

1. ÚVOD	10
2. ROZBOR ŘEŠENÍ.....	11
2.1 Výběr sběrnice	11
2.1.1 SPI	11
2.1.2 I ² C	12
2.2 Stabilizace napájecího napětí	13
2.3 A/D převodník	13
2.4 Paměť DataFlash.....	13
2.5 Hodiny reálného času (RTC)	13
3. BLOKOVÉ SCHÉMA	14
4. NÁVRH ZAPOJENÍ	15
4.1 Napájecí obvody	15
4.1.1 Stabilizace napětí	15
4.1.2 Ochrana proti přepólování a připojení vyššího napětí.....	15
4.2 Přizpůsobovací obvody	16
4.2.1 Volba operačního zesilovače	16
4.2.2 Omezení vstupní impedance operačního zesilovače	17
4.2.3 Návrh odporového děliče na výstupu operačního zesilovače.....	18
4.2.4 Ochrana analogových vstupů.....	19
4.3 A/D převodník	20
4.3.1 Přepočítání hodnoty z převodníku na napětí.....	20
4.4 Hodiny reálného času (RTC)	21
4.5 Paměť DataFlash.....	22
4.6 Rozhraní RS232	24
5. SOFTWARE	25
5.1 smac_MC9S08GT60.h.....	25
5.2 SPI.H.....	25
5.2.1 Popis chodu SPI sběrnice.....	25
5.2.2 Popis jednotlivých funkcí v souboru SPI.H.....	26
5.2.3 Průběh komunikace s Pamětí dat pomocí sběrnice SPI.....	28
5.2.4 Průběh komunikace s hodinami reálného času (RTC)	31

5.3 Popis souboru ADC.h	33
6. UŽIVATELSKÝ MANUÁL	34
6.1 Připojení napájecího napětí	34
6.2 Konektor CAN1	35
6.3 Zapojení analogových vstupů	35
6.4 Propojky	35
6.5 Konektory JP5 - JP9	36
6.6 Trimry a mikrospínače	37
6.7 Popis příkazů a jejich formát při komunikaci přes SMAC	37
6.7.1 Nastavení data a času	37
6.7.2 Přečtení data a času	38
6.7.3 Přečtení obsahu paměti dat	38
6.7.4 Mazání paměti	39
6.7.5 Přečtení hodnoty z analogového vstupu	39
7. ZÁVĚR	40
8. POUŽITÁ LITERATURA	41
9. PŘÍLOHY	42

1. ÚVOD

Úkolem bylo navrhnout zapojení a vytvořit desku plošných spojů připojitelnou k Zigbee modulu. Deska má obsahovat analogový vstup 4x 0-10 V a měla by obsahovat vhodnou stabilizaci, hodiny reálného času, DataFlash, rozhraní RS232 a vhodné přizpůsobovací obvody. Požadováno bylo schéma zapojení, vytvořená a oživená plošná deska. Součástí zadání je i software pro mikroprocesor.

Zigbee je poměrně novým standardem platným od listopadu 2004. Je to bezdrátová komunikační technologie, založená na standartu IEEE 802.15.4. Jeho využití je velice univerzální – např. řízení budov, dálkové ovládání, monitorování a diagnostika zařízení, vzdálené čtení měřených hodnot, počítačové periferie nebo spotřební elektronika.

Tato technologie je podobná Bluetooth - je určeno pro spojení nízkovýkonových zařízení v sítích PAN na menší vzdálenosti do 75 metrů. Díky použití multiskokového ad-hoc směrování umožňuje komunikaci i na větší vzdálenosti bez přímé radiové viditelnosti jednotlivých zařízení. Hlavní určení směřuje do aplikací v průmyslu a senzorových sítích.

Pracuje v bezlicenčních pásmech na frekvencích 858 MHz, 902–928 MHz a 2,4 GHz. Přenosová rychlost činí 20, 40 a 250 kbit/s.

Navrhovaná deska je určena pro modul vytvořený na Ústavu automatizace a měřicí techniky. Tento modul je založen na čipu PAN4450 od firmy Panasonic. Nejdůležitější součástí tohoto čipu je rádiový čip MC13192 a mikrokontrolér HCS08.

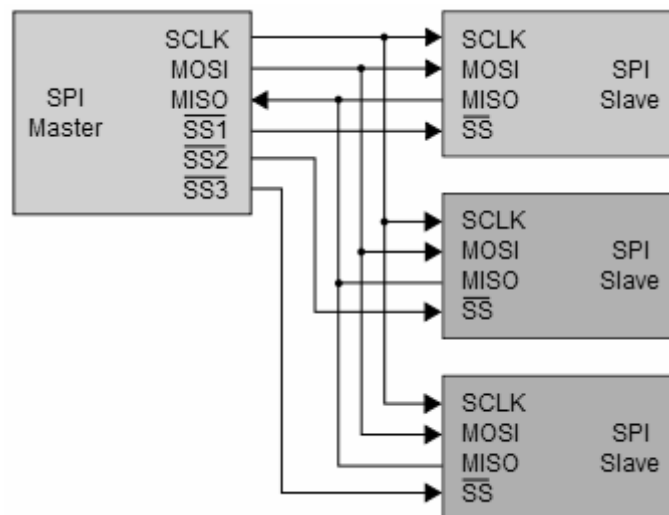
2. ROZBOR ŘEŠENÍ

Dle zadání musí zapojení obsahovat vhodnou stabilizaci, hodiny reálného času, DataFlash, rozhraní RS232 a vhodné přizpůsobovací obvody. Musel jsem tedy vybrat správné součástky a vhodně je zapojit.

2.1 VÝBĚR SBĚRNICE

2.1.1 SPI

Pro komunikaci mezi procesorem, hodinami reálného času a DataFlash budu používat sériovou SPI (Serial Peripheral Interface) linku. SPI linka se jednoduše používá i programuje. Protože se jedná pouze o dvě zařízení, tak i z tohoto hlediska SPI linka dostačuje. Zařízení s kterým bude mikroprocesor komunikovat budeme vybírat pomocí vývodů pro aktivaci čipů (V již hotovém projektu CE pro RTC a /CS pro DataFlash).

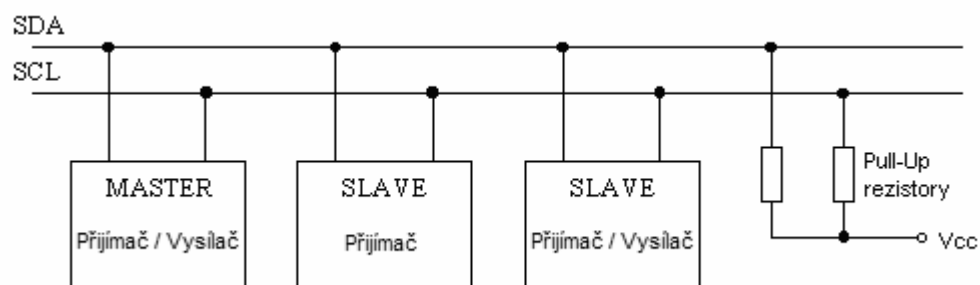


Obr. 2.1: Zapojení zařízení na SPI sběrnici [1]

V našem případě bude SPI Master mikroprocesor, SPI Slave pak DataFlash a hodiny reálného času. Signál /SS (Slave Select) často značí výrobci odlišně.

2.1.2 I²C

Další možností komunikace mezi mikroprocesorem a periferiemi je použití sběrnice I²C. Výhodou této sběrnice je nutnost použití pouze dvou vodičů a možnost připojení většího množství periferií (v základní verzi 128, v rozšířeném 10-bitovém adresování až 1024). Sběrnice I²C používá sériovou datovou linku SDA a linku hodinového signálu SCL. Data a adresy se přenášejí společně hodinovými impulsy. Obě linky je možno používat obousměrně. Jsou připojeny na pull-up rezistor a mohou být libovolným účastníkem sběrnice staženy na nízkou úroveň výstupem s otevřeným kolektorem. Neaktivní připojení účastníci mají vysokou impedanci a neustále vyhodnocují signály na sběrnici. Je-li použit jen jeden master, vydává hodinový signál pouze on. Data může vysílat master i slave.



Obr. 2.2: Zapojení zařízení na I²C sběrnici

Při použití I²C sběrnice může nastat omezení s nastavením adresy. Na sběrnici musí mít každé zařízení jedinečnou adresu. Na většině zařízení je možné nastavit pouze 2 až 3 bity adresy, někdy je už adresa určený výrobcem.

V mém případě I²C nakonec nevyužívám, protože pro mé účely dostačuje SPI linka. I²C sběrnice je jednodušší na připojení, ale o něco náročnější na programování. Tato sběrnice je spolu s SPI velmi často používaná.

2.2 STABILIZACE NAPÁJECÍHO NAPĚTÍ

Nejjednodušší je použití klasických stabilizátorů s pevně nastavenou hodnotou výstupního napětí. To umožňuje jednoduché a spolehlivé zapojení. V našem případě toto řešení vyhovuje a rozhodl jsem se ho použít.

