

ooJihočeská univerzita v Českých Budějovicích

Pedagogická fakulta
Katedra fyziky

**Použití mikrokontroléru pro realizaci různých
sériových rozhraní**

Bakalářská práce

Vedoucí práce: **Ing. Michal Šerý**

autor: **Miroslav Langmajer**

ČESKÉ BUDĚJOVICE 2008

ABSTRAKT

Tato práce se zabývá návrhem a realizací mikrokontroleru komunikujícím s PC a různými sériovými sběrnici. Mikrokontroler musí být naprogramován tak, aby bylo možné připojit jakékoli množství zařízení komunikující po příslušné sběrnici. Komunikace mezi nadřazeným PC a mikrokontrolerem je řešena pomocí portu RS-232. Část bakalářské práce seznamuje čtenáře s popisem mikroprocesorem rodiny 8051, sběrnici I2C a 1-wire. Druhá část práce řeší návrh hardware a způsob připojení PC a sběrnici. Další částí je naprogramování mikroprocesoru ke komunikaci a popis řídicího software Terminal 1.9b a jeho nasazení do komunikace se sběrnici.

ABSTRACT

This task deal with suggestion and realization of microcontroller communicatin with PC and serial bus. Microcontroller must be set to be able to add number of device communicating on relevate bus. Communication between PC and microcontroller is realized by port RS-232. The part of task describe microprocessor part 8051, buses I2C, and 1-wire. The other part of task is programming of microprocessor to communicate and discribe software Terminal 1,9b and his communication with buses.

Mé velké poděkování si zaslouží **Ing. Michal Šerý** za jeho cenné a odborné rady, organizační vedení a hlavně za čas, který se mnou strávil při zpracování této práce. Také děkuji rodině a přátelům za podporu a zázemí, které mi poskytovali po celou dobu studia.

Prohlašuji, že svoji bakalářskou práci jsem vypracoval samostatně pouze s použitím pramenů a literatury uvedených v seznamu citované literatury.

Prohlašuji, že v souladu s § 47b zákona č. 111/1998 Sb. v platném znění souhlasím se zveřejněním své bakalářské práce, a to v nezkrácené podobě elektronickou cestou ve veřejně přístupné části databáze STAG provozované Jihočeskou univerzitou v Českých Budějovicích na jejích internetových stránkách.

V Českých Budějovicích 24. dubna 2008

Miroslav Langmajer

OBSAH

1	ÚVOD	8
2	MIKROPROCESOR ŘADY 8051	9
2.1	JÁDRO PROCESORU	9
2.1.1	Bitový procesor	11
2.1.2	Organizace paměti	11
2.2	REGISTRY SPECIÁLNÍCH FUNKCÍ	13
2.2.1	A - střadač	13
2.2.2	Registr B	14
2.2.3	DPTR - Data pointer, ukazatel	14
2.2.4	PSW - Stavové slovo mikroprocesoru	14
2.2.5	RS1, RS0 - volba banky registrů R0 až R7	14
2.2.6	C - Přenos (Carry)	15
2.2.7	AC - Vnitřní přenos (Auxiliary carry)	15
2.2.8	F0 - Uživatelský příznak	15
2.2.9	OV - Příznak přetečení (Overflow)	15
2.2.10	SP - Ukazatel zásobníku (Stack pointer)	15
2.2.11	Další registry	16
2.3	PŘERUŠOVACÍ SYSTÉM	16
2.4	ČÍTAČE, ČASOVAČE	16
2.5	VSTUPNĚ VÝSTUPNÍ BRÁNY	17
2.6	SÉRIOVÝ KANÁL	17
2.6.1	Pracovní módy sériového kanálu	18
3	SBĚRNICE I2C	20
3.1	PROTOKOL SBĚRNICE I2C	20
3.1.1	SDA a SCL linky	21
3.1.2	Platnost dat	22
3.1.3	Start a stop podmínky	22
3.1.4	8bitový formát	23
3.1.5	Potvrzovací znak (ACK)	23
3.1.6	Adresace zařízení a R/W bit	24
3.1.7	Rezervované adresy	25
3.1.8	Prodloužení signálu CLK	25
3.1.9	Softwarový reset sběrnice	25
3.1.10	Rychlosti sběrnice	26
4	SBĚRNICE 1-WIRE	27
4.1	PROTOKOL SBĚRNICE 1-WIRE	27
4.1.1	DATA linka	28
4.1.2	Reset puls	28
4.1.3	Vysílání dat	29
4.1.4	Zápis logické 0 a logické 1	29
4.1.5	Čtení logické 0 a logické 1	30

4.1.6	Adresace slave zařízení	31
4.1.7	Příkazy 1-wire protokolu	31
5	KOMUNIKACE S PC.....	32
6	POUŽITÉ OBVODY NÁVRHU	34
6.1	MIKROPROCESOR Z RODINY 8051, AT89C2051	34
6.2	MAX 232.....	35
7	NÁVRH ŘEŠENÍ	36
7.1	HARDWAROVÁ ČÁST	37
7.1.1	Schéma zapojení.....	39
7.1.2	Deska plošného spoje	39
7.1.3	Rozložení součástek	40
7.1.4	Seznam součástek.....	41
7.2	SOFTWAREOVÁ ČÁST.....	41
7.2.1	Program mikroprocesoru	41
7.2.2	Program řízení Terminál 1.9b	44
8	ZÁVĚR	48
9	SEZNAM POUŽITÉ LITERATURY	49
10	SEZNAM OBRÁZKŮ.....	50
11	OBSAH PŘILOŽENÉHO CD	51

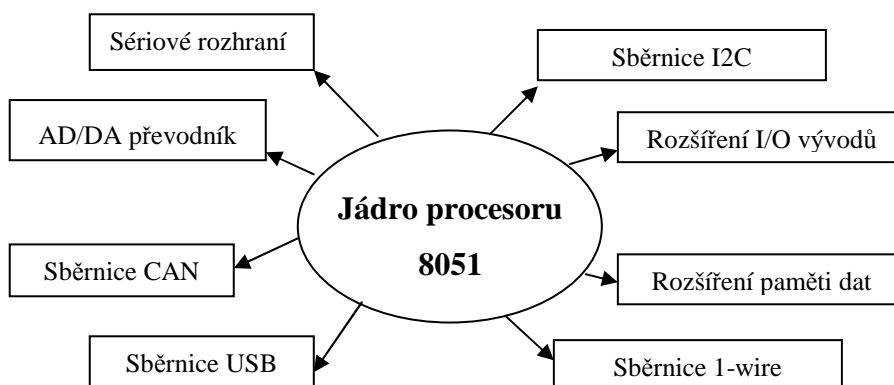
1 ÚVOD

Sériová komunikační rozhraní se dnes používají z důvodů minimalizace datových vodičů a to až na minimálně jeden vodič. Při použití sériových komunikací se zredukuje množství datových vodičů. To umožňuje zmenšit rozměry, protože odpadá prostorově náročné propojování velkým počtem vodičů. Ke komunikaci mezi jednotlivými moduly se navrhuje specializovaná rozhraní resp. Sběrnice (SPI, IR, I2C, 1-wire, Microwire). Tyto sběrnice slouží ke komunikaci mezi integrovanými obvody, případně mezi mikropočítačovými moduly. Pokud komunikujeme pouze po jednom až dvou vodičích sériovou komunikací, dosáhneme mnohem vyšších přenosových rychlostí, neboť dokonale od sebe izolované dva vodiče nemají prakticky omezení přenosových rychlostí, což je například u 16ti vodičů poměrný konstrukční problém. Sériová rozhraní budou předmětem následujícího popisu.

Další výhodou může být v některých případech připojení mikrokontroléru a to i bez integrované příslušné vnitřní sběrnice, v určitém případě i bez příslušného řadiče sériového rozhraní. Funkce je potom realizována programově s využitím několika vývodů vhodných portů mikrokontroléru. Nelze tak většinou dosáhnout plné rychlosti daného rozhraní. Touto problematikou se zabývá tato práce.

2 MIKROPROCESOR ŘADY 8051

[1, 2] Mikroprocesor s jádrem 8051 přišel na svět kolem roku 1980 firmou Intel. Jedná se sice o relativně zastaralý model procesoru, ale přesto je stále oblíben u vývojářů či slouží k výuce na středních či vysokých školách technického zaměření a řada výrobců procesor stále rozvíjí a rozšiřují o další periferie. Těmito periferiemi jsou například: A/D převodník, rozšířené vstupně/ výstupní vývody, paměť programu či rozšířenou vnitřní paměť RAM na 256b a jiné jak je možné vidět na obr. 1.



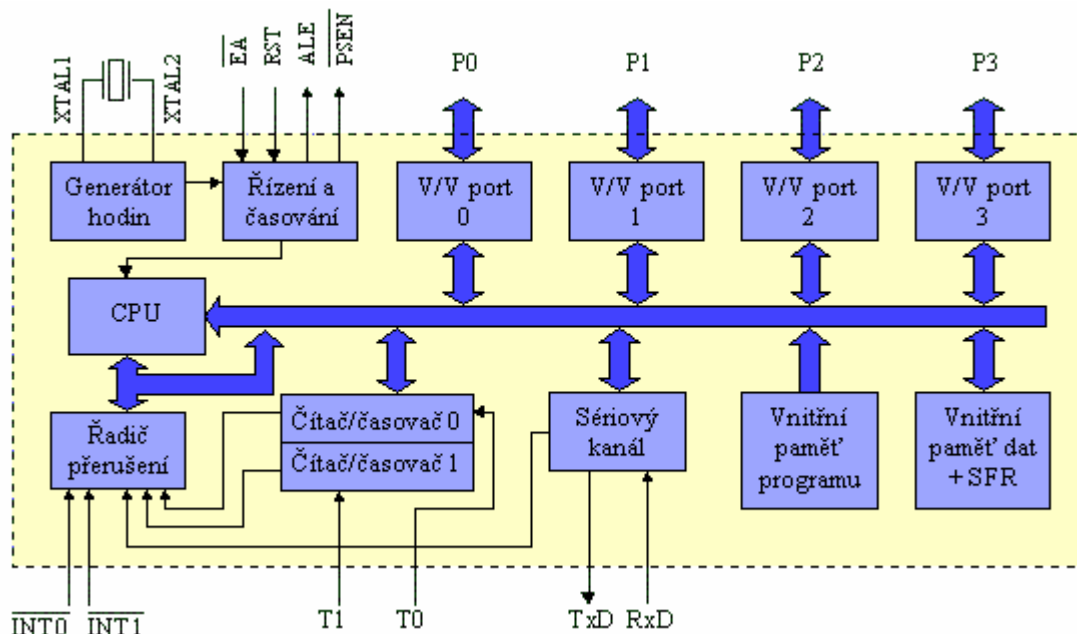
Obr. 1 Jádro procesoru a rozšíření o různé periferie

2.1 Jádro procesoru

Mikroprocesor 8051 je 8-bitový jednočipový mikroprocesor s harwardskou strukturou, u které je oddělena paměť dat a programu. Mikroprocesor tvoří centrální procesorová jednotka (CPU), jejíž podstatnou částí je aritmeticko-logická jednotka umožňující práci s jednotlivými bity paměti, vykonávat instrukce programu atd. Centrální procesorová jednotka je vnitřní 8-bitovou společnou sběrnici propojena s pamětí programu a pamětí dat. Vnitřní paměť programu o velikosti 4kB může být typu ROM (8051), EPROM (8751) nebo mikroprocesor nemusí mít žádnou vnitřní paměť programu (8031). Vnitřní paměť dat je typu RAM o velikosti 128 bytů. Ke společné sběrnici jsou dále připojeny 4 vstupně/výstupní porty P0 až P3, které umožňují styk mikroprocesoru s vnějšími periferiemi.

Chceme-li s mikroprocesorem používat větší paměť, než kterou nám poskytuje sám mikroprocesor, nebo chceme používat mikroprocesor 8031 bez vnitřní paměti programu, můžeme k mikroprocesoru připojit samostatnou vnější paměť programu anebo vnější paměť dat. K tomuto účelu jsou z mikroprocesoru vyvedeny řídicí signály PSEN(paměť programu) a WR, RD(paměť dat). Pro snazší styk s periferiemi je mikroprocesor vybaven řadičem přerušení, který zpracovává 5 zdrojů přerušení - 2 externí (vývod INT0,INT1), od každého ze dvou čítačů/časovačů a od sériového kanálu. Jednotlivá přerušení mají definovanou prioritu na každé ze dvou volitelných úrovní priority. Mikroprocesor obsahuje dva 16-bitové čítače/časovače, s volitelným režimem provozu. Pro snazší sériovou komunikaci s nadřazeným počítačem nebo spolupracujícími mikroprocesory je mikroprocesor vybaven duplexním sériovým kanálem, kterého využíváme k připojení k PC přes sériový port po přizpůsobení napěťových úrovní, které nám vykonává např. obvod MAX 232, MAX202.

