

Výukový modul pro předmět Programování mikropočítačů: obvod reálného času

Educational Module for Microcontrollers Programming Course:
Real Time Clock

Libor Smýkal



Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

akademický rok: 2012/2013

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Libor SMÝKAL**

Osobní číslo: **A10802**

Studijní program: **B3902 Inženýrská informatika**

Studijní obor: **Informační a řídicí technologie**

Forma studia: **prezenční**

Téma práce: **Výukový modul pro předmět Programování
mikropočítačů: obvod reálného času**

Zásady pro vypracování:

1. Prostudujte hardwarové vlastnosti vývojového kitu M68EVB908GB60 a navrhnete způsob připojení modulu reálného času.
2. Provedte hardwarový návrh modulu reálného času na bázi obvodu MCP7940M ovládaného prostřednictvím IIC sběrnice.
3. Realizujte prototyp modulu a ověřte jeho funkci na vývojovém kitu.
4. Vytvořte podpůrné programové vybavení pro obsluhu modulu ve formě knihovny podprogramů v jazyce symbolických adres a v C jazyce.
5. Vytvořte ukázkovou aplikaci s využitím realizované knihovny podprogramů, která bude demonstrovat funkci modulu.

Rozsah bakalářské práce:

Rozsah příloh:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

1. AXIOM MANUFACTURING. M68EVB908GB60 Development Board for Freescale MC9S08GB60, Rev. C [online]. 2006. Dostupné z: [www.axman.com]
2. FREESCALE SEMICONDUCTOR. CPU08 Central Processor Unit Reference Manual [online]. 2001. Dostupné z: [www.freescale.com]
3. FREESCALE SEMICONDUCTOR. HCS08 Family Reference Manual, Rev.1. [online]. 2003. Dostupné z: [www.freescale.com]
4. FREESCALE SEMICONDUCTOR. MC9S08GB/GT Data Sheet, Rev.2.3. [online]. 2004. Dostupné z: [www.freescale.com]
5. JURÁNEK, Antonín a Miroslav HRABOVSKÝ. EAGLE pro začátečníky /: uživatelská a referenční příručka :. 2. vydání. Praha: BEN – technická literatura, 2007, 191 s. ISBN 80-730-0213-2.
6. MICROCHIP. MCP7940M Low-Cost I2C Real-Time Clock/Calendar with SRAM [online]. 2011. Dostupné z: [www.microchip.com]

Vedoucí bakalářské práce:

Ing. Petr Dostálek, Ph.D.

Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce:

24. února 2013


Termín odevzdání bakalářské práce:

14. června 2013

Ve Zlíně dne 24. února 2013


prof. Ing. Vladimír Vašek, CSc.
děkan




prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Cílem této bakalářské práce bylo navrhnout a vyrobit výukový modul s obvodem reálného času. Jako obvod reálného času byl vybrán integrovaný obvod MCP7940M, který umožňuje komunikaci s vývojovým kitem přes I2C sběrnici.

V teoretické části je popsán vývojový kit s mikropočítačem MC9S08GB60. Dále jsou popsány způsoby jak čas měřit a samotný obvod reálného času.

V praktické části je popsán návrh výukového modulu. Na závěr je popsána knihovna funkcí pro práci s modulem a ukázkový program.

Klíčová slova: MCP7940M, obvod reálného času, mikropočítač, MC9S08GB60, I2C

ABSTRACT

The aim of this bachelor thesis was to design and produce education module with real-time clock. As the real-time clock was used MCP7940M integrated circuit that enables communication with the development board through the I2C bus.

In the theoretical part is described the development board with microcomputer MC9S08GB60. Then I describe the ways to measure time and actual real-time clock.

The practical part describes the design of the education module and finally a library of functions for working with the module and demonstration program.

Keywords: MCP7940M, real time clock, microcomputer MC9S08GB60, I2C

Rád bych poděkovat mému vedoucímu práce, Ing. Petru Dostálkovi, Ph.D. za jeho ochotu, pomoc, rady a připomínky při řešení této práce.

Prohlašuji, že

- beru na vědomí, že odevzdáním bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze bakalářské práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně

.....
podpis diplomanta

OBSAH

ÚVOD.....	9
I TEORETICKÁ ČÁST	10
1 MIKROPOČÍTAČ A VÝVOJOVÝ KIT	11
1.1 MIKROPOČÍTAČ MC9S08GB60	11
1.1.1 Parametry.....	11
1.1.2 Registry	11
1.1.3 Paměť	14
1.1.4 Porty	14
1.2 VÝVOJOVÝ KIT M68EVB908GB60	14
1.2.1 Parametry kitu	15
1.2.2 Napájení	16
2 REÁLNÝ ČAS	17
2.1 MĚŘENÍ ČASU.....	17
2.2 OBVOD REÁLNÉHO ČASU MCP7940M	18
2.2.1 Parametry.....	18
2.2.2 Piny.....	18
2.2.3 Paměť	19
2.3 SBĚRNICE I2C	22
2.3.1 Základní pojmy.....	22
2.3.2 Přenos dat	23
2.3.3 Komunikační protokol.....	23
3 EAGLE.....	25
3.1 PRÁCE V EDITORU.....	25
3.1.1 Editor schémat.....	25
3.1.2 Editor plošného spoje	26
II PRAKTICKÁ ČÁST	27
4 NÁVRH OBVODU	28
4.1 NÁVRH ZAPOJENÍ.....	28
4.1.1 Napájení	28
4.1.2 Krystal	28
4.1.3 Schéma zapojení.....	28
4.1.4 Deska plošného spoje	29
4.2 PROPOJENÍ MODULU S VÝVOJOVÝM KITEM	31
5 PROGRAMOVÉ VYBAVENÍ.....	33
5.1 UŽIVATELSKÉ FUNKCE.....	33
5.2 POMOCNÉ FUNKCE.....	38
5.3 UKÁZKOVÝ PROGRAM	39
ZÁVĚR	41

ZÁVĚR V ANGLIČTINĚ.....	42
SEZNAM POUŽITÉ LITERATURY.....	43
SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	45
SEZNAM OBRÁZKŮ	46
SEZNAM TABULEK.....	47
SEZNAM PŘÍLOH.....	48

ÚVOD

Znalost přesného času se v řadě oborů a aplikací stává naprostou nezbytností. K měření času můžeme využít obvody reálného času. Jsou to integrované obvody, které nám umožňují měřit přesný čas s relativně malou chybou. Jsou používány v téměř všech elektrických zařízeních, které potřebují uchovávat přesný čas.

Cílem této bakalářské práce je navrhnout a vyrobit modul s obvodem reálného času, který bude spolu s vývojovým kitem sloužit jako výuková pomůcka v předmětu Programování mikropočítačů.

Teoretická část se zabývá vývojovým kitem obsahujícím mikropočítač od firmy Freescale Semiconductor. Je uveden popis jeho základních vlastností, registrů, paměti a portů. Dále je popsán zvolený obvod reálného času MCP7940M. Jsou uvedeny jeho parametry, piny a podrobněji popsána paměť. Na závěr je popsán program EAGLE, který slouží k návrhu desek plošných spojů.

Praktická část je rozdělena na dvě části. První část se zabývá návrhem výukového modulu. Je popsáno napájení, krystal, schéma zapojení a připojení modulu k výukovému kitu. Druhá část se zabývá softwarovým vybavením, které se skládá z knihovny funkcí a ukázkového programu. Knihovna poskytuje jednoduché ovládání obvodu a je napsaná v jazyce C a v jazyce symbolických adres. Funkčnost obvodu a knihovny je pak ověřena pomocí ukázkového programu.

