



**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**

---

**Fakulta elektrotechnická  
Katedra telekomunikační techniky**

**Obsluha vstupních a výstupních periférií pomocí VHDL**

**Bakalářská práce**

Studijní program: Komunikace, multimédia, elektronika  
Studijní obor: Síťové a informační technologie

Vedoucí práce: Ing. Pavel Lafata, Ph.D.

**Tomáš Pehnelt**

---

**Praha 2014**

## Čestné prohlášení

Prohlašuji, že jsem zadanou bakalářskou práci zpracoval sám s přispěním vedoucího práce a konzultanta a používal jsem pouze literaturu v práci uvedenou. Dále prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé bakalářské práce nebo její části se souhlasem katedry.

V Praze dne: .....

.....

Tomáš Pehnelt

České vysoké učení technické v Praze  
Fakulta elektrotechnická

katedra telekomunikační techniky

## ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Student: **Pehnelt Tomáš**

Studijní program: Komunikace, multimédia a elektronika  
Obor: Síťové a informační technologie

Název tématu: **Obsluha vstupních a výstupních periférií pomocí VHDL**

Pokyny pro vypracování:

Seznamte se s přípravky Digilent Nexys 3 a Nexys 4 a jejich obsluhou pomocí jazyka VHDL. Navrhněte a realizujte připojení jednoduchých externích čidel fyzikálních veličin (teplota, tlak, vlhkost) a vestavěných čidel na přípravku Nexys 4. Vytvořte v prostředí jazyka VHDL funkční programy pro obsluhu připojených čidel, vyčítání naměřených údajů a jejich zobrazení pomocí displeje na přípravku nebo stavových LED diod. Využijte přepínače na přípravku pro přepínání výstupů čidel na displej.

Seznam odborné literatury:

- [1] Pong, P.Ch.: FPGA Prototyping by VHDL Examples (3rd edition). John Wiley & Sons, Inc., New Jersey 2008. ISBN 978-0-470-18531-5.
- [2] Pinker, J.; Poupa, M.: Číslicové systémy a jazyk VHDL (1. vydání). BEN - Technická literatura, Praha 2006. ISBN 80-7300-198-5.

Vedoucí: Ing. Pavel Lafata, Ph.D.

Platnost zadání: do konce letního semestru 2014/2015

  
prof. Ing. Boris Šimák, CSc.  
vedoucí katedry



  
prof. Ing. Pavel Ripka, CSc.  
děkan

V Praze dne 9. 12. 2013

## **Poděkování**

Rád bych poděkoval vedoucímu bakalářské práce Ing. Pavlu Lafatovi, Ph.D. za zapůjčení materiálů, přípravků, čidel, za cenné rady, připomínky, za trpělivost při konzultacích a pomoci na všech mnou nalezených problémech.

**Anotace:**

Tato bakalářská práce zahrnuje popis teplotních čidel ADT7420, DS18B20 a teplotního a vlhkostního čidla DHT11. Dále se zabývá popisem komunikace z těchto čidel přes konkrétní příslušné sběrnice. Další částí této práce bylo vytvořit funkční programy, pro přípravky Digilent NEXYS 4 nebo Digilent NEXYS 3, k obsluze těchto čidel a zobrazení naměřených dat na vestavěném segmentovém displeji nebo odeslání naměřených dat po sběrnici RS-232 pomocí jazyka VHDL.

**Klíčová slova:** FPGA, VHDL, Digilent NEXYS 4, ADT7420, DS18B20, DHT11, I<sup>2</sup>C One-Wire, sedmi segmentový displej, UART

**Summary:**

This bachelor thesis contains description of temperature sensors ADT7420, DS18B20 and temperature and humidity sensor DHT11. Furthermore, this thesis includes description of communication with these sensors, through specific relevant bus. Another part of this thesis is focused on creating VHDL codes for product board Digilent NEXYS 4 or Digilent NEXYS, to operate with these sensors and to display measured data on the built-in display or to send this measured data through RS-232 bus.

**Key words:** FPGA, VHDL, Digilent NEXYS 4, ADT7420, DS18B20, DHT11, I<sup>2</sup>C, One-Wire, Seven-segment display, UART

# Obsah

1. Cíle .....	4
2. Popis přípravků Digilent Nexys .....	5
2.1 Digilent Nexys 3.....	5
2.2 Digilent Nexys 4.....	5
3. Programovatelné logické obvody .....	6
3.1 CPLD (Complex Programmable Logic Device) .....	6
3.2 FPGA (Field Programmable Gate Array).....	6
4. Použité sběrnice.....	7
4.1 I <sup>2</sup> C .....	7
4.1.1 Princip přenosu na sběrnici I <sup>2</sup> C .....	7
4.2 One-Wire .....	8
4.2.1 Princip přenosu na sběrnici One-Wire.....	9
4.2.2 Příkazy na sběrnici One-Wire („ROM commands“) .....	10
4.3 UART.....	11
5. Popis použitých čidel .....	12
5.1 ADT7420 .....	12
5.2 DS18B20 .....	14
5.3 DHT11 .....	16
6. Použité nástroje .....	19
6.1 Jazyk VHDL.....	19
6.2 Xilinx ISE .....	19
6.3 Analyzátor Sigma.....	19
7. Popis postupu řešení práce .....	20
7.1 Sedmi-segmentový displej .....	20
7.2 Řízení sedmi segmentového displeje .....	21
7.3 Řízení obvodu UART .....	22
7.4 Řízení čidla ADT7420.....	25
7.5 Řízení čidla DS18B20 .....	27
7.6 Řízení čidla DHT11.....	28
7.7 Komunikace mezi bloky.....	30
8. Závěrečné zhodnocení .....	32

## **Seznam Obrázků:**

Obrázek 1.1 – Přípravek Digilent NEXYS 4 společně s čidly DHT11, DSB1820 a ADT7420

Obrázek 2.1.1 - Přípravek Digilent Nexys 3

Obrázek 2.2.1 - Přípravek Digilent NEXYS 4

Obrázek 4.1.1.1 – Typický průběh komunikace na I<sup>2</sup>C

Obrázek 4.2.1.1 – časový diagram pro inicializační pulzy (reset, presence)

Obrázek 4.2.1.2 – Časový diagram pro logické pulzy

Obrázek 5.1.1 – Schéma zapojení čidla ADT7420

Obrázek 5.1.2 – Znárodnění pamětí a jejich stavů po zapnutí čidla

Obrázek 5.2.1- Podoba čidla DS18B20

Obrázek 5.2.2 – Znárodnění paměti čidla DS18B20

Obrázek 5.3.1 – Podoba krytu čidla DHT11

Obrázek 5.3.2 – Typický průběh komunikace

Obrázek 5.3.3 – Detailně znárodněné průběhy bitů

Obrázek 6.3.1 – Analyzátor ASIX Sigma

Obrázek 7.1 – Schéma programu

Obrázek 7.1.1 – Znárodnění sedmi-segmentového displeje

Obrázek 7.2.1 – Blok DISPLAY\_CONTROLLER

Obrázek 7.2.2 – Hodnoty z různých čidel a) teplota z ADT7420 b)teplota z DS18B20 c)teplota z DHT11 d)relativní vlhkost z DHT11

Obrázek 7.3.1 – Přijatá data v počítači

Obrázek 7.3.2 – Komponenta pro řízení rozhraní UART

Obrázek 7.3.3 – Princip funkce automatu pro řízení odesílání dat na rozhraní UART

Obrázky 7.4.1 – Komponenta pro řízení čidla ADT7420

Obrázek 7.4.2 – Ukázka signálu uvnitř komponenty ADT

Obrázek 7.4.3 – Zjednodušená funkce automatu, je řízen signálem o 500kHz

Obrázek 7.5.1 - Komponenta pro řízení čidla DS18B20

Obrázek 7.5.2 – Zjednodušená funkce automatu pro řízení DS18B20

Obrázek 7.6.1 – Komponenta pro řízení čidla DHT11

Obrázek 7.6.2 – Zjednodušená funkce automatu k ovládní čidla DHT11

Obrázek 7.7.1 – Komponenta určená k řízení ostatních komponent

Obrázek 8.1 – Průběh analyzovaný pomocí logického analyzátoru ASIX SIGMA



## 1. Cíle

Cílem této práce bylo seznámit se s přípravky Xilinx Spartan 3E, Digilent Nexys 3, Digilent Nexys 4 a jejich obsluhou pomocí jazyka VHDL. Naučit se ovládat jejich součásti a periférie, jako je například sedmi segmentový displej u přípravků Digilent Nexys 3 a Digilent Nexys 4. Podrobněji prostudovat materiály dostupné z oficiálních stránek společnosti Digilent.

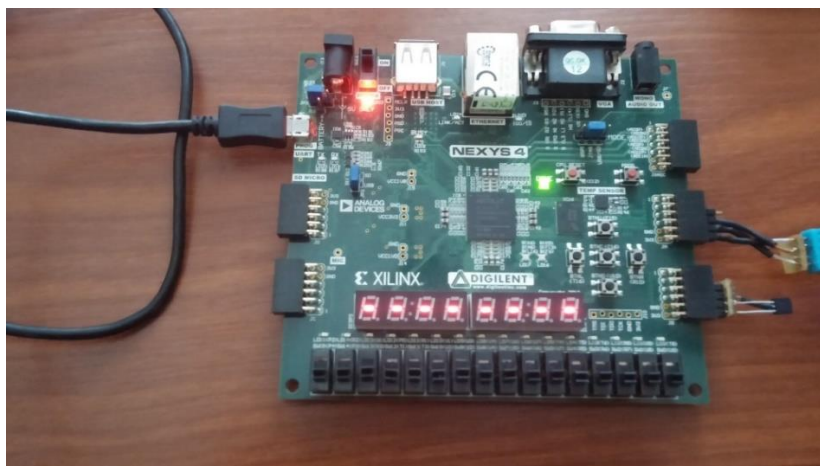
Dalším aspektem této práce bylo navržení a realizace připojení jednoduchých externích čidel fyzikálních veličin (teplota, vlhkost) k uvedeným přípravkům. Důležité pro návrh bylo studium přiložených katalogových listů od jednotlivých čidel (ADT7420, DS18B20 a DHT11) a simulace vlastní komunikace pro individuální sběrnice náležící každému čidlu.

Pro ověření správnosti návrhu a pro ladění programů jsem dostal zapůjčen logický analyzátor od firmy ASIX SIGMA. Další nástroje pro simulaci jsou zakomponovány do vývojového nástroje od firmy XILINX, jsou to například ModelSim nebo ISim.

A konečně mým úkolem bylo vytvořit funkční programy v prostředí jazyka VHDL pro obsluhu připojených čidel, vyčítání naměřených údajů a jejich zobrazení pomocí displeje na přípravku, či jejich odeslání pomocí vhodného výstupního rozhraní, jako je například RS-232.

Jelikož jsem pro svou práci neměl k dispozici čidlo tlaku (jedním z důvodů byla vysoká pořizovací cena), tak jsem se rozhodl společně po konzultacích s vedoucím práce použít dvojici teplotních čidel (DS18B20 a ADT7420) a čidlo které je schopno měřit jak teplotu, tak vlhkost (DHT11). Dále jsem vytvořil modul pro komunikaci s PC přes sběrnici RS-232 (respektive USB na straně počítače), aby bylo možné zobrazovat data i na PC.

Výsledkem mé bakalářské práce je komplexní program realizující formu meteorologické stanice. Ta je schopna měřit teplotu pomocí tří čidel, z nichž jedno je zabudované na desce přípravku, a dvě jsou připojeny externě. Dále pomocí čidla DHT11 měří relativní vlhkost vzduchu v běžném domácím prostředí (čidlo má omezenou přesnost a rozsah hodnot, viz v textu dále). Tato stanice by mohla najít využití v kancelářských nebo domácích prostorách, nebo všude tam kde jsou níže uvedená čidla schopny fungovat.



Obrázek 1.1 – Přípravek Digilent NEXYS 4 společně s čidly DHT11, DSB1820 a ADT7420

## 2. Popis přípravků Digilent Nexys

Následuje popis dvou využitých přípravků při mé práci, pro které jsem vytvářel potřebné VHD kódy. Při zpracovávání jsem čerpal z [1] pro Digilent Nexys 4 a [2] pro Digilent Nexys 3.

### 2.1 Digilent Nexys 3

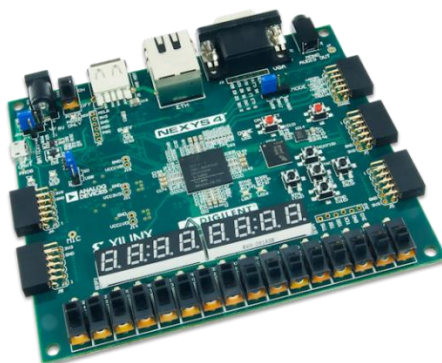
Digilent Nexys 3 je vývojová platforma založená na integrovaném obvodu Xilinx Spartan-6 LX16 FPGA. Také obsahuje několik konektorů pro různé účely: čtyři 12-pinové Pmod<sup>TM</sup> konektory, vysokorychlostní 40-pinový konektor VHDC, VGA port, 10/100 Ethernet, několik USB portů určených k připojení zařízení nebo k programování přípravku. Možností programování přípravku je několik, nejsnazší je způsob programování přes USB2 port pomocí programu Digilent Adept. Mimo jiné přípravek obsahuje pět tlačítek, 8 přepínačů, osm LED a čtyřčíselný sedmi-segmentový displej.



Obrázek 2.1.1 - Přípravek Digilent Nexys 3 [2]

### 2.2 Digilent Nexys 4

Digilent Nexys 4 je vývojová platforma založená na integrovaném obvodu Artix-7<sup>TM</sup> FPGA od firmy Xilinx. Obsahuje několik konektorů pro různé účely: čtyři 12-pinové Pmod<sup>TM</sup> konektory, jeden XADC Pmod<sup>TM</sup> konektor, VGA port, 10/100 Ethernet, několik USB portů určených k připojení zařízení nebo k programování přípravku a také slot na microSD kartu. Možností programování je několik, nejsnazší je způsob programování přes Micro A-B USB pomocí programu ISE iMPACT. Dále také přípravek obsahuje pět tlačítek, 16 přepínačů, 16 LED, osmimístný sedmi-segmentový displej, PWM (Pulzně šířková modulace) audio výstup, dvě trojbarevné LED, PDM mikrofon, trojosý akcelerometr ADXL362 a teplotní čidlo ADT7420 (viz dále).