Procesor je schopen samostatné činnosti po připojení vnějšího piezokeramického rezonátoru(krystalu) na vývody XTAL1 a XTAL2, nebo v druhém případě vývod XTAL1 uzemníme a na vývod XTAL2 přes hradlo TTL připojíme výstup externího obvodu oscilátoru. U druhého případu je nutné uvést, že u každého typu mikroprocesoru se může způsob připojení signálu z externího oscilátoru lišit. Napájecího stejnosměrného napětí +5V na vývod Ucc a připojení nulovacího, resetovacího, obvodu na vývod RST. Pokud není použita vnější paměť programu, je ještě nutné nastavit vstup EA (negovaný) do log.1. Výjimku tvoří 20ti pinové procesory ATMEL 89C1051, 2051, 4051, které vstup EA (negovaný) nemají.



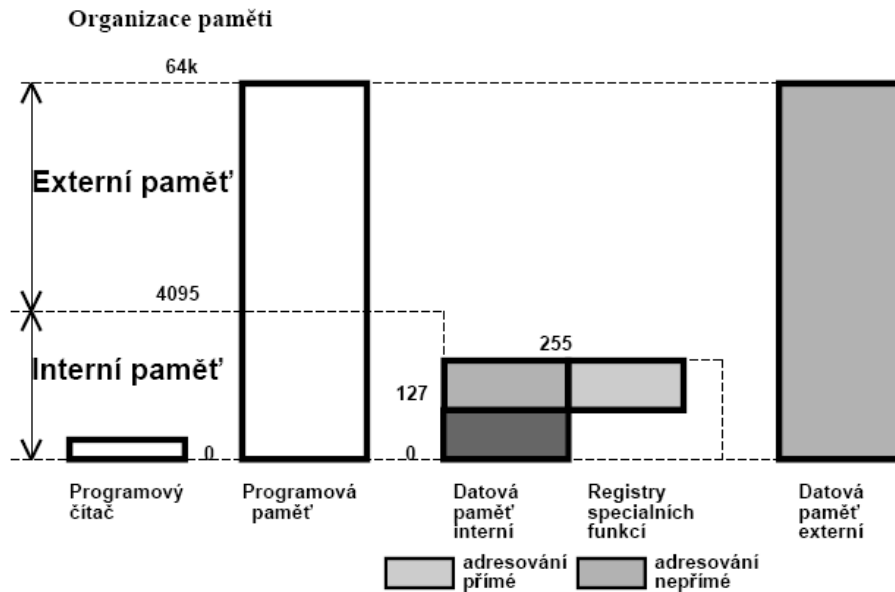
Obr. 2 Bloková struktura mikroprocesoru

2.1.1 Bitový procesor

Bitový někdy označovaný jako booleovský procesor v procesorech 8051 obsahuje vlastní soubor instrukcí a střadač pracující na bitově adresovatelné části paměti RAM, bity adresovatelných speciálních registrů a vstupně/ výstupní registry. Instrukční soubor umožňuje nastavit, nulovat, invertovat bit. Přesun je umožněn pouze s pomocí použití střadače, kterým je procesor vybaven a tím je umožněno realizovat logický součet, součin jednotlivých přímo adresovatelných bitů. Výkon procesoru je obohacen i instrukcemi větvení(např. JB, JNB, JZ), viz. [2].

2.1.2 Organizace paměti

Paměť mikroprocesoru 8051 se skládá ze dvou oddělených částí. Těmito částmi jsou adresovatelné paměti dat a programu, ke kterým se přistupuje pomocí různých instrukcí. Prostory paměti můžeme dělit na paměť vnitřní(velikostí do 4 kB), jenž je umístěna přímo na čipu a paměť vnější(velikostí maximálně 64 kB). Proti překrývání vnitřní a vnější paměti je procesor vybaven vývodem EA, nejčastěji negovaný. Nabývá-li vstup EA hodnoty logická 0, EA=0, je programová paměť tvořena celou vnější pamětí, je-li EA na log. 1, EA=1, pak se čtou programové instrukce v adresovaném prostoru vnitřní paměti ROM nebo EPROM a mimo tento prostor, zbývajících 60 kB je tvořeno pamětí vnější.

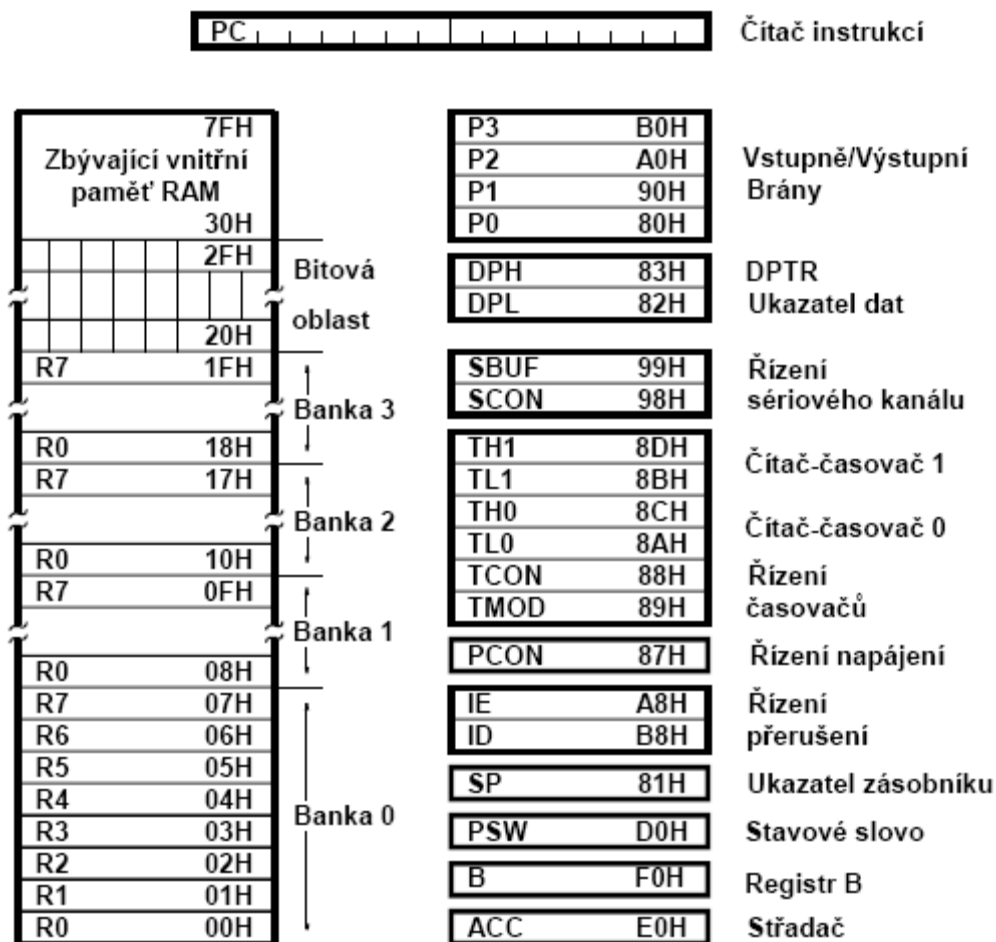


Obr. 3 Organizace vnější a vnitřní paměti

Překrývání jednotlivých adresovaných datových prostorů vnější či vnitřní paměti je řešeno pomocí různých instrukcí. Na obr. 4 je znázornění základního rozdělení vnitřní datové paměti. Vnitřní paměť, RAM, je tvořena 128bity obsahující čtyři banky po osmi registrech (R0 až R7 adresované 00H-1FH). Následuje tzv. bitová oblast (20H-2FH). Jednotlivé bity paměťových míst počínaje nejnižší adresou (20H) a nejnižším bitem (b0 - adresa bitu 00H) jsou vzestupně přímo adresovatelné. Přímě adresovatelných bitů v této oblasti je 128 (poslední bit 127 s adresou (7FH) je bit b7 na adrese 2FH). Další 128 bitů se využívá k adresaci některých významných bitů příslušejících speciálním registrům. Zbývající datová paměť RAM začíná adresou 30H a končí adresou 7FH a je uživateli volně přístupná pro přímé i nepřímé adresování.

Jeli připojena rozšířená paměť procesoru o vnější paměť, přistupuje se k ní přes 16bitový pomocný ukazatel datové paměti, registr DPTR (data pointer). Tato vnější paměť může být přístupná i pro práci s registry R0 a R1 příslušné aktivní banky.

Následující oblast speciálních registrů (SFR) je umístěna v prostoru 80H až FFH a je tvořena 21 registry. Tyto registry jsou přístupné pouze přes přímé adresování, protože může nastat, že stejný adresový prostor bude jak na vnější tak na vnitřní paměti a mohlo by docházet k přepisu.



Obr. 4 Rozložení vnitřní paměti procesoru

2.2 Registry speciálních funkcí

Tyto registry umožňují činnost některých periferních obvodů integrovanými na čipu mikroprocesoru jako jsou např. čítače, sériový kanál a přerušovací systém. Hodnoty nastavení těchto periférií jsou ukládány právě v těchto registrech speciálních funkcí.

2.2.1 A - střadač

Je nejzákladnější registr aritmetickologické operace. Též uváděný jako akumulátor. Obsahuje jeden operand operace do něhož se ukládá její výsledek.

2.2.2 Registr B

Je užíván jako druhý registr pro násobení, nebo dělení. Instrukce MUL AB vynásobí $A \cdot B$ a výsledek násobení dvou osmibitových čísel (zde bez znaménka), může být větší než 255. Proto se výsledek uloží do A nižší byte výsledku a do B vyšší byte. Při dělení DIV AB zase vydělí A/B a výsledek uloží do A a zbytek po dělení do B.

2.2.3 DPTR - Data pointer, ukazatel.

Skládá se ze dvou bajtů. DPH pro vyšších 8 bitů a DPL pro nižších 8 bitů. Využívá se jako ukazatel paměti, obsah této externí paměti se následně ukládá do registru A. Umožňuje přičítání a odčítání paměťové adresy, např. pro výběr v cyklu.

2.2.4 PSW - Stavové slovo mikroprocesoru

Skládá se ze 7 bitů, jejichž umístění je zobrazeno na obrázku a jejich význam je následující:

b7	b6	b5	b4	b3	b2	b1	b0	bit
C	AC	F0	RS1	RS0	OV	--	P	adresa RAM = D0H

Obr. 5 Rozložení bitů v stavovém registru

2.2.5 RS1, RS0 - volba banky registrů R0 až R7

Určují banku, jejíž registry R0 až R7 budou používány obr.4. Jednotlivé bity RS1 a RS0 lze ovládat programově pomocí logických operací. Procesor není vybaven instrukcí pro přepínání bank a po jeho vynulování je aktivní bankou banka 0 ($RS1=RS0=0$). Kombinací těchto dvou registrů dostaneme dekadické číslo od 0 do 3, což nám představuje kombinaci použité banky.

2.2.6 C - Přenos (Carry)

Je nastaven při aritmetické operaci, při které dochází k přenosu z osmého (b7) do devátého (b8) bitu a při některých instrukcích porovnání.

2.2.7 AC - Vnitřní přenos (Auxiliary carry)

Je nastaven dojde-li při sčítání k přenosu mezi čtvrtým (b3) a pátým (b4) bitem střadače. Příznak se využívá pouze v instrukci dekadické korekce DAA, kterou aplikujeme na výsledek součtu dvou dekadických čísel vyjádřených v BCD kódu.

2.2.8 F0 - Uživatelský příznak

F0 může být libovolně využíván (např. přetečení při výpočtu v aritmetice, identifikace vnější události, atd.).

2.2.9 OV - Příznak přetečení (Overflow)

Indikuje přetečení při aritmetické operaci sčítání nebo odečítání, jestliže zpracovávaná čísla považujeme za čísla se znaménkem. Příznak je též využíván při operaci dělení k identifikaci dělení nulou a při instrukci násobení.

2.2.10 SP - Ukazatel zásobníku (Stack pointer)

Je osmibitový a na rozdíl od většiny procesorů je při plnění zásobníku inkrementován. Vlastní zásobník je umístěn ve vnitřní datové paměti RAM a může být umístěn kdekoli v této paměti tj. i v rozšířené části vnitřní datové paměti (adresy 128H až 255H u následovníků 8051), která je přístupná pouze pomocí nepřímého adresování. Po vynulování je nastaven na hodnotu SP=07H a ve většině aplikací musí být pomocí odpovídající instrukce přestaven na jinou hodnotu.

2.2.11 Další registry

P - Příznak parity (Parity flag) Indikuje lichou paritu střadače. Je-li ve střadači lichý počet jedniček, potom příznak parity je nastaven ($P=1$). Příznak je aktualizován po každé instrukci.

PC - Čítač instrukcí (Program counter) je 16-bitový čítač instrukcí, který není přímo programově přístupný.

IE - Interrupt enable - Registr povolení přerušení.

