

Na tomto místě bude oficiální  
zadání vaší práce



České vysoké učení technické v Praze  
Fakulta elektrotechnická  
Katedra počítačů



Bakalářská práce

## **Vývojová deska XILINX**

*Jakub Halák*

Vedoucí práce: Ing. Pavel Kubalík, Ph.D

Studijní program: Elektrotechnika a informatika, dobíhající, Bakalářský

Obor: Výpočetní technika

3. ledna 2011



## Poděkování

Rád bych zde poděkoval vedoucímu práce Ing. Pavlu Kubalíkovi PhD. za jeho ochotu okamžitě se mnou řešit problémy, které nastaly během tvorby této práce, jeho bleskovou odezvu při komunikaci a celkovou vstřícnost.

Zvláštní poděkování patří mým rodičům za jejich podporu po celou dobu mého studia.



## Prohlášení

Prohlašuji, že jsem práci vypracoval samostatně a použil jsem pouze podklady uvedené v příloženém seznamu. Nemám závažný důvod proti užití tohoto školního díla ve smyslu §60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonu (autorský zákon).

V Kněžmostě dne 3. 1. 2011

.....





# Abstract

The thesis deals design and realization of development board based on XILINX solution. Board is meant to be used as design tester and for studies. It contains all basic periferies that are required to create many kinds of design.

# Abstrakt

Práce se zabývá návrhem a realizací vývojové desky založeném na řešení firmy XILINX. Deska je určena pro ladění návrhu aplikací a ke studiu. Kit obsahuje všechny základní periferie, se kterými je možné zpracovat velkou řadu potenciálních návrhů.



# Obsah

<b>1</b>	<b>Úvod</b>	<b>1</b>
<b>2</b>	<b>Popis problému, specifikace cíle</b>	<b>3</b>
<b>3</b>	<b>Analýza a návrh řešení</b>	<b>5</b>
3.1	Průzkum trhu . . . . .	6
3.2	Návrh desky . . . . .	7
3.3	Návrh demoaplikace . . . . .	8
3.3.1	7-segmentový displej . . . . .	8
3.3.2	LCD . . . . .	8
3.3.3	VGA . . . . .	8
3.4	Standard VGA . . . . .	9
3.5	Náklady . . . . .	10
<b>4</b>	<b>Realizace</b>	<b>13</b>
4.1	Napájecí zdroj . . . . .	13
4.2	7-segmentový displej . . . . .	14
4.3	LED, tlačítka a přepínače . . . . .	15
4.4	VGA . . . . .	15
4.5	LCD . . . . .	16
4.6	I/O brána . . . . .	17
4.7	Rozvržení pinů CPLD . . . . .	18
<b>5</b>	<b>Demoaplikace</b>	<b>19</b>
5.1	7-segmentový displej . . . . .	19
5.2	LCD . . . . .	21
5.3	VGA . . . . .	24
5.4	I/O brána . . . . .	27
<b>6</b>	<b>Závěr</b>	<b>29</b>
<b>7</b>	<b>Seznam použitých zkratk</b>	<b>33</b>
<b>A</b>	<b>Schéma desky</b>	<b>35</b>
<b>B</b>	<b>DPS a rozmístění součástek</b>	<b>37</b>

<b>C Osazená deska</b>	<b>43</b>
<b>D Obsah přiloženého CD</b>	<b>45</b>

# Seznam obrázků

3.1	Ukázky kitů firem ALTERA vlevo, XILINX vpravo . . . . .	5
3.2	Blokové schéma desky . . . . .	7
3.3	Blokové schéma VHDL návrhu aplikace pro obsluhu 7-seg. displeje . . . . .	8
3.4	Blokové schéma VHDL návrhu aplikace pro obsluhu LCD displeje . . . . .	8
3.5	Blokové schéma VHDL návrhu aplikace pro obsluhu VGA . . . . .	9
3.6	Popis funkce VGA . . . . .	9
3.7	Graf časování VGA signálu . . . . .	10
4.1	Zapojení zdroje . . . . .	13
4.2	Zapojení 7-seg. displeje . . . . .	14
4.3	Zapojení VGA portu . . . . .	15
4.4	Zapojení LCD panelu . . . . .	16
4.5	Zapojení I/O brány . . . . .	17
4.6	Rozvržení vývodů I/O brány, pohled zředu. . . . .	18
5.1	Inicializace LCD displeje. . . . .	21
B.1	Strana Top desky . . . . .	38
B.2	Strana Bottom desky . . . . .	39
B.3	Rozmístění součástek desky Top . . . . .	40
B.4	Rozmístění součástek desky Bottom . . . . .	41



# Seznam tabulek

3.1	Srovnání desek jednotlivých výrobců . . . . .	6
3.2	Rozvržení periferii . . . . .	7
3.3	VGA časování signálu . . . . .	10
3.4	Cena desky za 1 ks . . . . .	11
4.1	Konfigurace pinů LCD . . . . .	16
4.2	Funkce DIR pinu . . . . .	17
4.3	Rozvržení pinů CPLD . . . . .	18





# Kapitola 1

## Úvod

Úkolem práce je navrhnout a zrealizovat vývojový kit založený na řešení od firmy XILINX. Tento kit je možné použít pro výuku návrhu v jazyce VHDL, jednoduché zobrazování obrazu přes VGA port, řídicí jednotku pro mnoho druhů systémů a další... Desku je možné rozšířit o další periferie pomocí 8-bitové I/O brány. Základní periferie, které deska obsahuje jsou:

- Tlačítka
- Přepínače
- LED diody
- 7-segmentový displej
- LCD displej
- VGA port
- I/O bránu

Programování desky by mělo probíhat přes standardní rozhraní JTAG. V této práci budou popsány funkce jednotlivých periférií.

Pro otestování funkcí desky byli napsány dílčí jednoduché návrhy, které umožňují otestovat jednotlivé periferie.



## Kapitola 2

# Popis problému, specifikace cíle

Hlavním úkolem je navrhnout programovatelnou desku se standardními periferiemi.

- Navrhnout a vyrobit vývojovou desku.
- Navrhnout základní periferie.
- Naprogramovat demoaplikaci.

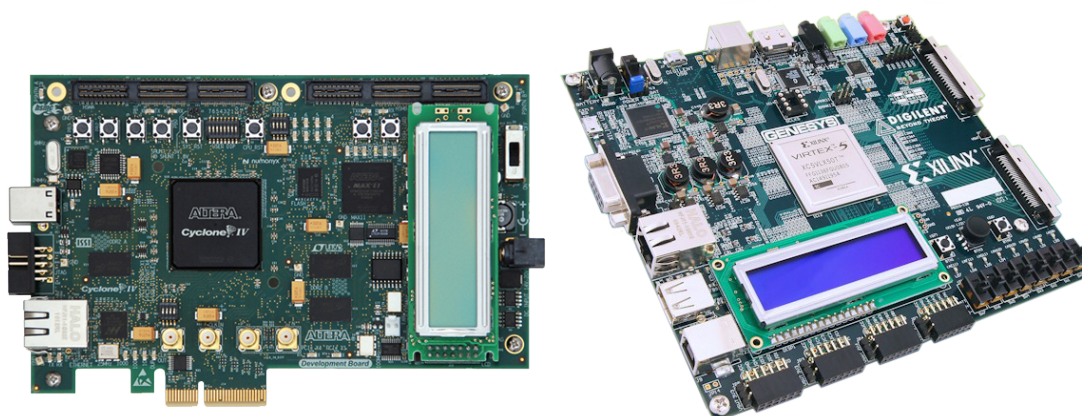


## Kapitola 3

# Analýza a návrh řešení

S vývojovými kity je možné se setkat ve vývojových centrech, nebo ve školách. Jedná se o víceúčelné systémy, pomocí kterých je možné provádět ukázky návrhů finálních produktů. Poměrně známou firmou zabývající se vývojem různých aplikací je firma DIGILENT. Kit většinou obsahuje základní periferie jako jsou tlačítka, nebo přepínače pro simulaci vstupu digitálního signálu (např. detekce objektu laserem) a periferie pro zobrazování výstupu od jednoduchých LED diod po LCD panely. Výjimečně je možné na kitech nalézt i VGA port pro komplexnější zobrazení. Pomocí těchto základních periférií je možné simulovat například řízení výrobní linky, ovládání informačních tabulí (letišť, metro...), a mnohé další.

Mezi nejznámější řešení patří produkty od firem XILINX a ALTERA (obrázek 3.1). Obě firmy poskytují obsáhlé dokumentace ke svým řešením s podrobnými popisy a užitečnými radami pro návrh jednotlivých kitů.



Obrázek 3.1: Ukázky kitů firem ALTERA vlevo, XILINX vpravo

Deska, která je cílem této práce, je založená na řešení od firmy XILINX. Obsahuje veškeré potřebné periferie, navíc bude obsahovat VGA port a 8-bitovou I/O bránu pro komunikaci s dalšími potenciálními zařízeními. Zvolil jsem integrovaný obvod XC9572XL, který spadá

do kategorie výkonných CPLD pracujících na frekvenci až 178Mhz a zároveň do takzvaných low power systémů. Dle odhadu, založeném na informacích od firmy XILINX, by odběr neměl při nulovém vytížení přesáhnout hodnotu 40 mA a při maximálním vytížení 70 mA. Tyto hodnoty se však týkají samotného čipu, spotřeba periférii navýší celkovou hodnotu až k 500 mA. Při vytváření návrhu je potřeba brát zřetel na fakt, že čip obsahuje pouze 72 logických buněk tudíž je potřeba neplýtvat poskytnutou pamětí. Původně bylo zamýšleno do návrhu zahrnout obvod XC95144XL, který se od zvoleného liší pouze dvojnásobným počtem logických buněk a malým rozdílem v několika pinech, takže budoucí výměna není možná bez změny v návrhu. Bohužel je tento obvod třikrát dražší, a proto bylo rozhodnuto použít méně výkonný typ. Aplikace pro navrhovanou desku se budou programovat pomocí jazyka VHDL.

### 3.1 Průzkum trhu

Při průzkumu trhu jsem objevil mnoho podobných výrobků. Většina těchto desek byla však konstruována jako samostatný modul bez periférii s velkou možností rozšíření. Mezi mnoha výrobky jsem našel dva, které se navrhovanému obvodu podobají. Jeden produkt vyrábí firma DIGILENT a druhý firma ALTERA.

