

UNIVERZITA PARDUBICE
Fakulta elektrotechniky a informatiky

Digitální hodinky
Tomáš Borůvka

Bakalářská práce
2014

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: 2013/2014

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Tomáš Borůvka**
Osobní číslo: **I11246**
Studijní program: **B2612 Elektrotechnika a informatika**
Studijní obor: **Komunikační a mikroprocesorová technika**
Název tématu: **Digitální hodinky**
Zadávací katedra: **Katedra elektrotechniky**

Z á s a d y p r o v y p r a c o v á n í :

Cílem práce je vytvoření vlastního demonstračního přípravku "digitálních hodinek" s bateriovým napájením a zobrazením času na displeji. Digitální hodinky budou řízeny procesorem a budou vybaveny obvodem reálného času a programovacím rozhraním pro pozdější změnu SW. Při návrhu zařízení se snažte minimalizovat rozměry a spotřebu zařízení.

V závěru práce proveďte ověření přesnosti měření času v různých teplotních podmínkách.

Zásady pro vypracování:

- Analýza problematiky návrhu časových obvodů a digitálních hodin.
- Návrh zařízení a DPS.
- Konstrukce řídicí části a optimalizace parametrů.
- Konstrukce a oživení přístroje, návrh řídicího SW.

Rozsah grafických prací:

Rozsah pracovní zprávy:

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

Seznam odborné literatury:

[1] Matoušek D., **Práce s mikrokontroléry Atmel AVR**, BEN technická literatura, Praha 2006, ISBN: 80-7300-209-4

[2] Záhlava V., **Návrh a konstrukce desek plošných spojů**, BEN technická literatura, Praha 2011, ISBN: 978-80-7300-2

Vedoucí bakalářské práce:

Ing. Martin Dobrovolný, Ph.D.

Katedra elektrotechniky

Datum zadání bakalářské práce: **20. prosince 2013**

Termín odevzdání bakalářské práce: **9. května 2014**



A handwritten signature in blue ink, belonging to prof. Ing. Simeon Karamazov, Dr.

prof. Ing. Simeon Karamazov, Dr.
děkan

L.S.

A handwritten signature in blue ink, belonging to Ing. Zdeněk Němec, Ph.D.

Ing. Zdeněk Němec, Ph.D.
vedoucí katedry

V Pardubicích dne 31. března 2014

Prohlašuji:

Tuto práci jsem vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 9. 5. 2014

Tomáš Borůvka

Poděkování

Rád bych poděkoval vedoucímu práce panu Ing. Martinu Dobrovolnému, Ph.D. za věnovaný čas a připomínky při tvorbě práce.

Také bych chtěl poděkovat panu Ing. Pavlu Rozsivalovi za zprostředkování výroby DPS. Také si vážím jeho pomoci při oživování zařízení.

Poděkovat chci také rodině za podporu a poskytnutí možnosti rozvíjet své vzdělání na vysoké škole.

Anotace

Práce se věnuje návrhu a výrobě demonstračního přípravku digitálních hodinek. Popisuje použité komponenty, které byly vybrány s ohledem na minimalizaci rozměrů a spotřeby zařízení. Dále se věnuje návrhu DPS, osazení a tvorbě řídicího softwaru.

Klíčová slova

Digitální hodinky, obvod reálného času, řadič SSD1332, ATmega328P

Title

Digital watch

Annotation

Bachelor's thesis is focusing on the design and development of a demonstration device – digital wrist watch. In the thesis is described the selection of suitable components according to the size and power consumption optimization. The second part of the thesis deals with process of PCB creation, surface parts mounting and design of control software.

Keywords

Digital wrist watch, real time clock, SSD1332 controller, ATmega328P

Obsah

Seznam zkratk.....	9
Seznam obrázků.....	10
Seznam tabulek.....	10
Úvod.....	11
1 ÚVOD DO PROBLEMATIKY	12
1.1 Požadavky na navrhované zařízení.....	13
1.1.1 Časový obvod.....	14
2 MOŽNOSTI ŘEŠENÍ.....	17
2.1 Hodinky EZ430 – Chronos	17
2.2 Chytré hodinky Pebble.....	18
2.3 Vlastní výroba hodinek.....	20
2.4 Rešerše možností řešení.....	22
3 ZVOLENÉ ŘEŠENÍ.....	23
3.1 Řídící obvod.....	23
3.2 Obvod reálného času	24
3.3 Napájecí obvod	25
3.4 Displej.....	26
3.5 Uživatelské rozhraní.....	27
3.6 Shrnutí řešení	28
4 NÁVRH DESKY A OBVODŮ	29
4.1 Celkové blokové schéma zapojení	29
4.2 Zapojení Atmega328P	30
4.3 Zapojení displeje.....	31
4.3.1 DC-DC konvertor	32
4.4 Zapojení napájecího obvodu.....	33
4.5 Zapojení RTC	35
4.6 Zapojení ostatních obvodů	36
5 VÝROBA.....	37
6 SOFTWARE	38
6.1 Řadič SSD1332 a GDDRAM	42
7 OVĚŘENÍ ZAŘÍZENÍ.....	44

8 ZÁVĚR	45
Literatura	46
Příloha A – Celkové schéma vytvořené v prostředí EAGLE.....	47
Příloha B – Návrh DPS v prostředí EAGLE.....	48
Příloha C – Seznam použitých součástek	49
Příloha D – CD.....	50

Seznam zkratek

DPS	Deska Plošných Spojů
IO	Integrovaný Obvod
MCU	Micro Controller Unit
RTC	Real Time Clock
PC	Personal Computer
LCD	Liquid Crystal Display
OLED	Organic Light-Emitting Diode
SMD	Surface Mounted Device
SDK	Software Development Kit
ISP	In System Programming
SPI	Serial Peripheral Interface
I2C	Inter-Integrated Circuit
TWI	Two Wire Interface
USB	Universal Serial Bus
UART	Universal Asynchronous Receiver and Transmitter
BCD	Binary Coded Decimal
LDO	Low Drop Output
SW	Software
RAM	Random Access Memory
GDDRAM	Graphic Display Data RAM
LED	Light-Emitting Diode

Seznam obrázků

Obrázek 1 - Digitální hodinky	12
Obrázek 2 - Blokové schéma zapojení digitálních hodinek	13
Obrázek 3 - Blokové schéma elementárního obvodu pro zobrazení času	14
Obrázek 4 - Referenční signál čítače/časovače	15
Obrázek 5 - Inkrementace registrů RTC obvodu	15
Obrázek 6 - Vnitřní blokové schéma RTC obvodu	16
Obrázek 7 - EZ430 – Chronos	17
Obrázek 8 - Hodinky Pebble	19
Obrázek 9 - Hodinky od Zak Kemble	20
Obrázek 10 - Hardware amatérských digitálních hodinek	21
Obrázek 11 - Vnitřní uspořádání obvodu PCF8563T	24
Obrázek 12 - Blokové schéma napájecího obvodu	25
Obrázek 13 - Zvolený OLED displej	26
Obrázek 14 - Blokové schéma zapojení vyráběných hodinek	29
Obrázek 15 - Zapojení ATmega328P-AU	30
Obrázek 16 - Rozložení OLED displeje	31
Obrázek 17 - Zapojení DC-DC konvertoru	32
Obrázek 18 - Zapojení napájecího obvodu	33
Obrázek 19 - Nabíjecí schéma li-pol baterie	34
Obrázek 20 - Zapojení obvodu PCF8563T	35
Obrázek 21 - Zapojení programátoru	36
Obrázek 22 - Zapojení tlačítka	36
Obrázek 23 - Zapojení obvodu zvuku	36
Obrázek 24 - Vyrobená a osazená DPS	37
Obrázek 25 - Vývojový diagram softwaru	39
Obrázek 26 - Spouštěcí a vypínací procedura displeje	42
Obrázek 27 - Rozložení GDDRAM řadiče SSD1332	43
Obrázek 28 - Bitová posloupnost pro barevnost 65k	43
Obrázek 29 - Bitová posloupnost pro barevnost 256	43

Seznam tabulek

Tabulka 1 - Vybrané vlastnosti kontroléru ATmega328P	23
Tabulka 2 - Vlastnosti OLED displeje	26
Tabulka 3 - Přehled použitých obvodů	28
Tabulka 4 - Funkce pinů displeje	31
Tabulka 5 - Popis pinů dohlížecího obvodu	33

Úvod

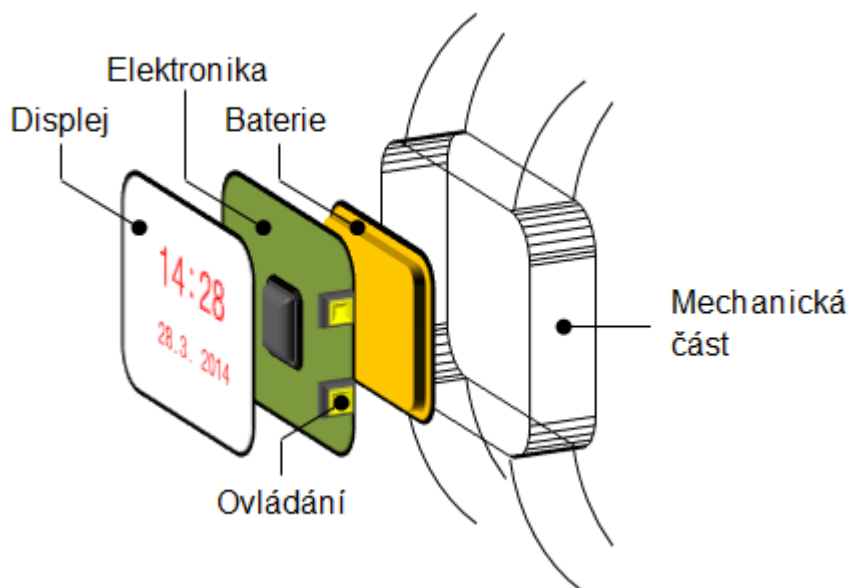
Téma této práce jsem si zvolil především kvůli specifickým požadavkům, které jsou kladeny na konstrukci a výrobu zařízení. Digitální hodinky jsou jedinečné svou mobilitou a velikostí, která klade určité nároky na řídicí elektroniku. Jednotlivé obvody musí být sestaveny tak, aby splňovali požadavky uživatele a zároveň byly šetrné k baterii. Trend dnešní doby vybízí k technologické evoluci hodinek a ty už tak neslouží jen jako poskytovatel informace o čase, ale jako multifunkční a hlavně interaktivní zařízení, které uživateli usnadňuje každý den.

Cílem práce je na základě rešerše dostupných řešení vyrobit demonstrační přípravek, který bude svou podstatou splňovat základní kritéria konstrukce digitálních hodinek. Úvodní kapitola práce zahrnuje problematiku časových obvodů. Následuje přehled dostupných řešení výroby. Na základě toho jsou vytyčeny konkrétní vlastnosti vyráběného zařízení a s tím spojené požadavky na elektroniku. V praktické části se pak práce věnuje použitým součástkám a jejich zapojení do obvodu a následné výrobě DPS. Poslední kapitoly obsahují vývoj řídicího softwaru hodinek a ověření funkčnosti vyrobeného zařízení.

Výroba takto malého zařízení v amatérských podmínkách vyžaduje co nejoptimálnější minimalizaci obvodů. Proto je důležité zvolit vhodné komponenty, které svými vlastnostmi umožní docílení konstrukčních a energetických požadavků zařízení.

1 ÚVOD DO PROBLEMATIKY

Následující kapitoly popisují postup výroby demonstračního přípravku, který bude plnit funkci digitálních hodinek. Výroba tohoto zařízení klade důraz na výběr takových komponentů, které budou plnit předpoklady celkové funkčnosti zařízení. Těmi jsou spotřeba a rozměr. Zařízení by se tedy mělo skládat z malých obvodů, které mají nízkou spotřebu. Na obrázku níže (Obrázek 1) jsou znázorněny digitální hodinky a jejich uživatelské a vnitřní rozložení.

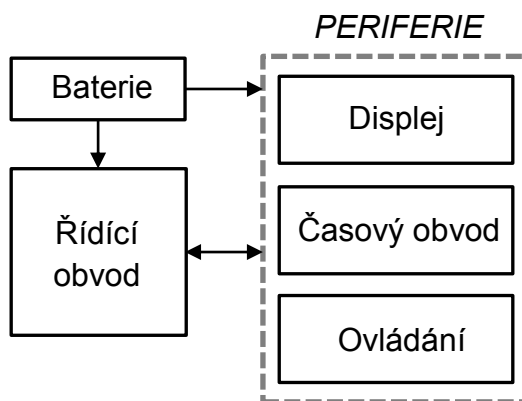


Obrázek 1 - Digitální hodinky

Při pohledu na obrázek z pravé strany vidíme nejdříve mechanickou část, která není součástí práce, ale je nedílnou součástí digitálních hodinek. Práce se věnuje dalším částem obrázku a následující z nich je baterie. Ta je připojena na desku plošných spojů, která obsahuje veškerou elektroniku potřebnou pro funkci zařízení. Nachází se na ní ovládací prvky, které zprostředkovávají interakce mezi uživatelem a hodinkami. Ty jsou zobrazovány na displeji, který je zpětnou vazbou pro obsluhu a předává jí užitečné informace, například čas. Ovládání a displej zařízení jsou důležitými komponenty. Jsou to pro uživatele jediné části zařízení, se kterými přijde do kontaktu. Je důležité, aby manipulace s nimi a jejich přehlednost byla snadná a srozumitelná.