2.3 Přerušovací systém

Přerušovací systém mikroprocesoru umožňuje jakousi komunikaci mezi probíhajícím hlavním programem a určitými hardwarovými částmi mikroprocesoru. Je-li přerušení od dané hardwarové části povoleno, procesor přeruší vykonávání hlavního programu a začne vykonávat program zajišťující obsluhu vzniklého přerušení. Je-li program obsluhy přerušení dokončen, procesor se vrací k vykonávání hlavního programu od místa, ve kterém skončil před přechodem do programu obsluhy přerušení.

2.4 Čítače, časovače

Čítače a časovače tvoří nepostradatelnou součást mikroprocesoru. Funkce čítače/časovače je zde stejná jako u jiných zařízení číslicové techniky. Čítač spravuje určité paměťové místo, ke kterému přičítá jedničku na základě zjištění náběžné nebo sestupné hrany sledovaného signálu. Sledovaný signál je u čítače nějaký vnější signál, např. výstupní signál z fotoelektrického čidla.

Časovač také spravuje určité paměťové místo, ke kterému přičítá jedničku, funguje také jako čítač, ale na rozdíl od čítače je zde sledovaným signálem nějaký vnitřní signál se známým průběhem a konstantní frekvencí. Časovač se používá k vytvoření určitého časového intervalu, nejčastěji ve funkci zpoždění.

Mikroprocesor 8051 má dva 16-bitové čítače/časovače. Lomítko mezi čítačem a časovačem v předchozí větě znamená, že jeden z nich může pracovat jako

čítač nebo časovač, nebo mohou oba pracovat jako čítače nebo oba jako časovače. Oba čítače/ časovače pracují jako vzestupné. Paměťovým místem je zde registr TH0, TL0 pro čítač/ časovač0 a TH1, TL1 pro čítač/ časovač1. TH představuje vyšší byte, TL nižší byte z celkového 16-bitového obsahu čítače/ časovače.

2.5 Vstupně výstupní brány

Vstupně/ výstupní brány(lze označit i jako porty) nám umožňují připojit k mikroprocesoru jakýkoliv další logický obvod nebo součástku, co pracuje s logickými úrovněmi napětí. Vstupně/ výstupní brány jsou slučitelné s TTL/ CMOS logikou.

U standardní verze 8051 máme k dispozici 4 v/ v brány, z nichž každý má po 8mi pinech. Tyto čtyři brány jsou označovány P0, P1, P2 a P3. Pokud chceme označit konkrétní pin určité brány, pak označení vypadá např. takto: P1.4 , kde 1 před desetinou tečkou určuje bránu a 4 za tečkou je číslo konkrétního pinu příslušejícího k bráně 1.

2.6 Sériový kanál

Jednou z velkých výhod procesoru 8051, zvláště v době jeho vzniku, byl a je plně duplexní sériový kanál integrovaný na čipu procesoru umožňující komunikaci ve standardním 8 a 9-bitovém asynchronním režimu nebo 8-bitovém synchronním režimu s pevnou přenosovou rychlostí. Tím byla výrazně usnadněna komunikace s nadřazeným počítačem např. typu PC. Příjímací kanál je vybaven vyrovnávacím registrem, do kterého jsou uložena právě přijatá data, čímž je umožněn okamžitý příjem další dat. Přijatá data však musí být převzata dříve než je dokončen příjem následujících dat, který by způsobil přepsání původních dat.

Příjímací i vysílací registr je přístupný na adrese registru speciálních funkcí se symbolickým označením **SBUF** - registr sériového kanálu. Sériový kanál může pracovat ve čtyřech módech v závislosti na naprogramování registru **SCON** a nejvyššího bitu v registru **PCON**.

TI - Příznak prázdného vysílacího posuvného registru se nastavuje obvodově v okamžiku vysílání 8 přenášeného bitu v módu 0 nebo na začátku vysílání stop bitu v ostatních módech. Příznak TI je společně s příznakem RI zdrojem žádosti o přerušování sériového kanálu a proto nemůže být z důvodu identifikace zdroje (od TI nebo RI) přerušování nulován obvodově.

RI - Příznak přijatých platných dat se nastaví na konci příjmu 8 bitu v módu 0 nebo uprostřed přijímaného stop bitu v ostatních módech.

2.6.1 Pracovní módy sériového kanálu

Mód 0 - Sériová data se vysílají nebo přijímají vstupem označeným RxD synchronně s hodinovým posouvacím signálem vysílaným na výstupu TxD. Vysílá se 8 datových bitů, počínaje bitem s nejmenší vahou. Přenosová rychlost je pevná a rovna 1/12 kmitočtu oscilátoru.

Mód 1 - 8-bitový UART . Hodnoty se vysílají výstupem TxD a přijímají vstupem RxD a skládají se z deseti intervalů určených převrácenou hodnotou přenosové rychlosti v baudech pro přenos jednotlivých bitů. První bit je vždy nulový (log. 0) a představuje tzv. start bit, po němž následuje 8 přenášených bitů počínaje bitem s nejmenší vahou a posledním, který je vždy v log. 1 a představuje tzv. stop bit (přesně jeden stop bit). Pro přenosovou rychlost můžeme snadno odvodit tento vztah

$$\frac{1}{T} = \frac{2^{SMOD}}{32} \cdot \frac{f_{osc}}{12 \cdot [256 - (TH1)]} \quad (1)$$

,kde TH1 je obsah registru TH1 a f_{osc} je kmitočet oscilátoru. Pro dosažení nízkých přenosových rychlostí se využívá čítač v módu 1, kde po vzniku přerušování provedeme programové přednastavení potřebnou 16-bitovou hodnotou.

Mód 2 – 9ti bitový UART. Při vysílání je na vývodu TxD generováno 11 bitů v odpovídajících 11 intervalech daných přenosovou rychlostí uvedených start bitem (log. 0) a ukončených jedním stop bitem (log. 1) jako v předcházejícím módu. Přijímaný znak přichází přes vstup RxD. Devátým vysílaným bitem je hodnota bitu TB8 z registru

SCON a přijatý 9tý bit se ukládá do bitu RB8 v registru SCON a stop bit se ignoruje. Devátý bit může být využit k přenosu hodnoty(9 bitů) nebo k přenosu zabezpečovacího bitu například parity P(ochrany proti chybě v jednom bitu při přenosu), kterou získáme z registru PSW, jestliže ve střadači je uložena přenášená hodnota. Přenosová rychlost je dána $1/32$ nebo $1/64$ kmitočtu oscilátoru v závislosti na hodnotě bitu SMOD v registru PCON.

Mód 3 – 9ti bitový UART s programovatelnou přenosovou rychlostí. Příjem i vysílání hodnot probíhá stejně jako v módu 2 s tím, že přenosová rychlost je určena periodou přetečení čítače/časovače 1 a nastavení bitu SMOD.

3 SBĚRNICE I2C

[11] Sběrnice I2C(I2C-bus, Inter-Integrated Circuit), nazývaná též jako „i kvadrát c“, je standardní sériové dvou vodičové datové vedení. Jeden vodič přenáší data samotná(SDA) a druhý tvoří linku hodinového signálu(SCL). Sběrnice byla navržena firmou Philips začátkem roku 1980. Dodnes je používána u více než 50ti výrobců v řadě různých aplikací. Najdeme ji v mnoha typech řídicích systémů.

Patří mezi Multimasterové sběrnice, což znamená, že na jedné sběrnici může být více než jeden řídicí obvod nazývaný master. **Master** je obvod, jenž generuje příkazy podřízenému obvodu, jenž je nazýván jako **slave**.

3.1 Protokol sběrnice I2C

Jak bylo již zmíněno, jedná se o dvou vodičové vedení, SDA a SCL. Tyto vodiče přenášejí informace v podobě logických úrovní mezi všemi připojenými zařízeními a sběrnici samotnou. Každé zařízení (LCD displej, paměť, mikrokontrolér, klávesnice a další) připojené na sběrnici má svou jedinečnou adresu a je schopno vysílat či přijímat data v závislosti na jeho funkci. Např. LCD displej dokáže jen data přijmout, ale není konstruován tak, aby mohl data vysílat. Paměť na rozdíl od LCD displeje umožňuje obě tyto funkce, tedy přijímat i vysílat.

Zařízení připojené na sběrnici se může chovat jako master, či jako slave. Master většinou tvoří mikrokontrolér či jiné zařízení naprogramované k ovládání. Master také generuje časový signál pro umožnění přenosu, zavádí přenos a určuje, které zařízení s ním bude komunikovat a jak.

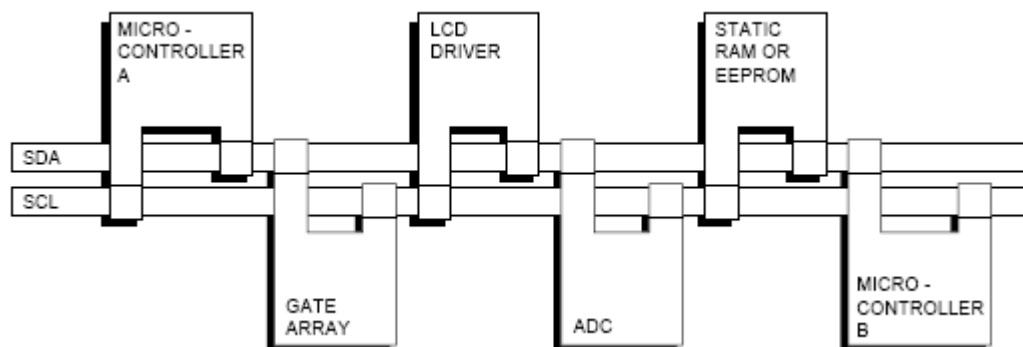
Princip zapojení zařízení na sběrnici je zobrazen na obr. 6. Také jsou zde vidět dva masterové obvody a zavedení multimasterové komunikace. Podle toho, jestli se o chod sběrnice stará vnitřní či vnější řízení, tak je sběrnice řízena buď jedním nebo více mastery, jenž si jsou vzájemně nadřazeny. Přenos probíhá ve dvou principech:

Mikrokontrolér A(master) posílá data Mikrokontroléru B(slave)

- Master vysílá adresový signál po sběrnici a přepíná do režimu přijímání signálu, zařízení tímto adresovým signálem oslovené (slave) pošle potvrzovací znak
- Master vysílá data a slave je přijímá.
- Master ukončí přenos

Mikrokontrolér A(master) přijímá data od mikrokontroléru B(slave)

- Master vysílá adresový signál po sběrnici a nastavuje do vysílacího režimu, oslovené zařízení (slave) opět pošle potvrzovací znak
- Slave vysílá data a master je přijímá.
- Master ukončí přenos



Obr. 6 Princip připojení zařízení na sběrnici I2C

3.1.1 SDA a SCL linky

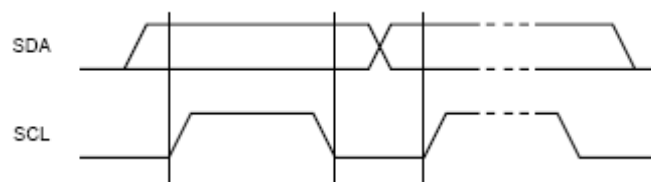
Oba tyto vodiče jsou obousměrné a jsou připojeny ke kladnému zdroji napájecího napětí přes proudový zdroj či zdvihací rezistor R_p . Pokud na sběrnici nejsou připojena žádná zařízení a neprobíhá po ní komunikace, jsou hodnoty na obou vodičích automaticky na vysoké logické úrovni.

Jejich logické úrovně nejsou pevně dané. Je to způsobeno různou technologickou konstrukcí připojených zařízení (CMOS, NMOS, bipolární tranzistory), jenž lze naráz

na sběrnici připojit. Proto je velikost napětí logické úrovně 1 a 0 závislá na velikosti napájecího napětí sběrnice V_{DD} . Hodnota V_{DD} je rovna hodnotě $U_{DD} = +5V$ maximálně $U_{DD} = +5,5V$. Z této hodnoty jsou pak odvozeny logické úrovně SDA a SCL. Logická úroveň 1 je rovna $0.7U_{DD}$ a logické 0 pak $0.3U_{DD}$. Dříve bylo použito hodnot 1.5V pro log. 0 a 3V pro log. 1, ale většina dnes vyráběných zařízení vyžaduje tuto specifikaci, tedy $0.3U_{DD}$ a $0.7U_{DD}$.

3.1.2 Platnost dat

Posílaná data na SDA vodiči musí mít stabilní průběh. To znamená, že během vysoké úrovně časovacího signálu SCL nesmí data poslána na SDA linku měnit svůj logický stav, neboť by mohlo dojít k přečtení nesprávné informace. Změna log. úrovně na SDA je možná pouze při nízké úrovni SCL. Pro každý přenesený bit je generován jeden časový puls.