I. TEORETICKÁ ČÁST

1 MIKROPOČÍTAČ A VÝVOJOVÝ KIT

1.1 Mikropočítač MC9S08GB60

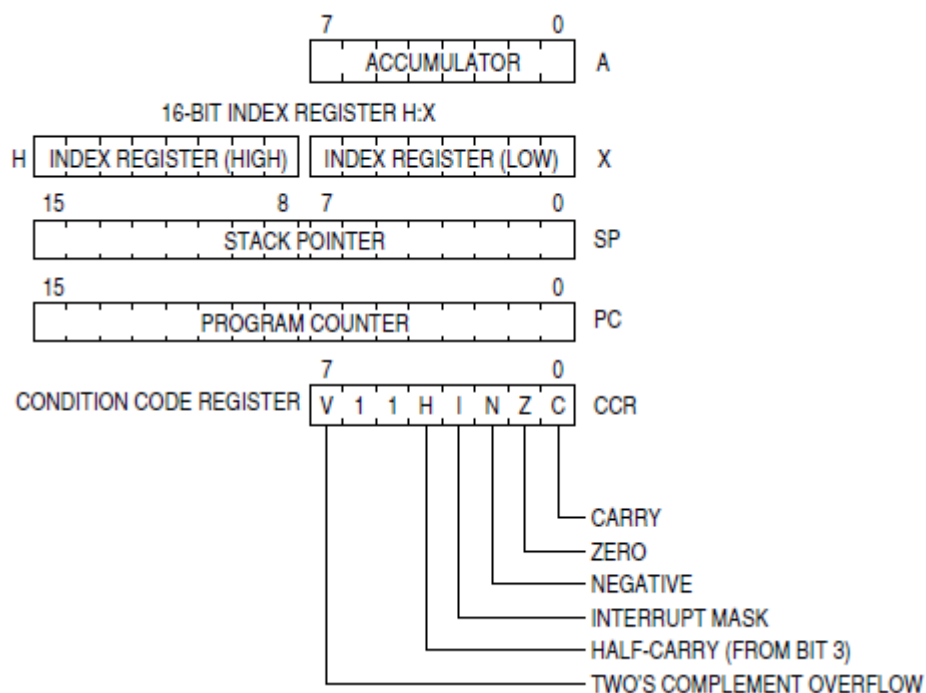
MC9S08GB60 je levný a vysoce výkonný 8 bitový mikropočítač patřící do rodiny HCS08. Mikropočítače patřící do této rodiny jsou dostupné s různými typy a velikostmi paměti, moduly a pouzdry. [1]

1.1.1 Parametry

- 8 bitová centrální procesní jednotka HCS08
- 60KB FLASH paměti
- 4KB RAM paměti
- 8 kanálový, 10 bitový A/D převodník
- 2x asynchronní sériové komunikační rozhraní (SCI)
- 1x synchronní sériové komunikační rozhraní (SPI)
- 1x I2C rozhraní
- 3 kanálový TMP2 časovač
- 5 kanálový TPM1 časovač
- 8 pinový klávesnicový přerušovací modul
- 56 vstupně/výstupních linek na 7 portech (porty A-G)
- 16 pinů pro vysoké proudy
- COP watchdog systém s nastavitelnou časovou prodlevou na 2^{18} nebo 2^{13} cyklů sběrnice

1.1.2 Registry

Procesor obsahuje 5 registrů (viz Obr. 1).



Obr. 1 – Registry procesoru MC9S08GB60 [1]

Akumulátor (A)

Akumulátor je univerzální 8 bitový registr. Jeden operand aritmeticko-logické operace se načte do ALU, druhý je uložen v akumulátoru a výsledek operace je opět uložen do akumulátoru. Reset nemá na obsah akumulátoru vliv. [1]

Indexový registr (H:X)

Je 16 bitový registr složený z dvou oddělených 8 bitových registrů (H a X), které se často používají společně pro uložení 16 bitové adresy, kde H obsahuje vyšší byte adresy a X nižší byte. Všechny instrukce používající indexovou adresaci využívají všech 16 bitů, ale z důvodu kompatibility některé instrukce využívají pouze spodní byte (X).

Mnoho instrukcí využívá registr X jako další univerzální 8bitový registr. [1]

Ukazatel zásobníku (SP)

Tento 16 bitový registr ukazuje na další volné místo v LIFO zásobníku. Zásobník může být umístěn kdekoliv v adresovém prostoru paměti RAM a může využívat veškerou dostupnou paměť RAM. Používá se pro automatické uložení návratové adresy při volání podprogramu, uložení registrů CPU při přerušení a pro uložení lokálních proměnných. [1]

Programový čítač (PC)

Je to 16 bitový registr obsahující adresu další instrukce nebo operandu, který se má zpracovat. Při normálním vykonávání programu je čítač automaticky inkrementován po každé zpracované instrukci. Při skoku, větvení, přerušení nebo návratu se do čítače načte příslušná adresa. [1]

Stavový registr (CCR)

Je to 8 bitový registr obsahující masku přerušení (I) a pět příznaků indikujících výsledek právě dokončené instrukce. [1]

- V – Two's Complement Overflow Flag

Procesor nastaví příznak přetečení, když dojde k přetečení dvojkového doplňku.

- H – Half-Carry Flag

Příznak je nastaven, když v akumulátoru dojde k přenosu z 3 na 4 bit. Využívá se při práci v BCD kódu.

- I – Interrupt Mask Bit

Pokud je nastaven na jedna, jsou všechna maskovatelná přerušení zakázána. Když nastane přerušení, je příznak automaticky nastaven po uložení registrů CPU na zásobník, ale před vykonáním první instrukce obsluhy přerušení.

- N – Negative Flag

Příznak je nastaven, pokud je výsledek aritmetické nebo logické operace záporný.

- Z – Zero Flag

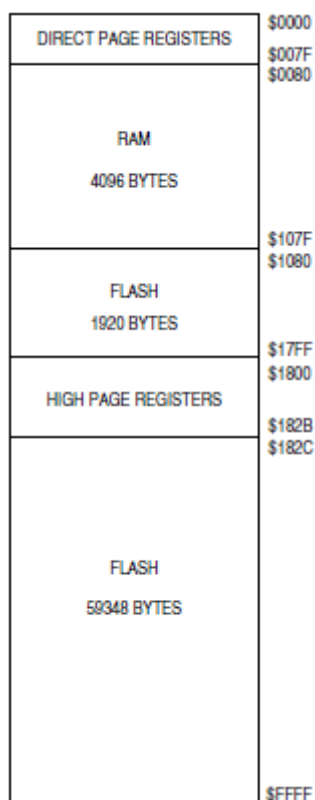
Příznak je nastaven, pokud je výsledek aritmetické nebo logické operace nulový.

- C – Carry/Borrow Flag

Když při nějaké instrukci dojde k přenosu ze 7 bitu akumulátoru, tak je nastaven na jedna.

1.1.3 Paměť

Paměťová mapa mikropočítače MC9S08GB60 se skládá z paměti RAM, FLASH pro program a vstupních, výstupních a kontrolních registrů. [1]



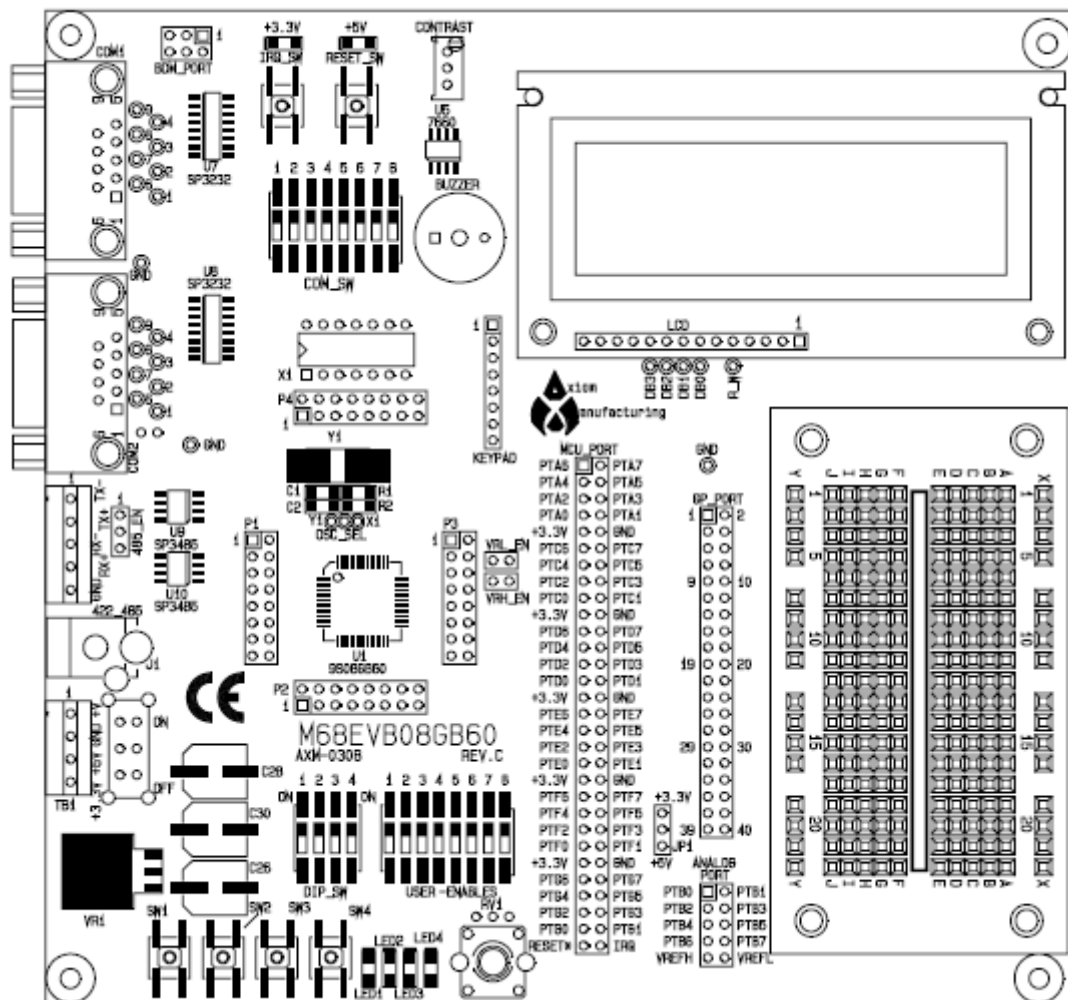
Obr. 2 – Mapa paměti [1]

