



Středoškolská technika 2009
Setkání a prezentace prací
středoškolských studentů na ČVUT

VÝVOJOVÁ DESKA
S MIKROKONTROLÉREM ATMEL AVR

Ladislav Synek

Střední průmyslová škola elektrotechnická a Vyšší odborná škola,
Pardubice, Karla IV. 13

„Prohlašuji, že jsem tuto práci vypracoval samostatně a použil jsem literárních pramenů a informací, které cituji a uvádím v seznamu použité literatury a zdrojů informací.“

V Pardubicích, dne

Březen 2009

Obsah

1.	Zadání	3
2.	Anotace	4
3.	Annotation	5
4.	Seznam použitých značek a symbolů	6
5.	Úvod	7
6.	Návrh a popis vývojové desky	8
6.1.	Mikrokontrolér	9
6.2.	LCD displej	12
6.3.	Maticová klávesnice 4 x 4	15
6.4.	Potenciometr 50 k	16
6.5.	Expandér PCF8574P	17
6.6.	Vstupy	20
6.7.	Konektor pro přídavné desky	21
6.8.	Konektor SPI	22
6.9.	Konektor USART	24
6.10.	IR přijímač	25
7.	Návrh a popis přídavné desky reálného času	26
8.	Konstrukční řešení	27
8.1.	Popis řešení jednotlivých elektronických částí vývojové desky	27
8.1.1.	DPS vývojové desky	28
8.1.2.	Osazovací výkres vývojové desky	29
8.1.3.	DPS LCD displeje	30
8.1.4.	Osazovací výkres LCD displeje	30
8.1.5.	DPS přídavné desky reálného času	31
8.1.6.	Osazovací výkres přídavné desky reálného času	31
8.2.	Popis řešení jednotlivých konstrukčních částí výrobku	32
8.2.1.	Pohled na vývojovou desku – bokorys	32
8.2.2.	Rozmístění děr na vývojové desce	33
9.	Oživení	34
9.1.	Popis oživení jednotlivých částí výrobku	34
10.	Rozpiska materiálů a součástek	37
10.1.	Rozpiska elektronických součástek	37
10.2.	Rozpiska elektronických součástek přídavné desky reálného času	38
10.3.	Rozpiska elektrická	39
10.4.	Rozpiska mechanická	40
11.	Seznam použitých odborných výrazů	41
12.	Seznam použitých obrázků	43
13.	Seznam příloh	44
14.	Použité materiály	45
15.	Soupis souborů na CD	46
16.	Závěr	46

1. Zadání

Navrhněte a sestrojte vývojovou desku s mikrokontrolérem Atmel AVR, která by byla srovnatelná s profesionálními vývojovými deskami. Tato deska by měla být víceúčelová, tzn. použitelná ve více elektronických oborech. Měla by být využitelná jak v praxi, tak ve škole jako pomůcka při vyučování elektronických předmětů, např. Radiokomunikační zařízení, Telekomunikační zařízení, Elektronické měření, atd..

2. Anotace

Jestliže chce někdo být v dnešní době dobrý v programování mikrokontrolérů, neměla by mu chybět v elektronické dílně vývojová deska. Vývojová deska pomůže konstruktérovi doladit program a reálně ho odzkoušet. Obsahuje obvody a součástky na co nejlepší odzkoušení a odladění programu. Tato vývojová deska obsahuje:

mikrokontrolér do jehož paměti se ukládá program, který chceme odzkoušet
obsahuje 8 vstupů a výstupů na které lze připojit např. zařízení která jsou na jiné desce
klávesnici s 16-ti tlačítky pro ovládání programu nebo pro funkce které jsme si v programu
nadefinovali

LCD displej na zobrazení toho, co jsme si předem v programu nadefinovali
přijímač infračerveného signálu z dálkových ovladačů, slouží např. na zjištění adres
jednotlivých tlačítek na přijímači, nebo na ovládání pomocí dálkového ovladače
reproduktor na reprodukci jednoduchých tónů

potenciometr na odzkoušení A/D převodníku

konektor pro připojení přídatných desek

konektor USART

konektor SPI na programování mikrokontroléru

Steckverbindung SPI an Programmierung Mikrochip

3. Annotation

Wenn man heutzutage im Microprocessor Programmieren gut ein will, sollte man über eine Entwicklungsplatte verfügen. Diese Entwicklungsplatte hilft der Konstrukter das Programm einzustellen und real abzutüpfen. Sie enthält Distrikte und Bestandteile an möglichst gut Prüfung und Austesten. Diese entwicklungsplatte enthält:

mikrochip in dessen Speichers sich legt ein Programm, die möchten wir erproben
sie enthält 8 Kosteninputs und Aufwärtsgängen an die kann man verbinden beispielsweise
Ausstattungen die sind an andere Platte

Tastatur mit 16- die Druckknopf für Handhabung des Programms oder für Funktionen die wir
sich im Programm definieren

LCD Display an Abbildungen, was wir sich voraus im Programm definierten

Rezeptor von Infrarot-Signal aus Fern-Befehlschalter, dient beispielsweise an Ablesung
Anschriften einzelner Druckknöpfe an Rezeptor, oder an Beherrschung mittels Fern-
Befehlschalter

Lautsprecher an Abspielen reiner Töne

Potentiometer an Prüfung Analog/Digital – Wandler

Steckverbindung für Beifügen erweitert Platte

Steckverbindung USART

4. Seznam použitých značek a symbolů

Kb – 1024 bitů (bit – nejmenší jednotka informace)
kB – 1024 bytů (byte = 8 bitů – jednotka množství dat)
MHz – 10^6 Hz (Hz – Hertz, základní jednotka frekvence)
kHz – 10^3 Hz
0 V – nulové napětí
+5 V – napětí +5 voltů (měřeno proti zemi)
MIPS – milion instrukcí za sekundu (výkon mikroprocesoru)
50 k – 50 kilo ohmů (ohm – jednotka elektrického odporu)
 μ s – miliontina sekundy (sekunda – základní jednotka času)
ms – tisícina sekundy
S2 – název součástky – spínače
X2 – název součástky – konektoru

5. Úvod

Nejen v elektrotechnice se nám probíraná látka zapamatuje lépe, když si ji názorně odzkoušíme. Ve školách v dnešní době panuje spíše takový zvyk, že vše musíme umět teoreticky, ale většinou se už nedozvíme na co lze např. nějaký algoritmus v praxi použít. Až na výjimky lze říci, že když se zrovna nepohybujeme v nějakém problému, kde bychom si museli osvojit daný algoritmus, příklad, vzorec, atd., tak tento algoritmus do hloubky nepochopíme.

Vývojové desky by měly sloužit právě na odzkoušení různých algoritmů v programování, měla by pomoci pochopit jednotlivé algoritmy. Lze s nimi názorně vyučovat jak programování daných mikrokontrolérů, tak i ukázat například jak vypadá PWM modulace jak lze řídit pomocí dálkových ovladačů, atd. Vývojová deska by měla sama o sobě obsahovat co nejvíce periferních obvodů, přídavných modulů, desek, vstupů/výstupů, atd.

Vývojová deska je vlastně něco jako hračka u malého dítěte. Když si s ní hraje, postupně zjišťuje co všechno s ní může dělat, ale když mu ji seberete začne křičet. Vývojová deska je něco podobného, když si na ní budete zkoušet různé aplikace, tak se postupně budete učit co vše se dá udělat. Když vám ji pak vezmou rodiče, že jim nepomáháte a děláte jen na této desce, tak začnete taky rvát, ale ne jen tak do okolí, nýbrž cíleně.

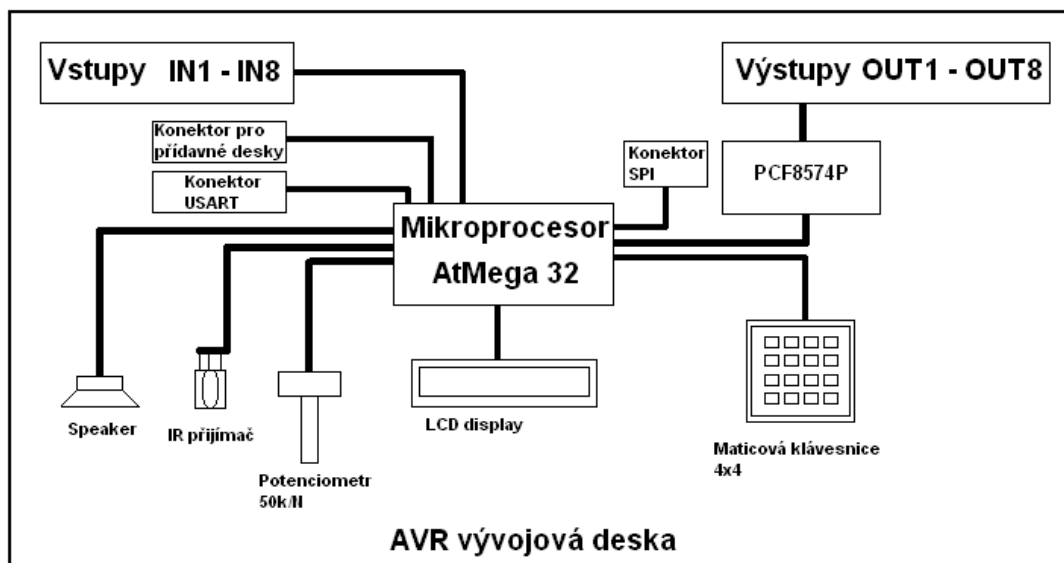
Vývojová deska je tedy elektronické zařízení obsahující mikrokontrolér, na kterém jsou připojeny různé periferie. Vývojová deska se využije všude tam, kde je potřeba vymyslet, odladit a aplikovat program do mikročipu. Využije se nejen v elektrotechnické praxi, ale také v elektrotechnických školách, elektrotechnických kroužcích a vlastně v dnešní době u každého, kdo se něco více dozvědět o mikrokontrolérech.

6. Návrh a popis vývojové desky

V prvním kroku bylo nutné zvolit vhodný typ mikrokontroléru. Věnuji se mikrokontrolérům Atmel AVR a proto jsem si vybral tento typ a tuto architekturu mikrokontrolérů. Mikrokontrolér bylo nutné zvolit s co největším počtem vstupů/výstupů, aby se mohlo připojit mnoho externích zařízení. Seznam vhodných typů mikrokontrolérů se tedy zúžil. S dalším hledáním vhodného typu mikrokontroléru jsem hleděl na jeho parametry a jeho integrované periferie. Jako vhodný mikrokontrolér se zdál být typ At90S8515. Později jsem zjistil, že tento typ mikrokontroléru se už nevyrábí. Hledal jsem proto jiný typ mikrokontroléru a našel jsem AtMega16 a Atmega32. Tyto mikrokontroléry se liší pouze tím, že mají různou paměť Flash pro uložení programu (16 KB a 32 KB). Vybral jsem si nakonec mikrokontrolér AtMega32 na taktovací frekvenci 8 MHz, protože oproti ostatním mikrokontrolérům má nejvíce integrovaných periférií a také poměr ceny a výkonu není špatný.

Jako druhý krok jsem musel sepsat všechny periferie, které budu chtít k mikrokontroléru připojit. Musel jsem navrhnout jakou periférii zapojit na jaký vstup / výstup, abych co nejlépe využil všechny vstupy / výstupy. Periferie, které jsem připojil na jednotlivé vstupy / výstupy mikrokontroléru jsou zachyceny v tabulce v příloze př. 1. Na obrázku obr. 1 jsou zakresleny jednotlivé části vývojové desky.

Blokové schéma



Obr. 1 Blokové schéma vývojové desky

6.1. Mikrokontrolér

Jak bylo řečeno byl vybrán mikrokontrolér Atmel AVR AtMega32. AtMega32 je 8-mi bitový mikrokontrolér pracující s architekturou RISC.

Základní rysy RISC architektury:

- mikrokontrolér s tzv. redukovanou instrukční sadou
- jednoduché instrukce procesoru,
- stejná délka provádění jedné instrukce tzn., že každá instrukce má stejný počet bitů,
- mikroinstrukce jsou implementovány na procesoru – zvýší se rychlost jejich provádění,
- registry jsou víceúčelové (nezáleží na tom, jaký registr si instrukce vybere – zjednodušení překladačů),
- řetězení instrukcí (pipelining)

Základní rysy mikrokontroléru AtMega32:

- Architektura RISC,
- 131 instrukcí,
- pouzdro DIP 40,
- pracovní frekvence až 16 MHz (typ AtMega32) a až 8 MHz (typ AtMega32L),
- pracovní napájecí napětí mikrokontroléru je 2,7 – 5,5 V (typ AtMega32L) a 4,5 – 5,5 V (typ AtMega32),
- minimální proudový odběr,
- Interní RC oscilátor s možností kalibrace,
- 32 8-mi bitových registrů,
- 32 kB programovatelné paměti FLASH pro uložení programu,
- 2 kB paměti SRAM na data,
- 1 kB paměti EEPROM,
- JTAG rozhraní.

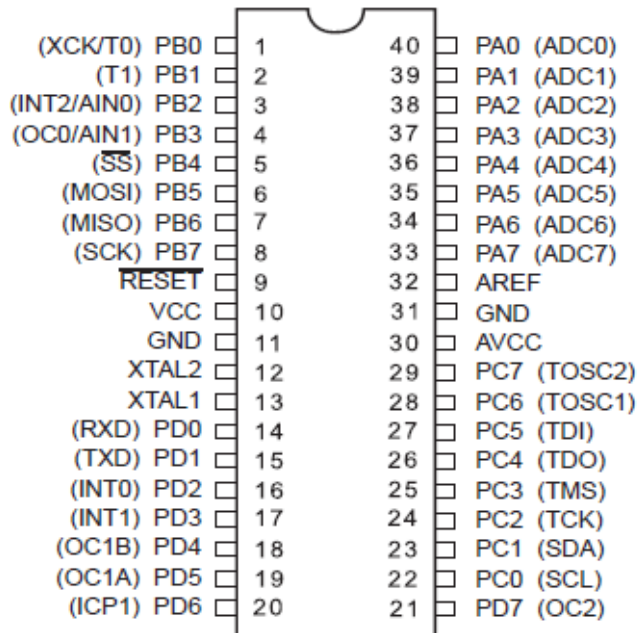
Integrované periférie:

- dva 8-mi bitové a jeden 16-ti bitový čítač/časovač,
- čtyři PWM kanály,
- osm kanálů 10-ti bitového A/D převodníku,
- integrované rozhraní I2C,
- programovatelné sériové rozhraní USART,
- sériové rozhraní SPI,
- programovatelný Watchdog Timer,
- integrovaný analogový komparátor.

Mikrokontrolér postavený na této architektuře pracující na frekvenci 1 MHz má výpočetní výkon 1 MIPS. Kdybychom stejného výpočetního výkonu chtěli dosáhnout u mikrokontroléru s architekturou CISC (INTEL 8051), musel by pracovat na pracovní frekvenci 12 MHz. Mikrokontroléry s architekturou RISC jsou navrženy tak, aby bylo možné pro tyto mikrokontroléry programovat programy ve vyšších programovacích jazycích (jazyk C, Basic, Pascal, atd.).

Mikrokontrolér AtMega32 pracuje na frekvenci až 16 MHz , ale v této konstrukci pracuje tento mikrokontrolér na frekvenci „jen“ 8 MHz (typ AtMega32L)

Na obrázku obr. 2 jsou znázorněny vstupy a výstupy daného mikrokontroléru.

