

# **Mikropočítač**

**Dokumentace k maturitní práci**

**Jméno: Jiří Hajda**

**Třída: P4.A**

**Rok: 2001/2002**

Toto zařízení bylo vyrobeno v rámci maturitní práce na Střední Průmyslové Škole Elektrotechnické v Rožnově pod Radhoštěm.

## Obsah

<b>1 Návrh</b> .....	2
1.1 Důvody k zhotovení.....	2
1.2 Základní návrh.....	2
<b>2 Realizace jednotlivých bloků</b> .....	3
2.1 Realizace klávesnice.....	3
2.2 Realizace displeje.....	4
2.3 Realizace řídicího bloku.....	5
2.3.1 Volba procesoru.....	5
2.4 Volba paměťového systému.....	7
2.5 Realizace zvukového generátoru.....	8
2.6 Realizace hodin reálného času.....	9
2.7 Realizace paralelního portu.....	10
2.7.1 Tiskárna Barevný Plotter MZ-1P16.....	11
2.8 Napájecí blok.....	11
2.9 Programovatelný čítač 8253.....	11
<b>3. Oživení</b> .....	12
<b>4. Deska plošného spoje</b> .....	12
4.1 Rozpiska součástek.....	15
<b>5. Krabička</b> .....	16
<b>6 Programová výbava</b> .....	16
6.1 Popis portů.....	17
<b>7 Závěr, zhodnocení a doporučení</b> .....	17
<b>8 Literatura</b> .....	17

# 1 Návrh

## 1.1 Důvody k zhotovení

První počítač, s kterým jsem se setkal byl osmibitový mikropočítač SHARP MZ-800. Tento osmibit byl osazen mikroprocesorem Z80-A na frekvenci 3,5MHz a měl jen základní paměť 64kB. Na svou dobu docela dobrý stroj, ale oproti klasickým PC neměl ani disketovou mechaniku a ani pevný disk. Proto jsem pro něj chtěl vyrobit nějaký přídatný hardware. K tomu jsem se však již nedostal. Zhruba v polovině třetího ročníku se mi dostal do ruky LCD displej 20x4 znaků a malá klávesnice. A protože se mi líbily kapesní počítače a diáře, rozhodl jsem se vyrobit si vlastní kapesní počítač. Hlavní nevýhodou komerčních minipočítačů je právě jejich omezenost (těžko si něco naprogramujete v diáři) a jejich cena. Neměl jsem mnoho zkušeností s číslicovou elektronikou, ale měl jsem bohaté zkušenosti s programováním procesoru Z80A a hardwaru na zmíněném SHARPOvi. V té době jsem věděl, že si budeme muset zvolit vypracování nějakého výrobku k maturitě. Využil jsem tedy této příležitosti, abych se donutil k rychlejšímu vytvoření svého malého počítače.

## 1.2 Základní návrh

Cílem tedy bylo v rámci možností vytvořit mikropočítač s ohledem na následující požadavky. Tyto požadavky se mi povedlo z větší části splnit.

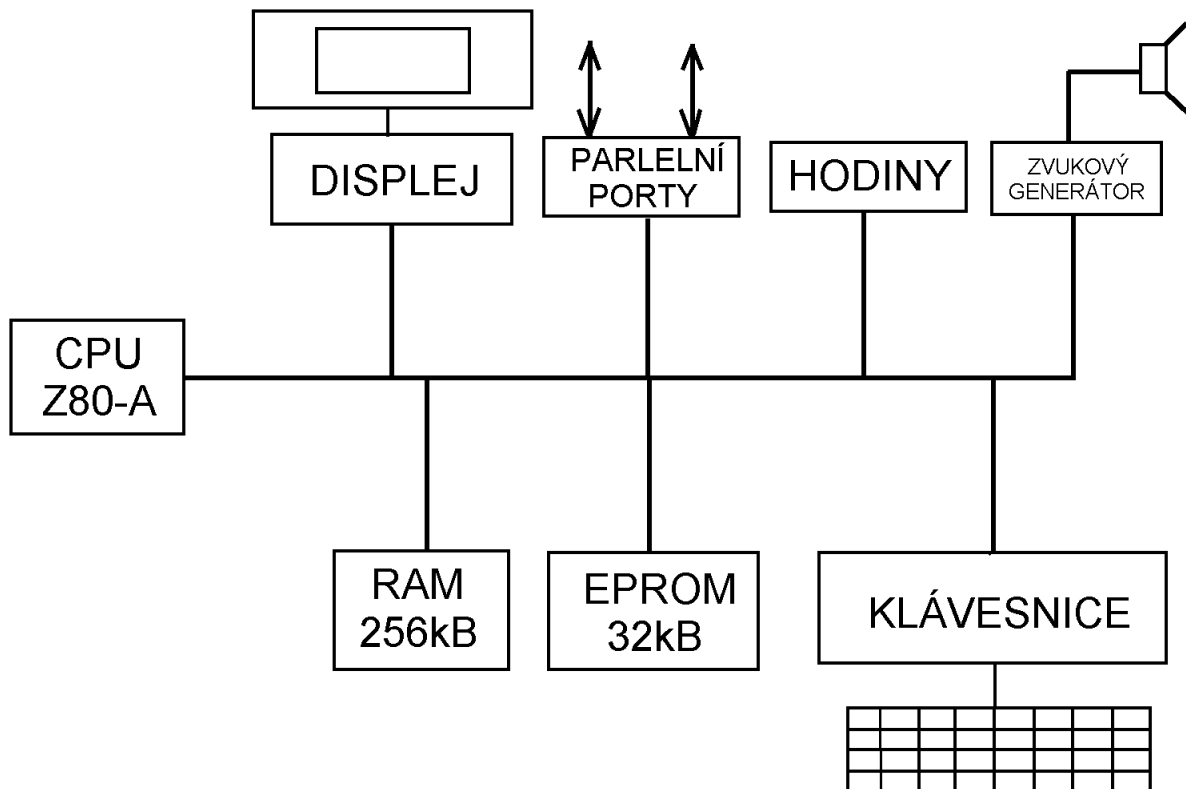
### Hlavní parametry ideálního mikropočítače:

- Velké množství periférií
- Velká kapacita operační paměti
- Velká kapacita dlouhodobé paměti
- Jednoduché zapojení
- Malé rozměry
- Nízká cena
- Malá spotřeba

### Naplánovaná vstupně-výstupní zařízení

- Paralelní porty pro komunikaci s okolím
- LCD displej 20x4 znaků
- Maticová klávesnice
- Zvukový obvod
- Hodiny reálného času

Celý výrobek byl navrhnout v grafickém navrhovacím programu Eagle. Zpočátku ve verzi 3.55 a později ve verzi 4.01(což přineslo jisté komplikace).

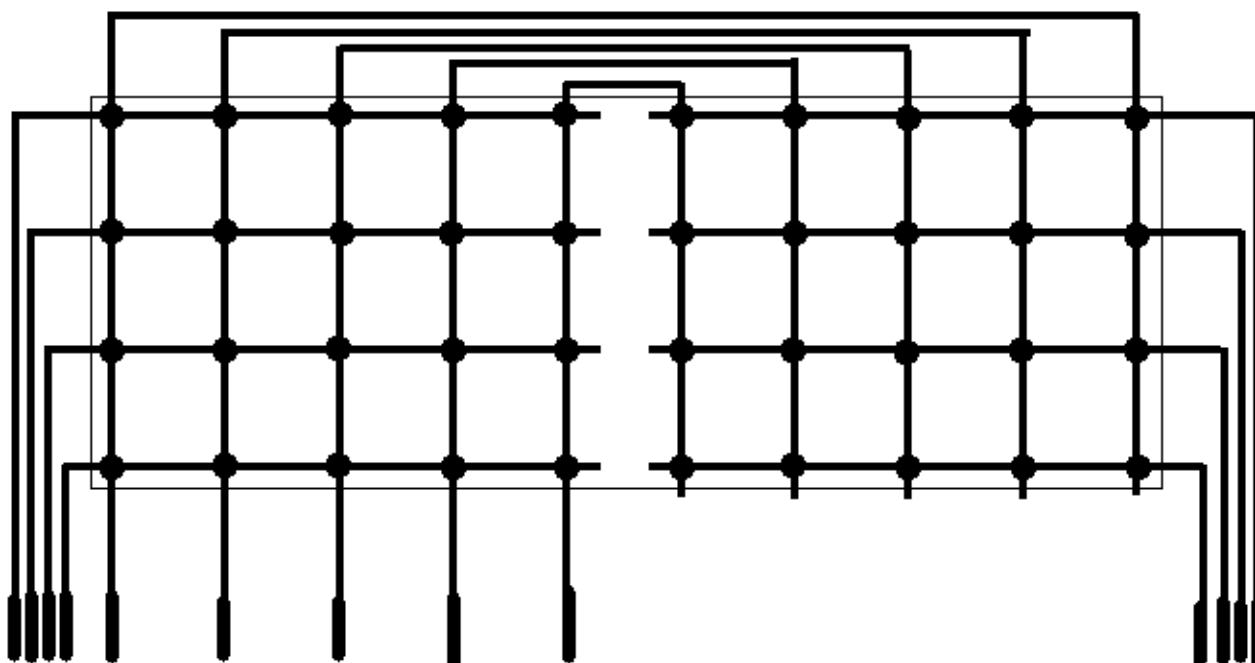


Obr. 1 – Blokové schéma mikropočítače

## 2 Realizace jednotlivých bloků

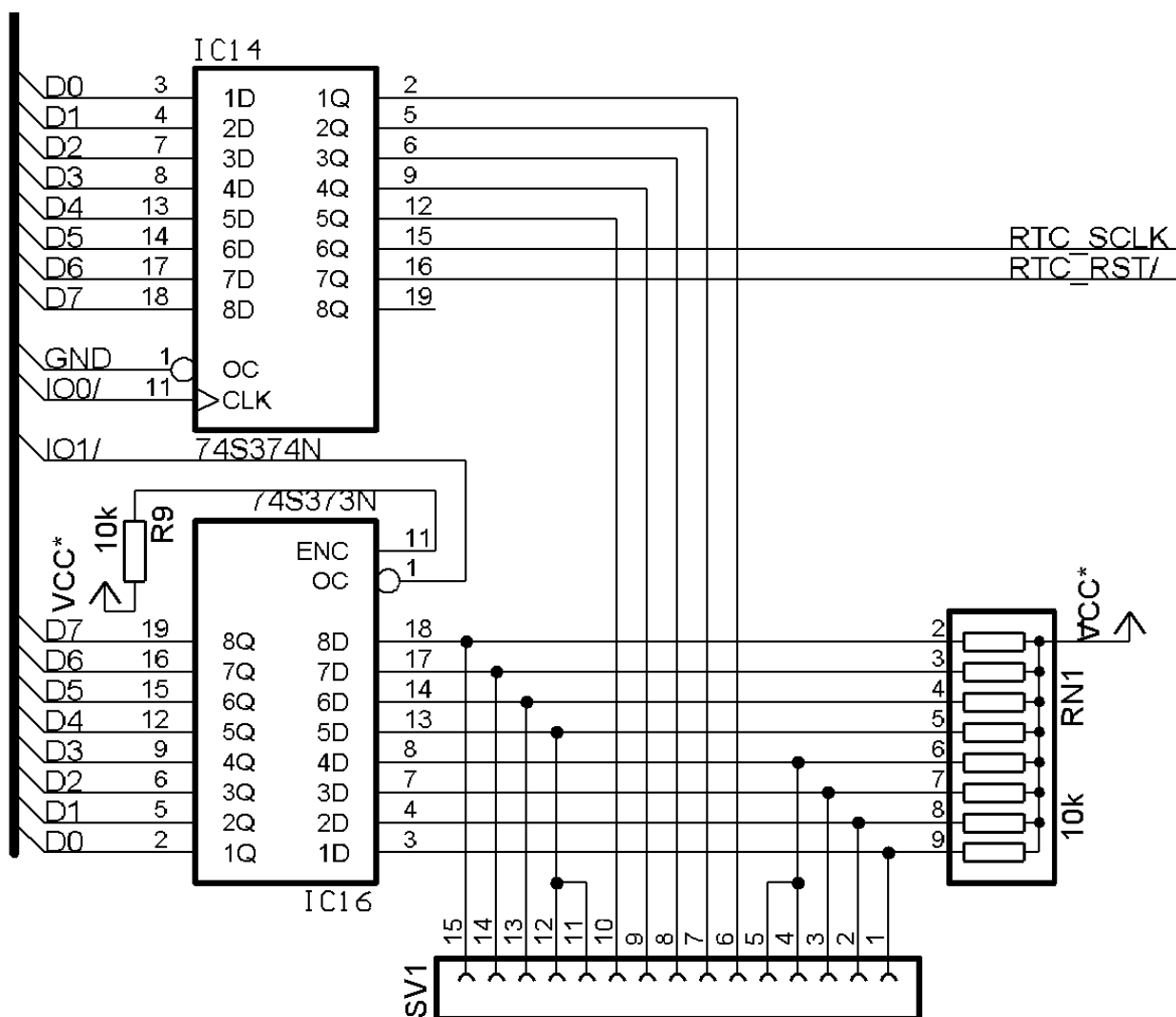