1.1.4 Porty

Procesor má sedm 8 bitových vstupně výstupních portů (PTA-PTG), celkem tedy 56 univerzálních pinů (jeden z nich je pouze výstupní). Mnoho z těchto pinů je sdíleno s integrovanými periferiemi jako časovač, klávesnice. [1]

1.2 Vývojový kit M68EVB908GB60

M68EVB908GB60 je vývojový kit pro mikropočítač MC9S08GB60. Vývoj aplikací je s vývojovým kitem rychlý a snadný. Vývojový kit obsahuje sériovou komunikaci, ukázkové softwarové nástroje, softwarem pro ladění programu a kontaktním nepájivým polem. [2]



Obr. 3 – Vývojový kit M68EVB908GB60 [2]

1.2.1 Parametry kitu

- 32kHz až 4MHz oscilátor
- Stabilizované napájecí napětí +3.3V a 5V
- COM1 sériový port s DB9 konektorem (SCI0)
- COM2 sériový port s DB9 konektorem nebo RS422/485 konektorem (SCI1)
- ON/OFF vypínač a signalizaci napájení
- Uživatelské prvky
 - 4 LED diody (PTF0-3)

- 4 DIP přepínače (PTB4-7)
- 4 tlačítka (PTA4-7)
- 2x16 znaků LCD modul (PTG3-7, PTE6-7)
- Bzučák (PTD0)
- MCU port s digitálními vstupy/výstupy
- Analogový port s analogovými vstupy a výstupy
- Kontaktní nepájivé pole

1.2.2 Napájení

Napájení je realizováno externím zdrojem připojeným do J1 konektoru nebo ke svorkovnici TB1. Vstupní napětí je stabilizováno na +5V stabilizátorem VR1 a +5V je stabilizováno na +3.3V stabilizátorem VR2. Stabilizátory jsou chráněny proti přepólování diodou D3. Přítomnost napětí +5V a +3.3V je indikována LED diodami. [2]

2 REÁLNÝ ČAS

Čas nám umožňuje řadit události od minulosti přes přítomnost do budoucnosti a měřit dobu jakou událost trvá a dobu mezi jednotlivými událostmi.

2.1 Měření času

Pro měření uplynulého času může být použit každý opakující se jev. Naši předkové využívali k měření času tři přírodní úkazy.

Jeden den (střídání světla a tmy), který je dán rotací Země kolem své osy.

Jeden měsíc (oběh Měsíce kolem Země), kdy se využívají fáze Měsíce. Měsíc v první čtvrti, celý Měsíc a Měsíc v poslední čtvrti viditelný pro každého a odkudkoli a časový interval mezi fázemi je sedm dní (jeden týden) a 9 hodin.

Jeden rok jako oběh Země kolem Slunce. Rytmus života v biosféře je dán tropickým rokem. Je to časový interval mezi dvěma přechody Slunce přes bod jarní rovnodennosti. Tropický rok je o 20 minut kratší než celý oběh Země kolem Slunce zvaný hvězdný rok.

Během historie lidstva se však přístroje k měření času neustále vyvíjely. Dny se pomocí slunečních hodin začaly rozdělovat po 12 hodinách. Následovaly různá vylepšení hodin jako vodní hodiny, přesýpací hodiny, mechanické hodiny (s kyvadlem), elektrické hodiny a nakonec atomové hodiny (s Cesium 133). [6]

Na základě mezinárodních dohod je sekunda jako jednotka času definována následovně:

Sekunda je doba rovnající se 9 192 631 770 periodám záření, které odpovídá přechodu mezi dvěma hladinami velmi jemné struktury základního stavu atomu cesia 133. [9]

Realizace jednotky času je podle této definice provedena pomocí césiových atomových hodin. Jejich nejistota je 10^{-10} s za den.

Fyzikální a technický spolkový úřad (PTB) vlastní čtyři cesiové atomové hodiny CS1 až CS4. Tento přesný čas je dostupný pomocí dlouhovlnného vysílače DCF77 umístěného v blízkosti Frankfurtu a nebo online na NTP serverech. [5]

Obvod reálného času je integrovaný obvod poskytující přesný čas s relativně malou chybou. Používají se všude, kde je potřeba uchovávat přesný čas. Mezi jejich hlavní výhody patří nízká spotřeba, přesnost a osvobození hlavního systému od počítání času.

2.2 Obvod reálného času MCP7940M

MCP7940M je nízkonapěťový obvod reálného času s kalendářem. Obvod je vybaven I2C sběrnici pro komunikaci a pro čítání času využívá externí 32,768kHz krystal.

Obvod obsahuje automatickou korekci pro měsíce s méně než 31 dny a pro přestupný rok. Hodiny pracují v 24 hodinovém nebo 12 hodinovém formátu s indikací AM/PM. [3]

2.2.1 Parametry

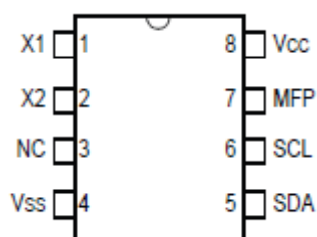
- Hodiny, Minuty, Sekundy, Dny v týdnu, Dny, Měsíce, Roky
- 2 budíky s jedním výstupem
- Kalibrace v rozsahu -127 až +127 ppm
- Možnost nastavení 4 výstupních frekvencí
- 64 bytů SRAM
- Podpora 100kHz nebo 400kHz frekvence I2C sběrnice
- Rozsah teplot -40°C až +85°C

2.2.2 Piny

Jednotlivé piny jsou popsány v tabulce 1 a jejich rozložení v obrázku 3.

Tab. 1 – Popis pinů MCP7940M [3]

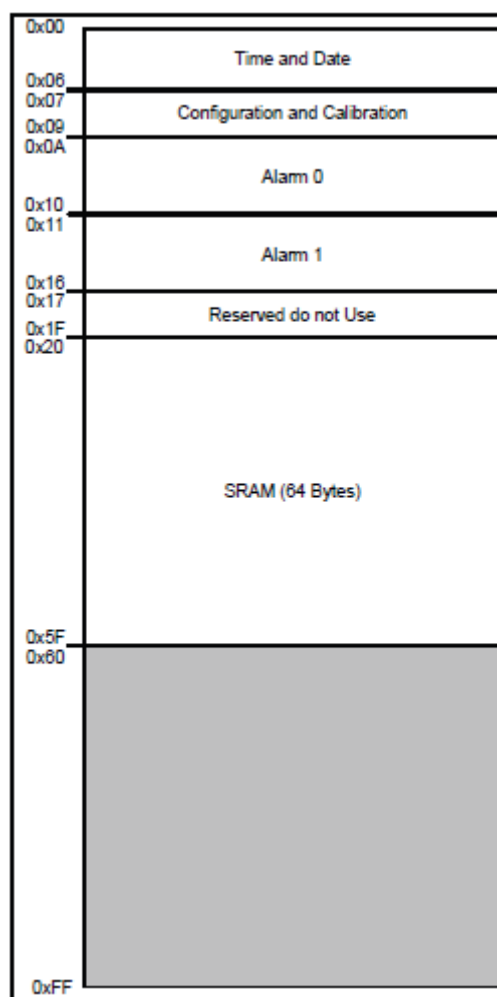
Název pinu	Funkce
Vss	Zem
SDA	Obousměrná sériová komunikace - data
SCL	Sériová komunikace - hodinový signál
X1	Vstup externího oscilátoru
X2	Výstup
NC	Nezapojeno
MFP	Výstup budíku/frekvence
Vcc	Napájení +1.8V až +5.5V



Obr. 4 – Rozložení pinů [3]

2.2.3 Paměť

Rozložení paměti je znázorněno mapou paměti (viz Obr. 4). Nevyužité oblasti nejsou přístupné (šedá oblast) a jejich načtením dostaneme nulu. [3]



Obr. 5 – Mapa paměti [3]