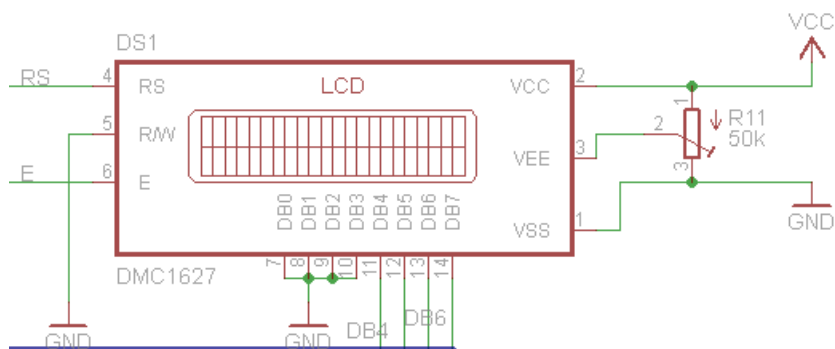


Obr. 2 Konfigurace jednotlivých pinů mikrokontroléru

6.2. LCD displej

V mém výrobku je použit klasický alfanumerický displej s řadičem HD44780. Tento displej se dá ovládat buď 4-bitovým paralelním přenosem dat nebo 4-bitovým paralelním přenosem dat. Ve výrobku je použit 4-bitový přenos, aby se neobsadilo zbytečně moc vstupů / výstupů, které se dají použít pro další periférie.

Displej obsahuje: 8 pinů na vlastní přenos dat do i z displeje, jsou zapojeny jen 4 piny DB4 – DB7 (piny 11 – 14) a piny DB0 – DB3 (piny 7 - 10) jsou spojeny se zemí (4 – bitový přenos); pin RS (pin 4) na resetování displeje; pin E (pin 6) na zapnutí přijímání dat displeje – plní svou funkci např. v zapojení více displejů na jednu sběrnici; pin R/W (pin5) na přepínání stavu zápis nebo čtení – je nastaven napevno zápis (pin je spojen natrvalo se zemí); první tři piny jsou na napájení displeje a na nastavení kontrastu. Detailnější popis LCD displeje s řadičem HD44780 lze nalézt v katalogovém listu na CD. Jednotlivé znaky jsou uloženy v paměti ROM v řadiči displeje. Zapojení LCD displeje je na obrázku obr. 3.



Obr. 3 Zapojení LCD displeje

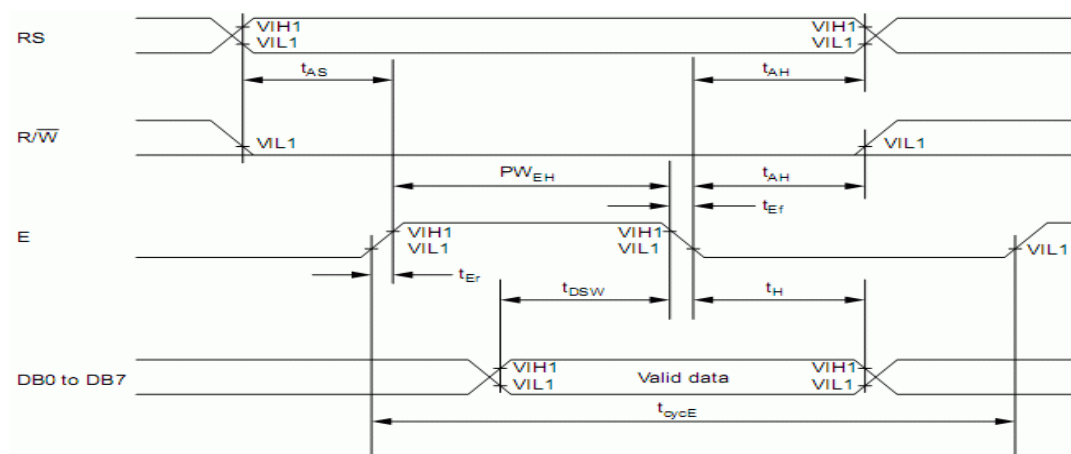
Ovládání displeje:

Displej se ovládá buď pomocí 8-mi bitové sběrnice, nebo pomocí 4 bitové sběrnice. Pro zobrazení jednotlivých znaků z paměti znaků se musí na tyto vstupy poslat určitá posloupnost log. 1 a log. 0. Znaková sada a její log. posloupnosti jsou zobrazeny na obrázku obr. 4 .

Lower 4 Bits	Upper 4 Bits	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
xxxx0000	CG RAM (1)			0	0	P	\	P				-	夕	ミ	α	p	
xxxx0001	(2)		!	1	A	Q	a	q			。	ア	チ	△	ä	q	
xxxx0010	(3)		"	2	B	R	b	r			「	イ	ツ	×	β	θ	
xxxx0011	(4)		#	3	C	S	c	s			」	ウ	テ	ε	ε	∞	
xxxx0100	(5)		\$	4	D	T	d	t			、	エ	ト	†	μ	Ω	
xxxx0101	(6)		%	5	E	U	e	u			・	オ	ナ	1	σ	Ü	
xxxx0110	(7)		&	6	F	V	f	v			ヲ	カ	ニ	ヨ	ρ	Σ	
xxxx0111	(8)		'	7	G	W	g	w			ア	キ	ヌ	ラ	g	π	
xxxx1000	(1)		(8	H	X	h	x			イ	ク	ネ	リ	∫	∞	
xxxx1001	(2))	9	I	Y	i	y			ウ	ケ	ル	レ	∫	∞	
xxxx1010	(3)		*	:	J	Z	j	z			エ	コ	ン	レ	j	∫	
xxxx1011	(4)		+	;	K	[k	[オ	サ	ヒ	ロ	*	∫	
xxxx1100	(5)		,	<	L	¥	l	l			カ	シ	フ	ワ	∫	∞	
xxxx1101	(6)		-	=	M]	m	}			ユ	ズ	ヘ	ン	∫	∞	÷
xxxx1110	(7)		.	>	N	^	n	‡			ヨ	セ	ホ	°	ñ		
xxxx1111	(8)		/	?	O	_	o	€			ツ	リ	マ	°	ö		

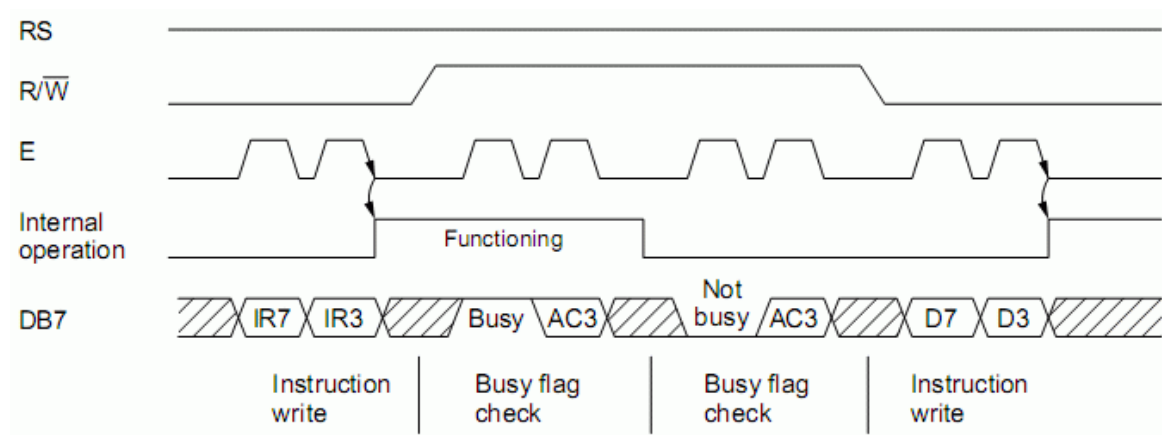
Obr. 4 Znaková sada A00

Data do displeje jsou posílány pomocí pinů DB0 – DB7. Displej nám nebude přijímat žádná data, dokud nenastavíme pin E (povolení přenosu) na log. 1. Na přepnutí mezi instrukcí (log.0) a dat (log.1) je určen pin RS. Princip komunikace s displejem pomocí 8-mi bitového přenosu je zobrazen na obrázku obr. 5.



Obr. 5 Komunikace s LCD displejem – 8-mi bitový přenos

Jestliže se využívá 4 bitový datový přenos, musí se na začátku poslat do displeje instrukce, která řekne displeji, aby se nastavil do 4 bitového módu. Po této instrukci se do displeje posílají data nadvrát. Princip 4 bitové komunikace mikrokontroléru s displejem je na obrázku obr. 6.



Obr. 6 Komunikace s LCD displejem – 4 bitový přenos

Displej pracuje na napájecí napětí 5V. Kontrast displeje lze ovládat pomocí napětí, které jde do pinu VEE na displeji. Čím větší nastavíme na napěťovém děliči napětí, tím větší je kontrast.

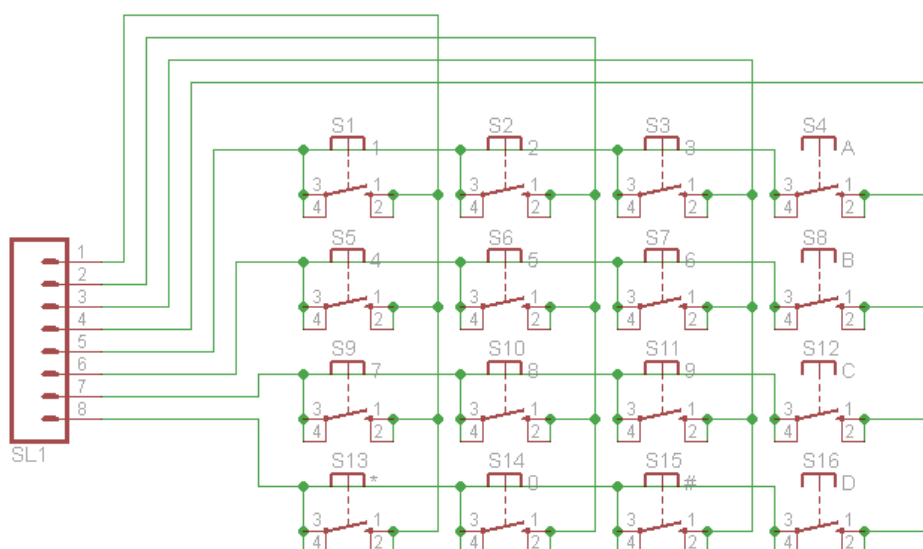
Jednotlivé zapojení pinů LCD displeje je zachyceno v tabulce tab. 2.

LCD displej	
Jednotlivé piny na LCD	vstupy a výstupy procesoru [pin / port]
11 (DB4)	27 / PC5
12 (DB5)	26 / PC4
13 (DB6)	25 / PC3
14 (DB7)	24 / PC2
4 (RS)	29 / PC7
6 (E)	28 / PC6

Tab. 2 Zapojení jednotlivých pinů LCD displeje

6.3. Maticová klávesnice 4 x 4

Maticová klávesnice je zapojena na jeden celý port a to na PORT A. Pokud budeme potřebovat i A/D převodník, nebude nám fungovat sloupec tlačítek A, B, C, D. Jestliže chceme využívat u klávesnice i tlačítka A, B, C, D musíme propojit JUMPEREM JP1 piny 2 a 3, tím pádem zase nebude fungovat A/D převodník. Na vývojové desce je poznačeno do jaké polohy tento JUMPER dát. Jakékoli tlačítko na klávesnici lze nastavit na jakoukoli funkci. Nápis na tlačítkách na funkci tlačítka nemají žádný vliv. Klidně tlačítko 5 může v programu znamenat 4, ale většinou se to takhle nedělá. Například tlačítka 2 a 8 fungují jako šipka nahoru a dolů a tlačítkem 5 se potvrdí nějaká funkce. Schéma vnitřního zapojení maticové klávesnice je zobrazeno na obrázku obr. 7.

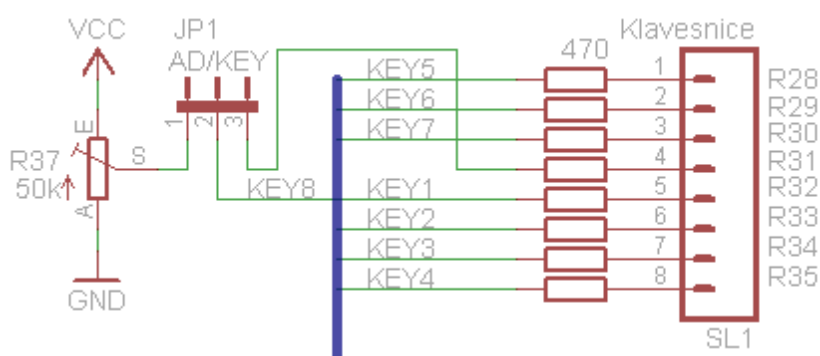


Obr. 7 Vnitřní zapojení maticové klávesnice

Z obrázku je vidět, že klávesnice má 8 vývodů. Uvažujeme-li, že bude vždy zmáčknuté maximálně jedno tlačítko, máme 16 možností jak pospojovat 4 a 4 vodiče v této klávesnici. Proč čtyři a čtyři? Protože máme 4 vodiče tažené svisle a 4 tažené vodorovně a tyto vodiče spojujeme do kříže. Program zajistí postupné přepínání log. 1 na 4 vodičích buď vodorovných nebo svislých. Na druhé straně vyhodnocuje, které tlačítko bylo stisknuto.

6.4. Potenciometr 50 k

Potenciometr je zapojen jako napěťový dělič. Vstupní napětí děliče je 5 V a jeho výstupní napětí je nastavitelné od 0 do 5 V a je přivedeno na vstup 7. kanálu integrovaného A/D převodníku v mikrokontroléru AtMega32. A/D převodník v tomto mikrokontroléru je 10-ti bitový (nabývá hodnoty od 0 do 1023). Pomocí tohoto přípravku si lze odzkoušet např. při jakém otočení potenciometru (při jakém stupni otočení) je jaké napětí. Jestliže chceme ale zkoušet takto A/D převodník, musíme zapojit JUMPER JP1 na polohu pro propojení mikrokontroléru s výstupním napětím z děliče. Při tomto zapojení nebude fungovat sloupec se znaky A, B, C, D. Tento vstup A/D převodníku lze také využít u přídatných desek. V tomto případě se musí JUMPER JP1 vypojit, tzn. že nebude zapojen ani na klávesnici ani na napěťový dělič. A/D převodník je dimenzován pro maximální vstupní napětí +5 V a minimální napětí 0 V. Na obrázku obr. 8 je zakreslen napěťový dělič tvořený potenciometrem.

