarter_counter=heat_start_n) then           -- citac ctvrthodin topeni, zda je
denni ci nocni rezim
128         t_ref <= ((heat_temp_night)*16)+4368);      --nastav denni konstantu
referencni teploty
129     end if;
130     if(quarter_counter=(heat_stop_n+1)) then
131         t_ref <= ((heat_temp_day)*16)+4368);        --nastav nocni konstantu
referencni teploty
132     end if;
133
134
135     if(sw_14_flag = '1') then           -- je-li pristroj zapnut, proved vyhodnoceni
136         led_sig_14 <= '1';             -- signalizace, ze vystupy jsou zapojeny
137         case sw_12_flag is             -- zjistit stav prepinace modu LETO/ZIMA
138
139             when '0' =>                 -- mod zima
140
141                 led_sig_12 <= '0';     -- kontrolka indikujici zimu
142
143
144                 if(ventilate_w = '1') then           -- je-li doba vetrani, nebo byl
zapnut spinac vetrani, vetrej
145                     if ((sw_06_flag = '1')or(ventilate_3min_flag = '1')) then
146                                                                 -- zde
nutno jeste jednou vlozit vetrani toalety, a kuchyne, ma prioritu pred vetranim domu
jako celku
147                         if(temperature_outdoor > temperature_indoor) then           -- kdyz je
venku tepleji nez uvnitr, vetrej bez vymeniku
148                             led_sig_06 <= '1'; -- zapni vetrani
149                             led_sig_04 <= '0'; -- vypni vymenik
150                             heat <= '0';      -- netop externe
151                             semiheat <= '0';  -- netop teplotvzduzne
152                         elsif(temperature_outdoor < temperature_indoor) then -- kdyz je
venku chladneji, vetrej s vymenikem
153                             led_sig_06 <= '1'; -- zapni vetrani
154                             led_sig_04 <= '1'; -- zapni vymenik
155                             heat <= '0';      -- netop externe
156                             if(temperature_indoor < (t_ref-16))then           --jestlize je
zima a vnitri teplota je nizsi, nez pozadovana, pritapej
157                                 semiheat <= '1';    -- top
teplotvzduzne
158                                                                 -- je implementovana hystereze
2°C, 1°C na d apod pozadovanou teplotou
159                                 elsif(temperature_indoor > (t_ref+16))then
160                                     semiheat <= '0';    -- netop teplotvzduzne
161                                 else
162                                     semiheat <= '0';    -- netop teplotvzduzne
163                                 end if;
164                             else           -- jinak nevetrej a netop; osetreni vyjimky
165                                 led_sig_06 <= '0'; -- vypni vetrani
166                                 led_sig_04 <= '0'; -- vypni vymenik
167                                 heat <= '0';      -- netop
168                                 semiheat <= '0';  -- netop teplotvzduzne
169                             end if;
170                         elsif(temperature_outdoor > temperature_indoor) then           -- kdyz je
venku tepleji nez uvnitr, vetrej bez vymeniku
171                             led_sig_06 <= '1'; -- zapni vetrani
```



```
171         led_sig_04 <= '0';    -- vypni vymenik
172         heat <= '0';          -- netop
173         semiheat <= '0';     -- netop teplovzdušne
174         elsif((temperature_indoor > temperature_outdoor)and(
temperature_indoor < (temperature_outdoor + 480))) then -- kdž je venku chladneji o
mene nez 30 °C, vetrej s vymenikem
175             led_sig_06 <= '1'; -- zapni vetrani
176             led_sig_04 <= '1'; -- zapni vymenik
177             heat <= '0';      -- netop
178             if(temperature_indoor < (t_ref-16))then      --jestlize je zima
a vnitřni teplota je nizsi, nez pozadovana, zatop
179                 semiheat <= '1';    -- je-li uvnitr chladneji nez referenci
teplota, pritapej      -- je implementovana hystereze 2°C, 1°C nad a pod pozadovanou
teplotou
180                 elsif(temperature_indoor > (t_ref+16))then
181                     semiheat <= '0'; -- je-li uvnitr chladneji nez referenci
teplota, netop teplovzdušne
182                 else
183                     semiheat <= '0'; -- osetreni vyjimky
184                 end if;
185             elsif((temperature_indoor > temperature_outdoor)and(
temperature_indoor > (temperature_outdoor + 480))) then -- kdž je venku chladneji o
vice nez 30 °C, nevetrej
186                 led_sig_04 <= '0'; -- vypni vymenik
187                 if(temperature_indoor < (t_ref-16))then      --jestlize je zima