Address	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0	Function	Range	Reset State
00h	ST	10 Seconds			Seconds			Seconds	00-59	00h	
01h		10 Minutes			Minutes			Minutes	00-59	00h	
02h		12/24	10 Hour AM/PM	10 Hour	Hour			Hours	1-12 + AM/PM 00 - 23	00h	
03h		OSCON			Day			Day	1-7	01h	
04h		10 Date			Date			Date	01-31	01h	
05h		LP	10 Month	Month			Month	01-12	01h		
06h		10 Year			Year			Year	00-99	01h	
07h	OUT	SQWE	ALM1	ALM0	EXTOSC	RS2	RS1	RS0	Control Reg.	80h	
08h	CALIBRATION								Calibration	00h	
09h	Reserved – Do Not Use									00h	
0Ah		10 Seconds			Seconds			Seconds	00-59	00h	
0Bh		10 Minutes			Minutes			Minutes	00 - 59	00h	
0Ch		12/24	10 Hour AM/PM	10 Hours	Hour			Hours	1-12 + AM/PM 00-23	00h	
0Dh	ALM0POL	ALM0C2	ALM0C1	ALM0C0	ALM0IF	Day			Day	1-7	01h
0Eh		10 Date			Date			Date	01-31	01h	
0Fh		10 Month			Month			Month	01-12	01h	
10h		Reserved – Do not use								Reserved	01h
11h		10 Seconds			Seconds			Seconds	00-59	00h	
12h		10 Minutes			Minutes			Minutes	00-59	00h	
13h		12/24	10 Hour AM/PM	10 Hours	Hour			Hours	1-12 + AM/PM 00-23	00h	
14h	ALM1POL	ALM1C2	ALM1C1	ALM1C0	ALM1IF	Day			Day	1-7	01h
15h		10 Date			Date			Date	01-31	01h	
16h		10 Month			Month			Month	01-12	01h	
17h-1Fh		Reserved – Do not use								Reserved	00h

Obr. 6 – Detail datových registrů [3]

Všechny časové hodnoty jsou uloženy v BCD tvaru. Následující popis vychází ze zdroje [3]

0x00h – Obsahuje sekundy v rozsahu 00-59. Nastavením 7 bitu na 1 se spustí oscilátor a 0 se zastavuje.

0x01h – Obsahuje minuty v rozsahu 00-59.

0x02h – Bity 3:0 obsahují jednotky hodin a bity 5:4 obsahují buď desítky hodin pro 24 hodinový formát nebo indikátor AM/PM (0 = AM) v bitu 5 a desítky hodin pro 12 hodinový formát. Bit 6 rozhoduje o hodinovém formátu. Nastavením 0 pro 24 hodinový formát a 1 pro 12 hodinový formát.

0x03h – Obsahuje den v týdnu v rozsahu 1-7. Bity 3:4 nejsou implementovány. Bit 5 je nastavován hardwarem. Pokud je nastaven na 1, oscilátor běží, 0 neběží. Neukazuje zda běží na správné frekvenci.

0x04h – Obsahuje datum v rozsahu 01-31.

0x05h – Obsahuje měsíce v rozsahu 01-12. Bit 5 určuje přestupný rok a je pouze pro čtení.

0x06h – Obsahuje roky v rozsahu 00-99.

0x07h – Kontrolní registr

- Bit 7 je výstupní (OUT) bit. Nastavuje logickou úroveň na MFP pinu, když není využíván jako výstup obdélníkového signálu.
- Bit 6 nastavením povolí dělič výstupního signálu.
- Bity 5:4 určují který budík je zapnutý.
 - 00 – Oba budíky jsou vypnuté
 - 01 – Budík 1 je zapnutý
 - 10 – Budík 2 je zapnutý
 - 11 – Oba jsou zapnuté
- Bit 3 EXTOSC
- Bity 2:0 nastavují vnitřní děličku frekvence oscilátoru, která má jít na výstup MFP.
 - 000 – 1Hz
 - 001 – 4,096kHz
 - 010 – 83192kHz
 - 011 – 32,768kHz
 - 1xx – povolí Cal výstupní funkci

0x08h – Kalibrační registr. Obsahuje hodnotu frekvence, kterou každou minutu přičte nebo odečte od čítače RTC. Nejvyšší bit určuje, jaká operace se bude provádět.

0x0Ah-0x0Fh a 0x11h-0x16h – jsou registry pro budík 0 a budík 1 a jsou stejné jako předchozí registry čas/datum s následujícím rozdílem.

0x10h a 0x17h jsou rezervované a neměly by být používány.

0x0Dh/0x14h – Nastavení budíku.

- AMLxPOL – Určuje, na jakou logickou úroveň bude nastaven výstup MFP při spuštění alarmu.
- ALMxIF – Nastaven hardwarem při spuštění alarmu, musí být vynulován softwarem.

- ALMxC2:0 – Určuje, kdy se spustí alarm
 - 000 – Sekundy
 - 001 - Minuty
 - 010 - Hodiny (12/24 hodinový formát)
 - 011 – Den
 - 100 - Datum
 - 101 a 110 – Rezervované
 - 111 – Sekundy, Minuty, Hodiny, Den, Datum, Měsíc
- Bit 6 na 0x0Ch a 0x13h – 12/24 hodinový formát jsou kopie bitu 6 na 0x02h a jsou pouze pro čtení.

2.3 Sběrnice I2C

I2C je obousměrná, dvou vodičová sběrnice propojující master a slave zařízení. Každý přenos zahajuje master a slave je potvrzuje pomocí ACK nebo odpovídá. [4]

2.3.1 Základní pojmy

Následující pojmy jsou použity k popisu sběrnice I2C. [4]

- Vysílač – Zařízení, které posílá data na sběrnici, nebo posílá data jako odezvu na požadavek.
- Přijímač – Zařízení, které přijímá data ze sběrnice.
- Master – Zahajuje přenos, generuje hodinový signál a ukončuje přenos. Může být vysílač i přijímač.
- Slave – Zařízení adresované masterem. Může být vysílač i přijímač.
- SDA – Sériová datová linka.
- SCL – Sériová hodinová linka.

2.3.2 Přenos dat

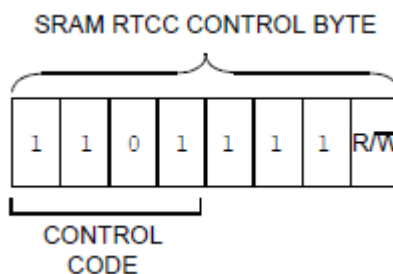
Změna logické úrovně na SDA může proběhnout pouze v případě, že je SCL v logické nule. V případě změny úrovně SDA, když je SCL v logické jedničce se jedná o START nebo STOP podmínku.

- START podmínka – změna logické úrovně na SDA z jedna na nula
- STOP podmínka – změna logické úrovně na SDA z nula na jedna

Přenosový protokol je založen na vyslání 8 bitů a jednoho potvrzovacího bitu (ACK). Přenos začíná od nejvýznamnějšího bitu (MSB). S každým impulsem na SCL je vyslán jeden bit.

Každý přenos dat je zahájen masterem vysláním START podmínky a kontrolního bytu. Kontrolní byte vybírá a aktivuje dané zařízení na sběrnici (viz Obr. 6). První 4 bity představují adresu zařízení a 8 bit je tzv. R/W bit a určuje, zda bude probíhat čtení nebo zápis. [4]

- Pokud je R/W bit nastaven na 1 provádí se čtení.
- Pokud je R/W bit nastaven na 0 provádí se zápis.



Obr. 7 – Kontrolní byte RTC [3]

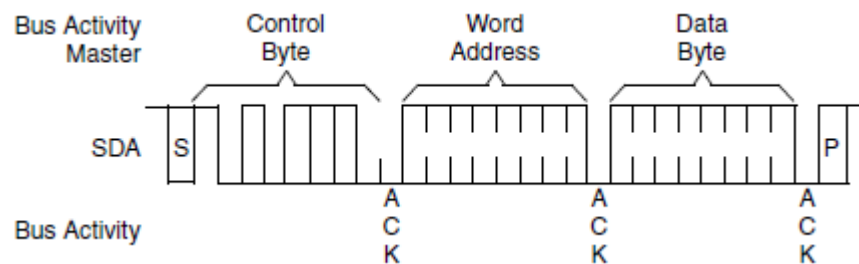
2.3.3 Komunikační protokol

IIC má přesně daný komunikační protokol, který se musí dodržet pro daný typ přenosu.