2.3 A/D PŘEVODNÍK

V tomto případě není potřeba připojovat externí A/D převodník. Zapojení obsahuje mikroprocesor, který již integrovaný A/D převodník obsahuje. Jeho rozlišení je buď 8-mi a nebo 10-ti bitové. Pro naše účely toto řešení dostačuje.

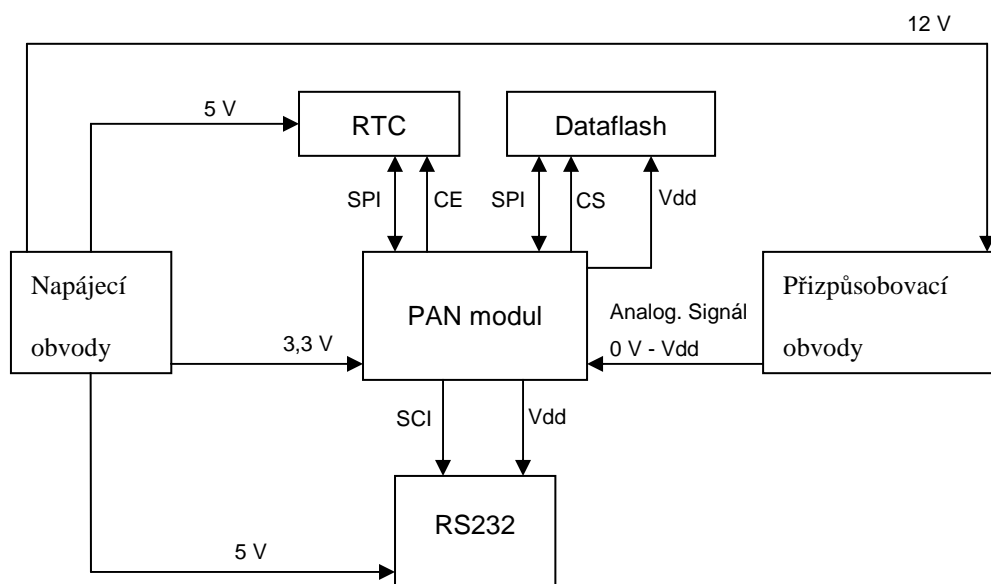
2.4 PAMĚŤ DATAFLASH

Paměť o velikosti přibližně 1 Mbit je pro naše účely dostačující. Rozhodující pro výběr paměti je, aby pracovala na stejné logické úrovni jak mikroprocesor. Svoji úlohu zde hraje i typ použité sběrnice pro přenos dat.

2.5 HODINY REÁLNÉHO ČASU (RTC)

Opět zde hraje svoji roli typ použité komunikace mezi mikroprocesorem a hodinami reálného času. Dále by měly hodiny reálného času rovněž používat spolu s mikroprocesorem stejné logické úrovně. Můžeme použít v podstatě libovolný typ vyhovující těmto požadavkům.

3. BLOKOVÉ SCHÉMA



Obr. 3.1: Blokové schéma zapojení

- | | | |
|--------|---|--|
| SPI | - | Sériová linka pro připojení periferních zařízení |
| SCI | - | Sériová linka pro komunikaci |
| CS, CE | - | Signály pro aktivaci čipů |
| RTC | - | Hodiny reálného času |

4. NÁVRH ZAPOJENÍ

4.1 NAPÁJECÍ OBVODY

4.1.1 Stabilizace napětí

V zadání je požadováno, aby zapojení obsahovalo stabilizaci napětí. Stabilizace je nezbytná pro napájení a správnou funkci jednotlivých obvodů.

Největší odběr proudu má PAN modul. PAN modul vyžaduje napětí 3,3 V a jeho maximální odebíraný proud může být až 500 mA. Tento proud je využíván pro napájení jednotlivých komponent modulu a pro napájení dalších externích obvodů, které jsou PAN modulem napájeny. Pro toto napětí jsem zvolil SMD stabilizátor LF33CDT.

Na desce se nacházejí součásti, které vyžadují i jiné napájecí napětí, než jen 3,3 V. Část z nich využívá stabilizované napětí 2,7 V z PAN modulu a dále je potřeba i napětí 5 V a 12 V. Proto jsem do schématu přidal klasické stabilizátory 7805 a 7812. Z důvodu vysokého odebrání proudu PAN modulem je výhodné zapojit jednotlivé stabilizátory kaskádovitě za sebou. Tím se ztrátové výkony rovnoměrněji rozloží. Stabilizátor LF33CDT je typu LowDrop a proto není na závadu, že je zapojen za 5 V stabilizátorem 7805. Navržené zapojení dovoluje použít trvalé napájecí napětí od 14 V do 28 V. Vyšší napětí by způsobovalo přehřívání stabilizátoru 7812, na kterém vzniká nejvyšší úbytek napětí.

PAN modul	3,3 V
OZ v přizpůsobovacích obvodech	12 V
RTC, RS232	5 V

Tab. 4.1: Napětí vyžadované jednotlivými obvody

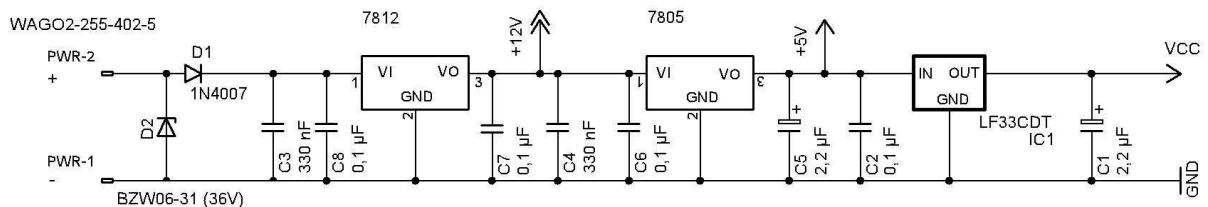
4.1.2 Ochrana proti přepólování a připojení vyššího napětí

Ochrana proti přepólování je řešena připojením standardní křemíkové diody 1N4007 v provedení SMD a zapojení do série mezi napájecí svorky a 12 V stabilizátor 7812. Úbytek napětí cca 0,7 V zde není na závadu.

Ochrana proti připojení vyššího napětí je řešena použitým transilem BZW06-31. Tento transil je připojen paralelně mezi napájecími svorkami + a -. Transil je vyroben na napětí 36 V.

Transil je součástka podobná Zenerově diodě. Rozdílem je, že transil je navrhnutý speciálně pro využití jako ochranný prvek.

Z uvedeného zapojení tedy předpokládá, že napájecí zdroj obsahuje nadproudovou ochranu.



Obr. 4.2: Schéma použitého zapojení ochran a stabilizátorů

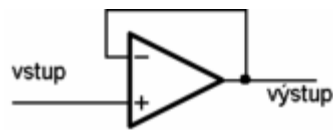
4.2 PŘÍZPŮSOVACÍ OBVODY

Obecně platí, že napěťové vstupy mají mít co největší vstupní impedanci. Vstupní impedance mikroprocesoru MC9S08GB60 v PAN modulu je 10kΩ. Toto je příliš malá impedance a je potřeba ji zvýšit. Jako impedanční přizpůsobení jsem se rozhodl použít operační zesilovač.

4.2.1 Volba operačního zesilovače

Pro přizpůsobení analogových vstupů je tedy použit operační zesilovač. Použitý typ operačního zesilovače je OPA2703UA (SMD). Operační zesilovač jsem vybíral z dodaného seznamu součástek a tento OZ vyhovoval nejvíce.

Operační zesilovač je zapojen jako sledovač napětí. V tomto zapojení má velmi vysokou vstupní impedanci rovnou impedanci samotného OZ a jeho zesílení je rovno 1. OZ je napájen napětím 12 V.



Obr. 4.3: Zapojení operačního zesilovače jako sledovače napětí.

OPA2703UA obsahuje v pouzdře dva operační zesilovače, takže bylo potřeba použít dva kusy. Je typu Rail-to-Rail, to znamená, že dokáže pracovat se vstupním a výstupním napětím, které je blízké napájecímu napětí. OPA2703UA vyhovoval v těchto parametrech: má velmi malý offset – 160 μV a požadovanou velkou vstupní impedanci – $5 \times 10^{12} \Omega$. Maximální výstupní proud je 10 mA.

4.2.2 Omezení vstupní impedance operačního zesilovače

Vstupní impedance operačních zesilovačů v zapojení jako napěťového sledovače se ideálně blíží nekonečnu. Reálně je však vstupní impedance rovna impedanci vstupu samotného operačního zesilovače. V datasheetu OZ OPA2703UA je uvedena vstupní impedance $5 \times 10^{12} \Omega = 5 \text{ T}\Omega$, což je velmi vysoká impedance.