1.1 Požadavky na navrhované zařízení

Elektronika se skládá z řídicího obvodu, baterie a periférií. Hlavním požadavkem na elektroniku zařízení je především nízká spotřeba, celková minimalizace a možnost řízení základních funkcí hodinek. Na obrázku (Obrázek 2) je příklad blokového schéma zapojení elektroniky digitálních hodinek.



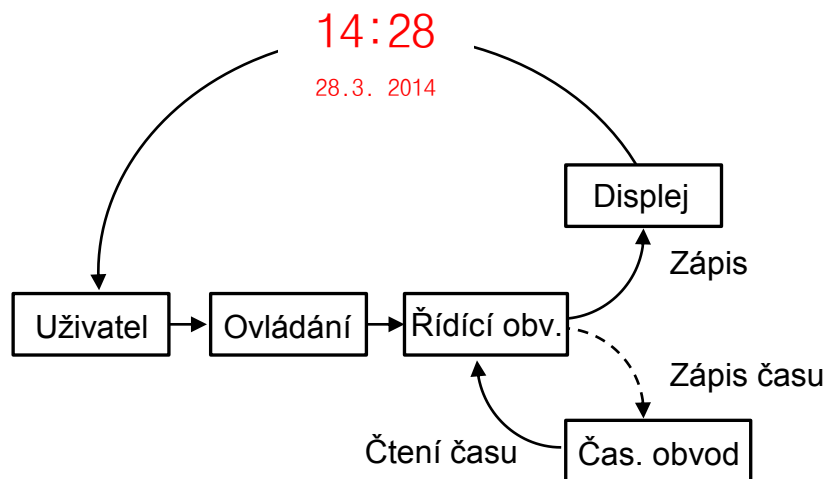
Obrázek 2 - Blokové schéma zapojení digitálních hodinek

Baterie – Zajišťuje energii pro všechny části obvodu. Baterie musí mít dostatečnou kapacitu a také co nejmenší hmotnost. Obecně totiž tvoří velké procento z celkové váhy zařízení.

Řídící obvod – Definiuje chování systému. Obsahuje software, který postupuje podle jistého algoritmu s určitým taktem – frekvencí. Přijímá informace od připojených zařízení (tlačítko). S informacemi se pak dá pracovat na programové úrovni např. v jazyce C a výsledek předat jiným obvodům (displej). Kritéria pro tento obvod jsou dostatečně velká paměť pro vybrané zařízení, možnosti připojení periférií a také rychlost s jakou provádí instrukce.

Periferie – Jsou obvody či zařízení sloužící jako obsluha či výstup zařízení. Dělí se tedy na vstupní a výstupní. Sami o sobě neplní žádnou funkci a je potřeba dalšího zařízení, ke kterému se připojí. Příkladem vstupní periferie může být klávesnice a zařízením počítač. K němu je pak připojena výstupní periferie - monitor. U periférií je důležité zajistit kompatibilitu s řídicím obvodem. U mobilních zařízení je důležitým faktorem nízká spotřeba.

Blokové schéma (Obrázek 3) znázorňuje elementární funkční obvod pro zobrazení času, který také splňuje náležitosti pro zařízení jako digitální hodinky. Analogie obrázku je popsána níže.



Obrázek 3 - Blokové schéma elementárního obvodu pro zobrazení času

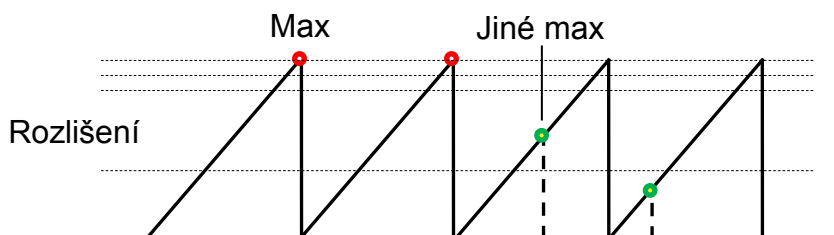
Informace o čase je systematicky předávána uživateli pomocí displeje. Ten může ovládacími prvky přistupovat k řídicímu obvodu. Řízení je naprogramované pro plnění určitých funkcí a uživatel je při ovládání zařízení omezen softwarem. Program běží v nekonečné smyčce a uživatel může neustále zadávat podněty řídicímu obvodu a ten je musí vždy vyhodnotit. Tak je zajištěna interakce a jistá kontinuita mezi řídicím obvodem a uživatelem. Obsluha by měla mít přístup k nastavení času s přehlednými a srozumitelnými pokyny na displeji.

Uživatelské rozhraní je důležité pro správnou činnost systému a vytváří určitý komfort při používání. Při výrobě jakéhokoliv zařízení je vhodné zhotovit manuál uživatelského rozhraní a poučit budoucího uživatele, jak správně zacházet se zařízením a co od něj může a nemůže očekávat.

1.1.1 Časový obvod

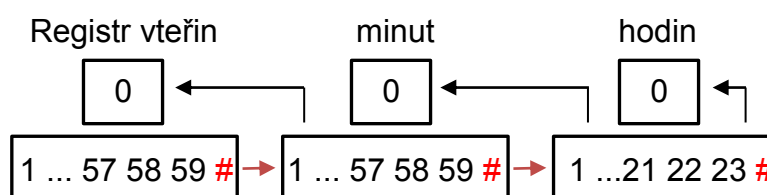
Časový obvod je takový systém, který se stará o interpretování informace o čase či stavu počítačů. Čítáním známých podnětů lze vyhodnotit momentální stav čítače a porovnat jej s nadefinovanou podmínkou, která vyvolá příslušnou reakci. Přitom by měla být zajištěna kontinuita systému, aby nedošlo k nedefinovaným operacím.

Většina řídicích obvodů nabízí funkci čítače/časovače. Tato integrovaná funkce pracuje na principu čítání nastavených maxim dosažených v určitém referenčním signálu, kterým může být například pilový průběh (Obrázek 4). Ten je rozdělen na několik úrovní podle rozlišení čítače (8-bitů = 256 úrovní s maximem 0xFF). Dosažením maxima nebo úrovně nastavené obsluhou je splněna podmínka s příslušnou odezvou. Rychlost čítání závisí na taktu řídicího obvodu, který se dá zpravidla měnit předděličkami. Tento způsob čítání je možné správným nastavením použít jako reprezentaci reálného času.



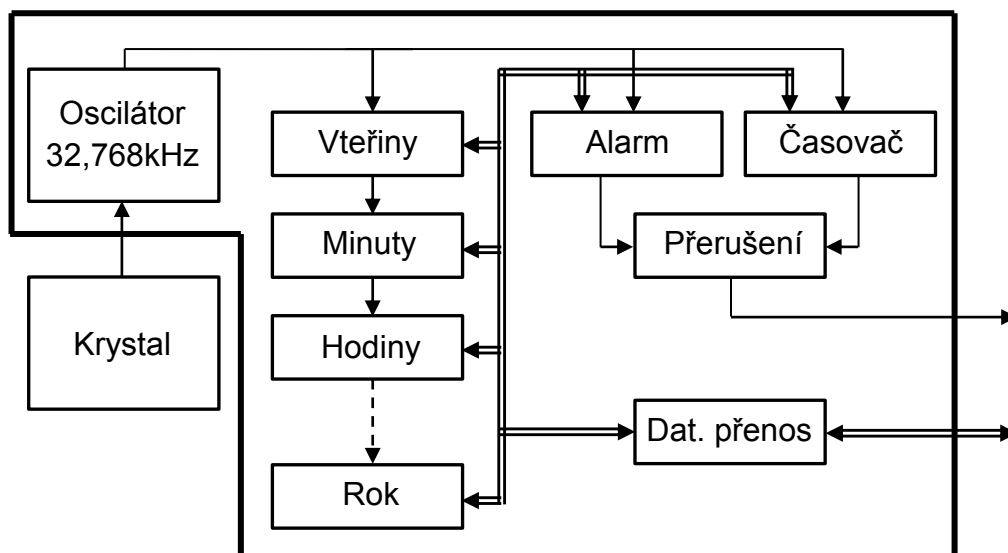
Obrázek 4 - Referenční signál čítače/časovače

Existují i speciální obvody pro časové systémy. Jedním z nich je takzvaný obvod reálného času neboli *RTC*. Stará se o informaci reálného času tím, že poskytuje jinému systému svá data. Těmi jsou například informace o stavu vteřin, minut a hodin, ale i dní, měsíců a roku. Princip je takový, že jsou do příslušných pamětí nahrána výchozí data obsahující určitý stav času. Poté se data v paměti, která je ve formě registrů, automaticky inkrementují a při splnění definované podmínky následuje příslušná reakce. Analogie procesu je znázorněna na obrázku (Obrázek 5).



Obrázek 5 - Inkrementace registrů RTC obvodu

Blokové schéma (Obrázek 6) zobrazuje vnitřní uspořádání *RTC* obvodu. Podmínkou pro automatické inkrementování dat je přítomnost referenčního signálu. Ten je generován oscilátorem, který má zpravidla frekvenci 32,768 kHz. Oscilátor vyžaduje zapojení příslušného krystalu, který nejvíce ovlivňuje přesnost obvodu. Některé obvody využívají integrovaného resonátoru. Takové obvody již nevyžadují zapojení téměř žádných externích součástek a jsou velmi přesné.



Obrázek 6 - Vnitřní blokové schéma *RTC* obvodu

Každý registr obsahuje data určující aktuální stav proměnné. Možnost do registrů data zapisovat nebo z nich číst je zajištěna datovým přenosem zpravidla přes *I2C* sběrnici. Obvod také zpravidla obsahuje i jiné registry pro funkce alarmů a časovačů. Jejich výstup je často vyveden speciálním pinem jako externí přerušení.

V případě použití časového obvodu jako zprostředkovatele informace o reálném čase je vhodnější použít dedikovaný *RTC* obvod. Při dlouhodobém používání jsou rozdíly v čase mezi univerzálním časovačem a *RTC* obvodem znatelné. Stav času z čítače může být po několika hodinách nepřesný o několik vteřin. Další výhodou použití *RTC* obvodu je jeho nízká spotřeba a nezávislá provozuschopnost spojená s použitím záložní baterie. Proto se tyto obvody využívají například v počítačích a všude tam, kde je potřeba informace o čase.

2 MOŽNOSTI ŘEŠENÍ

K návrhu je možné přistoupit více způsoby. Je vhodné prozkoumat možná řešení například na internetu a nahlédnout tak do problematiky časových obvodů, konkrétně digitálních hodinek. Můžeme využít vývojových kitů určených pro práci s periferiemi nebo se můžeme setkat přímo s amatérskou výrobou digitálních hodinek a jejím popisem. Při hledání dostupných řešení bylo vybráno několik zařízení, která budou detailněji popsána.

2.1 Hodinky EZ430 – Chronos

Digitální hodinky od firmy Texas Instruments (Obrázek 7) jsou komplexním bezdrátovým zařízením s možností editace firmwaru. Dodávají se v základním nastavení s určitými funkcemi. Ty uživatel může pomocí externího eZ430 USB emulátoru přeprogramovat. Externí přístupový bod pro bezdrátovou komunikaci CC1111 USB je určen pro komunikaci mezi počítačem nebo například s pulsmetrem či krokoměrem. Spolu s hodinkami je dodáván vývojový software a tzv. Control Center, který umožňuje například kalibraci akcelerometru, synchronizaci času s *PC*, ukládání dat a další zajímavé funkce. To vše bezdrátově.



Obrázek 7 - EZ430 – Chronos¹

¹ Zdroj [1]

System je postavený na řídicím obvodu CC430F6137. Tento 16-bitový mikrokontrolér nabízí 32 kB Flash a 4 kB RAM paměti s taktem až 20 MHz. Je vybaven integrovanými systémy jako je bezdrátová komunikace, senzor baterie a teploty, obvod reálného času, akcelerometr a LCD řadič. To vše je spolu s dalšími obvody integrováno v jednom čipu s 64 piny. Jeho spotřeba v aktivním režimu je udávána výrobcem na 160 $\mu\text{A}/\text{MHz}$. Ve Standby Mode je spotřeba udávána 2 μA . Hodinky disponují monochromatickým LCD displejem s 96 segmenty. Ten je řízen přímo přes řídicí obvod CC430. Energii systému dodává lithiová baterie CR2032. Při továrním nastavení a bez používání měřících funkcí hodinek vydrží baterie až 2 roky. [1]

Hodinky tedy nabízí široké využití s cenou okolo 1500 Kč. Jsou obchodně vedeny spíše jako Dev Kit, tedy zařízení určené pro modifikace. Firma Texas Instruments vyvíjí další verze a podporuje ty starší různými aktualizacemi. Také vývojový software dostává nová rozšíření a aktualizace.

2.2 Chytré hodinky Pebble

Hodinky vyvinuté společností Pebble Technology v roce 2013 jsou naopak od předchozího modelu firmy TI určené pro běžné nošení a sledují poslední trendy ve světě mobilních technologií. Nejedná se již o Dev Kit, ale ryze mainstreamové zařízení. O chod systému se stará closed-source operační systém Pebble OS. Hodinky jsou do jisté míry programovatelné díky podpoře SDK². Uživatel například může vytvořit aplikaci v jazyce C a používat jí ve svých hodinkách. Nemůže však přímo přistupovat k vlastnímu řídicímu obvodu hodinek. Uživatelům jsou zpřístupněny obvody pro řízení vibrací nebo akcelerometru.

Hlavní vlastností hodinek Pebble je konektivita pomocí Bluetooth 4.0 a s tím spojená kompatibilita se zařízeními používající operační systém Android a iOS. Hodinky komunikují s mobilním telefonem a dokáží upozornit na příchozí hovor či email. Už dnes existuje přes tisíc aplikací podporující různá zařízení s různým účelem. [2]

² Software Development Kit jsou nástroje pro vytváření aplikací pro určitý systém.