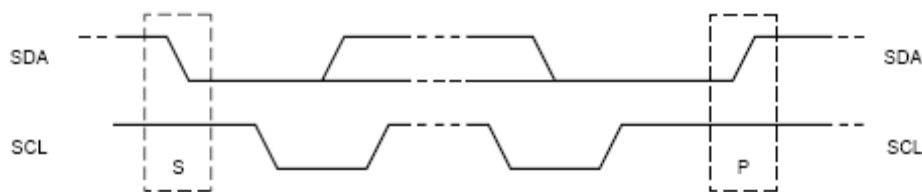
Obr.7 Znárodnění přenosu bitu sběrnice I2C

3.1.3 Start a stop podmínky

Veškeré přenosy jsou určeny podmínkou Start(S) a ukončeny podmínkou Stop(P).

- Startovací podmínka je definována přechodem z vysoké logické úrovně na nízkou během vysoké logické úrovně časovacího SCL signálu
- Ukončující podmínka je definována obráceně, tedy přechodem z nízké logické úrovně na vysokou během vysoké logické úrovně časovacího signálu SCL.

Obě tyto podmínky jsou generovány pouze masterem. Sběrnice je tedy v pracovním stavu od okamžiku vygenerování podmínky start až do generace podmínky stop.



Obr.8 Přenos dat s podmínkami startu a konce přenosu

3.1.4 8bitový formát

Každý byte poslaný po sběrnici I2C musí mít velikost 8bitů. Počet bytů přenesený po sběrnici je neomezený. Každá 8bitová kombinace dat je následována potvrzovacím bitem, dle kterého master určí, zda byl přenos správný či nikoli. Tento poslaný byte se přenáší od nejvýznamnějšího bitu (MSB) k bitu nejméně významnému (LSB).

3.1.5 Potvrzovací znak (ACK)

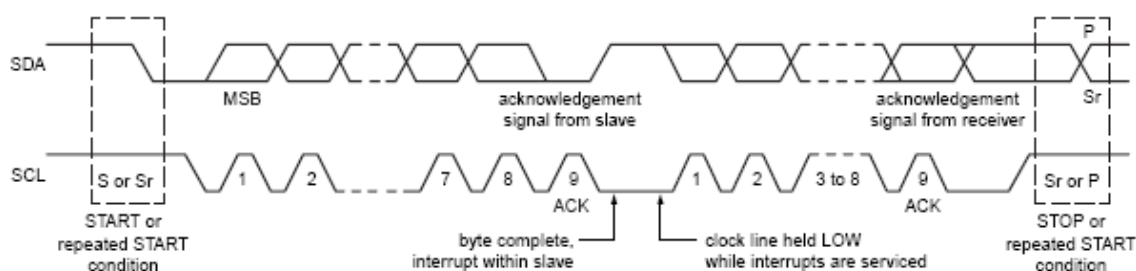
Tento znak se vysílá po každém přenosu jednoho bytu. Slouží k tomu, aby signalizoval vysílači, zda byl příslušný byte úspěšně či neúspěšně odeslán, aby mohl nastat přenos dalšího bytu. Všechny hodinové impulsy i tento 9tý potvrzovací znak je generován pouze masterem.

ACK potvrzovací znak je definován následujícím způsobem. Master „uvolňuje“ neboli nastaví během 9tého hodinového impulsu SDA linku do úrovně log. 1. To umožňuje přijímači stáhnout SDA linku do úrovně log. 0. Stane-li se tak při vysoké úrovni 9tého hodinového impulsu, je SDA linka na úrovni log. 0. Pak je přenos úspěšný a byl odeslán potvrzovací znak (ACK). Je-li SDA linka na úrovni log. 1, přijímač nestihl SDA stáhnout čili je na něm stále log. 1 a tím nebyl odeslán ACK, ale NACK (nepotvrzovací znak). Stane-li se tak, master může buď přenos opakovat (nastává generace startovací podmínky a celý proces se opakuje), nebo vygeneruje stop podmínku a přenos ukončí.

Existuje pět podmínek pro generaci NACK (nepotvrzovací znak):

- Na sběrnici není žádné zařízení nebo žádné zařízení s vyhovující adresou, čili není připojené žádné zařízení umožňující generaci ACK.

- Příjímač není schopný přijmout data z důvodu nějaké jiné činnosti a není připraven ke komunikaci s masterem.
- Během přenosu již přijímá přijímač data.
- Během přenosu, kdy už přijímač nemůže přijmout více bytů.
- Když master přijímá data, generací NACK signalizuje konec přenosu k opakovanému vysílání.



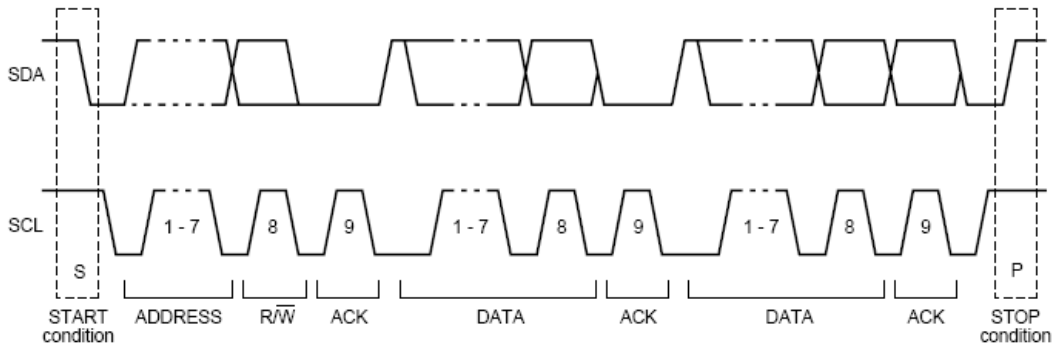
Obr. 9 Princip přenosu jednotlivých bitů a naznačení ACK a NACK

3.1.6 Adresace zařízení a R/W bit

Každé zařízení má svou 7mi bitovou adresu. Tato adresa určuje jednoznačnou identifikaci zařízení připojeného na sběrnici. Součástí této 7bitové adresy je 8mí bit určující pracovní stav adresovaného zařízení. Přenos dat při adresaci je znázorněn na obr. 10 . Přenos je spuštěn startovací podmínkou, po té následuje přenos 7mi bitů adresy zařízení a přenos 8mého bitu k nastavení, zda zařízení bude vysílat nebo přijímat. 9tý bit je potvrzovací bit, po kterém master rozhodne, jestli bylo zařízení osloveno a jestli tedy může být spuštěn přenos dat či nikoli. Následuje přenos dvou bytů dat. Oba tyto byty jsou ukončeny potvrzovacím znakem. 8mí bit R/W se nastavuje do těchto stavů:

- Logická 1, **Read**, jednoznačně určuje, že volané zařízení bude číst data vyslané masterem.
- Logická 0, **Write**, určuje, že master bude přijímat data od slavu. Tedy slave se stává vysílacím zařízením a master přijímacím.

Pro sběrnici I2C byla navržena i 10ti bitová adresace. Umožňovala rozšíření přípojných zařízení na sběrnici. Zařízení s 10ti či 7mi bitovou adresou lze připojit na stejnou sběrnici a to ve všech rychlostních režimech. Tato 10ti bitová adresace se ale v praxi používá jen velmi zřídka.



Obr. 10 Znáornění start a stop podmínky

3.1.7 Rezervované adresy

Několik vnitřních adres zařízení je na I2C vyhrazeno pro speciální funkce. Např. adresa 0000000 je určena pro vysílání adresy oslovující všechna zařízení na sběrnici, tzv. „broadcast“. Další adresy jako jsou 0000011, 00001XX a 11111XX jsou rezervovány pro různé účely.

3.1.8 Prodloužení signálu CLK

Prodloužení hodinového signálu neboli pozastavení přenosu spočívá v tom, že se hodinový signál podrží na úrovni log. 0. Z kapitoly 3.1.2 vyplývá, že přenos nemůže pokračovat, dokud se hodnota CLK nedostane opět na úroveň log. 1.

Tento zpožďovací systém není nezbytný, proto také většina zařízení tento obvod v sobě nainstalovaný nemají.

3.1.9 Softwarový reset sběrnice

Tento softwarový reset umožňuje jednoduchým způsobem obnovit činnost na zamrzlé, či jinak nekomunikující sběrnici. Spočívá vysláním 2 bytů dat. 1. byte

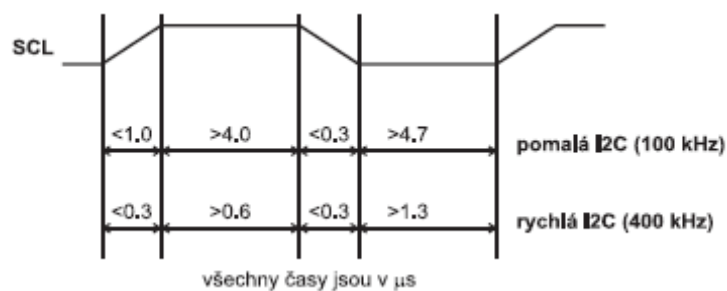
představuje všeobecné volání, general call, adresy 00H, po které se nahlásí všechna zařízení do stavu přijímačů. Následující 2. byte, adresa 06H, vyvolá reset. Bohužel opět touto funkcí lze ovládat jen malou hrstku některých zařízení.

3.1.10 Rychlosti sběrnice

První verze této sběrnice byla omezena na maximální rychlost přenosu 100 kbit/s. Postupem času se rozšířila na 4 různé rychlostní režimy.

- Standart-mode (Sm), první verze sběrnice s přenosovou rychlostí 100kbit/s
- Fast-mode (Fm), rychlost navýšena na 400kbit/s
- Fast-mode plus (Fm+), rychlost 1Mbit/s
- Hight-speed mode (Hs-mode), rychlosti pohybující se kolem 3,4 Mbit/s

Všechna zařízení, která byla vyrobena pro vyšší přenosové rychlosti, musí být zpětně kompatibilní s pomalejšími.



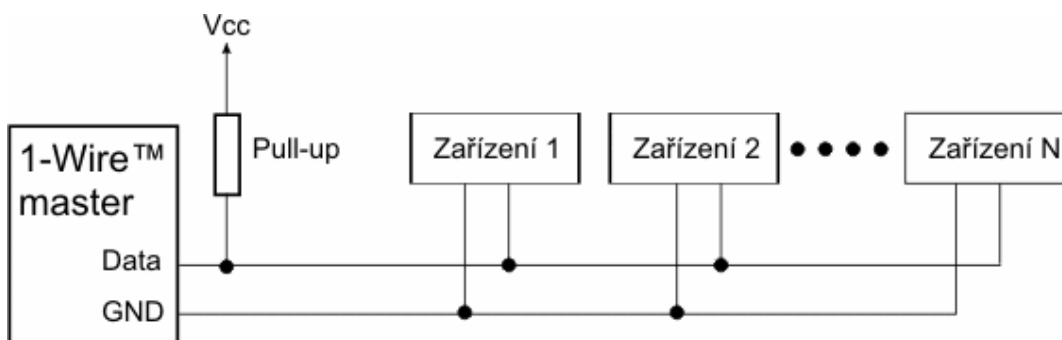
Obr.11 Ukázka časových intervalů přechodů z logického stavu pro uvedené rychlosti přenosu

4 SBĚRNICE 1-WIRE

[7, 10] Rozhraní sběrnice 1-Wire bylo vyvinuto zhruba v 90. letech firmou Dallas Semikonduktor. Její jednoduchost umožňuje komunikaci po jednom jediném vodiči. Připojení je ovšem po dvou vodičích, přičemž druhý vodič tvoří společnou zem. Většina zařízení na sběrnici připojené jsou jednoúčelové obvody. Především teplotní čidla. Tato čidla můžeme nalézt například v PC, kde pomocí kroucené dvoulinky je snímána teplota procesoru. Tato sběrnice je použita také u technologie iButton, která se používá v elektronických zabezpečovacích systémech.

4.1 Protokol sběrnice 1-Wire

Sběrnice 1-Wire je jednomasterová sběrnice, nemohou zde být dva řídicí obvody. Master ovládá všechny připojené slave zařízení. Pokud se na sběrnici připojí pouze jedno zařízení slave je systém označen jako jednobodové vedení, pokud je jich více, označuje se jako více bodové zapojení. Všechna zařízení, ať jedno či více, jsou připojeny na stejný datový vodič a společnou zem, tak jak je naznačeno na obrázku 12 s vlastní napájecí složkou.



Obr.12 Zapojení jednotlivých zařízení na sběrnici 1-wire

Data a příkazy jsou vysílány v pořadí od nejméně významnému k nejvíce významnému bitu. Komunikace je řízena asynchronně a poloduplexně. Každá jednotka, ať master nebo slave, musí mít v sobě integrován generátor synchronního pulsu. Přenos je oboustranný, umožňuje vysílání i přijímání, ale lze data odvyšlat a až teprve po té data přijímat. Nelze dosáhnout obojího v témže přenosu.

