

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV RADIOELEKTRONIKY

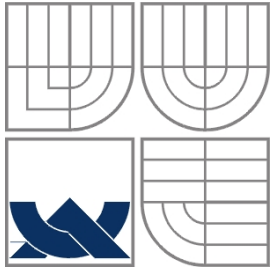
FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF RADIO ELECTRONICS

Využití sběrnice I2C pro komunikaci s externím zařízením

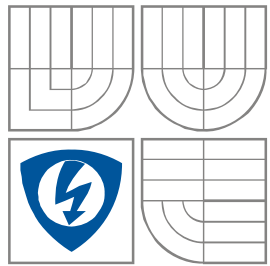
DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR
BRNO 2007

Bc. Jakub Prax



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A
KOMUNIKAČNÍCH
TECHNOLOGIÍ
ÚSTAV RADIOELEKTRONIKY

FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF RADIO ELECTRONICS

Využití sběrnice I2C pro komunikaci s externím zařízením

Communication with external devices by I2C bus

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. Jakub Prax

VEDOUCÍ PRÁCE

Ing. Tomáš Frýza Ph.D.

Prohlášení

Prohlašuji, že svou diplomovou práci na téma Využití sběrnice i2C pro komunikaci s externími zařízeními jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne 30. května 2008

.....
podpis autora

Poděkování

Děkuji vedoucímu diplomové práce Ing. Tomáši Frýzovi Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé diplomové práce.

V Brně dne 30. května 2008

.....
podpis autora

Abstrakt

Diplomová práce se zabývá sběrnicí I2C a přípravkem, který má sloužit pro demonstraci v předmětu Mikroprocesorová technika vyučovaném na ústavu Radioelektroniky.

Sběrnice I2C je výhodná tam kde potřebujeme realizovat přenos dat na krátkou vzdálenost s omezenými prostorovými možnostmi, ale nepožadujeme vysokou rychlost přenosu dat.

Přípravek je sestaven tak aby bylo možno realizovat na přípravku i jiné úlohy než jenom úlohy zabývající se sběrnicí I2C. V přípravku je sběrnice využita pro přenos data, času, teploty, dat pro D/A převodník, dat z a do paměti EEPROM a dat do 16-ti bitového expandéru.

Abstract

This diploma paper deals with the data bus line I2C and a supportive demonstration tool for the subject Microprocessor technology taught at the Radioelectronics faculty.

The data bus line I2C is convenient when short-distance data transmission with restricted spatial possibilities is required but high velocity of data transmission is not necessary.

The tool is compiled in such a way that enables it to also perform other tasks apart from those dealing with the I2C data bus line. The data bus line in the tool is utilized for transmission of date, time, temperature, data for the D/A converter, data from and to the EEPROM memory and data to 16 bits expander.

Klíčová slova

I2C, sběrnice, Atmel, AVR, ATtiny26, ATmega16, SDA, SCL, RTC, DS1337, DS1621, D/A, MAX518, MCP23016, SPI, ATtiny2313, TWI, podmínka START, podmínka STOP.

Key words

I2C, bus, Atmel, AVR, ATtiny26, ATmega16, SDA, SCL, RTC, DS1337, DS1621, D/A, MAX518, MCP23016, SPI, ATtiny2313, TWI, START condition, STOP condition.

Bibliografická citace mé práce:

PRAX, J. Využití sběrnice I2C pro komunikaci s externím zařízením. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2008. 47 s. Vedoucí diplomové práce Ing. Tomáš Frýza, Ph.D.

Obsah

Obsah.....	1
Seznam obrázků	2
Seznam tabulek	3
Úvod.....	4
1 Sběrnice I ² C.....	5
1.1 Vlastnosti popisované sběrnice I2C	5
1.2 Možnosti využití sběrnice I2C	5
1.3 Koncept sběrnice I2C	5
1.4 Celkový pohled na I2C.....	6
1.5 Přenos bitu.....	8
1.5.1 Parametry signálů (pro správné dekodování na přijímací straně).....	8
1.5.2 START a STOP podmínky	8
1.6. Adresování	9
1.7 Přenášená data.....	10
1.7.1 Formát Byte.....	10
1.7.2 Potvrzující bit ACK (acknowledge).....	10
1.8 Arbitrace a generace hodinového signálu	11
1.8.1 Synchronizace	11
1.8.2 Arbitrace.....	11
1.9 Možné způsoby komunikace po sběrnici I2C	12
2. Návrh přípravku	14
2.1 Požadavky	14
2.2 Ovládací procesor.....	14
2.3 Možnosti programování procesoru ATmega16.....	15
2.3.1 JTAG programátor	15
2.3.2 SPI programátor	16
2.4 Periferie pro sběrnici I2C	16
2.4.1 Sériová EEPROM 24C64.....	17
2.4.2 Klávesnice s procesorem ATtiny26	17
2.4.3 Obvod reálného času DS1337	18
2.4.4 Teplotní čidlo DS1621	20
2.4.5 2-kanálový 8-mi bitový digitálně-analogový převodník MAX518.....	20
2.4.6 16-ti bitový expandér MCP23016.....	21
2.5 Komunikace po sběrnici I2C (SW pohled)	23
2.5.1 Používání SFR registrů pro TWI procesoru ATmega16.....	24
2.5.2 Komunikace s pamětí 24C64P	26
2.5.3 Komunikace s obvodem reálného času DS1337	29
2.5.4 Komunikace s teplotním čidlem DS1621.....	30
2.5.5 Komunikace s expandérem MCP23016.....	31
2.6 Návrh obvodového zapojení a návrh plošného spoje.....	31
3. SW popis přípravku.....	31
3.1 Struktura menu	31
3.2 Klávesnice ATtiny26.....	33
3.3 Ovládací procesor ATmega16.....	35
3.3.1 Řízení displeje LCD	36
3.3.2 Příjem dat z klávesnice.....	36
3.3.3 Ovládání sběrnice I2C	36
4. Laboratorní úloha	41
4.1 Cíle laboratorní úlohy:	41

4.2	Zadání:.....	41
4.3	Teoretické poznatky:.....	41
4.4	Pokyny k zadání	43
4.5	Shrnutí	45
	Závěr.....	46
	Seznam použité literatury	47

Seznam obrázků

Obr. 2	Ukázka možnosti připojení dvou mikrokontrolerů na jednu sběrnici. (převzato z [1])..	7
Obr. 3	Připojení zařízení ve standardním a fast módu na sběrnici I2C (převzato z [1]).....	7
Obr. 4	Průběh dat po sběrnici (změna dat při nízké úrovni hodinového signálu) (převzato z [1]).....	8
Obr. 5	START(S) a STOP(P) podmínka (převzato z [1]).....	8
Obr. 6	Přenos dat na sběrnici I2C (převzato z [1]).....	9
Obr. 7	Potvrzující bit (Acknowledge) na sběrnici I2C (převzato z [1]).....	10
Obr. 8	Synchronizace hodinového signálu během arbitračního procesu (převzato z [1])	11
Obr. 9	Arbitrace na sběrnici mezi dvěma řídicími (převzato z [1])	12
Obr. 10	Kompletní přenos dat po sběrnici I2C (převzato z [1])	12
Obr. 11	Master-vysílač adresuje slave-přijímač 7-bitovou adresou. Směr přenosu dat se nemění. (převzato z [1])	13
Obr. 12	Master čte ze slave okamžitě po prvním bytu. (převzato z [1]).....	13
Obr. 13	Kombinovaný formát komunikace (převzato z [1]).....	14
Obr. 14	JTAG programátor - schéma.....	15
Obr. 15	SPI programátor - schéma.....	16
Obr. 16	Blokové schéma paměti 24C64 (převzato z [4]).....	17
Obr. 17	Zapojení procesoru ATtiny26 a tlačítek realizujících klávesnici.....	18
Obr. 18	Zapojení napájení obvodu DS1337 pomocí záložní baterie	19
Obr. 19	Blokové zapojení obvodu reálného času DS1337(převzato z [6]).....	19
Obr. 20	Blokové schéma teplotního čidla DS1621(převzato z [7])	20
Obr. 21	Blokové schéma D/A převodníku MAX518 (převzato z [8]).....	21
Obr. 22	Blokové schéma expandéru MCP23016 (převzato z [9])	22
Obr. 23	Zapojení obvodu MCP23016	22
Obr. 24	Blokové zapojení procesoru ATmega16 (převzato z [10]).....	23
Obr. 25	Zápis do paměti 24C64 po bytu	27
Obr. 26	Uložení čísla 0x92 do paměti 24C64 na adresu 0x0000.....	27
Obr. 27	Zápis do paměti 24C64 po stránce (převzato z [4])	28
Obr. 28	Čtení paměti podle hodnoty adresního registru (převzato z [4])	28
Obr. 29	Čtení z paměti, z určeného místa (převzato z [4])	28
Obr. 30	Zápis dat do obvodu reálného času DS1337 (převzato z [6])	30
Obr. 31	Čtení dat obvodu reálného času DS1337 (převzato z [6])	30
Obr. 32	Vývojový diagram aplikace v procesoru klávesnice.....	33
Obr. 33	Vývojový diagram aplikace v ovládacím procesoru.....	35
Obr. 34	fyzická realizace sběrnice I2C (převzato z [1])	41
Obr.35	START(S) a STOP(P) podmínka (převzato z [1]).....	42
Obr. 36	Master-vysílač adresuje slave-přijímač 7-bitovou adresou. Směr přenosu dat se nemění. (převzato z [1])	42
Obr. 37	Master čte ze slave okamžitě po prvním bytu. (převzato z [1]).....	43
Obr. 38	Kombinovaný formát komunikace (převzato z [1]).....	43

Seznam tabulek

Tab. 1 Definice terminologie na sběrnici I2C	7
Tab. 2: Tabulka použitých periférií pro sběrnici I2C	16
Tab. 3 Předdělička přenosové rychlosti sběrnice I2C	25
Tab. 4 Schéma paměťových buněk v obvodu DS1337	29
Tab. 5 Příkazy pro teplotní čidlo DS1621	31
Tab. 6 Závislost výstupního čísla na změřené teplotě	44

Úvod

Tento text má sloužit k pochopení přípravku v předmětu Mikroprocesorová technika. Úkolem tohoto textu je popsat přípravek pro ověření principů a pochodů na sběrnici I²C. Přípravek bude sloužit jako učební pomůcka pro studenty druhého ročníku bakalářského studijního programu na Fakultě elektrotechniky a komunikačních technologií.

Předložená práce obsahuje popis sběrnice I²C. V jednotlivých kapitolách je popsána realizace adresace na sběrnici I²C, následně je zde popsán průběh přenosu dat po této sběrnici, rozebrána je i funkce potvrzujícího bitu, synchronizace a arbitrace na sběrnici.

Dále se text věnuje samotnému návrhu přípravku. Nejdříve je zaměřena na periferie využití v tomto přípravku, které jsou popsány svými blokovými schémata. Text obsahuje popis programátoru ovládacího a klávesnicového procesoru. A v neposlední řadě je zaměřen na popis jednotlivých zařízení ze softwarového pohledu.

Součástí textu je i návrh laboratorní úlohy do předmětu Mikroprocesorová technika kde si studenti osvojí jednoduché pochody a práci se sběrnici I²C.

1 Sběrnice I²C

1.1 Vlastnosti popisované sběrnice I2C

Následující informace jsou platné pro verzi 2.0 z roku 1998. 7-bitové a 10-bitové adresování, tímto je možnost na jednu sběrnici připojit až 1024 různých adres, bohužel počet zařízení je omezen parazitní kapacitou linek a připojených zařízení do 400 pF. High-speed mód, umožňuje přenos dat rychlostí až 3,4Mbit/s, přitom na jedné sběrnici mohou být připojena zařízení podporující tuto rychlost, ale také zařízení pomalejší. V podstatě se přenosová rychlost může pohybovat v rozmezí 0 až 3,4Mbit/s. Sběrnice I2C využívá pouze dva vodiče, sériový datový (SDA) a sériový hodinový (SCL). Každé zařízení připojené na tuto sběrnici je softwarově adresovatelné jednou unikátní adresou pro celý systém a v každou chvíli existuje jednoduchý vztah master/slave (řídící/řízený obvod) obvod. Master může operovat jako vysílač či přijímač. Jde o pravou sběrnici typu multi-master se zabudovanou detekcí kolizí či více obvodů, které vysílají v jednom okamžiku. Sériová, 8-bitově orientovaná sběrnice pro obousměrnou komunikaci může pracovat rychlostí 100kbit/s ve Standart-mode, do 400kbit/s v Fast-mode nebo do 3,4Mbit/s ve High-speed mode. Počet zařízení připojitelných na sběrnici je omezen maximální kapacitancí sběrnice do 400 pF a počtem využitelných adres.

1.2 Možnosti využití sběrnice I2C

S touto sběrnici se můžeme setkat v běžné domácí elektronice, v telekomunikacích, nebo i v průmyslu. Například, inteligentní řízení, využívající nejčastěji mikroprocesor, všestranně použitelné obvody např.: řadiče LCD, V/V (vstupně výstupní) porty, paměti nebo převodníky dat. Aplikačně orientované obvody, digitální tunery a signálové procesory pro radiové či televizní systémy nebo DTMF generátory pro telefony s tónovou volbou. Příklad použití je zdokumentován na Obr. 1.

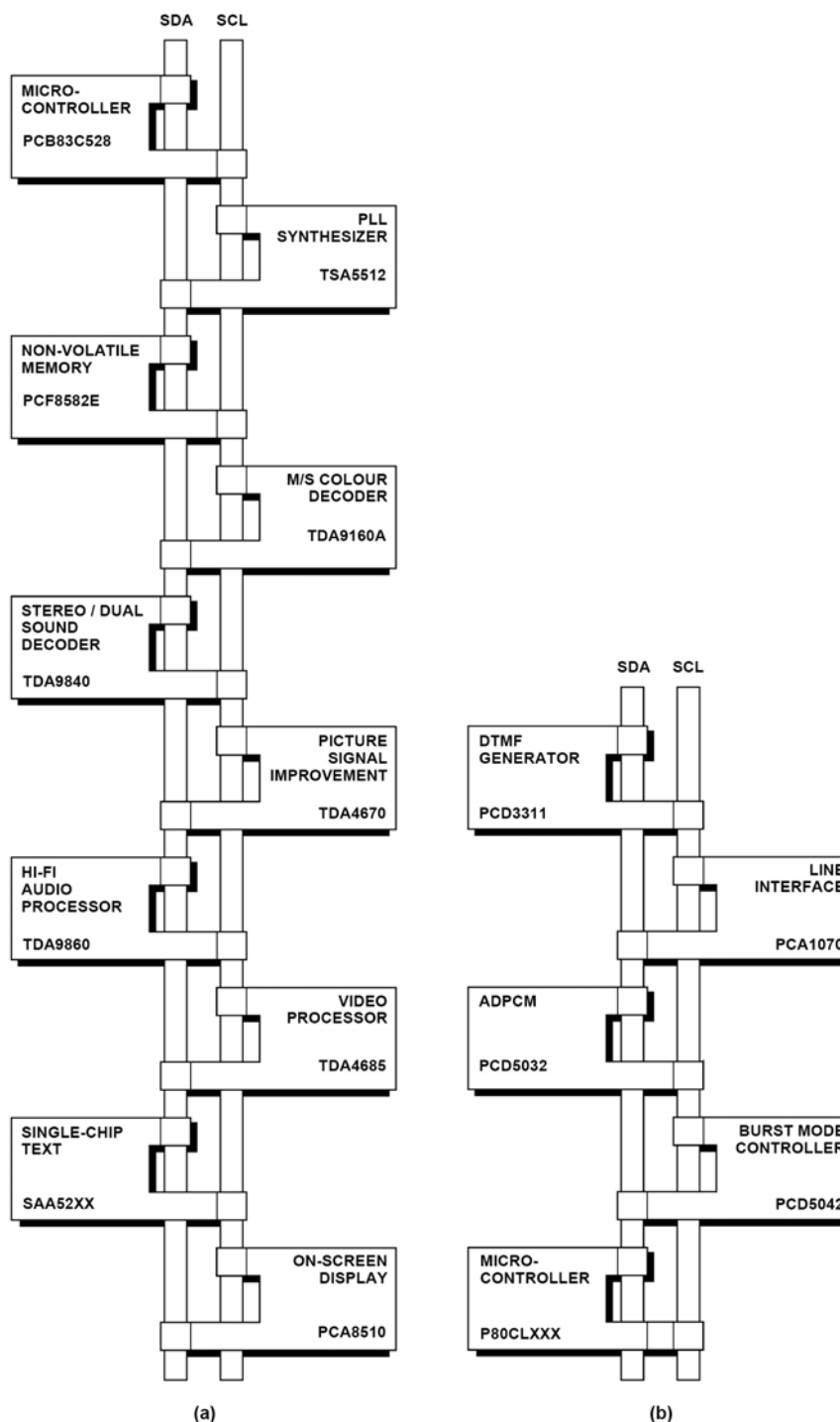
1.3 Koncept sběrnice I2C

Na sběrnici I2C je možnost připojit součástky vyrobené různými výrobními procesy (NMOS, CMOS, bipolární). Dva vodiče, sériová data (SDA) a sériové hodiny (SCL), přenášejí informace mezi zařízeními připojenými na sběrnici. Každé zařízení je identifikovatelné jedinečnou adresou a může pracovat buď jako vysílač nebo jako přijímač, záleží na funkci zařízení. Zřejmě LCD řadič je především použit jako přijímač i když musí potvrzovat správné přijetí dat, kdežto paměť může vysílat i přijímat data. Během připojení vysílače a přijímače mohou být zařízení chápána jako master a slave během přenosu dat (viz Tab.1). Masterem se stává obvod jestliže zahajuje přenos dat po sběrnici a generuje hodinový signál. V tomto čase jsou chápána ostatní zařízení jako slave.