Výrobek od firmy DIGILENT je tomu našemu nejvíce podobný. Veškeré periférie, které jsou zde použity, jsou použity i u našeho návrhu. Jediná periférie, která zde chybí je LCD displej a I/O brána. Je však navíc obohacen o sériovou linku a PS2 port, takže není problém jej o LCD panel doplnit. Dalším rozdílem je integrovaný obvod Spartan III, který se řadí do kategorie FPGA. Celková cena a možnosti jsou tedy větší.

Firma ALTERA používá rozdílný druh obvodu, základní myšlenka je však stejná. Deska je založena na obvodu Stratix III, který firma vyrábí. Kit je osazen všemi perifériemi, ale na místo I/O brány má grafický displej.

Ceny jednotlivých desek spolu s parametry jsou uvedeny v tabulce 3.1.

Periférie	DIGILENT Spartan III	ALTERA Stratix III
Tlačítka	✓ 4x	✓ 4x
Přepínače	✓ 8x	✓ 8x
LED	✓ 8x	✓ 8x
7-seg	✓	✓
LCD	X	✓ 16x2
VGA	✓	X
I/O brána	✓ rozšiřující konektor	X
ostatní	PS2, Serial port, SRAM	128 x 64 grafický displej
Cena	109 \$	2495 \$

Tabulka 3.1: Srovnání desek jednotlivých výrobců

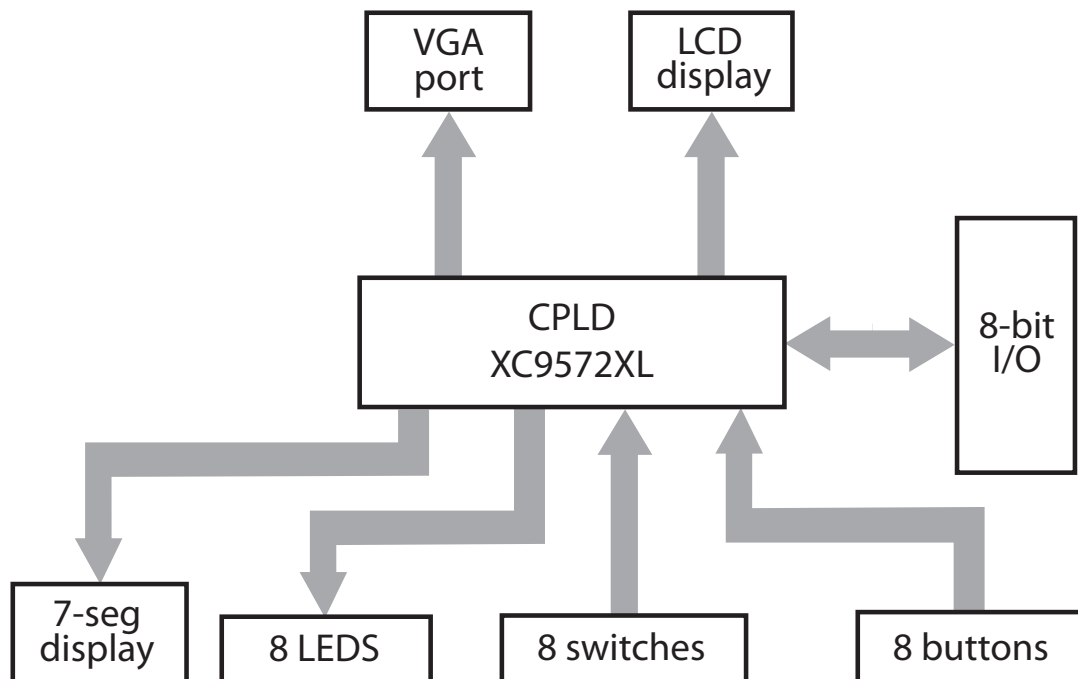
## 3.2 Návrh desky

Na desce se nachází všechny základní periferie, jejich rozvržení je uvedeno v tabulce 3.2.

Periferie	Počet pinů
Tlačítka	8
Přepínače	8
LED diody	8
7-seg	12
LCD	11
VGA	10
I/O brána	8
Hodiny	1
Reset	1

Tabulka 3.2: Rozvržení periferii

Deska je osazena dvěma zdroji napětí, protože obsahuje i periferie, které pracují s různým napětím, nežli je napětí potřebné pro napájení CPLD. I/O brána vyžaduje jako jediná napájení z obou zdrojů, kvůli převodu mezi úrovněmi signálu.



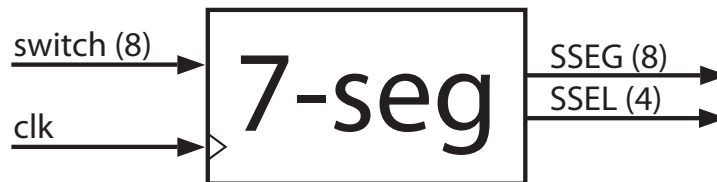
Obrázek 3.2: Blokové schéma desky

### 3.3 Návrh demoaplikace

Jelikož je paměť obvodu velice omezená, byly jednotlivé návrhy rozděleny do samostatných projektů. V této části práce nastíníme, jak by měly návrhy vypadat, abychom si následně usnadnili práci. Dle odhadu by složitost jednotlivých částí neměla být vysoká, jelikož se jedná jen o ukázky a ne o plnohodnotné aplikace s větším účelem.

#### 3.3.1 7-segmentový displej

Vstupem displeje jsou hodiny, které použijeme k přepínání jednotlivých digitů, protože je displej multiplexovaný. Přepínače nám pouze poslouží jako vstup, který se na displeji zobrazí v podobě hexadecimální hodnoty.



Obrázek 3.3: Blokové schéma VHDL návrhu aplikace pro obsluhu 7-seg. displeje

#### 3.3.2 LCD

U LCD provedeme pouze inicializaci s následným vypsáním slova READY ve chvíli, kdy je displej připraven. Uvnitř aplikace bude třeba použít děličku, která nám zmenší takt hodin, jelikož aktuální hodnota hodin 25MHz by vyžadovala nemalou investici logických buněk k tomu, abychom je mohli využít.



Obrázek 3.4: Blokové schéma VHDL návrhu aplikace pro obsluhu LCD displeje

#### 3.3.3 VGA

VGA bude, co se logických jednotek týče, nejspíše nejnáročnější. Uvnitř aplikace musí neustále čítat dva poměrně rozsáhlé čítače, které nám udávají na kterém pixelu se právě nacházíme. Podrobnější informace o funkci VGA jsou uvedeny v sekci **Standard VGA**.

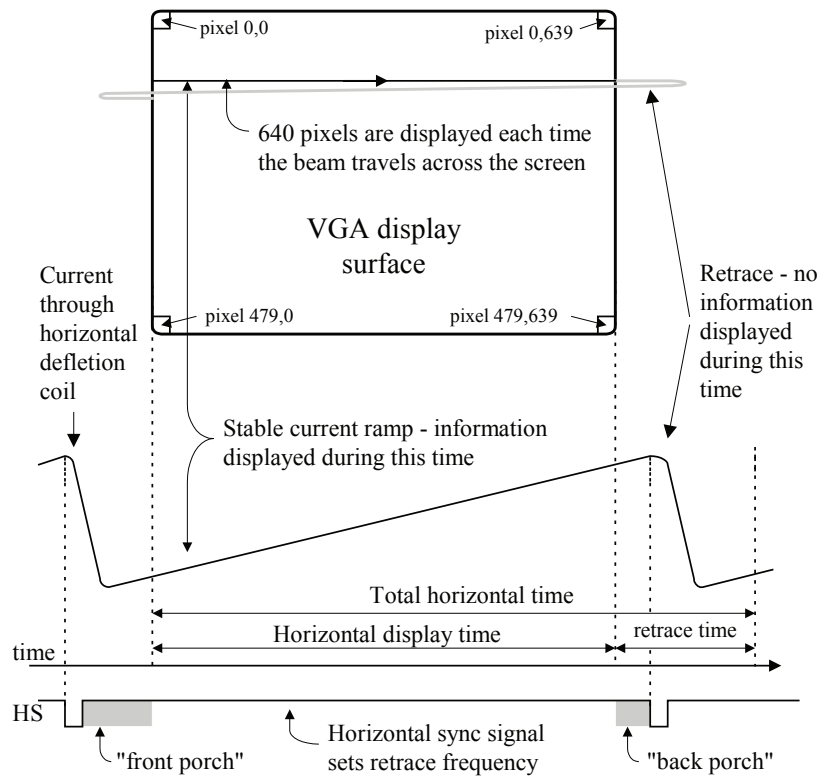




Obrázek 3.5: Blokové schéma VHDL návrhu aplikace pro obsluhu VGA

### 3.4 Standard VGA

Video Graphics Array (VGA) je počítačový standard pro počítačovou zobrazovací techniku, vydaný roku 1987 společností IBM. VGA patří do rodiny starších IBM video standardů. Konektor se dělí na signály RED (R), GREEN (G), BLUE (B), Horizontal Sync (HS) a Vertical Sync (VS). Síť rezistorů poskytuje možnost použití 8-bitové barevné škály, neboli 256 barev, kde máme 3 bity pro zelenou, 3 bity pro modrou a pouze 2 bity pro červenou barvu. Lidské oko není totiž tak citlivé na červenou barvu, proto jsou použity pouze 2 bity místo 3 bitů.

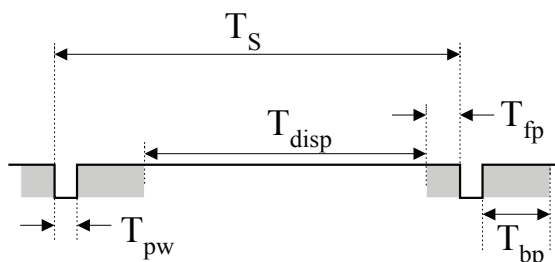


Obrázek 3.6: Popis funkce VGA

Obvod pro řízení VGA musí generovat HS a VS časovací signály a koordinovat výstup video dat založených na pixel clock. Pixel clock definuje čas potřebný pro zobrazení jednoho pixelu. VS signál určuje obnovovací frekvenci displeje, nebo frekvenci, která určuje, kdy se vykreslí všechny informace na obrazovku. Pro správnou funkci displeje je potřeba minimální obnovovací frekvence 60 Hz až 120 Hz. Pro 640-pixelů na 480-pixelů je zapotřebí 25Mhz pixel clock a 60 +/- 1 Hz obnovovací frekvence.