Obrázek 2.2.1 - Přípravek Digilent NEXYS 4 [1]

### 3. Programovatelné logické obvody

Programovatelné logické obvody, nebo také PLD (Programmable Logic Device), jsou obvody, které lze naprogramováním sestavit do číslicového obvodu. V této kapitole jsou popsány dva základní typy architektur programovatelných logických obvodů PLD, které se v dnešní době nejvíce používají. Jsou to CPLD a FPGA. Právě FPGA hradla jsem využil ve svojí práci, neboť tvoří základ obou přípravků Digilent Nexys 4 a Digilent Nexys 3. Při zpracovávání jsem čerpal z knihy [10].

#### 3.1 CPLD (Complex Programmable Logic Device)

Základní princip těchto obvodů je sestavení součinů pomocí matice členů AND po kterém následuje hradlo OR, což znamená, že logické funkce se sestavují v disjunktivní formě, a obvody dále obsahují vždy na každou matici AND a OR hradel jednu makrobuňku. Makrobuňka je hlavním blokem, ve kterém se vytvářejí sekvenční nebo kombinační logické obvody. Toto tvoří tzv. funkční bloky, které lze mezi sebou propojit centrálním spojovacím polem. Funkční bloky jsou dále připojeny na výstupní propojovací pole, které je dále připojeno na vstupně-výstupní I/O bloky.

Tyto obvody tedy obsahují následující programovatelné prvky:

- Centrální propojovací pole
- Matice členů AND
- I/O bloky
- Makrobuňky
- Výstupní propojovací pole

Většinou se tyto obvody programují pomocí rozhraní JTAG. Programování těchto obvodů je non-volatilní což znamená, že naprogramovaná funkce v obvodu zůstane i po odpojení napájení. Jednou z nevýhod těchto obvodů je omezený počet re-programování.

#### 3.2 FPGA (Field Programmable Gate Array)

Na rozdíl od obvodů CPLD a SPLD (Simple PLD, tedy jednodušší a starší varianta CPLD) se v těchto obvodech netvoří logické funkce připojením na matici hradel AND a hradla OR na makrobuňku. Tyto obvody jsou založeny na princip generátorů funkcí s pamětí (vyhledávací tabulkou LUT, což je zkratka ze slov LookUp Table), připojením na klopné obvody, vertikálními a horizontálními propojeními a na vstupní výstupní programovatelné I/O bloky. Metody tvorby těchto obvodů jsou například pomocí paměti SRAM nebo pomocí Anti-Fuse. Dále mohou být na tyto bloky připojené specializované obvody, jako jsou například násobičky či paměti.

S ohledem na využití pamětí typu RAM vyžadujících napájení dojde vždy po jeho odpojení ke ztrátě uloženého obsahu. Toto je často řešeno přídatnou pamětí EEPROM, díky které se může program zachovat v obvodu. FPGA obvody nemají obvykle omezení počtu re-programování (počet cyklů je řádově mnohem vyšší, než v případě CPLD). Tyto obvody se také často programují přes rozhraní JTAG. Jsou vhodnější pro komplexnější programy než obvody CPLD, z těchto obvodů lze vytvořit na jednom čipu několik procesorů a využívat tak výhod paralelního zpracování.

## 4. Použité sběrnice

Sběrnice je soustava vodičů, která nám zajišťuje přenos dat mezi jedním a více elektronickými zařízeními nebo obvody. Informace o jednotlivých sběrnících jsem čerpal z katalogových listů čidel[5][6][7], které tyto sběrnice používají, a z internetových zdrojů [3] a[4]. Informace o rozhraní UART jsem čerpal z [9].

Pro zabudované čidlo ADT7420 je použita sběrnice I<sup>2</sup>C integrovaná do přípravku Nexys 4, pro čidlo DS18B20 je použita sběrnice tzv. One-Wire kterou lze na přípravek připojit pomocí pinů na Pmod™ konektoru. Vlhkostní a teplotní čidlo DHT11 ke komunikaci s přípravkem používá vlastní jedno-vodičovou sběrnici (nepočítá se napájecí vodič a zemnicí vodič). Ke komunikaci mezi přípravkem a počítačem jsem zvolil sériovou komunikaci USB UART pomocí vestavěného FTDI FT2232HQ USB-UART bridge. Připojení k počítači lze realizovat připojením přes terminál na virtuální COM port v počítači.

### 4.1 I<sup>2</sup>C

I<sup>2</sup>C je zkratka pro Internal-Integrated-circuit bus (IIC). Z názvu této sběrnice vyplývá, že jde o vnitřní sběrnici, která slouží pro přenos dat nebo pro komunikaci mezi jednotlivými integrovanými obvody většinou uvnitř jednoho zařízení. Tuto sběrnici vyvinula pro svá zařízení společnost Philips v roce 1982 a postupně byla dále rozšiřována. V dnešní době tuto sběrnici podporuje nespočet integrovaných obvodů, jako jsou LCD, mikro kontroléry a DAC ADC převodníky, a mimo jiné jím lze ovládat i čidlo teploty ADT7420.

Přenos po sběrnici I<sup>2</sup>C, který je obousměrný, probíhá pomocí dvou vodičů SDA a SCL. Obousměrný přenos je zajištěn tak, že v určitý moment může vysílat pouze jedno zařízení na sběrnici. Což může být velmi výhodné z důvodu počtu vývodů jednotlivých obvodů nebo zařízení. Vodič SDA (Serial Data) je datový kanál a vodič SCL (Serial Clock) přenáší hodinový takt od Mastera na sběrnici. Oba vodiče jsou přes „pull-up“ rezistor připojené na napájení k zajištění vysoké úrovně na obou vodičích (tzv. stav vysoké impedance), také jsou oba typu otevřený kolektor z elektrického hlediska. Délka přívodů této sběrnice a zároveň i nejvyšší počet připojených zařízení nebo obvodů je omezen nejvyšší přípustnou kapacitou 400 pF. Sběrnice I<sup>2</sup>C může pojmout více integrovaných obvodů. Všechny obvody na sběrnici I<sup>2</sup>C jsou adresovány 7bity respektive 10bity v rozšířené verzi, což nám teoreticky umožňuje připojit 128 nebo 1024 zařízení na tuto jednu společnou sběrnici I<sup>2</sup>C.

Na sběrnici lze pro komunikaci použít frekvence jen do 400 kHz a v případě, že některá zařízení nedokáží pracovat s touto frekvencí, ale pouze s frekvencí nižší, tak se pro tuto sběrnici použije nejnižší povolená frekvence. Dále je dána maximální doba náběžné a sestupné hrany bitů (logických stavů 1 a 0) sběrnice (jak vodiče SDA a SCL) a to na 0,3 μs.

#### 4.1.1 Princip přenosu na sběrnici I<sup>2</sup>C

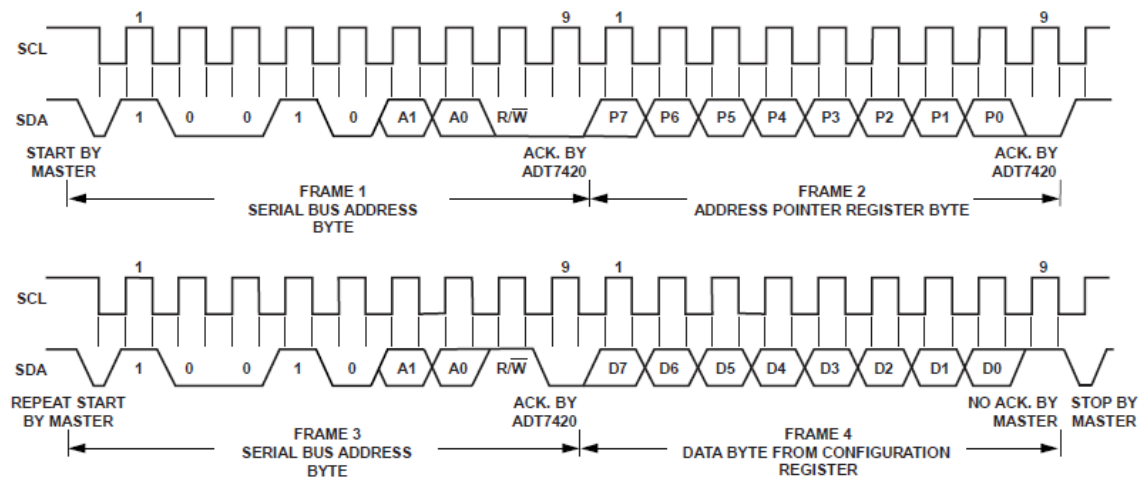
Master zařízení vysílá SCL takt pro všechny ostatní zařízení, SCL se vysílá v takovém kmitočtu, v jakém je nastaveno nejpomalejší zařízení. Sběrnice I<sup>2</sup>C je multi-masterová, to znamená, že takovýto Master zařízení může být na sběrnici více. Když jedno zařízení na sběrnici vysílá, všechny ostatní zařízení přijímají. Jen podle adresy si každé zařízení nebo obvod určí, zdali jsou data určená tomuto zařízení. Co se týče vysílání

na vodiči SDA, tak nejprve se vyšle adresa a současně s ní indikace operace (čtení/zápis, R/W, 1/0). Při přenosu po I<sup>2</sup>C se sběrnice nachází v několika stavech:

- **Klidový stav** - oba vodiče jsou ve stavu vysokých napěťových úrovní (to je zajištěno rezistory viz výše), to znamená, že žádný Master nevysílá SCL a tudíž neprobíhá přenos.
- **Start bit** jak již říká název, zahajuje přenosovou část a to tak, že SCL = 1 a SDA se změní z 1 na 0.
- **Přenos dat** vždy je přenášen 1 B (8 bitů), SDA se mění pouze, když SCL = 0, v každém taktu jeden bit.
- **Acknowledgment (ACK) bit** jako devátý bit, když je vše správně a zařízení je připraveno pro přenos dalšího bajtu, tak nastaví na SDA logickou nulu, v opačném případě pokud zařízení ukončuje přenos, nastaví na SDA logickou jedničku.
- **Stop bit** ukončuje přenos, použije se pouze když ACK nastaví na SDA logickou jedničku, stejně jako **Start bit**.

Sběrnice I<sup>2</sup>C nekontroluje přenos, je zde pouze ACK, který jen potvrzuje dokončení přenosu, nekontroluje se však, zda byl bit správně přenesen.

Na následujícím obrázku je vidět typický průběh komunikace na sběrnici I<sup>2</sup>C. První průběh zobrazuje zápis přes sběrnici do zařízení. V tomto případě jde o zápis adresy registru, ze kterého se bude číst u čidla ADT7420, skládající se z bitů A1 a A0 v adresové části (viz v kapitole 5.1 o tomto čidle). Druhý průběh znázorňuje naopak čtení ze zařízení.



Obrázek 4.1.1.1 - Typický průběh komunikace na I<sup>2</sup>C [5]

## 4.2 One-Wire

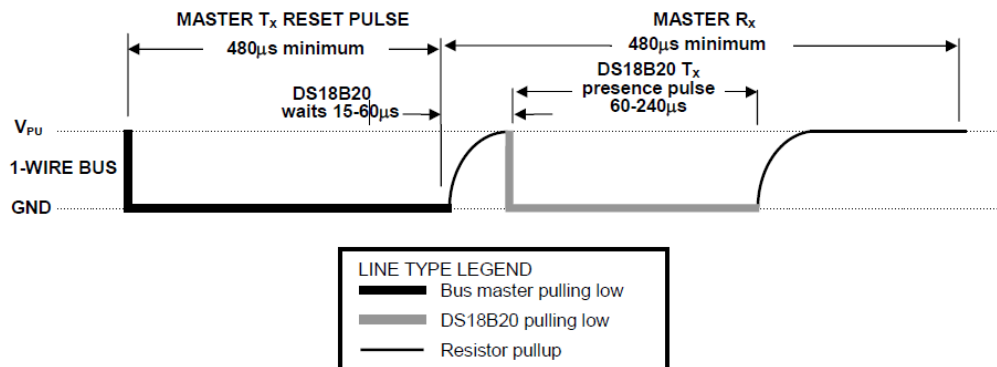
Sběrnice One-Wire je typ sběrnice která pro samotný přenos dat a časování používá jen jeden vodič (jeden „drát“, odtud One-Wire). Tato sběrnice byla vyvinuta firmou Dallas Semiconductor. Oproti sběrnici I<sup>2</sup>C je přenos komplexnější, zřejmě z důvodu méně vodičů. Mimo jiné všechna zařízení na One-Wire jsou připojena přes společný zemnicí

vodič. Napájení zařízení na sběrnici může být zajištěno dvěma způsoby. Zařízení mohou být napájena parazitně přes datový vodič, nebo lze použít samostatný napájecí vodič.

Obdobně jako u sběrnice I<sup>2</sup>C je implicitně na sběrnici One-Wire nastavena logická vysoká úroveň a na datový vodič je zapojen „pull up“ rezistor k nastavování této logické úrovně („zvedání“, odtud „pull up“). Na rozdíl od sběrnice I<sup>2</sup>C není sběrnice One-Wire typu multi-master, takže je zde pouze jeden obvod typu Master a ostatní jsou typu Slave. Dále lze tuto sběrnici použít pro „Multi-Drop“ aplikaci nebo „Single-Drop“, tzn., že je připojeno více zařízení nebo jedno zařízení na jednu sběrnici.

#### 4.2.1 Princip přenosu na sběrnici One-Wire

Základem komunikace je tzv. „Reset pulse“ který vždy zahajuje zařízení Master na sběrnici. Což znamená, že se na sběrnici nastaví logická nula po dobu minimálně 480  $\mu$ s. Následně sběrnici „pustí“ (nastaví se stav vysoké impedance, tzn. sběrnice je buď tří stavová, nebo s otevřeným kolektorem) a čeká, zda přijde prezenční impuls, který oznamuje, že je na sběrnici připojeno nějaké zařízení nebo obvod, viz následující obrázek.