Sběrnice I2C je typu multi-master. To znamená, že více jak jedno zařízení může převzít řízení sběrnice a zároveň k ní může být připojeno. Jako master se běžně používají mikrokontrolery, které připouští možnost posílání dat mezi sebou (viz Obr. 2).

1.4 Celkový pohled na I2C

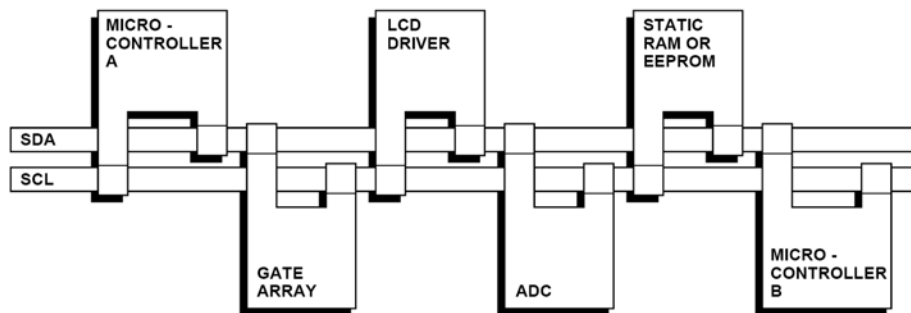
Obě linky SDA i SCL jsou obousměrné, připojené ke kladnému napájecímu napětí přes proudový zdroj nebo zdvihací rezistory (viz Obr. 3). Pokud je sběrnice volná, obě linky jsou na vysoké úrovni. Ve výstupním stupni je použito zapojení s otevřeným kolektorem.



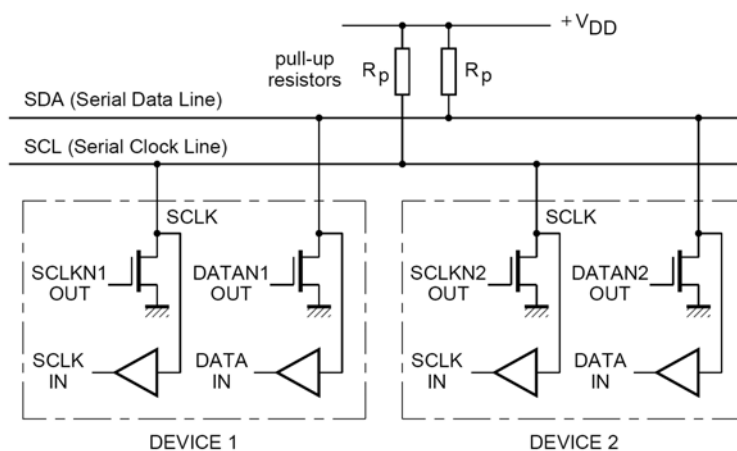
Obr. 1 Dva příklady aplikace sběrnice I2C (a) vysoce výkonný televizní přijímač s vysokým stupněm integrace, (b) bezdrátový telefon základní stanice. (převzato z [1])

Tab. 1 Definice terminologie na sběrnici I2C

Termín	Popis
Vysílač	Zařízení, které vysílá data na sběrnici
Příjímač	Zařízení, které přijímá data ze sběrnice
Master (řídící)	Zařízení, které zahajuje přenos, generuje hodinový signál a ukončuje přenos dat
Slave (řízený)	Zařízení adresované řídícím
Multi-Master (více řídících na jedné sběrnici)	Více jak jeden řídící se může pokusit řídit sběrnici ve stejném čase bez ztracení přenášených informací
Arbitration (rozhodčí řízení)	Postup k zajištění, aby v případě více jak jednoho řídícího, kteří se snaží současně řídit sběrnici, nedošlo ke ztrátě informace na sběrnici. Jen jednomu je to povoleno a vítězná přenášená informace není poškozena
Synchronizace	Postup pro synchronizaci hodinového signálu ze dvou či více zařízení.



Obr. 2 Ukázka možnosti připojení dvou mikrokontrolerů na jednu sběrnici. (převzato z [1])



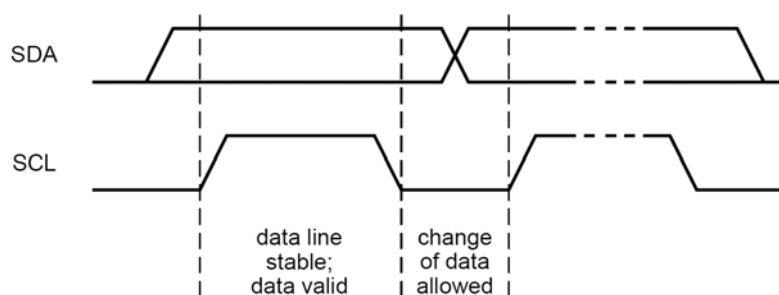
Obr. 3 Připojení zařízení ve standardním a fast módu na sběrnici I2C (převzato z [1])

1.5 Přenos bitu

Díky rozdílným technologiím (CMOS, NMOS, bipolar), které jsou na sběrnici připojitelné, nejsou pro ně pevná napěťová specifikace pro log. 0 a log. 1. Jeden hodinový puls je generovaný pro každý datový bit.

1.5.1 Parametry signálů (pro správné dekódování na přijímací straně)

Data na lince SDA musí být stabilní během vysoké úrovně (log. 1) hodinového signálu. Při přenosu dat je možné měnit hodnotu na lince SDA jen a právě když je hodinový signál SCL na nízké úrovni (viz Obr. 4). Jiná situace je při začátku a ukončování komunikace, při vysílání podmínky START a STOP, zde se mění data na lince SDA během vysoké úrovně na lince SCL.

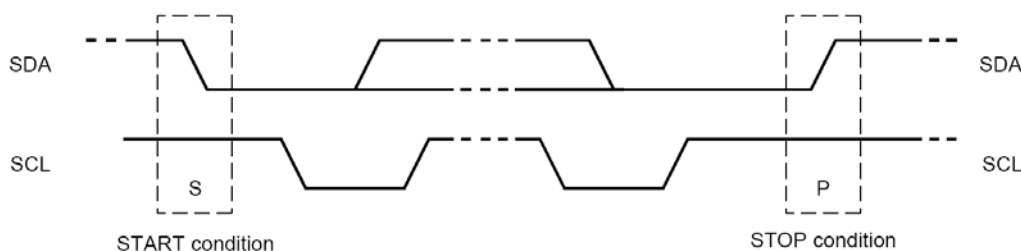


Obr. 4 Průběh dat po sběrnici (změna dat při nízké úrovni hodinového signálu) (převzato z [1])

1.5.2 START a STOP podmínky

Komunikace po sběrnici I²C začíná START(S) podmínkou a končí podmínkou STOP (P) (viz Obr. 5).

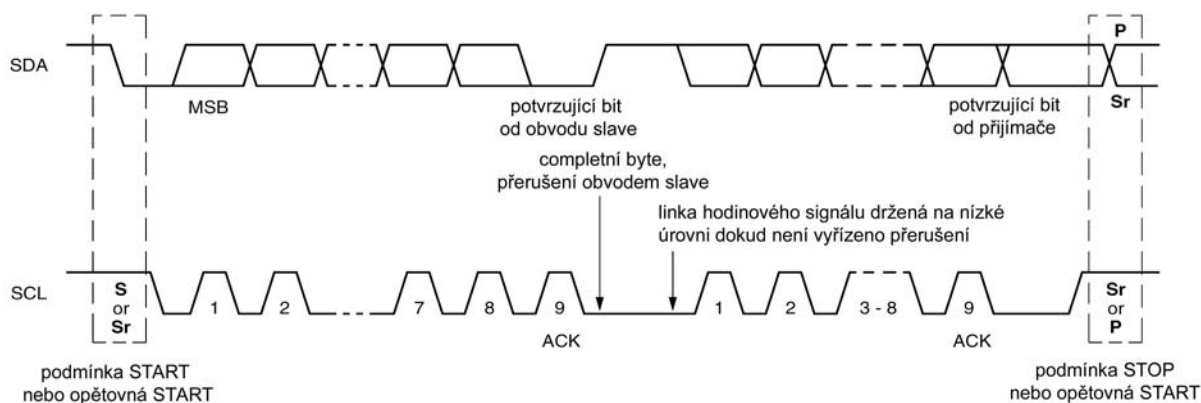
Změna datového signálu SDA z log. 1 na log. 0 pokud je hodinový signál SCL na úrovni log. 1 specifikuje START (S) podmínku. Naopak změna signálu na datovém vodiči SDA z log. 0 na log. 1 a hodinového signálu na vysoké úrovni log. 1 definuje STOP (P) podmínku.



Obr. 5 START(S) a STOP(P) podmínka (převzato z [1])

START i STOP podmínky jsou generovány pouze řídicím obvodem sběrnice I2C. Sběrnice je považována za zaneprázdněnou po příchodu podmínky start. Sběrnice je považována za volnou po příchodu podmínky STOP a určitém čase na zotavení. Sběrnice zůstává zaneprázdněná i v případě, že místo STOP podmínky přijde podmínka opakovaná START (Sr) (repeated). S ohledem na podmínku START je podmínka opakovaná START ekvivalentní prosté podmínce START (viz Obr. 6).

Detekce START a STOP podmínky je pro zařízení připojená na sběrnici jednoduchá jestliže jsou vybavená hardwarovou podporou sběrnice I2C. Nicméně i pokud zařízení není vybaveno hardwarovou podporou sběrnice I2C stačí vzorkování datového vodiče 2x za periodu hodinového signálu ke zjištění změny datového signálu.



Obr. 6 Přenos dat na sběrnici I2C (převzato z [1])

1.6. Adresování

Každá stanice (v praxi např. každý integrovaný obvod) připojená na sběrnici I2C má přidělenou 7bitovou adresu. Po zachycení podmínky START porovnávají všechny obvody svou adresu s adresou, která je vysílána na sběrnici. Zjistí-li některý z obvodů shodu, je vysílání určeno právě jemu a musí přijetí adresy potvrdit bitem ACK. Potom přijímá nebo vysílá další data.

V praxi je většinou několik adresních bitů jednotlivých obvodů určeno už při výrobě. Zbývající bity se volí pomocí příslušných vývodů daného obvodu, které se podle potřeby připojí na úroveň L nebo H. Několik adres je na I2C vyhrazeno pro speciální účely. Např. adresa 0000000 je určena pro vysílání „broadcast“, adresy 0000011, 00001XX a 11111XX jsou rezervovány pro další účely.

Adresa 11110AA indikuje 10bitové adresování: "AA" zde označuje dva nejvyšší bity adresy stanice, zbývajících 8 bitů je vysíláno v následujícím bytu.

Poznámka: Popisované adresování se týká adresy jednotlivých stanic na I2C. Je-li stanicí např. paměťový obvod, jedná se o adresu celého obvodu. Adresa jednotlivých slov v paměti a požadovaná operace se do paměti přenáší v datové části rámce I2C.

1.7 Přenášená data

1.7.1 Formát Byte

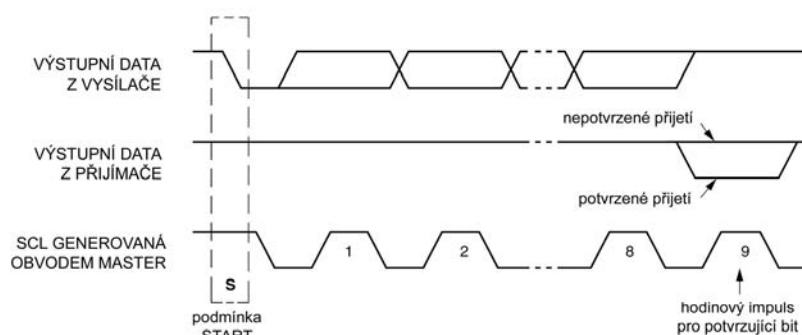
Každý byte přenášený po sběrnici I2C musí mít délku 8 bitů. Počet bytů, které je možné během jednoho přenosu přenést není omezený. Každý byte musí být následován potvrzujícím bitem (acknowledge). Data jsou přenášena s nejdůležitějším bitem jako prvním (viz Obr. 6).

1.7.2 Potvrzující bit ACK (acknowledge)

Každý přenesený byte musí následovat potvrzující bit. Hodinový signál pro potvrzující bit je generován řídicím obvodem. Vysílač uvolní datovou linku SDA při očekávání potvrzovacího bitu. Vysílač ponechá datový vodič SDA během očekávání potvrzujícího bitu ve vysoké úrovni. Příjímač musí během hodinového pulsu pro potvrzující bit stáhnout datovou linku na nízkou úroveň tak aby byla zachována pravidla pro správný přenos bitu (viz Obr. 7).

Pokud slave nepotvrdí příjem adresy pro něho určenou (například není schopný přijímat či vysílat protože je zaneprázdněn nějakou funkcí v reálném čase), datová linka musí být ponechána ve vysoké úrovni. Master může poté generovat podmínku STOP aby ukončil komunikaci s tímto zařízením a nebo podmínku opětovného START pro novou komunikaci.

Pokud slave-příjímač potvrdí svoji adresu, ale někdy později není schopný přijímat další data, master musí znovu přerušit komunikaci. Toto je indikováno nevysláním potvrzujícího bitu příjímačem. Slave opouští datovou linku ve vysoké úrovni a řídicí generuje STOP nebo opětovnou START podmínku.