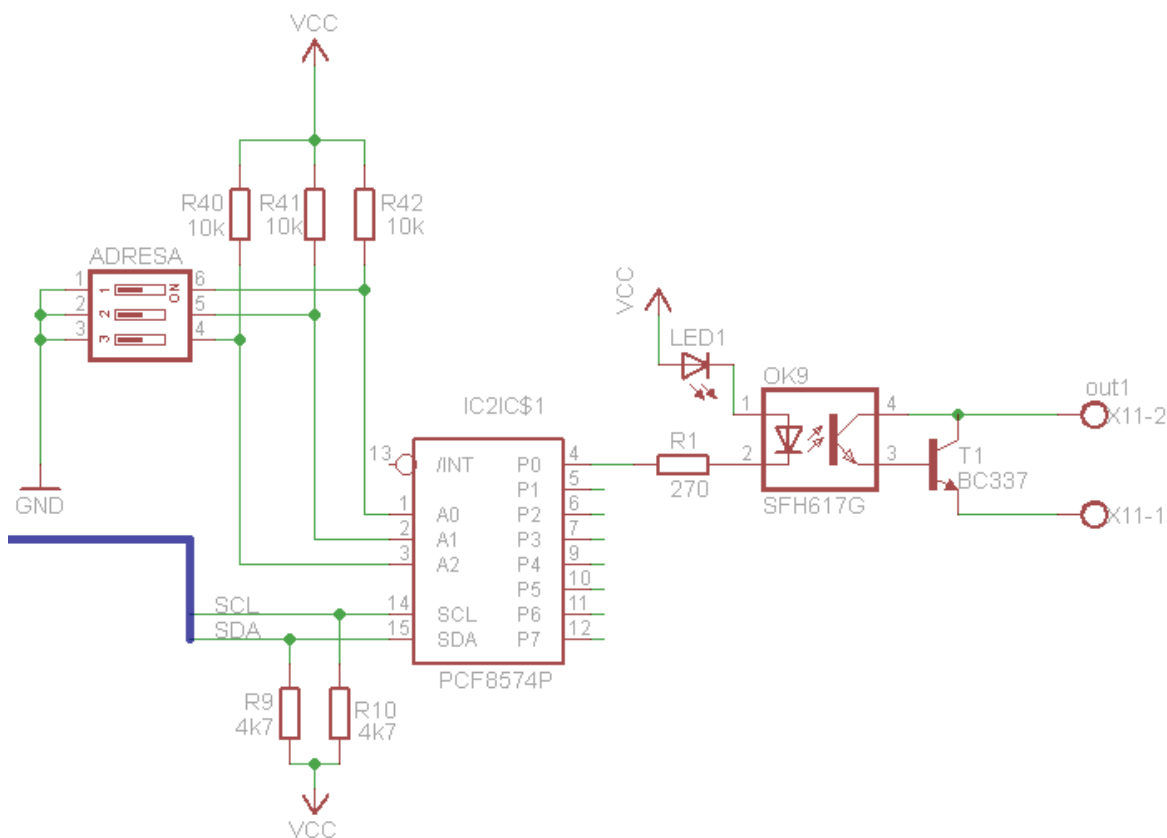


Obr. 8 Zapojení potenciometru, přepínače JP1 a konektoru klávesnice

A/D převodník v mikrokontroléru pracuje na principu postupné aproximace. Doba převodu se pohybuje okolo 65 μ s - 260 μ s. Abychom dosáhli rychlého převodu, museli bychom vhodně vyfiltrovat napájecí napětí A/D převodníku. Zapojení s potenciometrem nevyžaduje velkou rychlost vzorkování, proto nám bohatě stačí na vyfiltrování obyčejný kondenzátor. Jestliže by byla potřeba například na přídatné desce digitalizovat zvuk, bylo by vhodné zvolit jiný zdroj, který by byl velice dobře vyfiltrován. Na obrázku obr. 8 je mimo jiné i zapojení konektoru maticové klávesnice a zapojení přepínače JP1, o kterém bylo psáno výše.

6.5. Expander PCF 8574P (výstupy)

Expander je elektronická součástka, která nám zajišťuje více výstupů nežli vstupů. Tento expander je řízen přes sběrnici I2C, která využívá k přenosu dva vodiče SDA a SCL. Vodič SDA slouží k šíření dat k expanderu, vodič SCL přenáší z mikrokontroléru předem nastavenou pracovní frekvenci, která řídí jak přenos, tak zajišťuje i synchronizaci s posílanými daty. Expander PCF8574P umožňuje i přenos dat směrem k mikrokontroléru. Tato vývojová deska ho ale využívá jen jako výstupní obvod, do kterého se posílají binární data po sběrnici I2C a na výstupech PCF8574P se mění podle těchto binárních dat hodnoty (log. 1 a log. 0). Výstupy z expanderu jsou připojeny na vstupy optočlenů. Na výstupech optočlenů je zapojen spínací tranzistor pro výkonnější spínání. Každý výstup je signalizován LED diodou, která nám indikuje, že daný výstup OUT1 – OUT8 je sepnut. Zapojení jednoho výstupu je zobrazeno na obrázku obr. 9.



Obr. 9 Zapojení jednoho výstupu s expanderem PCF8574P

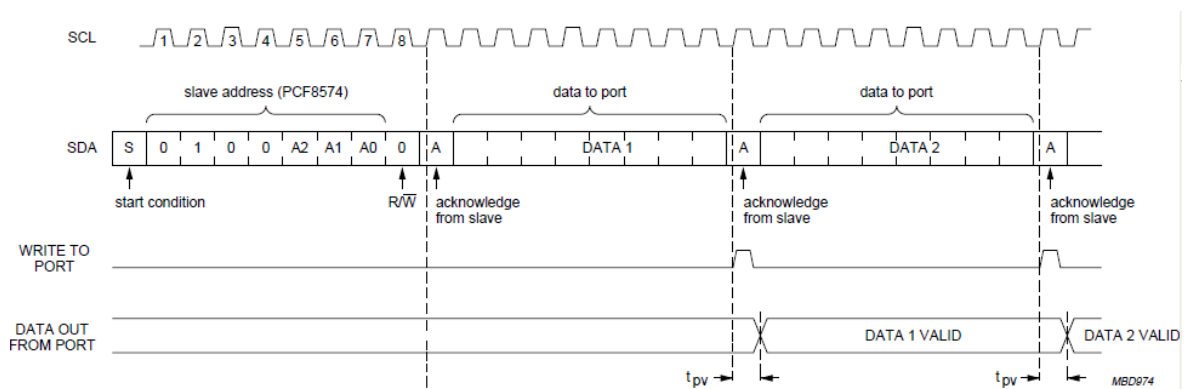
Ovládání expanderu:

Každý obvod, který je ovládán sběrnicí I2C, má přiřazenou určitou adresu. Adresa je složena z adresy, kterou má daný obvod již v sobě od výroby a z adresy, kterou si nastavíme hardwarově pomocí tří vstupů na daném integrovaném obvodu. Některé

integrované obvody mají již v sobě napevno nastaveny třeba jeden nebo dva tyto piny na nastavení hardwarové adresy a jen jeden mají vyveden. Každý obvod, který má jinou funkci, má jinou adresu (např. obvod reálného času má jinou adresu než obvod paměti). Adresa určitého obvodu je napsána v katalogovém listu.

U expandéru budeme potřebovat znát k přenosu dat hlavně jakou si nastavíme přepínačem ADRESA adresu a jaké výstupy obvodu budeme chtít uvést do log. 1 a jaké do log. 0.

Jako první se přenáší tzv. start bit, který nám uvede do činnosti obvod PCF8574P. Následuje adresa, která se skládá ze čtyř bitů. Tyto bity jsou již napevno nastaveny výrobcem, následují tři bity, které si nastavíme přepínačem ADRESA. Poslední bit nám říká, jestli jde o zápis nebo o čtení. Poslední bit se nastavuje v programu mikrokontroléru a u této vývojové desky se nastavuje vždy na 0. Celková adresa je tedy nastavitelná přepínačem ADRESA v rozmezí od 40 hexadecimálně do 4E hexadecimálně. Po odeslání adresy do integrovaného obvodu se odesílá 8-mi bitový řetězec který nám určuje jaký pin na výstupu bude mít jakou logickou hodnotu. Pozor výstupní hodnota je negována, tzn. je-li hodnota řetězce 10000000 tak na výstupu bude 01111111. Je to dáno tím že výstupy jsou zapojeny proti +5 V. Původně jsem počítal, že výstupy z tohoto obvodu jsou negované. Proto je to udělané takto. Mezi jednotlivými daty posílá integrovaný obvod PCF8574P tzv. acknowledge bit, který říká mikrokontroléru, že byly data správně přijata. Jako poslední bit v přenosu je tzv. stop bit, který říká, že je ukončen přenos dat. Na obrázku obr. 10 je nakresleno jak tento přenos vypadá.



Obr. 10 přenos po vodičích SCL a SCK k expanderu PCF8574P

Vodiče SDA a SCL musí být připojeny na pull up rezistory, které jsou připojeny na +5 V a zajišťují log. 1 na těchto vodičích. Bez těchto rezistorů by přenos mezi mikrokontrolérem a daným integrovaným obvodem, který je ovládán přes tuto sběrnici, nefungoval. Toto platí u všech obvodů, které jsou připojeny na tuto sběrnici.

V tabulce tab. 3 je zobrazeno na jaké piny mikrokontroléru je zapojen expander PCF8574P.

Expander PCF 8574P (výstupy)	
vstupy a výstupy periferní součástky	vstupy a výstupy procesoru [pin / port]
SDA	23 / SDA
SCL	22 / SCL

Tab. 3 Zapojení expanderu PCF8574P na mikrokontrolér

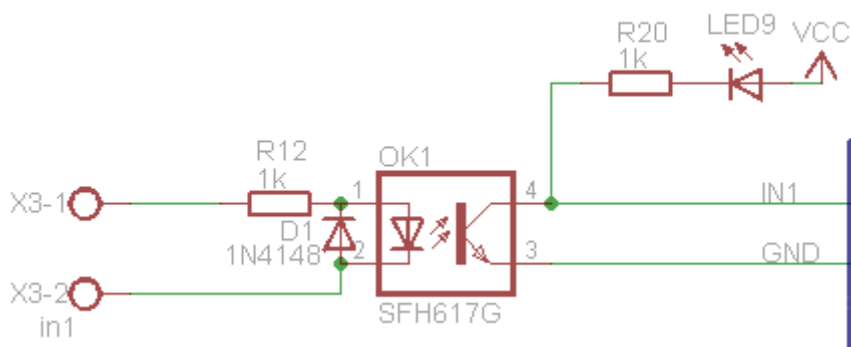
6.6. Vstupy

Vstupy na vývojové desce jsou zapojeny na optočleny, které nám oddělují externí obvody zapojené k vývojové desce od vstupů na mikrokontroléru. Před těmito optočleny je zapojena omezovací obvod (dioda s rezistorem). Za optočleny jsou zapojeny indikační LED diody, které nám indikují na jakém vstupu je log. 1 a na kterém log. 0 (LED svítí = log. 1, LED nesvítí = log. 0) Tyto vstupy jsou dimenzovány na minimální napětí 5 V a na maximální napětí 20 V. Jaký vstup (IN1 – IN8) je zapojen na kterém vstupu mikrokontroléru je rozepsáno v následující tabulce tab. 4.

Vstupy	
vstupy a výstupy periferní součástky	vstupy a výstupy procesoru [pin / port]
IN1	19 / PD5
IN2	18 / PD4
IN3	17 / PD3 (INT1)
IN4	20 / PD6
IN5	4 / PB3 (AIN1)
IN6	3 / PB2 (AIN0)
IN7	2 / PB1 (T1)
IN8	1 / PB0 (T0)

Tab. 4 Připojení jednotlivých vstupů k pinům mikrokontroléru

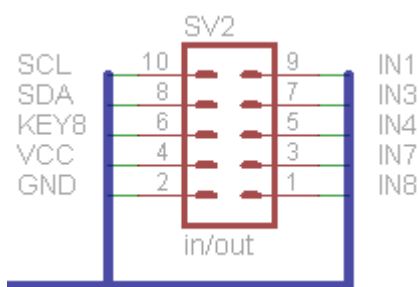
Na obrázku obr. 11 je zobrazen jak vypadá jeden vstup. Těchto vstupů je na vývojové desce celkem 8.



Obr. 11 Zapojení jednoho vstupu

6.7. Konektor pro přídavné desky

Tento konektor obsahuje důležité vstupy / výstupy na připojení přídavných desek. Vstupy a výstupy, které jsou na tomto konektoru zapojeny jsou rozepsány v tabulce tab.5. Na tento konektor lze připojit přídavné desky které nám budou pracovat třeba i nezávisle na vývojové desce a mikrokontroléru. Mikrokontroléru se jen posílají data, která se na této přídavné desce zpracovávají. Takto jsem například navrhl přídavnou desku, která zajišťuje reálný čas. Data z této desky se posílají na vyhodnocení do mikrokontroléru. Toto podrobněji popíši dále. Tento konektor je takřka univerzální, ale musíme brát ohled na to, že tyto vstupy / výstupy jsou propojeny ještě s externími vstupy, na které jsou připojeny optočleny, proto když použijí například vstup IN7, nemohu zároveň používat na konektoru pro přídavné desky pin 3. Na obrázku obr. 8 je zachyceno zapojení konektoru pro přídavné desky. Následující tabulka tab. 5 zachycuje jak jsou zapojeny piny na konektoru pro přídavné desky a jaké piny jsou použity vícekrát.



Obr. 12 Zapojení konektoru pro přídavné desky

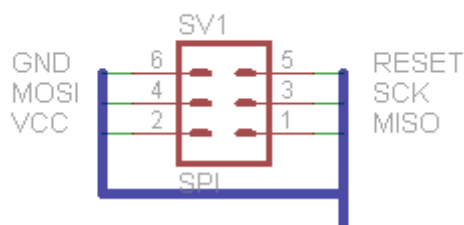
Konektor pro přídavné desky	
vstupy a výstupy periferní součástky	vstupy a výstupy procesoru [pin / port]
1	1 / PB0 (T0)
2	GND
3	2 / PB1 (T1)
4	VCC
5	20 / PD6
6	33 / PA7 (ADC7)
7	17 / PD3 (INT1)
8	23 / SDA
9	19 / PD5
10	22 / SCL

Tab. 5 Propojení konektoru s mikrokontrolérem

6.8. Konektor SPI

Konektor SPI slouží na propojení vývojové desky s programátorem a na případné propojení s obvodem, který je ovládán přes toto rozhraní.

Programátor je takové zařízení, které zajišťuje propojení mikrokontroléru s počítačem a jeho naprogramování přes k tomu určenou aplikaci, která komunikuje s programátorem a ten pak provádí vlastní programování mikrokontrolérů. Bez tohoto konektoru bych byl nucen při každém přeprogramování vyndávat mikrokontrolér z patice, což by bylo velmi nepraktické. Tento konektor mě tedy usnadňuje práci. Zapojení jednotlivých pinů tohoto konektoru je zaznamenáno v tabulce tab. 6. zapojení konektoru je zakresleno na obrázku obr. 9 .



Obr. 13 Zapojení konektoru SPI

Konektor SPI	
Pin na konektoru	vstupy a výstupy procesoru [pin / port]
1	7 / PB6 (MISO)
2	VCC
3	8 / PB7 (SCK)
4	6 / PB5 (MOSI)
5	9 / RESET
6	GND

Tab. 6 Zapojení jednotlivých pinů na mikrokontrolér

Rozhraní SPI:

SPI (Serial Peripheral Interface) vysokorychlostní sériové rozhraní, které se například používá na propojení mikrokontroléru s paměťmi FLASH, A/D převodníky, propojení mezi mikrokontroléry a v neposlední řadě se používá na programování mikrokontrolérů.

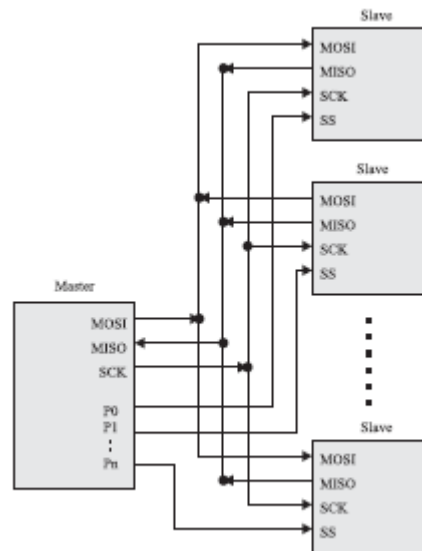
V obvodu může být zapojeno dva a více obvodů. Výběr obvodu se provádí pomocí vodiče SS (Slave Select) a posláním adresy obvodu, s kterým chceme komunikovat.

Datovým výstupem MOSI (Master Out, Slave In) se provádí přenos dat směrem od mikrokontroléru k danému obvodu nebo obvodům.