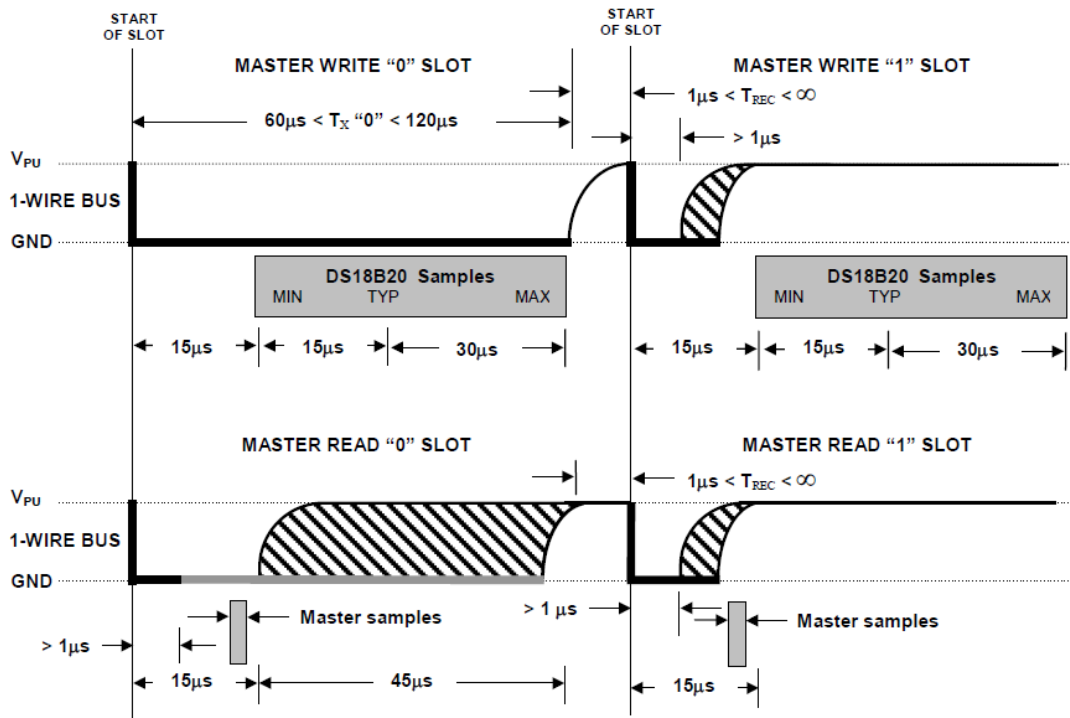


Obrázek 4.2.1.1 – časový diagram pro inicializační pulzy (reset, presence) [6]

Po sekvenci pulzů reset a prezenze lze komunikovat pomocí tzv. „time slotů“, jež vždy iniciuje Master (je jedno, jestli jde o čtení nebo zápis). Tyto „sloty“ jsou dlouhé minimálně 60  $\mu$ s a maximálně 120  $\mu$ s, s tím že mezera mezi jednotlivými „time sloty“ je minimálně 1  $\mu$ s. Existují čtyři typy slotů:

- **Zápis log. 1** – Master stáhne sběrnici na alespoň 1  $\mu$ s na logickou nízkou úroveň a po nejdéle 15  $\mu$ s ji pustí na logickou 1 a nechá po minimálně 45  $\mu$ s na jedničce.
- **Zápis log. 0** - Master stáhne opět sběrnici na logickou nulu ale tentokrát po celou dobu trvání „time slotu“ (tzn. na 60  $\mu$ s).
- **Čtení log. 1** – Master opět stáhne, na alespoň 1  $\mu$ s, sběrnici na logickou nízkou úroveň, ale do cca. 6  $\mu$ s jí pustí, zařízení si samo nastaví logickou jedničku. Master pak vzorkuje signál do cca. 15  $\mu$ s od začátku „time slotu“.
- **Čtení log. 0** – analogicky jako u čtení logické jedničky, až na to, že zařízení stáhne sběrnici vysokou logickou úroveň po 15  $\mu$ s od začátku „time slotu“.

Vše je znázorněno na následujícím obrázku.



Obrázek 4.2.1.2 – Časový diagram pro logické pulzy [6]

Zařízení na sběrnici se identifikují pomocí ROM (Read Only Memory) paměti, kde je uložena jejich 64bitová adresa. Toto unikátní číslo obsahuje v prvních osmi bitech signalizaci rodiny zařízení (u DS18B20 je to 0x28), dalších 48 bitů je sérové číslo zařízení a posledních 8 bitů je kód CRC (Cyclic Redundancy Check). Jako první komunikace po prezenčním pulzu Master vyšle tzv. „ROM command“, pomocí kterých zjistí nebo adresuje zařízení na sběrnici, tento krok lze přeskočit, je-li jen jedno zařízení připojené na sběrnici. Po tomto kroku následuje většinou příkaz pro určité zařízení na sběrnici a odpověď na tento příkaz ze zařízení. Následně se relace přeruší pomocí „reset pulzu“, nebo se po dokončení komunikace nechá sběrnice puštěná. Resetovací pulz může přijít kdykoliv během komunikace.

Komunikace probíhá většinou v následujících krocích (existují výjimky, jako je například hledání alarmů u čidla DS18B20):

- Inicializace (resetovací pulz a prezenční pulz)
- „ROM command“ (například ROM skip, nebo Read ROM)
- Funkční příkaz (u čidla DS18B20 je to například převod teploty)

#### 4.2.2 Příkazy na sběrnici One-Wire („ROM commands“)

Jako první se musí zjistit 64bitová adresa zařízení, díky čemuž lze poté adresovat jedno specifické zařízení na sběrnici. K tomuto slouží příkaz „Search ROM“ (0xF0), který se provede tolikrát, aby byla identifikována všechna zařízení na sběrnici pomocí procesu eliminace. Je-li na sběrnici pouze jedno zařízení, lze použít jednodušší příkaz

„*Read ROM*“ (0x33), který získá 64bitovou adresu jednoho zařízení, jestliže je na sběrnici více zařízení, dojde tím však ke kolizi dat na sběrnici.

Po zjištění adresy lze zadat příkaz „*Match ROM*“ (0x44), díky čemuž se adresuje jedno specifické zařízení a ostatní zařízení budou čekat, než přijde resetovací pulz. Chceme-li adresovat všechna zařízení na sběrnici, lze použít příkaz „*Skip ROM*“ (0xCC). Tento příkaz lze také použít, když je sběrnice typu „*single-drop*“. To je tehdy, když je na sběrnici připojeno jen jedno zařízení, ke kterému tak lze přistupovat i bez znalosti jeho adresy. Je však nutné počítat s tím, že když se daný příkaz použije pro „*Multi-drop*“, dojde ke kolizi dat podobně jako u příkazu „*Read ROM*“.

Posledním příkazem je „*Alarm Search*“ (0xEC), který pomáhá při vyhledávání poplachových stavů na sběrnici. Při tomto příkazu se postupuje podobně, jako v případě „*Search ROM*“, pouze ta zařízení, která generovala poplachový stav, budou reagovat na tento příkaz. Po dokončení se Master musí vrátit ke kroku inicializace (resetovací a prezenční impuls).

### **4.3 UART**

UART (Universal Asynchronous Receiver Transmitter) neboli univerzální asynchronní přijímač/vysílač, je obvod realizujícího sériové asynchronní rozhraní. Většinou se používá ve spojení se standardy, jako jsou RS-232, RS-422 nebo RS-485. V mojí práci jsem jej použil pro odesílání dat z čidla, ale nikoliv pro jejich příjem. UART rozhraní v mém případě byl použit ve spojení s obvodem FTDI FT2232HQ USB-UART bridge, který převáděl komunikace ve stylu přenosu na RS-232 přes rozhraní USB do počítače, přes který lze zobrazit virtuální sériové rozhraní počítače. Komunikace probíhala tak, že jako první se odesílá start bit, poté 8 bitů dat (od nejméně významného bitu, nebo také LSB, po nejvýznamnější bit, nebo také MSB) a na závěr byl odeslán jeden stop bit bez jakýchkoliv paritních bitů.



## 5. Popis použitých čidel

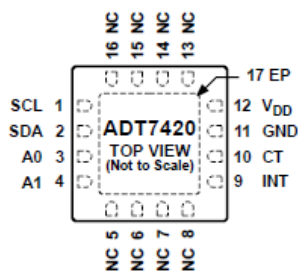
Následuje popis čidel, která mají být použita, dvojice čidel teploty (ADT7420 a DS18B20) a čidlo, které měří jak vlhkost, tak teplotu (DHT11). Informace o čidlech jsem čerpal z individuálních katalogových listů [5] [6] [7].

### 5.1 ADT7420

Toto čidlo je integrováno do desky Nexys 4 a komunikace s ním probíhá přes sběrnici I<sup>2</sup>C. Toto čidlo má rozsah teplot od -20 °C do 105 °C s přesností ±0,25 °C (při napájecím napětí 3,3 V) a v rozsahu od -10 °C do 85 °C je přesnost ±0,2 °C (při napájecím napětí od 2,7 V do 3,3 V). Hystereze teploty je ±0,002 °C. Rozlišení teploměru je implicitně nastaveno na 13 bitů (0,0625 °C), lze přenastavit na rozlišení 16bitů (0,0078 °C).

Čidlo obsahuje vstupy A0 a A1 k nastavení adresy čidla, což znamená, že čidlo ADT7420 má 4 použitelné adresy. LSB adresy se mění vstupy A0 a A1 a to tak, že se buď připojí, nebo odpojí. Na přípravku Digilent Nexys 4 jsou tyto piny pevně zapojeny a signalizují oba vysokou logickou úroveň. Dále je čidlo opatřeno pinem CT, který je typu otevřený kolektor a stává se aktivním, přesáhne-li teplota naprogramovanou kritickou hodnotu a také pin INT, který obdobně informuje o přesažení naprogramované hodnoty, k těmto dvěma pinům by měl být připojen 10 kΩ „pull-up“ rezistor.

Na následujícím obrázku je zobrazeno čidlo ADT7420, se všemi vývody, vývody NC nejsou zapojené, přívod EP je většinou uzemněn nebo nechán nepřipojen. Vývod GND a V<sub>DD</sub> slouží k připojení čidla k zemi a napájení. Piny SCL a SDA jsou pro účely sběrnice IIC (*Serial Clock* a *Serial Data* viz kapitola 4.1 o I<sup>2</sup>C). Piny A0, A1, INT a CT jsou popsány v odstavci výše.



Obrázek 5.1.1 – Schéma zapojení čidla ADT7420 [5]

Čidlo lze nechat běžet v různých módech:

- **Normální mód** – implicitní mód, při spuštění je provedené „rychlé měření“, které trvá typ. 6 ms (má přesnost typicky ±5 °C), poté probíhá měření, které trvá 240 ms. Toto měření se provádí neustále, po dokončení měření je hodnota teploty zapsána do registru a probíhá další měření.
- **One-shot mód** – tento mód se musí nastavit, po nastavení proběhne jedno měření a pak se přepne do Shutdown módu (viz dále). K nastavení tohoto módu

se musí vyčkat nejdříve 240 ms po zapnutí, aby bylo jisté, že se dokončil předchozí převod teploty.

- **SPS mód** – tento mód provádí jedno měření za sekundu (čidlo změří teplotu a čeká do konce jedné sekundy a pak měří znovu).
- **Shutdown mód** – v tomto módu se neprovádí žádné měření, ale lze číst a zapisovat do registrů čidla.

Převod teploty se provádí pomocí zabudovaného sigma-delta převodníku ( $\Sigma$ - $\Delta$ ). Externí hodinový generátor není potřebný (kromě čtení a psaní na sběrnici IIC), protože si čidlo generuje svůj vlastní takt.

Pro napájení čidla ADT7420 je vyžadováno napětí v rozpětí od 2,7 V do 5,5 V. Při napájení 3,3 V je průměrný napájecí proud okolo 210  $\mu$ A, maximálně však 265  $\mu$ A (při 5,5 V to jsou typicky 250  $\mu$ A a maximálně 300  $\mu$ A), v shutdown režimu jsou to typicky 2  $\mu$ A. Výkonová ztráta v normálním módu je 700  $\mu$ W (při teplotě 25 °C a napětí 3,3 V).

Pro datové vstupy čidla (SDA, SCL) platí, že mají vstupní kapacitu typicky 2 pF, maximálně však 10 pF (výstupní kapacita pinu SDA je také 2 pF) a jsou typu otevřený kolektor. Hodnota, po které se vyhodnotí nízká logická úroveň (logická nula), je maximálně 0,3 krát napětí napájecího vodiče a hodnota, po které se vyhodnotí vysoká logická úroveň (logická jednička), je minimálně 0,7 krát hodnota napájecího napětí. Pro logické vstupy A0 a A1 je nízká úroveň maximálně 0,4 V a vysoká úroveň minimálně 2 V. Pro potlačení šumových špiček, menších typicky než 50 ns, je na datové logické vstupy připojen filtr.

Čidlo ADT7420 obsahuje 14 registrů (každý po 8 bitech), 9 registrů pro záznam hodnot teploty, registr konfigurace (uvnitř tohoto registru se mění nastavení registru, například mód běhu čidla nebo rozlišení), registr softwarového resetu (slouží k resetování nastavení čidla a čidlo poté neodpovídá 200  $\mu$ s), registr s ukazatelem na adresu (do tohoto registru se zapisuje jako první, ukazuje na registr, s kterým při transakci budeme pracovat), registr statusu (ukazuje zda se dokončilo měření, nebo jestli byla překročena nějaká z nastavených hodnot, viz odstavec níže) a poslední registr uchovávající ID čidla. Oba poslední registry jsou určeny jen pro čtení a nelze do nich zapisovat.

Register Address	Description	Power-On Default
0x00	Temperature value most significant byte	0x00
0x01	Temperature value least significant byte	0x00
0x02	Status	0x00
0x03	Configuration	0x00
0x04	T <sub>HIGH</sub> setpoint most significant byte	0x20 (64°C)
0x05	T <sub>HIGH</sub> setpoint least significant byte	0x00 (64°C)
0x06	T <sub>LOW</sub> setpoint most significant byte	0x05 (10°C)
0x07	T <sub>LOW</sub> setpoint least significant byte	0x00 (10°C)
0x08	T <sub>CRIT</sub> setpoint most significant byte	0x49 (147°C)
0x09	T <sub>CRIT</sub> setpoint least significant byte	0x80 (147°C)
0x0A	T <sub>HYST</sub> setpoint	0x05 (5°C)
0x0B	ID	0xCB
0x2F	Software reset	0xFF

Obrázek 5.1.2 – Znárodnění paměti a jejich stavů po zapnutí čidla [5]

Z 9 teplotních registrů jsou první dva registry určeny pro samotnou naměřenou teplotu a jsou určeny jen pro čtení. Při rozlišení 13 bitů jsou tři LSB použity pro signální bity pro poplachové registry (viz dále), při rozlišení 16 bitů jsou tyto LSB použity pro uchování teploty. U obou rozlišení je MSB použit jako signálizace znaménka. Teplota je zaznamenána do těchto registrů ve formátu dvojkových doplňků. U čtení teploty z 16bitového registru se automaticky po přečtení nejvýznamnějšího bajtu inkrementuje ukazatel adresy, takže lze okamžitě přečíst nejméně významný bajt.