Obrázek 8 - Hodinky Pebble³

Hodinky jsou řízeny mikrokontrolérem STM32F205RE z rodiny ARM Cortex-M3. Tato výkonná 32-bitová jednotka s taktem 120 MHz nabízí až 1 MB Flash a 128 kB SRAM paměti. Má až 140 vstupně výstupních pinů a podporu *LCD*, *USB* a jiných komunikací, obvod reálného času a mnoho dalšího. Výkonný řídicí obvod jako tento je nutný pro rychlou komunikaci a řešení složitých algoritmů. [4]

Pro dlouhou výdrž je použit 1,26“ (144x168 pixelů) grafický transflektivní⁴ displej. Malá lithium-polymerová baterie s kapacitou 130 mAh postačuje potřebám zařízení a je nabíjena pomocí *USB* kabelu. Na jedno nabití vydrží až 7 dní a to za předpokladu, že zařízení obdrží 20 - 30 notifikací za den a čas se aktualizuje jednou za minutu. Ovládání je vyřešeno tlačítky na bocích zařízení. Mechanická konstrukce dokonce splňuje standard vodotěsnosti do 5 atm. [2]

Hodinky Pebble jsou samostatné zařízení, které po spárování s mobilním telefonem dostává nový rozměr využití. Tento typ hodinek se často označuje jako smartwatch, česky chytré hodinky. Na trhu je více podobných produktů za různé ceny. Za hodinky Pebble zájemce zaplatí kolem 3500 Kč.

³ Zdroj [3]

⁴ Transflektivní displej spojuje technologie transmisivního a reflektivního displeje a tím nepotřebuje podsvícení.

2.3 Vlastní výroba hodinek

Při výrobě digitálních hodinek v amatérských podmínkách zpravidla nejsou použity žádné speciální technologie výroby a jsou sestaveny z dostupných součástek. Právě dostupnost součástek a jejich cena je největším rozdílem oproti plošné výrobě jako u předchozích modelů. Výroba vyžaduje nároky na minimalizaci obvodů, která se obtížně realizuje.

Digitální hodinky od Zak Kemble z Velké Británie jsou vyrobeny v amatérských domácích podmínkách. Zařízení i přesto vypadá po vizuální stránce kompaktně a plní funkce digitálních hodinek. Kvůli nedokonalé integraci a minimalizaci postrádá zařízení oproti ostatním modelům akcelerometr, bezdrátovou konektivitu a další funkce. Nahrání a změna programu je vyřešena připojením programátoru do určených vývodů rozmístěných po *DPS*. Propracovaný software nabízí i hraní jednoduchých her.

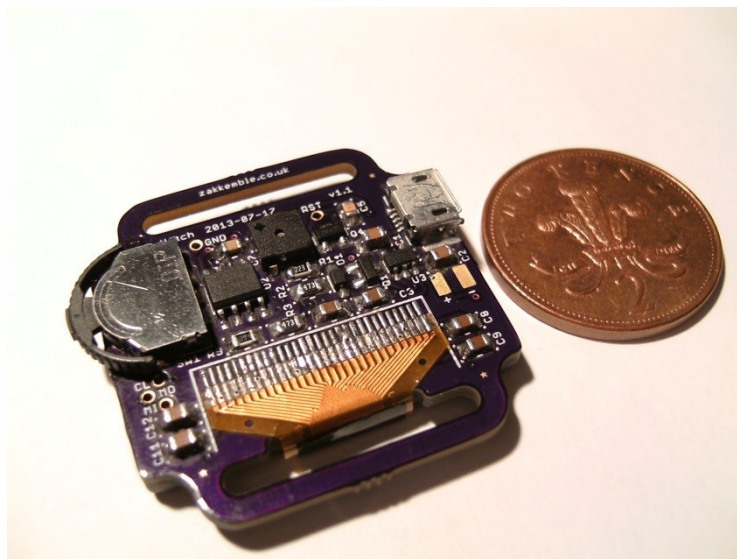


Obrázek 9 - Hodinky od Zak Kemble⁵

Na snímku výše jsou zmíněné hodinky. Při prvním pohledu zaujme 1,3“ velký monochromatický *OLED* displej s rozlišením 128x64 pixelů. Poskytuje dostatečnou aktivní plochu pro zobrazení hodin a dalších indikací. Do hodinek je připojen kabel s microUSB konektorem, který slouží k nabíjení lithium-polymerové baterie s kapacitou 150 mAh.

⁵ Zdroj [5]

Řídicím obvodem je v tomto případě dostupný 8-bitový mikrokontrolér Atmega328P. Nenabízí tolik funkcí jako čipy předchozích modelů. Nevýhodou je absence integrovaného časového obvodu. Ten je doplněn samostatným integrovaným obvodem DS3231M. Paměť 32 kB Flash a 2 kB RAM dávají dostatečný prostor pro základní využití *OLED* displeje. Frekvence až 20MHz je vhodná i pro náročnější aplikace. Odběr proudu samotného mikrokontroléru je přibližně 1,5 mA při taktu 8 MHz a napájecím napětí 2,5 V. V režimu spánku 100 nA.



Obrázek 10 - Hardware amatérských digitálních hodinek⁶

Hodinky mají celkový odběr v aktivním režimu okolo 10 mA a v režimu spánku 6,1 μ A. Teoretická výdrž zařízení na jedno nabití je 1 rok a 4 měsíce. Vychází se z podmínek, že hodinky budou v aktivním režimu 12x za den po dobu pěti vteřin. [5]

Tyto amatérsky vyrobené hodinky jsou komplexním zařízením, které splňuje standardní používání hodinek. Umožňuje uživateli plnou kontrolu nad hardwarem nahráním vlastního softwaru. Cena výroby hodinek může být různá. Záleží především na dostupnosti klíčových komponentů jako je displej a baterie. Částka se tak může blížit až k 1000 Kč.

⁶ Zdroj [5]

2.4 Rešerše možností řešení

Vzhledem k rozmanité nabídce elektroniky na trhu je mnoho možností jak k výrobě přistoupit. Na hodinkách EZ430 – Chronos je zajímavým způsobem vyřešeno bezdrátové spojení zařízení. To využívá vlastností mikrokontroléru CC430F6137, který se perfektně hodí pro tyto aplikace. Navíc podporuje segmentové *LCD* a obsahuje integrovaný *RTC* obvod. I přes vyšší náročnost osazení je tento kontrolér vhodný k výrobě digitálních hodinek. Jeho cena se pohybuje kolem 150 Kč. Naopak displej modelu EZ430 je sice přehledný a nízkoenergetický, ale postrádá volnost při zobrazování informace kvůli segmentovému uspořádání. Stejně jako ostatní segmentové displeje je pro složitější efekty nevhodný.

Chytré hodinky Pebble také nabízí bezdrátovou konektivitu, tentokrát však pomocí Bluetooth technologie. Tento způsob konektivity se jeví jako ideální vzhledem k možnostem propojení s jinými zařízeními. Jeho implementace s ostatními obvody do tak malého zařízení vyžaduje vysoké nároky na minimalizaci. Mikrokontrolér z rodiny ARM Cortex-M3 svou výkonností a cenou okolo 280 Kč přesahuje potřeby vyráběného zařízení. Displej u těchto hodinek je ideální. Kombinuje nízkou spotřebu s dostačujícím efektem zobrazení. Bohužel je těžko dostupný a tak nemůže být použit při výrobě.

Uvedené amatérské řešení odpovídá představám vyráběného demonstračního zařízení. Využití snadno dostupného a levného mikrokontroléru Atmega328p je adekvátní pro požadavky zařízení. Kvůli absenci integrovaného obvodu *RTC* je však zapotřebí obvod doplnit externě nebo použít čítač/časovač mikrokontroléru. Autor zvolil externí řešení a to *RTC* obvod DS3231M. Dohromady je tak cena a spotřeba těchto zařízení vyšší než při použití čipu CC430F6137. Řešení je i přes to vhodné díky rozšířenosti a podpoře firmy Atmel, která tyto kontroléry vyrábí. Příkladem je softwarová podpora vývojovým prostředím AVR Atmel studio. Zvolený 1,3“ *OLED* displej nabízí velké rozlišení a relativně nízkou spotřebu. Zvolený poměr stran displeje však považuji za nevhodný. Hodinky jsou vizuálně nevyvážené. Naopak multifunkční tlačítko je praktické a je vhodné ho použít jako ovládací prvek. Zvolená li-pol baterie s nabíjením přes *USB* je také vhodným řešením. Za nevhodnou však považuji zvolenou obsluhu baterie. Ta je řešena dohlížecím integrovaným obvodem, který má omezené funkce. Za nepatrně vyšší cenu lze pořídit komplexnější dohlížecí obvod.

3 ZVOLENÉ ŘEŠENÍ

Možnost řešení výroby demonstračního přípravku limituje dostupnost součástek a komponentů, ze kterých se zařízení skládá. Dalším omezením je nedokonalá minimalizace obvodů a náročnost osazení *DPS*. Energetické nároky zařízení by měly splňovat odhadovanou maximální spotřebu, která klade nároky na baterii. Tato kapitola se věnuje konkrétním součástkám, které autor vybral pro vyráběný přípravek.

3.1 Řídící obvod

Pro funkci řídicího obvodu v tomto zařízení byl zvolen mikrokontrolér od firmy Atmel *Atmega328P-AU*. Při řešení autor zohlednil své zkušenosti s tímto čipem. Dále rozhodovala jeho dostupnost a jeho časté použití v jiných aplikacích s ukázkovými kódy v jazyce C.

Osmi-bitový mikrokontrolér je založen na architektuře AVR. Jeho důležité vlastnosti popisuje následující tabulka (Tabulka 1). Pouzdro *TQFP* je vhodné pro ruční osazení. Nabízí 23 vstupně výstupních pinů a tak je možné ovládat dostatečné množství periférií. Důležité jsou i specifikace týkající se odběru proudu. V aktivním režimu se spotřeba pohybuje v jednotkách miliampér v závislosti na taktu mikrokontroléru. Odběr v režimu spánku je udáván přibližně na 150 nA. Důležitá je flash paměť s prostorem 32 kB. Její velikost s rezervou postačuje pro uložení bitmapových obrázků, které budou použity k zobrazení času na displeji. Pořizovací cena tohoto řídicího obvodu je 76 Kč. [6]

Tabulka 1 - Vybrané vlastnosti kontroléru *ATmega328P*

MCU	Pouzdro	I/O Pin	I_{active}	I_{sleep}	Flash paměť	Cena
<i>ATmega328P</i>	<i>TQFP</i>	23	2 mA	~150 nA	32 kB	76 Kč

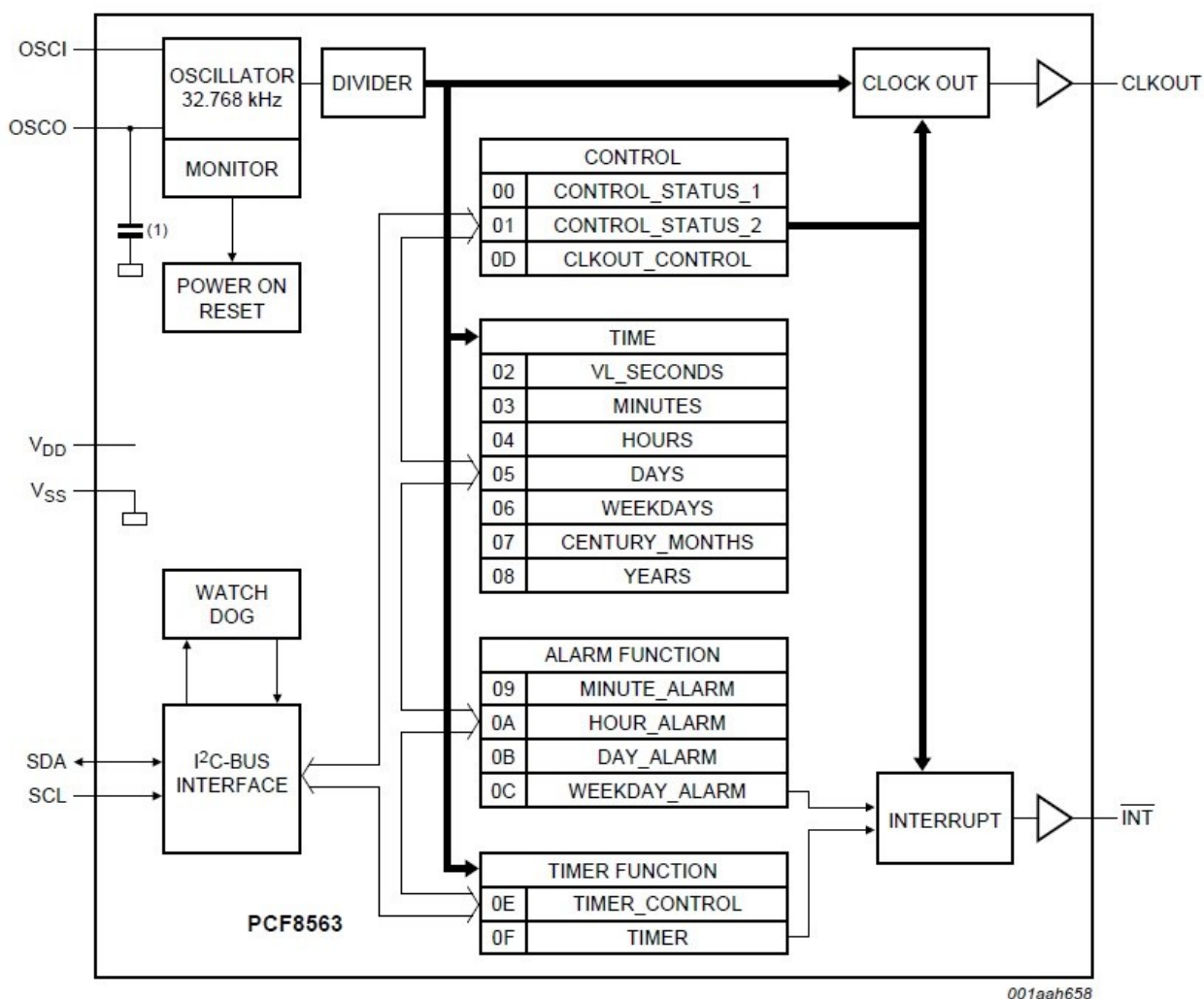
Mikrokontrolér disponuje komunikačními sběrnici *SPI* a *TWI (I2C)* a sériovou linkou *UART*, pro které má přímo dedikované piny. Se sběrnici se pracuje pomocí speciálních registrů. Jejich nastavení je detailněji popsáno ve výrobních listech. [6]

Absence integrovaného *RTC* obvodu nabízí dvě možnosti řešení jak pracovat s informací o čase. Jednou z nich je využití funkce čítač/časovač, která byla vysvětlena v první kapitole. Je ovládána nastavením příslušných registrů a řízena výpočtem algoritmem. Pro preciznější práci s časem je vhodné zvolit externí *RTC* obvod. Ten zajistí relativně přesnou informaci o čase.