Zápis dat

Po vyslání START podmínky a kontrolním bytu s nastaveným R/W bitem na 0 je vyslán adresový byte. Ten představuje skutečnou adresu v paměti, na kterou se bude

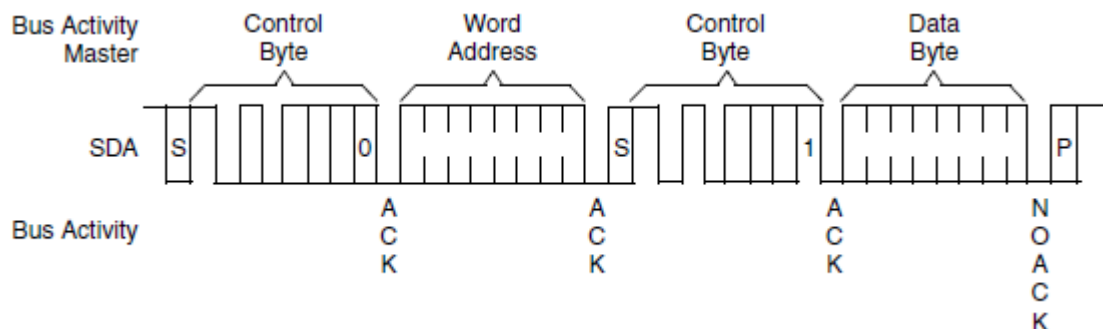
zapisovat. Jako poslední se vyšle datový byte. Každý přenesený byte je potvrzen ACK bitem. [4]



Obr. 8 - Posloupnost bitů pro zápis dat do slave [4]

Čtení dat

Přenos začíná stejně jako při zápisu. Po START podmínce, kontrolním bytu a adresovém bitu master zopakuje START podmínku a v kontrolním bytu nastaví bit R/W na 1. Slave zařízení pošle ACK a datový byte, o který master požádal. Po odeslání datového bytu už slave neposílá ACK, ale vygeneruje STOP podmínku. [4]



Obr. 9 – Posloupnost bitů pro čtení ze slave [4]

3 EAGLE

EAGLE je uživatelsky přívětivý a výkonný nástroj pro návrh desek plošných spojů. Název EAGLE znamená Easily Applicable Graphical Layout Editor.

Návrhový systém EAGLU se skládá z tří hlavních modulů:

- Schematický editor
- Editor plošného spoje
- Autorouter

Po spuštění se na obrazovce objeví Hlavní panel (Control Panel – CP), který nám umožní provést základní nastavení, přecházet mezi jednotlivými editory a složkami. [8]

3.1 Práce v editoru

Existují dva způsoby vytvoření schématu. První přes založení nového projektu a druhý vytvořením samostatného souboru přes CP/File/New/Schematic.

Vytvoření nového projektu provedeme v Control panelu následujícím způsobem. File/New/Project a napíšeme název projektu. Všechny vytvořené soubory k danému projektu se budou automaticky ukládat do složky projektu. Zelený bod vedle složky projektu ukazuje, který projekt je aktivní.

Vytvoříme nové schéma přes /File/New/Schematic a následně se nám zobrazí pracovní prostředí schematického editoru. [8]

3.1.1 Editor schémat

Editor schémat nám umožňuje rychlé a efektivní návrh schémat. V editoru je možné nastavit viditelnost spojů, značek, jmen a hodnot součástek. Důležité je nastavení mřížky (GRID), po které se nám pohybuje kurzor a na kterou se ukládají vývody součástek, spoje a další objekty.

Všechny dostupné součástky jsou uloženy v knihovnách, které jsou organizovány podle typu součástek nebo podle výrobců pomocí stromové struktury. Nabídku knihovny je možné otevřít pomocí ikony ADD z levého menu. Po vybrání součástky ze seznamu její

výběr potvrdíme tlačítkem Ok a součástka bude uchycena na kurzor. Pravým tlačítkem myši můžeme součástkou rotovat a levým tlačítkem ji umístíme.

Propojení součástek provedeme pomocí ikony NET. Kreslit spoj začneme kliknutím na vývod první součástky a kreslení ukončíme dvojklikem na vývod druhé součástky.

Pomocí ikony ERC můžeme provést elektrickou kontrolu a odhalit tak chyby. [8]

3.1.2 Editor plošného spoje

Pomocí tlačítka switch to board se přepneme do editoru plošného spoje. Pracovní prostředí a způsob ovládání je podobné schematickému editoru.

Na pracovní ploše už můžeme vidět pouzdra součástek, které obsahuje schéma, jednotlivé vývody jsou propojeny „gumovými spoji“. Pouzdra součástek přesouváme pomocí ikony Move. Po přesunu součástek nahradíme současné gumové spoje ikonou Rastnest za gumové spoje s nejkratší vzdušnou cestou.

Nejčastěji se spoje propojují ručně, ale lze použít i Autorouter, který nám udělá funkční propojení. Jeho nastavení je ale složitější a výsledný návrh je nevzhledný, proto je vhodné ho ručně upravit. [8]

II. PRAKTICKÁ ČÁST

4 NÁVRH OBVODU

Cílem této bakalářské práce je navrhnout a vyrobit výukový modul s obvodem reálného času pro předmět Programování Mikropočítačů. Při návrhu se vycházelo z datasheetu obvodu reálného času. Obvod vyžaduje pro čítání času externí krystal s frekvencí 32,768kHz. Komunikace mezi modulem a vývojovým kitem probíhá pomocí I2C sběrnice.

4.1 Návrh zapojení

4.1.1 Napájení

Aby mohl obvod reálného času uchovávat aktuální čas, musí být neustále připojen k napájecímu napětí. V případě, že je modul připojen k vývojovému kitu a ten je zapnutý, je obvod napájen napětím 3,3V přes MCU_PORT. V opačném případě je napájen z 3V lithiové baterie umístěné na desce plošného spoje.

Deska pošlého spoje také obsahuje svítivou LED diodu, která svítí, pokud je obvod napájen z vývojového kitu.

4.1.2 Krystal

Správná frekvence krystalu je splněna zajištěním požadované zatěžovací kapacity, která je uvedena výrobcem a je dána vzorcem:

$$C_z = \frac{C_1 * C_2}{C_1 + C_2} + C_p \quad (1)$$

C_z je zatěžovací kapacita

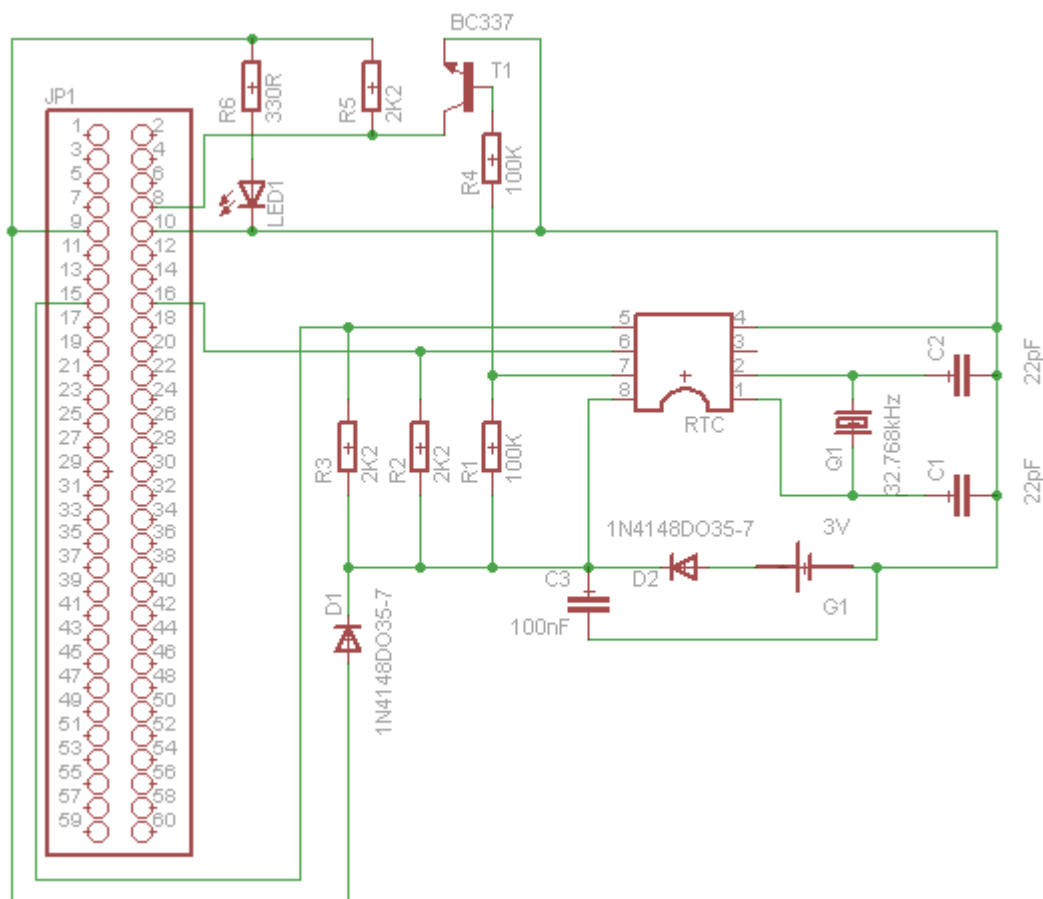
C_p je parazitní kapacita (kapacita pinů a vodičů)

Krystal je vhodné umístit co nejblíže k obvodu a zajistit tak co nejkratší spoje. Malou chybu frekvence pak odstraníme pomocí kalibrace obvodu reálného času.