Symbol	Parametr	Vertical Sync			Horizontal Sync	
		Time	Clocks	Lines	Time	Clocks
$T_s$	Sync pulse time	16.7 ms	416,800	521	32 $\mu$ s	800
$T_{disp}$	Display time	15.36 ms	384,000	480	25,6 $\mu$ s	640
$T_{pw}$	VS pulse width	64 $\mu$ s	1,600	2	3,84 $\mu$ s	96
$T_{fp}$	VS front porch	320 $\mu$ s	8,000	10	640 $\mu$ s	16
$T_{bp}$	VS back porch	928 $\mu$ s	23,200	29	1,92 $\mu$ s	48

Tabulka 3.3: VGA časování signálu



Obrázek 3.7: Graf časování VGA signálu

### 3.5 Náklady

Náklady na desku jsou stanoveny hrubou kalkulací. Cena DPS je stanovena na velikost desky 1 dm<sup>2</sup>, ale i přes větší rozměr desky (1,2 dm<sup>2</sup>) se cena změní jen o jednotky korun. V případě sériové výroby by se náklady rozložily mezi jednotlivé kusy a výsledná deska by byla o poznání levnější, viz tab. 3.4.

Materiál	Cena [Kč]	
	1 ks	1000 ks
DPS 1 dm <sup>2</sup>	840	58,864
XC9572XL	92,09	63,66
Konektory	28	6,92
Stabilizatory	42,64	28,43
Pasivní součástky	99,228	53,358
Převodník úrovní	21,26	12,3
Periferie	302,882	205,724
<b>CELKEM</b>	<b>1711,32</b>	<b>515,1072</b>

Tabulka 3.4: Cena desky za 1 ks

Z tabulky 3.4 plyne, že cena desky při výrobě pouze jednoho kusu by mohla některé uživatele odradit, jelikož je potřeba vzít v potaz i náklady na vývoj, čímž se konečná cena ještě navýší. Pokud bychom ale desku vyráběli sériově, je tu velká šance pro úspěch mezi běžnými uživateli, kterým nejde o převratné výkony, ale rádi by se s podobnými zařízeními naučili pracovat a nechtějí utrácet velké peníze.



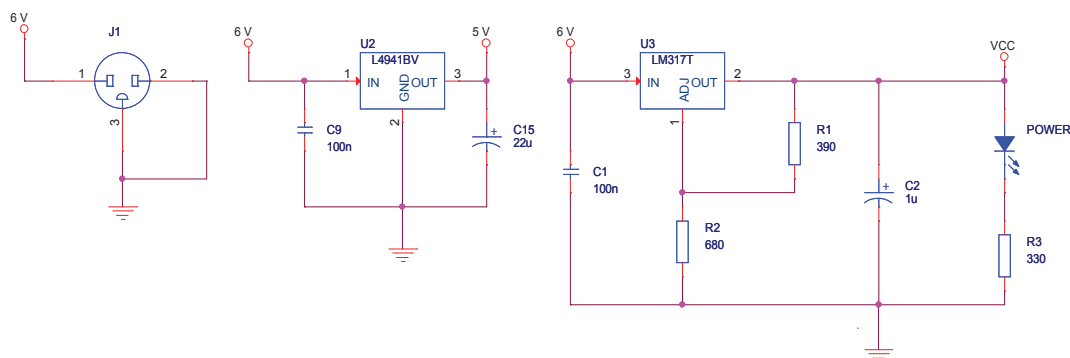
# Kapitola 4

## Realizace

V této části práce je popsáno zapojení jednotlivých periférií, a dále jsou zde popisy portu.

### 4.1 Napájecí zdroj

Deska je napájena dvěma zdroji, zapojení na obrázku 4.1.



Obrázek 4.1: Zapojení zdroje

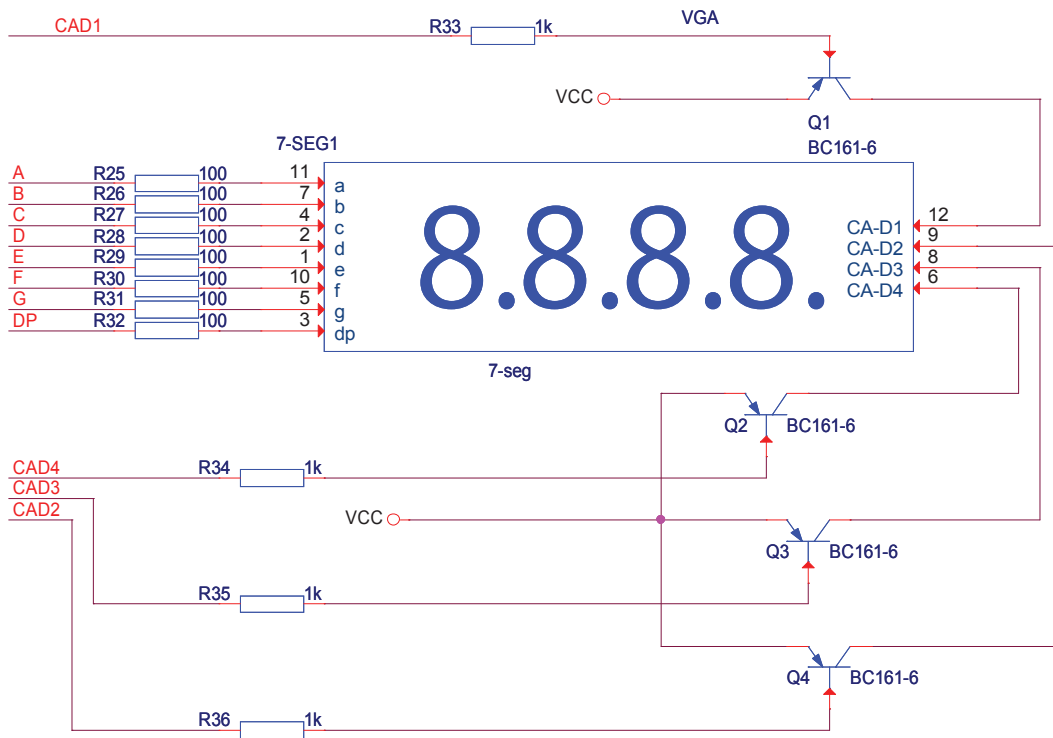
V zapojení je použit low-drop stabilizátor L4941BV a tří-svorkový regulovatelný stabilizátor LM317T, které jsou zapojeny podle doporučení výrobce. Stabilizátor L4941BV byl zvolen kvůli nízkému rozdílu napětí mezi vstupem a výstupem. V případě, že bychom chtěli použít zdroj s větším napětím je možné stabilizátory ponechat, nebo vyměnit L4941BV za standardní 7805.

LED dioda označená "POWER" indikuje, zda je deska pod napětím, pro funkčnost desky však není nezbytná. Tantalové kondenzátory C15 a C2 zlepšují přechodovou odezvu stabilizátorů, nejsou však nezbytné.

Jelikož má deska poměrně značný odběr, je nutné stabilizátory chladit malým chladičem.

## 4.2 7-segmentový displej

Zapojení 7-segmentového displeje je na obrázku 4.2. Jednotlivé segmenty jsou zapojeny na I/O piny CPLD přes 100  $\Omega$  odpory jako ochrana vnitřních segmentů, tato hodnota vychází z elektrických parametrů displeje. Ze stejného důvodu jsou tranzistory PNP připojeny na 1k $\Omega$  odpor.



Obrázek 4.2: Zapojení 7-seg. displeje

Každý tranzistor spíná jeden digit. Sepnutí tranzistoru je provedeno přivedením logické "0" na jeho bázi, tím se rozsvítí daný digit. Přivedením logické "0", na některý segment v daném aktivním digitu, se segment rozsvítí. K tomu, aby každý ze čtyř digitů byl neustále rozsvícen, je zapotřebí spínat každý digit v čase od 1 do 16ms (tj. s obnovovací frekvencí 60Hz až 1KHz), což je mnohem menší frekvence nežli je frekvence základních hodin desky. Tento problém je řešen v demoaplikaci pomocí binární děličky.

### 4.3 LED, tlačítka a přepínače

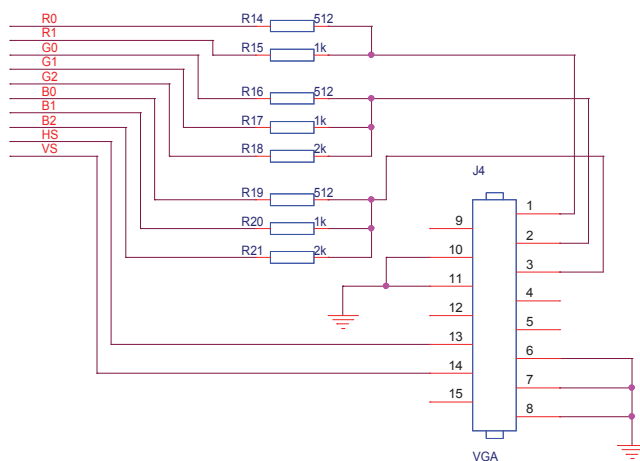
LED diody jsou použity pro detekci vstupu a výstupu. Anoda LED diody je připojena do CPLD přes ochranný rezistor  $330\Omega$ , kvůli omezení proudu, zatímco katoda je připojena na GND. Je-li na výstupu obvodu CPLD stav logická "1", dojde k signalizaci tohoto stavu rozsvícením LED diody.

Tlačítka slouží pro simulaci vstupu. Není-li tlačítko stisknuto, nachází se jeho hodnota ve stavu logické "0", při stisku dojde ke změně stavu na logickou "1". Při stisku tlačítka dochází většinou k nežádoucím záskmitům, které jsou způsobeny mechanickými vlastnostmi tlačítka. Rezistor zajišťuje filtrování odskočení tlačítka. Logická "1" je vystavována po celou dobu stisknutí tlačítka.

Přepínače slouží k přepínání vstupu do CPLD mezi logickou "1" a logickou "0". Na desce jsou spínače zapojeny tak, aby vstup do CPLD "odřízly" od napětí tím, že napětí přes odpor uzemní. Tato metoda není nejlepší, lepší by bylo přepínačem měnit vstup do CPLD mezi Vcc a GND.