3.2 Obvod reálného času

Do výběru obvodu *RTC* se promítá několik významných faktorů. V některých aplikacích je důležitá přesnost času, která se udává v ppm⁷. U přesných obvodů je tato hodnota +/- 2 ppm. Dále je pro minimalizaci podstatné, aby obvod nevyžadoval mnoho externích součástek. Pro vyráběné zařízení byl tedy zvolen dostupný *RTC* obvod PCF8563T. Hlavním důvodem byl integrovaný oscilátor s kondenzátorem a dostatečně nízká spotřeba. Pro digitální hodinky nabízí všechny potřebné kalendářní funkce a možnost nastavení dvou alarmů.

Na obrázku (Obrázek 11) je vnitřní uspořádání obvodu. Základem je oscilátor kmitající na frekvenci 32,768 kHz. Dělička upravuje frekvenci podle potřeby a poté jsou s každým taktem inkrementovány registry. K datům v registrech se přistupuje sběrnici *I2C*. [7]



Obrázek 11 - Vnitřní uspořádání obvodu PCF8563T⁸

⁷ Výraz ppm je zkratkou anglického Parts Per Million a dá se také vyjádřit v exponenciálním tvaru jako 10⁻⁶.

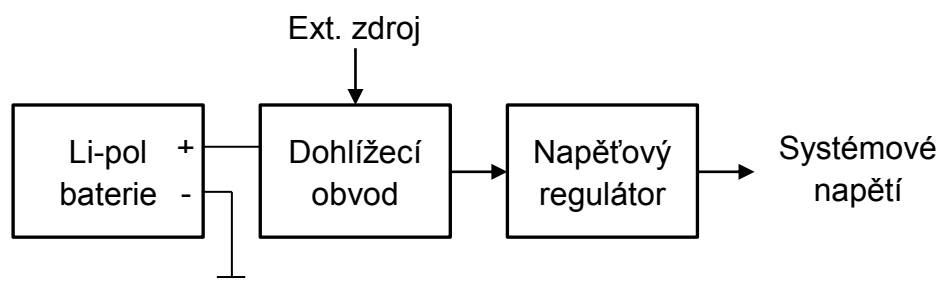
⁸ Zdroj [7]

Informace jsou ve formě *BCD* a díky obousměrné komunikaci s registry je možné data číst i zapisovat. Jedním z výstupů obvodu je pin INT, který reaguje na splnění podmínky alarmu či časovače. Na výstupní pin CLKOUT může být přiveden jeden ze čtyř signálů o definovaném kmitočtu. Tento *RTC* obvod tak může být zdrojem hodin pro jiné obvody. [7]

Obvod byl zvolen s ohledem, že zařízení nevyžaduje od obvodu reálného času extrémní přesnost a jeho energetická náročnost a funkce jsou téměř stejné jako u ostatních obvodů.

3.3 Napájecí obvod

Na napájecí obvod jsou kladeny důležité požadavky. Přímou ovlivňuje výdrž zařízení a jeho správná volba je klíčová pro všechna mobilní zařízení. Je hned několik způsobů jak lze k řešení napájení přistoupit. Při zkoumání možných řešení se nejvíce nabízí použití li-pol baterie. Ta pro bezpečné aplikování potřebuje dohlížecí obvod, který se stará o správné nabíjení. Veškerou energii potřebnou pro funkčnost zařízení tvoří blok několika obvodů, který znázorňuje ilustrace níže (Obrázek 12).



Obrázek 12 - Blokové schéma napájecího obvodu

Výběr li-pol baterie je pro řešení mobilních aplikací ideální. Má jeden z nejlepších poměrů uchované energie k hmotnosti. Zvolená jedno článková baterie, která má jmenovité napětí 3,7 V, má rozměr 30x20x3 mm s kapacitou 150 mAh. Maximální odebíraný proud vyráběného zařízení je odhadován na 50 mA. Zvolená baterie by tedy měla pokrývat potřeby vyráběného přípravku. Aby se baterie nesprávnou manipulací nepoškodila či dokonce nevzplanula, je potřeba zapojit dedikovaný dohlížecí obvod. Tím je čip MCP73871, který řídí specifické nabíjení li-pol baterií. Klíčovým faktorem jeho volby byla integrovaná elektronika, která při zapojení externího zdroje energie odpojí baterii od obvodu. Ta je poté nabíjena přes dohlížecí obvod, zatímco je systém napájen z externího zdroje. Stav baterie je možno pomocí tří pinů posílat přímo na mikrokontrolér.

Výstup z dohlížecího obvodu, ať už se jedná o napětí baterie či externího zdroje, je přiveden na regulační obvod. Jeho úkolem je napětí upravit na požadovanou úroveň. Výstup regulátoru určuje napětí pro celý systém. Do zařízení byl zvolen tzv. *LDO* regulátor díky svému malému klidovému odběru. Vybraný model LD6805K/25H upravuje vstupní napětí v rozsahu 2,3 V až 5,5 V na fixní výstupní napětí 2,5 V.

Podobné napájecí systémy využívá celá řada mobilních zařízení. Použití li-pol baterie je prostorově úsporné, ale klade nároky na dohlížecí elektroniku. Ta se dá vyřešit jedním integrovaným obvodem, který se kompletně stará o připojenou baterii a zároveň informuje řídicí obvod. Cena celého napájecího obvodu se blíží k 200 Kč, přičemž částku ovlivňuje hlavně výběr baterie. Zvolená baterie byla pořízena za 150Kč.

3.4 Displej

Displej je jedním z komponentů, který umožňuje interakci s uživatelem. Je zapotřebí zvolit takový displej, který zohlední náročnost zobrazované informace. Musíme také uvažovat proudový odběr displeje, který bývá u mobilních zařízení tím největším ze všech součástí. Nabízí se tedy použití *OLED* displeje, který kombinuje malý odběr a kvalitní podání zobrazované informace. Je vyráběn v rozměrech od 0,96“.

Zvolen byl barevný 1,04“ *OLED* displej (Obrázek 13) s řadičem SSD1332. Má dostatečně velkou aktivní plochu 20 x 13,5 mm a celkový rozměr panelu je přibližně 25x25 mm. Rozlišení je 96x64 bodů s pasivní maticí. Komunikace je zprostředkována přes *SPI* nebo paralelně k jednotlivým portům. Jeho spotřeba není nejmenší a to především kvůli jeho barevnosti. Je to kompromis pro výsledný efekt zobrazení a budoucí modifikace. Tabulka (Tabulka 2) shrnuje vlastnosti použitého displeje.

Tabulka 2 - Vlastnosti OLED displeje

Displej	Provedení	Barevnost	$I_{\text{activeMAX}}$	I_{sleep}	Cena
1,04“ OLED	COG ⁹	65k / 256	50 mA	1 μ A	150 Kč



Obrázek 13 - Zvolený OLED displej

⁹ Zkratka COG znamená Chip On Glass. Jedná se o mechanické provedení displeje, kde je řadič umístěn na skleněném panelu.

3.5 Uživatelské rozhraní

Obsluha zařízení je nedílnou součástí digitálních hodinek. Zpracování by tak mělo zajišťovat přehledné ovládání a dopřát uživateli jistý komfort. Ovládací prvky spolu se softwarem jsou navrženy tak, aby uživatel mohl řídit všechny funkce hodinek jednou rukou. Samotné funkce jsou pak řízeny algoritmem, který je možné editovat přes vyvedené programovací rozhraní.

Pro splnění požadavků obsluhy zařízení bylo zvoleno malé třípolohové tlačítko MCPL3-AC-V. Jeho půlkruhová geometrie se podobá kolečku na myši. Přes přepjatý prvek je pohyb kolébkově přenesen na první polohu tlačítka. Opačný směr pohybu spíná druhý kontakt. Stlačením prvku ve výchozí poloze je sepnuta třetí poloha tlačítka. Díky tomu je možné všechny funkce ovládat jedním prstem. Změnu celého softwaru umožňuje rozhraní *ISP*, které ovšem není pevně připájeno na desku, ale jsou pouze vyvedené potřebné plošky na kraj desky. Samotné rozhraní se pak nasune na tyto plošky a v případě nedokonalosti spoje se připájí. Toto řešení je prostorově výhodné a přístupné. Z kapitoly 3.3 o napájecím obvodu vyplývá potřeba připojení externího napětí. To je vyřešeno použitím microUSB konektoru. Pro nabití baterie pak stačí uživateli propojit hodinky s *USB* portem počítače. O stavu nabíjení je informován příslušnou indikací na displeji. Mimo předání informace pomocí displeje zařízení disponuje také zvukovou odezvou. Ta je generována malým magnetickým měničem ABMT-801-RC. Uživatel tedy dostává informace od zařízení pomocí těchto dvou výstupních prvků. Naopak uživatel k zařízení přistupuje pomocí třípolohového tlačítka, které mu umožňuje ovládat veškeré funkce jedním prstem.

3.6 Shrnutí řešení

Zvolené součástky dávají celému zařízení určité vlastnosti a parametry, které splňují předpoklady z kapitoly 1.1. Následující tabulka (Tabulka 3) představuje kompletní pohled na odběr proudu jednotlivých obvodů v aktivním a spánkovém režimu a také pořizovací cenu součástek.

Tabulka 3 - Přehled použitých obvodů

Obvod	Typ	I_{active}	I_{sleep}	Cena
ATmega328P-AU	MCU	2 mA	150 nA	76 Kč
MCP73871	Řízení baterie	50 μ A	50 μ A	50 Kč
LD6805K/25H	LDO	100 μ A	100 μ A	4 Kč
1,04" OLED	Displej	50 mA ¹⁰	1 μ A	150 Kč
PCF8563T	RTC	800 μ A	300 nA	45 Kč
Li-Pol 150mAh	Baterie	/	/	150 Kč
Σ	/	\sim 53 mA	\sim 151,5 μ A	475 Kč

Z tabulky (Tabulka 3) lze dopočítat teoretickou dobu provozu na baterii. Při zanedbání spotřeby ostatních součástek, které mají minimální odběr, a za předpokladu, že zařízení se bude v aktivním režimu nacházet 10 minut denně (aktivování hodinek a opětovný režim spánku) a po zbytek dne se budou nacházet v režimu spánku, dojdeme ke vztahu 1. Snadno se pak vypočítá vztah 2 a 3.

$$T_K = \frac{C_{bat}}{I_{active} \cdot t_{D-active} + I_{sleep} \cdot t_{D-sleep}} = \frac{150}{3,66 + 3,61} = \mathbf{20,63 \text{ dní}} \quad (1)$$

T_K - doba kombinované výdrže C_{bat} - kapacita baterie, I_{active} - proud v aktivním režimu, I_{sleep} - proud ve sleep režimu, $t_{D-active}$ - $t_{D-sleep}$ - doba trvání režimu

$$T_a = \frac{C_{bat}}{I_{active}} = \frac{150}{53} = \mathbf{2,83 \text{ hodin}} \quad T_a - \text{Doba výdrže v aktivním režimu} \quad (2)$$

$$T_s = \frac{C_{bat}}{I_{sleep}} = \frac{150}{0,151} = \mathbf{41,4 \text{ dní}} \quad T_s - \text{Doba výdrže ve spánkovém režimu} \quad (3)$$

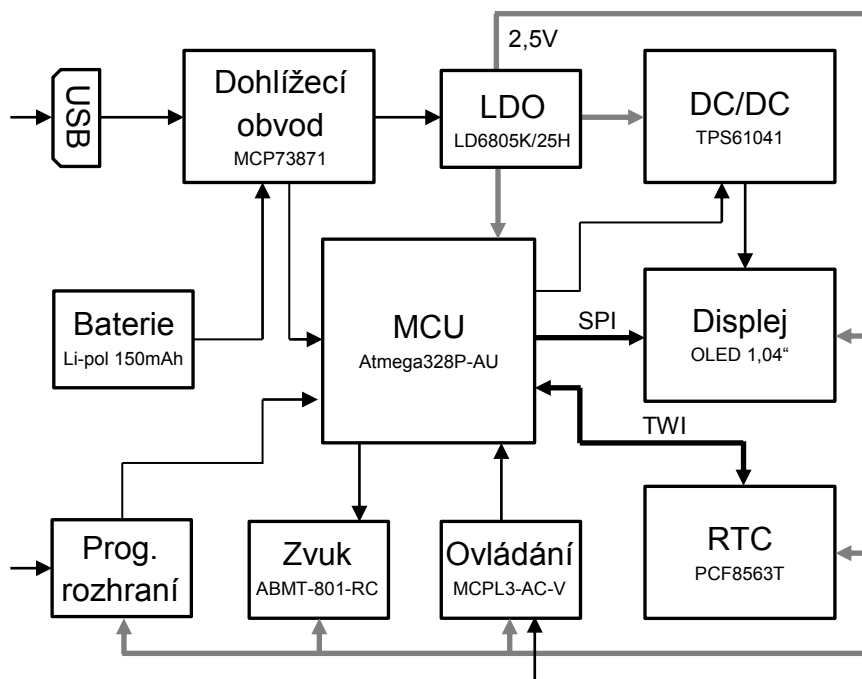
¹⁰ Odběr displeje závisí na barevném podání a právě aktivních pixelech. Při zobrazení pouhých číslic jednou barvou spotřeba značně klesne a to přibližně na 20 mA.