Přenosová rychlost sběrnice 1-wire je dána na hodnotu 16 kbit/s. Je to rychlost, která byla vybrána jednak pro zajištění maximální spolehlivosti přenosu dat na velké vzdálenosti. Rychlost 16 kbit/s se může mírně posouvat k nižší či vyšší přenosové rychlosti a to z důvodu zpomalení jednotlivých bitů dat. V některých případech lze dosáhnout rychlosti 128 kbit/s, tak zvaný overdrive. Tento režim si klade velké nároky na kvalitu vodičů a komunikaci na kratších vzdálenostech.

Zařízení, které jsou připojené na sběrnici, mají svou vlastní paměť. Tato paměť má v sobě uloženou unikátní 64 bitovou adresu připojeného zařízení, která, umožňuje jednotlivá připojená zařízení od sebe odlišit a komunikovat pouze s těmi, které byli adresou osloveny.

4.1.1 DATA linka

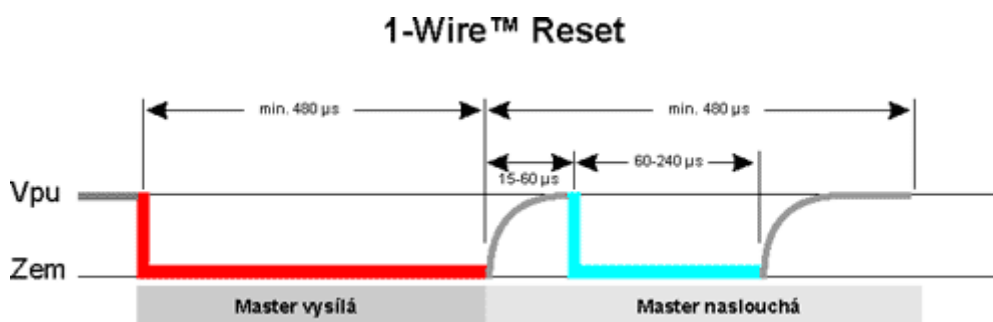
Jedná se vodič přenášící data mezi masterem a slavem. Sběrnice je neaktivní v logické 1, což je zabezpečeno zdvihacím rezistorem. Všechna zařízení mají budič s otevřeným kolektorem, pomocí kterého mohou sběrnici stáhnout na logickou 0.

Komunikace je rozdělena do tzv. časových slotů, trvajících přibližně 60 až 120 μ s. Jeden bit je přenesen během jednoho časového slotu. Master zahajuje veškerou komunikaci generováním sestupné hrany, kterou se synchronizují všechna slave zařízení. Na sběrnici je definováno pět základních operací: zápis 1, zápis 0, čtení 1, čtení 0 a reset.

4.1.2 Reset puls

Začátek každé komunikace na 1-wire sběrnici začíná tzv. „reset pulsem“ a tím se i zjišťuje připojení jednoho či více zařízení. Reset puls spočívá v stáhnutí datového vodiče do logické nuly a drží ho na této úrovni přibližně 480 μ s. Poté se sběrnice opět uvolní. Zdvihací rezistor zatím vrátí sběrnici do logické jedničky. Hned po té nastavíme sběrnici do logické 1 po dobu 15 – 60 μ s. Necháme ji chvíli ustálit a čteme ji. Pokud se dostane hodnota na logickou nulu, znamená to, že na sběrnici je nějaké zařízení, které sběrnici zemní. Tento proces trvá přibližně 100 μ s, čekáme a při tom čteme stav sběrnice. Nejlépe znázorněné je to na obrázku 13.

- Stáhnout sběrnici na 0, počkat 480 μs
- Uvolnit sběrnici, počkat 70 μs
- Přečíst sběrnici. je-li ve stavu log. 0, je na ní připojeno nějaké zařízení. Pokud je v log. 1, žádné zařízení připojeno není.
- Počkat 410 μs



Obr.13 Průběh resetovacího pulsu sběrnice 1-wire

4.1.3 Vysílání dat

Jestliže správně detekujeme zařízení a správně se nám ohlásí, můžeme začít vysílat a přijímat data. Data jsou vysílána a přijímána v časových intervalech. Časový interval je dlouhý 60 – 120 μs a během každého tohoto intervalu je přijat, nebo vyslán jeden bit informace. Mezi časovými intervaly musí být nějaká časová mezera, alespoň 1 μs , kdy je sběrnice v klidu.

4.1.4 Zápis logické 0 a logické 1

Zápis logické úrovně vysílá pouze master. Zápis logické jedničky funguje tak, že master stáhne sběrnici k nule minimálně opět na 1 μs a nejpozději do 15 μs od začátku ji opět uvolní a ponechá uvolněnou. Zdvihací odpor ji opět vytáhne k logické jedničce.

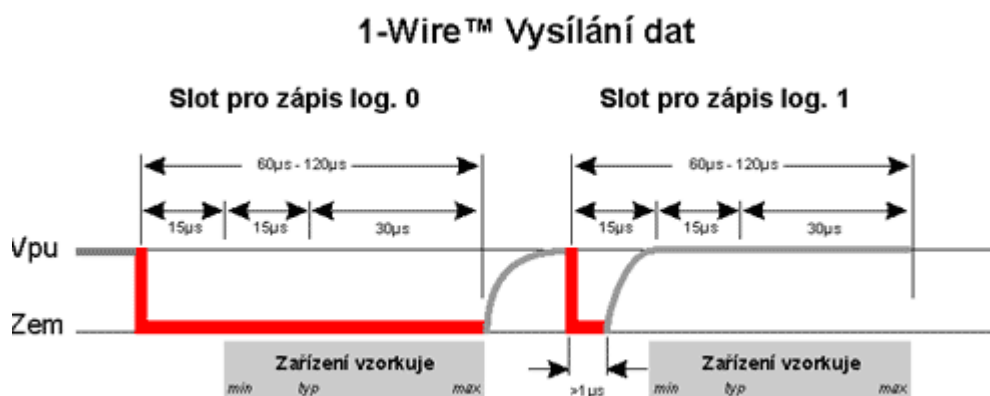
Zápis logické nuly master stáhne sběrnici do logické 0 a nechá ji tam po celou dobu časového intervalu, tedy 60 – 120 μ s. Zařízení, slave, naslouchá a vzorkuje asi po 30 μ s.

Logická jedna

- Stáhnout sběrnici na 0, počkat 6 μ s
- Uvolnit sběrnici, počkat 64 μ s

Logická nula

- Stáhnout sběrnici na 0, počkat 60 μ s
- Uvolnit sběrnici, počkat 10 μ s



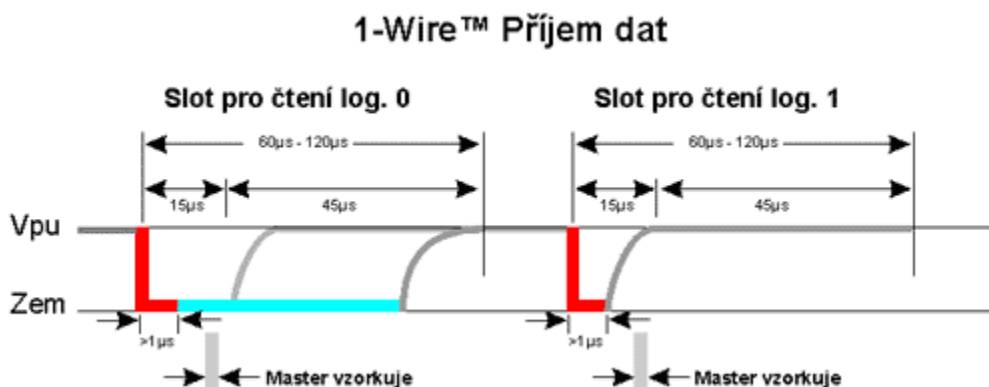
Obr.14 Zápis logické 1 a logické 0 na sběrnici 1-wire

4.1.5 Čtení logické 0 a logické 1

Čtení inicializuje opět master. Stáhne sběrnici na logickou nulu opět minimálně na 1 μ s a uvolní jí. Tím může zařízení vyslat jeden bit ponecháním sběrnice v klidu (logická 1), nebo že ji stáhne (logická 0).

- Stáhnout sběrnici na 0, počkat 6 μ s

- Uvolnit sběrnici, počkat 9 μs
- Přečíst sběrnici, její stav udává přečtený bit.
- Počkat 55 μs



Obr.15 Čtení logické 1 a logické 0 na sběrnici 1-wire

4.1.6 Adresace slave zařízení

Každé 1-wire zařízení má výrobcem přiřazené 64 bitové unikátní identifikační číslo. Toto číslo se skládá z typu zařízení (spodních 8 bitů je kód rodiny produktů, teplotní čidla, lcd a pod), sériového čísla (48 bitů) a z CRC kódu (bytu který je vypočítaný z prvních 56ti bitů ROM kódu). Pomocí tohoto čísla je každé zařízení jednoznačně identifikovatelné.

4.1.7 Příkazy 1-wire protokolu

Po iniciaci reset pulsu a potvrzení připojeného zařízení, očekávají slave zařízení připojená na sběrnici tzv. „ROM“ příkaz. Tímto příkazem se adresuje konkrétní zařízení na sběrnici (lze jich adresovat i více naráz). Vybrané zařízení poté očekává příkaz k provedení určité operace (tzv. funkční příkaz). Mikroprocesor emuluje všechny základní ROM příkazy: READ ROM, SKIP ROM, MATCH ROM a SEARCH ROM. Tyto příkazy a slave adresy jsou uvedeny pro většinu zařízení na této adrese <http://owfs.sourceforge.net/commands.html>, kde jsou i uvedeny hexadecimální hodnoty těchto příkazů.

5 KOMUNIKACE S PC

[3, 4] Komunikace s PC je ovládána přes komunikační port, sériový port, známí jako RS-232. RS 232 je dnes součástí každého stolního počítače. Jeho označení bývá uváděno jako COM1 nebo COM2. Prvotní účel bylo spojit počítač s okolním světem pomocí modemu. Postupem času se na něj začaly připojovat různé periferní zařízení jako je tiskárna, myš apod.. Dnes je bohužel využíván jen zřídka a je nahrazen USB sběrnici (Pokud PC není vybaven portem RS-232, můžeme použít tzv. redukci USB/RS-232, která by měla plnohodnotně sériový port nahradit. Tyto redukce musí být plnohodnotně kompatibilní s modemovou komunikací.). Nicméně RS-232 pro měřicí a mikroprocesorovou techniku stále svůj účel plní, neboť je odolný proti zničení, což umožňuje provádět různé pokusy. Je ovladatelný pod všemi operačními systémy, nicméně systém Windows Vista určité funkce blokuje. Zařízení lze odpojit a připojit i při zapnutém PC. Napájecí napětí lze odebírat přímo z portu.

Port má k dispozici celkem 4 vstupní a 2 výstupní porty, které lze snadno měnit díky instrukcím v různých programovacích jazycích.

Přenos přes sériový port s modemem, našim výrobkem a jinými produkty pracující na této komunikační lince se uskutečňuje vývody 2, TxD a 3, RxD. Tyto dvě linky umožňují snadný přenos dat, pomocí jednoho příkazu, přičemž ostatní se musí programovat krok po kroku, jednotlivý bit po bitu. Výhodou je okamžité čtení a přímé nastavování jejich stavů.

Elektrické vlastnosti jsou dány normou RS-232, tj. ve stavu logické 0 je na výstupu -12 V, při log. 1 +12 V. Jejich odebíraný proud může být přibližně 10 mA, což umožňuje připojení některých součástek s nízkou spotřebou přímo na linku. Vstupy mají odpor 10 kΩ. Stav logických úrovní pro log. úroveň 1 je napětí nad 1,25 V a pod 1 V se jedná o log. 0. Dochází-li k přepínání setkáme se s problémem malé hystereze, proto se rozpoznání stav ukáže teprve tehdy, bude-li velikost vstupního napětí jiná než tento rozsah pro log. úrovně. Díky malé napěťové náročnosti může port RS-232 pracovat i v úrovni TTL logiky nebo v přenosných počítačích, kde pracovní napětí jsou ±3V.

V sériovém portu se využívá obvod UART 8250 firmy National Semiconductor. Zkratka UART představuje Universal Asynchronous Receiver/ Transmitter. Tento

obvod zajišťuje přeměnu sériové posloupnosti bitů do paralelního znaku a naopak, přičemž obě funkce jsou na sobě nezávislé, tím je umožněn plně duplexní přenos.

6 POUŽITÉ OBVODY NÁVRHU

Obvody, které jsou použity v návrhu jsou vybírány díky své cenové relaci, jednoduchosti a informační dostupnosti.

Mikroprocesor z rodiny 8051 byl vybrán z důvodu integrovaného odvodu UART a předchozími zkušenostmi s tímto typem mikroprocesoru. Z tohoto hlediska, je nutné realizovat převod logických úrovní mezi TTL logikou a napěťovými úrovněmi sériovém portu. K tomuto účelu je použit obvod MAX232, který zajišťuje potřebnou napěťovou přeměnu.