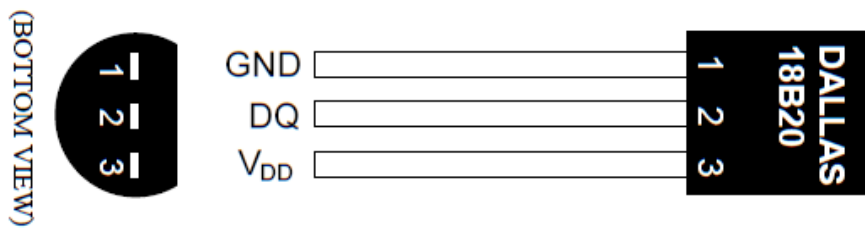
Další teplotní registry jsou T<sub>CRIT</sub>, T<sub>LOW</sub>, T<sub>HIGH</sub> (všechny tyto registry jsou 16bitové, takže ve skutečnosti jde o 6 registrů po 8 bitech a jsou rozdělené obdobně jako registr uchovávající hodnotu teploty) a T<sub>HYST</sub> (tento registr je pouze 8bitový). Čtení z těchto 16bitových registrů je obdobné jako čtení z 16bitového registru uchovávající hodnotu teploty. Při klesnutí naměřené teploty pod teplotu uloženou v registru T<sub>LOW</sub> (implicitně nastaven na 10 °C) a při překročení teploty uložené v registru T<sub>HIGH</sub> (implicitně nastaven na 64 °C) se aktivuje pin INT. Obdobné je to u registru T<sub>CRIT</sub> (ten je implicitně nastaven na hodnotu 147 °C), kde se aktivuje pin CT. Hodnoty v těchto registrech jsou uloženy ve formátu dvojkových doplňků (stejně jako u naměřené teploty), také se společně s aktivací pinů INT a CT změni příslušné hodnoty v registru statusu. Poslední registr T<sub>HYST</sub> uchovává hodnotu hystereze pro tři (respektive 6) předchozí registry. Funguje tak, že je hodnota odečtena z registrů T<sub>CRIT</sub>, T<sub>HIGH</sub> a přičtena k hodnotě T<sub>LOW</sub>.

Za zmínku také stojí tzv. „Fault Que“, který se nastavuje v konfiguračním registru a který udává, kolikrát může měření překročit jeden poplachových registrů, aniž by se aktivoval pin INT nebo CT, maximální hodnota je ovšem 4 překročení.

## 5.2 DS18B20

Čidlo od společnosti Dallas Semiconductor využívá sběrnice One-Wire, vyvinuté touto firmou. Obsahuje 64bitovou paměť ROM s unikátní adresou, díky němuž lze toto zařízení adresovat na sběrnici One-Wire, což také umožňuje mít mnoho těchto zařízení v rámci jedné sběrnice. Zařízení One-Wire se řídí pomocí tzv. „ROM Command“ (viz vysvětlení komunikace na sběrnici One-Wire v podkapitole 3.2).

Na následujícím obrázku je vidět podoba čidla, jeho výstupů, výstup GND (zemní vodič) výstup  $V_{DD}$  (napájecí vodič) a DQ (vývod pro sběrnici One-Wire).



Obrázek 5.2.1- Podoba čidla DS18B20 [6]

Rozsah měřených teplot je od  $-55\text{ }^{\circ}\text{C}$  do  $125\text{ }^{\circ}\text{C}$  s přesností  $\pm 0,5\text{ }^{\circ}\text{C}$  v rozsahu od  $-10\text{ }^{\circ}\text{C}$  do  $85\text{ }^{\circ}\text{C}$ , mimo tento rozsah je přesnost  $\pm 2\text{ }^{\circ}\text{C}$ . Rozlišení je nastavitelné od 9 do 12 bitů, přičemž rozlišení 12 bitů je nastaveno implicitně. Po zadání příkazu „Convert T“ se převádí teplota a ukládá se do jednoho ze dvou-bajtových registrů (LS a MS zvlášť). Číslo se ukládá ve tvaru 16-ti bitového čísla (dvojkový doplněk), kde je od patnáctého do jedenáctého bitu zapsán údaj o znaménku, tzn. při kladných hodnotách jsou rovny log. 0 a při záporných hodnotách log. 1. Ve zbylých 11 bitech je uložena hodnota teploty, kde třetí až nultý bit zaznamenává desetinnou část teploty. Po dokončení převodu teploty se zařízení přepne do nečinného stavu.

Doba převodu do binární soustavy je závislá na velikosti rozlišení, při implicitním 12 bitovém rozlišení ( $0,0625\text{ }^{\circ}\text{C}$ ) je doba převodu 750 ms. Při rozlišení 11, 10 a 9 bitů ( $0,125\text{ }^{\circ}\text{C}$ ,  $0,25\text{ }^{\circ}\text{C}$  a  $0,5\text{ }^{\circ}\text{C}$ ) je doba převodu 375 ms, 187,5 ms a 93,75 ms

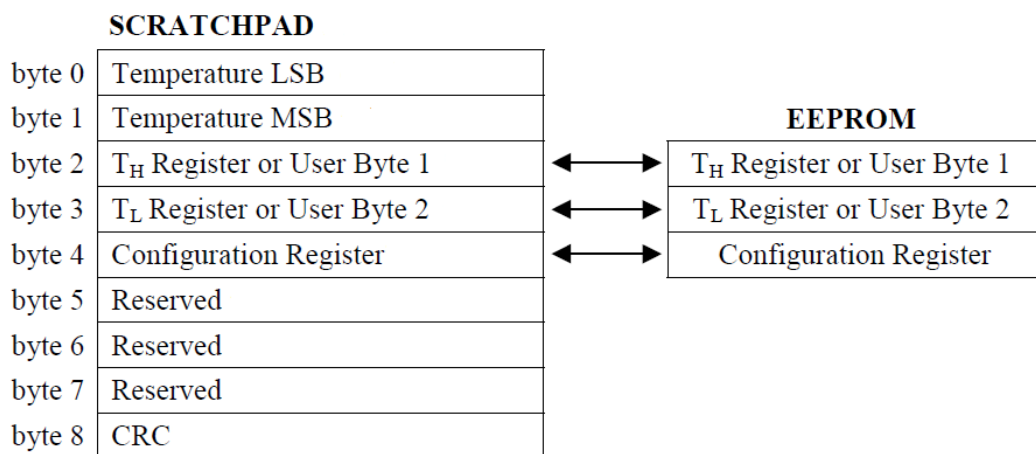
Při čekání na dokončení převodu lze generovat čtecí „time sloty“, které budou vykazovat logickou nulu v průběhu měření a logickou jedničku při dokončení převodu. Toto nelze použít při napájení čidla přes sběrnici One-Wire, jelikož při parazitním napájení musí být na sběrnici vysoká napěťová úroveň, při některých operacích (jako je převod teploty nebo zápis do paměti EEPROM) musí být na sběrnici tzv. „strong pull-up“, aby byl dodán dostatečný proud pro napájení čidla. Toto se provádí připojením MOSFET tranzistoru za pull-up rezistor na sběrnici. Také se nedoporučuje používat parazitní napájení při teplotách nad  $100\text{ }^{\circ}\text{C}$ , jelikož čidlo může mít problémy udržovat komunikaci kvůli vyšším svodovým proudům. Ke zjištění typu napájení lze použít funkční příkaz „Read Power Supply“.

Z elektrického hlediska je vodič sběrnic One-Wire (pin DQ na čidle) typu otevřený kolektor. K napájení čidla je požadováno stejnosměrné minimální napětí 3 V a maximální 5,5 V. Při parazitním napájení čidla jsou hodnoty pro napájení sběrnice stejné, při napájení přes napájecí vodič je minimální napětí 3 V a maximální napětí je stejné napětí, jaké je dodáváno přes napájecí vodič. Nízká napěťová úroveň se vyhodnocuje při přiloženém napětí mezi  $-0,3\text{ V}$  a  $0,8\text{ V}$ . Vysoká napěťová úroveň se vyhodnotí při napájení přes napájecí vodič od 2,2 V a při parazitním napájením je to 3,3 V. Maximální možná napěťová úroveň pro vysoké logické úrovni musí být menší než 5,5 V, nebo do hodnoty nižší o 0,3 V, než napájecí napětí.

Dále je do tohoto čidla zakomponovaná „zápisníková“ paměť (tzv. scratchpad), která je typu SRAM, kde se nachází dva 8bitové registry pro zápis hodnoty převedené teploty, dva registry pro alarmové hodnoty a konfigurační registr, kde lze nastavit rozlišení převodu a tím i čas převodu teploty, dále scratchpad obsahuje 3 bajty pro vnitřní použití čidla (při přečtení těchto tří registrů se nám vrátí v každém bitu log. 1 a nelze tyto registry přepsat), poslední registr této zápisníkové paměti slouží ke kontrole obsahu celé paměti a to pomocí CRC kódu. Hodnoty registrů alarmů a konfigurace jsou ukládány do zápisníkové paměti, ale pomocí příkazu „Copy scratch pad“ lze zapsat tyto registry do paměti EEPROM, takže nastavení zůstávají v platnosti i po vypnutí nebo odpojení čidla DS18B20. Naopak lze z paměti EEPROM kdykoliv nahrát nastavení konfiguračních a poplachových registrů a zapsat je do zápisníkové paměti a to pomocí příkazu „Recall E<sup>2</sup>“.

Čidlo DS28B20 je schopno při dosažení vymezených hodnot generovat poplachové stavy (alarm). Pro generování těchto stavů se musí nastavit registry  $T_H$  a  $T_L$ , kde se nastaví daná teplota po překročení respektive sestoupení pod tyto hodnoty, se v zařízení vygeneruje poplachový stav. Tyto stavy lze zjistit zadáním „ROM Command“ a to 0xEC (neboli „Alarm Search“). Při porovnávání teploty se porovnávají bity teploty od jedenáctého do čtvrtého bitu, jelikož registry  $T_H$  a  $T_L$  jsou pouze osmibitové. Registry  $T_H$  a  $T_L$  lze při nepoužívání poplašných stavů použít jako univerzální paměť.

Na následujícím obrázku je znázorněna zápisníková paměť čidla („scratchpad“) a paměť EEPROM.



Obrázek 5.2.2 – Znázornění paměti čidla DS18B20 [6]

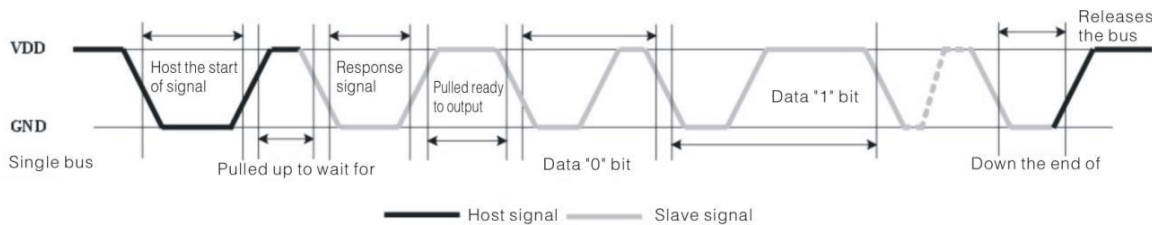
### 5.3 DHT11

DHT11 je čidlo teploty a tlaku od společnosti Aosong Electronics, toto zařízení má v sobě zakomponované kapacitní čidlo vlhkosti a termistor. Jelikož toto čidlo není od firmy Dallas Semiconductor, využívá svojí vlastní proprietární sběrnici s jedním datovým vodičem, podobnou sběrnici One-Wire. Sběrnice tohoto čidla funguje tak, že nejdřív Master pošle Start signál (podobný signálu reset u One-Wire) dlouhý alespoň 18 ms, po té se čeká na odpověď ve formě 80  $\mu$ s nízké napěťové úrovně a 80  $\mu$ s vysoké napěťové úrovně.

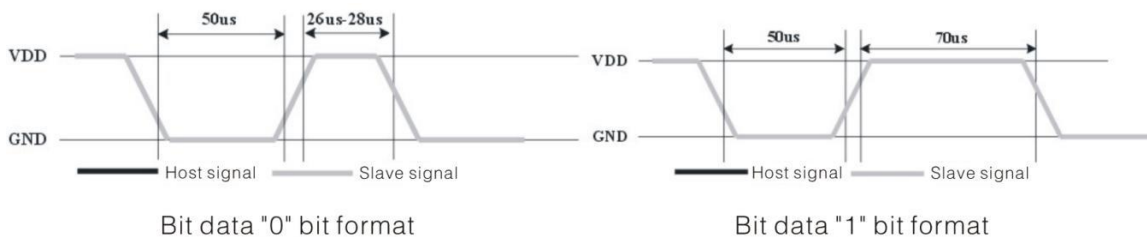


Obrázek 5.3.1 – Podoba krytu čidla DHT11 [7]

Po připojení k čidla k napájení je nutné vyčkat alespoň 1 s k přečkání nestabilního stavu, který znemožňuje vyslat jakékoliv instrukce. Rozlišení logické jedničky a logické nuly je podle doby trvání vysoké úrovně v každém intervalu. Probíhá tak, že čidlo stáhne sběrnici na nízkou napěťovou úroveň po 50  $\mu$ s a pak pull-up rezistor vytáhne sběrnici na vysokou napěťovou úroveň, na této úrovni sběrnice setrvá buď 26-28  $\mu$ s, což znamená logickou nulu, nebo 70  $\mu$ s, a to signalizuje logickou jedničku. Na následujících dvou obrázcích jsou vidět typické průběhy komunikace a logických symbolů.



Obrázek 5.3.2 – Typický průběh komunikace



Obrázek 5.3.3 – Detailně znázorněné průběhy bitů [7]

Vlastní přenos probíhá takto: 8bitů vlhkosti jako celé číslo, 8bitů vlhkosti jako desetinné číslo, 8 bitů teploty jako celé číslo, 8 bitů teploty jako desetinného čísla a nakonec 8 bitů paritních, což je součet všech předešlých. Čidlo má rozlišení nastavené na 16 bitů, ale stojí za zmínku fakt, že čidlo DHT11 neumí měřit teploty na desetinná místa, při přečtení bitů, které odpovídají desetinným místům, se vždy vrátí hodnota logická 0. To v podstatě znamená, že reálné rozlišení tohoto čidla je 8 bitů (tzn. 1 °C respektive 1% relativní vlhkosti). Toto čidlo pouze posílá naměřená data. Každé čtení teploty

je výsledek posledního měření teploty pro data, v reálném čase by se mělo provést dvoje čtení ihned po sobě, což se ovšem od výrobce nedoporučuje. Nové hodnoty jsou dostupné jednou za 2 s, což je značná nevýhoda oproti dvěma předchozím čidlům.