4 NÁVRH DESKY A OBVODŮ

Zařízení bylo od začátku koncipováno jako přenosný modul s ohledem na malý rozměr a nízkou spotřebu. Proto byla zvolena dvouvrstvá deska plošných spojů. Důležité bylo správně rozmístit součástky tak, aby výsledné zařízení bylo co možná nejtenčí a nepřesahovalo rozměry 30x30 mm. Deska je osazena součástkami z obou stran, přičemž displej je zapájen zespodu a přes okraj desky je přehnut na horní stranu. Baterie je pomocí dvou vodičů připájena do dvou vrtaných otvorů a je umístěna na zadní straně zařízení. Modul tedy využívá tzv. sendvičové konstrukce, kdy jsou jednotlivé části vrstveny na sebe. Obrázek (Obrázek 1) v úvodní kapitole znázorňuje takové rozložení. V této kapitole je nejdříve uvedeno celkové blokové schéma zařízení a poté jsou vysvětlena konkrétní zapojení obvodů, která byla navržena v softwaru EAGLE 6.5.0. Na základě navržených obvodů byla vytvořena *DPS*. Kompletní schéma zapojení spolu s *DPS* a seznamem součástek a se nachází v přílohách A, B, C.

4.1 Celkové blokové schéma zapojení

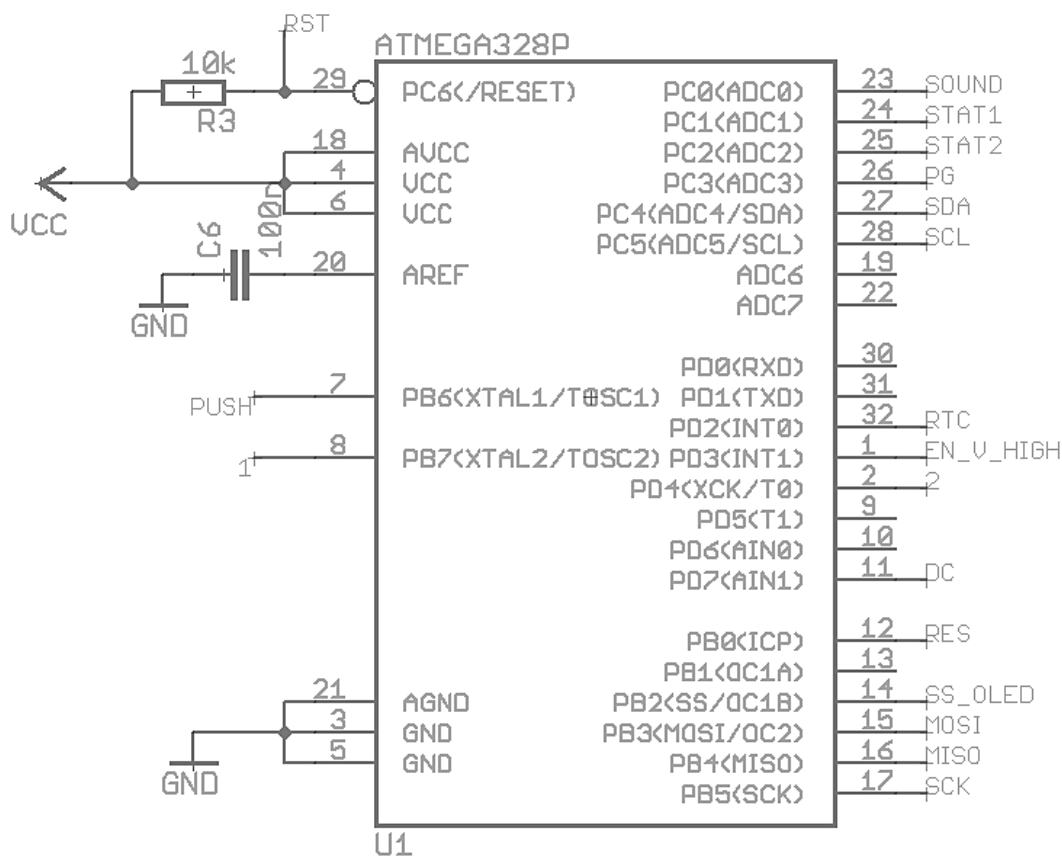
Blokové schéma (Obrázek 14) odpovídá funkčnímu principu zařízení. Uprostřed je *MCU*, které řídí chod systému a jsou na něj připojeny různé periferie. Ty buď mikrokontrolér aktivuje pomocí jednoho z pinů nebo s nimi komunikuje přes sběrnice I2C a SPI. Systém je napájen napětím 2,5 V z *LDO* regulátoru. Uživatel přímo přistupuje ke konektoru *USB*, programovacímu rozhraní a hlavně k ovládání. Některé funkční bloky potřebují pro svou aplikaci doplňující kondenzátory, odpory a cívky.



Obrázek 14 - Blokové schéma zapojení vyráběných hodinek

4.2 Zapojení Atmega328P

Zapojení mikrokontroléru (Obrázek 15) je důležité pro celkové uspořádání *DPS*. Jednotlivé piny jsou zapojeny tak, aby bylo na omezeném prostoru desky co nejlépe možné zapojit všechny obvody.

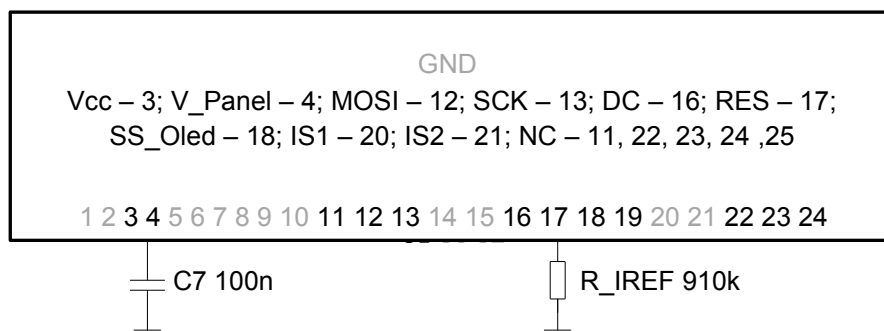


Obrázek 15 - Zapojení ATmega328P-AU

Na levé straně je připojeno napájení a zemní piny. Reference (pin 20) není využita a tak je přes blokovací kondenzátor připojena na zem. Kondenzátor by měl být co nejbližší pinu. Dále je tam RESET (pin 29), který slouží pro resetování *MCU*. Pro provedení resetu stačí propojit tento pin se zemí. Také je tento pin jedním z vodičů programátoru. Piny 7, 8 jsou vstupními piny, které jsou připojené na ovládání. Pravá strana obsahuje zapojení periférií obvodu. K dispozici máme porty PB, PC a PD, které slouží jako vstupně výstupní piny *MCU*. Sběrnice *I²C* využívá dedikovaných pinů 27, 28 a pro *SPI* jsou to piny 14,15,16,17. Pin 32 je určen k externímu přerušení a je vyveden na *RTC*.

4.3 Zapojení displeje

Při zapojování displeje nastal problém při určování funkčnosti jednotlivých pinů. Displej je vybaven integrovaným řadičem SSD1332 a v jeho výrobních listech není uvedeno konkrétní rozložení pinů pro tento model displeje. Obsahuje však rozložení vlastního řadiče, který má přes 100 pinů. Ty jsou tak malé, že pouhým okem nejsou vidět. Byl tedy pořízen makrosnímek kvalitním fotoaparátem a postupně bylo určeno všech 27 pinů tohoto displeje. Výsledné zapojení představuje následující obrázek (Obrázek 16).



Obrázek 16 - Rozložení OLED displeje

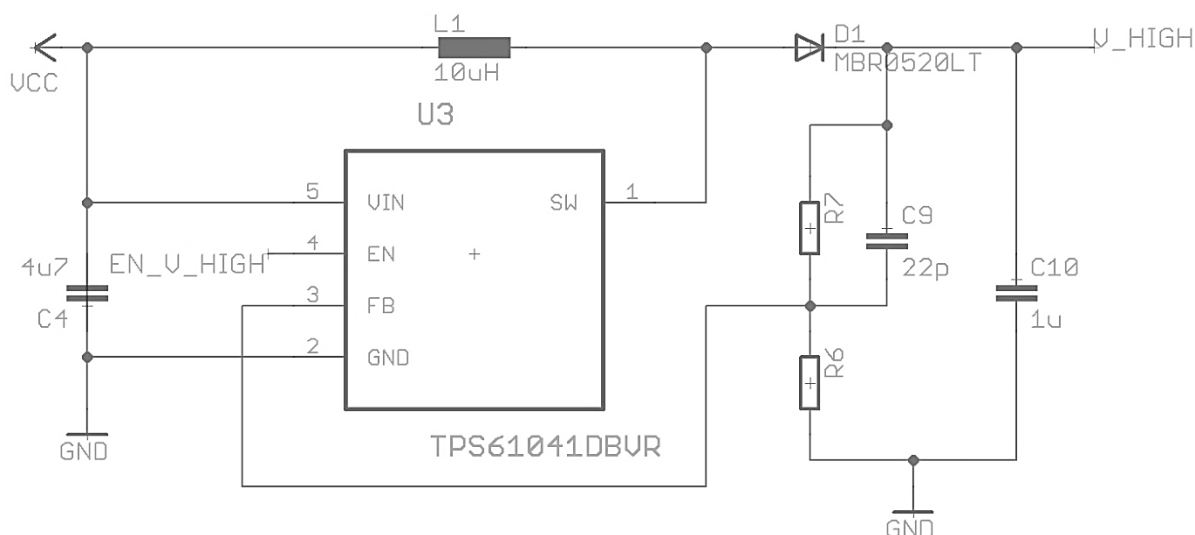
OLED displej potřebuje dvě různá napájení. Jedním z nich je V_{cc} (2,4 V - 3,5 V), které slouží jako napájení logických obvodů řadiče. Druhé je vysokoúrovňové V_{Panel} (7 V - 18 V). Komunikace displeje s *MCU* je zprostředkována přes sběrnici *SPI*. Funkce pinů popisuje tabulka (Tabulka 4).

Tabulka 4 - Funkce pinů displeje

Pin	Popis funkce
IS1, IS2	Slouží k výběru typu komunikace s MCU
MOSI, SCK	Zajišťují komunikaci přes SPI (MOSI, SCK)
SS_Oled	Povoluje komunikaci s MCU
RES	Resetovací signál
DC	Určuje, zda příchozí bity jsou data nebo příkaz
5	V_{COMH} - Vysokonapěťový výstup signálů Doporučuje se zapojení kapacity mezi pin a zem
19	Pomocí R_{IREF} určuje referenční proud pro nastavení jasu

4.3.1 DC-DC konvertor

Displej pro svou funkčnost potřebuje napětí 7 – 18 V. To však baterie s 3,7 V není schopna poskytnout a proto je nutné toto napětí zajistit jiným způsobem. Tím je realizace tzv. DC-DC step up konvertoru, který ze vstupního napětí generuje napětí vyšší. Pro tento účel byl zvolen integrovaný obvod TPS61041.