Obr. 7 Potvrzující bit (Acknowledge) na sběrnici I2C (převzato z [1])

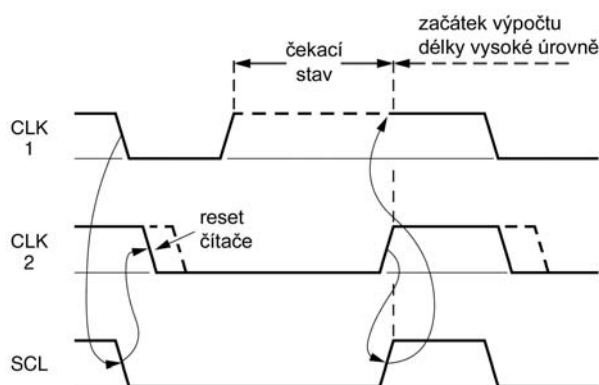
Pokud master-příjímač účastní se přenosu, musí ukončit tok dat směrem k slave bez generování potvrzujícího bitu, musí slave opustit sběrnici pro povolení generace podmínky STOP nebo opětovné START řídicím.

1.8 Arbitrace a generace hodinového signálu

1.8.1 Synchronizace

Všechny obvody typu master generují svůj vlastní hodinový signál na vodiči SCL kvůli přenesení požadované informace po sběrnici I²C. Data jsou platná pouze v čase vysoké úrovně hodinového signálu. Definovaný hodinový signál je tedy potřeba aby bylo možné arbitrovat přístup na sběrnici bit-od-bitu.

Synchronizace hodinového signálu je provedena, použitím připojení na linku hodinového signálu ve formě hradla AND. To znamená, že přechod z vysoké úrovně na nízkou na vodiči SCL může v zásadě způsobit jakékoliv zařízení generující hodinový signál, ale přechod z nízké na vysokou úroveň může nastat pouze v případě vysoké úrovně všech zařízení generujících hodinový signál na linku SCL (viz. Obr. 8). Jestliže jsou připojena na sběrnici zařízení s kratší periodou a zároveň delší periodou zařízení s kratší periodou přejdou ještě před příchodem hodinového impulsu do čekacího stavu.



Obr. 8 Synchronizace hodinového signálu během arbitračního procesu (převzato z [1])

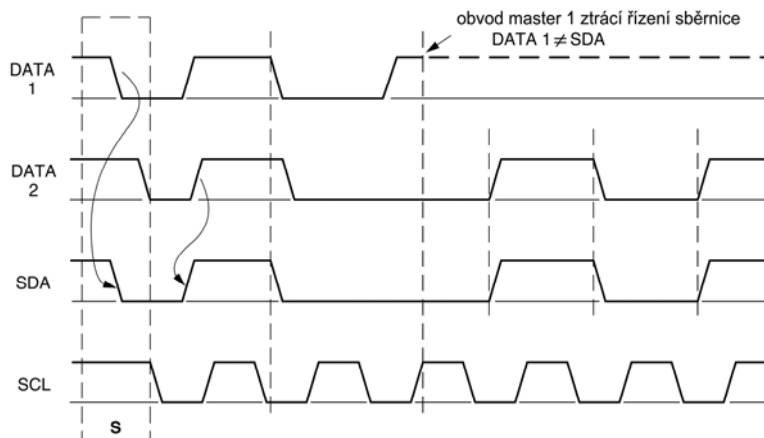
Pokud všechna zúčastněná zařízení mají hotový proces probíhající během nízké úrovně hodinového signálu, uvolní linku SCL a tato přejde do vysoké úrovně. V tuto chvíli není rozdíl mezi hodinovým stavem jednotlivých zařízení a stavem na lince SCL a všechna zařízení mohou pokračovat procesem při vysoké úrovni hodinového signálu. První zařízení, které dokončí svůj proces opět stáhne linku SCL na nízkou úroveň.

V tomto případě, synchronizovaného hodinového signálu je tento signál generován s délkou nízké úrovně podle nejdelšího a vysoká úroveň je řízena nejkratším možným průběhem.

1.8.2 Arbitrace

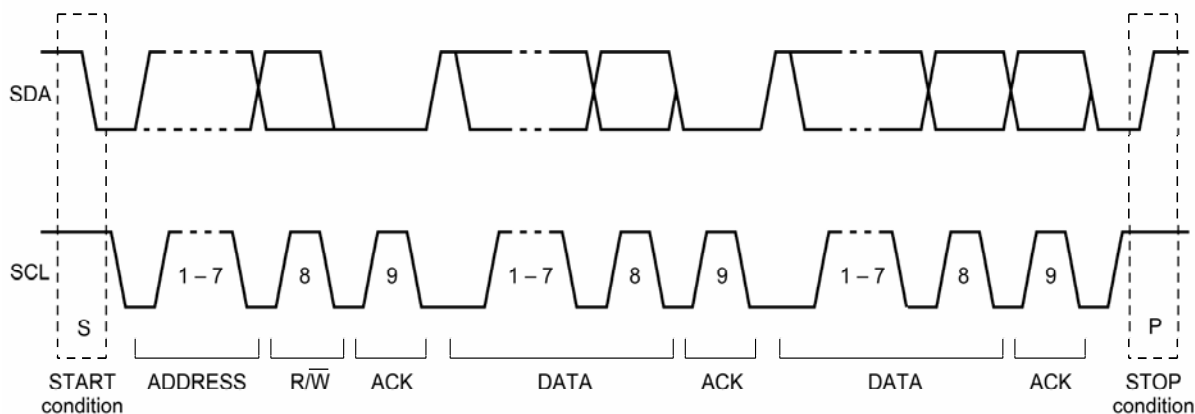
Pro arbitraci se na I²C používá metoda s detekcí kolize. Každý z obvodů typu master může zahájit vysílání, je-li předtím sběrnice v klidovém stavu. Během vysílání musí neustále porovnávat vysílané bity se skutečným stavem SDA. Je-li zjištěn rozdíl mezi očekávaným a skutečným stavem linky SDA, je to indikace kolize mezi několika stanicemi. Vzhledem k charakteru sběrnice (otevřené kolektory) může k této situaci dojít, pokud jeden master vysílá vysokou úroveň, zatímco druhý master vysílá nízkou úroveň. Stanice, která na lince zjistí

nízkou úroveň zatímco sama vysílá vysokou úroveň musí vysílání okamžitě ukončit (viz. Obr.9).



Obr. 9 Arbitrace na sběrnici mezi dvěma řídicími (převzato z [1])

K arbitraci většinou dochází během vyslání několika prvních bitů, kdy je vysílána adresa přijímací stanice. Pokud by se např. dvě stanice současně pokusily o zápis do stejného obvodu, nastane kolize až při přenosu vlastních zapisovaných dat. V krajním případě, kdy několik stanic současně zapisuje stejná data na stejnou adresu, nemusí být kolize vůbec detekována. Z obrázku (Obr. 10) je patrný celý průběh komunikace po sběrnici I2C.

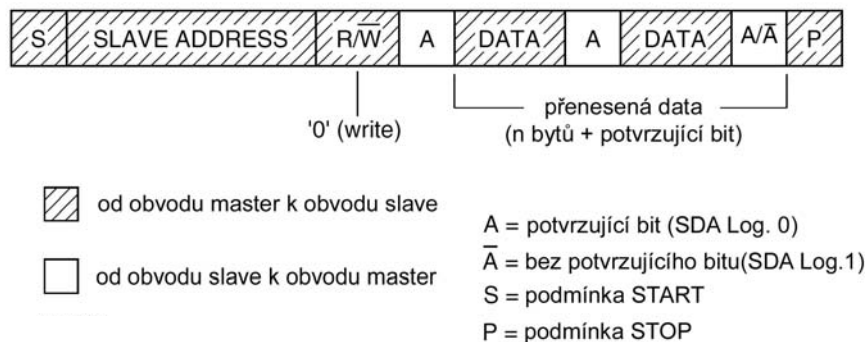


Obr. 10 Kompletní přenos dat po sběrnici I2C (převzato z [1])

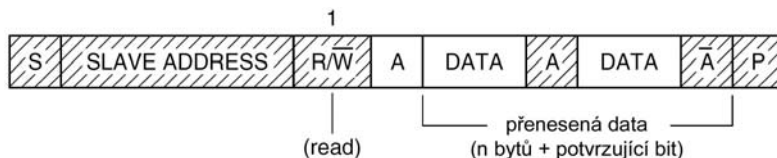
1.9 Možné způsoby komunikace po sběrnici I2C

Adresace obvodu typu slave obvodem typu master a zápis do něj (viz. Obr.11). Z grafiky je patrný přenos veškerých dat od adresy až po pakety dat od masteru k slave, včetně potvrzujícího bitu ACK, kterým slave dává najevo správný příjem dat.

Adresace obvodu typu slave obvodem typu master a čtení dat z něj (viz. Obr.12). Na obrázku lze vidět přenos adresy z masteru do slave a následné potvrzení správného přijetí adresy obvodem slave, poté vyslání paketu dat obvodem slave a potvrzení správného příjmu obvodem typu master. Následně se přenáší ještě jeden byte, který už ale není obvodem typu master potvrzen a komunikace je ukončena podmínkou STOP.

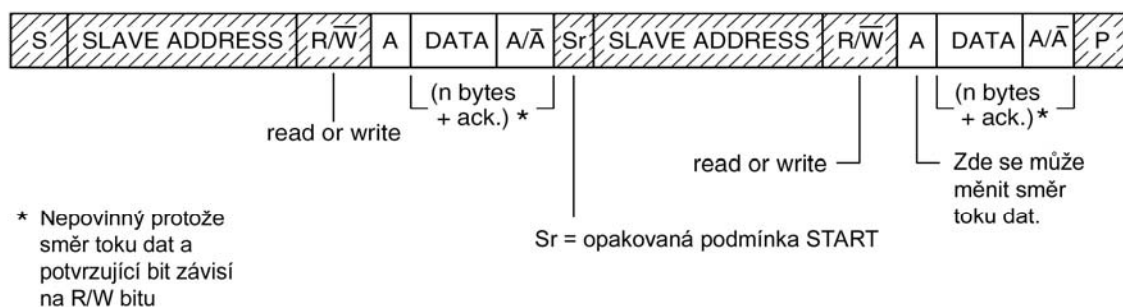


Obr. 11 Master-vysílač adresuje slave-přijímač 7-bitovou adresou. Směr přenosu dat se nemění. (převzato z [1])



Obr. 12 Master čte ze slave okamžitě po prvním bytu. (převzato z [1])

Na obrázku (Obr.13) jde o kombinovaný přenos dat, nejdříve se adresuje obvod, čtou se z něj data, vysílá se opětovná podmínka START (Sr), adresuje se další obvod typu Slave a čtou se z něj další data, vše je zakončeno podmínkou STOP. Tento způsob se využívá například u paměti kde je potřeba nejdříve zapsat z jaké adresy chceme číst a poté vyčíst data.



Obr. 13 Kombinovaný formát komunikace (převzato z [1])

2. Návrh přípravku

2.1 Požadavky

Od přípravku je očekávána názornost, možnost sledování průběhů na sběrnici I2C a možnost ovládání různých obvodů.

2.2 Ovládací procesor

Jako ovládací procesor je použit procesor AVR od firmy ATMEL typ ATmega16. Tato volba byla jasná vzhledem k tomu, že se tento procesor používá při výuce předmětu BMPT ve třetím ročníku bakalářského studijního programu.

Vnitřní periferie procesoru ATmega16

- Dva 8-mi bitové časovače/čítače s oddělenými předděličkami a komparačními módy
- Jeden 16-ti bitový časovač/čítač a oddělenou předděličkou, komparačním módem a snímacím (capture) módem
- Obvod reálného času s odděleným oscilátorem
- Čtyři PWM kanály
- 8-mi kanálový 10-ti bitový A/D převodník
- Bytově orientované Two-wire sériové rozhraní
- Programovatelný sériový USART
- Master/Slave SPI sériové rozhraní
- Programovatelný Watchdog časovač s odděleným oscilátorem na čipu
- Analogový komparátor na čipu

2.3 Možnosti programování procesoru ATmega16

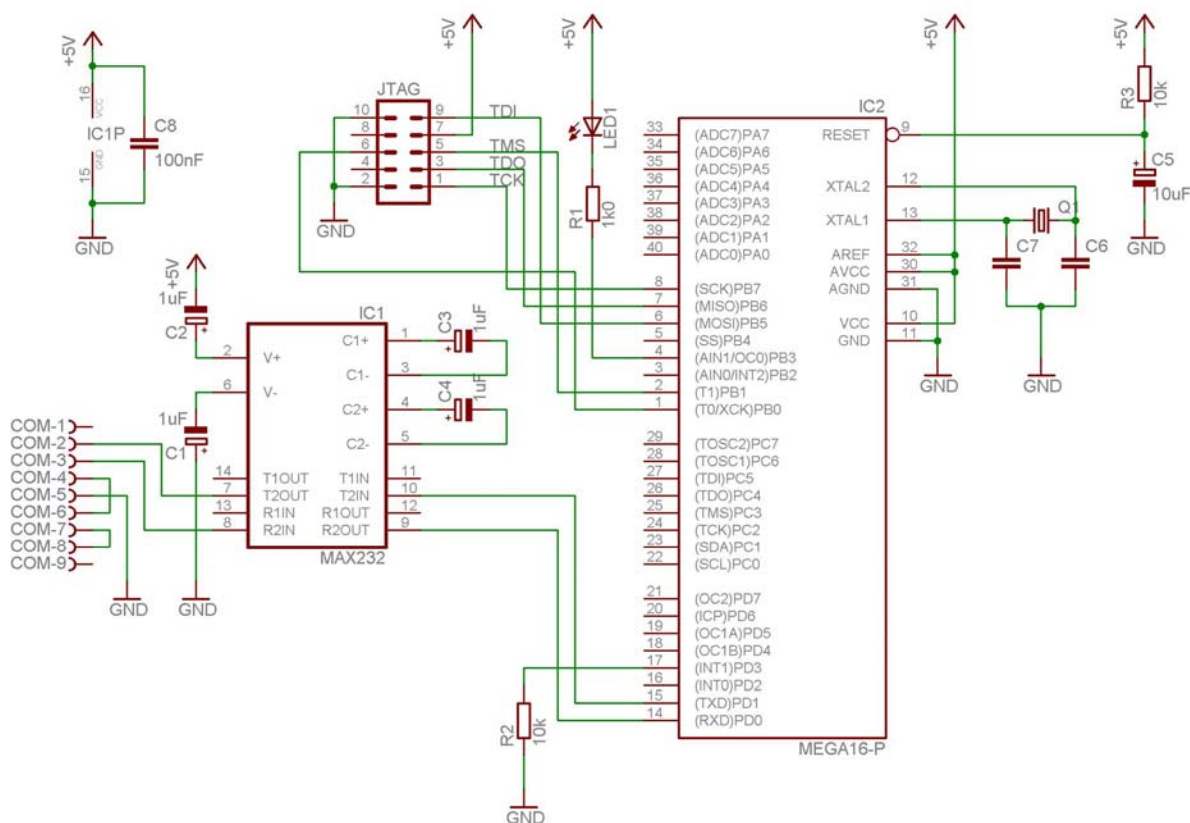
- JTAG programátor
- SPI programátor

2.3.1 JTAG programátor

Sestavený pomocí procesoru ATmega16 (programátor + ovládací procesor = 2x procesor ATmega16). Do programovacího procesoru se nahraje pouze tzv: BootLoader, který zajistí i aktualizace ovládacího Firmware přes AVR Studio.

Výhodou tohoto zapojení (viz Obr. 14) je možnost krokování programu přímo na čipu, samozřejmostí je programování paměti FLASH, EEPROM, pojistek bezpečnostních bitů.

Nevýhodou je nutnost použití velkého procesoru v podobě ATmega16 a nemožnost využít piny využité pro programování. A v neposlední řadě, vzhledem k využití dalšího procesoru k ovládání klávesnice a nutnosti jeho programování, tento procesor (ATtiny26) neobsahuje podporu JTAG. Vzhledem k těmto faktům bylo použití zavrhnuto.