Proto je vhodné vstupní impedanci snížit na použitelnou úroveň. To řeším přidáním odporů 10 M Ω . Odporů jsou zapojeny na vstupech paralelně se zemí. V případě ideálního operačního zesilovače by byla vstupní impedance rovna 10 M Ω . Reálně však v navrženém zapojení budou mít vstupy vstupní impedanci:

$$\frac{1}{R} = \frac{1}{R_1} + \frac{1}{R_2}$$

$$\frac{1}{R} = \frac{1}{10^6} + \frac{1}{5 \cdot 10^{12}}$$

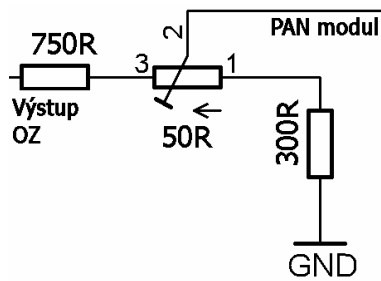
$$R = \underline{0,999 \text{ M}\Omega \cong 1 \text{ M}\Omega}$$

Vstupní impedance 1 M Ω je podle mého názoru pro připojená napětí do 10 V dostačující. Při maximálním připojeném napětí 10 V budou totiž vstupy odebírat proud 10 μA . Tento proud je dostatečně nízký a vyhovuje.

4.2.3 Návrh odporového děliče na výstupu operačního zesilovače

Operační zesilovač je v zapojení jako sledovač napětí, proto velikost výstupního napětí z operačního zesilovače je rovna vstupnímu - tj. 0 – 10 V. Vstupní napětí integrovaného AD převodníku v mikroprocesoru MC9S08GB60 je však nastaveno na 0 - Vdd. Proto jsem za operační zesilovač umístil odporový dělič.

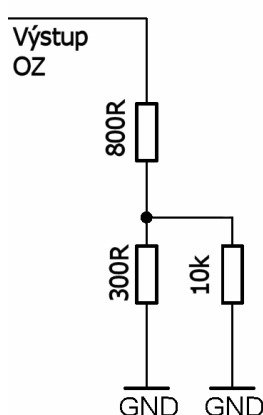
Odporový dělič jsem navrhl tak, aby při napětí 10 V jím procházel proud necelých 10 mA. Tento proud jsem zvolil s ohledem na maximální výstupní proud operačního zesilovače. Aby děličem procházel proud 10 mA, musí být jeho celkový odpor 1 k Ω .



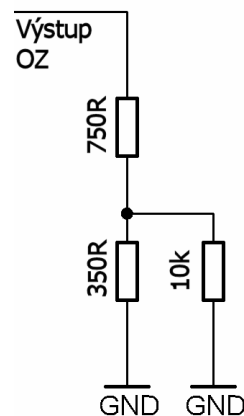
Obr. 4.4: Zapojení odporového děliče

Trimr je typ 64 W 50R SPECTROL, osa shora, 25 otáček, cermetový. Do zapojení jsem ho umístil z toho důvodu, aby bylo možné napětí vstupující do PAN modulu jemně regulovat. Použitý dělič spolu s trimrem umožňuje v případě vstupního napětí 10 V nastavit výstupní napětí na 2,7 - 3V (Vdd).

Mikroprocesor MC9S08GB60 v PAN modulu má podle údajů výrobce vstupní impedanci 10 k Ω . Odporový dělič se tedy bude chovat v krajních polohách trimru takto (Odpor 10 k Ω zastupuje impedanci mikroprocesoru):



Obr. 4.5: Levá krajní poloha trimru



Obr. 4.6: Pravá krajní poloha trimru

V levé krajní poloze trimru bude mezi zemí a vstupem odporového děliče odpor $1091,3 \Omega$. Tomu odpovídá při vstupním napětí 10 V procházející proud $9,16 \text{ mA}$. Napětí vstupující do mikroprocesoru je potom rovno $2,66 \text{ V}$ (podle nákresu odpovídá napětí na rezistorech 300R a 10k).

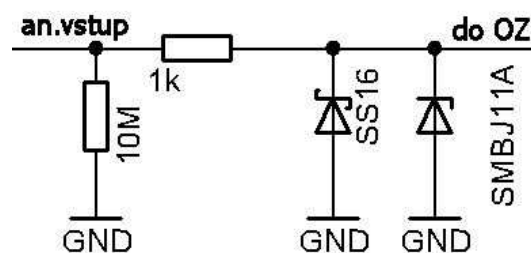
V pravé krajní poloze trimru bude mezi zemí a vstupem odporového děliče odpor $1088,2 \Omega$. Tomu odpovídá při vstupním napětí 10 V procházející proud $9,19 \text{ mA}$. Napětí vstupující do mikroprocesoru je potom rovno $3,11 \text{ V}$ (podle nákresu odpovídá napětí na rezistorech 350R a 10k).

Podle údajů výrobce může vstupující napětí do mikroprocesoru překročit $V_{dd \text{ max.}}$ o $0,3 \text{ V}$. Proto je vhodnější při nastavování trimru použít vstupní napětí max. např. 9 V aby nedocházelo v pravé krajní poloze k překročení tohoto napětí.

4.2.4 Ochrana analogových vstupů

V praxi se může stát, že se na vstupy dostane např. při poruše buď vyšší napětí než je 10 V a nebo napětí s opačnou polaritou. Proto bylo nutné vyřešit ochranu vstupů.

Ochrana je řešena následovně:



Obr. 4.7: Schéma ochrany analogového vstupu

Samotný operační zesilovač má v sobě integrovanou ESD diodu, která chrání jeho vstupy proti krátkodobému přivedení vyššího napětí. To ale není pro naše účely dostatečná ochrana.

Jak je možné vidět ve schématu, přidal jsem mezi analogový vstup a operační zesilovač do série zapojený odpor 1 k Ω a za ním paralelně se zemí Schottkyho diodu SS16 a 11 V transil SMBJ11A. V případě připojení nevhodného napětí vznikne na odporu 1 k Ω úbytek, který zabrání poškození operačního zesilovače.

4.3 A/D PŘEVODNÍK

Jak již bylo uvedeno, přizpůsobený analogový signál je přiveden přímo na port B mikroprocesoru MC9S08GB60 v PAN modulu. Tím nám odpadá starost s výběrem vhodného A/D převodníku, neboť tento mikroprocesor již obsahuje integrovaný A/D převodník, který našim požadavkům vyhovuje. Jeho rozlišení je buď 8 bitů (256 úrovní) a nebo 10 bitů (1024 úrovní). Pro 8 bitů odpovídá jedné úrovni 39,06 mV, pro 10 bitů odpovídá jedné úrovni 9,76 mV. Z důvodu vyšší přesnosti proto využívám 10 bitového převodu.

Dalším faktorem ovlivňující přesnost převodu je referenční napětí přivedené na pin mikroprocesoru. Tento pin by měl být nejlépe přiveden na externí zdroj referenčního napětí, avšak v použitém zapojení je vyveden přímo na Vdd. Takové řešení sice ztrácí na přesnosti, ale je stále použitelné. Ponechal jsem jej, protože takto byl již vytvořen návrh v rámci semestrálního projektu. Případné korekce lze provést trimry u analogových vstupů.

4.3.1 Přepočítání hodnoty z převodníku na napětí

V případě potřeby převodu z hodnoty A/D převodníku na úroveň napětí přivedeného na analogový vstup, použijeme tento vzorec:

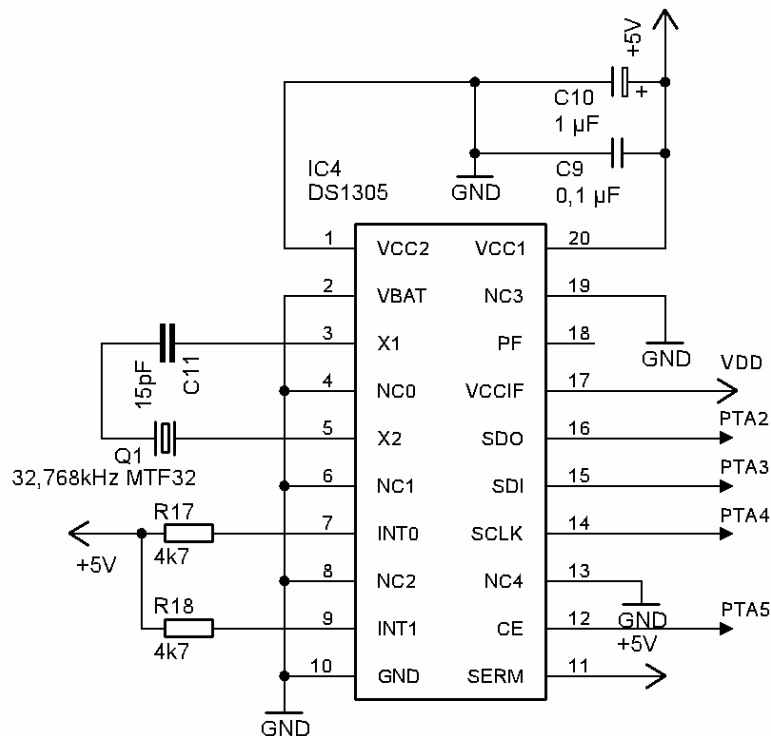
$$U = \frac{D}{1023} \times 10 \quad (4.1)$$