a vnitřni teplota je nizsi, nez pozadovana, zatop
188                     heat <= '1';
189                     semiheat <= '1';
190                     led_sig_06 <= '1'; -- zapni vetrani
191                 elsif(temperature_indoor > (t_ref+16))then
192                     heat <= '0';      -- netop externe ani teplovzdušne
193                     semiheat <= '0';
194                     led_sig_06 <= '0'; -- vypni vetrani
195                 else
196                     heat <= '0';      -- netop externe ani teplovzdušne
197                     semiheat <= '0';
198                     led_sig_06 <= '0'; -- vypni vetrani
199                 end if;
200             else      -- jinak nevetrej
201                 led_sig_06 <= '0'; -- vypni vetrani
202                 led_sig_04 <= '0'; -- vypni vymenik
203                 if(temperature_indoor < (t_ref-16))then      --jestlize je zima
a vnitřni teplota je nizsi, nez pozadovana, zatop
204                     heat <= '1';
205                     semiheat <= '1';
206                 elsif(temperature_indoor > (t_ref+16))then
207                     heat <= '0';      -- netop externe ani teplovzdušne
208                     semiheat <= '0';
209                 else
210                     heat <= '0';      -- netop externe ani teplovzdušne
211                     semiheat <= '0';
212                 end if;
213             end if;
214             elsif ((sw_06_flag = '1')or(ventilate_3min_flag = '1')) then
--osetreni vetrani na toaletě a v kuchyni, vetra se za kazdeho pocasi
215                 if(temperature_outdoor > temperature_indoor) then      -- kdž je
venku tepleji nez uvnitr, vetrej bez vymeniku
```

```
216         led_sig_06 <= '1'; -- zapni vetrani
217         led_sig_04 <= '0'; -- vypni vymenik
218         heat <= '0';      -- netop externe
219         semiheat <= '0';  -- netop teplotvdzusne
220         elsif(temperature_outdoor < temperature_indoor) then -- kdyz je
venku chladneji, vetrej s vymenikem
221             led_sig_06 <= '1'; -- zapni vetrani
222             led_sig_04 <= '1'; -- zapni vymenik
223             heat <= '0';      -- netop externe
224             if(temperature_indoor < (t_ref-16))then          --jestlize je zima
a vnitрни teplota je nizsi, nez pozadovana, pritapej
225                 semiheat <= '1';      -- top
teplotvdzusne                                -- je implementovana hystereze
2°C, 1°C na d apod pozadovanou teplotou
226             elsif(temperature_indoor > (t_ref+16))then
227                 semiheat <= '0';      -- netop teplotvdzusne
228             else
229                 semiheat <= '0';      -- netop teplotvdzusne
230             end if;
231         else -- jinak nevetrej a netop
232             led_sig_06 <= '0'; -- vypni vetrani
233             led_sig_04 <= '0'; -- vypni vymenik
234             heat <= '0';      -- netop externe
235             semiheat <= '0'; -- netop teplotvdzusne
236         end if;
237         else -- jinak nevetrej, je-li zima, zatop
238             led_sig_04 <= '0'; -- vypni vymenik
239             semiheat <= '0'; -- nepritapej
240             if(temperature_indoor < (t_ref-16)) then          --jestlize je zima a
vnitрни teplota je nizsi, nez uzivatelska pro danou denni dobu, zatop
241                 heat <= '1'; -- zatop
242                 semiheat <= '1'; -- top teplotvdzusne
243                 led_sig_06 <= '1'; -- zapni vetrani
244                 elsif(temperature_indoor > (t_ref+16)) then -- kdyz je vyssi nez
uzivatelska pro danou denni dobu, netop
245                     heat <= '0'; -- netop
246                     semiheat <= '0'; -- netop teplotvdzusne
247                     led_sig_06 <= '0'; -- vypni vetrani
248                 else
249                     heat <= '0'; -- netop
250                     semiheat <= '0'; -- netop teplotvdzusne
251                     led_sig_06 <= '0'; -- vypni vetrani
252                 end if;
253             end if;
254
255
256         when '1' => -- mod leto
257             led_sig_12 <= '1'; --kontrolka indikujici leto
258             heat <= '0';      -- nikdy netop externe
259             semiheat <= '0';  -- nikdy netop teplotvdzusne
260             if(ventilate_s = '1') then
261
262                 if(sw_06_flag = '1')or(ventilate_3min_flag = '1') then          --
vetrani toalety a kuchyne ma prednostS
263                     if ((temperature_outdoor < temperature_indoor)and(
temperature_indoor > ((heat_temp_night)*16)+4368)) then -- je-li venku chladneji
nez uvnitr a uvnitr je vice nez dolni nastavena teplota, tak vetrej bez vymeniku
```