Obr. 14 JTAG programátor - schéma

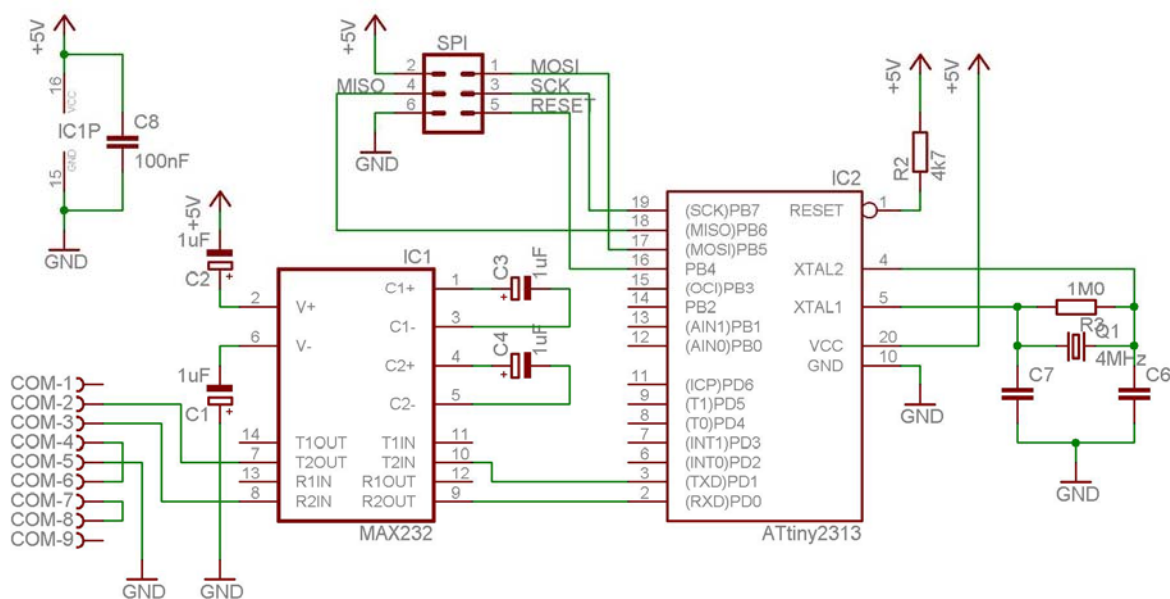
2.3.2 SPI programátor

Využívá zapojení přímo od firmy Atmel (viz Obr. 15), která pro převod úrovní využívala zapojení s tranzistory, toto bylo nahrazeno jednoduchým levným obvodem MAX232. Procesor ATtiny2313 se musí programovat softwarem, který zajistí rozhraní mezi klasický sériovým portem a SPI.

Výhodou tohoto zapojení je možnost použití pinů, které se využívají na programování i pro vyvíjené aplikace. Programování přes rozhraní SPI má také větší možnost využití, protože s tímto umí pracovat více procesorů, než s programátorem JTAG.

Nevýhodou zůstává nemožnost krokovat a ladit aplikaci přímo na čipu.

Tuto nevýhodu vyvážila možnost programovat více procesorů a připojit zařízení i na programovací piny. Proto byl zvolen programátor SPI.



Obr. 15 SPI programátor - schéma

2.4 Periferie pro sběrnici I²C

V tabulce Tab. 2 jsou uvedeny periferie použité pro komunikaci po sběrnici I²C.

Tab. 2: Tabulka použitých periférií pro sběrnici I²C

Obvod	Funkce	Pouzdro	Výrobce	Adresa I2C
24C64P	Sériová EEPROM 64k	DIL8	Microchip	0x50
DS1337	Obvod reálného času	DIL8	Dalas semiconductor / Maxim	0x68
DS1621	Teplotní senzor	DIL8	Dalas semiconductor / Maxim	0x48
MAX518	D/A převodník	DIL8	Dalas semiconductor / Maxim	0x2C
MCP23016	16-ti bitový expandér	DIL28	Microchip	0x20

V následujícím textu jsou popsány jednotlivé periferie podrobněji.

2.4.1 Sériová EEPROM 24C64

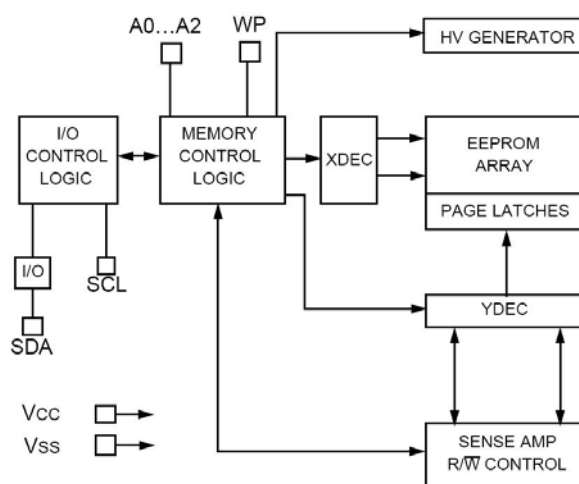
Paměť o velikosti 64kB sloužící pro stálé uchování dat, není napětově závislá. V paměti mohou být uloženy hodnoty pro generování průběhů pomocí D/A převodníku a nebo nějaké nastavovací konstanty.

Adresa pro tuto paměť je:

1	0	1	0	A2	A1	A0	R/W
---	---	---	---	----	----	----	-----

Bitsy A0-A2 jsou hardwarově volitelně připojením pinů na VCC nebo GND.

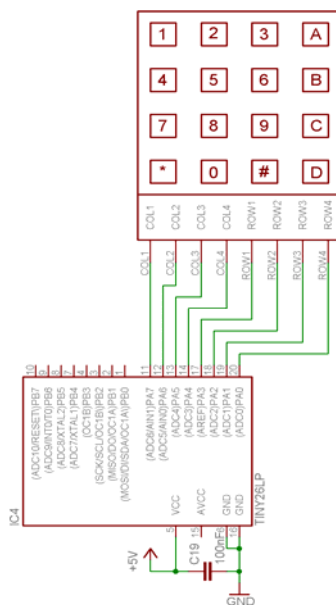
Blokové schéma vypadá podle Obr. 16. V tomto schématu jsou vidět vyvedené piny pro volbu adresy A0...A2. Dále nemůže chybět napájení a v neposlední řadě sběrnice I2C.



Obr. 16 Blokové schéma paměti 24C64 (převzato z [4])

2.4.2 Klávesnice s procesorem ATtiny26

Jedná se o procesor AVR firmy Atmel, ke kterému je připojena klávesnice 4x4 tlačítek. Tento procesor je možno připojit i na programátor zabudovaný na desce a tímto ho programovat. Obvod je programovaný tak aby po stisku klávesy generoval externí přerušení do ovládacího procesoru a vystavil 4 bitový ekvivalent stisknuté klávesy na svoji výstupní bránu přivedenou na vstupní bránu procesoru ATmega16. Po přijetí přerušení od procesoru ATtiny26 procesor ATmega16 přečte 4-bitovou hodnotu vystavenou na vstupních pinech. Obrázek Obr. 17 ukazuje připojení klávesnice k procesoru ATtiny26. Zbývající volné piny jsou využity pro programování a přenos dat směrem k ovládacímu procesoru. ATmega16 pro své taktování využívá vnitřní oscilátor. Pro tlačítka jsou použity vnitřní Pull-Up rezistory.



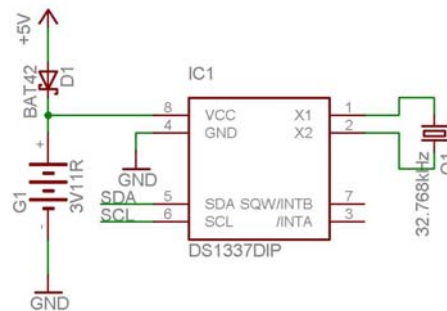
Obr. 17 Zapojení procesoru ATtiny26 a tlačítek realizujících klávesnici

Další procesor byl použit s výhledem na možnost komunikace po sběrnici I2C mezi dvěma procesory. Ovšem postupným vývojem jsem ověřil, že toto není možné vzhledem k tomu, že piny procesoru určené pro komunikaci po sběrnici I2C jsou zároveň určeny pro programování přes sběrnici SPI. Zřejmě by bylo možné naprogramovat oba procesory tak aby takto komunikovat dokázali, ale odpadla by možnost tyto procesory programovat přímo na přípravku, protože sběrnice I2C by kolidovala se sběrnici SPI.

Procesor ATtiny26 v přípravku zůstal, jelikož je jednodušší a jsou k němu připojeny LED diody a klávesnice tak je možnost na tomto přípravku navrhnout úlohy pro seznámení se s procesory AVR.

2.4.3 Obvod reálného času DS1337

Tento obvod počítá sekundy, minuty, hodiny, dny v týdnu, dny v měsíci, měsíce a stovky let (0-99). Tímto je obvod bez problémů funkční do roku 2100. Obvod ke své správné funkci potřebuje externí krystal 32,768kHz a napájení, proto je doplněn o baterii, která je doplněna o obvod, kterým se dobíjí při připojení napájecím napětí (viz. Obr. 18). Po odpojení napájecího napětí napájí tato baterie pouze obvod DS1337. Obvod taktovaný na maximální frekvenci má spotřebu 150μA, ale v provozu StandBy je to 1,5μA teoreticky obvod vydrží počítat čas na 110mAh akumulátor přibližně 8 let. Tato hodnota je skutečně teoretická.



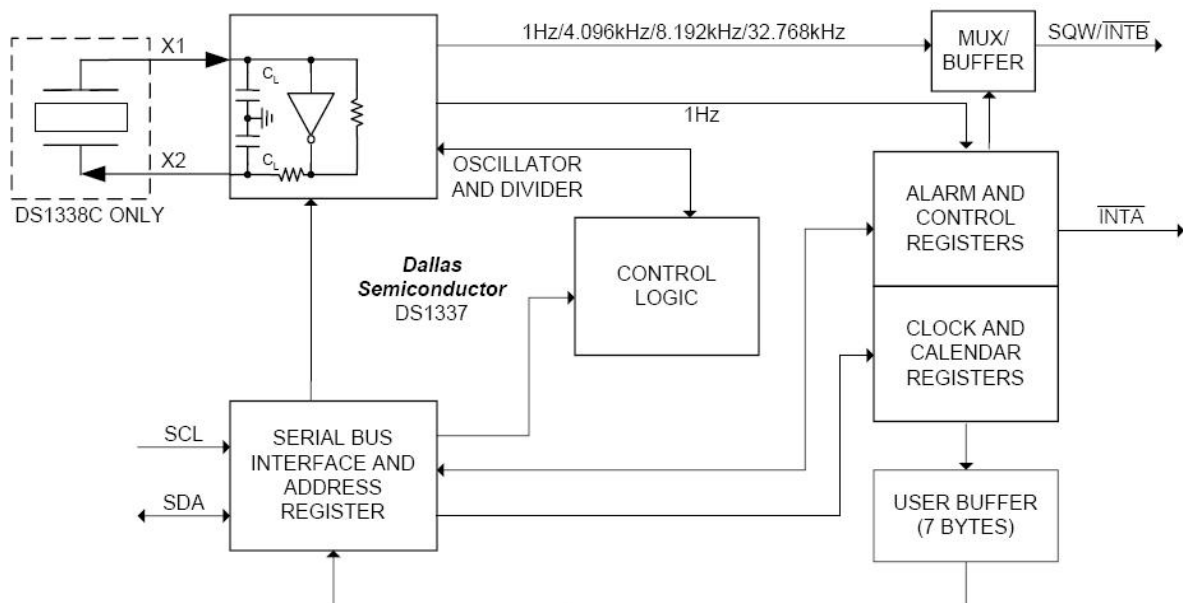
Obr. 18 Zapojení napájení obvodu DS1337 pomocí záložní baterie

Jak je patrné z Obr. 18 baterie je nabíjena konstantním napětím, které je z 5V snižené diodou BAT42 přibližně na 4,5V. Tato dioda také zajistí tok proudu z baterie pouze do obvodu reálného času.

Adresa obvodu DS1337 je:

1	1	0	1	0	0	0	R/W
---	---	---	---	---	---	---	-----

Na obrázku Obr. 19 je vidět blokové schéma obvodu reálného času DS1337. Na tomto schématu je vidět, že si obvod vnitřně generuje kmitočty od 1Hz až do taktovacího kmitočtu 32,768kHz a lze jej programovat i pro generování tohoto kmitočtu i na výstupní piny.



Obr. 19 Blokové zapojení obvodu reálného času DS1337(převzato z [6])

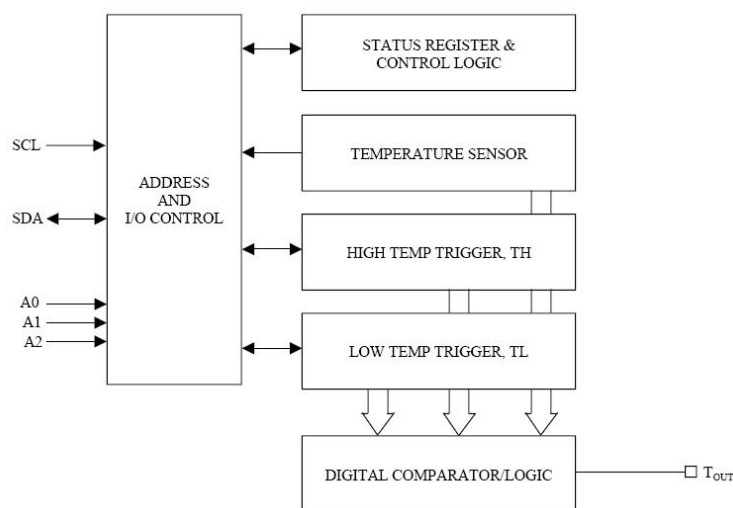
2.4.4 Teplotní čidlo DS1621

Obvod DS1621 snímá okolní teplotu a převádí ji na číslo, které lze z obvodu vyčíst přes sběrnici I2C.

Adresa tohoto obvodu je:

1	0	0	1	A2	A1	A0	R/W
---	---	---	---	----	----	----	-----

Bitsy A0-A2 jsou hardwarově volitelné. Na obrázku Obr. 20 je vidět blokové schéma, na tomto schématu je vidět jednoduchost tohoto obvodu. Obvod může sloužit jako termostat a nebo může vyvolat přerušení při určité teplotě. Tento obvod pracuje spolehlivě, nicméně při běžném provozu na sběrnici I2C se tímto sám ohřívá, po té měří vyšší teplotu. Při pokojové teplotě se zpravidla jedná o 3°C.