Datovým vstupem MISO (Master In, Slave Out) se provádí přenos dat směrem od daného obvodu k mikrokontroléru.

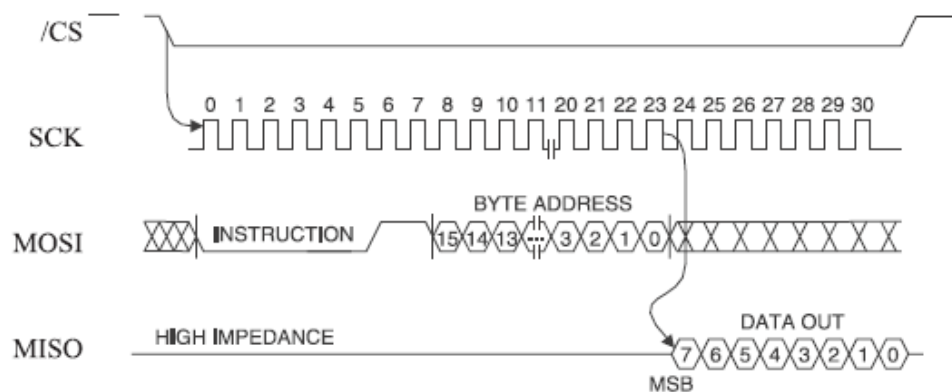
Vodičem SCK se posílá frekvence hodinového signálu (až 2 MHz) z mikrokontroléru směrem ke všem připojeným obvodům.

Na obrázku obr. 14 je znázorněno propojení obvodů s rozhraním SPI.



Obr. 14 Koncepce systému se sběrnici SPI

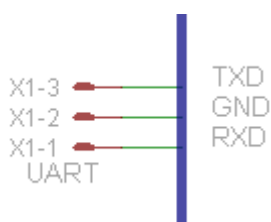
Každý připojený obvod ke sběrnici SPI má přidělenou vlastní adresu. Bez této adresy nelze daný obvod ovládat. Délka vysílaných dat je buď 8 bitů (byte) nebo 16 bitů (word). Na obrázku obr. 15 je znázorněn přenos dat při čtení z paměti FLASH připojené k mikrokontroléru pomocí sběrnice SPI.



Obr. 15 Časové průběhy na jednotlivých vodičích rozhraní SPI

6.9. Konektor USART

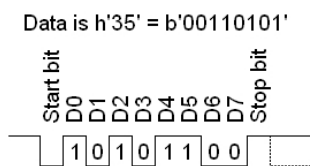
Tento konektor slouží na jednoduché propojení mikrokontroléru s PC nebo s dalším mikrokontrolérem (druhá varianta se moc nepoužívá). Jestliže je potřeba propojit mikrokontrolér s PC, například na vyhodnocování změřených či zaznamenaných hodnot, je tento konektor dostačující. Nejvyšší rychlost, která se pomocí tohoto rozhraní dá přenést, je 250 kb/s. Tato rychlost je opravdu nejvyšší, která se dá pomocí tohoto rozhraní přenášet, ale běžně se setkáváme s mnohem nižší rychlostí. Pro propojení s PC je možné využít obvod od firmy Maxim MAX232. Tento obvod je důležitou součástí, která nám odděluje rozhraní PC a rozhraní USART mikrokontroléru. Kdyby tato součástka chyběla mezi jednotlivými rozhraními, mohlo by dojít k poničení PC nebo USART mikrokontroléru. Pro tento přenos musí být u PC port COM. U nových počítačů se tento port již moc nevyskytuje, ale dají se koupit převodníky USB - COM, které nám toto rozhraní poskytnou. Sám používám pro propojení vlastní programátor, který má i rozhraní USART. Odkaz na tento programátor je na konci tohoto dokumentu. Zapojení konektoru USART je na obrázku obr. 16. Propojení s druhým zařízením musí být křížené tzn., že k pinu RX na vývojové desce musí být připojen pin TX zapojený na druhém zařízení a na pinu TX na vývojové desce musí být připojen pin RX na druhém zařízení. Kdyby se tyto vodiče nekřížily, nebylo by spojení navázáno. V tabulce tab. 7 je znázorněno zapojení pinů na konektoru USART.



USART	
pin na konektoru	vstupy a výstupy procesoru [pin / port]
1	14 / PD0 (RXD)
2	GND
3	15 / PD1 (TXD)

Obr. 16 Zapojení USARTn Tab. 7 Propojení konektoru SPI s mikrokontrolérem

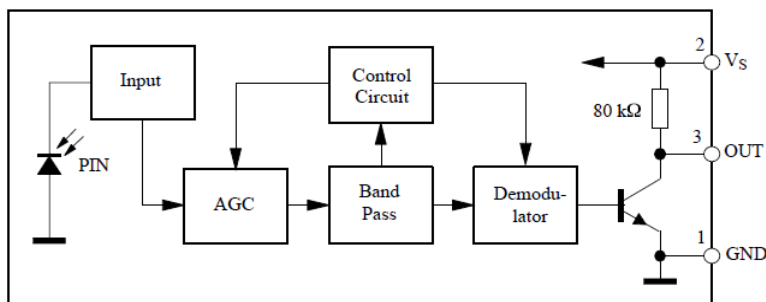
Přenos dat pomocí rozhraní USART je prováděn asynchronně a je plně duplexní, tzn. že lze zároveň zapisovat a zároveň číst. Na obrázku obr. 17 je znázorněn asynchronní přenos dat USARTu.



Obr. 17 Asynchronní přenos dat

6.10. IR přijímač

Aby bylo možné pracovat na této vývojové desce i s IR dálkovými ovladači, je nutné ji vybavit IR přijímačem. Tento přijímač bude přijímat IR signál a na jeho výstupu bude signál TTL připojitelný k mikroprocesoru. Jako IR přijímač je na této vývojové desce použit typ TSOP1230. IR přijímač je pro přijímání IR signálu frekvencí nosné vlny 30 kHz z takřka jakéhokoli dálkového ovladače. TSOP1230 obsahuje: vysoce citlivou PIN fotodiodu, automatické zesílení, pásmovou propust, která propustí jen frekvenci 30 kHz a demodulátor signálu z PAM na PCM. Vnitřní zapojení IR přijímače je zobrazeno na obrázku obr. 18. Vysílač IR většinou vysílá kódovaný protokol (RC5, RC6, NEC, atd.). O zpracování těchto protokolů se již stará připojený mikrokontrolér. Ten musí vyhodnotit adresu přijímače a příkaz, který vysílač vysílá při stisknutí klávese. Tato rutina probíhá u každého zmíněného protokolu.

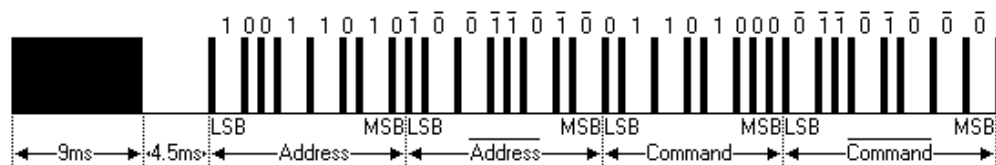


Obr. 18 Blokové schéma IR přijímače TSOP1230

Vysílač, který jsem použil, pracuje s protokolem NEC. Tento protokol ve shrnutí popíší.

Protokol NEC: Přenosový protokol NEC je modulován modulací PŠM. Zvláštností tohoto protokolu je, že má konstantní délku 110ms, i když je modulován PŠM. Adresa i příkaz, je proto vysílán dvakrát (jednou normálně a jednou negovaně), aby nedošlo k chybnému přenosu.

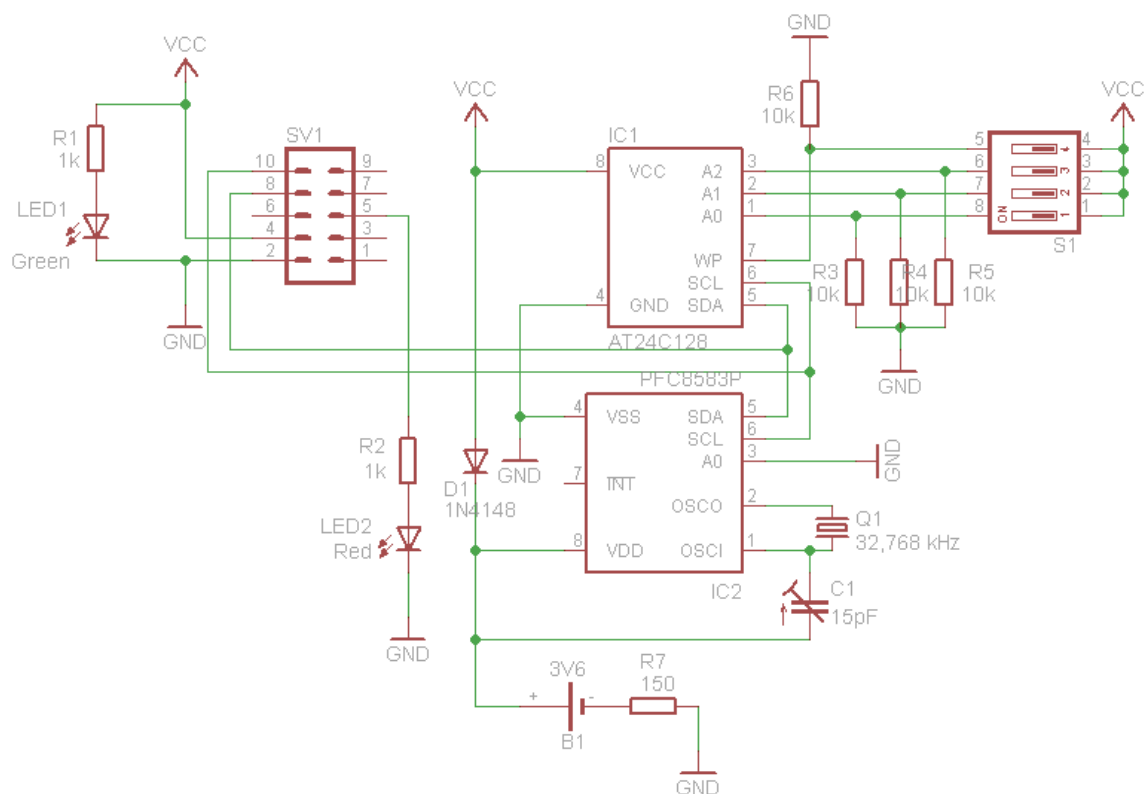
Přenos začíná tzv. burstem délky 9 ms následuje 4,5 ms. Po pauze následuje 8-mi bitová adresa vysílače a hned za ní negovaná 8-mi bitová adresa vysílače. Následuje 8-mi bitový příkaz a hned negovaný 8-mi bitový příkaz. Na obrázku Obr. 19 je tento přenos znázorněn.



Obr. 19 NEC protokol

7. Návrh a popis přídatné desky reálného času

Tuto desku jsem navrhl jen jako demonstraci toho, jak lze vyrobit přídatnou desku k této vývojové desce. Přídatná deska obsahuje integrovaný obvod reálného času PCF8583, záložní baterii pro tento obvod a ještě obvod paměti EEPROM 128 kb od firmy Atmel AT24C128. Tyto dva obvody mohou pracovat spolu, ale mohou pracovat i nezávisle na sobě. Obvod paměti je umístěn na této desce jen z důvodu, abych využil prostor na této desce. Bylo by nesmyslné dělat pro tento obvod vlastní desku, když na této desce bylo ještě dost místa na jeho umístění a tento integrovaný obvod nijak neovlivní funkci integrovaného obvodu reálného času. Integrovaný obvod reálného času je řízen krystalem na frekvenci 32,768 kHz, k mikrokontroléru na vývojové desce je připojen pomocí sběrnice I2C a má nastavenou napevno adresu pro čtení A1 hexadecimálně a pro zápis A0 hexadecimálně. U integrovaného obvodu paměti EEPROM lze nastavit adresu přepínačem S1 a lze u tohoto obvodu ještě povolit nebo zakázat zápis. Celkové schéma této přídatné desky je na obrázku obr. 20.



Obr. 20 Zapojení přídatné desky reálného času

8. Konstrukční řešení

8.1. Popis řešení jednotlivých elektronických částí vývojové desky

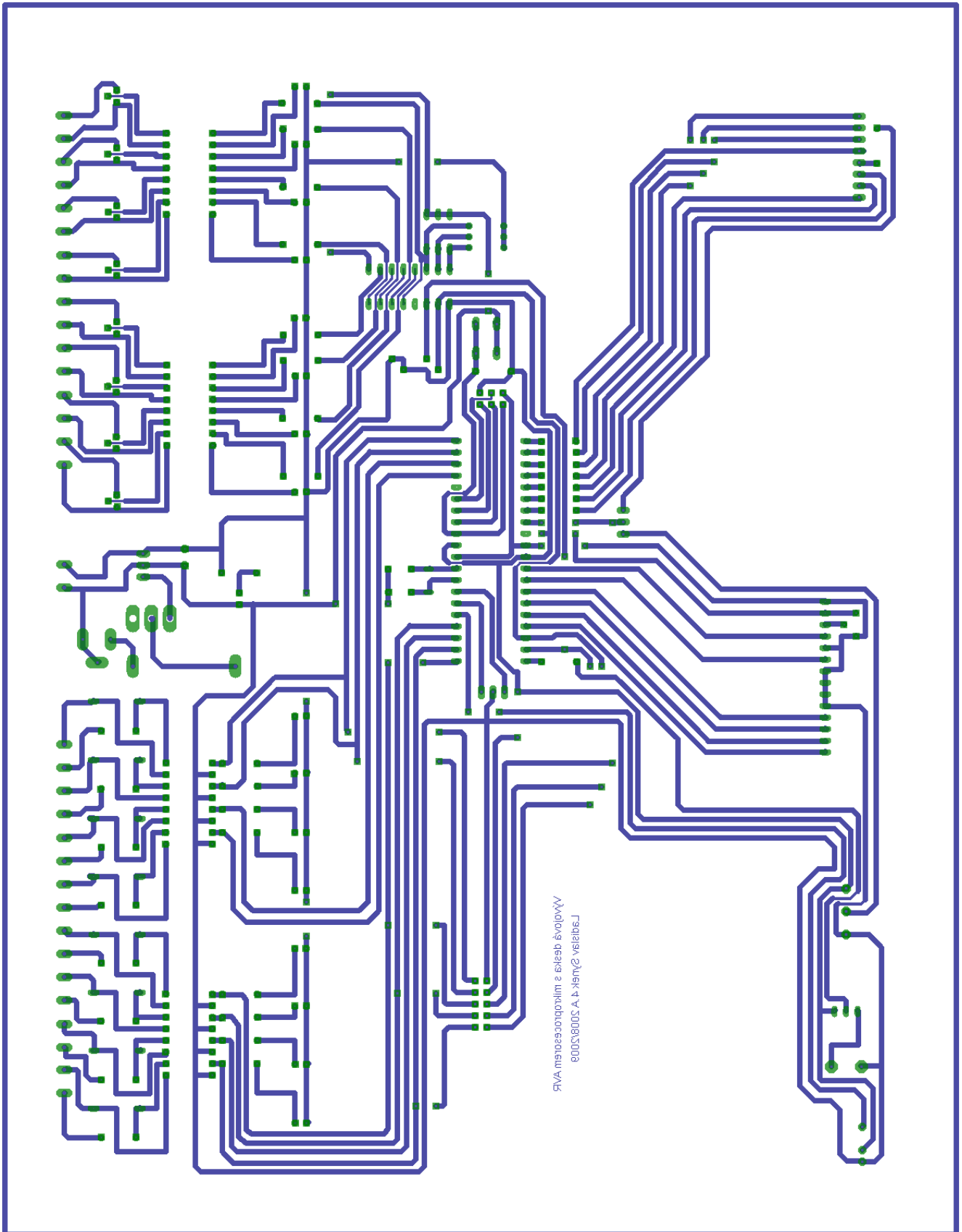
Vývojová deska byla nejprve nakreslena v elektronickém editoru EAGLE a poté byla zhotovena na jednostranném plošném spoji. Plošný spoj by se mohl zhotovit i na oboustranném plošném spoji, v tomto případě by odpadlo osazování několika propojek, ale na oboustranné spoje nemáme ve škole techniku. Vývojová deska byla zhotovena nažehlovací metodou. Tato metoda spočívá v několika krocích, které zde teď popíši.