### 2.1 Realizace klávesnice

Náhodou jsem získal malou klávesnici o rozměrech 213mm x 68mm. Pro malý mikropočítač je jako stvořená. Na menší by se špatně psalo a větší by způsobila zvětšení celého výrobku. Jde o kontaktní klávesnici zapojenou podle obrázku uvedeného níže.



Obr. 2 – Vnitřní zapojení klávesnice

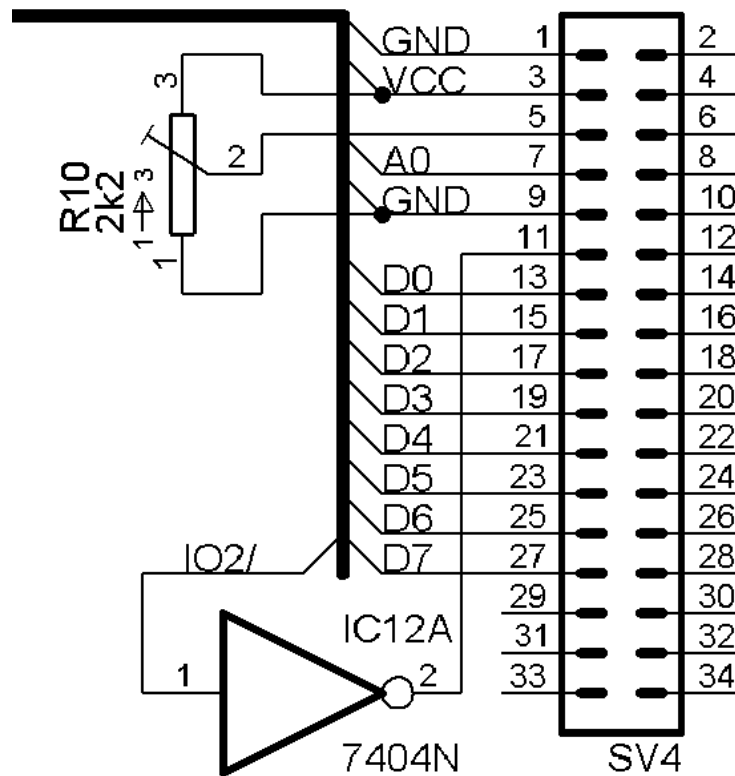
Pro jednoduchou realizaci jsem uvažoval 8 vodičů pro čtení „řádku“ a 5 pro výběr „sloupce“. Mezerník jsem připojil do odpovídajícího sloupce. Použil jsem registr 74374 pro výběr „sloupce“ a latch obvod 74373 pro čtení „řádku“. Zapojení pracuje tak, že pokud není stisknuta klávesa je přes odporovou řadu RN1 na vstupech obvodu IC16 logická 1. Stiskem klávesy se uzemňují vstupy IC16. Do registru se tedy zapisuje hodnota s jedním nulovým bitem.