Datový vodič by měl být opět připojen přes třístavový port nebo port s otevřeným kolektorem a ke sběrnici musí být připojen přes 5,1 k $\Omega$  odpor. Napájecí napětí je pouze přes napájecí vodič, není zde možnost jako u čidla DS18B20 napájet a přenášet data jedním vodičem. Napájecí napětí čidla je stejnosměrných 3,5–5,5 V. Při měření teploty je napájecí proud 0,3 mA a v pohotovostním režimu je to 60  $\mu$ A. Při napájení 3,5 V se nedoporučuje mít přívody delší než 20 cm, na druhou stranu při napájení 5,5 V je tato vzdálenost do 20 m.

Doporučená pracovní teplota je přibližně od 10 do 40 stupňů a pracovní relativní vlhkost je 60% a méně. Měření vlhkosti má při teplotě 25 °C přesnost  $\pm 5\%$  relativní vlhkosti a přesnost teploty je při 25 °C  $\pm 2$  °C. Hystereze při měření vlhkosti je menší než  $\pm 0,3\%$  relativní vlhkosti. Výrobce nedoporučuje dlouhodobější vystavení suchým prostředím nebo dlouhodobější kondenzaci. Dále se nedoporučuje nasazovat v prostředí s vyšším obsahem solí ve vzduchu, kyselým nebo oxidačním plynům jako je oxid siřičitý a kyselina chlorovodíková. Také se nedoporučuje vystavovat čidlo přímému slunečnímu světlu nebo silnému ultrafialové záření po delší dobu. Výhodou tohoto čidla je nízká cena a poměrně jednoduchá komunikace s čidlem.

## 6. Použité nástroje

### 6.1 Jazyk VHDL

Zkratka VHDL znamená VHSIC (Very High Speed Integrated Circuit) Hardware Description Language, tedy jazyk pro popis velmi rychlých integrovaných obvodů. Tento jazyk vznikl v 80. letech 20. století. Využívá se jak pro popis integrovaných obvodů, tak pro jejich simulaci. Je to univerzální jazyk pro všechny obvody, ale každý obvod má svá omezení, takže výsledný program nelze často použít pro jiné obvody, to znamená, že zdrojový kód je přenositelný ovšem implementace již ne.

Informace jsem čerpal z [10] a [8].

### 6.2 Xilinx ISE

Xilinx ISE je vývojové prostředí společností XILINX, které pomocí několika nástrojů umožňuje syntézu, testování, simulaci a mnohé další činnosti, které slouží k následnému programování čipu FPGA nebo CLPD. Výčetem to jsou například nástroje iSim a ModelSim určené pro simulaci chování VHDL či Verilog kódů, nebo například ISE iMPACT určený k nahrání programu do přípravků pomocí připojení přes USB.

### 6.3 Analyzátor Sigma

Logický analyzátor od společnosti ASIX vzorkuje vstupní signál frekvencí až 200 MHz. Vstupních signálů může být až 16, ale s rostoucím počtem vstupů se jeho vzorkovací frekvence snižuje, tudíž pro všech 16 vstupů je frekvence vzorkování 50 MHz, při osmi vstupech je vzorkovací frekvence 100 MHz a pro dosažení plné vzorkovací frekvence (200 MHz) je potřeba použít pouze 4 vstupy. Dále analyzátor obsahuje 256 Mbit SDRAM paměť určenou pro ukládání vzorků průběhů nebo také 16bitový čítač. Další zajímavou funkcí je přiřazení logických funkcí na vstupy (například lze na vstup přidělit hradlo s funkcí XOR). Velkou výhodou je také dodávaný software, ve kterém lze snadno zobrazovat průběhy na jednotlivých vstupech, či nahrát tzv. „pluginy“, které jsou schopny analyzovat různé sběrnice jako je SPI nebo I<sup>2</sup>C.

Pro více informací viz [11].



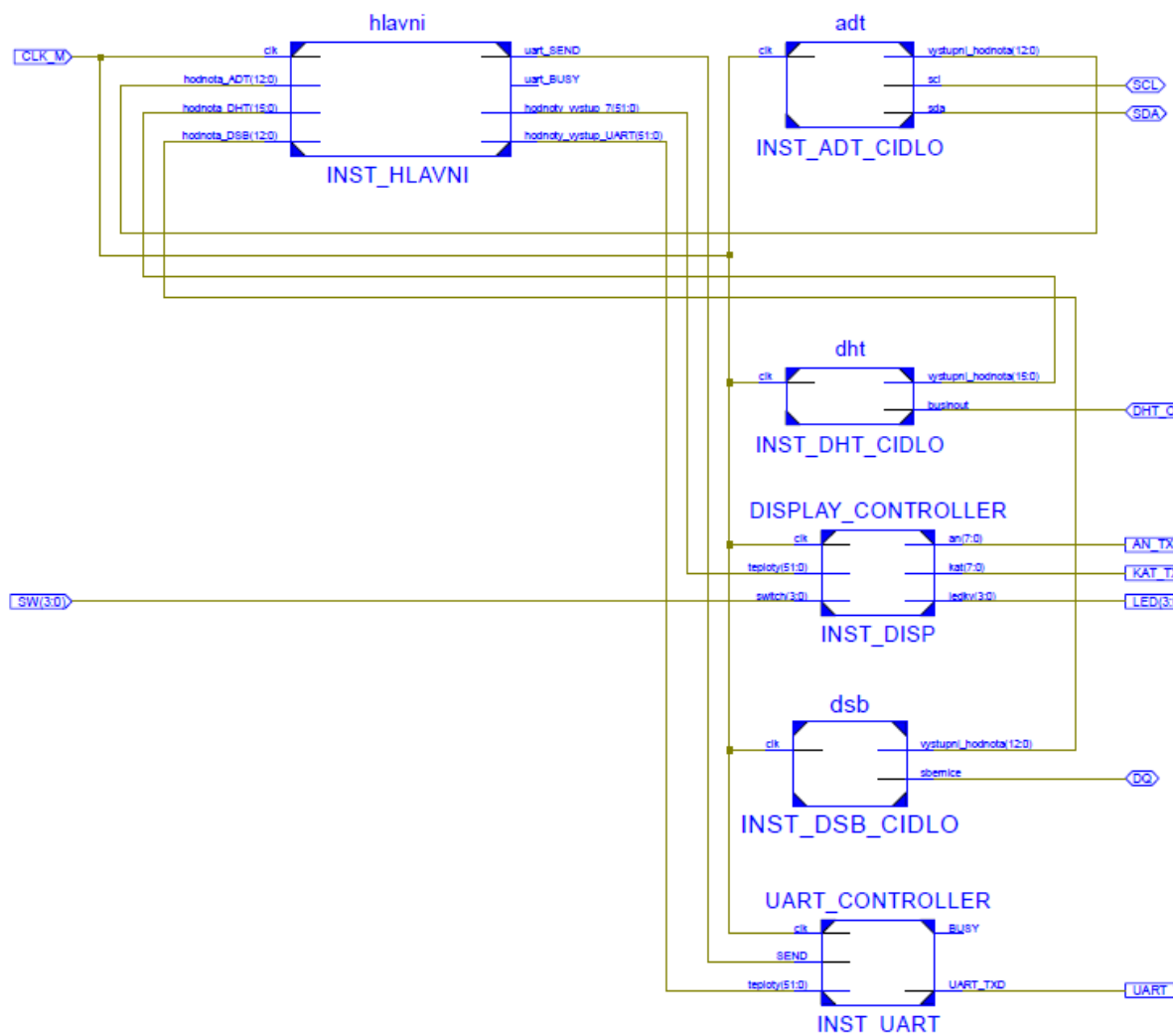
Obrázek 6.3.1 – Analyzátor ASIX Sigma [11]



## 7. Popis postupu řešení práce

V následující části jsou popsány jednotlivé bloky programu pro využití čidel na přípravku Digilent Nexys 4, který byl použitý pro všechny části programu (přípravek Digilent Nexys 3 nebyl nakonec použit z mnoha důvodů, jedním z nich je například fakt, že neobsahuje čidlo ADT7420 a má jen čtyř-místný sedmi-segmentový displej) a vysvětlení principů funkce některých součástí programů, které nebyly popsány výše. Dále je v této kapitole popsána samotná funkce celého programu.

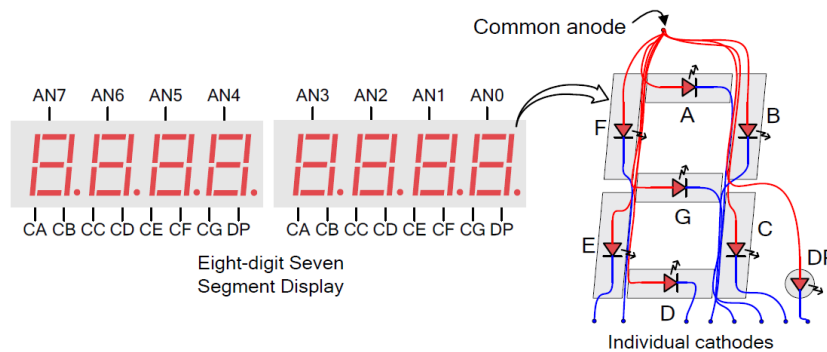
Následuje schéma složení jednotlivých bloků programu.



Obrázek 7.1 – Schéma programu

### 7.1 Sedmi-segmentový displej

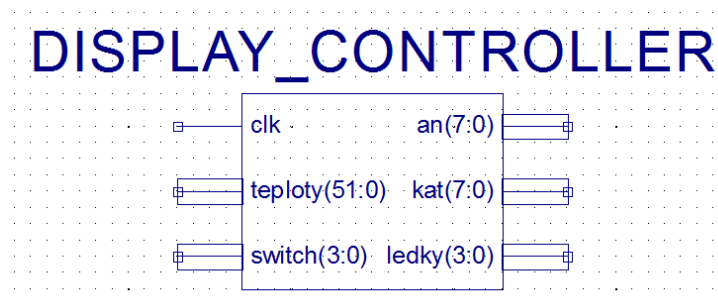
Pro rozsvícení sedmi-segmentového displeje je nutné nastavit katodové signály tak, aby byly na logické nule a anodové signály na logické jedničce. Jelikož anodové signály jsou invertovány, tak je nutné tyto signály nastavit u obou na logickou nulu u přípravku Nexys 3 a 4. K rozsvícení jednotlivých symbolů na vícečíselném sedmi-segmentovém displeji je nutné střídát anodové signály tak, aby lidské oko nepoznalo, že nejsou rozsvícené všechny symboly (frekvence blikání nejméně 50 Hz).



Obrázek 7.1.1 – Znárodnění sedmi-segmentového displeje [1]

## 7.2 Řízení sedmi segmentového displeje

Řízení sedmi-segmentového displeje probíhá v bloku DISPLAY\_CONTROLLER (na obrázku 7.1 a 7.2.1). Tento blok ve své podstatě zajišťuje dvě důležité věci. Za prvé provádí konverzi z binárního kódu do desítkové soustavy a rozřazuje jednotlivá řádová místa do individuálních proměnných. A za druhé zajišťuje zobrazení těchto proměnných na osmi-místný sedmi-segmentový displej přípravku Digilent Nexys 4.



Obrázek 7.2.1 – Blok DISPLAY\_CONTROLLER

Port clk na obrázku 7.2.1 je určen pro hodinový signál, který je poté dále upravován uvnitř tohoto bloku. Další port teploty, který je ve své podstatě polem 52 binárních hodnot, je určen pro přenos teplotních respektive vlhkostních dat z čidel do tohoto bloku, aby mohly tyto data být zobrazeny. Porty switch a ledky jsou určeny pro výběr respektive signalizaci zobrazovaného čidla. A konečně porty an a kat jsou určeny pro anodové a katodové signály, které jsou důležité pro chod samotného sedmi-segmentového displeje (viz předešlá podkapitola 7.1).

Pro účely zobrazování je v tomto bloku zakomponován čítač, který funguje jako dělička kmitočtu pro generování taktu s frekvencí 500 Hz (čili s periodou 2 ms) pro obnovování sedmi-segmentového displeje. V kombinaci s tímto čítačem pracuje proces, který zapíná jednotlivé pozice sedmi-segmentového displeje každé 2 ms.

Pro přidělení číslic jednotlivých řádových míst se nejdříve provede konverze do desítkové soustavy a poté se rozdělí číslo postupně pomocí dělení 100, 10, 1 čímž oddělím řádové čísla do samostatných proměnných, takže například z jedné proměnné 123 dostanu proměnné 1, 2 a 3. Desetinná místa se oddělují analogicky s tím rozdílem, že konverze do desítkové soustavy proběhne zvlášť pro část zobrazující

desetinnou část a vynásobí se číslem 625 a vynásobeno číslem 1000, protože s použitým rozlišením u čidel znamená LSB hodnotu 0,0625 °C. U čidla DHT11, které neumí odečítat desetinná místa, je toto také použito, ale konverze vždy vrátí hodnotu 0. Všechny hodnoty jsou zaokrouhlovány dolů na jedno desetinné místo.

Hodnoty se vypisují na jedno desetinné místo, je-li hodnota menší než 100, jinak se nevypisují desetinná místa z důvodu úspory místa na sedmi-segmentovém displeji. Znaménko se vypisuje u čidel ADT7420 a DS18B20 ale nikoliv u čidla DHT11, které neumí odečítat záporné hodnoty. Znaménko se zobrazuje pouze záporné při záporných hodnotách, ale při kladných se zobrazuje jen prázdný znak.

Pomocí přepínačů na přípravku lze přepínat mezi jednotlivými vstupy čidel, které jsou celkem čtyři. První Adt kdy se zobrazuje hodnota teploty z čidla ADT7420, dSb které zobrazuje hodnotu teploty z čidla DS18B20 a jako poslední dHtt a dHtV, které zobrazují teplotu respektive vlhkost naměřenou čidlem DHT11.



Obrázek 7.2.2- Hodnoty z různých čidel a) teplota z ADT7420  
b)teplota z DS18B20 c)teplota z DHT11  
d)relativní vlhkost z DHT11

Z obrázků je patrné, že ke každému čidlu je přiřazena jedna LED a jeden přepínač. Při stavu, kdy není přepnut ani jeden přepínač, tak se vypíší prázdné znaky ---- ---- (lze shlédnout na obrázku 1.1).

### 7.3 Řízení obvodu UART

Pro realizaci rozhraní obvodu UART byla použita komponenta UART\_TX\_CTRL.vhd dodávaná společností Digilent a běžně dostupná z internetových produktových stránek Digilent Nexys 4. Tato komponenta byla použita pro ovládání přímo UART rozhraní.

Blok UART\_TX\_CTRL používá symbolovou rychlost 9600 Bd. Data se posílají po bajtech, kde jako první se pošle start bit (logická 0) poté bajt od LSB po MSB a nakonec stop bit (logická 1).