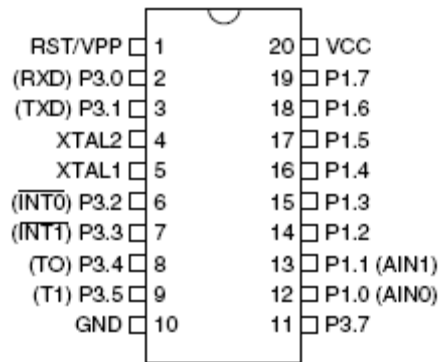
Tyto dvě nejdůležitější součástky tvoří základ hardwaru pro převod mezi sběrnicemi. Podle manuálu, který udává výrobce, se k nim zapojili ostatní pasivní součástky zajišťující správný chod.

6.1 Mikroprocesor z rodiny 8051, AT89C2051

[14] Na realizaci byl vybrán mikroprocesor od firmy ATMEL, AT89C2051. Tento obvod má dostačující počet vstupně/ výstupních obvodů. Pro sběrnice I2C a 1-wire je zapotřebí 3 vývodů. Tento procesor má vývodů celkem 15, což nám umožňuje do budoucna ještě realizace dalších sběrnic. Paměť procesoru má velikost 2 kB, což bohatě dostačuje na paměť pro program. Dále je procesor vybaven:

- Paměť typu flash, kterou lze ji řádově tisíckrát přepisovat
- Dvěma 16cti bitovými časovači/čítači
- Pracovní frekvencí 0 - 24MHz, dle použitého krystalu
- Programovatelný sériový UART obvod

Překládací program assembleru pro procesoru tohoto typu je též snadno přístupný na internetu např. na stránce <http://plit.de/asem-51/download.htm> nebo na přiloženém CD.

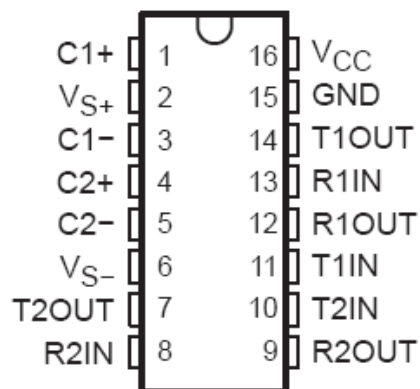


Obr.16 Přehled vývodů mikroprocesoru AT89C2051

6.2 MAX 232

[13] U převodníku z RS-232 na TTL logiku a naopak, požadujeme 5V napájení. Nejpoužívanější obvod pro připojení k RS-232 se stal MAX232 od firmy Maxim. Tento obvod je použit i v našem řešení. Některé firmy vytvořily podobné obvody, které se staly možnými alternativami (ICL23, DSC14C23).

Obvod MAX232 stačí napájení 5V. Jeho základní zapojení je zobrazeno na obrázku 17. S použitím čtyř kondenzátorů o kapacitě 1 μ F obvod zajišťuje správný chod. S mikrořadičem komunikuje pomocí TTL logiky.



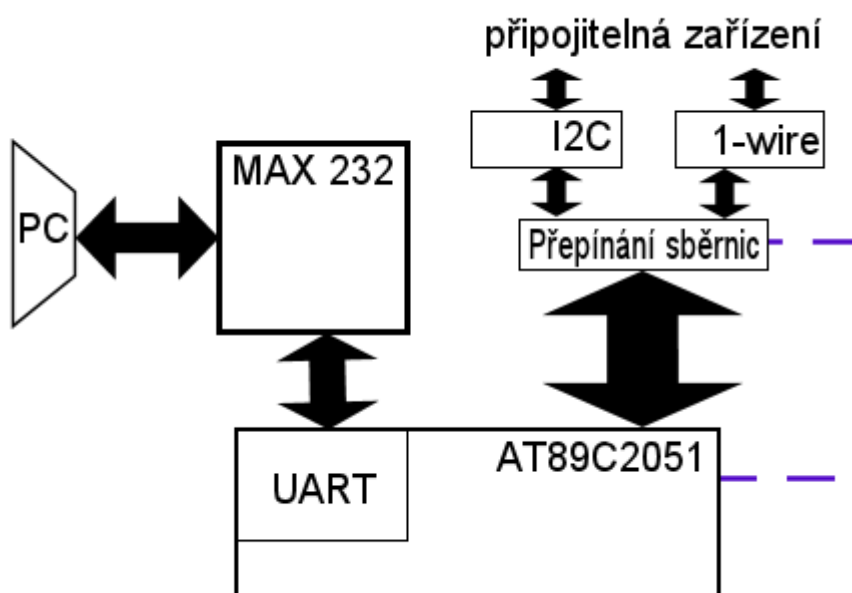
Obr. 17 Přehled vývodů obvodu MAX232

7 NÁVRH ŘEŠENÍ

Řešení se skládá ze dvou částí, z části hardwarové a z části softwarové. Jako programovací jazyk pro mikroprocesor je zvolen assembler, což je programovací jazyk velice blízký strojovému kódu, též se nazývá jazyk symbolických adres a je závislý na strojovém kódu každého typu procesoru.

S mikropočítačem je komunikace navázána přes freewarovou verzi programu Terminal 1.9b. Pomocí tohoto programu se přistupuje k sériovému portu a nastavení přenosové rychlosti ze strany PC. Také se jím posílají ovládací příkazy pro mikroprocesor a čtení dat vyslané mikroprocesorem. Program může být používán pro komunikaci s odlišnými zařízeními, např. modemy, routery, GSM telefony připojené na sériovém portu.

Na obrázku 18 je znázorněno blokové schéma realizovaného výrobku. Funkční princip je velmi jednoduchý, počítač připojený přes RS-232 s mikroprocesorem, potřebuje mezi sebou mít napěťový převodník úrovní, který je tvořen obvodem MAX232, popř. MAX202. K vývodům mikropočítače jsou pak připojeny jednotlivé sběrnice a pomocí přepínače sběrnic, je to pouze softwarová utilita, se nastavuje práce s jednotlivou sběrnicí.



Obr. 18 Blokové schéma realizovaného výrobku

7.1 Hardwarová část

Hardwarovou část tvoří tedy mikroprocesor AT89C2051. K němu bylo nutné připojit krystal pro chod procesoru bez nutnosti externího časování, který je připojen na vývodech 4 a 5. Použitím krystalu o frekvenci 11,0592 MHz se nejlépe počítá přenosová rychlost mezi PC a mikroprocesorem. Vzorec pro výpočet je uveden v kapitole 2.4.1 vzorec (1). Na pin 1 je umístěn resetovací obvod. Jelikož mikroprocesor AT89C2051 reaguje na vysokou logickou úroveň, je resetovací obvod složen z kondenzátoru připojeného na +5 V a pin 1. Při zapojení obvodu do sítě, se kondenzátor nabije a pomalu se přes zatěžovací odpor RST vstupu a rezistoru R1 připojeného k zemi vybíjí. Po vybití je na RST opět 0. Paralelně s kondenzátorem a RST vstupem je tlačítko umožňující nahrazení zapnutí/ vypnutí obvodu ze sítě. Vybití kondenzátoru trvá asi 2s, což je z hlediska funkčnosti zbytečné. Délka resetovacího pulsu by měla být nejméně 2 strojové cykly.

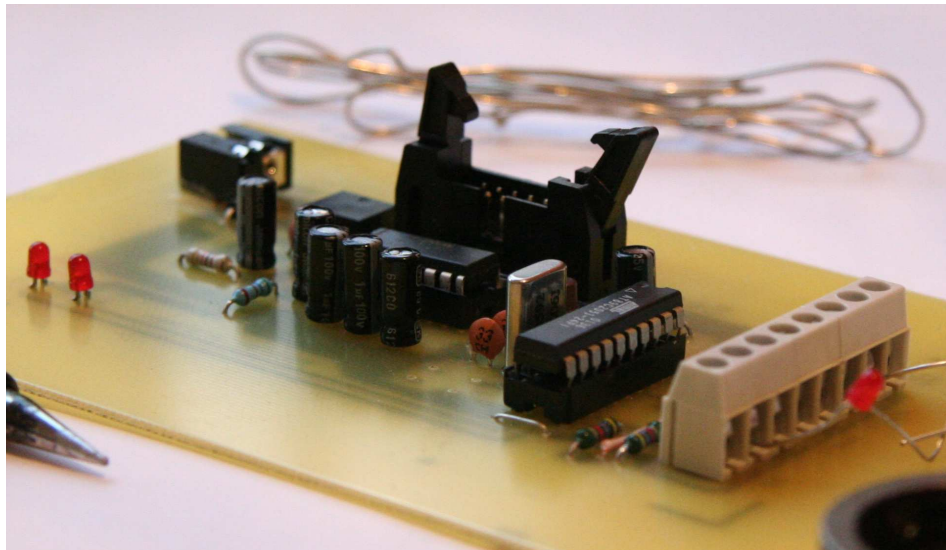
Piny 8 a 9 slouží k signalizaci vybrané sběrnice. Jejich reakce je nastavena na logickou 0, proto se na ně připojují katody LED diod LED2, LED3. Anody jsou přes rezistor připojeny ke kladnému napětí, tedy při logické nule, což tvoří zem, začne protékat proud a diody se rozsvítí.

Připojení sběrnic je na pinech 11 a 12 pro sběrnici I2C, přičemž 11 slouží pro vodič SDA a 12 jako SCL a 17 pro DATA vodič sběrnice 1-wire. Tyto zmiňované vodiče sběrnice jsou navíc připojeny k napájecímu napětí +5 V přes zdvihací rezistory. Zdvihací rezistory mají hodnotu 4,7 k Ω a zajišťují při uvolnění sběrnice zvednutí z nízké logické úrovně na vysokou.

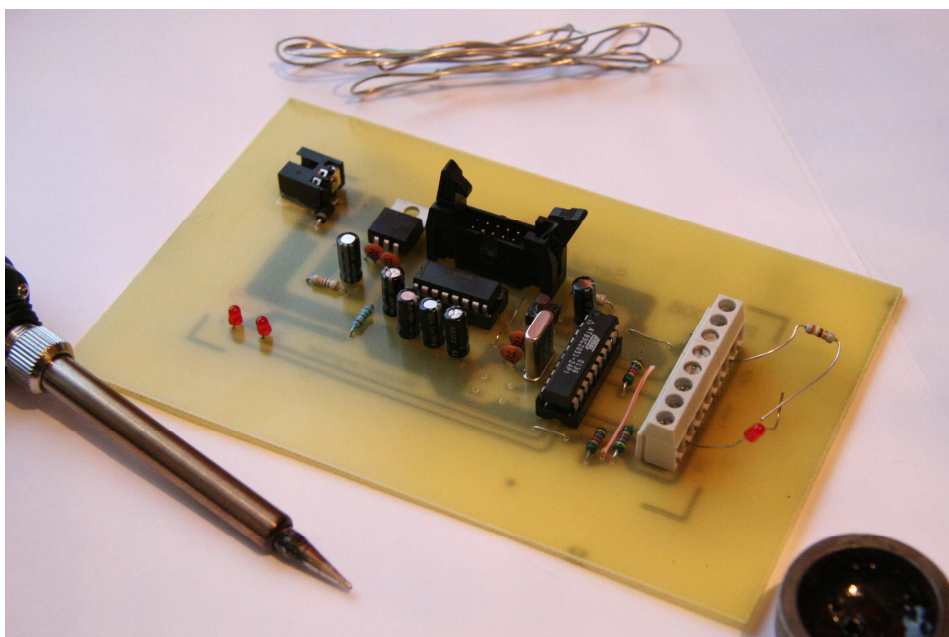
Na další piny 2 a 3 se přivádí sériová komunikace mezi mikroprocesorem IO1 a obvodem zajišťující převodu mezi TTL a RS-232, MAX232. Obsahuje dvě dvojice oddělovačů konvertující napěťové úrovně. Napětí pro RS-232 se získává pomocí nábojové pumpy a výstupní napětí proto značně závisí na kvalitě použitých kondenzátorů. Do obvodu MAX232 jsou přivedeny na vstup výstupy z PC konektoru, TxD, RxD.

Celý obvod je napájen napětím +5 V a zemí. Toto napájení je zajištěno stabilizačním obvodem 7805. Obvod je připojen přes ochranu usměrňovací diodu k napájecímu stejnosměrnému napětí, které se přivádí ze zdroje, nebo usměrňného

transformátoru. Doporučená hodnota napájení je 12 – 16 V. Ochranná dioda slouží k ochraně před přepólováním napájecího napětí.