Postup výroby DPS pomocí nažehlovací metody:

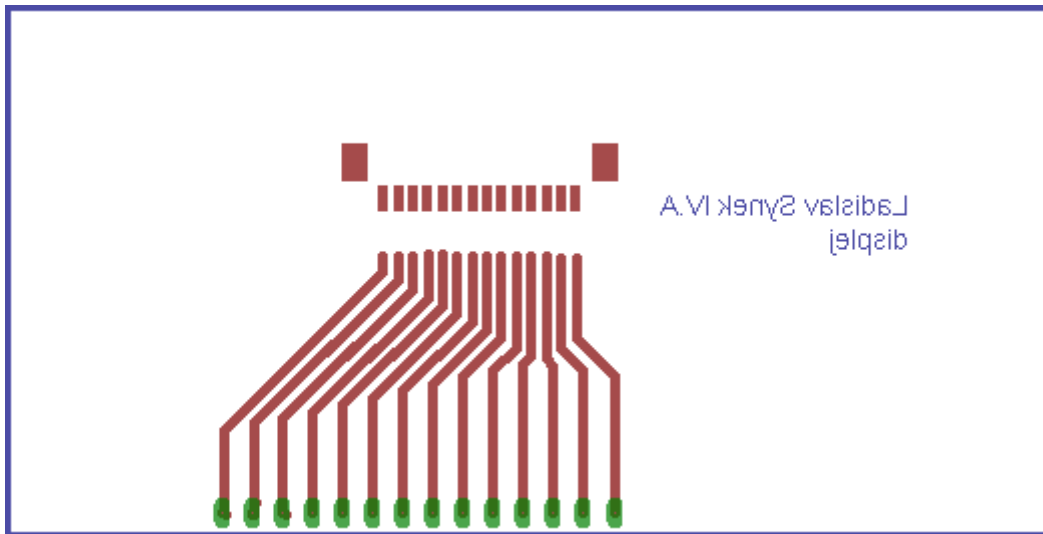
1. Byla vytisknuta předloha DPS na samolepící papír (používají ho děti ve školce),
2. byla vyčištěna a vyleštěna DPS (tak aby se leskla jako zrcadlo), nejlepší je vzít prášek na čištění zašlých mincí a potom celou desku umýt mýdlem,
3. na suchou a vyleštěnou desku byla přiložena předloha, která byla k této desce vhodně přichycena např. samolepícími nálepkami,
4. tato DPS několikrát projela laminovacím zařízením, které zapeklo toner z předlohy na DPS,
5. po zapečení toneru do DPS byla DPS vložena do horké vody, v této vodě zůstala dokud se papír neodlepil od DPS,
6. následovala kontrola DPS a případné drobné úpravy pomocí lihového fixu,
7. nakonec byla DPS vložena do chloridu železitého na vyleptání.

Po vyleptání a uschnutí DPS byla deska natřena šelakem. Tento postup pro vytvoření DPS byl nejen u vývojové desky, ale i u přídavné desky a u desky pro displej. Následně byly desky vyvrtány a osazeny. Po důkladné kontrole osazení bylo započato pájení součástek. K pájení byla použita mikropájka, bezolovnatý cín a kalafuna.

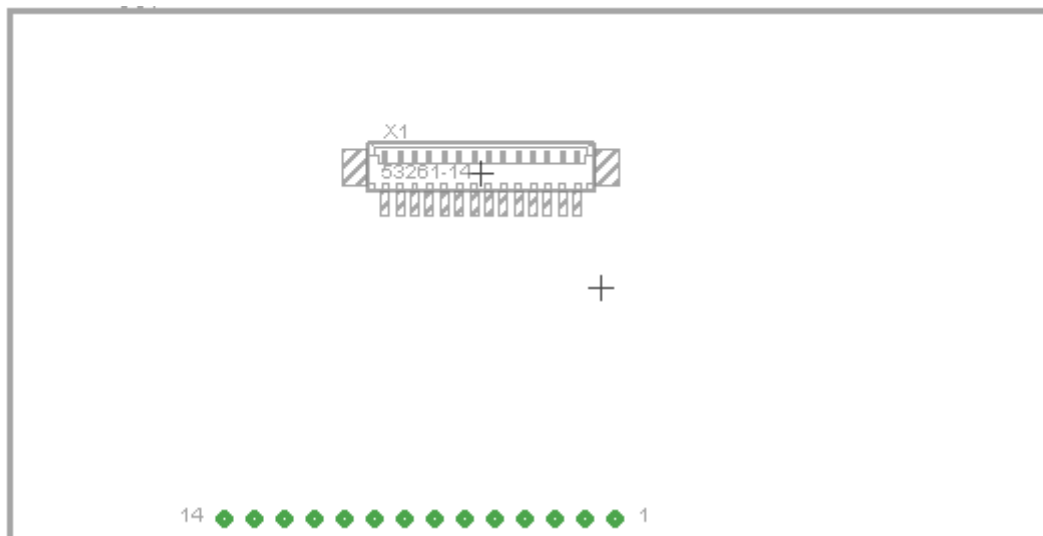
8.1.1. DPS vývojové desky, rozměr 270 x 210, měřítko 1 : 1,2



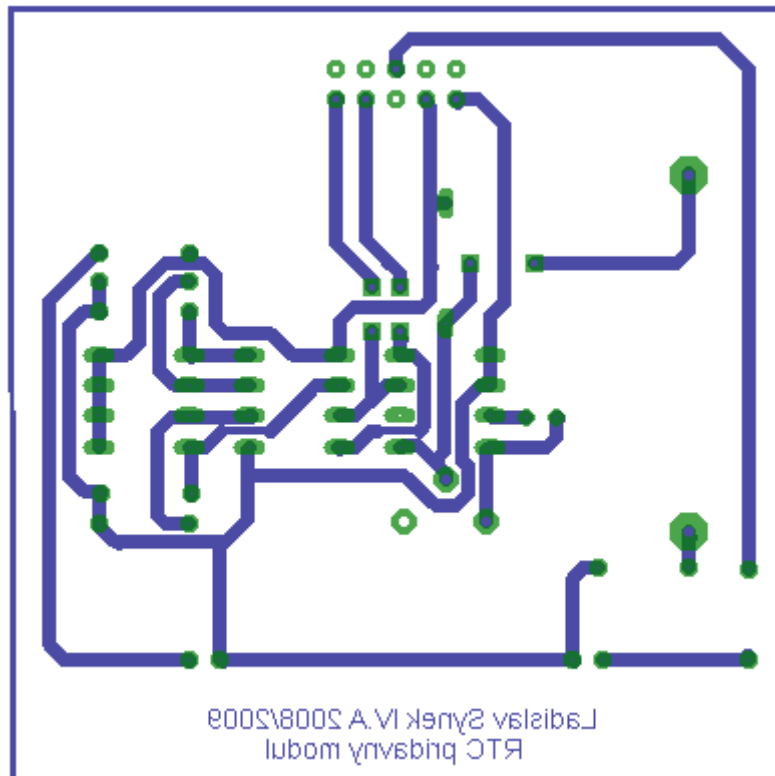
8.1.3. DPS LCD displeje, rozměr 88 x 45, měřítko 1 : 1



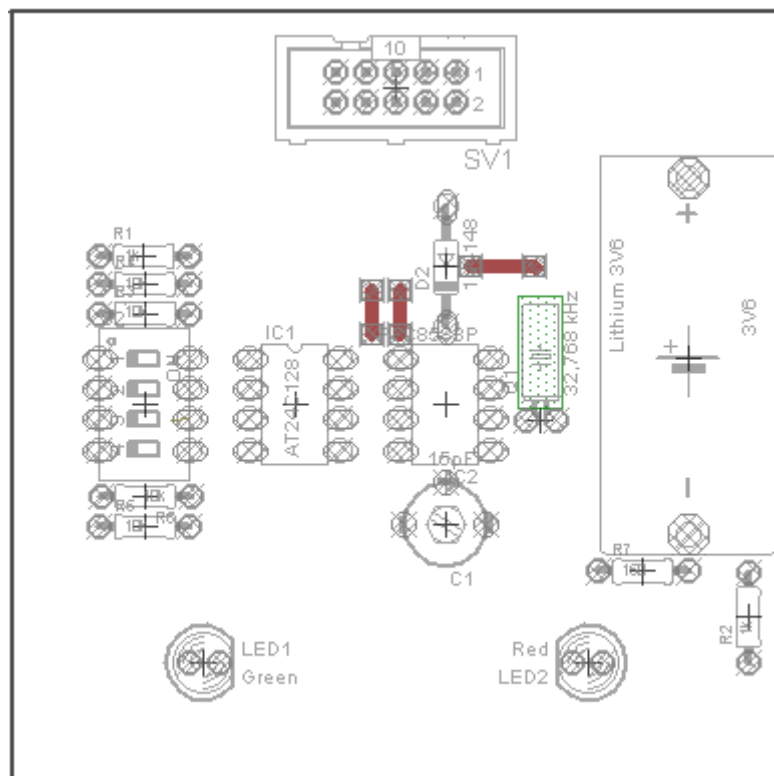
8.1.4. Osazovací výkres LCD displeje, rozměr 88 x 45, měřítko 1 : 1



8.1.5. DPS přídavné desky reálného času, rozměr 65,5 x 65,5, měřítko 1 : 1



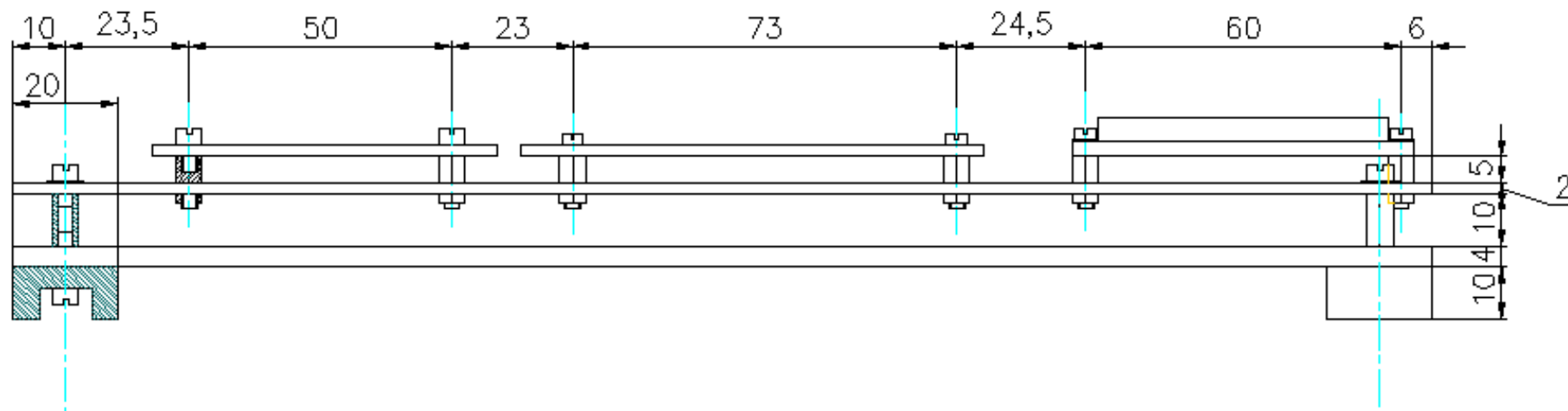
8.1.6. Osazovací výkres přídavné desky reálného času, rozměr 65,5 x 65,5, měřítko 1 : 1



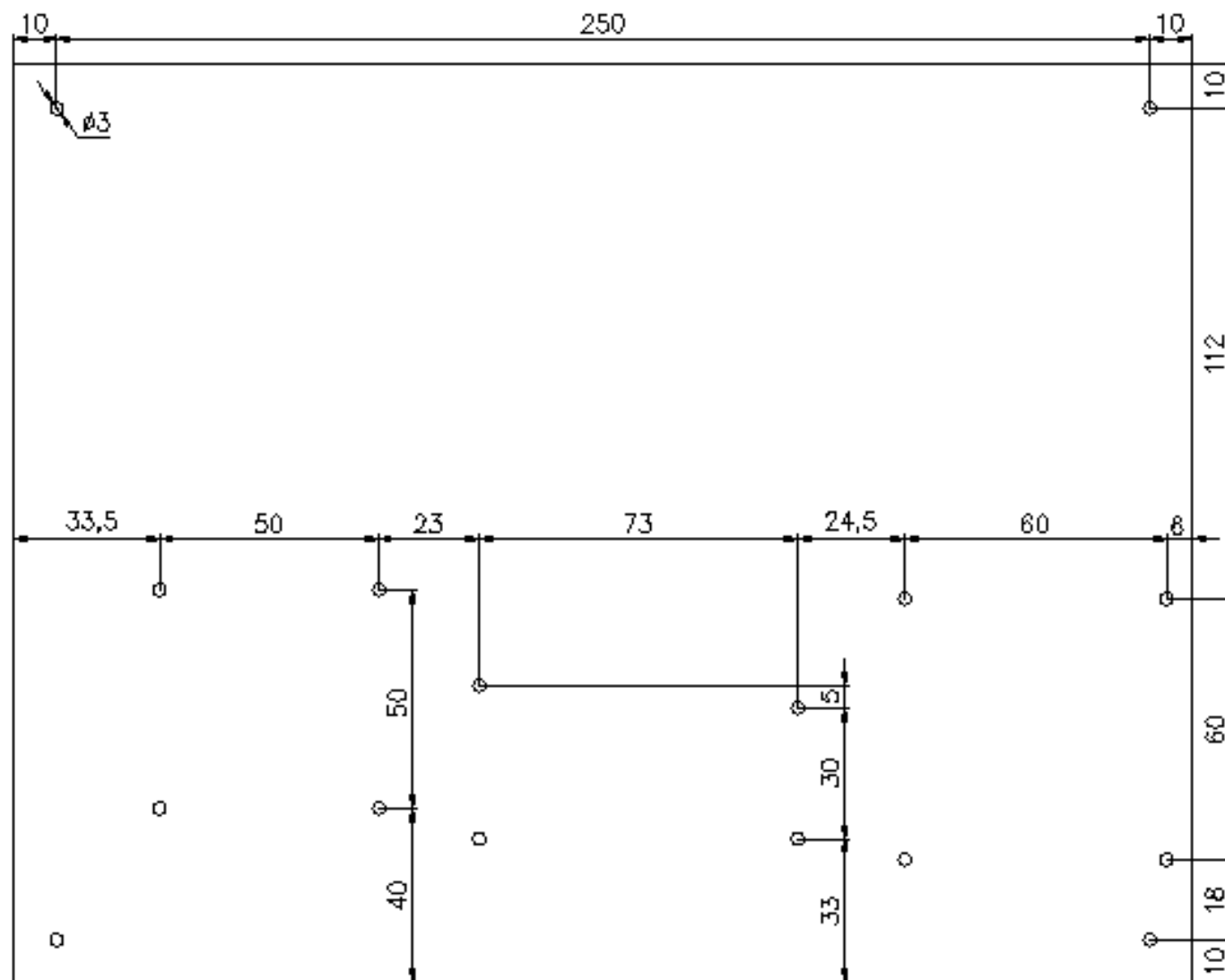
8.2. Popis řešení jednotlivých konstrukčních částí výrobku

Vývojová deska není umístěna v žádné krabičce, aby byl přístup ke všem součástkám. Vývojová deska má umístěn, v každém rohu, distanční sloupek, který tuto desku spojuje s plastovou ochrannou deskou. Plastová deska slouží jako zpevnění samotné vývojové desky, ale také jako ochrana plošných spojů. Na plastové desce jsou umístěny gumové „nožky“, ty slouží na držení desky i na stole s kluzkým povrchem. Přídavné deska, displej a klávesnice jsou taktéž na distančních sloupkách, které však mají menší výšku. Vše je pospojováno šrouby o průměru 3mm. Matice jsou jen na již zmíněných distančních sloupkách pro přídavné desky, displej a klávesnici.