### 4.4 VGA

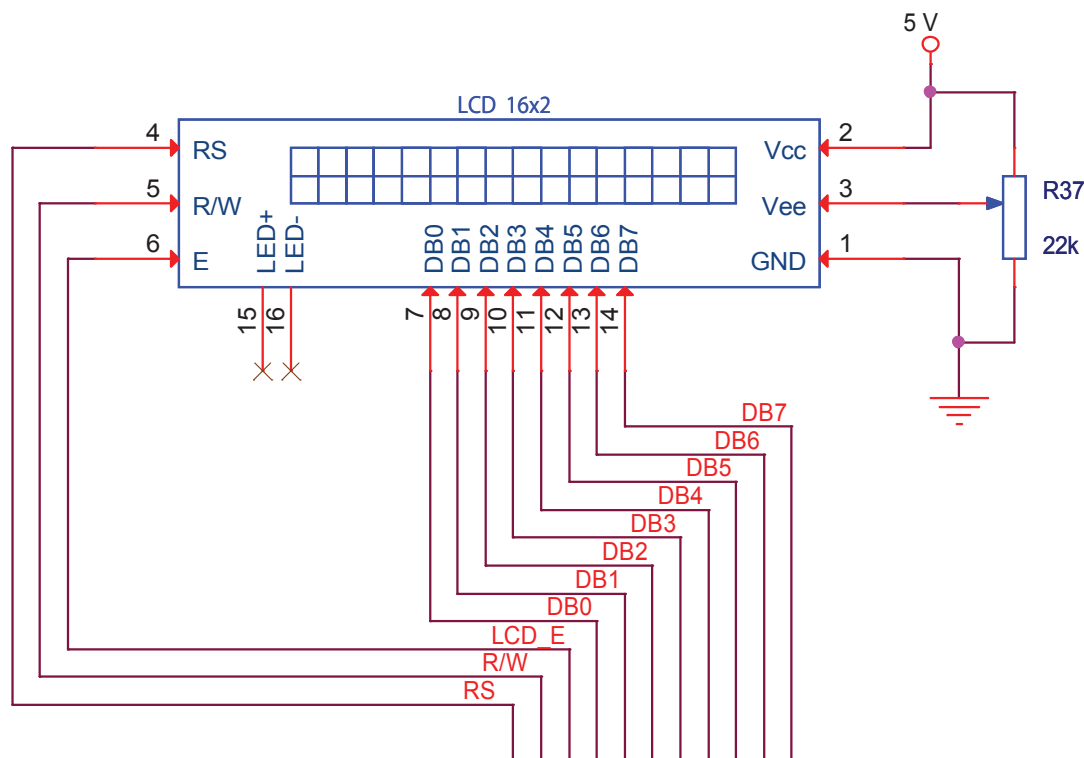
Zapojení VGA portu je v souladu s popisem VGA standardu uvedeného výše. Rezistory, které jsou na jednotlivé barvy připojeny, vytváří úroveň napětí, které jsou na monitoru reprezentovány různými odstíny tří základních barev RGB. Deska je proto schopná zobrazit 4 odstíny červené, 8 odstínů zelené a 8 odstínů modré. Různými kombinacemi je deska schopná zobrazit až  $2^8$  barev, tedy 256 barev.



Obrázek 4.3: Zapojení VGA portu

## 4.5 LCD

Zvolený panel je schopný zobrazit 16x2 znaků. Panel obsahuje 16 pinů, rozvržení v tabulce 4.1. Datové vodiče jsou přivedeny přímo do CPLD stejně jako řídicí signály LCD\_E, R/W a RS. Trimr R37 umožňuje nastavení jasu podsvícení displeje.



Obrázek 4.4: Zapojení LCD panelu

Před samotným použitím LCD je potřeba provést inicializaci, která se skládá z jednotlivých po sobě jdoucích instrukcí s daným časovým rozestupem. Podrobněji bude tato záležitost rozepsaná v popisu demoaplikace.

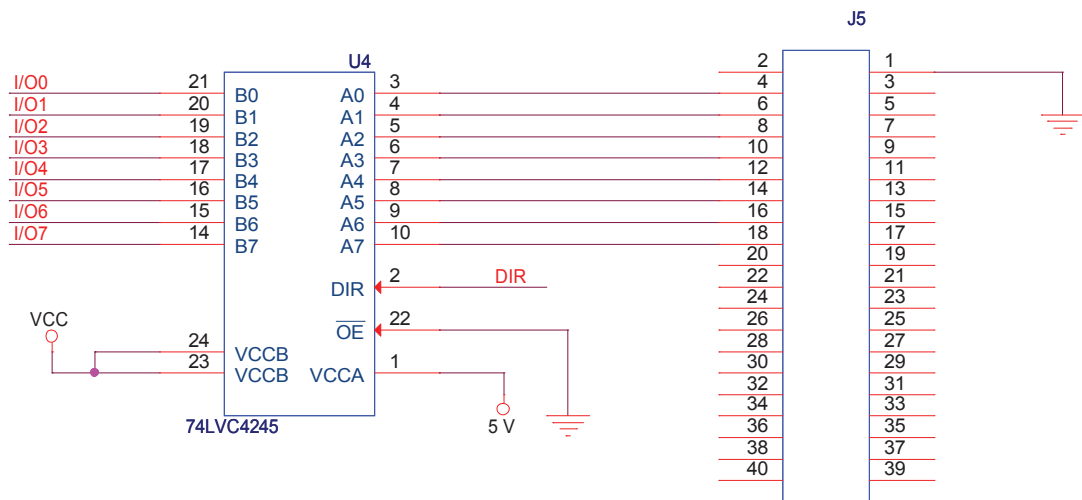
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
GND	V <sub>cc</sub>	V <sub>ee</sub>	RS	R/W	E	DB0	DB1	DB2	DB3	DB4	DB5	DB6	DB7	LED+	LED-

Tabulka 4.1: Konfigurace pinů LCD



## 4.6 I/O brána

Pro realizaci I/O brány byl použit integrovaný obvod 74LVC4245A, protože bylo potřeba, aby deska byla schopná komunikovat se zařízeními pracujícími na 5 V. Jedná se o zařízení s duálním napájením schopném převádět mezi napěťovými úrovněmi (z 3 V na 5 V a obráceně).



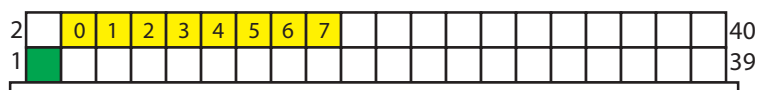
Obrázek 4.5: Zapojení I/O brány

Bohužel z důvodů potřeby postavit zařízení s nízkými náklady není tento obvod schopen automaticky přepínat mezi směry převodu, k tomu je zde pin (DIR), který určuje směr převodu (viz tabulka 4.2).

Vstup		Vstup/Výstup	
OE	DIR	$A_n$	$B_n$
1	0	A = B	inputs
1	1	inputs	B = A
0	X	Z	Z

Tabulka 4.2: Funkce DIR pinu

Na obrázku 4.6 je vidět rozvržení vývodů 40 pinového konektoru. Konektorem se nepřenáší napájení, jelikož se v návrhu nepočítalo s možností externího napájení, protože má deska sama o sobě značný odběr. Žlutě jsou význačeny vývody jednotlivých bitů, zeleně GND.



Obrázek 4.6: Rozvržení vývodů I/O brány, pohled zředu.

## 4.7 Rozvržení pinů CPLD

Pin	Označení	Název	Pin	Označení	Název	Pin	Označení	Název
22	CLK	Hodiny	67	SW0	Přepínač	65	LED6	LED dioda
99	Reset	Reset	68	SW1	Přepínač	66	LED7	LED dioda
48	TCK	JTAG	70	SW2	Přepínač	25	CAD1	7-seg
45	TDI	JTAG	71	SW3	Přepínač	28	CAD2	7-seg
83	TDO	JTAG	72	SW4	Přepínač	29	CAD3	7-seg
47	TMS	JTAG	74	SW5	Přepínač	30	CAD4	7-seg
56	BTN0	Tlačítko	76	SW6	Přepínač	35	A	7-seg
55	BTN1	Tlačítko	77	SW7	Přepínač	32	B	7-seg
54	BTN2	Tlačítko	58	LED0	LED dioda	40	C	7-seg
53	BTN3	Tlačítko	59	LED1	LED dioda	37	D	7-seg
52	BTN4	Tlačítko	60	LED2	LED dioda	36	E	7-seg
50	BTN5	Tlačítko	61	LED3	LED dioda	33	F	7-seg
49	BTN6	Tlačítko	63	LED4	LED dioda	41	G	7-seg
42	BTN7	Tlačítko	64	LED5	LED dioda	39	DP	7-seg
Pin	Označení	Název	Pin	Označení	Název	Pin	Označení	Název
93	I/O0	I/O brana	78	B0	VGA blue	18	R/W	LCD R/W
94	I/O1	I/O brana	79	B1	VGA blue	20	RS	LCD RS
95	I/O2	I/O brana	81	B2	VGA blue			
96	I/O3	I/O brana	90	HS	VGA Hsync			
97	I/O4	I/O brana	91	VS	VGA Vsync			
1	I/O5	I/O brana	16	DB0	LCD data			
6	I/O6	I/O brana	15	DB1	LCD data			
8	I/O7	I/O brana	14	DB2	LCD data			
92	DIR	I/O brana	13	DB3	LCD data			
87	R0	VGA red	12	DB4	LCD data			
89	R1	VGA red	11	DB5	LCD data			
82	G0	VGA green	10	DB6	LCD data			
85	G1	VGA green	9	DB7	LCD data			
86	G2	VGA green	17	LCD_E	LCD enable			

Tabulka 4.3: Rozvržení pinů CPLD

# Kapitola 5

## Demoaplikace

Demoaplikace je navržena tak, aby poskytla představu o funkci desky. Ukázány jsou zde jen pro pokročilejší periferie, protože ty základní jsou pro svou jednoduchost zakomponovány v ukázkách.

### 5.1 7-segmentový displej

Jedná se o jednoduchý návrh, kde se na displej zobrazuje hexadecimální hodnota vstupu. V tomto případě je vstup čtený z přepínačů. Pro ukázkou jsou deklarovány tyto porty:

```
port (clk: in std_logic;
      SWITCH : in std_logic_vector(7 downto 0);
      sseg : out std_logic_vector(6 downto 0);
      ssgsel : out std_logic_vector(3 downto 0)
    );
```

Uvnitř návrhu jsou důležité vnitřní signály, které vytváří nový hodinový impuls, protože hodiny na desce mají vyšší frekvenci.

```
signal clk500 : std_logic; — nové hodiny 500 Hz
signal sseg_reg : std_logic_vector(15 downto 0);
signal digit : std_logic_vector(3 downto 0);
signal count : unsigned(1 downto 0); — čítač do 4
signal prescaler : std_logic_vector(16 downto 0);
```

Každou náběžnou hranou hodin se přičte jednička k signálu prescaler, když je jeho hodnota rovna 50000, tak se překlápí signál clk500 a prescaler se vynuluje. Při dalším dovršení cyklu se hodiny opět překlápí. Tím se vytvoří nové hodiny s frekvencí 500 Hz, což je dostačující frekvence pro bezproblémový chod displeje.