kde D.....10-ti bitová hodnota z A/D převodníku

U.....vypočítané napětí připojené na an. vstup

4.4 HODINY REÁLNÉHO ČASU (RTC)

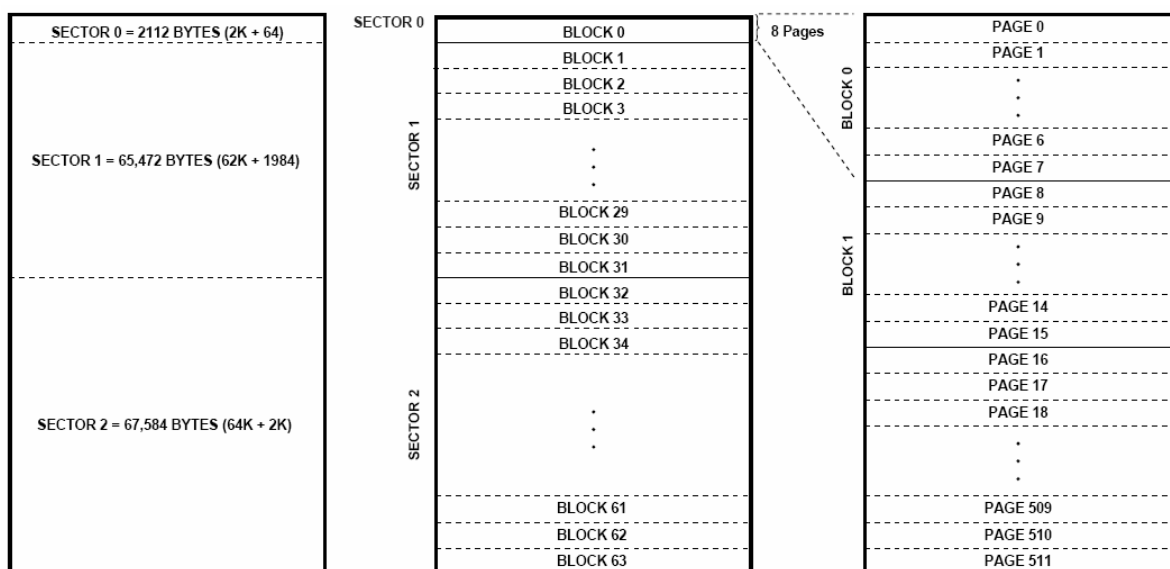
Hodiny reálného času jsem realizoval obvodem DS1305. Tento obvod počítá sekundy, minuty, hodiny, den, měsíc, den v týdnu a rok. Počítá různým počtem dnů v měsíci i s přestupnými roky. Jeho vnitřní paměť pro ukládání dat je 96 Bytů NV RAM (Non-volatile random access memory). Tato paměť neztrácí uložené data po vypnutí napájení. Dále obvod obsahuje dva časově řízené alarmy, které mohou generovat přerušení. Tato funkce není v našem případě využívána. Dále je možnost připojení záložní baterie (pin VBAT). I tato funkce zůstala nevyužita. Obvod komunikuje pomocí 3-vodičového nebo SPI rozhraní, které nastavujeme pinem SERMODE. V zapojení využívám SPI rozhraní, proto je tento pin připojen na 5 V. Pro přesný chod hodin by měl mít připojený krystal kapacitu 6 pF, avšak použitý typ má kapacitu 10 pF. Proto jsem pro zvýšení přesnosti ke krystalu připojil sériově kondenzátor 15 pF, který krystal kapacitně přizpůsobuje.



Obr. 4.8: Zapojení hodin reálného času DS1305

4.5 PAMĚŤ DATAFLASH

Paměť DataFlash je v zapojení vyřešena integrovaným obvodem AT45DB. Tato paměť pracuje s napětím pouze 2,7 V. Vstupy a výstupy můžeme připojit na porty mikroprocesoru přímo, protože logika funguje na stejné úrovni. Napájení je vyvedeno přímo z PAN modulu (Vdd). Tento integrovaný obvod komunikuje s mikroprocesorem rovněž po sériové SPI lince. Jak již bylo uvedeno, paměť sdílí sériovou SPI linku společně s hodinami reálného času.



Obr. 4.9: Rozložení paměti [2]

Paměť obsahuje celkem 1081344 bitů. Ty jsou rozděleny celkem do 512 stránek po 264 Byte. Dále obsahuje zásobník dat o velikost 256 Bytů statické RAM (S RAM). Tento zásobník umožňuje ukládání dat v průběhu změny stránky. Jak lze vidět na diagramu rozložení paměti, každý sektor obsahuje několik bloků a jednotlivé bloky obsahují stránky. K paměti se přistupuje po stránkách, ale některé mazací operace umožňují přístup i na jednotlivý blok nebo sektor.

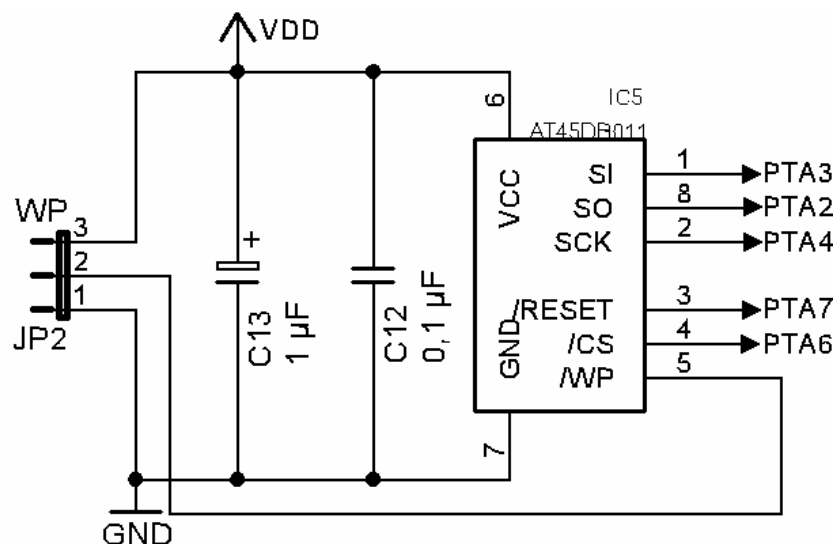
Jednotlivé operace jsou řízeny procesorem. Každá instrukce začíná sestupnou hranou na signálu /CS. Pokud je signál /CS v logické nule, překlápění SCK řídí načítání operačního kódu. Dále následuje adresa v paměti. Operace končí náběžnou hranou na /CS.

Čtení stránky paměti začíná osmibitovým operačním kódem 52H, následuje 24 adresových bitů (6 bitů je určeno pro kompatibilitu, 9 bitů PA8-PA0 udávají adresu stránky a posledních 9 bitů BA8-BA0 udávají adresu bajtu kde má čtení začít a 32 bitů don't care (tyto bity mohou být jakékoliv, doporučeny jsou logické nuly). Tyto bity inicializují operace. Poté už následují požadovaná data posílaná na pin SO (Seriál Output).

Zápis stránky (přes zásobník) začíná osmibitovým operačním kódem 82H, následuje 6 rezervovaných bitů, dále 9 adresních bitů PA8-PA0 a 9 adresních bitů BA9-BA0. Na pin SI následně posíláme data, která se ukládají do zásobníku. Do hlavní paměti se data zapíše, jakmile se zásobník zaplní.

Syntaxe dalších operací čtení nebo zápisu jsou podobné. Mění se jen operační kód. Jednotlivé operační kódy lze nalézt v katalogových listech výrobce.

Dále je možné zabránit zápisu do prvních 256 stránek paměti přivedením logické nuly na pin /WP (Write Protect). Nastavení tohoto pinu lze nastavit pomocí jumperu umístěného na plošné desce.