8.2.1. Pohled na vývojovou desku - bokorys



8.2.2. Rozmístění děr na vývojové desce



9. Oživení

9.1. Popis oživení jednotlivých částí výrobku

Vývojovou desku bylo nutné oživit a odzkoušet její funkčnost pomocí programů. Tyto programy byly vymyšleny a naprogramovány v jazyce BASCOM a jsou na CD.

Nejprve byl vysunut z patice mikrokontrolér AtMega32 a do konektoru SPI byl zasunut kabel od programátoru. Na konektor X2 byl připojen zdroj stejnosměrného napětí 9 V. Spínačem S2 bylo zapnuto napájení vývojové desky. Podle schématu byly najity piny od napájení mikrokontroléru +5 V a 0 V. Na těchto pinech bylo změřeno digitálním voltmetrem dané napětí. Napětí bylo měřeno, aby se zjistilo, jestli funguje správně stabilizátor LM7805. Napětí bylo přibližně 5 V. Zdroj stejnosměrného napětí 9 V byl odpojen. Mikrokontrolér byl zasunut zpět do patice. V PC byl nainstalován program AVR Studio na psaní programů a programování mikrokontrolérů Atmel AVR. V tomto programu byl zvolen správný typ programátoru a následně mikrokontroléru a tlačítkem ERASE byl mikrokontrolér vymazán. Tato část oživení výrobku byla nejlehčí.

Jako druhý krok byly odzkoušeny vstupy. Z konektoru X2 byl odpojen zdroj stejnosměrného napájecího napětí. Z patice byl vysunut mikrokontrolér AtMega32. Na vstup IN1 bylo přivedeno napětí +5 V. LED dioda LED9 se rozsvítila. Podle schématu bylo vyhledáno, na který pin mikrokontroléru je připojen tento vstup. Na tomto pinu bylo měřeno napětí voltmetrem proti zemi. Tyto vstupy mají negativní logiku, takže při napětí na vstupu +5 V a vyšší je napětí na pinu 0 V. Při napětí na vstupu +5 V bylo změřeno napětí na pinu patice mikrokontroléru 0 V. Napájení vstupu +5 V bylo odpojeno a bylo měřeno opět napětí na pinu patice mikrokontroléru. To bylo v tomto případě +5 V. Toto měření bylo postupně prováděno na všech vstupech.

Třetím krokem ožívování vývojové desky bylo oživení a odzkoušení výstupů. Do patice byl zasunut mikrokontrolér, do konektoru X2 napájecí zdroj a byla nastavena adresa integrovaného obvodu PCF8574P přepínačem ADRESA. Na tomto místě už nebyla jiná možnost, než vymyslet program, který bude postupně spínat výstupy OUT1 – OUT8. Pomocí Sběrnice I2C byla nejprve posílána tonuto obvodu jeho adresa pro zápis dat a následně mu byl posílán vždy balík 8-mi bitů (1 byte). Tímto balíkem bylo řečeno integrovanému obvodu jaký výstup má mít hodnotu log.1 a jaký hodnotu log. 0. Pracuji však s negativní logikou a s tím se musí počítat i v programu. Např. má - li se má sepnout

výstup OUT1 (při sepnutí se rozsvítí daná LED) bude bitový balík vypadat takto 01111111. Program je udělán takto: v nekonečné smyčce se přičítá číslo od nuly po jedné do 255, potom spadne zase na nulu a začne zase počítat do 255, takto to pokračuje do nekonečna. Mezi jednotlivým přičítáním je vložen zpomalovací cykl, který zajistí, aby toto spínání výstupů bylo registrovatelné. Na výstupu se to projeví jako postupné rozsvěcení a zhasínání diod, které jsou na výstupech na vývojové desce. Toto rozsvěcení je jako binární počítadlo, které je znegováno (dioda svítí log. 0, dioda nesvítí log. 1). Takto byly vyzkoušeny výstupy pro více adres, které byly vždy nastaveny přepínačem ADRESA a tato adresa zapsána pak v programu.

Oživení LCD displeje bylo opět prováděno pomocí programu. Tento program nebyl nijak složitý, protože programovací jazyk BASCOM má v sobě obsaženy knihovny na různé řadiče LCD displejů. Mezi těmito knihovnami je i knihovna na řadič HD44780 nebo ekvivalentní s ním. Použití této knihovny velmi usnadní program a je velmi jednoduchá na použití. Na počátku této knihovny je nutno zadat, jak je zapojený LCD displej k mikrokontroléru a kolik má LCD displej sloupců a kolik má řádků. Potom jednoduše stačí zadat příkaz lcd a za tento příkaz do uvozovek napsat to, co má být zobrazeno na LCD.

Maticová klávesnice 4 X 4 byla oživena pomocí programu s použitím knihovny právě pro tento typ klávesnice. Jedním kritériem této knihovny bylo, aby byla celá klávesnice připojena na jediném portu, který bude kompletně obsazený. Toto kritérium bylo splněno (klávesnice je připojena na PORT A). Jako první krok v programu bylo nutné nastavit na jakém portu byla klávesnice připojena. Dále už jen stačí pro přečtení stavu klávesnice zadat příkaz např. A = Getkbd() a do proměnné A byla zapsána hodnota od 0 do 15. Tato hodnota vystihuje jaká klávesa byla stisknuta. Např. klávesa s nápisem 1 má hodnotu 0, klávesa s nápisem * má hodnotu 3 a klávesa s nápisem 2 má hodnotu 4. Takto postupuje hodnota dále až na klávesu s nápisem D a hodnotou 15. Těmto hodnotám se dá přiřadit v programu samozřejmě i jiná funkce, číslice či písmeno. Na LCD displeji bylo zobrazena nejprve hodnota jednotlivých zmáčknutých kláves. Později, když vše fungovalo jak mělo, se každé klávese přiřadila v programu taková hodnota, jaká byla napsána na jednotlivých klávesách. Tato hodnota právě zmáčknuté klávesy byla zobrazena na LCD displeji.

Potenciometr a oživení A/D převodníku bylo velice jednoduché. JUMPEREM JP1 jsem zapojil napěťový dělič na integrovaný A/D převodník v mikrokontroléru AtMega32. Byl naprogramován program, který zobrazoval na displeji hodnotu A/D převodníku. Tato hodnota se pohybovala v rozmezí 0 – 1023 (10 - ti bitový A/D převodník).

Odzkoušení konektoru pro přídatné desky proběhlo na přídatné desce reálného času. Byl naprogramován program, který z obvodu PCF8583 vyhodnocoval data a ty pak ve formě hodin a datumu posílal na LCD displej.

Konektor SPI se nemusel zkoušet, protože se přes něj celou dobu programovalo.

Konektor USART byl odzkoušen na odesílání dat do PC a na přijímání dat z PC. Opět celé odzkoušení proběhlo pomocí programu, který zajišťoval, že se nejprve budou posílat data do PC dokud se nepošle z PC slovo stop.

K IR přijímači byl vybrán dálkový ovladač s protokolem NEC. Na tento protokol byl naprogramován program, který tento protokol rozeznal a aby z něho dokázal oddělit 8-bitovou adresu vysílače a jeho 8-bitovou instrukci. Při stisknutí tlačítka bylo na displeji zobrazena adresa dálkového ovladače a instrukce tlačítka.

Všechny tyto programy, s kterými byl oživen výrobek, jsou na CD.

10. Rozpiska materiálů a součástek

10.1. Rozpiska elektronických součástek vývojové desky

Pol.	Množství	Název	Typ	Hodnota, pouzdro
1	26	Rezistor R1 - R8, R12 - R27, R36, R39	Miniaturní uhlíkový	1 k Ω
2	8	Rezistor R28 - R35	Miniaturní uhlíkový	470 Ω
3	2	Rezistor R9, R10	Miniaturní uhlíkový	4 k 7
4	4	Rezistor R38, R40, R41, R42	Miniaturní uhlíkový	10 k Ω
5	1	Trimr R11	Miniaturní	50 k Ω
6	1	Potenciometr R37		50 k Ω
7	1	Krystal Q1		8 MHz
7	2	Kondenzátor C1, C2	Keramický	22 pF
8	1	Kondenzátor C3	Elektrolitický	100 μ F
9	8	Dioda D1 - D8	Univerzální	1 N 4148
10	16	LED dioda LED1 - LED16	Žlutá	průměr 5 mm
11	1	LED dioda LED17	Zelená	průměr 5 mm
12	4	Optočlen OK1 - OK16	S bip. tranzistorem	cosmo 4040,
13	8	Tranzistor T1 - T8	Křemíkový, bipolární	BC337, TO 92
14	1	Tranzistor T9	Křemíkový, bipolární	BC546A, TO 92
15	1	Integrovaný obvod IC3	Stabilizátor 1A	LM 7805, TO 220
16	1	Integrovaný obvod IC1	Mikrokontrolér	AtMega32, DIP 40
17	1	Integrovaný obvod IC2	Expandér 8 - bit.	PCF8574P, DIP 16
18	1	Displej DS1	LCD 20 x 2 znaků	
19	1	IR přijímač IR1		TSOP1230
20	1	Reproduktor SP1	Dynamický	

10.2. Rozpiska elektronických součástek přídavné desky reálného času

Pol.	Množství	Název	Typ	Hodnota, pouzdro
1	2	Rezistor R1, R2	Miniaturní uhlíkový	1 k Ω
2	4	Rezistor R3, R4, R5, R6	Miniaturní uhlíkový	10 k Ω
3	1	Rezistor R7	Miniaturní uhlíkový	150 Ω
4	1	Kondenzátorový trimr C1	Otočný	15 pF
5	1	Krystal X1	Miniaturní	32,768 kHz
6	1	Baterie B1	Lithiová	3,6 V
7	1	Dioda D1	Křemíková	1N4148
8	1	konektor SV1		ML 10
9	1	Spínač S1	DIL 4x	DIP04YL
10	1	Integrovaný obvod IC1	paměť EEPROM	AT24C128
11	1	Integrovaný obvod IC2	Real Time Clock	PCF8583P
12	1	LED 1	Zelená LED	
13	1	LED 2	Červená LED	

10.3. Rozpiska elektrická

Položka	Množství	Název dílu	Poznámka
1	1	Vypínač	max. 230 V 3 A
2	17	Svorkovnice	AK 100/2-DS-5.0-V-GREY
3	1	Zdířka 9V	Zdířka na zdroj 9 V
4	1	Pouzdro na pojistku do DPS	
5	1	Patice DIP40	
6	1	Patice DIP16	
7	1	Tlačítko S1	P - B1720A
8	1	Konektor SV1	ML6
9	1	Konektor USART	PFH02 - 03P
10	1	Jumper 3x	
11	1	Maticová klávesnice 4x4	F - KV16KEY

10.4. Rozpiska mechanická

Položka	Množství	Název dílu	Poznámka
1	16	Šroub M3 x 5	
2	2	Šroub M3 x 5	se zapuštěnou hlavou
3	4	Šroub M3 x 10	
4	8	Podložka M3	
5	12	Distanční sloupek M3	M3 x 5
6	4	Distanční sloupek M3	M3 x 10
7	1	Výztuha typu "L"	
8	1	Plastová deska	v. š. h.: 270 x 210 x 5
9	4	Gumové nožky	Ø 20mm

11. Seznam použitých odborných výrazů

Mikrokontrolér – elektronická součástka, která představuje plnohodnotný mikropočítač s rozdělenou pamětí na program a pamětí na data

LCD displej – zkratka pro displej s tekutými krystaly

IR přijímač – infračervený přijímač, používá se např. u dálkové ovládaných spotřebičů pro příjem infračerveného signálu z dálkového ovladače

Adresa – integrované obvody řízené přes I2C mají definovanou určitou adresu, když máme na jedné sběrnici několik obvodů, musí mít každý jinou adresu, podle adresy je pak rozlišujeme

Reproduktor – zařízení na přeměnu elektrické energie na akustickou

Potenciometr – elektronická součástka, v našem případě jako napěťový dělič

A/D převodník – převádí analogové hodnoty na digitální, v našem případě převádí napětí na binární číslo od 0 do 1023

Konektor – na propojení s jinými deskami nebo na propojení s jinými komponenty

USART – sériové rozhraní, pomocí něj lze jednoduše propojit mikrokontrolér s PC

SPI – vysokorychlostní sériové rozhraní, používá se na propojení s rychlými integrovanými obvody (A/D převodníky, FLASH paměti, atd.) nebo na programování mikrokontroléru

PWM regulace – regulace pomocí pulsně šířkové modulace

Periférie -

Atmel AVR – architektura mikrokontroléru od výrobce Atmel

Flash paměť – vysokorychlostní mazatelná paměť

RISC architektura – architektura mikrokontrolérů, RISC = redukováná instrukční sada

Instrukce procesoru – příkaz na který mikroprocesor nějak reaguje

Registry procesoru – do registrů se ukládají proměnné, velikosti proměnných a jiné různé informace při vykonávání jisté funkce

DIP 40 – norma pro pouzdra integrovaných obvodů, 40-ti nožičkové pouzdro, použito např. u mikrokontroléru AtMega32

Pracovní frekvence – frekvence na které pracuje např. mikrokontrolér

RC oscilátor – oscilátor tvořený pomocí rezistoru a kondenzátoru

SRAM paměť – paměť

EEPROM paměť – elektricky mazatelná paměť (pomalejší než FLASH)

JTAG rozhraní – rozhraní mikrokontroléru na odladění a programování mikrokontroléru

Čítač/časovač – část mikrokontroléru

I2C – sériové rozhraní na komunikaci mezi integrovanými obvody

Watchdog Timer – tento systém je zabudován v mikrokontroléru a má za úkol, buď po uplynutí nějaké doby nebo při „zaseknutí“ programu, resetovat mikrokontrolér

Analogový – znamená, že daný signál nemá pevně dané meze (např. log.1 nebo log.0), ale je složen povětšinou ze spojitého signálu (např. výstup na sluchátka u rádia)

Komparátor – elektronická součástka na porovnání dvou napětí

Alfanumerický – znamená, že displej umí zobrazovat znaky a číslice

Řadič HD44780 – integrovaný obvod, který ovládá displej použitý na vývojové desce

Paměť ROM – paměť, do které nelze zapisovat, má v sobě nahrány např. znaky pro LCD displej

Kontrast – lze nastavit u LCD displeje, jak moc chceme tučné znaky

Dělič napětí – na vstupu máme např. 5 V a na výstupu budeme mít 2,5 V, je složen z rezistorů