Signál count se navyšuje každou náběžnou hranou signálu clk500, toho se využije v pomyslném cyklu, kde se daná hodnota digitu z signálu sseg\_reg uloží na dobu jednoho impulsu do signálu digit.

```
with count select
  digit <= sseg_reg(3 downto 0) when "00",
          sseg_reg(7 downto 4) when "01",
          sseg_reg(11 downto 8) when "10",
          sseg_reg(15 downto 12) when others;
```

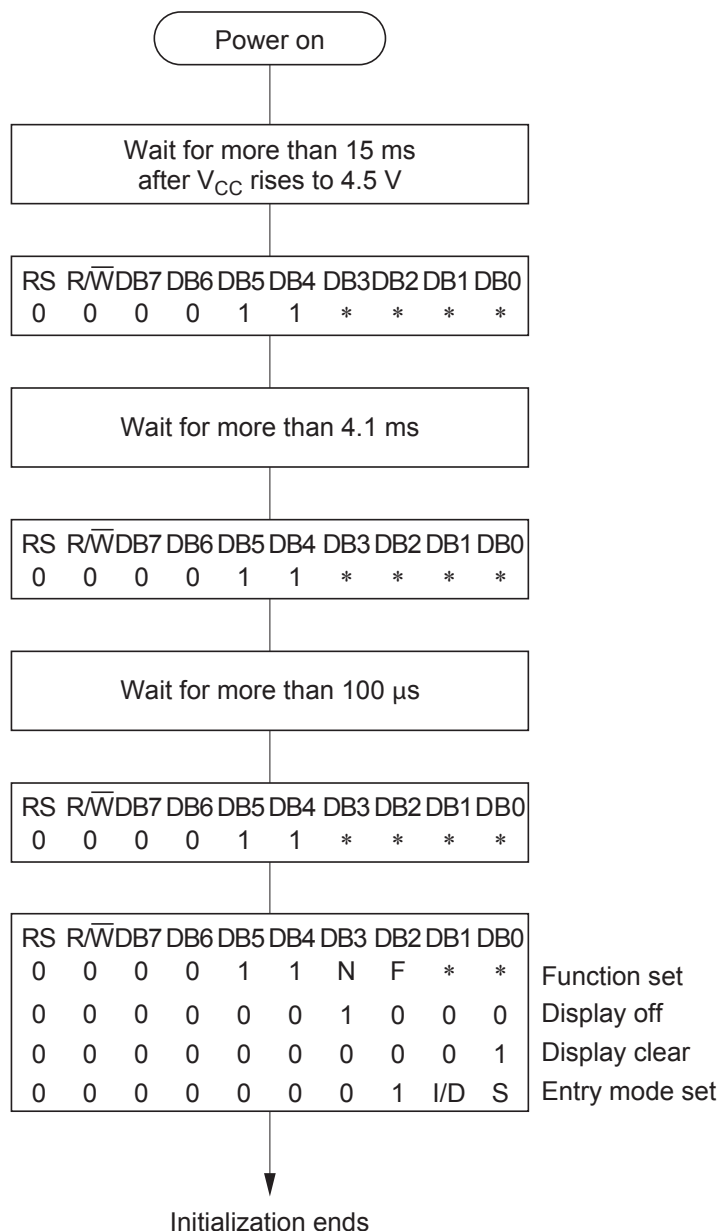
```
with count select
  sseg_sel <= "1110" when "00",
            "1101" when "01",
            "1011" when "10",
            "0111" when others;
```

Než přijde další náběžná hrana hodin, je třeba daný digit rozsvítit na displeji. K tomuto účelu poslouží kód 1 z N, s jeho pomocí rozsvítíme dané segmenty. Pro rozsvícení segmentu je třeba poslat logickou "0" do tranzistoru, který sepne. Segmenty, které se přes CPLD uzemní, neboli daný pin se nastaví do logické "0", se rozsvítí.

```
with digit select
  sseg <= "1111001" when "1110", --1
          "0100100" when "1101", --2
          "0110000" when "1100", --3
          "0011001" when "1011", --4
          "0010010" when "1010", --5
          "0000010" when "1001", --6
          "1111000" when "1000", --7
          "0000000" when "0111", --8
          "0010000" when "0110", --9
          "0001000" when "0101", --A
          "0000011" when "0100", --b
          "1000110" when "0011", --C
          "0100001" when "0010", --d
          "0000110" when "0001", --E
          "0001110" when "0000", --F
          "1000000" when others; --0
```

## 5.2 LCD

Před zahájením "psaní" na LCD displej je třeba provést jeho inicializaci, ta má podle druhu řadiče dané časování, které je potřeba dodržet. Náš zvolený displej je osazen řadičem HD44780 od firmy HITACHI. Tento řadič je hojně využíván, proto je jeho užívání jednoduché.



Obrázek 5.1: Inicializace LCD displeje.

Je důležité mezi každou operací dodržet dobu pro zadání instrukce. V našem případě se

jedná o řádově stovky ns. Všechny časové hodnoty jsou uváděny jako minimální doporučená hodnota, která může být mnohem větší. Je tedy dobré větší hodnoty používat, mohlo by se totiž stát, že se inicializace neprovede správně.

V inicializaci je potřeba nastavit, jak se má displej chovat. Příkaz Function Set má dva parametry N a F, kde N je počet řádků (1 - dvouřádkový, 0 - jednořádkový) a F (1 - 5x10 bodů, 0 - 5x8 bodů) je použitý font. Je důležité mít na paměti, že toto nastavení nelze měnit, když je už inicializace provedena. Další parametry jsou I/D a S v instrukci Entry Mode Set. Tato instrukce udává, jak se bude chovat kurzor při zadávání znaků. Pro I/D = 1 se adresa v paměti inkrementuje (kurzor se posouvá doprava) a při I/D = 0 se adresa dekrementuje (kurzor se posouvá doleva). Parametr S udává, zda se má v daných směrech posouvat celý displej.

Displej má 11 uživatelem ovladatelných pinů, 3 příkazové a 8 datových.

```
port (clk , reset : in std_logic ;
      rw  : out std_logic ; — LCD read/write
      rs  : out std_logic ; — LCD function select
      oe  : out std_logic ; — LCD enable
      data : inout std_logic_vector (7 downto 0) — LCD data
    );
```

Jelikož jsou naše hodiny příliš rychlé, je třeba je upravit dle naší potřeby, abychom mohli správně provést inicializaci. K tomuto účelu nám poslouží signál prescaler, který je rozdělen na dva, prescaler a prescaler1. Signál je rozdělen na dva, abychom ušetřili logické buňky čipu. Kdybychom nechali jeden velký signál, tak by jeho porovnávání spotřebovalo velké množství buněk.

```
signal prescaler : std_logic_vector (13 downto 0);
signal prescaler1 : std_logic_vector (7 downto 0);
signal lcd_ready : std_logic;
```

Prescaler1 zpomalí hodiny 256 krát, jeho nejvyšší bit se tedy stane novými hodinami a prescaler se stane počítadlem jejich taktů. Tím jsme schopni odměřit různé časy a provést inicializaci. Signál lcd\_ready nám pouze indikuje, zda je LCD připraven a zamezuje opětovnému spuštění inicializace.

```
process (clk)
begin
if clk 'event and clk = '1' then
    prescaler1 <= prescaler1 + 1;
end if;
end process;

process (prescaler1 (7))
begin
if prescaler1 (7) 'event and prescaler1 (7) = '1' then
    prescaler <= prescaler + 1;
end if;
end process;
```

Dále už následuje jen samotná inicializace. Hned při začátku inicializace je potřeba vynulovat ovládací piny a dále už se řídit jen pokyny uvedenými v grafu.

```

cpld_control: process (clk)
begin
  if prescaler = 1 and reset = '1' then
    rs <= '0';
    rw <= '0';
    oe <= '0';
    lcd_ready <= '0';
    data <= (others => 'Z');
  elsif clk'event AND clk = '1' then
    if lcd_ready = '0' then
      if prescaler = 2000 then
        oe <= '1';
        data <= "00111000";
      elsif prescaler = 2100 then
        oe <= '0';
      elsif prescaler = 2600 then
        oe <= '1';
        data <= "00111000";
      elsif prescaler = 2700 then
        oe <= '0';
      elsif prescaler = 2720 then
        oe <= '1';
        data <= "00111000";
      elsif prescaler = 2820 then
        oe <= '0';
      elsif prescaler = 3000 then
        oe <= '1';
        data <= "00111000";
      elsif prescaler = 3100 then
        oe <= '0';
      elsif prescaler = 3200 then
        oe <= '1';
        data <= "00001000";
      elsif prescaler = 3400 then
        oe <= '0';
      elsif prescaler = 3500 then
        oe <= '1';
        data <= "00000001";
      elsif prescaler = 3600 then
        oe <= '0';
      elsif prescaler = 3700 then
        oe <= '1';
        data <= "00001110";
    end if;
  end if;
end process;

```

```

    elsif prescaler = 3800 then
        oe <= '0';
    elsif prescaler = 3900 then
        oe <= '1';
        data <= "00000110";
    elsif prescaler > 6000 then
        oe <= '0';
        rs <= '1';
        lcd_ready <= '1';
        data <= (others => 'Z');
    else
        lcd_ready <= '0';
    end if;
    elsif lcd_ready = '1' then
        —LCD je připraveno
    end if;
end if;
end process;

```

Nyní je LCD připraveno k práci. Nedoporučuji zadávat znaky přímo v kódu, jak je použito v ukázce, jelikož by došlo ke značnému úbytku logických buněk. Z tohoto důvodu je třeba použít automat, který bude automatiky převádět znaky na příslušný binární kód a posílat ho na displej.

### 5.3 VGA

Pro otestování VGA portu ukáže navrhnutá aplikace na monitoru škálu šestnácti barev, což k otestování postačí.

```

port( clk , reset : in std_logic;
      VGA: out std_logic_vector(7 downto 0)
      —R, R1, G, G1, B, B1, hsync, vsync
    );

```

VGA je hlavní důvod, proč byly na desku zvoleny hodiny 25 MHz což je, jak bylo uvedeno v popisu VGA standardu, požadovaná hodnota pro zobrazení obrazu s rozlišením 640x480 a zobrazovací frekvencí 60 Hz.

```

signal blank , blankv , blankh , hsync , vsync : std_logic := '0';
signal color : std_logic_vector(5 downto 0);
signal hcount , vcount : integer range 0 to 1000:=0;

```

Signály hcount a vcount jsou čítače, které představují synchronizační signály a signály blank, blankh, blankv slouží k určení, kdy se nacházíme na viditelné ploše, abychom nekreslili mimo obraz. Signál color uchová požadovanou barvu pro zobrazení.