Obr. 20 Blokové schéma teplotního čidla DS1621(převzato z [7])

2.4.5 2-kanálový 8-mi bitový digitálně-analogový převodník MAX518

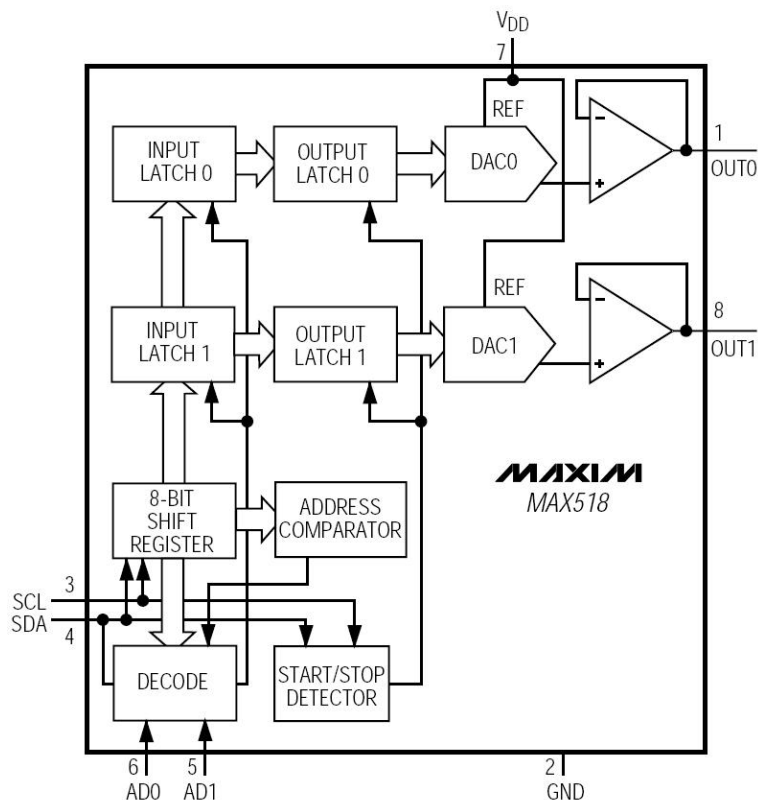
Obvod slouží k převodu čísla na napětí. Má 2 kanály takže lze generovat dvě různé napětí. Je 8-mi bitový, při referenčním napětí 5V je minimální napěťový krok cca 20mV.

Adresa tohoto obvodu je:

0	1	0	1	1	A1	A0	R/W
---	---	---	---	---	----	----	-----

Bitsy A0 a A1 jsou hardwarově volitelné.

Na obrázku Obr. 21 je blokové zapojení D/A převodníku MAX518. Ze schématu plyne i jeho činnost, po příchodu platných dat tyto převede na příslušnou velikost napětí. Těchto obvodů můžeme na sběrnici I2C připojit až čtyři.



Obr. 21 Blokové schéma D/A převodníku MAX518 (převzato z [8])

2.4.6 16-ti bitový expandér MCP23016

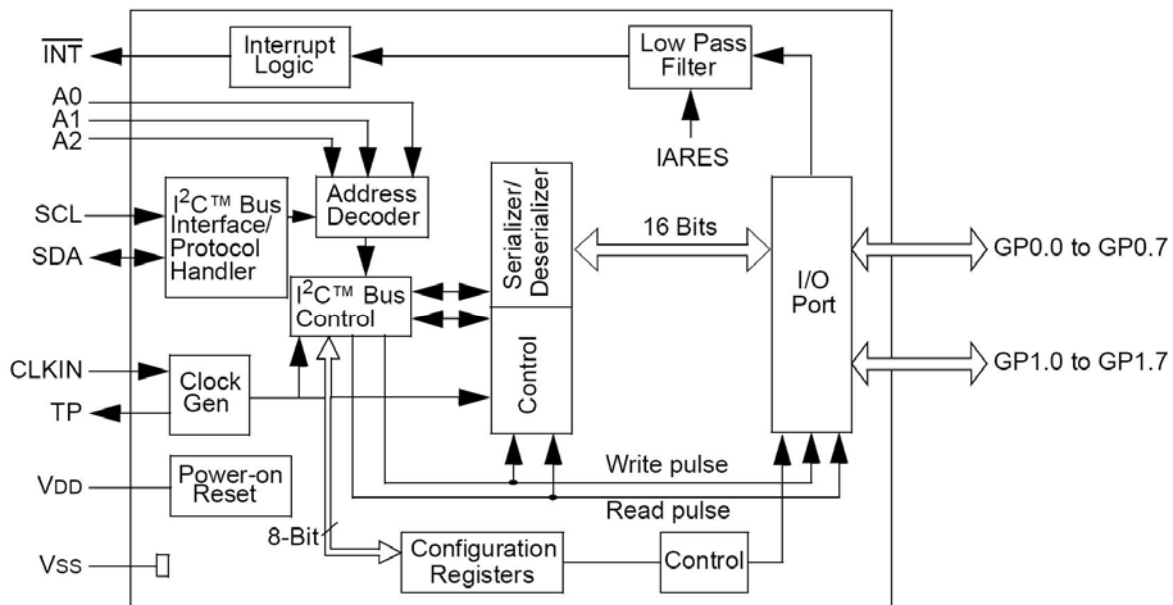
Tato součástka se využívá pro rozšíření sériového rozhraní na dva 8-bitové porty. Jednotlivé porty jsou na sobě nezávislé.

Adresa tohoto obvodu je:

0	1	0	0	A2	A1	A0	R/W
---	---	---	---	----	----	----	-----

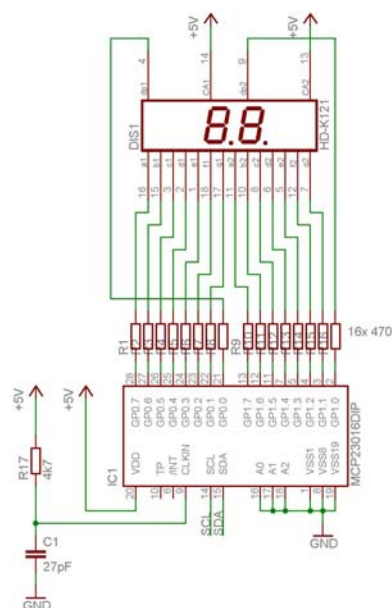
Bitů A0-A2 jsou hardwarově volitelné.

Na obrázku Obr. 22 je vidět blokové schéma expandéru MCP23016. Jak je vidět, porty jsou tvořeny 2x8 vstupně/výstupními linkami. V přípravku je využitý pouze jako výstupní ovládací dvě sedmi segmentová čísla.



Obr. 22 Blokové schéma expandéru MCP23016 (převzato z [9])

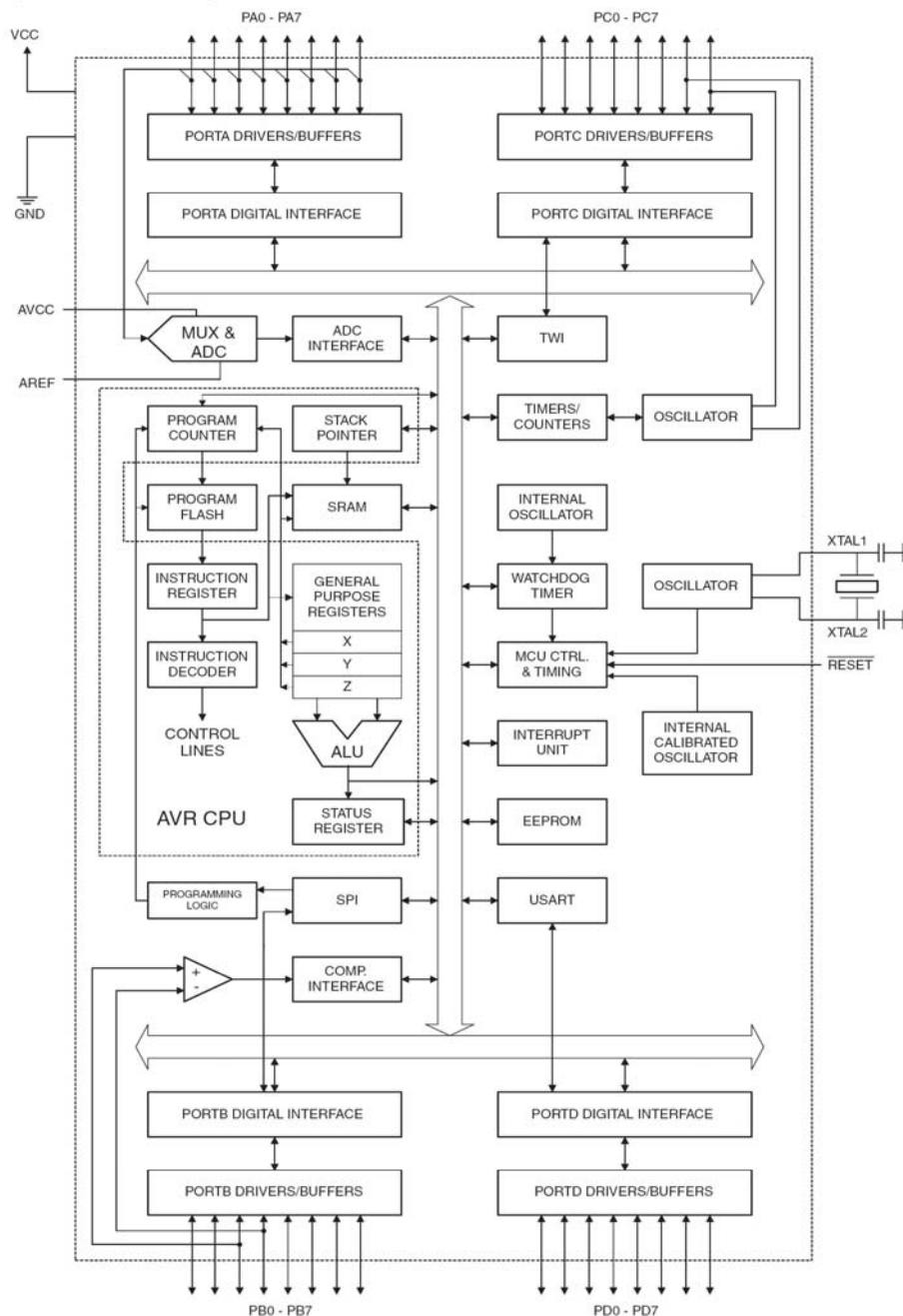
Zapojení expandéru a dvou sedmi segmentových čísel je patrné ze schématu na obrázku Obr. 23.



Obr. 23 Zapojení obvodu MCP23016

2.5 Komunikace po sběrnici I2C (SW pohled)

Na obrázku Obr. 24 je vidět blokové schéma procesoru ATmega16, jde o řídicí procesor celého přípravku. Z tohoto schématu vyplývá jeho konektivita na okolí. Jeho rozhraní se sestává ze čtyř 8-mi bitových vstupně/výstupních portů. Důležitou součástí procesoru je jednotka TWI (Two-Wire Serial Interface). TWI je kompatibilní s I2C.



Obr. 24 Blokové zapojení procesoru ATmega16 (převzato z [10])

Jednotka TWI je přivedena na piny PC0(SCL) a PC1(SDA). Komunikace po sběrnici I²C probíhá zápisem do SFR registrů TWBR, TWDR, TWSR a TWAR. Význam jednotlivých registrů je uveden v následujícím textu.

2.5.1 Používání SFR registrů pro TWI procesoru ATmega16

Podrobný popis jednotlivých SFR (Registry Speciálních funkcí) registrů:

TWBR (TWI Bit Rate Register) - registr přenosové rychlosti TWI

Tento registr volí společně s bity předděličky přenosovou rychlost.

Bit	7	6	5	4	3	2	1	0
	TWBR7	TWBR6	TWBR5	TWBR4	TWBR3	TWBR2	TWBR1	TWBR0
Čtení/zápis	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Výchozí hodnota	0	0	0	0	0	0	0	0

TWCR (TWI Control Register) - řídicí registr TWI

Tento registr je použit pro řízení operací na TWI. Je použit pro povolení funkce TWI, inicializaci master přístupu START stavem na sběrnici, generuje ACK přijímače, generuje STOP stav a řídí zastavení sběrnicových operací, když data pro zápis na sběrnici jsou zapisována do datového registru TWDR. Také indikuje kolizi zápisu při pokusu zapsat data do registru TWDR v okamžiku, kdy je tento registr nedostupný.

Jednotlivé bity registru TWCR:

Bit	7	6	5	4	3	2	1	0
	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
Čtení/zápis	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Výchozí hodnota	0	0	0	0	0	0	0	0

TWINT - příznak přerušení TWI. Tento bit je nastaven hardwarově, když TWI dokončí svůj aktivní úkol a čeká programovou odezvu. Je-li globální povolení přerušení I = 1 a TWIE = 1, vyvolá se po nastavení příznaku TWINT Obsluha přerušení TWI jednotky. V průběhu nastaveného příznaku TWINT setrvává linka SCL v log. 0. Příznak TWINT lze vynulovat programově zápisem 1 do tohoto příznaku. Příznak TWINT se vynuluje hardwarově po vstupu do obslužné rutiny přerušení TWI jednotky. Každé vynulování TWINT startuje operaci na TWI, takže všechny přístupy do registrů TWAR, TWSR a TWDR musí být dokončeny před vynulováním tohoto příznaku.

TWEA - povol potvrzování na TWI. Tento bit řídí generaci potvrzovacího pulzu. Pro TWEA = 1 je ACK generován I²C sběrnici, pokud nastane jedna z těchto podmínek:

- zařízení přijalo svou slave adresu,
- byla přijata broadcast adresa (000 0000b) a současně nastaven bit TWGCE,
- byl přijat datový bajt adresy v režimech přijímače (lhostejno jestli master nebo slave).

Zápisem 0 do bitu TWEA může být zařízení virtuálně dočasně odpojeno od I2C sběrnice. Rozpoznání adresy pak může být znovu povoleno až zápisem 1 do bitu TWEA.

TWSTA - START stav. Aplikace zapíše 1 do bitu TWSTA, pokud se chce stát masterem. Hardware testuje, je-li sběrnice volná a generuje START stav. Bit TWSTA musí být po odeslání START stavu programově vynulován.

TWSTO - STOP stav. Zápis 1 do bitu TWSTO v režimu master bude generovat STOP stav. Když je STOP stav proveden, je bit TWSTO automaticky vynulován. Ve slave režimu může být nastavení bitu TWSTO použito pro zotavení z chybového stavu sběrnice. V tomto případě se STOP stav negeneruje, ale I2C sběrnice se vrátí do definovaného stavu (nenaadresovaný slave režim s uvolněnými linkami SDA a SCL).

TWWC - příznak kolize zápisu. Příznak TWWC je nastaven při pokusu zapsat do registru TWDR v okamžiku, kdy je TWINT = 0. Tento příznak je vynulován zápisem do registru TWDR v okamžiku, kdy je nastaven příznak TWINT.

TWEN - povolení funkce TWI. Tento bit povoluje TWI operace a aktivuje I2C rozhraní. Po zápisu TWEN = 1 převezme TWI řízení vstupně/výstupních linek SCL a SDA, povolí omezovače rychlosti přeběhu a potlačovač šumu. Pro TWEN = 0 se TWI vypne a všechny TWI přenosy se přeruší nezávisle na probíhající operaci.

TWIE - povolení přerušení TWI. Pro TWIE = 1 a I = 1 je povolen příjem TWI přerušení.

TWSR (TWI Status Register) - stavový registr TWI

Tento registr obsahuje 5 bitů, které určují stav TWI jednotky a dále 2 bity předděličky přenosové rychlosti.

Bit	7	6	5	4	3	2	1	0
	TWS7	TWS6	TWS5	TWS4	TWS3	-	TWPS1	TWPS0
Čtení/zápis	R	R	R	R	R	R	R/W	R/W
Výchozí hodnota	1	1	1	1	1	0	0	0

TWS - stav TWI. Těchto 5 bitů odráží aktuální stav TWI logiky a sběrnice.

TWPS - TWI předdělička (viz Tab. 3).

Tab. 3 Předdělička přenosové rychlosti sběrnice I2C

TWPS1	TWPS0	Dělicí poměr
0	0	1
0	1	4
1	0	16
1	1	64

TWDR (TWI Data Register) - datový registr TWI

TWDR je datový registr TWI. Ve vysílacím režimu obsahuje další byte pro odeslání, v přijímacím režimu obsahuje poslední přijatý byte.