Pro řízení tohoto bloku (UART\_TX\_CTRL) jsem vytvořil automat, který jednou za čas (přibližně 21,45s, viz níže komunikace mezi bloky) vyšle sekvenci dat, které se odesílají pomocí FTDI FT232RL USB-UART bridge po USB. Tyto data mají tvar osmi-bitových

symbolů ASCII tabulky, které lze přes vhodný terminál (Hyperterminál či např. aplikace Putty) na platformě PC zobrazit.

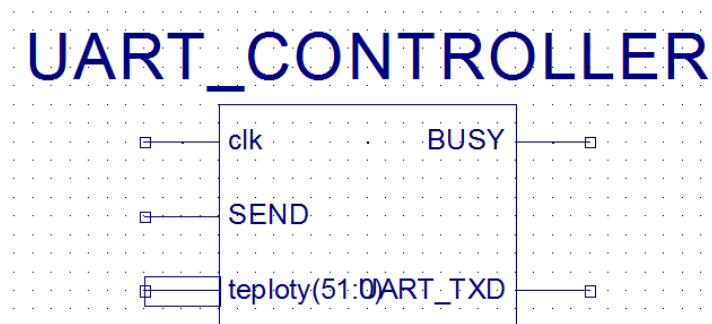
```

-----+
|      Vypis velicin      |
|      namerenych cidly  |
+-----+-----+-----+
|cidlo |velicina |hodnota |
+-----+-----+-----+
|ADT7420|teplota  | 25.2 C|
+-----+-----+-----+
|DS18B20|teplota  | 24.4 C|
+-----+-----+-----+
|DHT11  |teplota  | 24.0 C|
+-----+-----+-----+
|DHT11  |vlhkost   | 35.0% |
+-----+-----+-----+

```

Obrázek 7.3.1 – Přijatá data v počítači

Blok UAT\_TX\_CTRL a automat určený pro jeho řízení se nacházejí v komponentě UART\_CONTROLLER. V této komponentě se také provádí konverze hodnot teploty respektive vlhkosti z binárního tvaru do dekadického tvaru. Tato konverze probíhá velice podobně jako v případě řízení sedmi-segmentového displeje, až na to, že je-li číslo větší než 100, tak se stále zobrazují desetinná místa, jelikož zde není omezen zobrazovací prostor, jako tomu bylo u sedmi-segmentového displeje. Následuje obrázek komponenty pro řízení UART rozhraní.

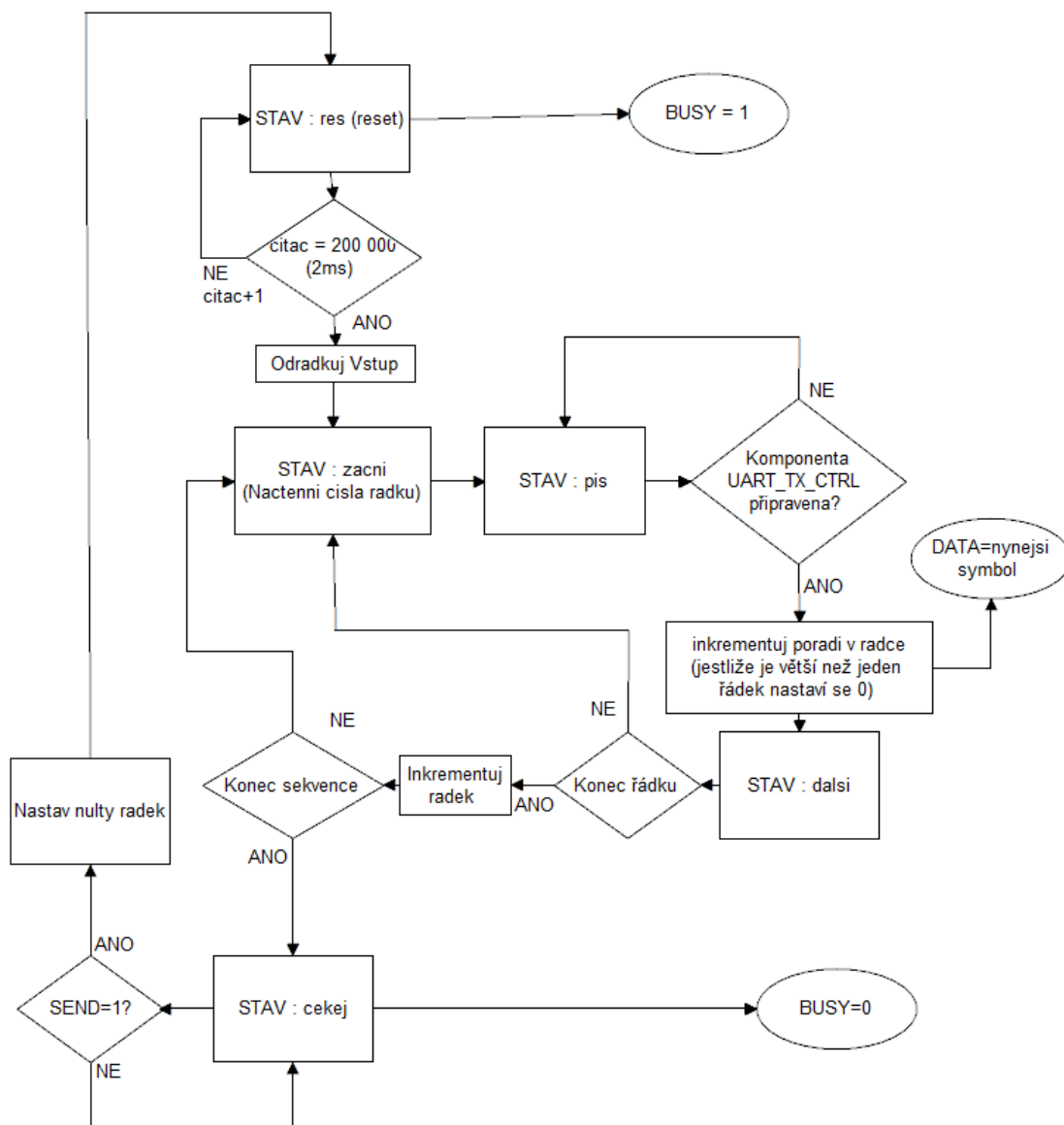


Obrázek 7.3.2 – Komponenta pro řízení rozhraní UART

Port clk slouží k přivedení hodinového pulzu. Porty BUSY a SEND slouží ke komunikaci s nadřazenou komponentou (tzn. k signalizaci, že je komponenta zaneprázdněná respektive že má komponenta vyslat sekvenci). Port teploty je podobný jako u komponenty použité pro sdružení teplotních respektive vlhkostních dat z jednotlivých čidel. A konečně port UART\_TXD, jenž je určen pro odesílání dat přes rozhraní UART.

Automat určený k ovládání odesílání dat na rozhraní UART se implicitně nachází ve stavu „res“ (zkráceně reset), v němž přečká stav nestability na sběrnici, který nastává po naprogramování přípravku a později jen přidává 2 ms prodlevu, než se začnou vysílat data. Po přečkání 2 ms se do řádkového registru načtou symboly pro odřádkování

a nastaví se stav „zacni“. V tomto stavu se načte aktuální řádek a nastaví se další stav „pis“. V tomto stavu se pošle do komponenty UART\_TX\_CTRL osmi-bitový znak a inkrementuje se index pořadí na řádku (pouze v případě je-li komponenta připravena, jinak se čeká, než bude připravena). Přepne se do stavu „dalsi“, kde se zjišťuje, zda se již došlo na konec řádku. Jestliže se ještě na konec řádku nedošlo, tak se automat vrací do stavu „zacni“. V opačném případě se inkrementuje index řádku a testuje se, jestli se nedošlo na konec vysílané sekvence. V případě že ne, tak se automat přepne do stavu „zacni“. Po odvysílání celé sekvence řádků se automat přepne do stavu „cekej“, ve kterém bude automat čekat, dokud nepřijde signál na odvysílání nových dat od nadřazené řídicí komponenty. Na následujícím obrázku je znázorněna funkce tohoto automatu.

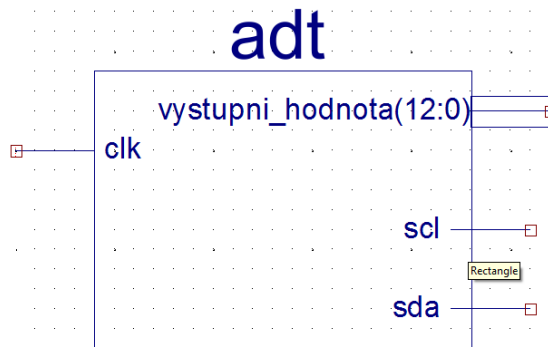


Obrázek 7.3.3 – princip funkce automatu pro řízení odesílání dat na rozhraní UART

Ke změnám dochází vždy při příchodu hodinového pulzu s frekvencí 100 MHz.

## 7.4 Řízení čidla ADT7420

K řízení čidla je využita sběrnice I<sup>2</sup>C, bližší informace o tom jak funguje toto čidlo nebo jak funguje tato sběrnice, lze najít v kapitolách 5.1 respektive 4.1. Na následujícím obrázku je komponenta pro řízení čidla ADT7420. V komponentě adt nedochází k jakékoliv úpravě příchodích dat.

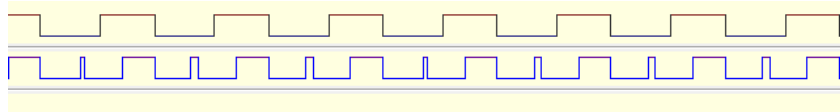


Obrázky 7.4.1 – Komponenta pro řízení čidla ADT7420

Jako v předchozích případech je port clk určen pro příchodí hodinový pulz. Port vystupni\_hodnota je určen pro odeslání přijatých dat od čidla do nadřazené komponenty pro následné zobrazení. Porty scl a sda jsou určeny pro vodiče sběrnice I<sup>2</sup>C (viz kapitola 4.1).

Důležitou součástí této komponenty je generátor taktu, který generuje takt 250 kHz který je určen pro sběrnici I<sup>2</sup>C, a generátor 500 kHz, který je určen pro stavový automat, který řídí komunikaci na sběrnici I<sup>2</sup>C.

Stavový automat tvoří hlavní část komponenty adt. Tento automat je časován pomocí vnitřního hodinového pulzu s frekvencí 500 kHz, který je posunut vůči signálu scl o čtvrt periody signálu na scl tak, že náběžná hrana tohoto vnitřního pulzu se vždy nachází uprostřed logické úrovně signálu na scl (viz následný obrázek, vrchní signál je signál scl a spodní signál je vnitřní signál s frekvencí 500 kHz).



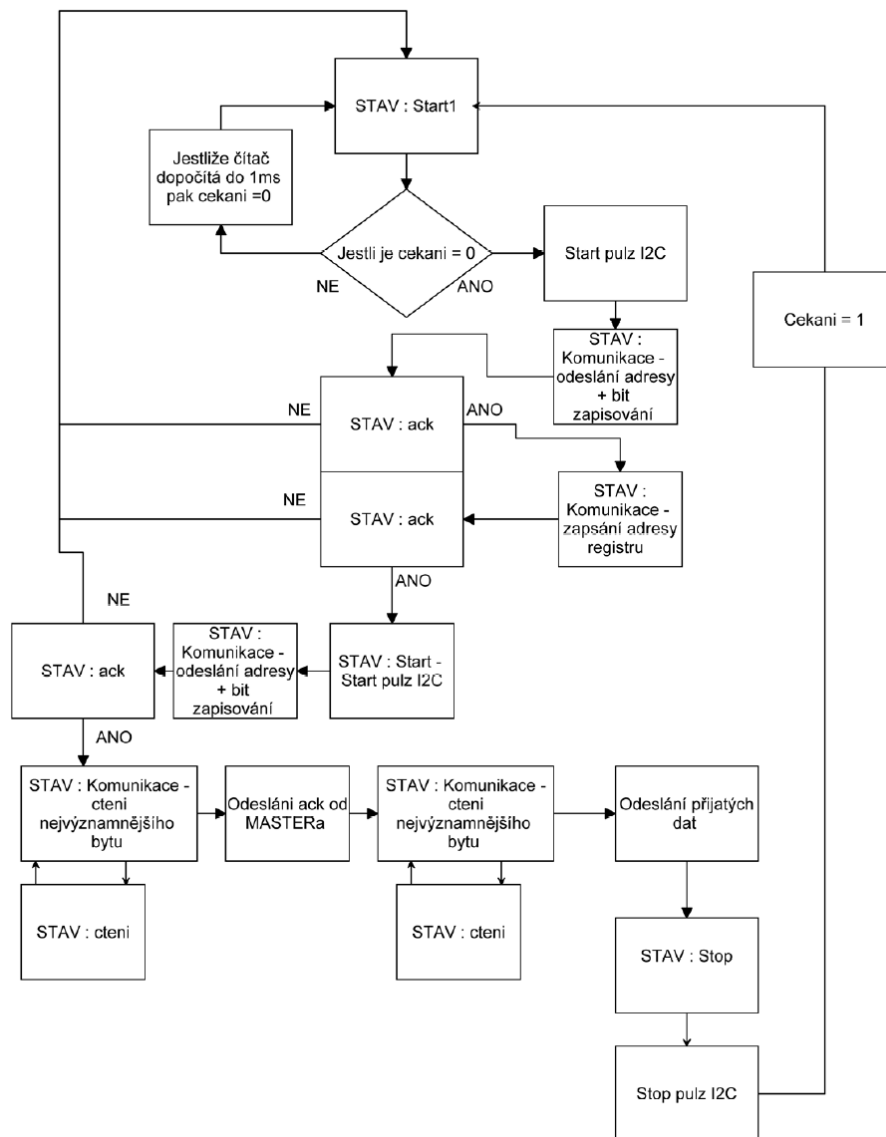
Obrázek 7.4.2 – Ukázka signálu uvnitř komponenty ADT

Co se týče sběrnice I<sup>2</sup>C, komunikace probíhá tak, že nejdřív se vyšle adresa zařízení (1001011) s nastaveným bitem pro psaní a následně proběhne zápis registru, ze kterého budu číst teplotu (0x00). Následně se znovu odešle adresa zařízení a poté se začínají číst registry obsahující hodnoty teploty. První je nejvýznamnější bajt, poté se odvíjí „acknowledgment bit“ od Mastera, načež se začne číst nejméně významný bajt, po kterém následuje „no-acknowledgment bit“. Tato komunikace se po 1 ms znovu opakuje.