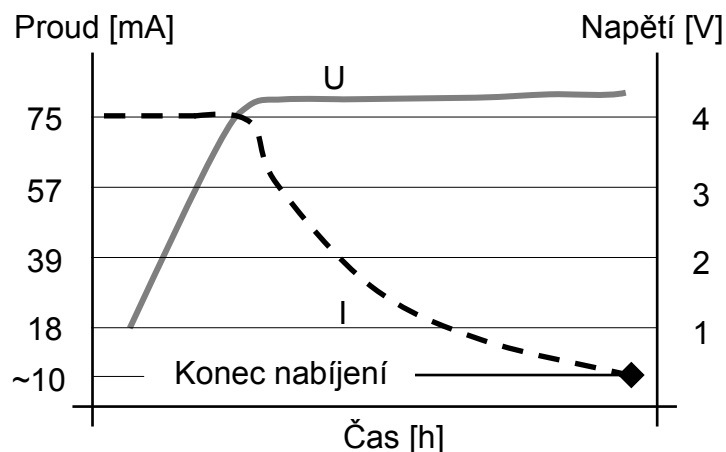


Obrázek 17 - Zapojení DC-DC konvertoru

Toto zapojení (Obrázek 17) je schopno ze vstupního napětí 1,8 V až 6 V vygenerovat výstupní napětí až 28 V. Integrovaný obvod kmitá na frekvenci 1 MHz, která spíná/rozepíná obvod. Při rozepnutí obvodu se cívka snaží udržet proud a tak se na ní zvýší napětí. Schottkyho dioda D1 je volena právě kvůli rychlosti změn stavů a je přes ni nabíjen kondenzátor C10, na který je pak zapojena samotná zátěž. Odporů R6 a R7 společně s C9 tvoří napěťový dělič, který nastavuje celkové výstupní napětí (vztah 4). Po dosazení konkrétních hodnot rezistorů dostáváme napětí podle vztahu 5. Účinnost těchto obvodů se pohybuje od 75% do 90% v závislosti na vstupním napětí. Výstupní proud může být od desítek mA až přes 1 A. Pro nižší odběr zařízení je celý obvod spínán mikrokontrolérem přes pin EN. [8]

$$U_{OUT} = 1,233 * \left(1 + \frac{R_7}{R_6}\right) \quad (4)$$

$$U_{OUT} = 1,233 * \left(1 + \frac{1,8 M\Omega}{200 k\Omega}\right) = 12,3 V \quad (5)$$



Obrázek 19 - Nabíjecí schéma li-pol baterie

Nastavení pinů PROGx z předchozí tabulky (Tabulka 5) je velice důležité pro splnění zmíněného nabíjecího schématu baterie (Obrázek 19). Pin PROG1 předpokládá zapojení rezistoru mezi zem, jehož hodnota nastavuje konstantní nabíjecí proud baterie. Maximální nabíjecí proud vybrané 150 mAh baterie je 0,5 C. Baterie má tedy maximální nabíjecí proud 75 mA. Podle vztahu 6 je vypočítána potřebná hodnota rezistoru R_PROG1 .

$$R_PROG1 = \frac{1000\text{ V}}{I_{REG}} = \frac{1000\text{ V}}{75\text{ mA}} \cong 13,5\text{ k}\Omega \quad (6)$$

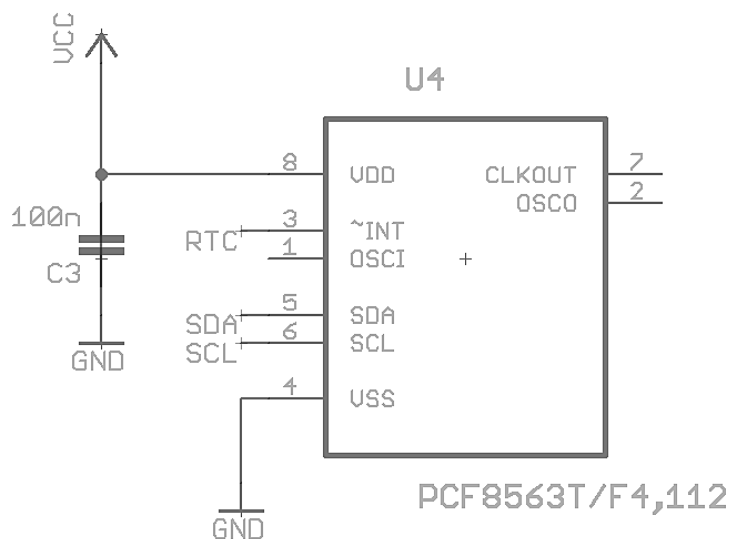
Hodnota vypočítaného odporu udává maximální nabíjecí proud, proto byla jeho hodnota zvolena 15 k Ω . Ta zajišťuje určitou rezervu a neměla by výrazně ovlivnit rychlost nabíjení. Pin PROG3 pomocí rezistoru zapojeného mezi zem umožňuje nastavení hranice ukončení nabíjení. Když nabíjecí proud klesne na nastavenou hranici, zareaguje obvod ukončením nabíjení. Vše opět znázorňuje obrázek výše (Obrázek 19). Hodnota rezistoru je doporučena mezi 5 k Ω až 100 k Ω , přičemž byla zvolena právě nejvyšší doporučovaná hodnota. Výpočet R_PROG3 uvádí vztah 7. Zařízení se teoreticky bude nabíjet delší dobu, ale zároveň s tím i na vyšší úroveň napětí.

$$I_{UKONČENÍ} = \frac{1000\text{ V}}{R_PROG3} = \frac{1000\text{ V}}{100\text{ k}\Omega} = 10\text{ mA} \quad (7)$$

Pin PROG2 už jen omezuje vstupní proud z USB portu. Dává na výběr mezi dvěma hodnotami 100 mA a 500 mA. Pro chod zařízení by neměla mít volba vstupního proudu výrazný vliv. Použit se teoreticky mohou obě možnosti a tak byla zvolena hodnota 500 mA. [9][10]

4.5 Zapojení RTC

Obvod *RTC* (Obrázek 20) vyžaduje minimální nároky na zapojení. Komunikaci s mikrokontrolérem zajišťuje sběrnice *TWI* pomocí pinů *SDA* a *SCL*. Speciální funkci má pin \sim INT, který je připojen na řídicí obvod a slouží jako výstupní pin pro alarm a časovač. Pin *CLKOUT* je nevyužitý a je výstupem frekvenčně programovatelného signálu. Výstup tohoto pinu se dá programově nastavit na čtyři definované kmitočty. Poslední dva piny slouží k zapojení externího krystalu.

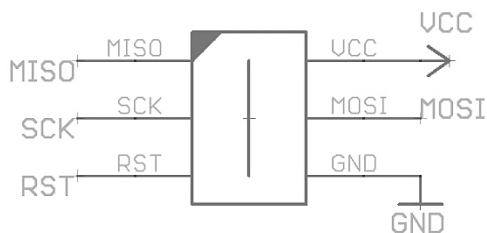


Obrázek 20 - Zapojení obvodu PCF8563T

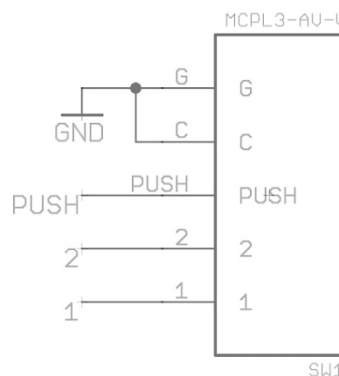
Ve výrobních listech zvoleného mikrokontroléru je doporučené zapojení, které obsahuje externí pull up rezistory pro hodinový signál *SCL* a datový *SDA*. Důvodem zapojení externích rezistorů je příliš velká hodnota těch interních, která je 30 k Ω . Komunikace *TWI* ze svého principu vyžaduje zapojení vodičů na vysokou úroveň. Přenos dat a hodin je zajištěn vhodným nulováním signálu. Jelikož mimo zapojeného interního rezistoru je v obvodu i parazitní kapacita, vzniká zde RC člunek, který svým nabíjením a vybíjením deformuje hrany vytvořené nulováním. Časová konstanta tohoto jevu je dána součinem *R* a *C*. Při zvolení velké rychlosti datového přenosu, a tím i vysoké frekvence, může dojít ke zkreslení přenášené informace či dokonce k nefunkčnosti komunikace. Proto při použití interních rezistorů musí být zvolena dostatečně nízká rychlost přenosu. Z důvodu minimalizace obvodu bylo zvoleno řešení s interními pull up rezistory.

4.6 Zapojení ostatních obvodů

Programátor – Zapojení níže (Obrázek 21) umožňuje nahrání softwaru do mikrokontroléru. To se provede přivedením šesti vodičů na příslušné plošky, které jsou propojeny se souhlasnými piny mikrokontroléru. Řešení z mechanického pohledu popisovala kapitola 3.5.



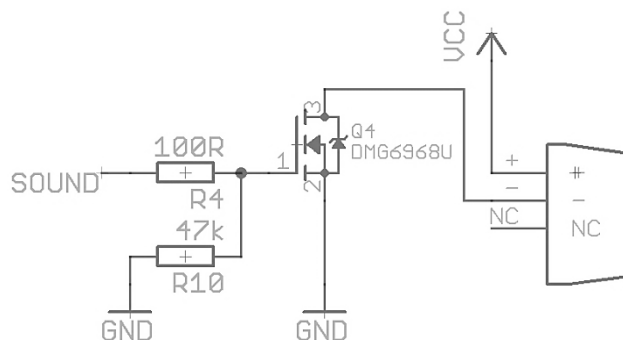
Obrázek 21 - Zapojení programátoru



Obrázek 22 - Zapojení tlačítka

Ovládací prvek – Zapojení obvodu ilustruje obrázek vpravo (Obrázek 22). Z principu zapojení je nutné, aby bylo spojení mezi tlačítkem a řídicím obvodem na vysoké úrovni. Toho se dosáhne aktivováním pull up rezistorů mikrokontroléru. Zmáčknutí tlačítka propojí příslušný pin MCU se zemí a následuje naprogramovaná reakce na stisk tlačítka.

Zvuk – Magnetický měnič je připojen na zem přes N-MOSFET tranzistor. Ten se na základě impulsu z *MCU* otevře a tak se obvod měniče uzavře a rozezní se. Příklad nízké úrovně tranzistor uzavře a obvod měniče je rozpojen. Rezistory R4 a R10 slouží jako napěťový dělič, který upravuje napětí na gate tranzistoru.



Obrázek 23 - Zapojení obvodu zvuku

5 VÝROBA

Navržené zařízení je specifické svým rozměrem a použitím součástek s malými smd pouzdry. Při výběru pouzder musí konstruktér nejdříve správně dimenzovat jednotlivé součástky, aby nedošlo například k překročení maximálního ztrátového výkonu rezistoru. Ten se pohybuje v řadě 0402 od 63 mW do 100 mW. Zohlednit se musí i výběr kondenzátorů, které jsou určeny pro jisté maximální napětí. Proto nejdříve byly zohledněny nároky jednotlivých obvodů a na základě toho byla zvolena právě řada 0402, která umožňuje ruční osazení. Pouzdra integrovaných obvodů byla vybrána tak, aby je bylo možné ručně osadit a zároveň byla co nejvíce minimalizována. Pouzdro dohlížecího obvodu baterie MCP73871 se vyrábí pouze v provedení QFN, které vyžaduje vyšší nároky pro ruční osazení. Ještě vyšší nároky klade pouzdro DFN101C-4 regulátoru napětí LD6805K s celkovým rozměrem 1x1mm. Tyto obvody vyžadovaly použití speciální techniky osazení horkým vzduchem. Výroba *DPS* o rozměrech 30x30 mm, navíc s malými izolačními mezerami cest a vysokým počtem prokovení, byla v amatérských podmínkách nerealizovatelná. Proto byla výroba zadána specializované firmě Pragoboard. Deska byla pokryta nepájivou maskou, která usnadnila osazení. Vyrobená a kompletně osazená deska je na obrázku níže (Obrázek 24).



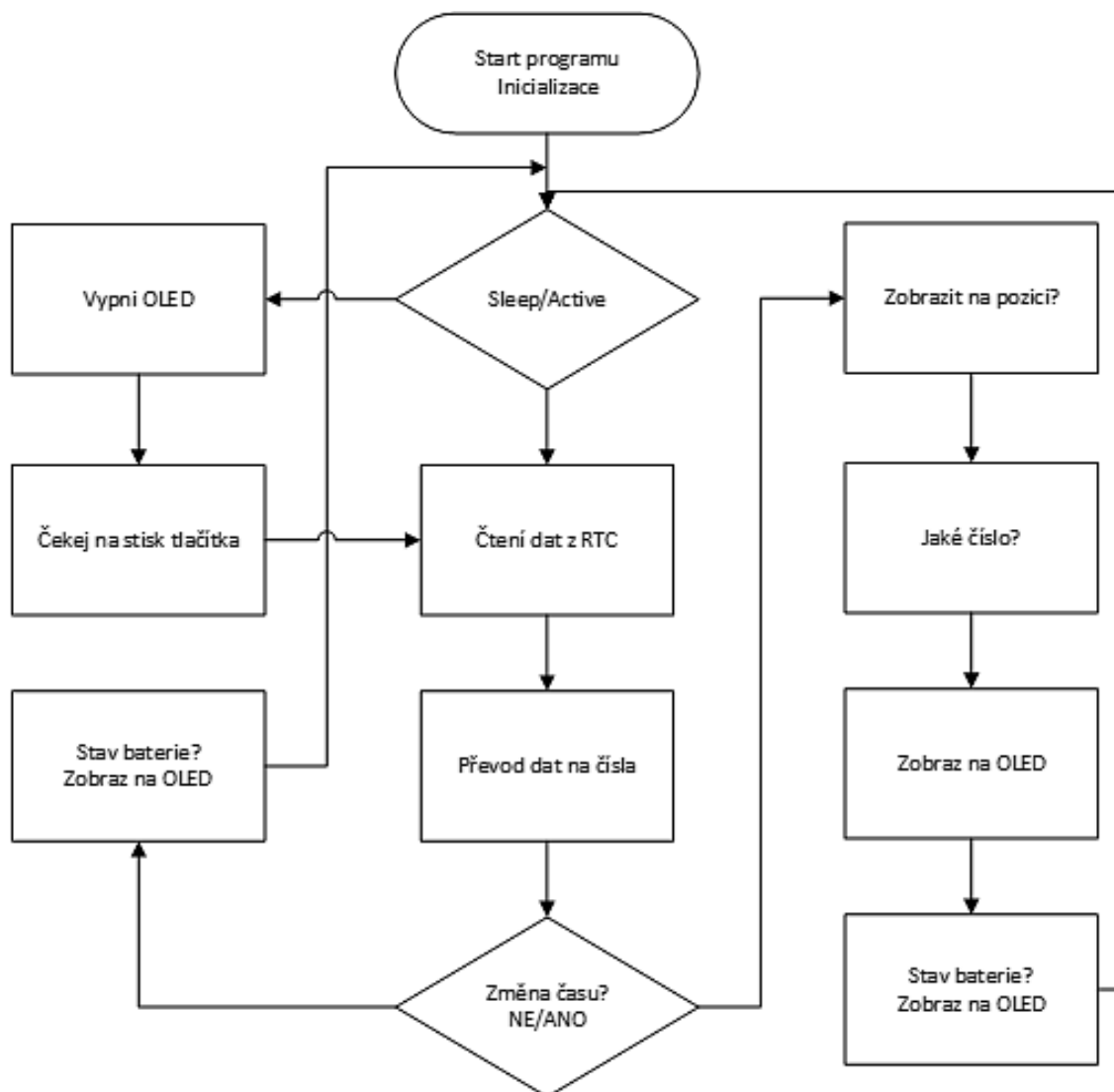
Obrázek 24 - Vyrobená a osazená DPS

Konstrukce zařízení vyžadovala specifické nároky na spotřebu a rozměr. Proto byly součástky voleny na základě těchto požadavků. Jako nepříliš vhodné se ukázalo použití zmíněného regulátoru, jehož rozměr kladl na ruční osazení vysoké nároky. Deska spolu se všemi součástkami stála do 1000 Kč. Za výrobu desky si firma Pragoboard účtuje 500 Kč/dm².