Tento registr je přístupný pro zápis, pokud TWI neprovádí vysouvání byte. Tento okamžik nastane, když je nastaven příznak TWINT. Registr TWDR nesmí být inicializován před prvním vykonáním TWI přerušení.

Data v registru TWDR zůstávají stabilní tak dlouho, dokud je TWINT =1. V průběhu vysouvání dat ven jsou současně ze sběrnice vsouvána vstupní data. TWDR vždy obsahuje poslední bajt vložený na sběrnici vyjma okamžiku probuzení z režimu snížené spotřeby přes TWI přerušení. V tomto případě je obsah TWDR nedefinovaný.

Manipulace s ACK bitem je řízena automaticky TWI logikou, k bitu ACK nelze přistoupit přímo.

Bit	7	6	5	4	3	2	1	0
	TWD7	TWD6	TWD5	TWD4	TWD3	TWD2	TWD1	TWD0
Čtení/zápis	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Výchozí hodnota	1	1	1	1	1	1	1	1

TWAR (TWI Adress Register) - adresový registr TWI

Tento registr obsahuje 7-mi bitovou adresu slave obvodu, na kterou bude TWI odpovídat při naprogramování jako slave. V systémech s více mastery musí být registr TWAR nastaven v masterech, které mají být adresovány jako slave obvody ostatními mastery. Bit

TWGCE - pro rozlišení broadcast adresy (000 0000b). Připojený adresní komparátor generuje při shodě adres přerušení.

Bit	7	6	5	4	3	2	1	0
	TWA6	TWA5	TWA4	TWA3	TWA2	TWA1	TWA0	TWGCE
Čtení/zápis	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Výchozí hodnota	1	1	1	1	1	1	1	0

TWA - TWI adresa. 7-mi bitová adresa slave obvodu.

TWGCE - rozlišení broadcast. Je-li TWGCE = 1, je povoleno rozlišovat broadcast oslovení slave obvodu (reaguje na adresu 0000 0000b).

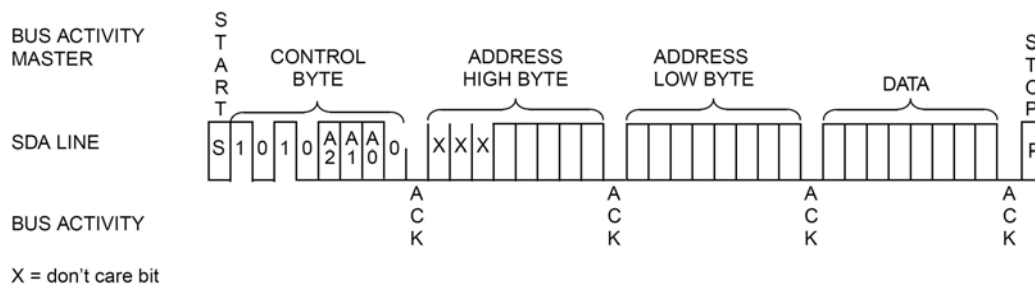
2.5.2 Komunikace s pamětí 24C64P

Komunikace s pamětí může mít dvě podoby:

- Zápis po bytu
- Zápis po stránkách
- Čtení z přímé adresy
- Čtení s nastavením adresy

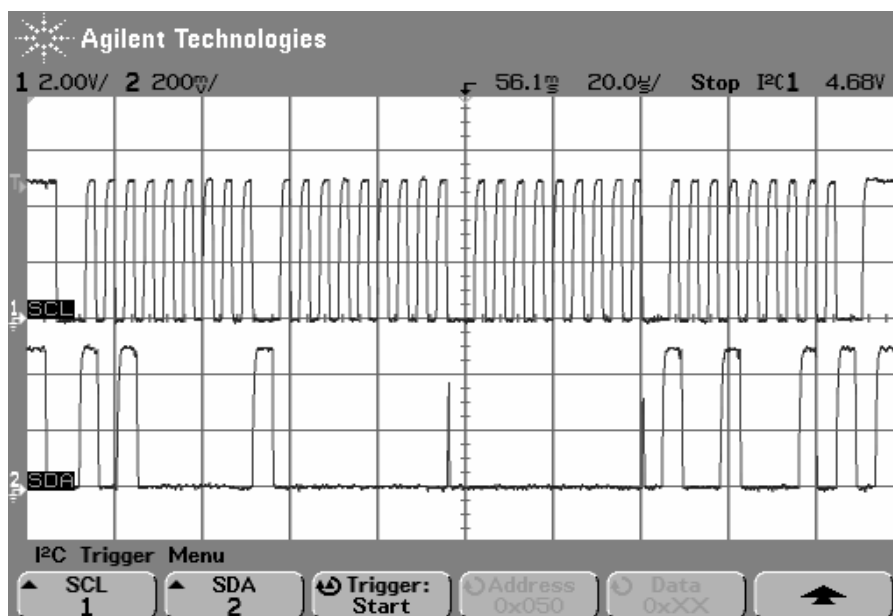
Zápis po bytu

Jak je vidět na obrázku Obr. 25, nejdříve se vyšle START stav, za ním následuje adresa paměti složená ze 4 bitů výrobcem pevně daných, 3 bitů hardwarově volitelných, z tohoto plyne možnost použití až osmi stejných pamětí na jedné sběrnici I²C, a bit signalizující zápis. Následující byte je horní byte adresy paměťové buňky s obsazenými 5 spodními bity (8kB => 13bitů, 5+8=13). Další (3-tí) byte je dolní byte adresy paměťové buňky. A poslední (4-tý) byte jsou data určená pro zápis do paměti.



Obr. 25 Zápis do paměti 24C64 po bytu

Na obrázku Obr. 26 je patrný přenos jednoho čísla na danou adresu v paměti. Nejdříve je vysílána adresa s bitem R/W (0x50 + R/W), další dva bajty představují adresu paměťové buňky (0x0000) a poslední bajt je samotné číslo (0x92).

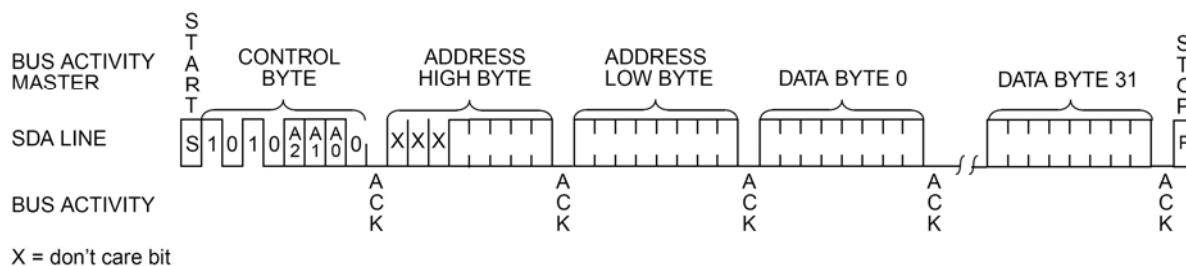


Obr. 26 Uložení čísla 0x92 do paměti 24C64 na adresu 0x0000.

Zápis po stránkách

Na obrázku Obr. 27 je vidět schéma zápisu dat do paměti po stránkách (stránka = 32B). Komunikace je obdobná s rozdílem, že se neposílá jeden byte dat ale může se jich zapsat 32

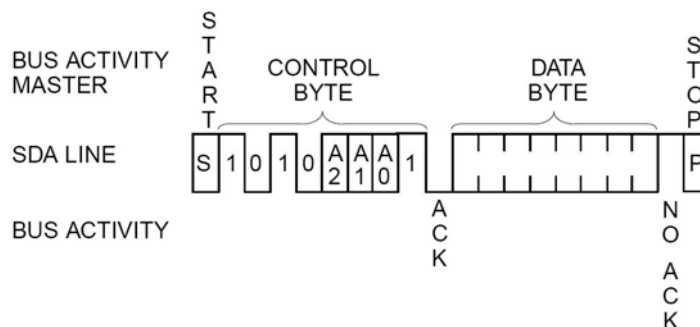
za sebou. Adresa paměťové buňky je po každém zapsaném byte inkrementována z čehož plyne, že další byte je zapsán na následující vyšší adresu do paměti.



Obr. 27 Zápis do paměti 24C64 po stránce (převzato z [4])

Čtení z přímé adresy

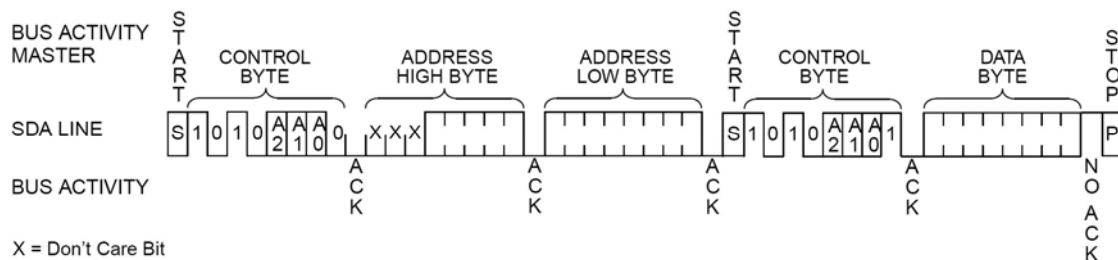
Obrázek Obr. 28 osvětluje možnost číst data přímo z adresy, která je uložena v adresním registru přímo v paměti. V tomto registru je adresa závislá na posledním zápisu nebo čtení z této paměti.



Obr. 28 Čtení paměti podle hodnoty adresního registru (převzato z [4])

Čtení s nastavením adresy

Na obrázku Obr. 29 je vidět čtení jednoho byte. Čtení probíhá takto, nejdříve se do paměti zapíše adresa požadovaných dat, stejně jako při zápisu dat, poté se vyšle opětovný start s vysláním adresy paměti s rozdílem v bitu R/W a poté již následuje čtení z přímé adresy.



Obr. 29 Čtení z paměti, z určeného místa (převzato z [4])

Při sekvenčním čtení se po prvním přijatém bytu dat vyšle bit ACK a očekává se další byte z paměti.

2.5.3 Komunikace s obvodem reálného času DS1337

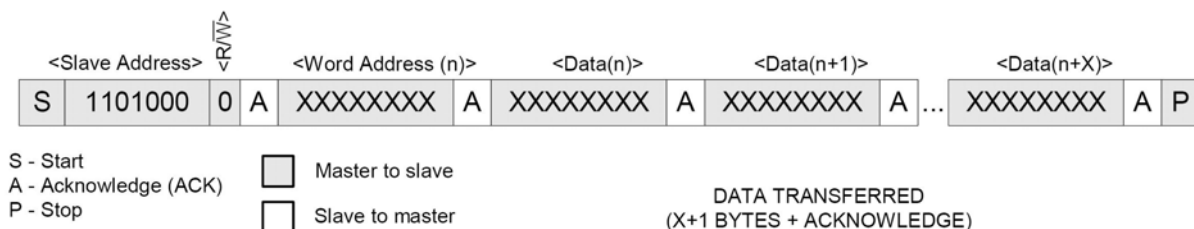
Komunikace s tímto obvodem je v zásadě velmi podobná komunikaci s pamětí (viz kapitola 2.5.2). Tento obvod obsahuje také paměťová místa, jen jsou v nich uloženy informace o čase a datu. Schéma paměťových buněk v obvodu DS1337 vypadá takto podle tabulky Tab.4.

Tab. 4 Schéma paměťových buněk v obvodu DS1337

ADDRESS	BIT 7	BIT 6	BIT 5	BIT 4	BIT 3	BIT 2	BIT 1	BIT 0
00H	0	10 Seconds	Seconds	Seconds	00–59			
01H	0	10 Minutes	Minutes	Minutes	00–59			
02H	0	12/24	AM/PM	10 Hour	Hour	Hours	1–12 +AM/PM00–23	
			10 Hour					
03H	0	0	0	0	0	Day	Day	1–7
04H	0	0	10 Date	Date	Date	01–31		
05H	Century	0	0	10 Month	Month	Month/ Century	01–12 + Century	
06H	10 Year	Year	Year	00–99				
07H	A1M1	10 Seconds	Seconds	Alarm 1 Seconds	00–59			
08H	A1M2	10 Minutes	Minutes	Alarm 1 Minutes	00–59			
09H	A1M3	12/24	AM/PM	10 Hour	Hour	Alarm 1 Hours	1–12 + AM/PM 00–23	
			10 Hour					
0AH	A1M4	DY/DT	10 Date	Day	Alarm 1 Day	1–7		
Date	Alarm 1 Date	1–31						
0BH	A2M2	10 Minutes	Minutes	Alarm 2 Minutes	00–59			
0CH	A2M3	12/24	AM/PM	10 Hour	Hour	Alarm 2 Hours	1–12 + AM/PM 00–23	
			10 Hour					
0DH	A2M4	DY/DT	10 Date	Day	Alarm 2 Day	1–7		
Date	Alarm 2 Date	1–31						
0EH	EOSC	0	0	RS2	RS1	INTCN	A2IE	A1IE
0FH	OSF	0	0	0	0	0	A2F	A1F

Zápis dat

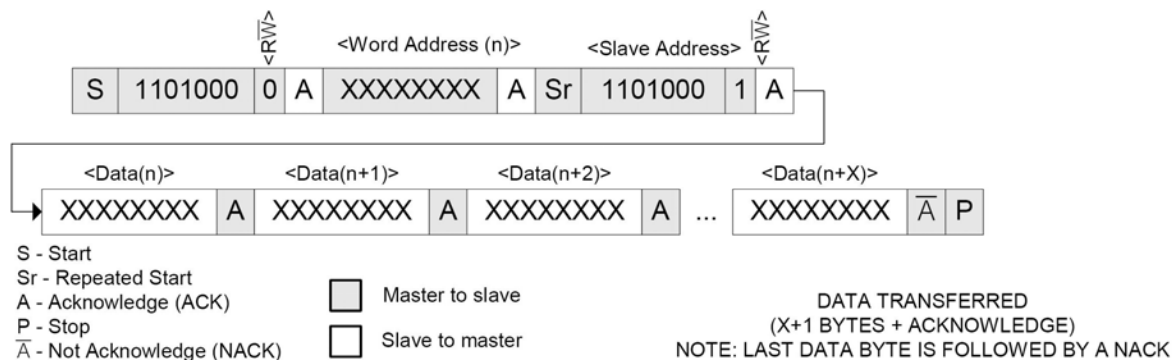
Při zápisu dat se musí nejdříve zapsat adresa, kam mají být data zapsána a teprve potom samotná data (viz Obr. 30).



Obr. 30 Zápis dat do obvodu reálného času DS1337 (převzato z [6])

Čtení dat

Čtení dat je obdobné jako u předešlé paměti. Nejdříve se zapíše adresa požadovaného údaje a po opětovné podmínce start se čte byte po byte jako u paměti (viz Obr. 31).



Obr. 31 Čtení dat obvodu reálného času DS1337 (převzato z [6])

Samozřejmě je možné přímé čtení z adresy jako u paměti, nicméně tato možnost není příliš vhodná protože ne všechna data z obvodu vždy potřebujeme.

2.5.4 Komunikace s teplotním čidlem DS1621

Tato komunikace se mírně liší, ale základ je pořád stejný jako u paměti. Rozdíl je jen v tom, že teplotnímu čidlu se posílají příkazy podle tabulky Tab. 5. Po tomto příkazu se přenáší data.

Nejdříve se vyšle adresa cílového obvodu, poté příkaz co se má udělat a nakonec data.