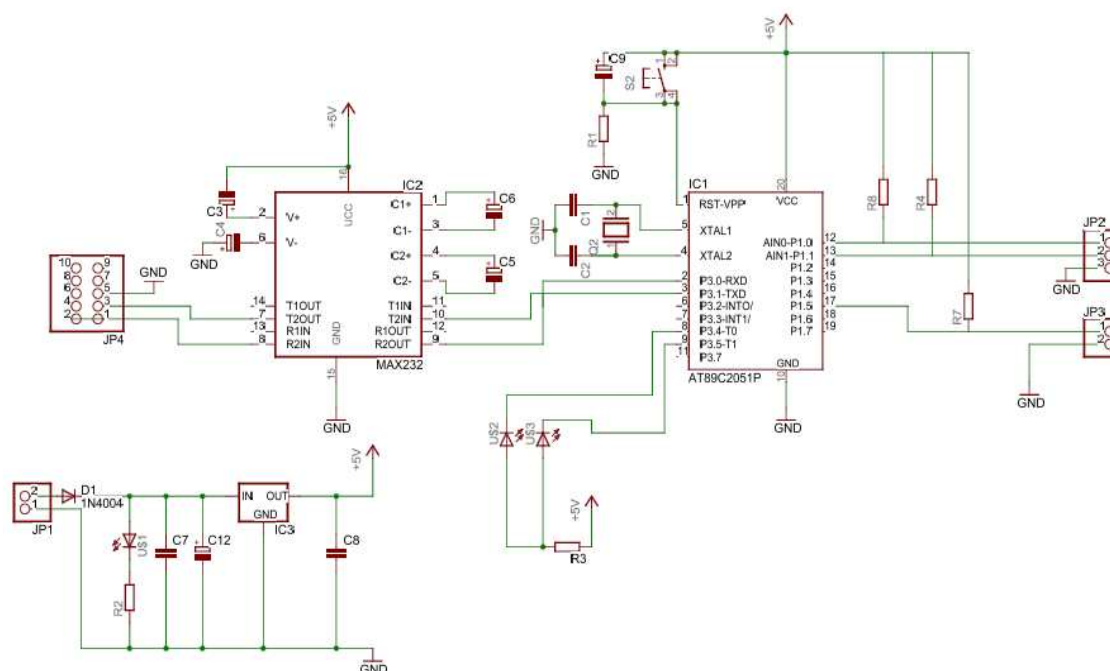
Obr. 19a Fotografie výrobku



Obr. 19b Fotografie výrobku

7.1.1 Schéma zapojení

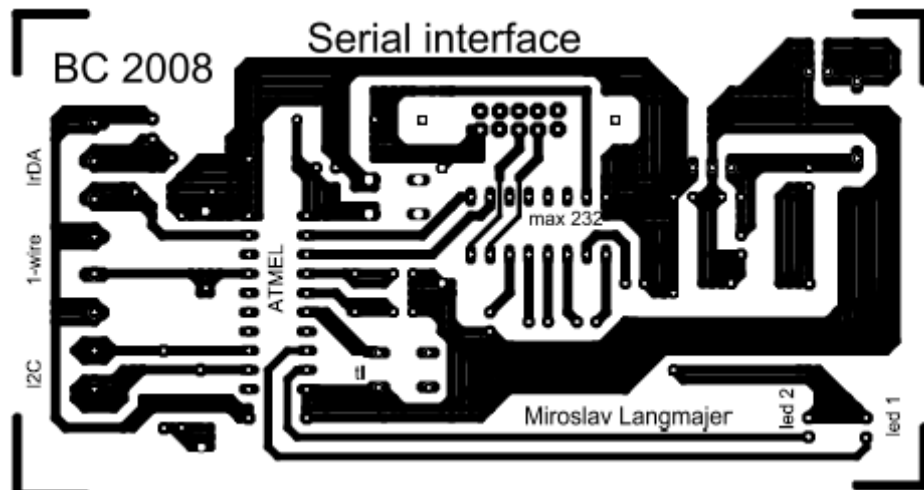
Toto schéma zapojení viz obrázek 20, bylo kresleno programem EAGLE verze 4.09r2. Při návrhu se vycházelo z data sheetu výrobců [13, 14] jednotlivých integrovaných obvodů a jejich postupné spojení do jednoho funkčního celku.



Obr. 20 Schéma zapojení výrobku

7.1.2 Deska plošného spoje

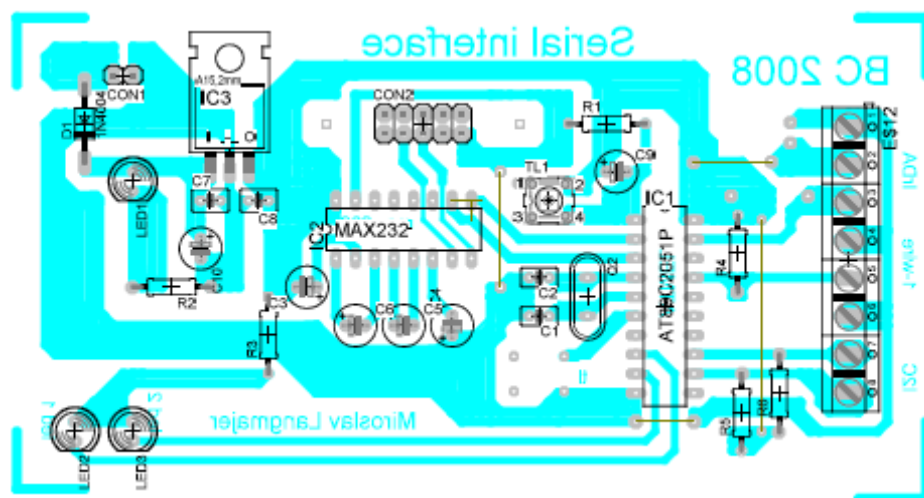
Plošný spoj je taktéž navržen programem Eagle, viz obrázek 21. Tento plošný spoj je připraven k zapojení ještě další, třetí, sběrnice. Touto třetí sběrnicí je komunikační infračervený port, IR, kterým by bylo možné například ovládat PC pomocí dálkového ovladače, nebo naopak ovládat nějaké zařízení na dálku počítačem.



Obr. 21 Plošný spoj výrobku

7.1.3 Rozložení součástek

Schéma, plošný spoj a rozložení součástek je uloženo též v příloženém CD ve formátu pdf a formátu pro EAGLE 4.09r2.



Obr. 22 Rozložení součástek výrobku

7.1.4 Seznam součástek

IC1	AT89C2051
IC2	MAX 232
IC3	TS7805, LM7805
C1, C2, C7, C8	33pF/16V
C3, C4, C5, C6	1 μ F/16V
R4,R5,R6	4k7 Ω
R1	47k Ω
R2	2k7 Ω
R3	470 Ω
LED1	zelená
LED2, LED3	červená

7.2 Softwarová část

Realizace softwarové části se skládá ze dvou částí. Jednou částí je program mikroprocesoru a druhou program Terminal 1.9b

7.2.1 Program mikroprocesoru

Tato část má v sobě definované jednotlivé procedury, které se volají na základě přijatého ASCII znaku od připojeného počítače a zároveň si mikroprocesor o tyto data žádá. Po deklaraci některých SFR, kterými na začátku nastavujeme komunikační rychlost po sériovém portu se mikroprocesor pokusí automaticky vybrat jednu ze sběrnic, na které je připojeno nějaké zařízení. Pokud jsou obě dvě spojeny s příslušnými

zařizování vybere se ta, která se v programu nachází jako první, čili I2C. Pokud není ani jedno zařizování na jednotlivé sběrnici připojené, nenastaví se ani jedna jako aktivní, led diody LED1 a LED2 začnou blikat a mikroprocesor začne pracovat podle následujícího popisu příkazů:

- Mikroprocesor odešle znak „B“ (Bus) a očekává jeden z následujících příkazů pro nastavení sběrnice:
 - „1“ – nastavuje jako aktivní sběrnici I2C
 - „2“ – nastavuje jako aktivní sběrnici 1-wire
- Pokud je nastavena sběrnice, procesor vygeneruje znak „I“ (instruction) čímž očekává jeden z následujících přijatých znaků. Každá sběrnice má jiný sled instrukcí, proto, i když jsou některé příkazy identické, je nutno pro každou sběrnici je popsat podrobněji. Výjimku tvoří příkaz „C“ (change) , který vyvolá přepnutí sběrnice.

1. Sběrnice I2C

- „r“ – (READ) Čtení dat, při přijmutí je vracen znak „a“ (adresa zařízení od něhož jsou data určena ke čtení)
- „w“ – (WRITE) Zápis dat, při přijmutí je vracen znak „a“ (adresa zařízení jemuž jsou data určena), zpět se vrátí znak „d“ (data), označující čekání na data, která se mají zapisovat.
- „G“ – (GENERAL CALL) = 00h. Tato adresa je užívána pro inicializaci všech zařízení na sběrnici, která jsou schopna reagovat na tento příkaz. Jestliže žádné zařízení neodpoví na přenesený bit, operace je zastavena Stop signálem na sběrnici a tím se vyšle zpět znak „e“(error).
- „S“ – (START BYTE) = 01h
- „R“ – (RESET) = 06h. Vyvolá resetování sběrnice I2C a vrací znak „y“ (yes) – zařízení na sběrnici připojeno a nebo „n“ (no) - žádné zařízení není připojené

2. Sběrnice 1-wire

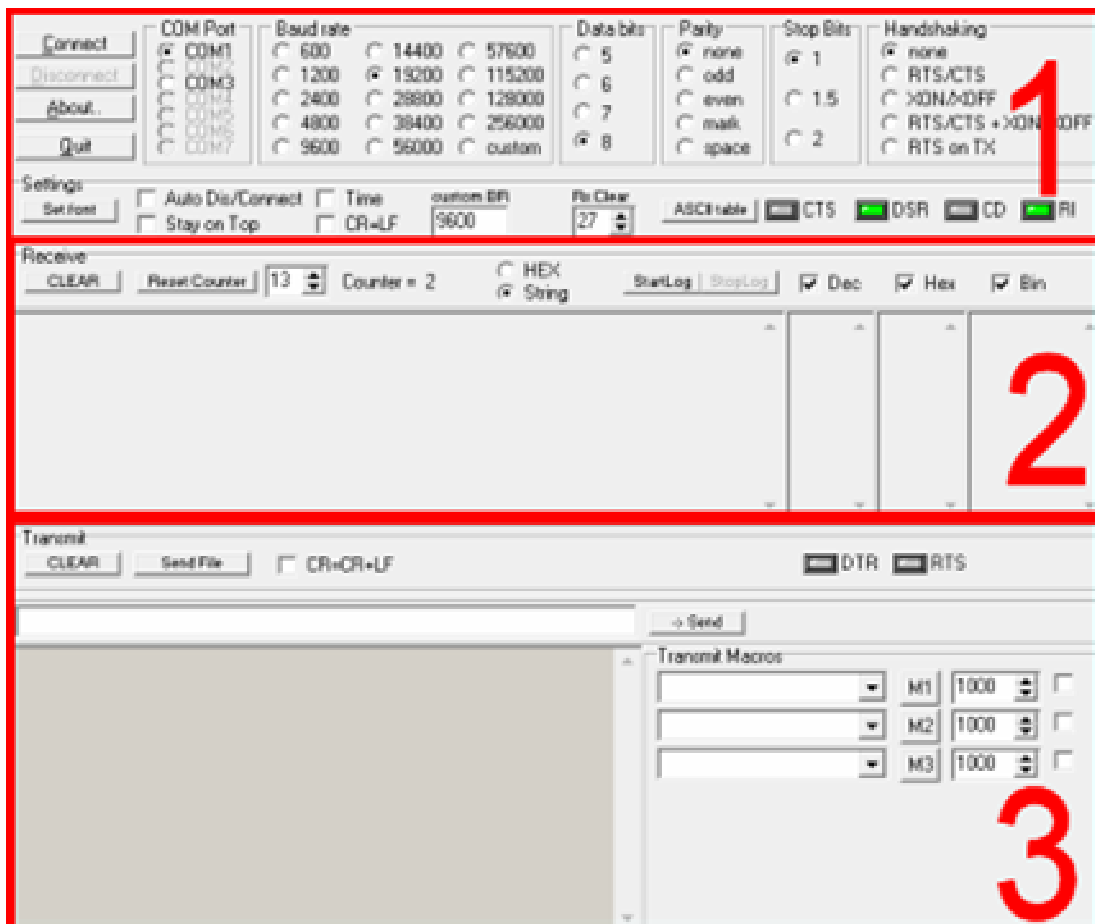
- „K“ – (SKIP) = CCH. Tento příkaz umožní, vyslat nějaký příkaz bez nutnosti odeslání ROM adresy. Např. pro všechna zařízení DS18B20(teplotní čidlo 1-wire společnosti Dallas), mohou po poslání SKIPu, hodnotou 44H nastavit teplotní konverzi všech čidel. Nebo může číst ze zařízení, aniž by se musela zadat 64 bitová ROM adresa.
- „M“ – (MATCH) = 55H. Po tomto příkazu následuje adresa připojeného zařízení, s nímž se má komunikovat. Toto zařízení bude nastavené jako aktivní a lze s ním komunikovat, jako by na sběrnici bylo jediné do té doby, než se vygeneruje resetovací puls.
- „S“ – (SEARCH) = F0H. Tento příkaz načítá všechna připojená zařízení a vrací jejich adresy, k podrobnému vysvětlení bych se odkázal na problematiku, zabývající se jí dopodrobna viz [12].
- „O“ – (READ ROM) = 33H. Tento příkaz umožňuje přečtení 64 bitové adresy pro pouhé jedno připojené zařízení. Pokud je jich tam více, začnou vysílat najednou všechny a dojde k nesprávnému přenosu informace.
- „E“ – (READ SCRATCHPAD) = BEH
- „W“ – (WRITE SCRATCHPAD) = 4EH. Příkaz umožňující zápisů několika bytů do zařízení, např. do čidla DS18d20 jsou to 3byty.
- „R“ – (RESET). Vyvolá reset sběrnice a přijme znak „y“ (yes) – zařízení na sběrnici připojeno a nebo „n“ (no) - žádné zařízení není připojené
- „r“ – (READ). Čtení dat, přijmutím znaku „n“ (number) se čeká na upřesnění, kolik bytů se má přečíst. Počet bytů určuje odeslání znaku „1“ až „9“, hned po odeslání mikroprocesor odešle do počítače příslušný počet bytů. Nejedná se o definovaný příkaz READ ROM, pouze o proceduru zajišťující další funkce.