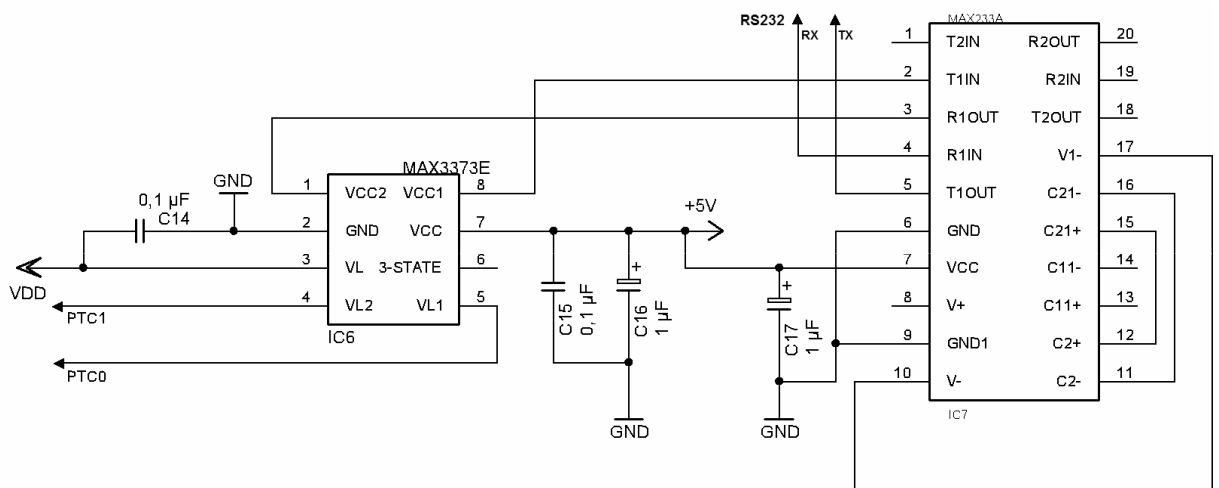
Obr. 4.10: Zapojení paměti DataFlash AT45DB011

4.6 ROZHRANÍ RS232

Mikroprocesor v PAN modulu pracuje s 3 V logikou, ale řadiče RS232 pracují s 5 V logikou. Proto je nutné mezi mikroprocesor a řadič vložit převodník úrovní. V tomto případě jsem použil převodník MAX3373E.

Tento převodník umožňuje obousměrný převod úrovní. Poskytuje galvanické oddělení až do 15kV a tepelnou ochranu. Obsahuje napěťové vývody VL (nižší napětí) a VCC (vyšší napětí) a poskytuje převod mezi napětími na těchto vývodech. Napětí VL může dosahovat hodnot +1,2 V – 5,5 V a napětí VCC 1,65 V – 5,5 V, což je vyhovující.

K převodníku je připojen řadič rozhraní RS232. Použitý řadič je MAX 233A. Tento řadič zprostředkovává přenos mezi 5 V logikou na rozhraní RS232. Obsahuje dva přijímače a dva vysílače RS232. V našem případě využíváme jen jednoho vysílače a přijímače, druhý není zapojen. Výhodou použitého řadiče, že je navržen pro co jednodušší zapojení. Jeho napájecí napětí je pouze 5 V a potřebné napětí ± 12 V si vytváří sám integrovaným DC/DC měničem. Rozhraní RS232 je vyvedeno na standardní 9-ti pinový konektor CANNON.



Obr. 4.11: Zapojení převodníku MAX3373E a RS232 řadiče MAX 233A

5. SOFTWARE

Software pro mikroprocesor byl napsán jazykem C v prostředí Metrowerks CodeWarrior IDE. Jako základ softwaru byl použit volně stažitelný kód SMAC v. 4.0, konkrétně aplikace Range Demo Plus RX.

Použitím této aplikace nám odpadá nutnost mnoha nastavení. Tento kód je kompatibilní s námi použitým hardwarem. Je zde již také vyřešená rádiová komunikace a komunikace mezi mikroprocesorem a rádiovým čipem. Do této aplikace stačí jen přidat potřebné funkce a vlastní program pro analogový vstup.

Aplikace se skládá s několika hlavičkových souborů:

5.1 SMAC_MC9S08GT60.H

V tomto hlavičkovém souboru jsou obsaženy definice I/O portů a registrů mikroprocesoru MC9S08GT60.

5.2 SPI.H

V tomto hlavičkovém souboru jsou vytvořeny rutiny pro posílání a příjem dat a řídicích signálů na sběrnici SPI. Kromě toho obsahuje definice funkcí pro komunikaci s hodinami reálného času a pamětí dat.

5.2.1 Popis chodu SPI sběrnice

Při komunikaci po SPI je mikroprocesor nastaven jako MASTER a ostatní zařízení jako SLAVE. MASTER vytváří hodinový signál CLK a současně určuje, se kterým zařízením momentálně komunikuje. Výběr zařízení se provádí zvláštním vodičem pro výběr čipu (CE,CS). Poté aktivaci zařízení následuje operační kód, který určuje jaká operace se má provést. Poté již MASTER data buď čte a nebo posílá. Žádný potvrzující bit u sběrnice SPI není. Rychlost sběrnice je pak omezena nejpomalejším zařízením, popř. rychlostí mikroprocesoru.

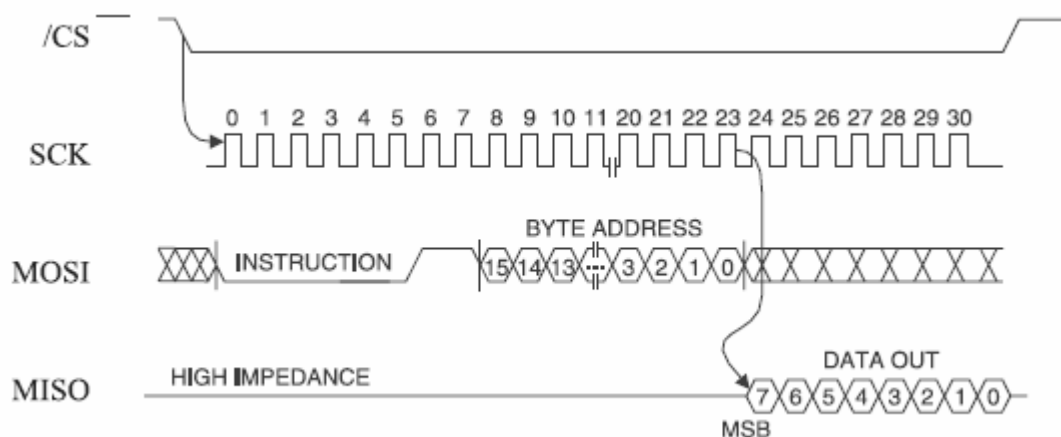
V klidovém stavu bývá obvykle vodič SDO (výstup se zařízení) ve vysoké impedanci a vodič SDI (vstup do zařízení) buď v logické 0, nebo logické 1.

Přenos po sběrnici SPI se skládá z těchto stavů:

Chip enabled – Start komunikace se zařízením. Podle typu zařízení může být zařízení aktivováno buď v log. 1 nebo 0.

Vlastní přenos dat – Data se posílají po 1 byte. Logické úrovně na SDI a SDO se mění vždy ve stavu CLK = 0 nebo CLK = 1. Stav kdy je CLK aktivní určuje moment při aktivaci čipu.

Chip disabled – Konec komunikace se zařízením. Na vodiči aktivujícím zařízení nastane opačná logická úroveň oproti Chip enabled.



obr. 5.1: Ukázka časového průběhu log. úrovní na SPI (zápis do paměti) [3]

5.2.2 Popis jednotlivých funkcí v souboru SPI.H

V tomto souboru jsou nadefinovány vývody pro komunikaci s DataFlash a hodinami reálného času (RTC). Dále jsou v tomto souboru definice funkcí pro ovládání SPI sběrnice. Samotné funkce jsou pak v souboru SPI.C.

Jedná se o tyto funkce:

void SPI_setup(unsigned int *page_write,unsigned int *buffaddr)

Nastavení linky, směru vývodů linky, vypnutí obou zařízení připojených přes linku a načtení poslední adresy, na kterou byly zapsány data. (Např. při výpadku napájení)

unsigned char SPI_in(void)

Vrací 1 byte přečtený do mikroprocesoru ze zařízení.

void SPI_out(unsigned char spidata)

Posílá 1 byte v spidata z mikroprocesoru do zařízení.

Dále soubor obsahuje definici funkcí pro ovládání hodin reálného času. Funkce jsou uloženy v souboru RTC.C.

Jsou to tyto funkce:

void RTC_setup(void);

Nastavení řídicího registru hodin. Tímto se povolí zápis do registru hodin a dojde ke spuštění oscilátoru.

void RTC_read_time(unsigned char *time)

Uložení aktuálního času do proměnné.

void RTC_set_time(unsigned char *time)

Nastavení času z hodnot uložených v proměnné.

V souboru SPI.H jsou také definice funkcí pro práci s DataFlash. Samotné funkce jsou uloženy v souboru DF.C.

Tyto funkce jsou:

unsigned char DF_buffer_read(unsigned int buffaddr)

Načte 1 byte z bufferu na adrese buffaddr.

void DF_buffer_write(unsigned int buffaddr, unsigned char bcislo)

Zapíše 1 byte uložený v proměnné bcislo na adresu buffaddr v bufferu.

void DF_memory_erase(void)

Maže celou paměť dat.

void DF_page_to_buffer(unsigned int page_read)

Uloží stránku na adrese page_read do bufferu.

void DF_buffer_to_page(unsigned int *page_write, unsigned int *buffaddr)

Uložení bufferu do stránky v paměti na adrese page_write.