6 SOFTWARE

Program by měl být navržen tak, aby plnil základní funkce hodinek na základě povelů obsluhy. Uživatel očekává určité chování systému, které má být srozumitelné a jasně definované. Řídící algoritmy, které jsou popsány v této kapitole, slouží pouze k základní obsluze jednotlivých periférií a interpretování informace uživateli. Popisovaný software nenabízí funkce alarmu a měření času (stopek). Pro jednoduchost se změna času provádí přes programovací rozhraní editací *SW*, nikoliv přes ovládací prvky. Tyto funkce mohou být později doplněny rozšířením programu.

Software byl navržen jako sekvenční algoritmus v jazyce C v prostředí Atmel Studio 6.1. Základem programu je několik dílčích procesů, které jsou potřeba pro získání nebo zprostředkování dat. Nejdříve je však potřeba nastavit pomocí programu parametry hardwarových prvků jako jsou jednotlivé piny mikrokontroléru, tlačítka a jiné. Dále potřebujeme získat informaci z *RTC*, zpracovat ji a následně zobrazit na displeji. Informaci získáme pomocí sběrnice *I2C* čtením jednotlivých registrů obvodu reálného času. Tato data jsou uložena v dedikovaném registru mikrokontroléru a jsou předána jako proměnná do algoritmu, který je zpracuje a vyhodnotí. Sběrnici *SPI* je výstup algoritmu poslán po bajtech na displej, kde se uloží do *RAM* paměti a s taktem displeje je zobrazen. K detailnějšímu pohledu na softwarovou část slouží vývojový diagram (Obrázek 25) a podrobnější popis jednotlivých bloků.



Obrázek 25 - Vývojový diagram softwaru

Inicializace – V tomto bloku dochází k nastavení všech periférií a samotného *MCU*. Zavoláním metody se inicializují jednotlivé periférie. Dojde tak k nastavení pinů na vstupní a výstupní, zapojení pull-up rezistorů, nastavení parametrů *SPI* a *TWI* sběrnic a také nastavení vlastností *RTC* a displeje. Zároveň jsou do příslušných registrů *RTC* nahrána data reprezentující nastavený čas.

Sleep/Active – Tato část je přímo řízena uživatelem. V každém cyklu programu se nejdříve kontroluje, zda-li nebylo stisknuto tlačítko. Při stisku tlačítka dojde k vyhodnocení podmínky a je volána metoda, která vypne displej, a čeká v cyklu *while* na další stisk tlačítka, který opět zapne displej. Metoda se ukončí a program se opět vrací do hlavní smyčky a zobrazí čas.

Čtení dat z RTC – Zavoláním příslušné metody dojde ke čtení registrů *RTC*. Ty se automaticky inkrementují. Zadáním příslušné adresy registru se postupně vyčtou minuty a hodiny ve formátu *BCD* a jsou uloženy do registru *TWDR*. Komunikace je zřízena *TWI* sběrnici.

Převod dat na čísla – *BCD* data získána z *RTC* mají 8 bitů. V horním nibblu je obsaženo číslo na pozici desítek, v dolním nibblu pak číslo jednotek. Registr minut může nabývat hodnot 0 až 59. Vyčteme-li z registru minut *RTC* kombinaci 00110101_{BCD}, potřebujeme ji rozložit na horní a dolní nibble. Řešením je například kód:

```
#define HI_NIBBLE(b) (((b) >> 4) & 0x0F)
#define LO_NIBBLE(b) ((b) & 0x0F)

DATA_Minutes = TWI_Read(REG_ADDR_MINUTES);

MinutesDES = HI_NIBBLE(DATA_Minutes);
MinutesJED = LO_NIBBLE(DATA_Minutes);
```

Tím dostaneme v proměnné *MinutesDES* kombinaci 00000011b a v *MinutesJED* se objeví kombinace 00000101 a ty představují čísla 3 a 5. S těmito bajty už můžeme pracovat.

Změna času? – Jelikož hodinky zobrazují pouze minuty a hodiny, mění se obraz displeje právě jednou za minutu. Porovnáváme tedy nově vyčtené bajty z *RTC* s předchozími, které jsou uloženy v pomocných proměnných. V případě rovnosti se nic neděje a program pokračuje. Nerovnají-li se tyto dva bajty, došlo ke změně času a je nutné vykreslit nový čas na displej. Zároveň se nový bajt přeuloží do pomocné proměnné.

Zobrazit na pozici? – K vykreslení času na displej jsou použity obrázky uložené v programové paměti a je nutné znát pozice pro vykreslení. Ty jsou čtyři a to pro hodiny a minuty tak, že zvlášť zobrazujeme desítky a jednotky. Postupně se tedy ptáme, zda-li nedošlo ke změně času na některé z pozic. Změní-li se například jednotka minut, víme, že je potřeba změnit poslední číslo na displeji.

Jaké číslo? – Jaké číslo se má zobrazit zjistíme například metodou níže. Jedná se o převod binárního čísla do dekadického. Binární číslo vložené do metody je převedeno na dekadické, které je výstupem metody. Tímto číslem lze pak vybrat příslušný index z pole, ve kterém jsou uloženy jednotlivé číslice 0 - 9.

```
int Bin_to_Dec(uint8_t BIN)
{
    long int zbytek,sum=0,mocnina=0;
    while(BIN>0)
    {
        zbytek = BIN%10;
        BIN = BIN/10;
        sum = sum + zbytek * pow(2,mocnina);
        mocnina++;
    }
    return sum;
}
```

Zobraz na OLED – Nejdříve se určitá výseč displeje smaže zavoláním funkce OLED_Clear(), a pak se určí začátek a konec vykreslování číslice na displeji metodou OLED_urci_pozici(), tedy opět 4 fixní pozice, protože jsou všechny stejné – 15x25 pixelů. Poté se pomocí SPI sběrnice posílají jednotlivé bajty konkrétního obrázku.

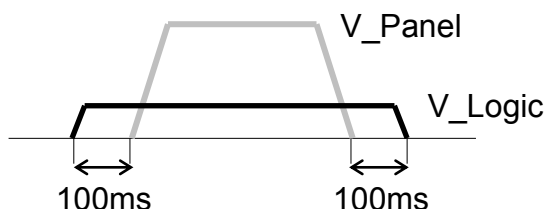
```
void OLED_Draw(const int Pole [][])
{
    int i = 0;
    OLED_Clear(SOURADNICE);
    OLED_urci_pozici(SOURADNICE);
    for (i,i<375,i++)
    {
        SPI_Transfer_Data(pgm_read_byte(Pole_Cislic_BMP[x][i]));
    }
}
```

Stav baterie – Tři stavové výstupy dohlížecího obvodu baterie jsou připojeny na piny MCU. V programu se určí možné kombinace těchto výstupů podmínkami. Je-li na pinech řídicího obvodu, které jsou nastaveny jako vstupní, právě jedna z těchto kombinací, je vyhodnocena podmínka a na displej se pošle příslušný obrázek.

6.1 Řadič SSD1332 a GDDRAM

Displej je vybaven řadičem SSD1332. Ten neobsahuje sadu fontů a tak je nutné ji nahrát do uložení. Tím může být programová paměť mikrokontroléru nebo v případě velkého počtu dat může posloužit externí paměť. Tou může být například microSD karta. V tomto případě je pro zobrazení informací na displej použito bitmapových obrázků. Díky omezenému počtu zobrazovaných obrázků s rezervou postačuje interní paměť *MCU*.

Řadič je ovládán přes *SPI* sběrnici základní instrukční sadou, která je obsažena ve výrobních listech. Jedná se například o příkazy pro souřadnice vykreslení, smazání, ale také kontrast, barevnost, zapnutí/vypnutí displeje a mnoho dalších. Důležitá pro životnost displeje je právě procedura zapínání a vypínání displeje. Správně by měla být doba mezi napětím pro logické obvody (2,5 V) a napětím pro panel (12 V) minimálně 100 ms. U vypínání je tomu naopak. Vše znázorňuje obrázek níže (Obrázek 26). Kód pro spouštěcí a vypínací proceduru je uveden dále.



Obrázek 26 - Spouštěcí a vypínací procedura displeje

```
void OLED_Init()
{
    DDRD |= (1<< PD3);
    DDRB |= (1<< PB0);
    RES=0;
    _delay_ms(100);
    RES=1;
    SPI_Transfer_Cmd(0xAE); ... Display OFF
    SPI_Transfer_Cmd(0x00);
    .
    .
    .
    SPI_Transfer_Cmd(0xA4);
    EN_V_HIGH=1; ... V_PANEL
    _delay_ms(100);
    SPI_Transfer_Cmd(0xAF); ... Display ON
}

void OLED_quit()
{
    SPI_Transfer_Cmd(0xAE);
    _delay_ms(200);
    EN_V_HIGH = 0;
    _delay_ms(200);
}
```

Řadič obsahuje tzv. Graphic Display Data RAM (*GDDRAM*), což je bitová paměť, do které je namapován obrázek. Její velikost je 96x64x16 bitů, tedy 98 304 bitů. Rozdělena je tak, že každý z 6 144 pixelů (96x64) obsahuje 16 bitů. Ty poskytují informaci o barvě, jakou se má pixel rozsvítit. Na obrázku (Obrázek 27) je tučně vyznačen jeden pixel displeje.

	Collum 1			Collum 2			Collum 3	
Row 1	R4	G5	B4	R4	G5	B4	R4	G5
	R3	G4	B3	R3	G4	B3	R3	G4
	R2	G3	B2	R2	G3	B2	R2	G3
	R1	G2	B1	R1	G2	B1	R1	G2
	R0	G1	B0	R0	G1	B0	R0	G1
		G0			G0			G0
Row 2								

Obrázek 27 - Rozložení GDDRAM řadiče SSD1332

K dispozici jsou dvě podání barev, která se dají vybrat pomocí příkazu z instrukční sady. První je 65k (Obrázek 28), kde je využito všech 16 bitů pro barevnost. To znamená, že pro rozsvícení jednoho pixelu řadič potřebuje dva bajty dat. Při počtu 375 pixelů pro jednu číslici se dostáváme na 750 bajtů. Číslic je celkem 10 a tedy obrázky by zabíraly 7,5 kB paměti.

Bit	7	6	5	4	3	2	1	0
První bajt	B4	B3	B2	B1	B0	G5	G4	G3
Druhý bajt	G2	G1	G0	R4	R3	R2	R1	R0

Obrázek 28 - Bitová posloupnost pro barevnost 65k

Druhé podání barev je 256 (Obrázek 29), tedy 2^8 . Z toho je zřejmé, že k rozsvícení jednoho pixelu si vystačíme s 8 bity a celá problematika se zjednodušuje, jelikož data posíláme přes *SPI* právě po 8 bitech. Počet bajtů v paměti se tak sníží na polovinu. Protože má datová paměť jen 2kB, je i tak nutné tyto obrázky uložit do programové paměti, která je 32 kB. [11]

Bit	7	6	5	4	3	2	1	0
První bajt	B2	B1	B0	G2	G1	G0	R1	R0

Obrázek 29 - Bitová posloupnost pro barevnost 256

7 OVĚŘENÍ ZAŘÍZENÍ

U vyrobeného demonstračního přípravku se podařilo oživit jen některé předpokládané funkce zařízení. Hlavní část zařízení, řídicí obvod, funguje bez problémů a lze ovládat všechny vstupně výstupní piny. Jeho funkčnost je podmíněna správně fungujícím napájecím obvodem, který se stará o obsluhu baterie. Dalším předpokladem je možnost nahrání softwaru do řídicího obvodu pomocí *ISP* rozhraní, které je bezproblémové. Uživatel má možnost zařízení ovládat pomocí třípolohového tlačítka, které se ukázalo po mechanické i praktické stránce jako velice vhodné.