Obr. 3 – Schéma zapojení bloku klávesnice

## 2.2 Realizace displej

Použil jsem inteligentní LCD displej 20x4 znaků bez podsvícení. Displej využívá klasický řadič HD44780, takže nebyl problém s jeho ovládáním. Malý problém byl s jeho zapojením. V doporučené zapojení pro Z80A byl výběrový signál Enable zapojen přes monostabilní obvod z důvodů příliš rychlého časování procesoru. Když jsem zapojil tento obvod přímo, tak opravdu docházelo k chybám. Protože jsem s tímto počítal dopředu přidal jsem do schématu taktovací jumper pro výběr frekvence 3,5MHz nebo poloviční 1,75MHz. Další problém byl s připojením na sběrnici. U většiny obvodů připojitelných na sběrnici je vyveden negovaný vstup Chip Select. Signál displeje Enable však zřejmě slouží jen jako hodiny. V dokumentaci jsem se nikde nedočel, že je datová sběrnice třístavová. Proto jsem uzemnil signál R/W, aby displej pouze přijímal, čímž jsem se ochudil o možnost testování zaneprázdněnosti displeje pomocí signálu Busy. Navíc jsem musel použít invertor (jediný ve schématu). Displej má sice jen 14 vývodů, ale pro zjednodušení jsem použil dvouřadý konektor samořezný. Nejbližší vyšší vyráběná velikost je 2x17(34). Kontrast displeje nelze regulovat z vnějšku, ale pouze odporovým trimrem. Signál RS (instrukce nebo data) je připojen na nejnižší bit adresové sběrnice. Mohu tedy data adresovat jiným portem než instrukce.