**void DF_record_to_buffer(unsigned int *buffaddr, unsigned char *data_record,
unsigned int *page_write)**

Uložení jednoho záznamu do bufferu.

void DF_memory_read(unsigned int *page_write, unsigned int *buffaddr)

Načte a odešle celý obsah zatím zapsané paměti.

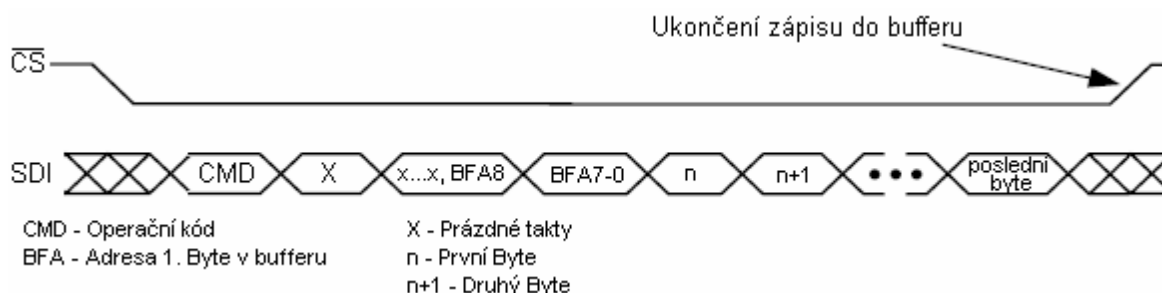
5.2.3 Průběh komunikace s Pamětí dat pomocí sběrnice SPI

DataFlash dokáže komunikovat rychlostí až 20 MHz. Protože paměť sdílí SPI linku s hodinami reálného času, musí se rychlost přizpůsobit pomalejšímu zařízení.

Komunikace s pamětí se zahájí nastavením /CE do logické nuly. Tím se paměť připraví pro příjem dat .

Zápis dat do bufferu

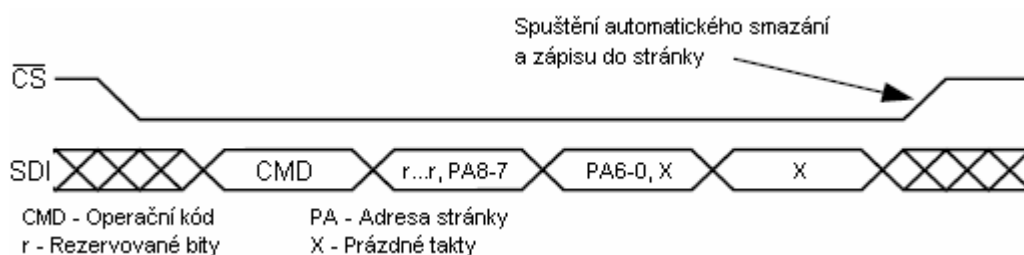
Nejprve se vyšle operační kód 84H po kterém následuje 15 prázdných taktů. Poté následuje 9 adresních bitů, které definují kam bude v bufferu zapsán první byte. Data se budou zapisovat od tohoto byte až do dokončení komunikace. Zápis dat do bufferu je pak ukončen log. 1 na vodiči /CS.



Obr. 5.2: Průběh zápisu dat do bufferu [2]

Zápis bufferu do stránky v paměti

Nejprve se vyšle operační kód 83H, který je následován 6 rezervovanými bity. Dále se posílá 9 adresových bitů určujících stránku v paměti, do které má být buffer zapsán. Nakonec se posílá 9 prázdných hodinových taktů. Zápis začne ihned po nastavení vodiče /CS do logické 1.

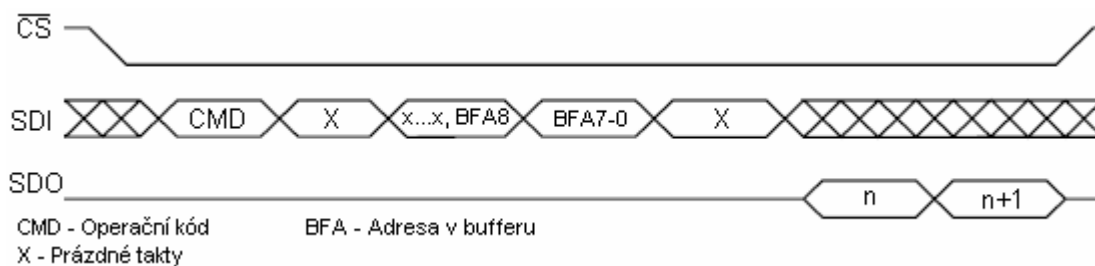


Obr. 5.3: Průběh zápisu bufferu do paměti [2]

Načtení dat z bufferu

Nejprve vyšleme 8-mi bitový operační kód 52H, po kterém musí následovat 15 prázdných taktů. Dále následuje 9 bitů adresy, od které budeme číst. Následuje 8

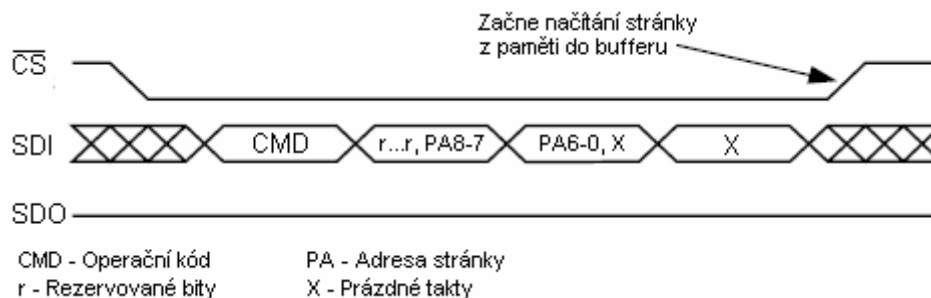
prázdných taktů. Po dobu čtení je vodič /CE v log 0. Jeho nastavením do log 1 se čtení ukončí.



Obr. 5.4: Průběh načítání dat z bufferu [2]

Načtení stránky z paměti do bufferu

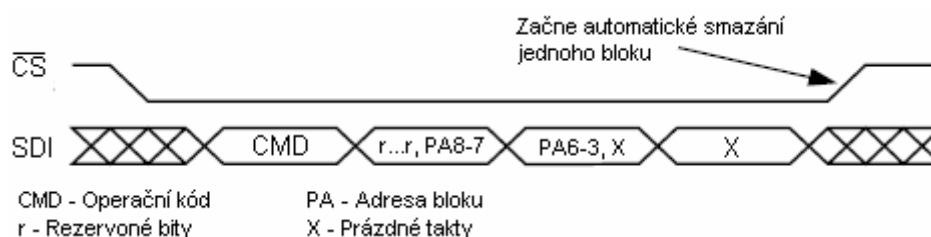
Pro načtení stránky z paměti do bufferu musíme poslat operační kód 53H, následovaný 6 rezervovanými bity. Následuj 9 bitů určujících adresu stránky, kterou budeme do bufferu načítat. Následuje 9 prázdných taktů. Načítání začne ve chvíli, kdy se na vodiči /CS objeví log. 1.



Obr. 5.5: Načtení stránky z paměti do bufferu [2]

Smazání bloku v paměti

Pro smazání bloku (8 stránek) v paměti, vyšleme operační kód 50H, následovaný 6-ti rezervovanými bity. Následně 6-ti bitovou adresou určíme blok, který chceme smazat. Instrukce je poté zakončena 12-ti prázdnými takty. Mazání začne po změně na log. 1 na vodiči /CS.



Obr. 5.6: Smazání bloku v paměti [2]

Čtení status registru :

Pošleme operační kód 57H a okamžitě po zadání se na výstupu objeví několik bitů. Nejvyšší bit určuje stav, v jakém se zařízení nachází. Tím zjistíme, jestli je paměť připravena a nebo dokončuje poslední operaci. Operace jako je zápis bufferu do paměti, mazání bloku a nebo načtení stránky z paměti do bufferu způsobují určitou časovou prodlevu. Z toho důvodu je výhodné pomocí status registru stav zařízení kontrolovat.

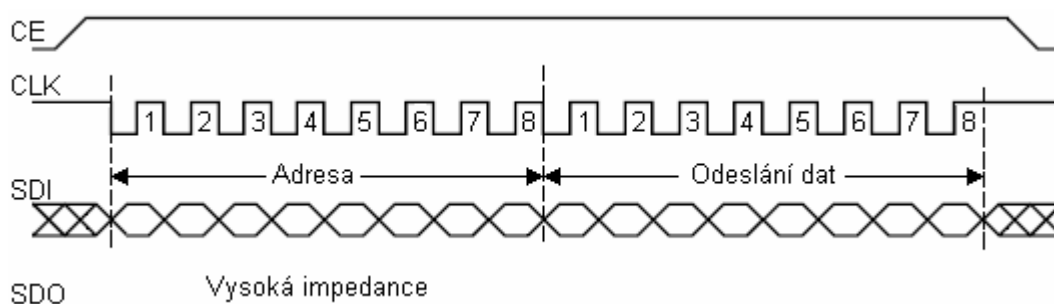
5.2.4 Průběh komunikace s hodinami reálného času (RTC)

Hodiny reálného času mohou komunikovat do rychlosti 2 MHz. Protože se jedná o pomalejší rychlost oproti paměti DataFlash, rychlost SPI sběrnice musíme nastavit podle této rychlosti.