Stavový automat, který řídí celou komunikaci, začíná vždy ve stavu „Start1“. V tomto stavu se vyšle startovací impulz pro sběrnici I<sup>2</sup>C (nastává při scl = 1),

následně se přepne do stavu „komunikace“. V tomto stavu, jak již název napovídá, probíhá řízení veškeré komunikace. Nejdříve se odvysílá sekvence s adresou zařízení i adresou registru, načez se přepne do stavu „Start“, ve kterém se vyšle nový startovací impuls sběrnice I<sup>2</sup>C. Následně se přepne opět do stavu „komunikace“, kde dojde k opětovnému odvysílání adresy a poté nastává čtení registrů s hodnotou teploty a to tak, že se vždy přepíná mezi stavy „komunikace“ a „pis“, kde dojde k zapsání do registru, který později odešle tuto hodnotu nadřizované komponentě pro řízení (dochází k tomu proto, že komunikace pracuje při scl=0, ale čtení musí probíhat při scl=1). Po přečtení obou bajtů z vnitřního registru čidla AT7420 nastává stav „stop“, který vyšle stop impuls. Poté dojde k přepnutí do stavu „Start1“, ve kterém se čeká 1 ms a pak se začne s komunikací od začátku.

Stav „Start“ a „Start1“ jsou v podstatě stejné jen s tím rozdílem, že ve stavu „Start1“ se čeká. Kdyby tyto stavy byly spojené, došlo by k potencionálně nežádoucímu prodloužení délky jednoho pulzu signálu scl. Toto by se projevilo chybou v komunikaci a čidlo by odeslalo „no-acknowledgment bit“ v následující komunikaci.



Obrázek 7.4.3 – Zjednodušená funkce automatu, je řízen signálem o 500kHz

## 7.5 Řízení čidla DS18B20

O tom jak funguje čidlo DS18B20 lze dohledat informace v kapitole 5.2 a o komunikaci na sběrnici One-Wire v kapitole 4.2. Komponenta na následujícím obrázku (dsb) je určena pro řízení sběrnice One-Wire s připojeným čidlem DS18B20, je-li na sběrnici více zařízení, program pravděpodobně nebude fungovat, jelikož je využito příkazu Skip-ROM, které adresuje všechna zařízení na sběrnici.



Obrázek 7.5.1 - Komponenta pro řízení čidla DS18B20

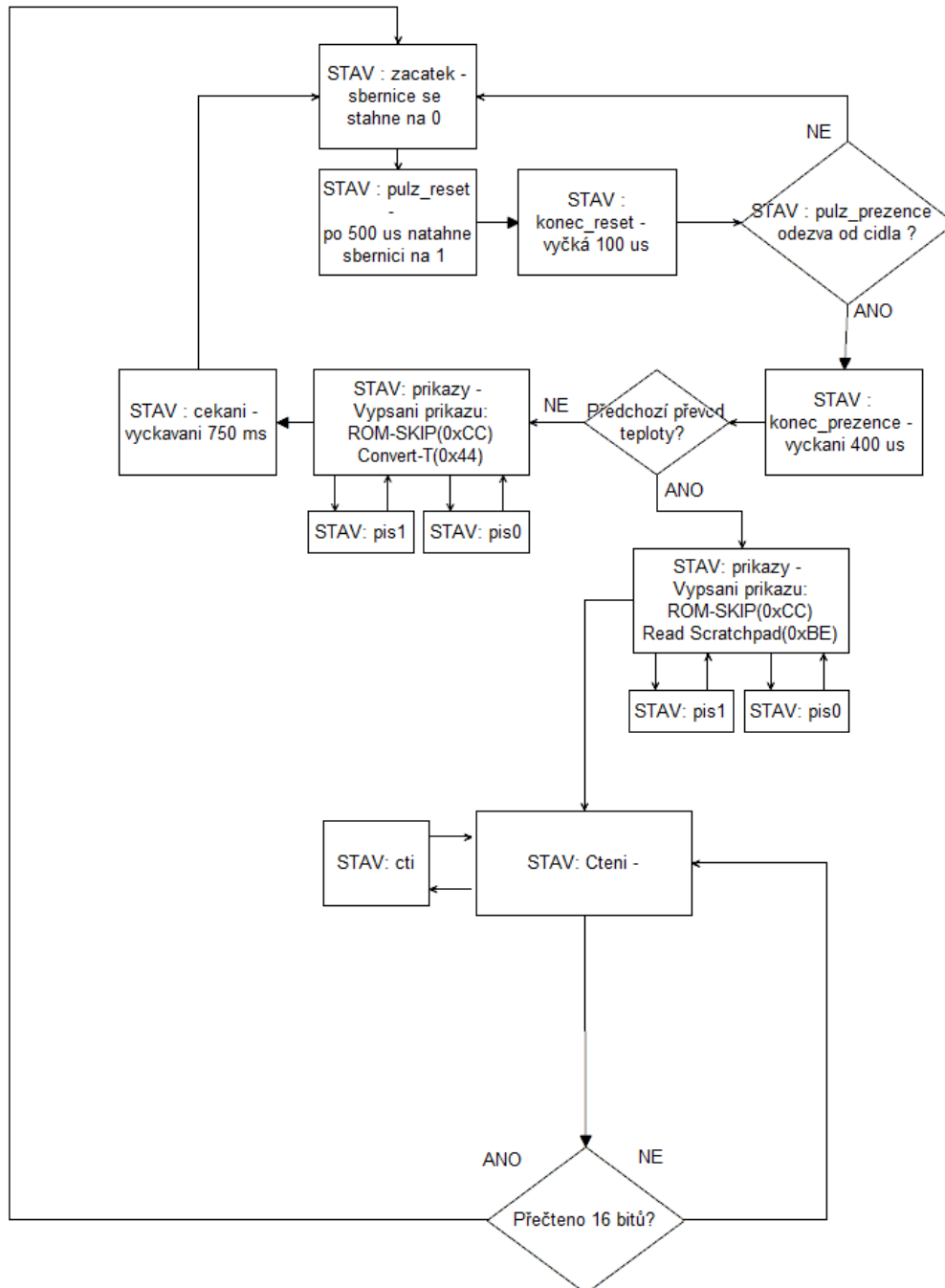
Clk je port sloužící k přívodu hodinového taktu o frekvenci 100 MHz. Port sbernice je určen pro přívod na datový vodič sběrnice One-Wire. Pro předávání dat obdržených od čidla do nařízené řídicí komponenty je určen port výstupní hodnota.

Součástí komponenty je generátor hodinového pulzu s periodou 1  $\mu$ s, jenž je použit pro řízení stavového automatu, který řídí komunikaci na sběrnici One-Wire.

Stavový automat, jenž řídí komunikaci, se nachází na začátku vždy ve stavu „zacatek“. V tomto stavu se stáhne sběrnice na nízkou logickou úroveň a dojde k přepnutí do stavu „pulz\_reset“. V tomto stavu se vyčkává po 500  $\mu$ s, než dojde k nastavení vysoko-impedančního stavu na sběrnici, a následně se stav přepne do „konec\_reset“. V tomto stavu se pouze vyčkává 100  $\mu$ s, než dojde k nastavení dalšího stavu „pulz\_prezence“. Nyní se vyhodnotí, zda je na sběrnici nízká logická úroveň, což by znamenalo, že čidlo odpovědělo prezenčním pulzem na pulz resetovací a komunikace může pokračovat. V opačném případě se stav přepne do stavu „zacatek“. Při pokračování komunikace se nastaví stav „konec\_prezence“, kde se vyčkává po 400  $\mu$ s, čímž se překlene doba potřebná na dokončení inicializační sekvence. Následuje přepnutí do stavu „prikazy“, kde dochází k odeslání příkazů Skip-ROM (11001100, 0xCC) a Convert-T (01000100, 0x44). Příkazy se odesílají přepínáním mezi stavem „prikazy“ a stavy „pis1“ a „pis0“. Příkazy se odesílají od LSB po MSB. Stavy „pis1“ a „pis0“ se liší pouze v tom, že ve stavu „pis1“ se nastaví vysoko-impedanční stav na vodič sběrnice 1  $\mu$ s po přepnutí do tohoto stavu, kdežto u stavu „pis0“ se tento stav nastaví až po 80  $\mu$ s. Po odeslání obou příkazů se nastaví stav „cekani“, během něhož se čeká na dokončení převodu teploty uvnitř čidla, tedy 750 ms. Po dokončení převodu teploty se musí projít inicializační část komunikace od počátku, což znamená, že dojde k přepnutí do stavu začátek. Až automat dojde ke stavu „prikazy“, odvíjejí se příkazy Skip-ROM (11001100, 0xCC) a Read Scratchpad (10111110, 0xBE), opět přepínáním mezi stavem „prikazy“



a stavy „pis1“ a „pis0“. Po odvysílání příkazů do čidla začne čidlo vysílat data (začne vypisovat data z celého „scratchpadu“) a automat se přepne do stavu „cteni“. Stav „cteni“ načte prvních 16 bitů přepínáním mezi stavy „cteni“ a „cti“. Ve stavu „cti“ se vzorkuje vždy 12  $\mu$ s od začátku čtecího slotu. Po přečtení těchto bitů se automat přepne do stavu „zacatek“ a celá komunikace začíná od počátku, čímž se zabrání čidlu vysílat další data.

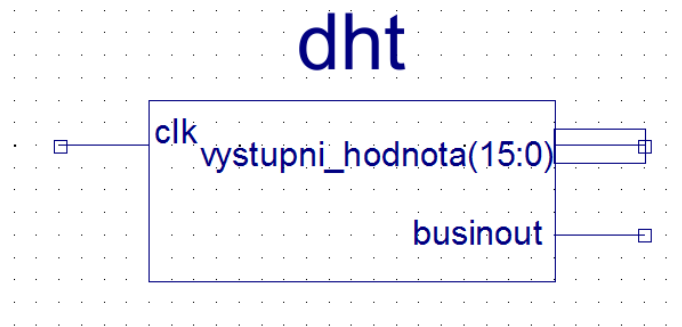


Obrázek 7.5.2 – Zjednodušená funkce automatu pro řízení DS18B20

## 7.6 Řízení čidla DHT11

Bližší informace o čidle DHT11 a o komunikaci s ním lze dohledat v kapitole 5.3. Komponenta pro řízení čidla DHT11 vždy čte data z čidla jednou za 5 s (viz. Kapitola 5.3).

Na následujícím obrázku je vyobrazená komponenta pro řízení čidla DHT11. K úpravám přijatých dat z čidla opět nedochází v této komponentě, ale až v komponentách určených k zobrazení příslušných dat.

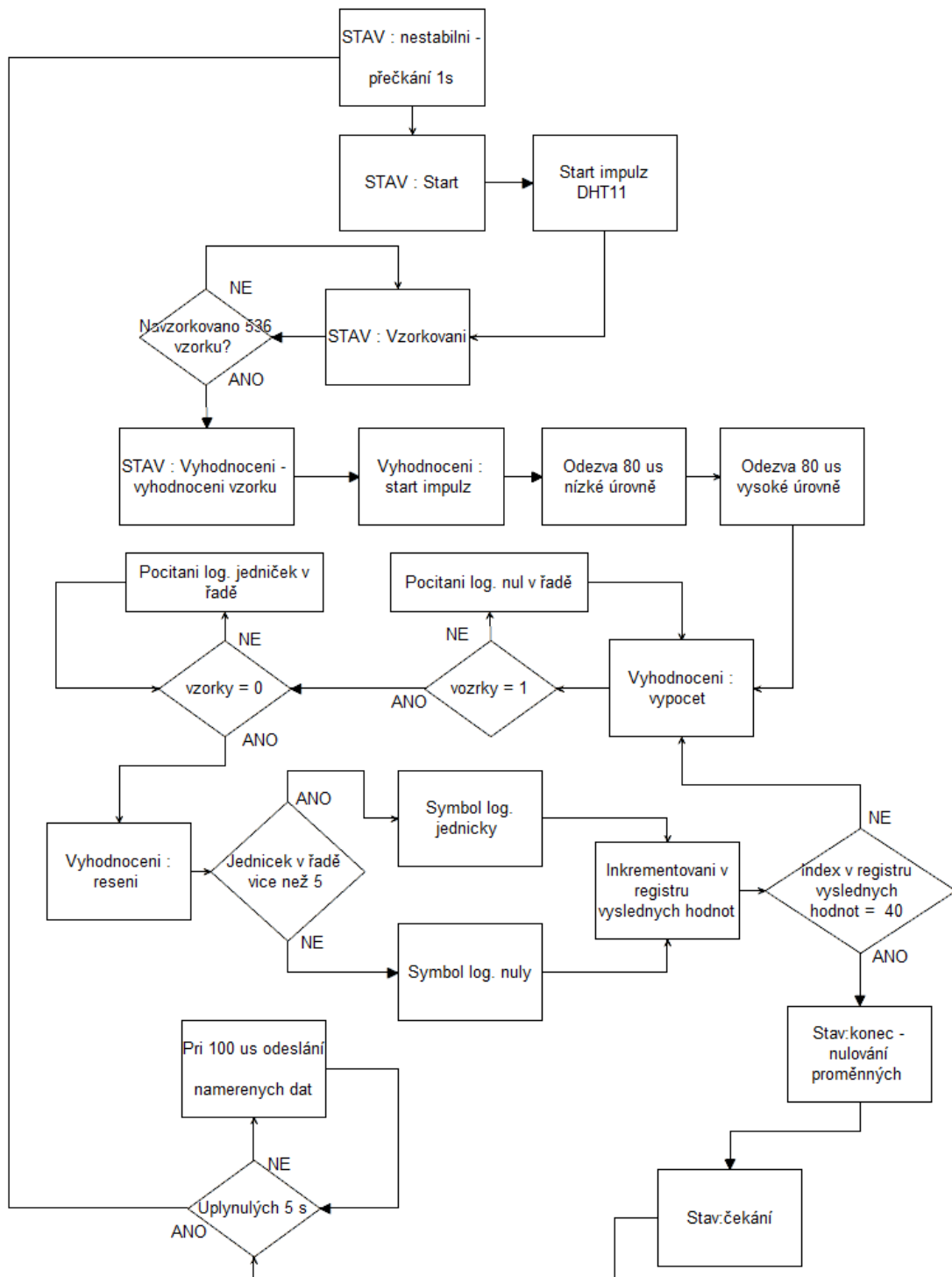


Obrázek 7.6.1 – Komponenta pro řízení čidla DHT11

Port clk slouží opět k přivedení hodinového pulzu pro řízení této komponenty, port businout slouží k připojení na datový vodič čidla. Port vystupni\_hodnota slouží k předání dat obdržených od čidla ke zpracování nadřazené řídicí komponentě.