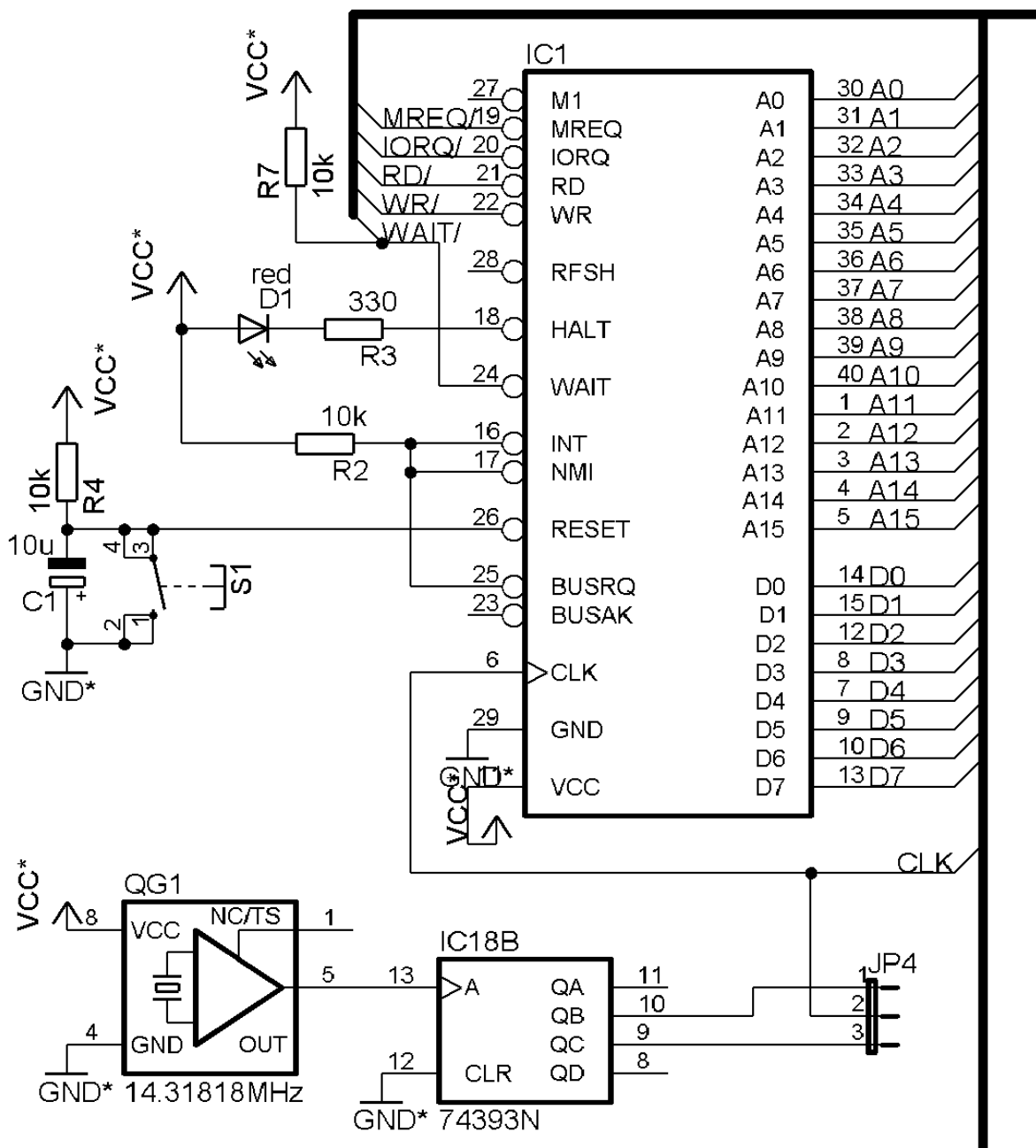
Obr. 4 – Zapojení displeje

## 2.3 Realizace řídicího bloku

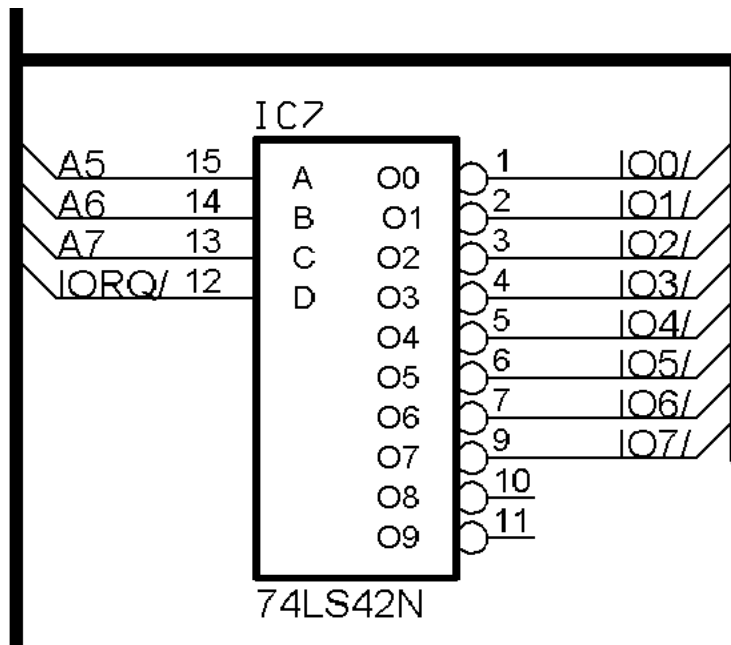
### 2.3.1 Volba procesoru

Bylo potřeba zvolit vhodný procesor. Nabízelo se několik možností. Mohl jsem použít mikrokontrolér 8051, ale ten se ukázal nevhodný. Jednou z nevýhod je rozdělení paměti na paměť programu a paměť dat. Chtěl jsem mít volnost ve volbě velkého programu nebo mnoha dat. Navíc jsem počítal s programováním v terénu a tedy spouštěním „dat“. Další nevýhodou je, že 8051 není přirozeně orientována na využití se sběrnici. Dále jsem mohl volit mezi procesory 8088 a Z80. Procesor 8088 by byl vhodný, ale jednak jsem ho neměl k dispozici a jednak jsem ho neznal tak dobře jako Z80. Výhodou Z80 je především v jednoduchosti zapojení a jednoduchosti instrukční sady. Frekvence 3,5MHz je pro mikropočítač s výstupem na textový displej dostačující.

Je tedy použit procesor firmy Zilog 80A. Max frekvence je 4MHz. Pro generování hodinového signálu jsem použil krystal 14.31818MHz. Tato frekvence je dělena čtyřmi obvodem 74393. Frekvence je tedy 3,57954MHz a lze ji snížit na polovinu (1,78977MHz) nastavením propojky. Ke vstupu RESET/ je zapojen resetovací obvod. K výstupu HALT/ je zapojena kontrolní červená dioda signalizující přechod CPU do stavu HALT, kdy procesor vykonává instrukci NOP do příchodu přerušení. (v mém případě do resetování) Vstupy NMI/, INT/, BUSRQ/ jsou trvale připojeny na logickou úroveň 1, neboť nejsou z důvodů jednoduchosti využity. Vstup WAIT/ je využíván zvukovým generátorem. Protože má zvukový generátor tento výstup zapojen s otevřeným kolektorem je nutno zapojit přídatný rezistor na VCC. Dále je pak vyvedena 16-bitová adresová a 8-bitová datová sběrnice. Sběrnice nebylo nutno posilovat, jak je to prováděno v jiných schématech. Řídící sběrnice obsahuje signály RD/, WR/, MREQ/, WAIT/ a CLK. Procesor umí adresovat 256 vstupních a 256 výstupních zařízení. V celém mikropočítači nevyužívám více než 8 zařízení, proto pro výběr vstupně-výstupních zařízení stačí dekodér 7442 s negovanými výstupy. Ten je zapojen tak, že vstup D je chytré využít jako výběrový signál obvodu.