Pro start komunikace s použitým obvodem RTC musí na vodiči CS nastat log 1. Poté posíláme adresu registru ze kterého čteme, popřípadě zapisujeme. Čtením registru zjišťujeme aktuální čas a datum, zápisem do registru čas a datum nastavujeme. Povolení zápisu a spuštění oscilátoru je nejprve nutné nastavit v řídicím registru. Pro oscilátor Bit 7, pro zápis Bit 6.

Zápis dat do hodin

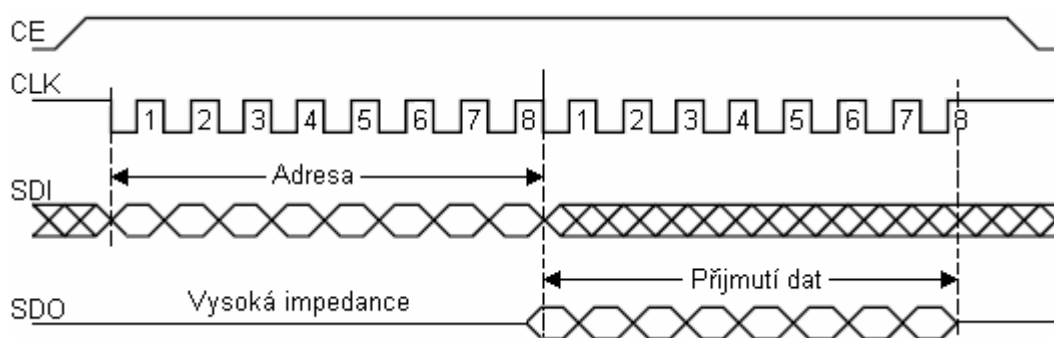
Při zápisu hodnoty do hodin posíláme první adresu kam budeme zapisovat, následovanou zapisovanou hodnotou. Pro úspěšný zápis dat musí být úroveň vodiče CS v log. 1 po celou dobu komunikace.



Obr. 5.7: Zápisi hodnoty do hodin reálného času [4]

Čtení dat z hodin

Pro čtení dat z hodin opět nejprve posíláme adresu od které chceme data vyčíst. Poté začnou hodiny reálného času posílat data.



Obr. 5.8: Načtení hodnoty z hodin reálného času [4]

Tabulka 5.1: Adresy registrů hodin reálného času [4]

Adresa HEX		Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Rozsah
Čtení	Zápis									
00H	80H	0	10-sek			sekundy			0 – 59	
01H	81H	0	10-min			minuty			0 – 59	
02H	82H	0	12	PM	10-h	hodiny			01-12 +	

				AM						
			24	10						0 – 23
04H	84H	0	0	10-den			dny			1 – 31
05H	85H	0	0	10-měsíc			měsíce			1 – 12
06H	86H	0		10-rok			roky			00 – 99
0FH	8FH	Control registr								-----
		EOSC	WP	0	0	0	INT	AIE1	AIE0	

5.3 POPIS SOUBORU ADC.H

V tomto souboru jsou definovány výsledkové registry a definována funkce pro přečtení hodnoty ze vstupu na A/D převodníku v mikroprocesoru.

unsigned int AD_control_status_setup(unsigned char kanal)

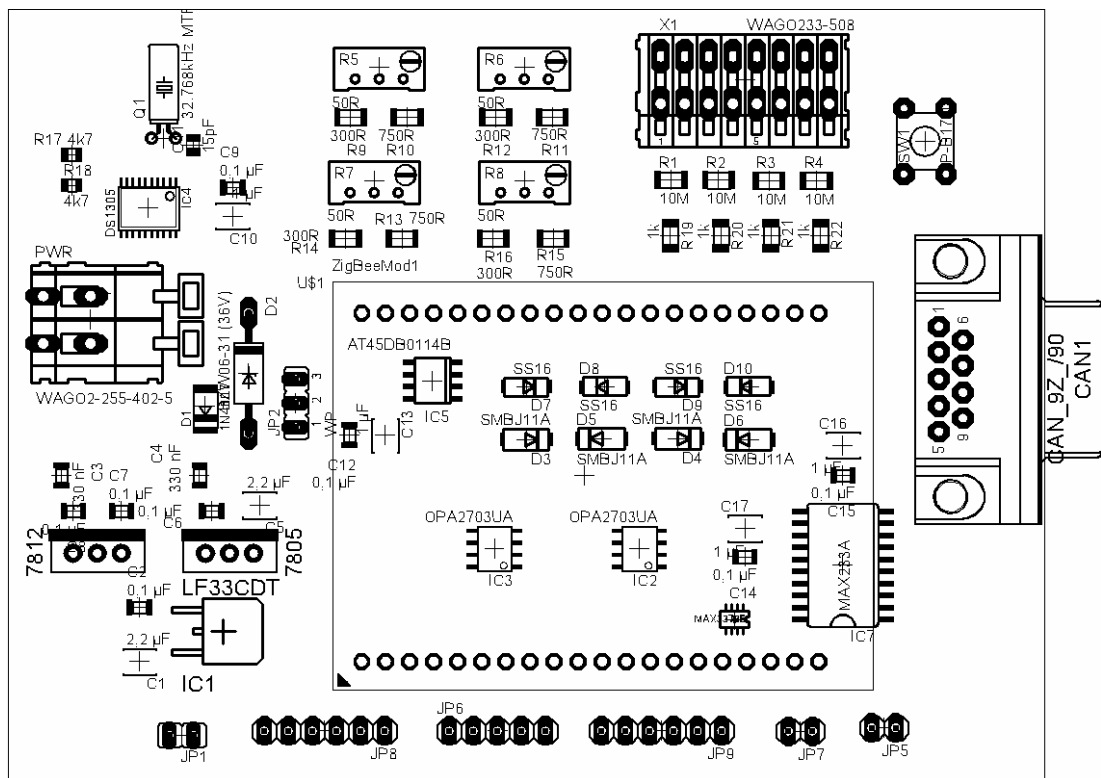
Vstupem této funkce je číslo kanálu (00 až 03), ze kterého budeme číst hodnotu. Převodník je nastaven jako 10-ti bitový. Výstupem je pak číslo 0 – 1023 v závislosti na velikosti napětí připojeného na analogový vstup (0 – 10 V).

Tato funkce je vytvořena v souboru ADC.c.

6. UŽIVATELSKÝ MANUÁL

Uživatelský manuál obsahuje popis připojení desky k napájecímu napětí, analogových vstupů a význam propojek. Dále obsahuje formát a definici příkazů pro komunikaci přes S-MAC.

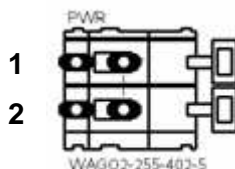
Rozmístění prvků na desce:



Obr. 6.1: Rozmístění prvků na desce

6.1 PŘIPOJENÍ NAPÁJECÍHO NAPĚTÍ

Napětí se přivádí na WAGO svorky. Přiváděné napětí by se mělo pohybovat v rozmezí 14 V až 28 V.



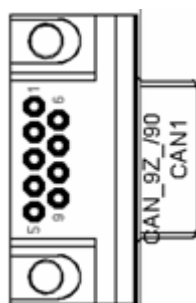
Obr 6.2: WAGO svorka

Pin	Signál
1	GND
2	Napájení (14 – 28 V)

Tabulka 6.1: Napájení - WAGO svorka

6.2 KONEKTOR CAN1

Tímto konektorem je možné modul spojit s jiným modulem nebo počítačem pomocí sériové linky RS232.



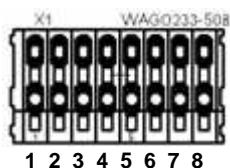
Pin	Signál
1, 4, 6	Propojeno
2	Tx
3	Rx
5	GND
7, 8	Propojeno
9	Nepřipojeno

Obr. 6.3: CANNON

Tabulka 6.2: Zapojení konektoru CANNON

6.3 ZAPOJENÍ ANALOGOVÝCH VSTUPŮ

Tyto svorky jsou určeny k připojení čtyř analogových vstupů 0 – 10 V.



Obr. 6.4: WAGO konektor an. vstupů

Vývod	Kanál	Signál
1	00	Napěťový vstup 1 (0-10 V)
2		GND
3	01	Napěťový vstup 2 (0-10 V)
4		GND
5	02	Napěťový vstup 3 (0-10 V)
6		GND
7	03	Napěťový vstup 4 (0-10 V)
8		GND

Tabulka 6.3: Zapojení svorky analogových vstupů

6.4 PROPOJKY

Pomocí propojek je možno zapnout ochranu proti zápisu do paměti DataFlash a nebo měřit odebíraný proud PAN modulem.