Tab. 5 Příkazy pro teplotní čidlo DS1621

Příkaz	Popis	Binární hodnota	Události na sběrnici I2C po použití	Pozn.
Čtení teploty	Přečte poslední konvertovanou teplotu z registru	AAh	<čte 2 byte data>	
Začátek konverze	Spouští převod teploty	Eeh	čekání	1
Konec konverze	Zastaví konverzi	22h	čekání	1

2.5.5 Komunikace s expandérem MCP23016

Komunikace probíhá podobně jako u teplotního čidla pomocí řídicího byte. A poté se posílají, či přijímají data.

2.6 Návrh obvodového zapojení a návrh plošného spoje

V příloze je uvedeno kompletní schéma zapojení (Příloha A), dále návrh plošného spoje pomocí programu Eagle (Příloha B - strana spojů, Příloha C - strana součástek).

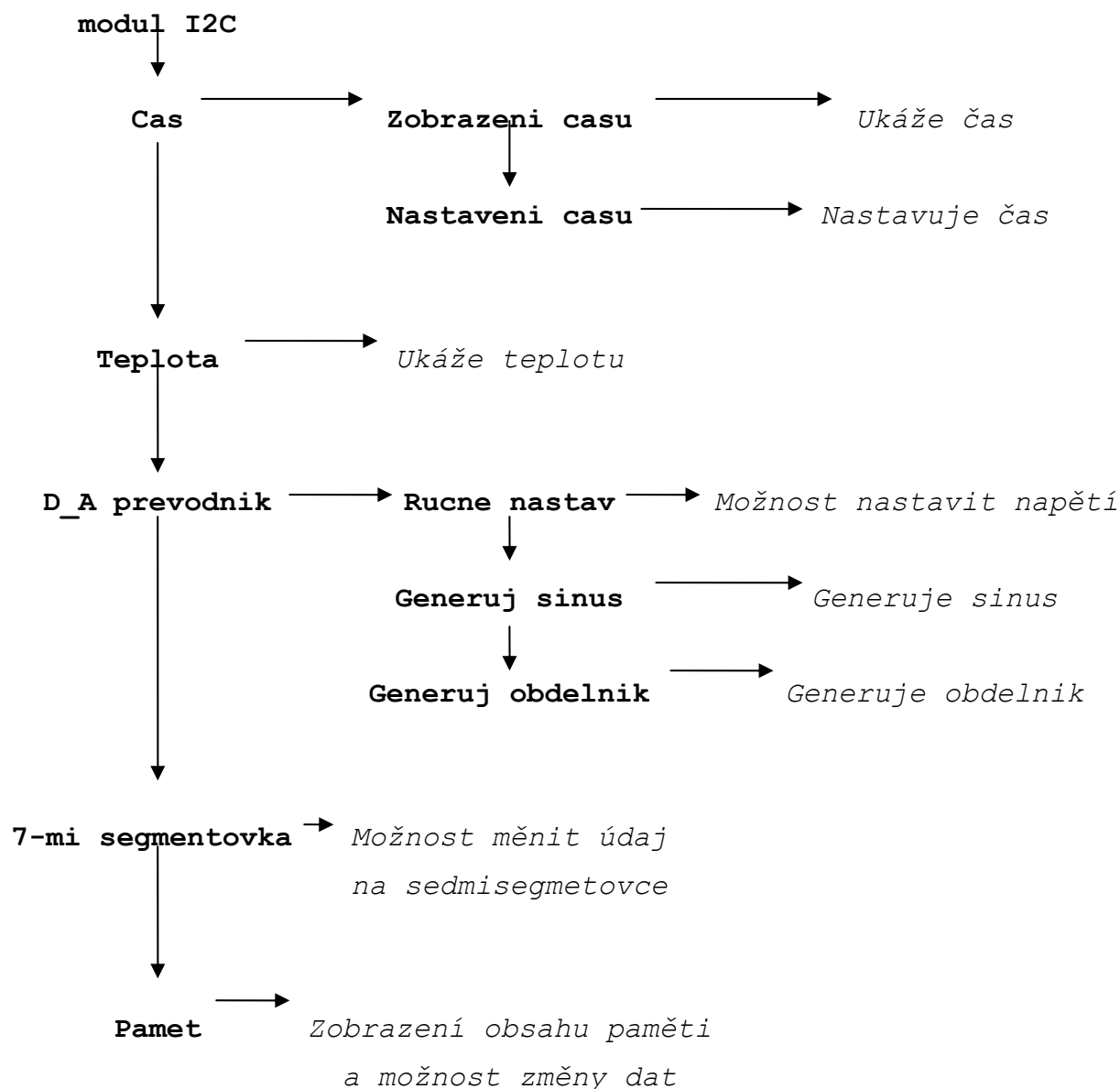
3. SW popis přípravku

Tato část se bude zabývat aplikací nahanou v ovládacím procesoru a v procesoru pro ovládání klávesnice.

3.1 Struktura menu

Struktura menu je zvolena tak jak je zvykem například u mobilních telefonů.

Vítejte v aplikaci



Pohyb v menu je realizován klávesami 2,4,5,6 a *. Kde klávesy mají tento význam:

2 - nahoru

5 - dolu

6 - vpřed / potvrď / doprava

4 - zpět / doleva

* - zpět z menu paměti


```
{  
    PORTA = 0xfe;           //nastavení nízké úrovně na LSB  
    __asm__ ("nop");       //NOP pro ustálení hodnoty na pinech  
    if (PINA != 0xFE)      //kontrola jestli je něco stisknuto  
    {  
        zmena();           //zaznamenání stisknutého tlačítka  
        return;  
    }  
  
    PORTA = 0xfd;          //nastavení nízké úrovně na dalším pinu  
    __asm__ ("nop");       //takto se zopakuje pro všechny 4 linky  
    if (PINA != 0xFD)  
    {  
        zmena();           //zaznamenání stisknutého tlačítka  
        return;  
    }  
  
    PORTA = 0xfb;          //nastavení nízké úrovně na dalším pinu  
    __asm__ ("nop");       //takto se zopakuje pro všechny 4 linky  
    if (PINA != 0xFB)  
    {  
        zmena();           //zaznamenání stisknutého tlačítka  
        return;  
    }  
  
    PORTA = 0xf7;          //nastavení nízké úrovně na dalším pinu  
    __asm__ ("nop");       //takto se zopakuje pro všechny 4 linky  
    if (PINA != 0xF7)  
    {  
        zmena();           //zaznamenání stisknutého tlačítka  
        return;  
    }  
    x = 0;  
}
```

Podprogram `zmena()`; je pouze zaznamenání hodnoty na vstupní portu a nastavení příznakové proměnné `x`.

PORT A procesoru je rozdělen na 4 výstupní a 4 vstupní linky, a je třeba na výstupních linkách nastavovat nízkou hodnotu a na vstupních sledovat kam se tato hodnota promítne.

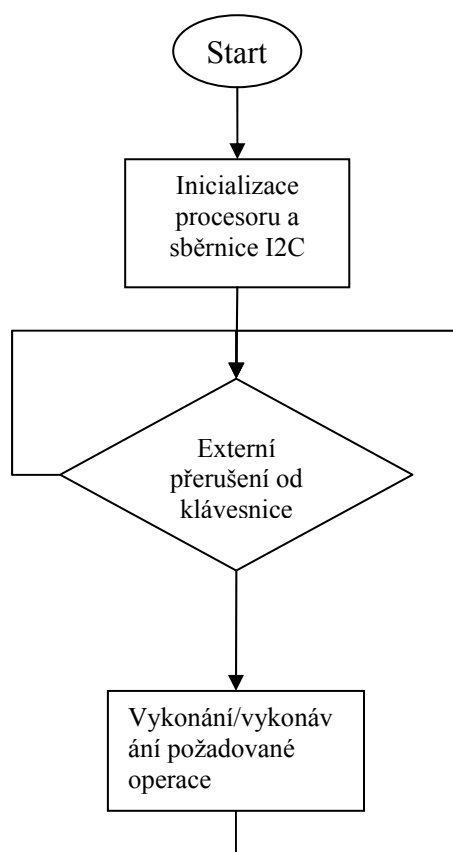
Vložený NOP je jen kvůli ustálení hodnoty na výstupní lince. Proměnná *x* je použita jako příznak stisknuté klávesy na klávesnici.

Ostatní program se skládá jen z vhodného uspořádání bitů na výstupních linkách do procesoru ATmega16 a vhodné generaci přerušovacího signálu, tak aby se ošetřilo trvale stisknuté tlačítko klávesnice.

3.3 Ovládací procesor ATmega16

Tento procesor je pro aplikaci, která řídí chod celé desky a jednotlivých periférií, které jsou připojeny na sběrnici I2C, ale i mimo ni.

Vzhledem k osazenému displeji LCD a klávesnici, je v přípravku použito jednoduché menu ve kterém se dá pohybovat pomocí klávesnice, kde je využito kláves ve tvaru šipek na počítačové klávesnici. Menu je sestaveno tak, aby šlo přes menu ovládat všechny periferie připojené na sběrnici I2C.



Obr. 33 Vývojový diagram aplikace v ovládacím procesoru

Vývojový diagram na Obr.33 ukazuje princip činnosti programu v ovládacím procesoru. Po zapnutí přípravku se zobrazí úvodní uvítací zpráva a čeká se na vstup z klávesnice. Po

zvolení žádané funkce pomocí menu a klávesnice procesor začne komunikovat po sběrnici a zobrazovat data na displeji případně na diodové dvoumístné sedmissegmentovce. Například komunikace s obvodem reálného času a čidlem teploty probíhá neustále ale komunikace s expandérem probíhá jednorázově.

3.3.1 Řízení displeje LCD

K řízení displeje je použit program používaný v laboratořích kde původním autorem je Peter Fleury a dále jej upravil Tomáš Frýza a pro účely přípravku bylo nutno změnit pouze piny na které je displej připojen. V samotném programu jsou už jenom použity funkce již navržené v tomto ovladači LCD displeje.

3.3.2 Příjem dat z klávesnice

Klávesnice je zapojená do matice 4x4 (16 tlačítek) z tohoto plyne že na prezentaci všech tlačítek plně postačují v binárním kódu 4 bity. Tyto 4 bity jsou přivedeny čtyřmi linkami na ovládací procesor ke zpracování a pro názornost ještě zobrazeny svítícími diodami.

Příjem samotných dat je řešen v přerušení od externího vstupu INT0 takto:

```
ISR (INT0_vect)
{
    Inp_Key = ((PINB & 0x0f) | 0x80);
}
```

Čtyři linky s daty jsou přivedeny na spodní čtyři bity PORTB a je nastaven nejvyšší bit jako příznak přijetí nové hodnoty z klávesnice. Další zpracování stisknutých kláves probíhá již v samotném menu.

3.3.3 Ovládání sběrnice I2C

I přes to že sběrnice I2C je typu MultiMaster v přípravku je poříto pouze jednoho obvodu, který může sloužit jako master a tím je ovládací procesor ATmega16. Nejdříve je nutno správně inicializovat SFR registry ovládající jednotku TWI.

Inicializace TWI:

```
void i2c_init (void)
{
    TWCR = 0b01000100;    //povolení TWI a generování ACK
    TWAR = 0b01010100;    //adresa procesoru
    TWBR = 10;            //nastavení kmitočtu SCL (výpočet níže)
    TWSR = 0b11111000;    //pouze inicializace základních hodnot
```


}

TWCR - TWI Control Register

Bit	7	6	5	4	3	2	1	0
	TWINT	TWEA	TWSTA	TWSTO	TWWC	TWEN	-	TWIE
Čtení/zápis	R/W	R/W	R/W	R/W	R/W	R/W	R/W	R/W
Výchozí hodnota	0	0	0	0	0	0	0	0

Aktivován je bit TWEN (TWI Enable), který povoluje funkci jednotky TWI a TWEA slouží k povolení generování bitu ACK po přijetí všeobecné adresy nebo adresy vlastního obvodu.

TWAR - Registr pro uložení adresy ovládacího procesoru, kdyby byl do obvodu připojen jiný procesor typu master.

TWBR - Registr pro generování hodinového taktu v módu master

TWSR - TWI Status Register

Bit	7	6	5	4	3	2	1	0
	TWS7	TWS6	TWS5	TWS4	TWS3	-	TWPS1	TWPS0
Čtení/zápis	R	R	R	R	R	R	R/W	R/W
Výchozí hodnota	1	1	1	1	1	0	0	0

Bity TWS3..7 prezentují status na sběrnici I²C a bity TWPS0..1 jsou bity předděličky pro generátor kmitočtu hodinového signálu pro sběrnici I²C. Vzorec pro výpočet frekvence hodinového signálu na sběrnici I²C:

$$SCL \text{ frekvence} = \frac{CPU \text{ frekvence}}{16 + 2(TWBR) + 4^{TWPS}} \quad 3.1$$

Z daného vzorce se dá lehce spočítat při frekvenci vnitřního oscilátoru ovládacího procesoru 8MHz:

$$SCL \text{ frekvence} = \frac{8 \cdot 10^6}{16 + 2(10) + 4^1} = 200kHz \quad 3.2$$

Ze vzorce je patrné, že při použití externího krystalu 16MHz bychom mohli dosáhnout frekvence hodinového taktu na sběrnici I²C až 400kHz.

Použité funkce při ovládní sběrnice I²C:

- Kontrola stavu sběrnice I2C:

```
void kontrola (unsigned short STAV)
{
    if ((TWSR & 0xF8) != STAV) //pokud nenní na sběrnici předpokládaný stav
        ERROR(); //zobrazí chybu na displeji
}
```

Jde o funkci pro snazší ladění programu. Jestliže se stav na sběrnici neshoduje s předpokládaným stavem tak se na displej vypíše chybové hlášení toto obstará podprogram ERROR();.

Generování podmínek Start a Stop:

```
void twi_start (void)
{
    TWCR = (1<<TWINT) | (1<<TWEN) | (1<<TWSTA); //Start
}

void twi_stop (void)
{
    TWCR = (1<<TWINT) | (1<<TWEN) | (1<<TWSTO); //Stop
}
```

Dané funkce slouží pro zahájení a ukončení komunikace na sběrnici I2C.

Čekání na dokončení přenosu dat po sběrnici

```
void hotovo (void)
{
    while (!(TWCR & (1<<TWINT))); //čekací smyčka na dokončení operací
}
```

Tato funkce pouze čeká na dokončení komunikace po sběrnici I2C, která je prezentována vystavením příznaku přerušování pro sběrnici I2C.

Funkce pro přenos veškerých dat po sběrnici

```
void vysli (unsigned short ADR)
```

```

{
    TWDR = ADR;                                     //uložení vysílaných dat do TWDR
    TWCR = (1<<TWINT) | (1<<TWEN);
}

```

Z pohledu předešlé funkce nezáleží na účelu dat protože vysílání adresy a dat je rovnocenný přenos informace.

```

void cti_data_ACK (void)
{
    TWCR = (1<<TWINT) | (1<<TWEN) | (1<<TWEA); //čte data a potvrdí přijetí ACK
}

```

```

void cti_data_NACK (void)
{
    TWCR = (1<<TWINT) | (1<<TWEN);           //čte data a nepotvrdí přijetí
}

```

Funkce **vysli (ADR)** ; slouží pro vyslání byte po sběrnici nezáleží na tom jestli jde o adresu nebo byte dat.

Další dvě funkce slouží pro příjem dat v prvním případě je po správném přenosu tento potvrzen bitem ACK a v druhé funkci, která se používá na poslední přenášený byte bit ack vysílán není.