Každý hodinový impuls se pomyslně přesuneme na následující pixel, až se dostaneme na konec řádku, je třeba signál vynulovat a tím se přesunout na další řádek.



```

hcounter: process (clk, reset)
begin
    if reset='1' then
        hcount <= 0;
    else
        if (clk'event and clk='1') then
            if hcount=799 then
                hcount <= 0;
            else
                hcount <= hcount + 1;
            end if;
        end if;
    end if;
end process;

```

```

process (hcount)
begin
    blankh <= '1';
    if hcount>639 then
        blankh <= '0';
    end if;
end process;

```

Proces zpracovávající vertikální synchronizace pomyslně ukazuje, na jakém se nacházíme řádku.

```

vcounter: process (clk, reset)
begin
    if reset='1' then
        vcount <= 0;
    else
        if (clk'event and clk='1') then
            if hcount=699 then
                if vcount=524 then
                    vcount <= 0;
                else
                    vcount <= vcount + 1;
                end if;
            end if;
        end if;
    end if;
end process;

```

```

process (vcount)
begin
    blankv <= '1';
    if vcount>479 then

```

```

        blankv <= '0';
    end if;
end process;

```

Nyní jsme schopni zobrazit na monitor jakýkoliv obraz, jelikož máme zavedený souřadnicový systém. Pokud bychom chtěly zobrazit uprostřed žlutý čtverec, tak je potřeba si uvědomit, že se kreslí po řádkách. Řekněme, že námi zobrazovaný čtverec bude mít strany o délce 50 pixelů, musíme tedy zadat do signálu color žlutou barvu, což je 111100. Pak si musíme spočítat, kdy začít kreslit, jelikož chceme čtverec uprostřed a na jednom řádku je 640 pixelů. Začneme kreslit na 295. pixelu a skončíme na 325. v horizontální ose a začít na 215. a skončit na 265. ve vertikální ose. V našem návrhu, ale zobrazíme jen barevnou škálu.

```

colors: process (clk, reset)
begin
    if reset='1' then
        color <= "000000";
    elsif (clk'event and clk='1') then
        case hcount is
            when 0 to 39 => color <= "000000";
            when 40 to 79 => color <= "110000";
            when 80 to 119 => color <= "001100";
            when 120 to 159 => color <= "000011";
            when 160 to 199 => color <= "111100";
            when 200 to 239 => color <= "001111";
            when 240 to 279 => color <= "110011";
            when 280 to 319 => color <= "100000";
            when 320 to 359 => color <= "001000";
            when 360 to 399 => color <= "000010";
            when 400 to 439 => color <= "101000";
            when 440 to 479 => color <= "001010";
            when 480 to 519 => color <= "100010";
            when 520 to 559 => color <= "101010";
            when 560 to 599 => color <= "010101";
            when 600 to 639 => color <= "111111";
            when others=> NULL;
        end case;
    end if;
end process;

VGA(7 downto 2) <= color and blank&blank&blank&blank&blank&blank;
VGA(1 downto 0) <= hsync & vsync;

```

## 5.4 I/O brána

Jelikož je používání brány velice jednoduché, tak pro ní není napsaná demoaplikace. Pro komunikaci s jiným zařízením je jen potřeba, dát si dobrý pozor na směr komunikace, který si musí uživatel určit sám pomocí pinu DIR.

```
port( clk , reset : in std_logic;  
      data: inout std_logic_vector(7 downto 0);  
      direction: out std_logic — log. "1" Rx  
                                   — log. "0" Tx  
    );
```

Nyní je ovládání již snadné, v proměnné *data* se nachází vždy poslaná nebo přijatá data, podle toho zda přijímáme, nebo odesíláme.



## Kapitola 6

### Závěr

Během oživování desky se zjistilo, že v hardwarovém návrhu jsou menší chyby, které zabraňovali bezproblémové funkčnosti desky. Největší chybou, která se zde vyskytla bylo prohození vstupních a výstupních pinů stabilizátoru, který i přes tento problém desku ochránil. Jeho následná výměna a provedená opatření problém odstranily. Dalším problémem byl zkrat mezi napájením a zemí, který vznikl v průběhu pájení. Kontrolou všech součástek se zjistilo, že jsou některé součástky připájeny nejen na plošku, ale i na rozlitou měď, která tvoří zem desky. Tento problém byl též odstraněn. Na příloženém CD i na konci práce jsou již upravená schémata, která neobsahují zmíněné chyby.

Desky plošných spojů, které vznikly během této práce, byly navrženy v prostředí ORCAD PCBeditor. Pro všechny součástky byly definovány padstacky a footprinty. Desky jsou vyrobeny v 5.třídě přesnosti, do které se vešla bez problému všechna pouzdra až na pouzdro TQFP100, které bylo pod normou o 0,4 milu a už by bylo třeba přejít do 6.třídy. Firma Pragoboard, která desky vyráběla i přesto vyrobila v 5.třídě celou desku bez problému.

Deska má odběr 35 mA při nulovém vytížení, což je o 5 mA méně než odhadovaná hodnota. Jelikož je deska osazena periferiemi, které vyžadují nemalý příkon, jako jsou LED diody a 7-segmentový displej, může se při běžné činnosti spotřeba vyšplhat k hodnotě 400 až 500 mA.

Před dokončením desky byla již přichystána demoaplikace. Všechny moduly bezchybně fungovali hned na první pokus. Problémy byly ale s aplikací pro LCD panel. V průběhu testování se zjistilo, že je program napsaný pro jiný radič a bylo tedy nutné přecházet inicializaci displeje. Další problémy už se nevyskytly.

Podařilo se tedy vytvořit prototyp pro vývojovou desku založený na řešení od firmy XILINX se všemi požadovanými periferiemi. Výsledná cena nákladů na jednu desku se pohybuje po rozpočítání nákladů kolem 1700 Kč.



# Literatura

- [1] Datasheet 74LVC4245AD.  
<http://www.datasheetcatalog.org/datasheet/philips/74LVC4245ADB.pdf>, stav ze 6. 10. 2010.
- [2] Datasheet BC807-25.  
<http://www.datasheetcatalog.org/datasheet/siemens/Q62702-C1736.pdf>, stav ze 6. 10. 2010.
- [3] Datasheet CA56-12SRWA.  
<http://zefiryn.tme.pl/dok/optoelektronika/CA56-12SRWA.pdf>, stav ze 6. 10. 2010.
- [4] Datasheet CFPS-39-25M.  
<http://zefiryn.tme.pl/dok/a09/cfps-39.pdf>, stav ze 6. 10. 2010.
- [5] Datasheet KP-2012SRC-PRV.  
<http://zefiryn.tme.pl/dok/a06/kp-2012src-prv.pdf>, stav ze 6. 10. 2010.
- [6] JAN VOBECKÝ, V. Z. *ELEKTRONIKA Součástky a obvody, principy a příklady*. Grada Publishing, 2005.
- [7] web:Digilent. Digilent Inc. - Digital Design Engineer's Source.  
<http://www.digilentinc.com/>, stav ze 20. 1. 2010.
- [8] web:infobp. K336 Info — pokyny pro psaní bakalářských prací.  
<https://info336.felk.cvut.cz/clanek.php?id=504>, stav ze 4. 12. 2010.
- [9] web:latexdocweb. L<sup>A</sup>T<sub>E</sub>X — online manuál.  
<http://www.cstug.cz/latex/lm/frames.html>, stav ze 10. 12. 2010.
- [10] web:VHDLRef. VHDL Language Reference Guide.  
<http://service.felk.cvut.cz/doc/vhdl/Refguide.htm/>, stav ze 11. 5. 2010.
- [11] web:Xilinx. FPGA and CPLD Solution from Xilinx, Inc.  
<http://www.xilinx.com>, stav ze 20. 1. 2010.
- [12] ZÁHLAVA, V. *Návrh a konstrukce desek plošných spoju*. 1. Vydavatelství ČVUT.
- [13] ZÁHLAVA, V. *OrCAD 10*. Grada Publishing, 2004.