Propojka	Zapojení	Význam
JP1	ON/OFF	Možnost měřit proud odebíraný PAN modulem.
JP2	1 – 2	Zápis do DataFlash zakázán
	2 – 3	Zápis do DataFlash povolen

Obr. 6.5: JP1, JP3 Tabulka 6.4: Zapojení propojek

6.5 KONEKTORY JP5 - JP9

Na těchto konektorech jsou vyvedeny vývody z mikroprocesoru. Mohou sloužit pro diagnostické účely a nebo pro budoucí využití.

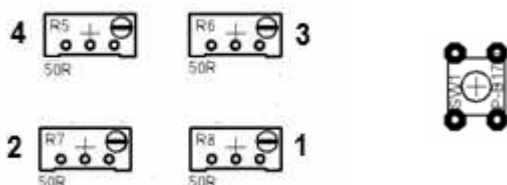


Obr. 6.6: Konektory JP5 – JP9

Konektor	Vývod	Připojen na
JP5	1	PTE1
	2	PTE0
JP6	1	PTD4
	2	PTD3
	3	PTD2
JP7	4	PTD1
	5	PTD0
	6	PTD0
JP7	1	PTC1
	2	PTC0
JP8	1	PTB7
	2	PTB5
	3	PTB3
	4	PTB2
	5	PTB1
JP9	6	PTB0
	1	PTA7
	2	PTA6
	3	PTA5
	4	PTA4
	5	PTA3
6	PTA2	

Tabulka 6.5: Zapojení konektorů JP5 – JP9

6.6 TRIMRY A MIKROSPÍNAČE



Obr. 6.7: Trimry na desce plošných spojů

Trimry na desce slouží k jemnému doladění hodnoty napětí přiváděné na analogový vstup.

Trimr	Číslo analogového vstupu	Kanál
R8	1	00
R7	2	01
R6	3	02
R5	4	03

Tabulka 6.6: Popis trimrů na desce

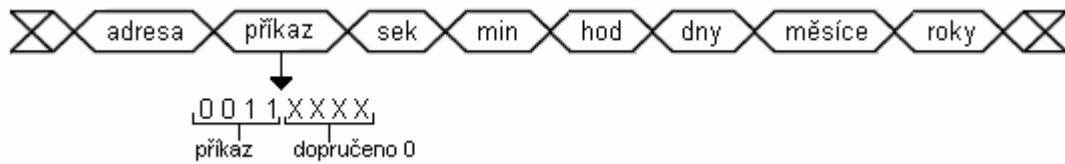
Dále se na desce nachází mikropínač SW1. Ten slouží resetu mikroprocesoru.

6.7 POPIS PŘÍKAZŮ A JEJICH FORMÁT PŘI KOMUNIKACI PŘES SMAC

Každý příkaz nejprve začíná adresou zařízení. Dále následuje 4-bitový operační kód, určující jaká operace se má provést. Další čtyři bity z bajtu mohou být buď data nebo mohou být nulové. Poté jsou poslána potřebná data k dané operaci.

6.7.1 Nastavení data a času

Operační kód pro tento příkaz je 1. Další 4 bity nejsou využity. Pak následuje 6 byte s informacemi o datu a čase.

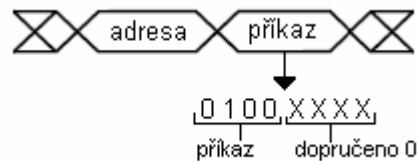


Obr. 6.8: Formát paketu pro nastavení data a času [9]

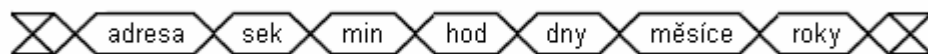
Datum a čas zadáváme v BCD kódu.

6.7.2 Přečtení data a času

Operační kód pro tento příkaz je 2. Poté program začne posílat datum a čas přes anténu.



Obr. 6.9: Formát paketu pro přečtení data a času

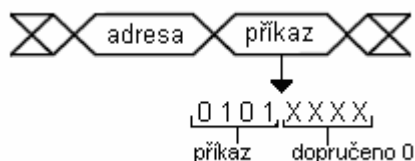


Obr. : Formát paketu odesílaného po příkazu k přečtení data a času [9]

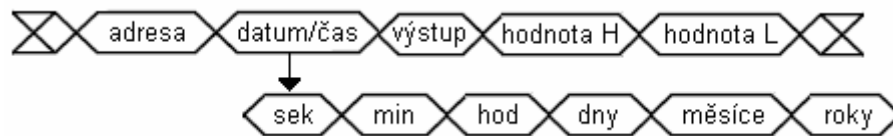
Adresa značí, ze kterého zařízení jsou tyto data posílány.

6.7.3 Přečtení obsahu paměti dat

Operační kód pro tento příkaz je 3. Dále program začne posílat data po 10 bytech přes anténu.



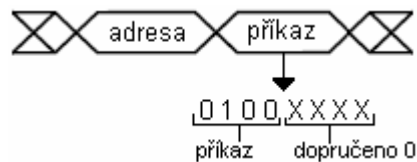
Obr. 6.10 : Formát paketu pro přečtení dat z paměti [9]



Obr. 6.11: Formát paketu odesílaného po příkazu k přečtení dat z paměti [9]

6.7.4 Mazání paměti

Po poslání operačního kódu 4 se celá paměť vymaže.



Obr. 6.12: Formát paketu pro smazání paměti [9]

6.7.5 Přečtení hodnoty z analogového vstupu

Pro přečtení hodnoty z analogového vstupu je potřeba poslat operační kód 5, následovaný dvěma bity (hodnota 00 – 03) určujícími číslo kanálu. Poté se zjištěná hodnota uloží do paměti dat. Data z paměti je pak možné poslat jiným příkazem.

7. ZÁVĚR

V této práci jsem úspěšně navrhl schéma zapojení i desku plošných spojů čtyř analogových vstupů 0 – 10 V pro ZigBee modul vytvořený na Ústavu automatizace a měřicí elektrotechniky.

Navržené zapojení obsahuje stabilizaci napětí, přizpůsobovací obvody, hodiny reálného času, paměť DataFlash a rozhraní RS232. Pro komunikaci mezi procesorem a periferiemi byla zvolena sběrnice SPI. Použití pouze této sběrnice se jevilo jako nejvýhodnější. Zařízení může být napájeno napětím od 14 V do 26 V.

Během sestavování zařízení byla nalezena drobná chyba v desce plošných spojů, která pravděpodobně nastala díky špatně vytvořené knihovně v návrhovém systému Eagle. V důsledku této chyby nebyly otvory pro vývody trimrů dostatečně velké a musely být dodatečně opraveny.

Vytvořené zařízení pracuje bez problémů a během provozu nenastaly žádné komplikace.

Vytvořené zařízení je možné ovládat bezdrátově přes SMAC. Přes rozhraní RS232 není prozatím možné komunikovat, protože nebylo z časových důvodů naprogramováno. Software je vytvořený v jazyce C, v prostředí Metrowerks CodeWarrior IDE v. 5.2.1149.

8. POUŽITÁ LITERATURA

- [1] Otevřená encyklopedie Wikipedia [online]. [2007-] [cit. 2007-05-30].
Dostupný z WWW: <<http://cs.wikipedia.org/wiki/SPI>>.
- [2] ATMEL. AT45DB011B DataFlash. [cit. 2007-05-30]. Dostupné na WWW:
<<http://www.atmel.com>>.
- [3] DUDÁČEK, K.. Sériová rozhraní SPI, Microwire, I2C a CAN [online]. 2002
[cit. 2007-05-30]. Dostupný z WWW:
http://home.zcu.cz/~dudacek/NMS/Seriova_rozhrani.pdf
- [4] DALLAS SEMICONDUCTOR. DS1305 Real-time clock. [cit. 2007-05-30].
Dostupné na WWW: <<http://www.maxim-ic.com>>.
- [5] DALLAS SEMICONDUCTOR. MAX233A RS232 Drivers/Receivers. [cit.
2007-05-30] . Dostupné na WWW: <<http://www.maxim-ic.com>>.
- [6] DALLAS SEMICONDUCTOR. MAX3373E Level translator. [cit. 2007-05-
30]. Dostupné na WWW: <<http://www.maxim-ic.com>>.
- [7] FREESCALE MOTOROLA. MC9S08GTxx HCs08 microcontrollers. [cit.
2007-05-30]. Dostupné na WWW: <<http://www.freescale.com>>.
- [8] STRATIL, Pavel. Návrh Zigbee zařízení - Analogový výstup 0-10 V. [s.l.],
2006. 41 s. Vedoucí bakalářské práce Ing. Radek Štohl, Ph.D.

9. PŘÍLOHY

1. Schéma zapojení
2. Deska plošných spojů
3. Osazovací výkres