Vlastní komunikace s čidlem teploty

```

void nacti_Therm (void)
{
    twi_start();
    hotovo();
    kontrola(TW_START);

    vysli(0x90);                                     //Adresa obvodu + Write bit
    hotovo();
    kontrola(TW_MT_SLA_ACK);

    vysli(0xee);                                     //Prikaz mereni
    hotovo();
    kontrola(TW_MT_DATA_ACK);

    twi_start();                                     //Opětovná podmínka Start
    hotovo();
    kontrola(TW_REP_START);
}

```

```
    vysli(0x90); //Adresa obvodu + Write bit
    hotovo();
    kontrola(TW_MT_SLA_ACK);

    vysli(0xaa); //Příkaz k přečtení teploty
    hotovo();
    kontrola(TW_MT_DATA_ACK);

    twi_start(); //Opětovná podmínka Start
    hotovo();
    kontrola(TW_REP_START);

    vysli(0x91); //Adresa obvodu + Read bit
    hotovo();
    kontrola(TW_MR_SLA_ACK);

    cti_data_ACK(); //MS Byte
    hotovo();
    kontrola(TW_MR_DATA_ACK);
    MS_Byte = TWDR;

    cti_data_NACK(); //LS Byte
    hotovo();
    kontrola(TW_MR_DATA_NACK);
    LS_Byte = TWDR;

    twi_stop(); //Podmínka Stop
}
```

Jako příklad komunikace po sběrnici je vybráno právě teplotní čidlo, protože se zde uplatní jak zápis opětovná podmínka start tak čtení. Nejdříve je nutno adresovat obvod a zadat příkaz ke konverzi teploty. Poté je nutno zaslat opětovnou podmínku start a zadat příkaz ke čtení teploty a nakonec znovu podmínka start a vyčítání teploty ve dvou bytech. Vše je nakonec zakončeno podmínkou Stop.

4. Laboratorní úloha

Využití sběrnice I2C pro komunikaci s externími zařízeními

4.1 Cíle laboratorní úlohy:

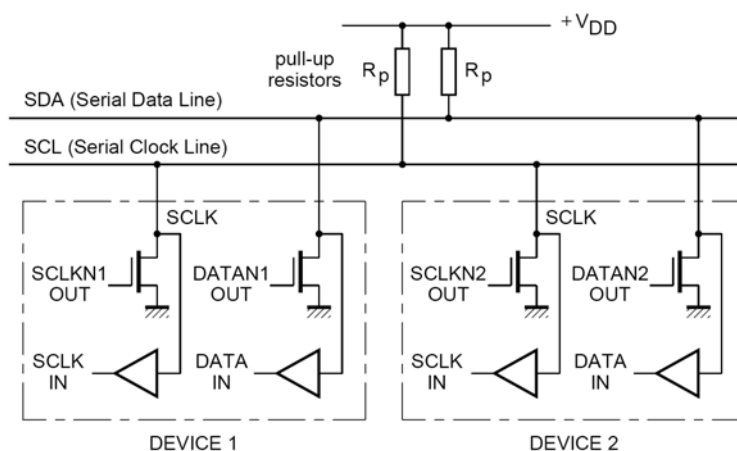
Cílem laboratorní úlohy je seznámit studenty se sběrnicí I2C a jejího ovládání pomocí procesoru AVR od firmy Atmel ATmega16 a jeho jednotky TWI pro dvoudrátovou sériovou komunikaci.

4.2 Zadání:

1. Seznamte se s typy komunikace po sběrnicí I2C a prostudujte sousled jednotlivých bytů za sebou.
2. Pomocí klávesnice a menu na displeji se naučte pracovat s přípravkem, vyzkoušejte všechny položky v menu.
3. Pomocí osciloskopu zjistěte adresu D/A převodníku MAX518 a 16-bitového expandéru MCP23016.
4. Přečtěte teplotu z teplotního čidla DS1621 a zobrazte ji v desítkovém formátu na displeji LCD.

4.3 Teoretické poznatky:

Sběrnice I2C je fyzicky realizována dvěma vodiči, datovým SDA a hodinovým SCL, dvěma zdvihacími rezistory a zemním vodičem podle obrázku Obr. 34.



Obr. 34 fyzická realizace sběrnice I2C (převzato z [1])

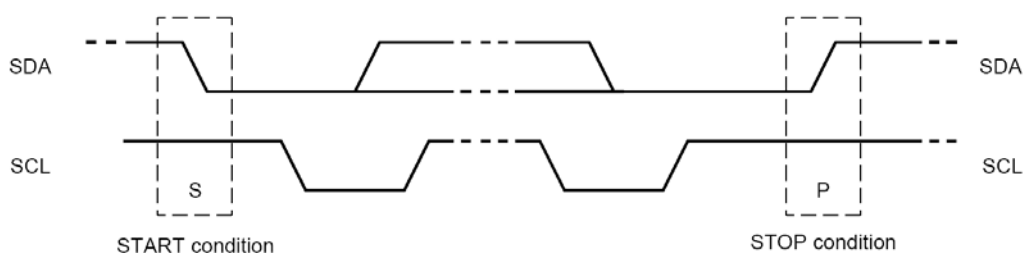
Adresa jednotlivých zařízení je 7-mi nebo 10-ti bitová (použijeme pouze 7-mi bitovou) což znamená, že na sběrnici můžeme najednou připojit až 128 zařízení, ale sběrnice má ještě kapacitní limit a to 400pF.

Zařízení připojené na sběrnici se dělí na master/slave a vysílač/přijímač. Zařízení, které generuje hodinový signál je master všechny ostatní jsou slave ať už se jedná o procesor nebo jiné periferie. Zařízení které vysílá data na sběrnici je vysílač a zařízení, které přijímá data ze sběrnice je přijímač. Takže neplatí jednoduchá rovnice master=vysílač a slave=přijímač.

Komunikace po sběrnici probíhá následovně:

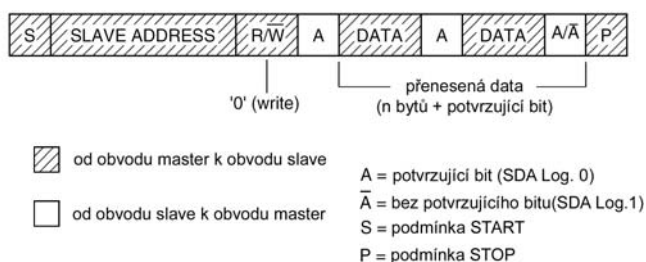
- Generování podmínky START
- Přenos prvního byte (7 bitů adresa a 1 bit R/W)
- Následné potvrzení přijaté adresy zařízením typu slave (1 bit)
- Postupný přenos dalších byte následovaných potvrzujícími bity (většinou následuje jeden řídicí a potom přenos datových bytů)

Jak je vidět na obrázku Obr.35 podmínky Start a Stop se generují tak že při vysoké úrovni na lince SCL (hodiny) se změní stav na lince SDA (data), ale během přenosu dat musí být na lince SDA, během vysoké úrovně na lince SCL, data stabilní.



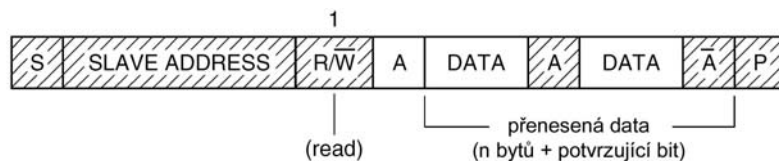
Obr.35 START(S) a STOP(P) podmínka (převzato z [1])

Po zahájení komunikace podmínkou Start se nastaví směr toku dat (jestli se bude zapisovat nebo číst z obvodu typu slave) a ten je stejný až do ukončení komunikace podmínkou Stop nebo opětovnou Start. Jednoduchá komunikace ve formě zápisu dat do obvodu typu slave je znázorněna na obrázku Obr.36.



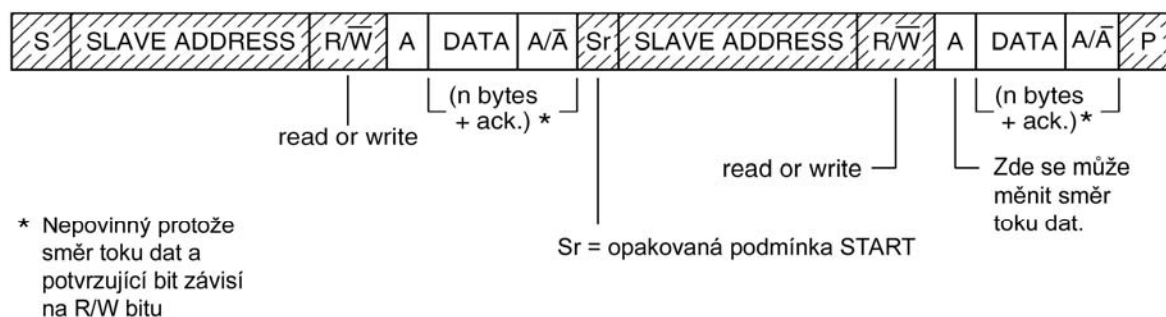
Obr. 36 Master-vysílač adresuje slave-přijímač 7-bitovou adresou. Směr přenosu dat se nemění. (převzato z [1])

Z obrázku Obr. 37 je patrné čtení z obvodu typu slave.



Obr. 37 Master čte ze slave okamžitě po prvním bytu. (převzato z [1])

Na obrázku Obr. 38 jde o kombinovaný přenos dat, nejdříve se adresuje obvod, čtou se z něj data, vysílá se opětovná podmínka START (Sr), adresuje se další obvod typu Slave a čtou se z něj další data, vše je zakončeno podmínkou STOP. Tento způsob se využívá například u paměti kde je potřeba nejdříve zapsat z jaké adresy chceme číst a poté vyčíst data.



Obr. 38 Kombinovaný formát komunikace (převzato z [1])

4.4 Pokyny k zadání

Bod 2. Na klávesnici jsou použita tlačítka 2, 4, 5, 6 a * kde vzhledem k uspořádání mají stejnou funkci jako šipky na počítačové klávesnici:

2 - nahoru

5 - dolů

6 - vpřed / potvrď / doprava

4 - zpět / doleva

* - zpět z menu paměti

Těchto tlačítek využijte pro pohyb v menu. Toto ovládání je jednoduché a nemělo by vám činit nejmenší problémy. Projděte si jednotlivé položky menu a zjistěte k čemu která funkce slouží.

Bod 3. Připojte si dvoukanálový osciloskop na sběrnici I²C a v menu zvolte zobrazení data a času, protože v tomto režimu probíhá neustálá komunikace mezi procesorem a obvodem reálného času. Generování časové základny synchronizujte podle signálu SCL. Po nastavení osciloskopu tak aby byl právě vidět jeden byte nastavte osciloskop do paměťového režimu a v menu vyberte možnost 7-mi segmentovka a měňte její hodnotu. Po každé změně hodnoty se do expandéru odešle jednorázově nová hodnota. V tomto okamžiku je potřeba sejmout průběh prvního bytu, nejlépe s podmínkou Start a přečíst adresu obvodu. Podobně postupujte i s obvodem D/A převodníku.

Bod 4. Otevřete si projekt **teplota.c** a tento doplňte o vyčtení teploty z obvodu DS1621.

Teplotní čidlo DS1621 je v podstatě dvoubajtová paměť na teplotu.

Postup správné komunikace s tímto obvodem:

- Podmínka Start
- Adresace obvodu (adresa je 0b10010000 - poslední bit označuje zápis)
- Příkaz ke konverzi teploty na číslo (0xEE)
- Opětovná podmínka Start
- Adresace obvodu (adresa je 0b10010000 - poslední bit označuje zápis)
- Příkaz, že v dalším kroku se bude vyčítat teplota (0xAA)
- Opětovná podmínka Start
- Adresace obvodu (adresa je 0b10010001 - poslední bit označuje čtení)
- Čtení významnějšího byte
- Čtení méně významného byte

Uspořádání bitů v bytech je patrné z tabulky Tab. 6.

Tab. 6 Závislost výstupního čísla na změřené teplotě

Teplota	Digitální výstup (Binární)	Digitální výstup (Hexadecimální)
+125°C	01111101 00000000	7D00h
+25°C	00011001 00000000	1900h
+½°C	00000000 10000000	0080h
+0°C	00000000 00000000	0000h
-½°C	11111111 10000000	FF80h

-25°C	11100111 00000000	E700h
-55°C	11001001 00000000	C900h

Zobrazení na displej provedete pomocí funkce `show_temp(temp)`; kde `temp` je 16-ti bitová proměnná reprezentující změřenou teplotu.

4.5 Shrnutí

Studenti mají možnost proniknout do komunikace po sběrnici I2C, uvidí jak fyzické signály na linkách, tak si vyzkouší naprogramovat jednoduchou komunikaci po sběrnici.

Závěr

V tomto textu je popsána sběrnice I2C a její ovládání. Sběrnici I2C jak je patrné z textu je možno využít prakticky kdekoliv, kde ke potřeba propojit nějaké čidlo, paměť, či nějaký zobrazovací prvek (např. LCD displej) v rámci jednoho plošného spoje. Výhodou této sběrnice je fakt že pro komunikaci s okolím stačí dvě linky, datový SDA a hodinový SCL. Toto prostorově úspornější řešení oproti například paralelnímu propojení je obrovskou výhodou tam kde je nedostatek místa. Nevýhodou je pomalejší komunikace, ale toto vyplývá už z podstaty sériové komunikace.

I přesto, že je sběrnice I2C pouze dvoudrátová její ovládání není nijak složité i při možnosti na jednu sběrnici připojit teoreticky až 1024 různých zařízení, toto není možné kvůli parazitním kapacitám vznikajícím připojením jednotlivých zařízení.

Sběrnice I2C si najde místo všude tam kde je preferována konstrukční jednoduchost na druhé straně vyvážená nižší rychlostí sběrnice, při porovnání například s paralelním přenosem dat.

Dále práce obsahuje popis jednotlivých zařízení využitých na sběrnici I2C, kterými jsou, sérová paměť EEPROM 64k, obvod reálného času DS1337, teplotní senzor DS1621, D/A převodník MAX518 a 16-ti bitový expandér MCP23016.

V neposlední řadě i popis ovládacího procesoru ATmega16 a rozbor možností jeho programování.

Součástí práce je popis navrženého přípravku. A nakonec návrh laboratorní úlohy do laboratoří předmětu Mikroprocesorová technika aby si studenti mohli sběrnici I2C osvojit i prakticky.

Daný přípravek co se týče programování a výuky není omezen pouze na sběrnici I2C je zde osazen displej LCD, další procesor s připojenou klávesnicí zapojenou do matice 4x4 a čtyřmi diodami. Toto skýtá možnost připravit studentům více úloh na tomto přípravku.

Seznam použité literatury

- [1] I2C. [Online] Available: <http://www.standardics.nxp.com/i2c/> (říjen 2006)
- [2] DUDÁČEK, K. Sériová rozhraní SPI, Microwire, I2C a CAN: home.zcu.cz/~dudacek/NMS/Seriova_rozhrani.pdf (duben 2007)
- [3] MATOUŠEK, David. Práce s mikrokontroléry Atmel AVR ATmega16
- [4] 24C64. [Online] Available: www.ife.tugraz.at/datashts/microchip/21189a.pdf (duben 2007)
- [5] ATtiny26. [Online] Available: www.atmel.com/dyn/resources/prod_documents/doc1477.pdf (duben 2007)
- [6] DS1337. [Online] Available: www.maxim-ic.com/getds.cfm?qv_pk=3128 (duben 2007)
- [7] DS1621. [Online] Available: pdfserv.maxim-ic.com/en/ds/DS1621.pdf (duben 2007)
- [8] MAX518. [Online] Available: hubbard.engr.scu.edu/embedded/spyglass/max518.pdf (duben 2007)
- [9] MCP23016. [Online] Available: <http://ww1.microchip.com/downloads/en/DeviceDoc/20090C.pdf> (duben 2007)
- [10] ATmega16. [Online] Available: www.atmel.com/dyn/resources/prod_documents/doc2466.pdf (duben 2007)
- [11] Atmel Corporation. [Online] Available: <http://www.atmel.com>