JUMPER - propojka

Expandér – z malého počtu výstupů udělá více výstupů, ale za cenu snížení rychlosti přenosu
SDA – vodič u I2C sběrnice, po kterém se posílá adresa a data
SCL – vodič u I2C sběrnice, po kterém se posílá pracovní frekvence
TX – vodič u USARTu, slouží na posílání dat
RX – vodič u USARTu, slouží na příjem dat
LED dioda – svítící dioda
Optočlen – slouží na galvanické oddělení, skládá se většinou z diody a z nějaké optoelektrické součástky (fototranzistor, fototyristor, hradlo, atd.)
Synchronizace -
Data – binární kód, který tvoří např. nějaký údaj
Binární – bi znamená dva, takže určitý kód bude složen z jedniček a nul
Sběrnice – dva a více vodičů na které je připojeno několik zařízení (I2C, SPI, atd.)
Tranzistor – elektronická součástka, základ každého elektronického obvodu
Hexadecimálně – kód nebo číslo bude v šestnáctkové soustavě
Rezistor – elektronická součástka
Programátor – zařízení, s kterým se programují mikrokontroléry
PC – osobní počítač
USB – vysokorychlostní sériové rozhraní na PC
COM – pomalé sériové rozhraní na starších PC
PAM – pulsně amplitudová modulace, mění se velikost (amplituda) signálu
PCM – pulsně kódová modulace
TTL – signál kde +5 V je log. 1 a 0 V je log. 0
PIN fotodioda – citlivá fotodioda s intrinzickým (čistým) polovodičem
PŠM – pulsně šířková modulace
DPS – deska plošných spojů
Port – např. u mikrokontroléru, jeden port u 8-bitového mikrokontroléru má 8 vstupů/výstupů
PLL – fázový závěs, ladění rádia po určitých krocích

12. Seznam použitých obrázků

Název obrázku	Číslo stránky
Obr. 1 Blokové schéma vývojové desky	9
Obr. 2 Konfigurace jednotlivých pinů mikrokontroléru	11
Obr. 3 Zapojení LCD displeje	12
Obr. 4 Znaková sada A00	13
Obr. 5 Komunikace s LCD displejem – 8-mi bitový přenos	13
Obr. 6 Komunikace s LCD displejem – 4 bitový přenos	14
Obr. 7 Vnitřní zapojení maticové klávesnice	15
Obr. 8 Zapojení potenciometru, přepínače JP1 a konektoru klávesnice	16
Obr. 9 Zapojení jednoho výstupu s expandérem PCF8574P	17
Obr. 10 Přenos po vodičích SCL a SCK po expandéru PCF8574P	18
Obr. 11 Zapojení jednoho vstupu	20
Obr. 12 Zapojení konektoru pro přídavné desky	21
Obr. 13 Zapojení konektoru SPI	22
Obr. 14 Koncepce systému se sběrníci SPI	23
Obr. 15 Časové průběhy na jednotlivých vodičích rozhraní SPI	23
Obr. 16 Zapojení USARTu	24
Obr. 17 Asynchronní přenos dat	24
Obr. 18 Blokové schéma IR přijímače TSOP1230	25
Obr. 19 NEC protokol	25
Obr. 20 Zapojení přídavné desky reálného času	26

13. Seznam příloh

Název a číslo přílohy
Př. 1 Připojení periférií k mikrokontroléru AtMega32
Př. 2.1 Celkové schéma vývojové desky - Vstupy
Př. 2.2 Celkové schéma vývojové desky - Hlavní část
Př. 2.3 Celkové schéma vývojové desky - Výstupy
Př. 3 Rozmístění jednotlivých periférií na vývojové desce
Př. 4.1 Pohled na vývojovou desku
Př. 4.2 Pohled na pracoviště
Př. 4.3 Programátor s propojovacími kabely
Př. 4.4 Přídavná deska reálného času

13. Použité materiály

Literatura:

- [1] Váňa, V.: Mikrokontroléry ATMEL AVR programování v jazyce BASCOM. 1. vydání Praha, BEN - technická literatura 2004. 144 s.
- [2] Váňa, V.: Mikrokontroléry ATMEL AVR – popis procesoru a instrukční soubor. 1. vydání Praha, BEN – technická literatura 2003. 327 s.
- [3] Váňa, V.: Mikrokontroléry ATMEL AVR – programování v jazyce C. 1. vydání Praha, BEN – technická literatura 2003. 205 s.
- [4] Ing. Frýza T., Ph.D. – Ing. Fedra Z., Ph.D. – Ing. Šebesta J., Ph.D.: MIKROPROCESOROVÁ TECHNIKA Laboratorní cvičení. 1.vydání Brno, Vysoké učení technické v Brně. Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky 2008. 50 s.

Internet:

- [1] <http://www.mcu.cz>
- [2] <http://www.avrfreaks.net>
- [3] <http://www.mcselec.com>
- [4] <http://www.atmel.com>
- [5] <http://www.mcu.cz>
- [6] <http://www.alldatasheet.com>
- [7] <http://www.datasheetcatalog.com>
- [8] <http://www.forum.mcontrollers.com>
- [9] <http://www.hw.cz>
- [10] <http://www.gme.cz>
- [11] <http://www.fichl.de>

14. Soupis souborů na CD

Název složky	Název souboru	Popis obsahu souboru
Dokumentace	Dokumentace.pdf	Dokumentace maturitní práce
Prezentace	Prezentace.ppt	Prezentace maturitní práce
Programy	různé	Programy pro vývojovou desku, napsané a odzkoušené
Datasheets	různé	Datasheety použitých obvodů

15. Závěr

Závěrem bych chtěl říci, že já osobně jsem s touto vývojovou deskou velice spokojen. Je to má zatím nejlepší práce a jsem na ni docela hrdý. Dále bych chtěl říci, že snad se moje práce bude někomu hodit a nedělal jsem ji tudíž zcela zbytečně. Tato vývojová deska je velice univerzální díky přídavným deskám a tudíž použití a rozšíření této desky se meze nekladou, spíše naopak. Na vývojové desce jsem strávil mnoho času a z velké části to byl můj soukromý čas, s kterým bych mohl naložit úplně jinak, ale já jsem se chtěl dozvědět něco nového a samozřejmě nabrat nové zkušenosti s mikrokontroléry.

Funkčnost celé desky je odzkoušena a mohu zcela v klidu říci, že funguje perfektně. Vylepšení desky do budoucna samozřejmě plánuji, ale zatím na to není čas, ani finance. Vylepšení by spočívalo: místo programovacího rozhraní SPI bych zvolil rozhraní JTAG, vyměnil bych displej za displej s podsvětlením, které by se spínalo pomocí mikrokontroléru, dalším spíš rozšířením než vylepšením by bylo navrhnout nějaké další přídavné desky např. s PWM regulací otáček, FLASH paměť, rádio s PLL, atd.

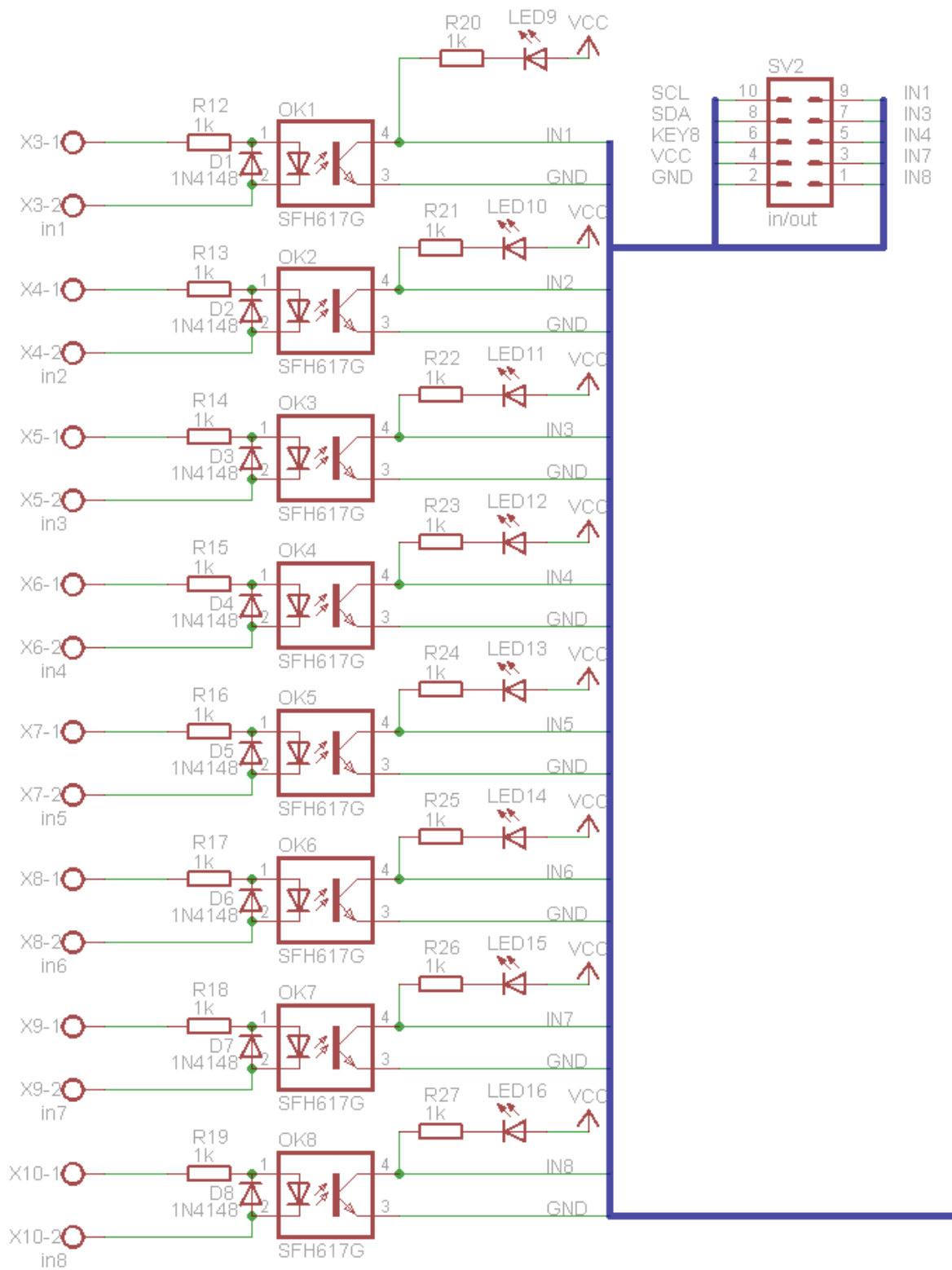
Všechny výpočty byly prováděny pomocí ohmova zákona a tudíž je zbytečné je zde uvádět. Jednalo se převážně o výpočty předřadných rezistorů k LED diodám a k optočlenům.

Součástky na vývojovou desku se dají sehnat velice levně. Ze začátku je programování malinko těžké, ale s touto vývojovou deskou je to zábava a hlavně základy se naučí takřka hned.

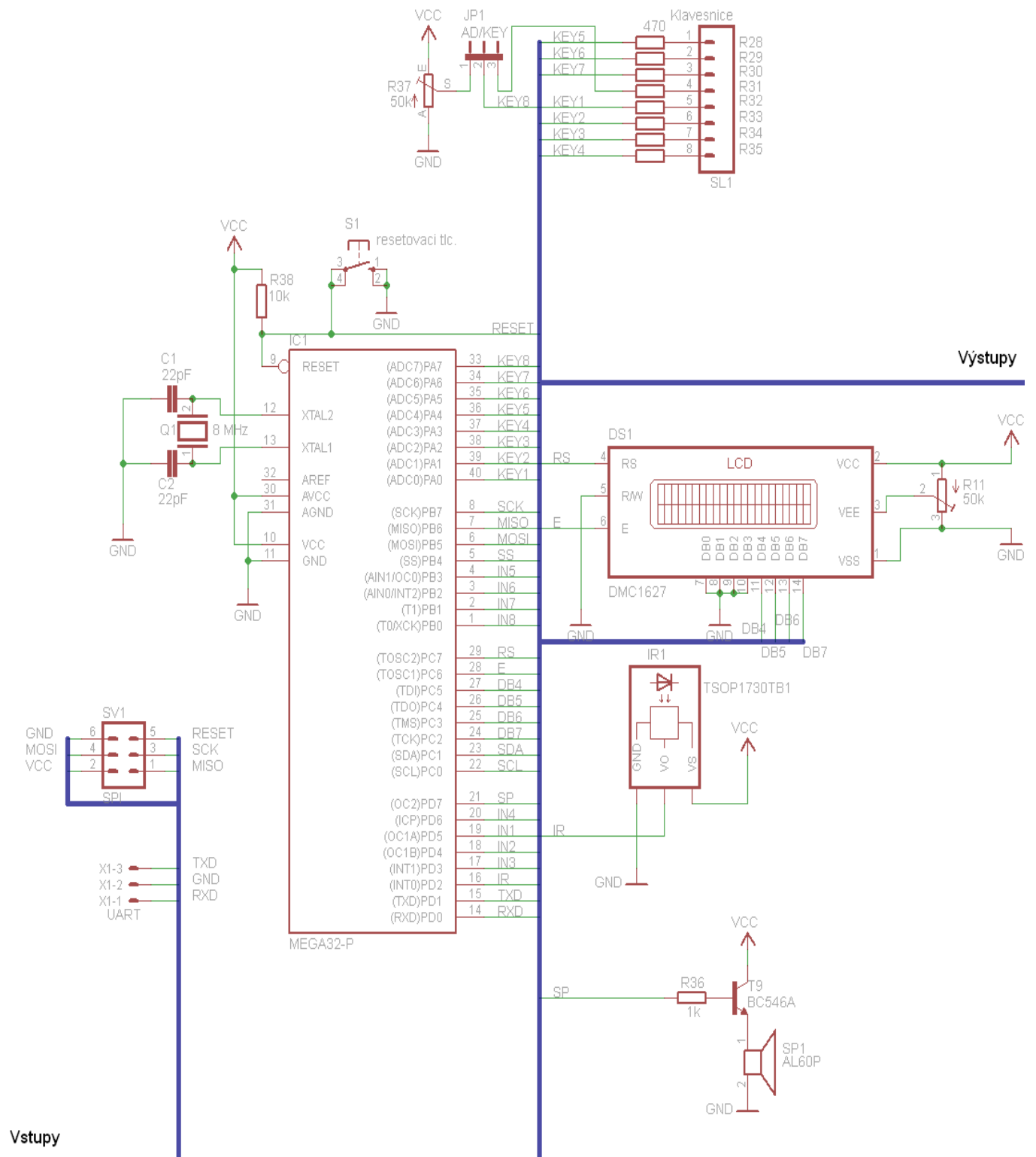
Př.1 Připojení periférií k mikrokontroléru AtMega32

Připojení periférií k mikroprocesoru AtMega32		
periferní součástka	vstupy a výstupy periferní součástky	vstupy a výstupy procesoru [pin / port]
LCD displej	DB4	27 / PC5
	DB5	26 / PC4
	DB6	25 / PC3
	DB7	24 / PC2
	RS	29 / PC7
	E	28 / PC6
Maticová klávesnice 4x4	1-8	33-40 / PA0 - PA7
Potenciometr 50k		33 / PA7 (ADC7)
Expander PCF8574P (výstupy)	SDA	23 / SDA
	SCL	22 / SCL
Vstupy z optočlenů	IN1	19 / PD5
	IN2	18 / PD4
	IN3	17 / PD3 (INT1)
	IN4	20 / PD6
	IN5	4 / PB3 (AIN1)
	IN6	3 / PB2 (AIN0)
	IN7	2 / PB1 (T1)
	IN8	1 / PB0 (T0)
Konektor pro přídavné desky	1	1 / PB0 (T0)
	2	GND
	3	2 / PB1 (T1)
	4	VCC
	5	20 / PD6
	6	33 / PA7 (ADC7)
	7	17 / PD3 (INT1)
	8	23 / SDA
	9	19 / PD5
	10	22 / SCL
Konektor SPI	1	MISO
	2	VCC
	3	SCK
	4	MOSI
	5	RESET
	6	GND
USART	1	RXD
	2	GND
	3	TXD
IR přijímač	IR	16 / PD2 (INT0)
Reproduktor	SP	21 / PD7