## Kapitola 7

# Seznam použitých zkratek

**VHDL** VHSIC hardware description language

**VGA** Video Graphics Array

**LED** Light emitting diode

**I/O** Vstupně/výstupní brána

**JTAG** Joint Test Action Group

**LCD** Liquid crystal display

**PS2** Personal System/2 connector

**FPGA** Field-programmable gate array

**CPLD** Complex programmable logic device

**PNP** PNP transistor

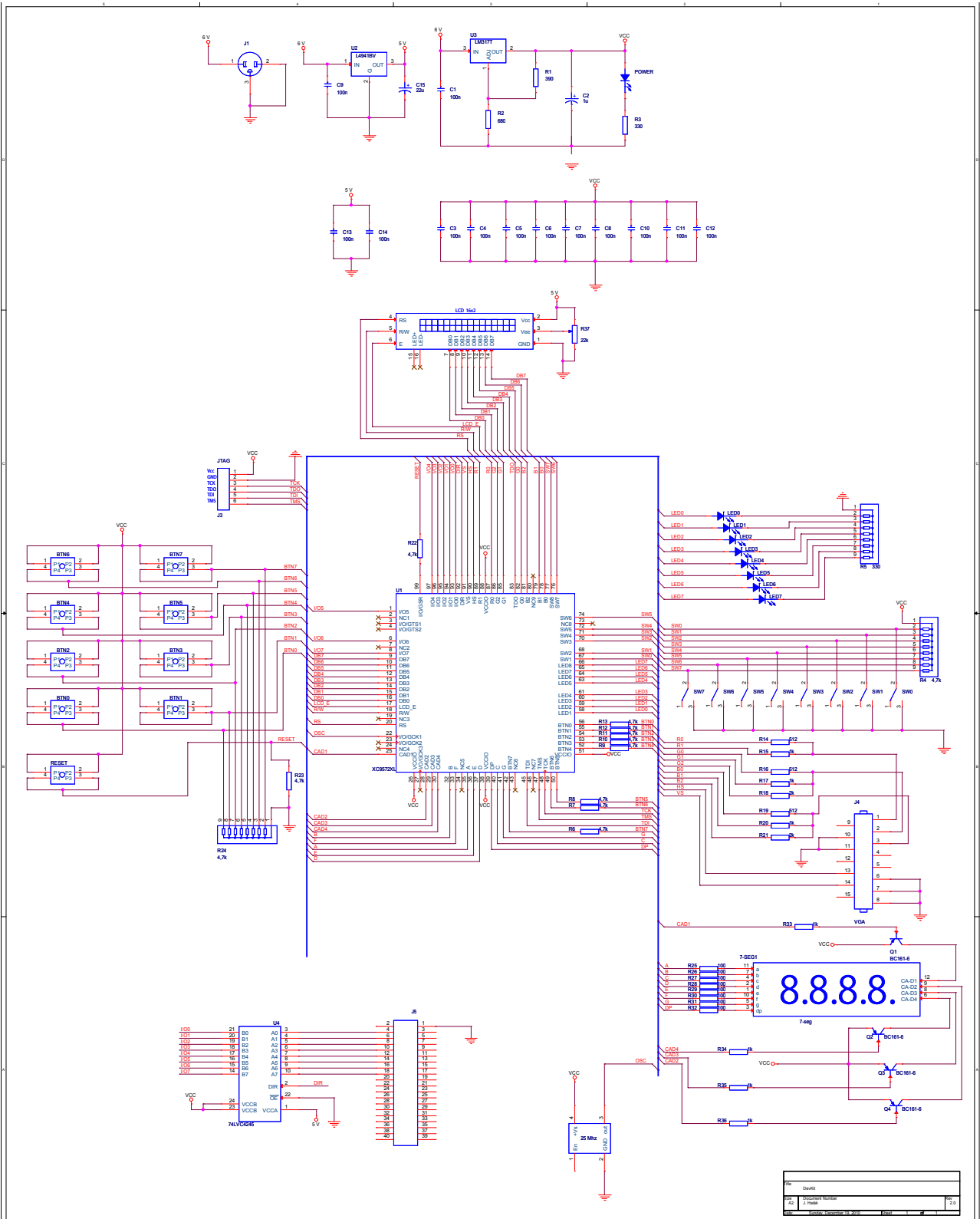
**RGB** RGB color model

⋮



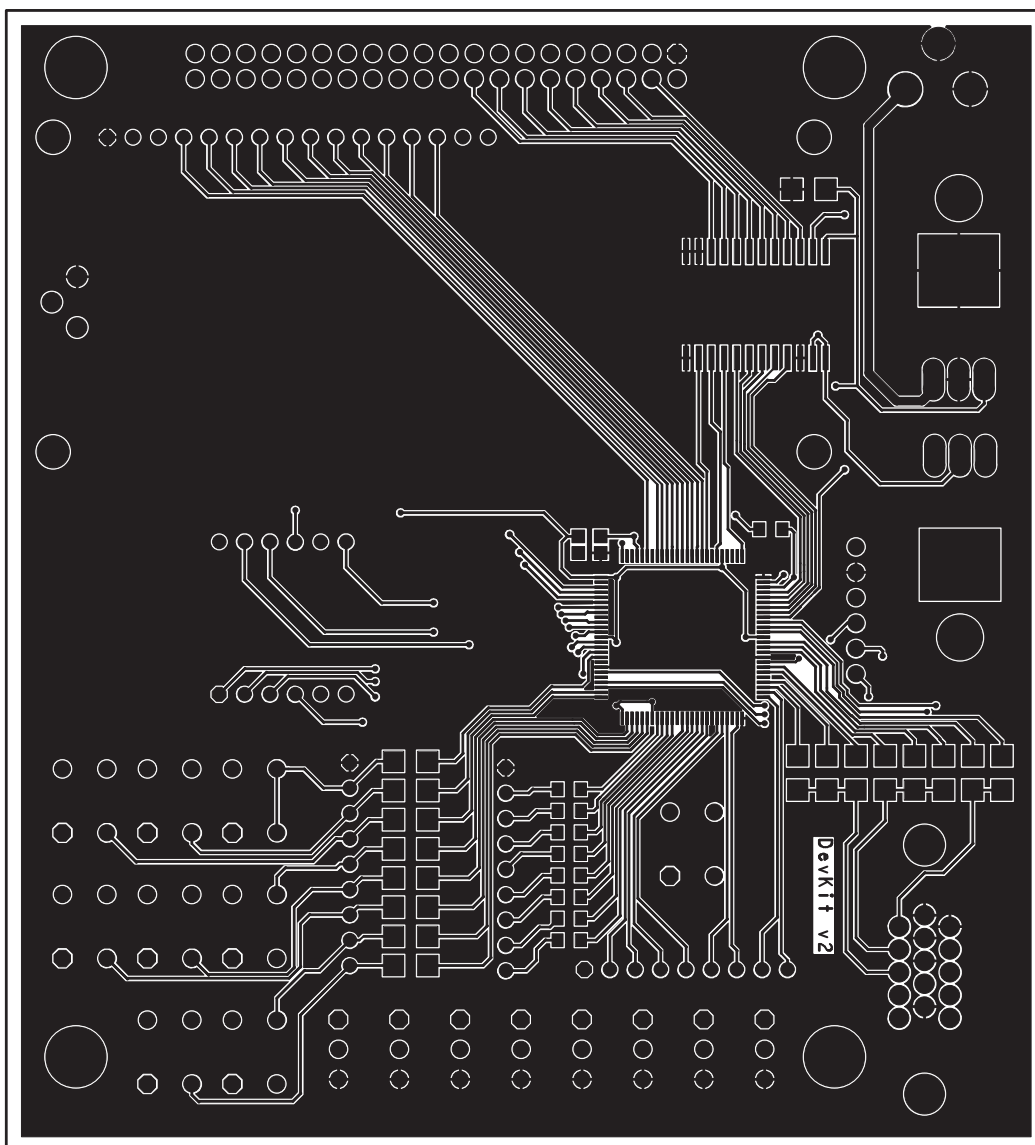
**Příloha A**

**Schéma desky**

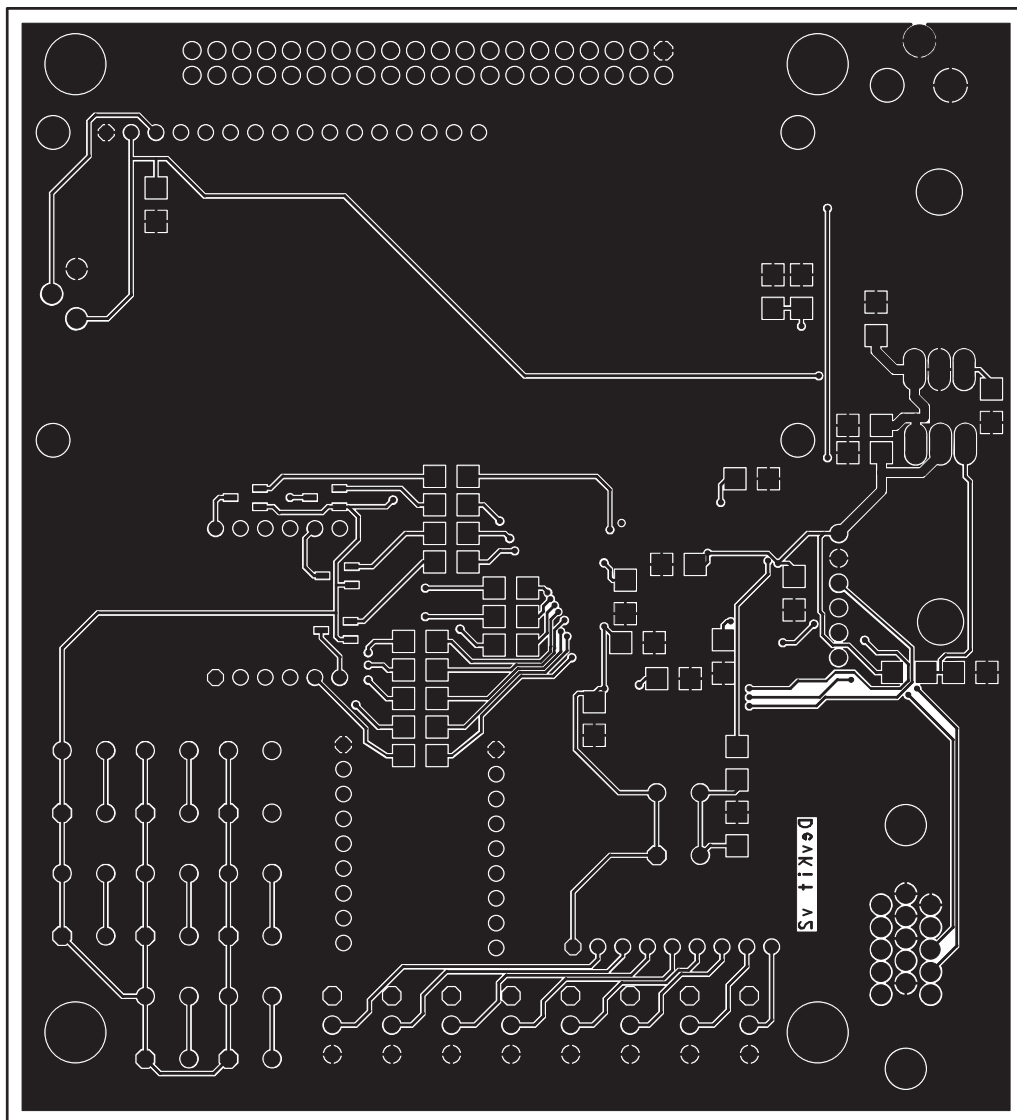


## Příloha B

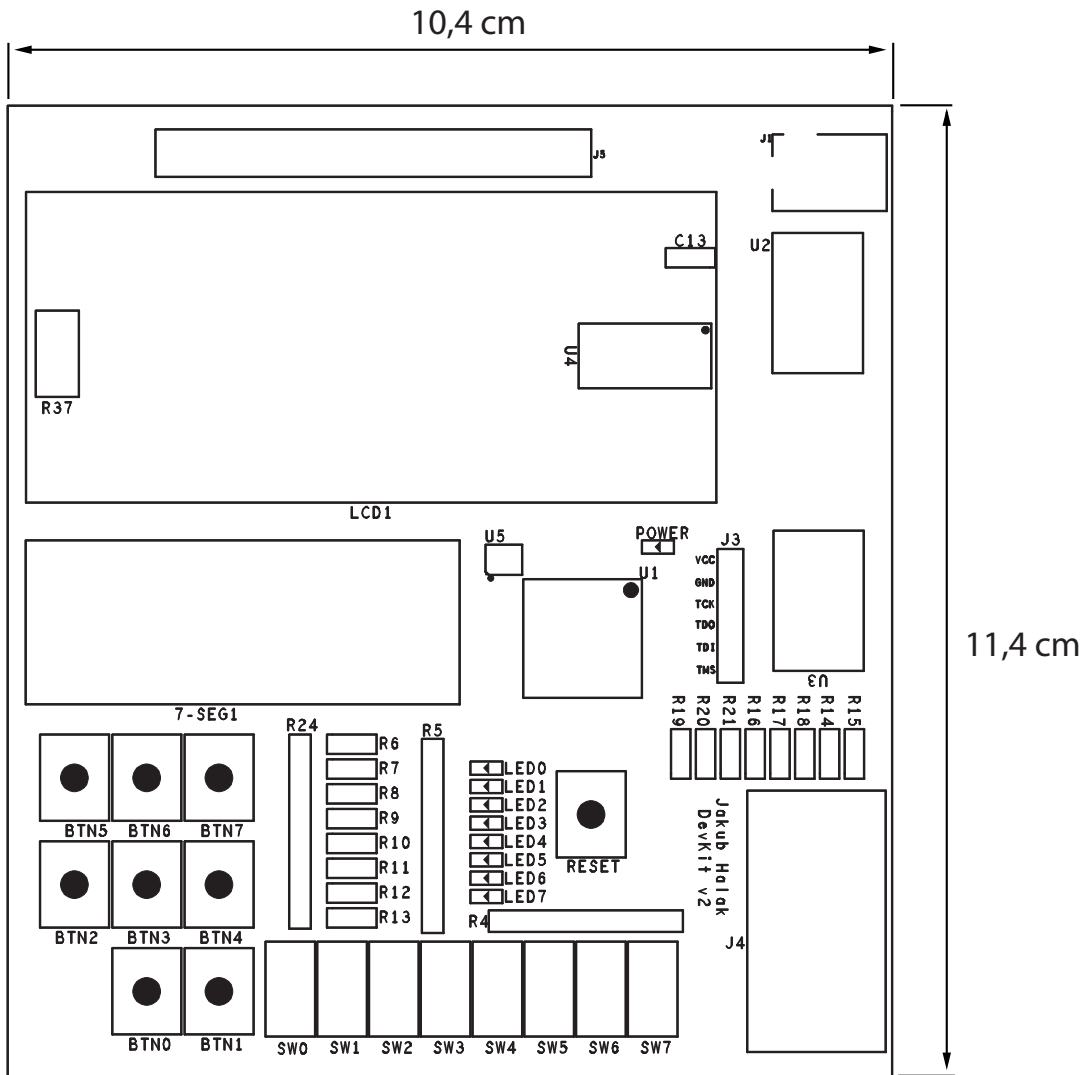