4.1.3 Schéma zapojení

Schéma zapojení je vytvořeno v programu EAGLE ve verze Light. Jako základ je použito doporučené schéma od výrobce obvodu reálného času, které určuje zapojení

krystalu a jeho kondenzátorů, pull-up rezistorů a oddělovacího kondenzátoru. Toto schéma zapojení je potom rozšířeno o zálohovací baterii, signalizační LED diodu a negaci výstupu MFP obvodu reálného času pomocí bipolárního NPN tranzistoru.



Obr. 10 – Schéma zapojení modulu

4.1.4 Deska plošného spoje

Velikost desky plošného spoje byla určena použitím 60 pinového konektoru, který zajistí správné připojení obvodu k vývojovému kitu. Celkové rozmístění součástek se poté odvíjelo od umístění obvodu reálného času.

4.2 Propojení modulu s vývojovým kitem

Výukový modul je s vývojovým kitem propojen přes MCU_PORT, který poskytuje propojení s I/O piny mikropočítače a napájecí napětí 3,3V.

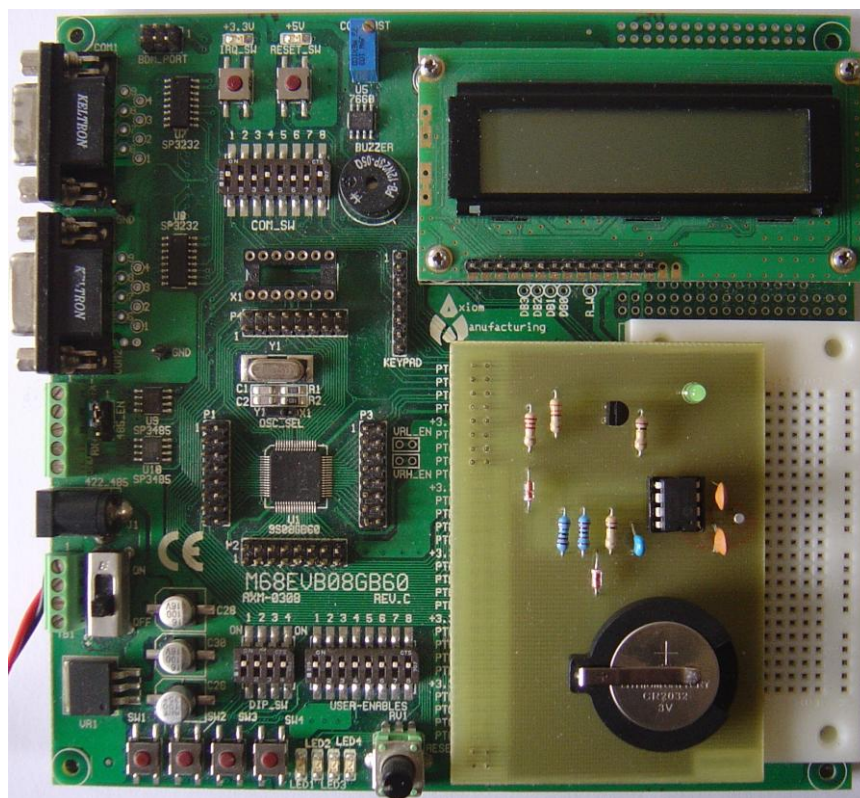
PTA6/KBD6	1	2	PTA7/KBD7
PTA4/KBD4	3	4	PTA5/KBD5
PTA2/KBD2	5	6	PTA3/KBD3
PTA0/KBD0	7	8	PTA1/KBD1
3.3V	9	10	GND
PTC6	11	12	PTC7
PTC4/CLKOUT	13	14	PTC5
PTC2/SDA	15	16	PTC3/SCL
PTC0/TXD2	17	18	PTC1/RXD2
3.3V	19	20	GND
PTD6/TPM2CH3	21	22	PTD7/TPM2CH4
PTD4/TPM2CH1	23	24	PTD5/TPM2CH2
PTD2/TPM1CH2	25	26	PTD3/TPM2CH0
PTD0/TPM1CH0	27	28	PTD1/TPM1CH1
3.3V	29	30	GND
PTE6	31	32	PTE7
PTE4/MOSI	33	34	PTE5/SPSCK
PTE2/SS*	35	36	PTE3/MISO
PTE0/TXD1	37	38	PTE1/RXD1
3.3V	39	40	GND
PTF6	41	42	PTF7
PTF4	43	44	PTF5
PTF2	45	46	PTF3
PTF0	47	48	PTF1
3.3V	49	50	GND
PTG6	51	52	PTG7
PTG4	53	54	PTG5
PTG2/EXTAL	55	56	PTG3
PTG0/BGND/MS	57	58	PTG1/XTAL
RESET*	59	60	IRQ

Obr. 13 – MCU_PORT vývojového kitu

Porty PTC2 a PTC3 jsou použity pro sériovou sběrnici I2C, kde na PTC2 je datová linka a na PTC3 je hodinový signál.

Port PTA1 je použit pro výstup budíku/frekvence z obvodu reálného času. Pin podporuje využití přerušení od klávesnice (KBI), reagující na sestupnou hranu.

Porty číslo 9 a 10 jsou použity pro napájení výukového modulu.



Obr. 14 – Modul připojený k vývojovému kitu

5 PROGRAMOVÉ VYBAVENÍ

Jako programové vybavení je vytvořena knihovna funkcí pro jednoduchou obsluhu všech možností, které obvod reálného času nabízí. Knihovna je naprogramována v jazyce symbolických adres a v C jazyce a obsahuje čtyři soubory nazvané RTC s příponami c, h, asm a inc.

Knihovnu naprogramovanou v jazyce symbolických adres je možné použít místo knihovny v jazyce C. Toho je dosaženo správným načítáním předávaných parametrů funkcí, které poskytuje překladač programu. Pokud má poslední parametr velikost 2 byty, je předán v registru H:X. Pokud má poslední parametr velikost 1 byte, je předán v registru A a pokud má předposlední parametr velikost 1 byte, je předán v registru X, jestliže má velikost 2 byty je předposlední parametr předán v registru H:X. Ostatní parametry jsou zleva vkládány na zásobník.

Návratová hodnota je předávána v registru A pokud má velikost 1 byte nebo v registru H:X pokud má velikost 2 byty. [10]

5.1 Uživatelské funkce

Tyto funkce jsou uživateli přístupné a umožňují mu nastavovat a číst čas. Protože bylo potřeba nastavit nebo načíst poměrně velké množství hodnot, má většina funkcí jako první parametr číslo, podle kterého se vybere co nastavit nebo načíst. Správnost a rozsah nastavovaného času musí zajistit uživatel.

IIC_init

Definice funkce: `void IIC_init(void);`

Je to inicializační funkce, která povolí a nastaví použití I2C sběrnice. Musí se spustit jako první.

setTime

Definice funkce: `void setTime(byte cislo, byte hodnota);`

Funkce má dva vstupní parametry a slouží pro nastavení data a času. První parametr určuje co nastavit.

- 1 – den v týdnu
- 2 – den v měsíci
- 3 – měsíc
- 4 – rok
- 5 – hodiny
- 6 – minuty
- 7 – sekundy
- 8 – AM/PM

Druhý parametr obsahuje hodnotu, která se má nastavit. Funkce nejdřív provede převod hodnoty do BCD kódu a pak ji podle prvního parametru uloží na odpovídající místo v paměti.

Pokud došlo k restartu obvodu reálného času je oscilátor vypnutý a nedochází k čítání času. Pro jeho spuštění je nutné zapsat jakoukoliv hodnotu sekund.

getTime

Definice funkce: byte getTime(byte cislo);

Funkce s návratovou hodnotou typu byte a jedním vstupním parametrem. Slouží pro načtení data a času, kde vstupním parametrem vybíráme co načíst.

- 1 – den v týdnu
- 2 – den v měsíci
- 3 – měsíc
- 4 – rok
- 5 – hodiny
- 6 – minuty
- 7 – sekundy
- 8 – nastavení AM/PM

Funkce vybere podle prvního parametru odpovídající místo v paměti a z něho načte hodnotu, kterou po převodu z BCD kódu vrátí.

setAlarm

Definice funkce: `void setAlarm(byte budík, byte cislo, byte hodnota);`

Funkce bez návratové hodnoty a třemi vstupními parametry. Slouží pro nastavení režimu budíku, data a času. Prvním parametrem volíme číslo budíku, 1 pro první budík a 2 pro druhý. Druhým parametrem vybíráme co nastavit.