Při návrhu zařízení došlo k pochybení ve výběru obvodu reálného času. Zvolený obvod vyžaduje pro svou činnost externí krystal. Ten však není součástí návrhu. Důvod absence krystalu byla co největší možná minimalizace zařízení a nesprávná úvaha autora o možnostech využití zvoleného *RTC* obvodu PCF8563T. Proto bylo využito řešení pomocí softwarových hodin z interního čítače/časovače řídicího obvodu. Ten byl nastaven tak, aby každou vteřinu vyvolal interní přerušení, ve kterém se inkrementují příslušné proměnné. Komplikace provázely i oživení displeje. Po zkoumání možných příčin jeho nefunkčnosti se ukázalo, že obvod step up konvertoru negeneruje nastavené napětí, navzdory sestavení obvodu podle doporučeného zapojení ve výrobních listech. Napětí bylo nahrazeno z externího zdroje, který ovšem problém nevyřešil. Přezkoumání datových vodičů osciloskopem potvrdilo správné nastavení rychlosti hodinového signálu i odesílání příslušných dat z řídicího obvodu. U zvukového výstupu došlo k proražení tranzistoru, který má spínat obvod. Elektricky byl obvod ověřen, a proto došlo zřejmě nevhodnou manipulací k proražení vlivem elektrostatického pole, na které jsou tyto typy tranzistorů náchylné.

I přesto se podařilo zařízení dostat do stavu, kdy uživatel pomocí tlačítka dá povel řídicímu obvodu, který na základě aktuálního stavu proměnných předá informaci o čase blikáním *LED* diody. Ověření výdrže baterie by bylo velice zkreslené, protože výdrž nejvíce závisí na displeji.

8 ZÁVĚR

Práce obsahuje popis vlastního návrhu digitálních hodinek. Její součástí byla výroba demonstračního přípravku, který by měl plnit na základě řídicího algoritmu elementární funkce digitálních hodinek.

Výroba tohoto zařízení se však ukázala náročná a vlivem co největší možné minimalizace došlo k několika pochybením a zařízení neplnilo všechny základní funkce hodinek. Výroba vyžadovala po konstrukční stránce specifické požadavky, které byly zohledněny, ale právě jejich vlivem došlo k nestandardním řešením. Ty mají za příčinu některá pochybení. Jejich oprava vyžaduje několik konstrukčních změn, které jsou vzhledem k minimalizaci rozměrů složité. Nejvhodnějším řešením je sestavit nové ověřené zapojení obvodů a následné vyrobení nové *DPS*. Například časový obvod by měl být doplněn o externí krystal nebo zvolen zcela jiný obvod, který tento krystal nevyžaduje. Tím může být například obvod reálného času DS3231M, který má interní rezonátor. Pro lepší konstrukční předpoklady je vhodné zvolit i jiný displej, například ve formě modulu s vlastní deskou, a vyhnout se tak problémům se zapojením displeje.

Vyvinutý software nebylo možné kompletně ověřit kvůli hardwarovým nedostatkům. Proto CD příloha obsahuje ověřený software pro zobrazení času pomocí *LED* diody a také neověřený software pro zobrazení času z *RTC* na displeji.

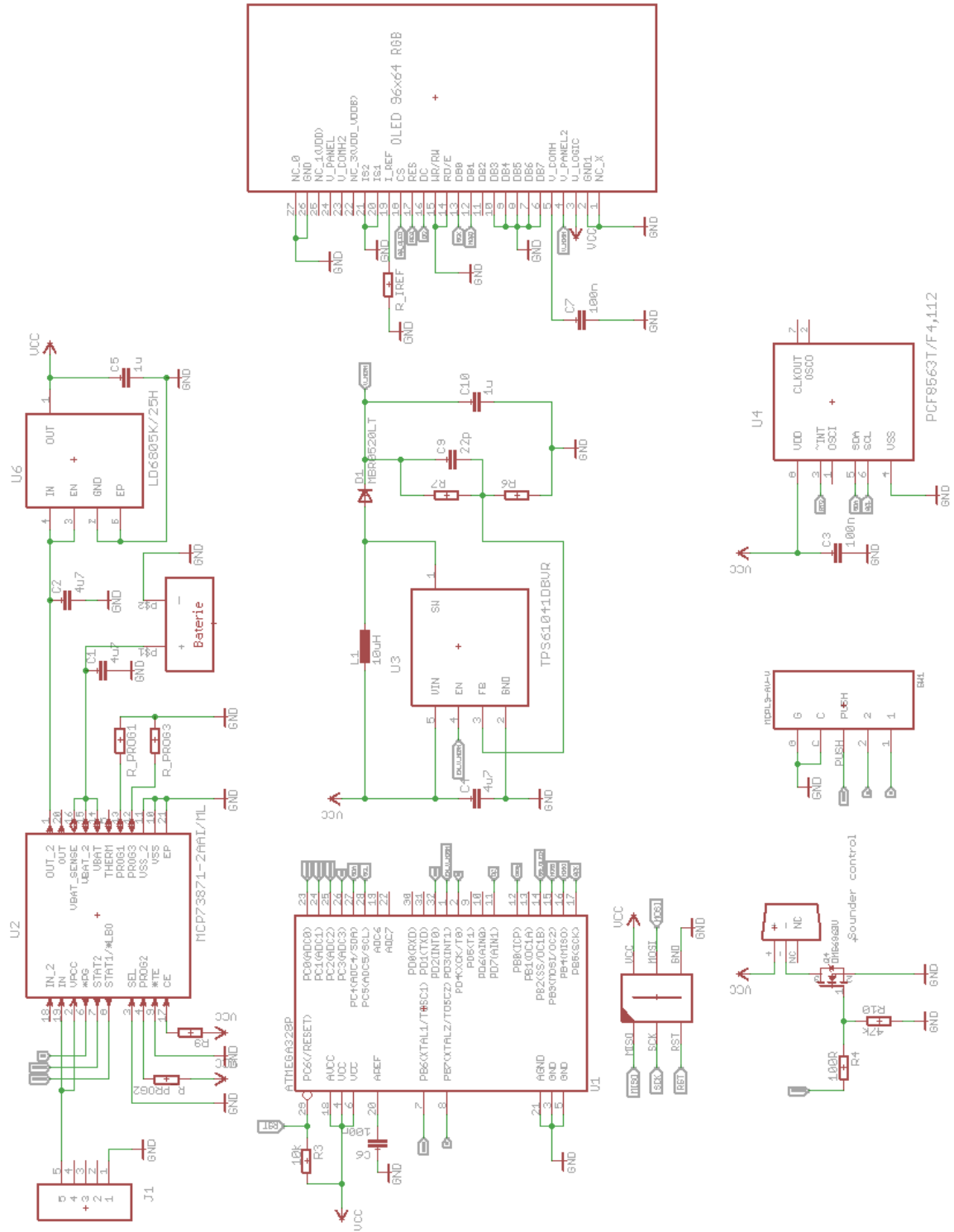
V zadání práce je jedním z úkolů provést měření přesnosti času v závislosti na teplotě. To nebylo možné provést z důvodů uvedených výše. Teoreticky lze ale říci, že *RTC* obvod řízený krystalem bude ovlivněn, ale změna bude malá a složitě měřitelná. Výrobci těchto obvodů udávají relativně velkou přesnost v různých teplotních podmínkách, ve kterých by se toto zařízení nemělo vyskytnout.

Dosažené výsledky a závěr ovlivnil fakt, že práce byla autorovou první zkušeností s návrhem a konstrukcí elektronického zařízení. Navzdory nedostatkům byla práce obrovským přínosem cenných zkušeností.

Literatura

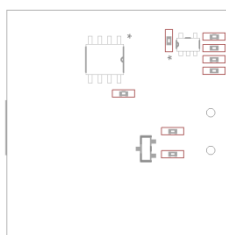
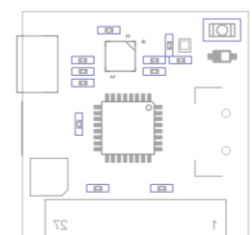
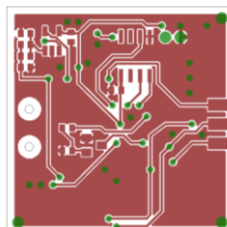
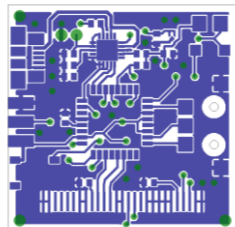
- [1] **Texas Instruments.** EZ430-Chronos. *processors.wiki.ti.com*. [Online] 11. 3 2014. [Citace: 13. 4 2014.] <http://processors.wiki.ti.com/index.php/EZ430-Chronos>.
- [2] **Ludlaw, David.** Pebble smart watch review - hands on. *www.expertreviews.co.uk*. [Online] 9. 1 2013. [Citace: 13. 4 2014.] <http://www.expertreviews.co.uk/general/1297036/pebble-smart-watch-review-hands-on>.
- [3] **Pebble Technology.** GroupTrio04[png]. *Pebble press pack 1[online]*. [Online] <http://ksimings.s3.amazonaws.com/PebblePressPack1.zip>.
- [4] **STMicroelectronics.** STM32F2 Series. *www.st.com*. [Online] [Citace: 13. 4 2014.] <http://www.st.com/web/en/catalog/mmc/FM141/SC1169/SS1575#>.
- [5] **Kemble, Zak.** DIY Digital Wristwatch. *blog.zakkemble.co.uk*. [Online] 25. 9 2013. [Citace: 12. 1 2014.] <http://blog.zakkemble.co.uk/diy-digital-wristwatch/>.
- [6] © 2009 **Atmel Corporation.** 8-bit Microcontroller with 4/8/16/32K Bytes In-System Programmable Flash. *www.atmel.com*. [Online] [Citace: 11. 3 2014.] <http://www.atmel.com/images/doc8161.pdf>. Rev. 8161D–AVR–10/09.
- [7] © **NXP.** PCF8563 Real-time clock/calendar. <http://www.nxp.com/>. [Online] 3. 4 2012. [Citace: 11. 3 2014.] <http://www.nxp.com/>.
- [8] **Slinták, Vlastimil.** DC / DC měniče. *uart.cz*. [Online] 16. 2 2013. [Citace: 15. 3 2014.] <http://uart.cz/952/dc-dc-menice/>.
- [9] **Microchip Technology Inc.** MCP73871. *www.microchip.com*. [Online] 20. 8 2013. [Citace: 12. 1 2014.] [2009/downloads/en/DeviceDoc/20002090C.pdf](http://www.microchip.com/download/en/DeviceDoc/20002090C.pdf).
- [10] **IBT Power.** LITHIUM ION POLYMER TECHNICAL DATA. *www.ibt-power.com*. [Online] 2001-2014 IBT Power Limited. [Citace: 4. 18 2014.] http://www.ibt-power.com/Battery_packs/Li_Polymer/Lithium_polymer_tech.html.
- [11] **Solomon SYSTECH.** SSD1332. *www.crystalfontz.com*. [Online] 3 2006. [Citace: 1. 15 2014.] http://www.crystalfontz.com/controllers/SSD1332_2.3.pdf.

Příloha A – Celkové schéma vytvořené v prostředí EAGLE

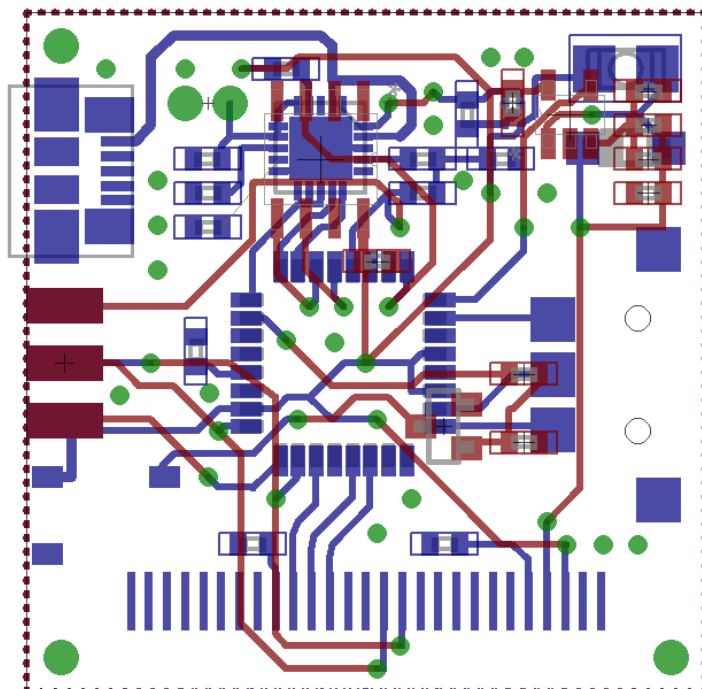


Příloha B – Návrh DPS v prostředí EAGLE

Zobrazení v poměru 1:1



Zobrazení v poměru 3:1



Příloha C – Seznam použitých součástek

Obvod	~	Pozice	Hodnota
Atmega328p-AU	U1	R3	10k
		C6	100n
MCP73871-2AAI/ML	U2	R9	10k
		R_PROG1	15k
		R_PROG2	10k
		R_PROG3	100k
		C1	4u7
		C2	4u7
TPS61041-DBVR	U3	R6	1,8M
		R7	200k
		C4	4u7
		C9	22p
		C10	1u
		L1	10μH
		D1	MBR0520LT
PCF8563T	U3	C3	100n
OLED	-	R_IREF	910k
		C7	100n
LD6805K/25H	U6	C5	1u
ZVUK	-	R4	100R
		R10	47k
		Q4	N-MOSFET
Magnetický měnič	U\$4	ABMT-801-RC	
microUSB	J1	MOLEX-47356-0001	
3-polohové tlačítko	SW1	MCPL3-AC-V	
Baterie	U\$2	Li-Pol, 150mAh	
Konektor programátoru	U\$3	3x2 ISP	

Příloha D – CD

- Bakalářská práce v PDF
- Návrh DPS v EAGLE 6.5.0
- Zdrojový kód v jazyce C