- o „w“ – (WRITE) Zápis dat, přijmutím znaku „d“ (data) označuje čekání na data, která se mají zapisovat.

7.2.2 Program řízení Terminál 1.9b

Program řízení, Terminal 1.9b, se skládá ze tří částí jak popisuje obrázek 23:

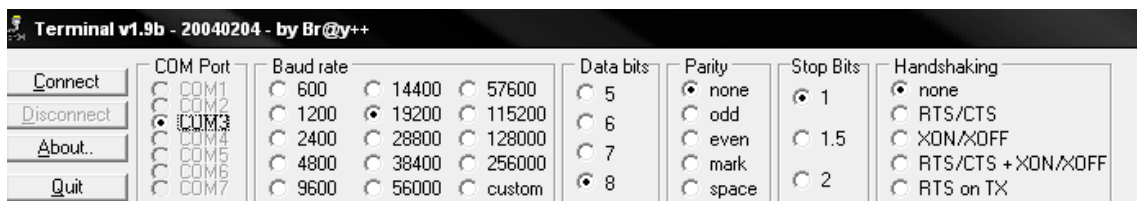
1. Tato část nastavuje typ a rychlost připojení, počet bitů, paritu a stop bi.
2. Zobrazuje přijmutí dat v ASCII, dekadickém, hexadecimálním a binárním kódu.
3. Posílá znaky, pouze ASCII, na port RS-232.



Obr. 23 Rozložení ovládacích částí programu Terminal 1.9b

Pro začátek komunikace je zapotřebí nastavit příslušný COM port počítače, na kterém je zařízení připojeno a komunikační rychlost. V době, kdy chceme navázat

komunikaci po tomto rozhraní, nesmí po vybraném COMu komunikovat jiné zařízení, tedy příslušný COM musí být volný. Pro navázání komunikace se zařízením je nutné před stiskem tlačítka „Connect“ nastavit COM1, nebo jiný, podle toho, na který se zařízení připojuje, přenosovou rychlost „Baud rate“ na 19200, „Data bits“ na 8, „Parity“ none, „Stop bits“ 1 a „Handshaking“ na none. Tak jak je uvedeno na obrázku 24. Po té stiskem tlačítka „Connect“ začíná komunikace mezi Terminálem 1.9b a převodníkem.



Obr. 24 Nastavení komunikace v programu Terminal 1.9b

Obrázek 25 popisuje část přijímání dat. Příjem dat probíhá automaticky po připojení zařízení. Jakmile zařízení vyšle řetězec z mikroprocesoru, či jiného vysílacího zařízení, jeho hodnota se zde objeví a to buď v podobě ASCII znaku a nebo jeho hexadecimálního kódu. Tlačítkem „CLEAR“ se vymaže obsah textového pole. Pokud chceme přijatá data vidět i v jiných kódech, např. binárním, musíme v pravé části tyto volby zaškrtnout.

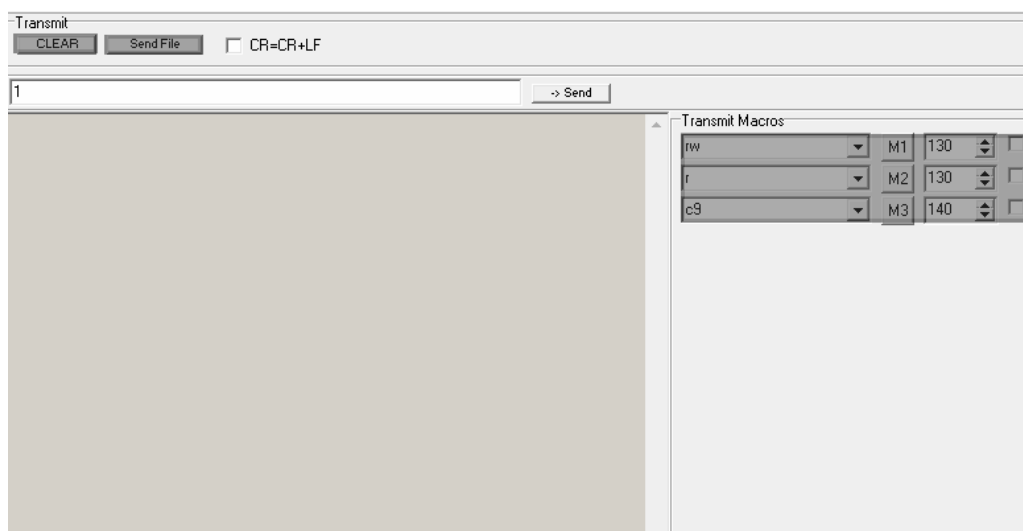
Chceme-li si přijatá data zálohovat pro další zpracování, stiskem tlačítka „Start Log“ vyběhne nabídka k uložení přijatých dat do souboru formátu txt, který si uložíme na vybrané místo. Ukončení ukládání hodnot se provádí stiskem tlačítka „StopLog“.



Obr. 25 Příjem dat programem Terminal 1.9b

Vysílání znaků jakožto příkazů pro mikroprocesor či jiné zařízení se nechá provádět třemi způsoby.

1. Napíšeme posloupnost znaků, o kterých víme, že budou mikroprocesorem očekávány a stiskem tlačítka „Send“ se odešlou po sériovém portu.
2. Vytvoříme si soubor, nebo několik souborů, do kterých uložíme posloupnosti příkazu a jednotlivé soubory odesíláme. Vhodné například pro programování pamětí, či mikroprocesorů, kde pomocí základních příkazů pro navázání komunikace přiložíme hexadecimální kód k zápisu do paměti.
3. pokud potřebujeme mezi jednotlivými příkazy nějakou časovou rezervu k jejich vykonání, nebo chceme-li je posílat v nějakém časovém intervalu, zaměříme se na pravou část programu, kde je „transit macros“. Zde je možnost vysílání až třech posloupností znaků za sebou v nastavitelném časovém intervalu(10 - 10000ms). Pokud chceme tento přenos spustit, stačí napsat znak, vložit interval zápisu a zaškrtnout checkbox a příkaz se provádí stále dokola.



Obr. 26 Vysílání dat programem Terminal 1.9b

Tento program navíc disponuje tabulkou ASCII znaků, takže je možné hned porovnat zda přijatý znak odpovídá příslušnému např. binárnímu, kódu. Bohužel, tento program není extra vhodný pro tuto komunikaci. Pokud chceme totiž poslat kód, který není v základním znaku ASCII tabulky, je to problém. Musí se proto tento problém

řešit pomocí zvláštních procedur mikropočítače a nebo si naprogramovat vlastní obslužný program. Program terminál slouží spíše k základním testovacím programům sériového rozhraní RS-232.

8 ZÁVĚR

V rámci této bakalářské práce jsem se seznámil s principy přenosu sériových sběrnic, funkcí mikrokontroleru a komunikací se sériovým portem.

Výsledkem této bakalářské práce je fungující sériové rozhraní, které umožňuje přepínat jednu ze dvou sběrnic. Výběr se provádí automaticky po zapnutí nebo softwarovým řízením ovládané počítačem. Zobrazení aktivní sběrnice znázorňuje rozsvícení konkrétní diody. Také bylo vyrobeno funkční schéma. Z tohoto schématu byly vytvořeny obrazce pro výrobu plošného spoje. Dále byl napsán program pro mikroprocesor v jazyce assembler, který obstarává komunikaci s RS232 a PC. Také ovládá sběrnice pomocí popsanych příkazů.

Realizovaný výrobek lze rozšířit ještě o další sběrnice. Návrh je konkrétně připraven na IR přenos, ke které se zapotřebí naprogramovat komunikační protokol.

9 SEZNAM POUŽITÉ LITERATURY

- [1] **PETR SKALICKÝ:** Mikroprocesor řady 8051, BEN technická literatura, 2005
- [2] **VÁCLAV VACEK:** Učebnice programování ATMEL s jádrem 8051, BEN technická literatura, 2002
- [3] **BURKHARD KAINKA:** Využití rozhraní PC pod Windows, HEL, 2000
- [4] **MIROSLAV ŠNOREK:** Standardní rozhraní PC, Grada
- [5] **JÁN HANÁK:** Visual Basic 2005 – pro pokročilé, Zoner Press
- [6] URL: <<http://www.dhservis.cz>>
- [7] URL: <<http://www.hw.cz>>
- [8] URL: <<http://www.volny.cz/d72>>
- [9] URL: <<http://www.mcu.cz>>
- [10] URL: <<http://www.maxim-ic.com/products/1-wire/>>
- [11] I2C DATA SHEET, URL:
<http://www.nxp.com/acrobat_download/usermanuals/UM10204_3.pdf>
- [12] URL: <<http://www.ibutton.com/ibuttons/standard.pdf>>
- [13] MAX232 DATA SHEET, URL: :
<<http://www.datasheetcatalog.org/datasheet/texasinstruments/max232.pdf>>
- [14] AT89C2051 DATA SHEET, URL:
<http://www.datasheetcatalog.org/datasheets/270/130451_DS.pdf>

10 SEZNAM OBRÁZKŮ

Obr. 1 Jádru procesoru a rozšíření o různé periferie.....	9
Obr. 3 Organizace vnější a vnitřní paměti	12
Obr. 4 Rozložení vnitřní paměti procesoru.....	13
Obr. 5 Rozložení bitů v stavovém registru	14
Obr. 6 Princip připojení zařízení na sběrnici I2C	21
Obr.7 Znázornění přenosu bitu sběrnice I2C	22
Obr.8 Přenos dat s podmínkami startu a konce přenosu.....	23
Obr. 9 Princip přenosu jednotlivých bitů a označení ACK a NACK.....	24
Obr. 10 Znázornění start a stop podmínky.....	25
Obr.11 Ukázka časových intervalů přechodů z logického stavu pro uvedené rychlosti přenosu	26
Obr.12 Zapojení jednotlivých zařízení na sběrnici 1-wire.....	27
Obr.13 Průběh resetovacího pulsu sběrnice 1-wire	29
Obr.15 Čtení logické 1 a logické 0 na sběrnici 1-wire.....	31
Obr.16 Přehled vývodů mikroprocesoru AT89C2051	35
Obr. 17 Přehled vývodů obvodu MAX232.....	35
Obr. 18 Blokové schéma realizovaného výrobku	36
Obr. 19a Fotografie výrobku.....	38
Obr. 19b Fotografie výrobku	38
Obr. 20 Schéma zapojení výrobku.....	39
Obr. 21 Plošný spoj výrobku.....	40
Obr. 22 Rozložení součástek výrobku	40
Obr. 23 Rozložení ovládacích částí programu Terminal 1.9b	44
Obr. 24 Nastavení komunikace v programu Terminal 1.9b.....	45
Obr. 25 Příjem dat programem Terminal 1.9b.....	45
Obr. 26 Vysílání dat programem Terminal 1.9b.....	46

11 OBSAH PŘILOŽENÉHO CD

- text bakalářské práce v elektronické podobě
- ovládací program Terminal 1.9b
- zdrojové kódy programu pro mikroprocesor a instalace řídicího programu
- schéma a tištěný spoj zařízení ve formátu pro EAGLE 4.09r2 a pdf
- data sheet mikroprocesoru AT89C2051
- datasheet obvodu MAX232
- manuál sběrnice I2C
- manuál sběrnice 1-wire
- manuál k teplotnímu čidlu DS18B20
- demoverze programu EAGLE 4.09r2
- simulátor procesorů rodiny 8051
- příklady a popis komunikace s vybranými zařízeními