- 1 – režim budíku
- 2 – den v týdnu
- 3 – den v měsíci
- 4 – měsíc
- 5 – hodiny
- 6 – minuty
- 7 – sekundy
- 8 – nastavení AM/PM

Třetí parametr obsahuje hodnotu, která se má nastavit. Funkce nejprve převede hodnotu do BCD kódu a pak podle prvního parametru vybere budík a podle druhého parametru odpovídající místo v paměti, na které zapíše hodnotu.

getAlarm

Definice funkce: `byte getAlarm(byte budík, byte cislo);`

Funkce má dva vstupní parametry a návratovou hodnotu typu `byte`. Prvním parametrem zvolíme číslo budíku, 1 pro první budík a 2 pro druhý. Druhým parametrem vybíráme co načíst.

- 1 – režim budíku
- 2 – den v týdnu

- 3 – den v měsíci
- 4 – měsíc
- 5 – hodiny
- 6 – minuty
- 7 – sekundy
- 8 – nastavení AM/PM

Funkce podle prvního parametru vybere budík a podle druhého parametru odpovídající místo v paměti, ze kterého načte hodnotu a po převodu z BCD ji vrátí.

resAlarm

Definice funkce: `void resAlarm(byte budik);`

Funkce má jeden vstupní parametr a slouží pro vypnutí zvonícího budíku. Vstupním parametrem vybíráme budík, 1 pro první budík a 2 pro druhý. Pokud je budík zapnutý, zvoní tak dlouho, dokud je podle zvoleného režimu budíku splněna shoda s aktuálním časem. Tzn., pokud nastavíme režim budíku na hodiny, bude zvonit celou hodinu, bez ohledu na to kolikrát bude funkce `resAlarm` zavolána.

writeByte

Definice funkce: `void writeByte(byte hodnota, byte adresa);`

Funkce má dva vstupní parametry a slouží pro zapsání jednoho bytu do volné paměti obvodu reálného času. První parametr obsahuje hodnotu a druhý adresu. Povolený rozsah adres je od 20h do 5Fh. Při zadání jiné než povolené adresy funkce neprovede žádnou operaci.

readByte

Definice funkce: `byte readbyte(byte adresa);`

Funkce s jedním vstupním parametrem a návratovou hodnotou typu byte. Slouží k načtení jednoho bytu z volné paměti obvodu reálného času. První parametr obsahuje adresu, ze které se má číst. Povolený rozsah adres je od 20h do 5Fh. Při pokusu číst z jiné než povolené adresy vrátí funkce 0.

enableAlarm

Definice funkce: void enableAlarm(byte alarm);

Funkce s jedním vstupním parametrem a slouží k povolení nebo zakázání jednotlivých budíků.

setTimeFormat

Definice funkce: void setTimeFormat(byte format);

Funkce s jedním vstupním parametrem, která slouží pro nastavení formátu hodin. Změní pouze formát hodin, neřeší správnost hodin.

getTimeFormat

Definice funkce: byte getTimeFormat(void);

Funkce bez vstupních parametrů s návratovou hodnotou typu byte. Funkce vrací zvolený formát času, 0 pro 24 hodinový formát a 1 pro 12 hodinový formát.

outFrequency

Definice funkce: void outFrequency(byte povolit, byte frekvence);

Funkce má dva vstupní parametry a slouží k povolení výstupu frekvence na MFP pin. Prvním parametrem se povolí nebo zakáže výstup frekvence. Druhý parametr vybírá výstupní frekvenci.

alarm

Definice funkce: byte alarm(byte cislo);

Funkce s jedním vstupním parametrem a návratovou hodnotou typu byte a slouží pro zjištění, který budík zvoní. Parametrem funkce se volí číslo budíku, 1 pro první budík a 2 pro druhý. Pokud budík zvoní, funkce vrátí 1, jinak vrátí 0.

isLeapYear

Definice funkce: byte isLeapYear(void);

Funkce bez vstupních parametrů s návratovou hodnotou typu byte. Slouží ke zjištění, jestli je aktuální rok přestupný. Pokud je přestupný, funkce vrátí 1, jinak 0.

5.2 Pomocné funkce**toBCD**

Definice funkce: void toBCD(byte hodnota);

Funkce s jedním vstupním parametrem, která převádí číslo z desítkové soustavy do BCD kódu.

fromBCD

Definice funkce: byte fromBCD(byte hodnota);

Funkce s jedním vstupním parametrem a návratovou hodnotou typu byte, která převádí číslo z BCD kódu do desítkové soustavy.

IIC_write_byte

Definice funkce: void IIC_write_byte(byte data, byte adresa);

Funkce má dva vstupní parametry a slouží pro zápis jednoho bytu do obvodu reálného času přes I2C sběrnici. První parametr obsahuje data a druhý adresu, na kterou se mají uložit.

Princip zápisu bytu je popsán v kapitole 2.3.3 Komunikační protokol.

IIC_read_byte

Definice funkce: byte IIC_read_byte(byte adresa);

Funkce má jeden vstupní parametr a návratovou hodnotu typu byte a slouží pro načtení jednoho bytu z obvodu reálného času přes I2C sběrnici.

Princip načtení bytu je popsán v kapitole 2.3.3 Komunikační protokol.

5.3 Ukázkový program

Pro ověření funkčnosti obvodu reálného času a knihovny funkcí je vytvořen ukázkový program, který demonstruje všechny možnosti obvodu. K ovládání programu jsou využity 4 tlačítka a potenciometr. V programu je možné nastavit čas, nastavit a aktivovat budíky, změnit formát času.

V normálním běhu program vypisuje na display aktuální datum a čas a jsou využity 3 tlačítka.

- SW1 – Vstup do menu nastavení
- SW2 – Vypnutí zvonícího budíku 1
- SW3 – Vypnutí zvonícího budíku 2

Pokud je program v režimu menu, je na displeji zobrazena aktuální položka z menu.

- SW1 – Ukončení menu
- SW2 – Listování v menu směrem nahoru
- SW3 – Listování v menu směrem dolů
- SW4 – Potvrzení výběru a přechod na další úroveň menu
- Potenciometr – Výběr hodnoty

Struktura menu:

- Nastavit čas – Obsahuje další úroveň menu pro nastavení Dne v týdnu, Dne v měsíci, Měsíce, Roku, Hodin, Minut, Sekund, AM/PM a Exit pro opuštění úrovně Nastavit čas.
- Nastavit budík 1 – Obsahuje další úroveň menu pro nastavení režimu, data a času pro budík 1. Umožňuje nastavit režim budíku, Den v týdnu, Den v měsíci, Měsíc, Hodiny, Minuty, Sekundy, AM/PM a Exit pro opuštění úrovně.
- Nastavit budík 2 – Má stejnou strukturu jako Nastavit budík 1, ale slouží k nastavení budíku 2.
- Aktivovat budík – Obsahuje další úroveň menu pro zapnutí nebo vypnutí jednotlivých budíků.
- Formát hodin – Obsahuje další úroveň menu, která umožňuje změnit formát hodin.
- Exit – Ukončení menu

ZÁVĚR

Cílem této bakalářské práce bylo navrhnout a vytvořit výukový modul s obvodem reálného času. Při konstrukci modulu byla využita dokumentace vývojového kitu M68EVB908GB60 a dokumentace obvodu reálného času MCP7940M. Pro snadnou obsluhu obvodu byla vytvořena knihovna funkcí napsaná v jazyce C a v jazyce symbolických adres. Funkčnost knihovny a celého modulu je předvedena na ukázkovém programu, který je napsaný v jazyce C.

V teoretické části je uveden základní popis vlastností a parametrů vývojového kitu s mikropočítačem MC9S08GB60 z rodiny HCS08. Dále jsou popsány způsoby měření času a informace o obvodu reálného času MCP7940M. Jako poslední je v teoretické části popsán program EAGLE, který slouží pro návrh desek plošných spojů.

První díl praktické části se zabývá návrhem výukového modulu. Modul je tvořen deskou plošného spoje, která je přes MCU port připojena k vývojovému kitu. Deska obsahuje obvod reálného času, krystal, zálohovací baterii, signalizační LED diodu a tranzistor pro negaci výstupu budíku.

Druhý díl praktické části obsahuje popis jednotlivých funkcí z knihovny. Nejprve jsou popsány uživatelské funkce a dále pomocné funkce, které jsou uživateli nepřístupné. Na závěr je popsán ukázkový program a jeho ovládání.