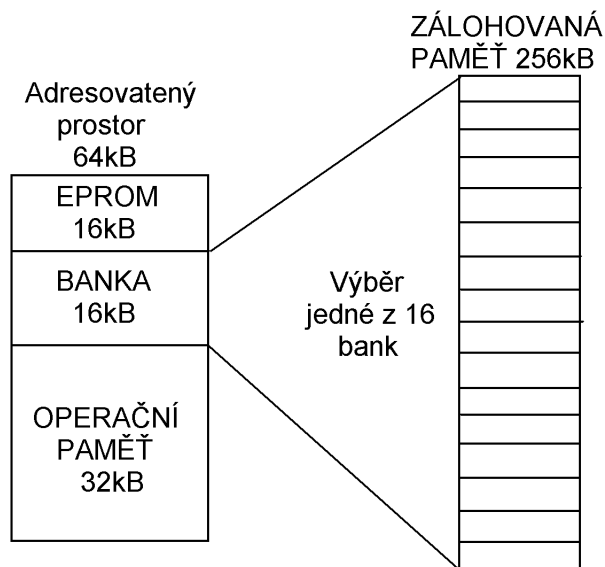
Obr. 5 – Zapojení řídicího bloku



Obr. 6 – Zapojení výběru IO zařízení

## 2.4 Volba paměťového systému

Procesor Z80 je schopen adresovat až 64kB. Nejdříve bylo potřeba zvolit ROM paměť z níž se bude zavádět operační systém. Prostudoval jsem katalogy firem prodávajících součástky v České republice a zakoupil jsem paměť s nejlepším poměrem cena-velikost. A to paměť typu FLASH 28F020 256kB. Chtěl jsem tuto paměť použít i k ukládání uživatelských dat, ale jak jsem pak dodatečně zjistil, pro programování je potřeba 12V, což jsem si nemohl dovolit. Navíc se mi tuto paměť nepovedlo „rozchodit“. Proto jsem sáhl po klasickém způsobu, a to paměť EPROM. Shromáždil jsem několik typů z nichž nejvhodnější byl typ 27256 32kB. Tuto paměť jsem přiřadil do celého paměťového prostoru na prvních 16kB. Druhou polovinu EPROM je pak možno přepínat přes paměťový registr. Dále jsem podle katalogů zakoupil dva obvody SRAM 128kB. Těchto 256kB jsem se rozhodl zálohovat baterií. Tato paměť je přístupná na druhé čtvrtině (druhých 16kB) celkového adresového prostoru. Přes paměťový registr se vybírá vždy jeden 16kB blok z 16 možných. Zbývá horní polovina adresového prostoru Z80 je použita jako aplikační paměť. Ta je pevně mapována na začátek paměti 256kB .



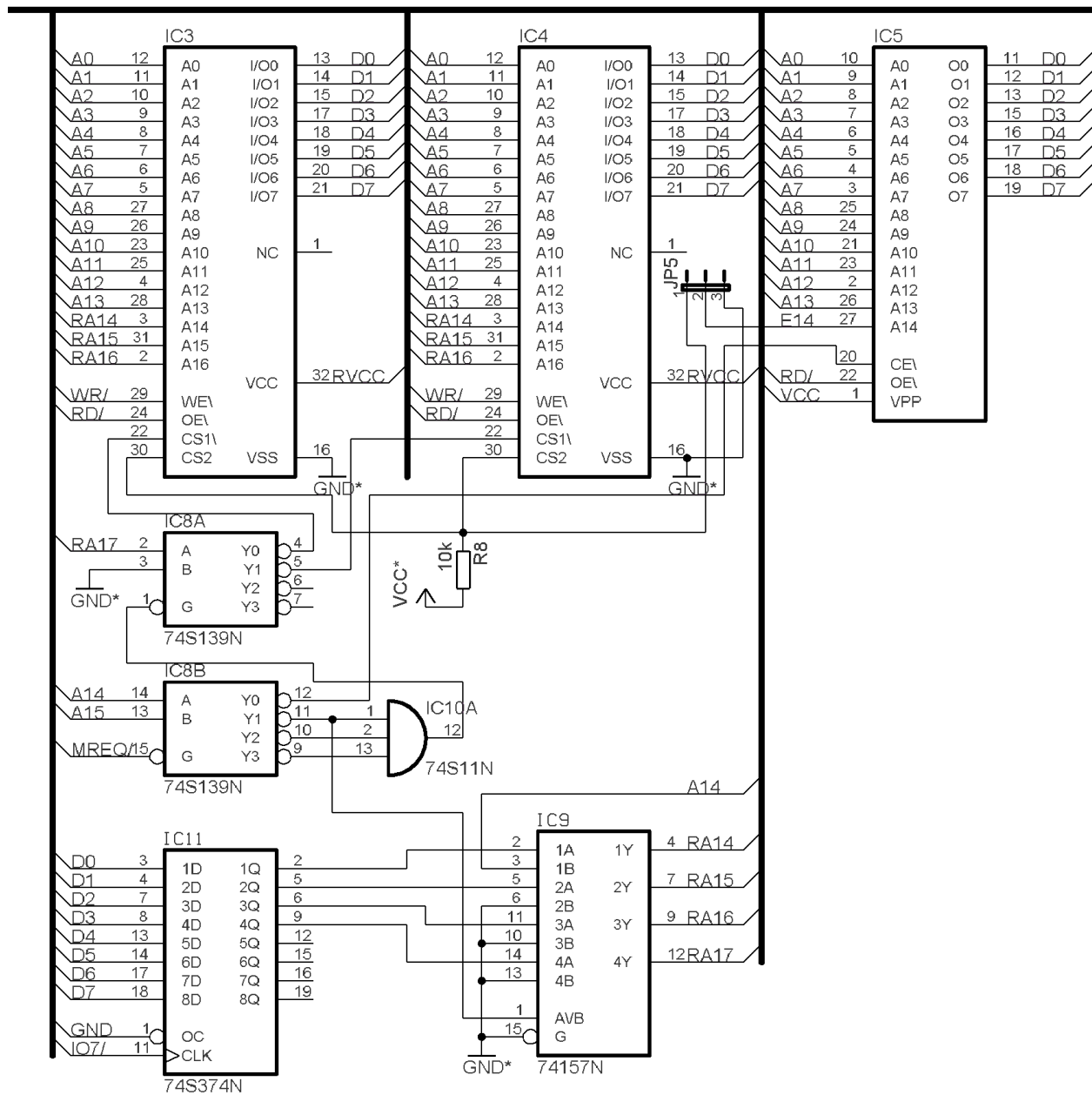
Obr. 8 - Zobrazení mapování paměti

Jsou tedy použity dva obvody statické paměti 628128 128kB a paměť EPROM 32kB. Dekodérem 74139 (část B) je vybírána vždy ¼ celé paměti (16kB z 64kB). 1. čtvrtina je přivedena na výběrový vstup EPROM.