# DPS a rozmístění součástí



Obrázek B.1: Strana Top desky

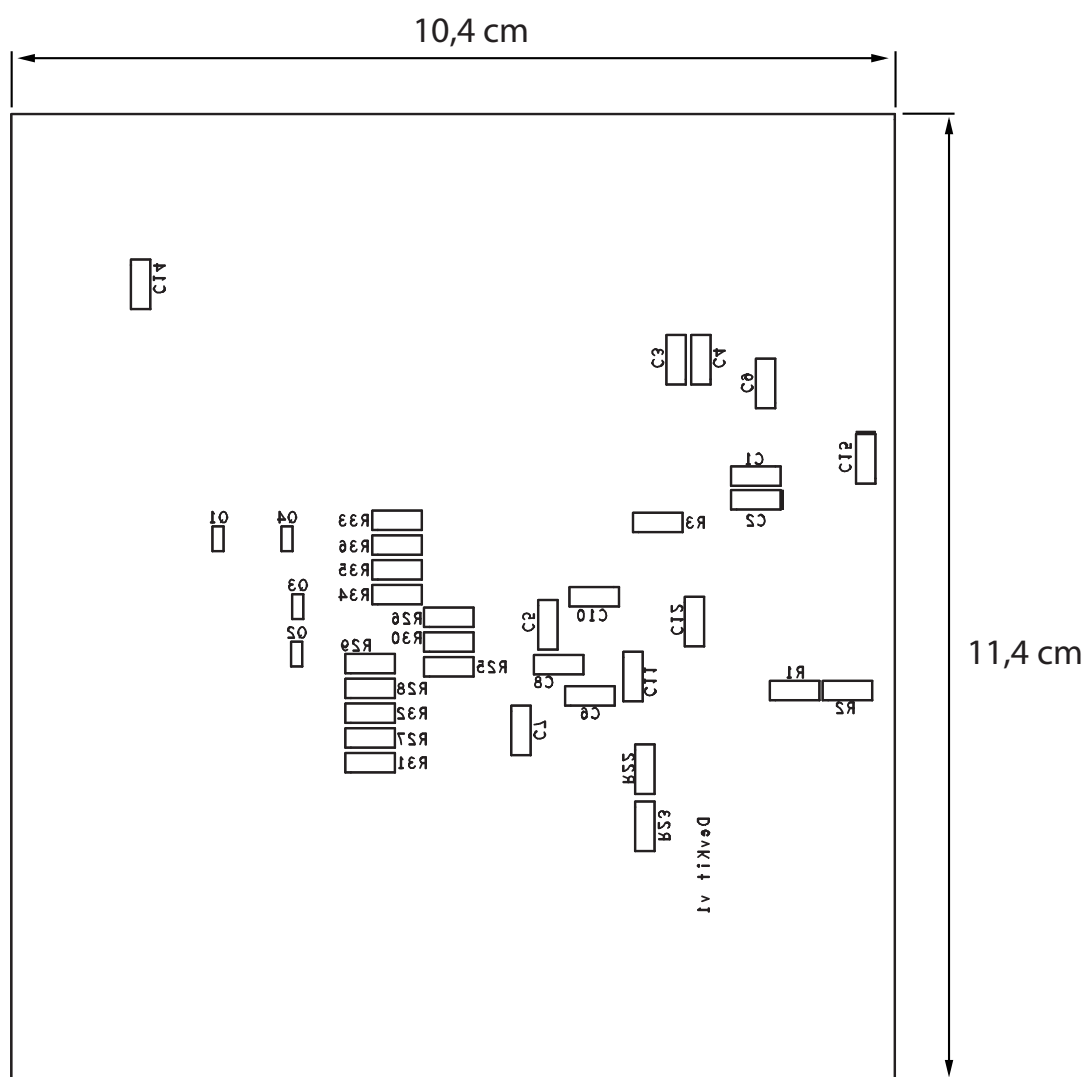


Obrázek B.2: Strana Bottom desky



Obrázek B.3: Rozmístění součástek desky Top



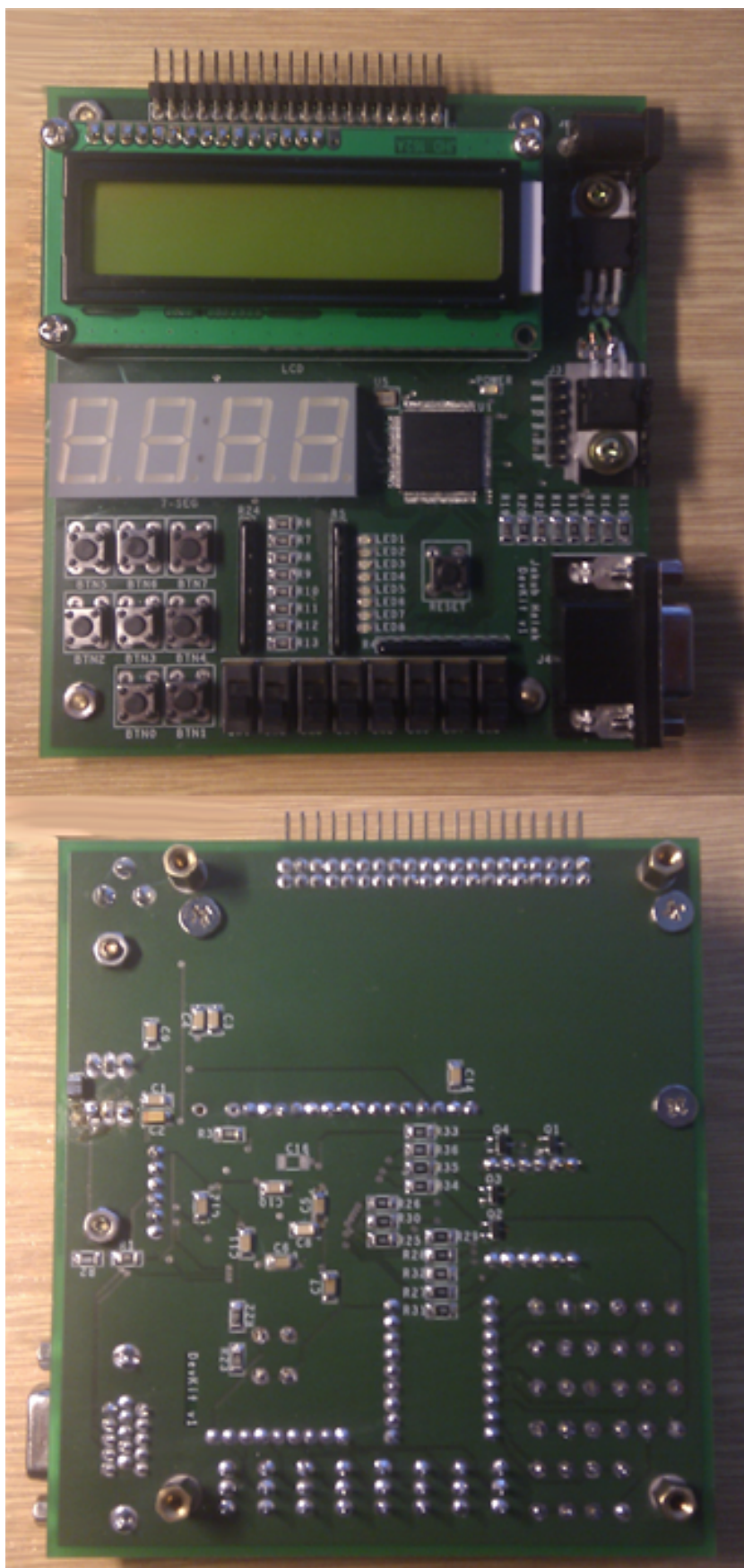


Obrázek B.4: Rozmístění součástek desky Bottom



**Příloha C**

**Osazená deska**



## Příloha D

# Obsah přiloženého CD

- CD
  - demoaplikace – obsahuje projekty demoaplikací
    - \* LCD\_test
    - \* Segment\_test
    - \* VGA\_test
  - podklady – obsahuje schéma a PCD druhé verze desky
    - \* deska
    - \* package
    - \* pad
    - \* schema
  - výroba – obsahuje soubory pro fotoploter a souřadnicovou vrtačku