Pro řízení čidla obsahuje blok dht generátor hodinového pulzu s periodou 1  $\mu$ s, který je generován pro účely stavového automatu této komponenty.

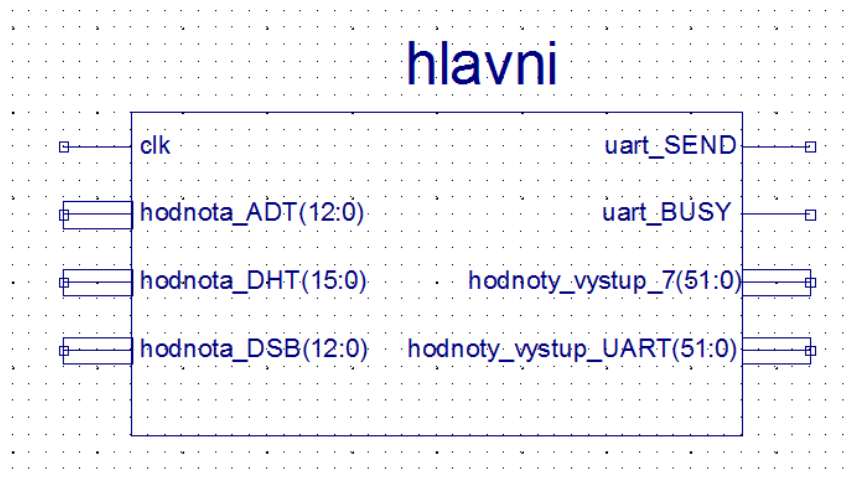
Automat této komponenty začíná vždy ve stavu „nestabilní“, ve kterém přečkává nestabilní stav na sběrnici během spuštění čidla, tento stav je nastaven po 1 s. Po přečkání 1 s následuje stav „start“, který stáhne sběrnici na nízkou napěťovou úroveň po dobu 18,5 ms, poté začíná reakce čidla DHT11, přičemž se automat přepne do stavu „vzorkování“, kdy se vzorkuje příchozí signál od čidla každých 10  $\mu$ s. Po získání 536 vzorků se automat přepne do stavu „vyhodnot“, ve kterém se zjišťují, jaká data čidlo vyslalo. 536 je maximum vzorků, které lze dostat za předpokladu, že čidlo by vyslalo sekvenci jen o logických jedničkách. Vyhodnocení nejprve přeskočí sekvenci, kde se nachází startovací a prezenční impuls od čidla. Poté se prochází navzorkovaná sekvence, vždy se spočítá, kolik je za sebou jdoucích navzorkovaných logických nul a kolik po sobě jdoucích jedniček. Z tohoto se vyhodnotí, zda pulz měl význam logické jedničky či logické nuly. Po obdržení 40 bitů, které získá automat z navzorkovaného signálu, se přepne stav „konec“, kde se vynulují všechny proměnné a ihned se přepne do stavu „čekání“, během něhož se odešlou získaná data nadřazené řídicí komponentě ke zpracování. V tomto stavu automat vyčká po 4 s a následně se přepne do stavu „nestabilní“, kde přečká poslední vteřinu před následným zopakováním komunikace.



Obrázek 7.6.2 – Zjednodušená funkce automatu k ovládání čidla DHT11

## 7.7 Komunikace mezi bloky

Pro komunikaci mezi jednotlivými komponentami je potřebná komponenta hlavní.vhd, která přijímá data od jednotlivých komponent určených k řízení čidel (ADT7420, DS18B20 a DHT11) a přeposílá tyto data ke komponentám určených k řízení výstupů (řízení sedmi-segmentového displeje a řízení rozhraní UART). Na následujícím obrázku je zobrazena komponenta hlavní.vhd. Tato komponenta je řídicí a je nadřazená všem předešlým komponentám.



Obrázek 7.7.1 – Komponenta určená k řízení ostatních komponent

Port clk je určen jako u všech předchozích bloků pro přívod hodinového pulzu s frekvencí 100 MHz. Porty hodnota\_ADT, hodnota\_DHT a hodnota\_DSB jsou určeny pro přivedení dat od příslušných čidel a porty hodnoty\_vystup\_7 a hodnoty\_vystup\_UART jsou určeny k odeslání těchto dat do komponent řídicích výstupy. A konečně porty uart\_SEND a uart\_BUSY jsou určeny k signalizaci odeslání dat respektive k signalizaci připravenosti komponenty.

Tato komponenta nevyužívá žádný automat, obsahuje jen dva čítače. Jeden čítač je určen k čítání do 21,45 s a následné signalizaci komponentě řídicí UART, že má odeslat sekvenci. Druhý čítač odesílá data z čidla každých 8 ms do komponent řídicích rozhraní UART a sedmi-segmentový displej.

## 8. Závěrečné zhodnocení

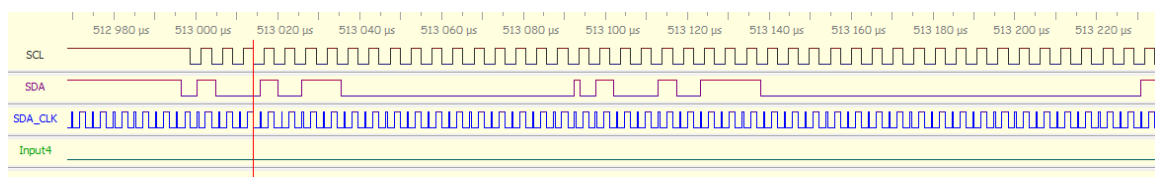
Výsledkem mojí bakalářské práce je program určený pro přípravek Digilent Nexys 4. Tento program řídí komunikaci s jednotlivými čidly (DHT11, ADT7420 a DS18B20) a také řídí odeslání naměřených obdržených dat na výstup a jejich zobrazení. Tento výstup jednak zahrnuje zobrazení na osmimístném sedmi-segmentovém displeji přípravku, a také zobrazení na PC pomocí terminálového klienta.

V rešeršní části práce jsem se zaměřil zejména na popis použitých čidel a komunikace s nimi pomocí jejich příslušných sběrnic, jako jsou One-Wire a I<sup>2</sup>C. Dále popisují stručně desku přípravku Digilent Nexys 4 a jazyk VHDL. Na tuto část navazuje vlastní praktický přínos práce, který spočíval ve vytvoření komplexního VHDL kódu pro obsluhu a komunikaci s jednotlivými čidly a zobrazováním získaných výsledků. Pro tyto účely jsem rovněž vytvořil několik pomocných komponent a signálů, které pomáhají řídit celý program. Druhá část mé bakalářské práce proto obsahuje zejména popis mnou navrženého řešení v jazyce VHDL.

Co se týče komunikace s PC, v plánu bylo použít sběrnici RS-232 ale, jelikož přípravek neobsahuje přímo port pro RS-232, bylo využito komunikace přes FTDI FT232RL USB-UART bridge (jenž je zabudován do přípravku Digilent Nexys 4), který překládá komunikaci z rozhraní UART na komunikaci typu USB. Tuto komunikaci lze na počítači zobrazit přes virtuální COM port. Po zvážení a konzultacích s vedoucím práce jsem se rozhodl nezahrnovat čidlo tlaku do programu, jelikož jsou čidla tlaku v porovnání s ostatními čidly poměrně drahá. Toto čidlo bylo nahrazeno možností komunikace s PC pomocí sběrnice RS-232, o čemž jsem se zmínil v odstavci výše.

Při návrhu a práci ve vývojovém prostředí XILINX ISE jsem narazil na mnoho problémů. Většina z těchto problémů měla co do činění s nekompatibilitou vývojových nástrojů společnosti XILINX s novou verzí MS Windows a to Windows 8 respektive Windows 8.1. Tyto problémy zahrnovaly například nefunkčnost simulačních programů iSim a Modelsim, které nedokázaly propojit návrh z vývojového prostředí XILINX ISE. Nebo občasné a zcela náhodné zastavení v programu v průběhu syntézy či při jiných úkonech prováděných před programováním přípravku.

Po konzultaci s vedoucím práce mi byl zapůjčen logický analyzátor ASIX SIGMA, který mi pomohl při ladění programů a jejich výstupních signálů. Pro nefunkčnost výše zmíněných simulačních programů byl tento analyzátor klíčový pro tvorbu mnou vytvořených programů pro přípravek Digilent Nexys 4. Na následujícím obrázku je příklad zaznamenané komunikace při ladění komponenty, která řídí sběrnici I<sup>2</sup>C s čidlem ADT7420.



Obrázek 8.1 – Průběh analyzovaný pomocí logického analyzátoru ASIX SIGMA

Co se týče práce s čidly, zajímavostí je že vestavěné čidlo ADT7420 po zapnutí přípravku měří přibližně stejnou hodnotu teploty jako přesné externí čidlo DS18B20 a dost podobnou hodnotu teploty jako čidlo DHT11. Ke změně ovšem dojde po dlouhodobějším provozu přípravku Digilent Nexys 4, neboť čidlo ADT7420 ukazuje stále vyšší hodnotu než externí čidlo DS18B20. Největší odchylku teploty od čidla DS18B20, kterou jsem byl schopen zaregistrovat, byla 5 °C. Toto je pravděpodobně způsobeno zahříváním čipu poblíž místa kde se nachází teplotní čidlo ADT7420.

Pro připojení externích čidel na přípravek bylo využito Pmod™ konektorů „JA“, pro čidlo DHT11, a „JB“, pro čidlo DS18B20. Při odpojení čidel z přípravku (čidlo ADT7420 nelze odpojit z přípravku), během doby kde běží program, je na výstupech vidět stále poslední naměřené hodnoty teploty, respektive relativní vlhkosti. Po připojení čidel zpátky k přípravku čidla provádí komunikaci opět od začátku.

Z důvodu časové náročnosti nejsou implementovány všechny funkce, které jsem zprvu zamýšlel. Program by šlo obohatit o inteligentní komponentu, která by po připojení čidla podle daného průběhu na sběrnici zjistila, o kterou sběrnici respektive čidlo se jedná a připojila by toto čidlo ke komponentě, která tuto sběrnici respektive čidlo dokáže ovládat. Dále jsem přemýšlel nad programem, pravděpodobně naprogramovaném v jazyce C nebo Java, který by na připojeném PC snadněji prezentoval data do grafické podoby, případně by vedl statistiku hodnot za posledních několik hodin nebo dní. Dalším zlepšením by mohlo být využití poplachových stavů, které generují čidla DS18B20 a ADT7420 pro signalizaci přesáhnutí nastavených hodnot pomocí audio výstupu. Nebo signalizací přes vícebarevné LED, které jsou vestavěné do přípravku Digilent Nexys 4. V průběhu řešení práce jsem rovněž zvažoval možnost využít Ethernetové rozhraní pro komunikaci a zobrazení naměřených výsledků na připojeném PC. Nakonec jsem však tuto myšlenku zavrhl z důvodu velké složitosti implementace rozhraní Ethernet. Dalším možným námětem na rozšíření mého programu by bylo zobrazování výsledků přes rozhraní VGA, jež je zabudováno do přípravku Digilent Nexys 4, a realizace výstupu na připojeném LCD displeji.

## Zdroje

- [1] Nexys™4 Artix-7 FPGA Board: Part # 410-274-KIT. DIGILENT INC. Digilent.com [online]. [cit. 2014-01-9]. Dostupné z: <http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,400,1184&Prod=NEXYS4>
- [2] Nexys™3 Spartan-6 FPGA Board: Part # 410-274-KIT. DIGILENT INC. Digilent.com [online]. [cit. 2014-01-9]. Dostupné z: <http://www.digilentinc.com/Products/Detail.cfm?NavPath=2,400,897&Prod=NEXYS3>
- [3] OLEJÁR, Martin. Stručný popis sběrnice I2C a její praktické využití k připojení externí eeprom 24LC256 k mikrokontroléru PIC16F877. In: [online]. Redakce HW serveru, 2000 [cit. 2014-01-11]. Dostupné z: <http://www.hw.cz/navrh-obvodu/strucny-popis-sbernice-i2c-a-jeji-prakticke-vyuziti-k-pripojzeni-externi-eeeprom-24lc256>
- [4] MALÝ, Martin. Sběrnice 1-Wire™. In: [online]. Redakce HW serveru, 2004 [cit. 2014-01-11]. Dostupné z: <http://www.hw.cz/navrh-obvodu/strucny-popis-sbernice-i2c-a-jeji-prakticke-vyuziti-k-pripojzeni-externi-eeeprom-24lc256>
- [5] ADT7420: Datasheet. In: [online]. [cit. 2014-01-14]. Dostupné z: [http://www.analog.com/static/imported-files/data\\_sheets/ADT7420.pdf](http://www.analog.com/static/imported-files/data_sheets/ADT7420.pdf)
- [6] DS18B20: Datasheet, One-Wire™ Digital Thermometer. In: [online]. [cit. 2014-01-14]. Dostupné z: <http://dlnmh9ip6v2uc.cloudfront.net/datasheets/Sensors/Temp/DS18B20.pdf>
- [7] Temperature and humidity module: DHT11 Product Manual. In: [online]. [cit. 2014-01-17]. Dostupné z: <http://akizukidenshi.com/download/ds/aosong/DHT11.pdf>
- [8] KRÁL, Jiří. Řešené příklady ve VSHDL: Hradlové pole FPGA pro začátečníky. 1. vyd. Praha: BEN, 2010. ISBN 978-80-7300-257-2.
- [9] SLINTÁK, Vlastimil. Arduino a sériová komunikace. In: [online]. [cit. 2014-05-05]. Dostupné z: <http://uart.cz/139/arduino-a-seriova-komunikac>
- [10] PINKER, Jiří a Martin POUPA. *Číslíkové systémy a jazyk VHDL*. 1. vyd. Praha: BEN - technická literatura, 2006, 349 s ISBN 80-730-0198-5.
- [11] SIGMA2 - logický analyzátor. ASIX. *ASIX SIGMA2* [online]. 2013 [cit. 2014-03-20]. Dostupné z: [http://www.asix.cz/dbg\\_sigma.htm](http://www.asix.cz/dbg_sigma.htm)

## Struktura uložených souborů na přiloženém médiu

- Volně vložená je PDF verze této bakalářské práce (**pehnelt\_tomas.pdf**)
- Dále se zde nachází složka **pehnelt\_tomas\_pouzite\_katalogove\_listy**, kde jsou uloženy katalogové listy pro čidla ADT7420, DS18B20 a DHT11. Dále se zde nacházejí katalogové listy přípravků Digilent Nexys 4 a Digilent Nexys 3.
- Na médiu je ve složce **pehnelt\_tomas\_nexys4\_program** též uložen program k ovládání přípravku (tedy celý projekt ve vývojovém prostředí XILINX ISE).