Zbýlé tři třetiny jsou sloučeny a přivedeny na vstup druhé poloviny dekodéru 74139 (část A), kde v závislosti na signálu RA17 je rozhodnuto, zda jde o první nebo druhý paměťový čip. Výstup Y1 obvodu IC8B je přiveden na vstup multiplexeru. Zde je rozhodováno, jestli jde o přístup do operační paměti mapované na počátek nebo o přístup do banky paměti. Banka paměti se nastavuje přes paměťový registr 74374.

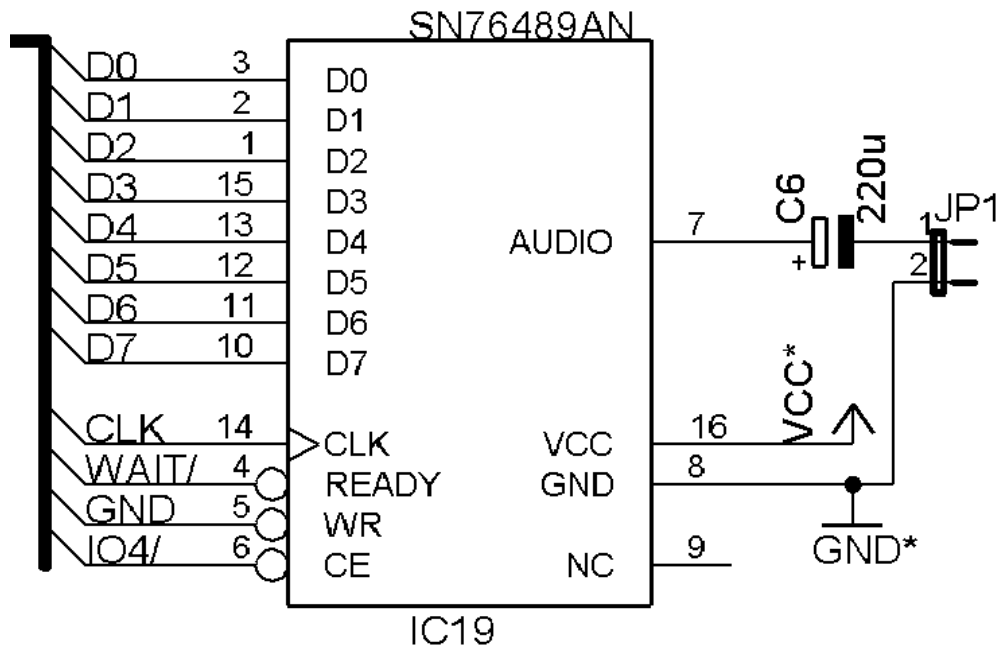
Protože je vstup A14 paměti EPROM 27256 shodný se vstupem WE/ EPROM 27128, zabudoval jsem do počítače přepínač vstupu A14 mezi logickou úrovní 1 a 0. Výhoda tohoto zapojení je také v tom, že mohou jednu paměť EPROM 27256 použít dvakrát mezi každým smazáním. Jednou nahrát program do dolní a podruhé do horní poloviny.



Obr. 7 – Schéma zapojení paměťového bloku

## 2.5 Realizace zvukového generátoru

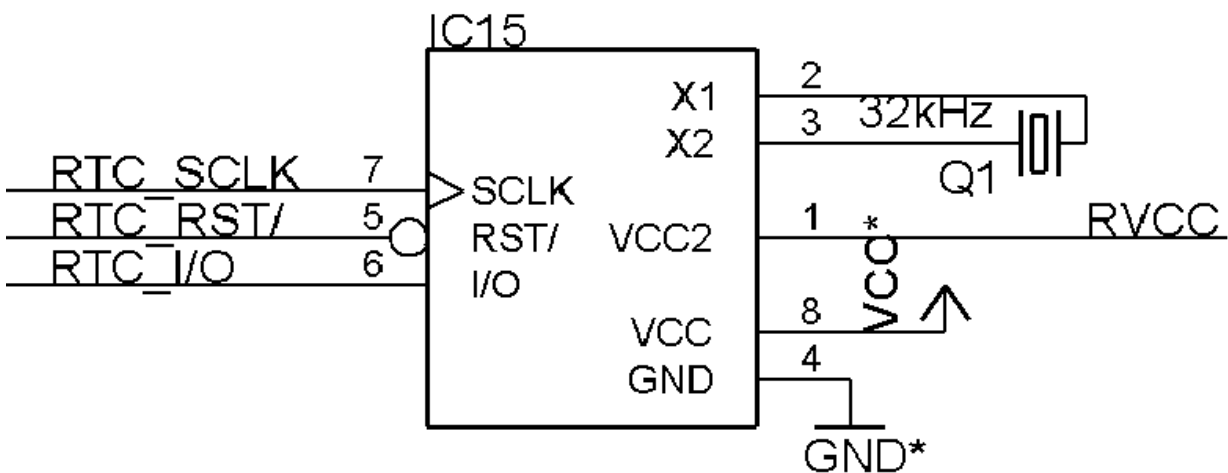
Jako zvukový generátor jsem použil obvod SN76489AN vyrobený firmou Texas Instruments. Je shodný se zvukovým generátorem ze jmenovaného osmibitového počítače SHARP. Tento generátor obsahuje tři zvukové generátory a jeden šumový generátor. U každého generátoru lze nastavit hlasitost. Na obvodu je také vyveden zvukový vstup, který jsem nepoužil. Obvod je přizpůsoben pro komunikaci po sběrnici, proto s jeho připojením nebyly velké potíže. Audio výstup je v SHARPU zesílen. Po otestování jsem se rozhodl zesilovač nezapojovat. Pro použitý reproduktor je výstup dosti silný. Protože je potřeba pro posílání dat do obvodu potřeba většího času, je synchronizován obvod s procesorem prostřednictvím signálu WAIT/.