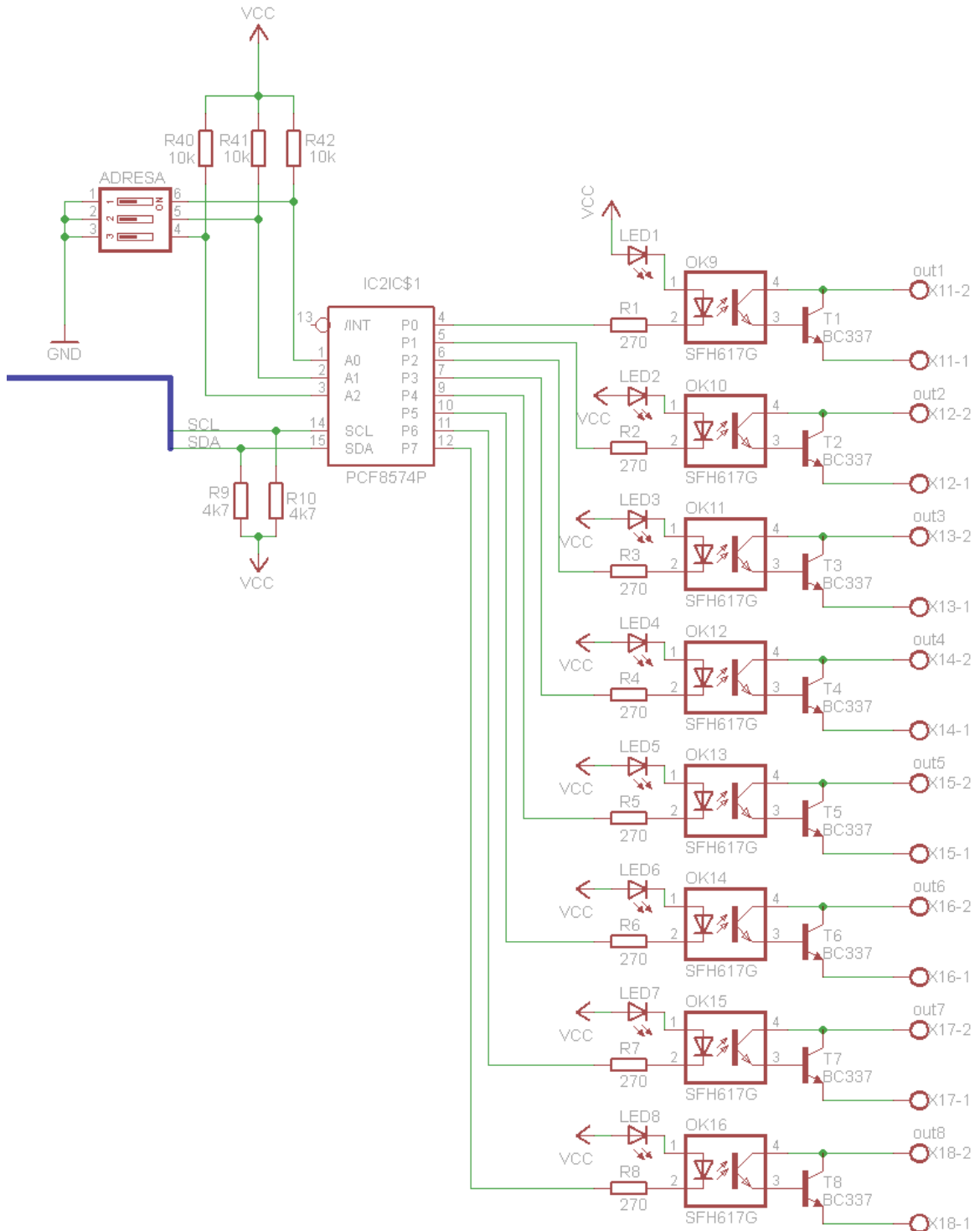
Př. 2.1 Celkové schéma vývojové desky



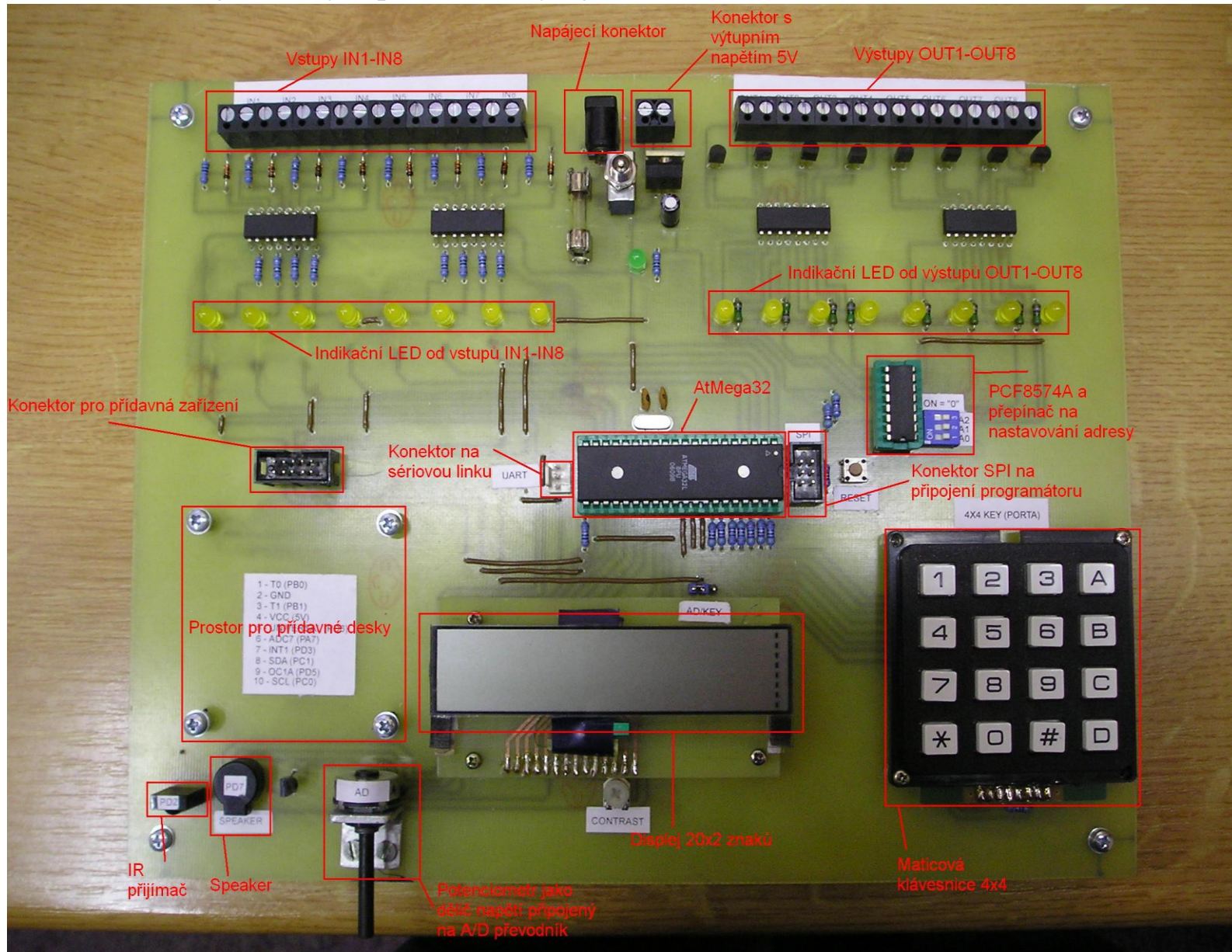
Př. 2.2 Celkové schéma vývojové desky



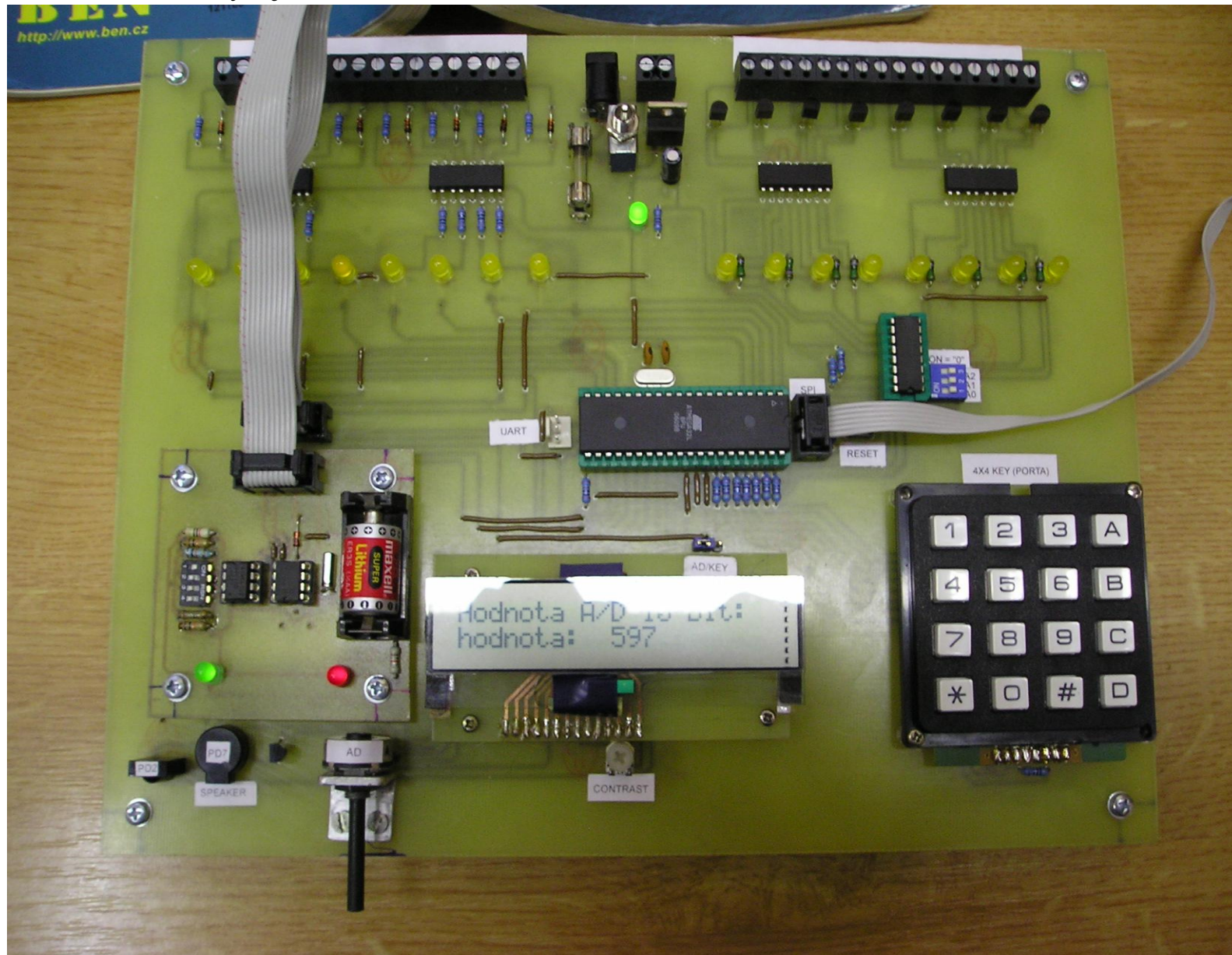
Př. 2.3 Celkové schéma vývojové desky



Př. 3 Rozmístění jednotlivých periférií na vývojové desce



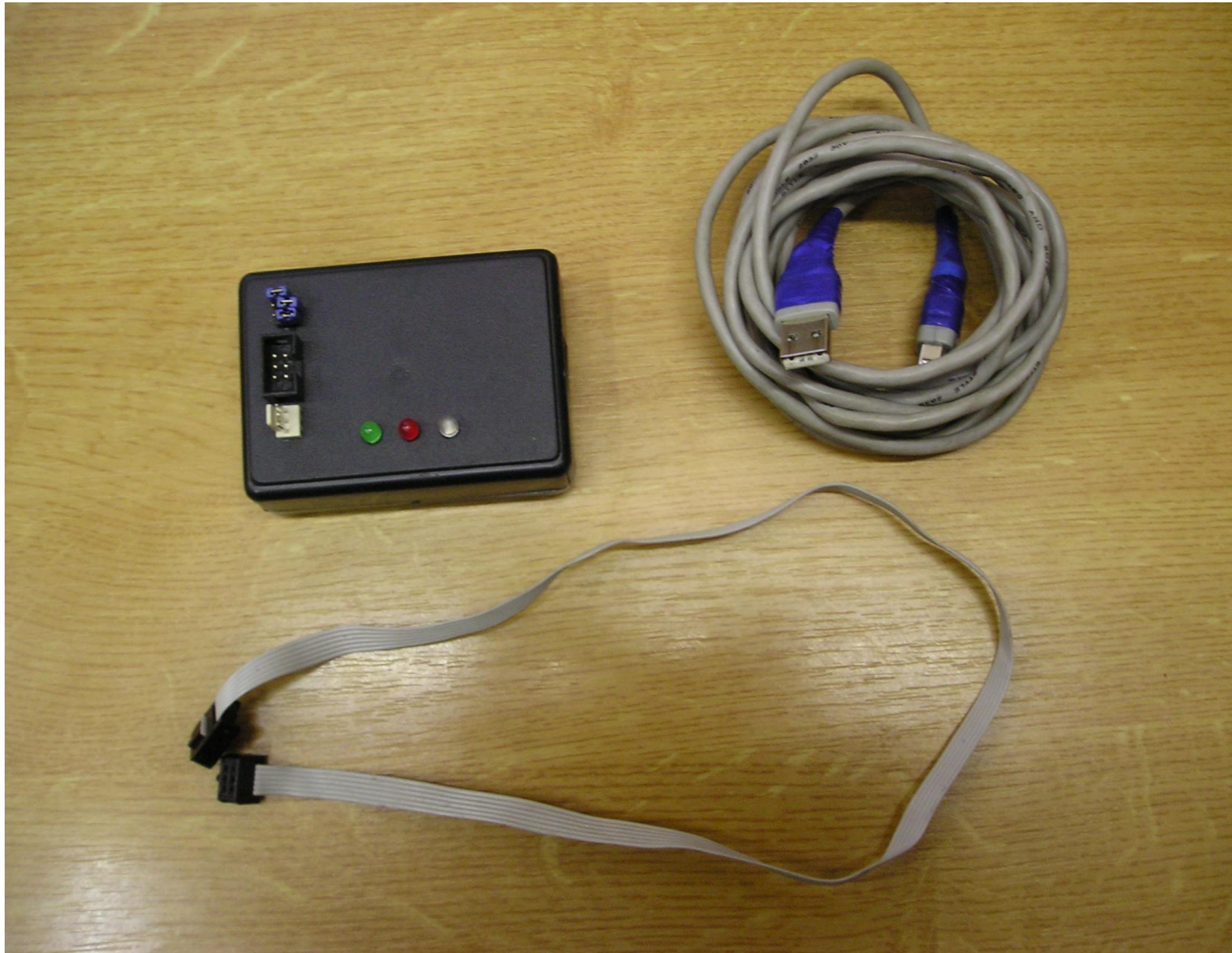
Př. 4.1 Pohled na vývojovou desku



Př. 4.2 Pohled na pracoviště



Př. 4.3 Programátor s propojovacími kabely



Př. 4.4 Přídavná deska reálného času