Při programování modulu se negace výstupu budíku ukázala jako zbytečná, nicméně nezpůsobila žádné problémy a byla tak v zapojení ponechána.

Během této práce jsem si rozšířil znalosti s návrhem elektronických obvodů a jejich výrobou. Při vývoje programu a knihovny funkcí jsem si rozšířil znalosti získané v předmětu Programování mikropočítačů.

ZÁVĚR V ANGLIČTINĚ

The aim of this bachelor thesis was to design and develop an education module with real-time clock. Documentation of development board M68EVB908GB60 and documentation of real time clock MCP7940M was used to design module. For easy operation of the circuit a library of functions was created written in C and assembler. The functionality of the library and the module is shown in the demonstration program, which is written in C.

The theoretical section provides a basic description of the characteristics and parameters of development board with a microcomputer MC9S08GB60 of the HCS08 family. The following section describes the ways to measure time and information about real-time clock MCP7940M. In the last section of the theoretical part the EAGLE program is described, which is used for printed circuit board design.

The first part of the practical part deals with the education module. The module consists of a printed circuit board which is connected to the development board over MCU port. The board has real-time clock, crystal, backup battery, LED diode and transistor for negation of the clock output.

The second part of the practical part contains a description of the functions of the library. Firstly it describes the user functions and auxiliary functions that are not accessible to users. Finally, it describes the demonstration program and its control.

When programming the module the output alarm proved unnecessary, however, it did not cause any problems and was thus left in the diagram.

During this work I have extended knowledge of electronic circuit design and manufacture. When developing program and function libraries I extended my knowledge in programming microcontrollers.

SEZNAM POUŽITÉ LITERATURY

- [1] FREESCALE SEMICONDUCTOR. MC9S08GB/GT Data Sheet, Rev.2.3. [online]. 2004, [cit. 2013-05-4]. Dostupné z: http://www.freescale.com/files/microcontrollers/doc/data_sheet/MC9S08GB60.pdf
- [2] AXIOM MANUFACTURING. M68EVB908GB60 Development Board for Freescale MC9S08GB60, Rev. C [online]. 2006, [cit. 2013-05-4]. Dostupné z: [\[http://cache.freescale.com/files/soft_dev_tools/doc/user_guide/M68EVB908GB60UM.pdf\]](http://cache.freescale.com/files/soft_dev_tools/doc/user_guide/M68EVB908GB60UM.pdf)
- [3] MICROCHIP. MCP7940M Low-Cost I2C Real-Time Clock/Calendar with SRAM [online]. 2011, [cit. 2013-05-4]. Dostupné z: [\[http://ww1.microchip.com/downloads/en/DeviceDoc/22292A.pdf\]](http://ww1.microchip.com/downloads/en/DeviceDoc/22292A.pdf)
- [4] FREESCALE SEMICONDUCTOR. How to Use IIC Module on HCS08 MCUs: AN3291. [online]. 2007, 4/2007 [cit. 2013-05-27]. Dostupné z: [\[http://www.freescale.com/files/microcontrollers/doc/app_note/AN3291.pdf\]](http://www.freescale.com/files/microcontrollers/doc/app_note/AN3291.pdf)
- [5] Vše o času. [online]. 2002 [cit. 2013-05-27]. Dostupné z: <http://home.zcu.cz/~poupa/ptb.html>
- [6] Co je čas [online]. [cit. 2013-06-03]. Dostupné z: <http://www.muzeumhodin.info/download/CoJeCas.pdf>
- [7] FREESCALE SEMICONDUCTOR. HCS08 Family Reference Manual, Rev.1. [online]. 2003. Dostupné z: [\[www.freescale.com\]](http://www.freescale.com)
- [8] JURÁNEK, Antonín a Miroslav HRABOVSKÝ. EAGLE pro začátečníky /: uživatelská a referenční příručka :. 2. vydání. Praha: BEN - technická literatura, 2007, 191 s. ISBN 80-730-0213-2.
- [9] Definice základních jednotek soustavy SI [online]. [cit. 2013-06-08]. Dostupné z: <http://fyzika.jreichl.com/main.article/view/139-definice-zakladnich-jednotek-soustavy-si>
- [10] FREESCALE SEMICONDUCTOR. HCS08 Compiler, Rev. April 15 [online]. 2013, [cit. 2013-06-09]. Dostupné z:

[http://www.freescale.com/files/soft_dev_tools/doc/ref_manual/MCU_HCS08_Compiler.pdf]

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

ACK	Acknowledge
BCD	Binary Coded Decimal
EAGLE	Easily Applicable Graphical Layout Editor
MSB	Most Significant Bit
IIC	Inter-Integrated Circuit
NTP	Network Time Protocol
LED	Light-Emitting Diode
PTB	Physikalisch-Technische Bundesanstalt
RTC	Real-Time Clock
SDA	Serial Data
SCI	Serial Communication Interface
SCL	Serial Clock
SP	Stack pointer
SPI	Serial Peripheral Interface

SEZNAM OBRÁZKŮ

Obr. 1 – Registry procesoru MC9S08GB60 [1]	12
Obr. 2 – Mapa paměti [1].....	14
Obr. 3 – Vývojový kit M68EVB908GB60 [2]	15
Obr. 4 – Rozložení pinů [3]	19
Obr. 5 – Mapa paměti [3].....	19
Obr. 6 – Detail datových registrů [3]	20
Obr. 7 – Kontrolní byte RTC [3]	23
Obr. 8 - Posloupnost bitů pro zápis dat do slave [4].....	24
Obr. 9 – Posloupnost bitů pro čtení ze slave [4].....	24
Obr. 10 – Schéma zapojení modulu.....	29
Obr. 11 – Deska plošného spoje v programu EAGLE.....	30
Obr. 12 – Vyrobená deska plošného spoje.....	30
Obr. 13 – MCU_PORT vývojového kitu.....	31
Obr. 14 – Modul připojený k vývojovému kitu	32

SEZNAM TABULEK

Tab. 1 – Popis pinů MCP7940M [3]	18
--	----

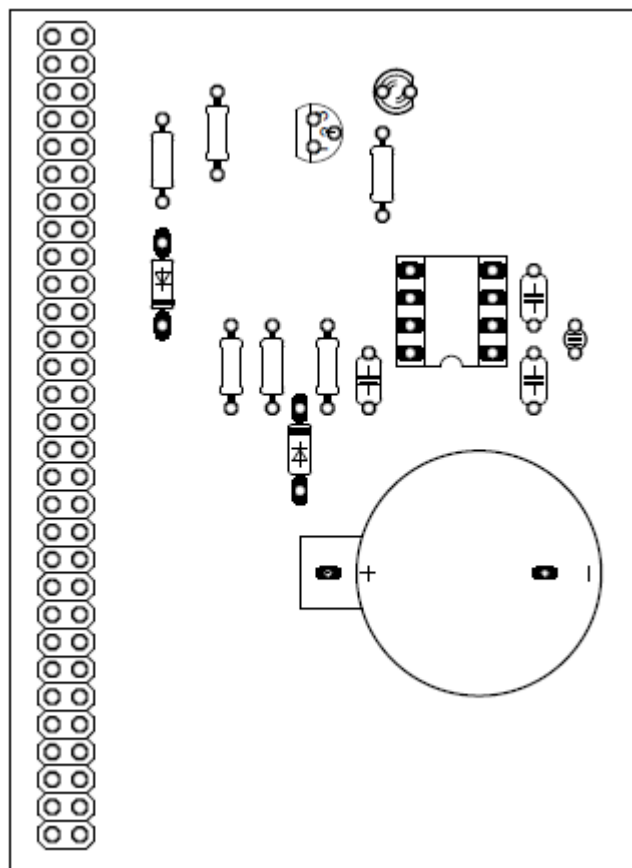
SEZNAM PŘÍLOH

- P I Seznam součástí
- P II Rozmístění součástí
- P III Předloha DPS

PŘÍLOHA P I: SEZNAM SOUČÁSTEK

	ks	typ	hodnota	reference
1	2	kondenzátor	22nF	C1,C2
2	1	kondenzátor	100nF	C3
3	1	krystal	32,768kHz	Q1
4	2	rezistor	100k Ω	R1,R4
5	3	rezistor	2,2k Ω	R2,R3,R5
6	1	rezistor	330 Ω	R6
7	1	LED zelená	-	LED1
8	2	dioda 1N4148	-	D1,D2
9	1	bipolární NPN tranzistor	-	T1
10	1	držák baterie	-	G1
11	1	patice 8 pinů	-	-
12	1	RTC MCP7940M	-	-
13	1	konektorový kolík 2x30	-	JP1

PŘÍLOHA P II: ROZMÍSTĚNÍ SOUČÁSTEK



PŘÍLOHA P III: PŘEDLOHA DPS