Obr. 9 – Zapojení zvukového generátoru

## 2.6 Realizace hodin reálného času

Obvod DS1203 slouží jako hodiny reálného času. Tento obvod jsem zvolil s ohlednutím na cenu. Pro mikropočítač by se hodil spíše obvod DS12887, který umí komunikovat po sběrnici a má více funkcí. Obvod DS1203 v sobě sdružuje RTC (Hodiny reálného času) a 31 bajtů zálohované paměti. Je vyroben v pouzdře DIL8 a komunikuje po jednoduché 3-vodičové sériové lince. Tato jednoduchost se však hodí spíše pro zapojení s mikrokontroléry 8051, kde tak ušetříme několik vodičů. Mně však tento obvod přinesl mnoho potíží. Během přenosu musí signál RST/ zůstat v logické 1, proto ho nemohu napojit na výběrový signál CS/ (IOX/). Signály RST/ a SCLK jsem tedy napojil na volné bity u registru klávesnice. Signál I/O jsem napojil na bit PC7 obvodu 8255.



Obr. 10 – Zapojení hodin reálného času

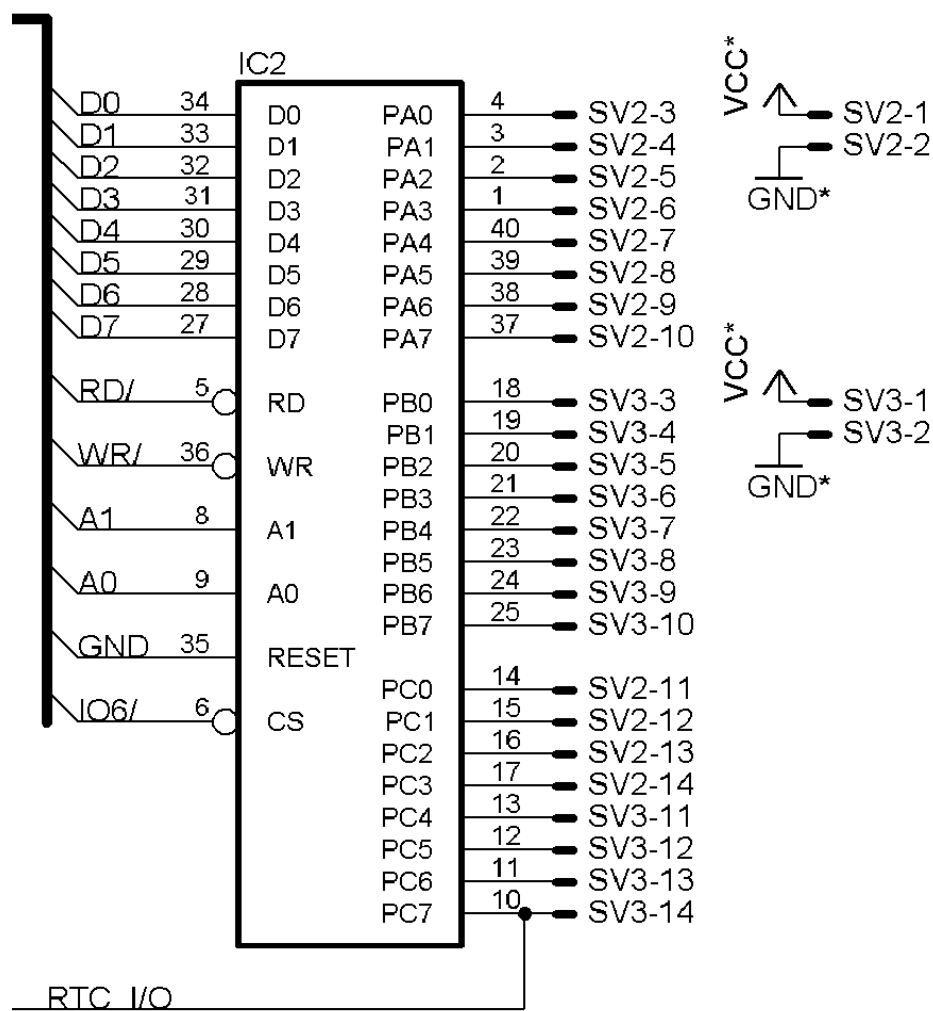
## 2.7 Realizace paralelního portu

Tento port měl sloužit k propojení s IBM PC a k připojení tiskárny. Měl jsem možnost zvolit z několika variant realizace:

1. Obvod 8255 – tento obvod obsahuje 3 osmibitové brány PA, PB a PC. Umí pracovat v několika módech: Mód 0: každá brána vstup nebo výstup. Mód 1: brány PA a PB vstup nebo výstup, brána PC je rozdělen na polovinu a slouží jako řídicí signály pro PA a PB. Mód 2: v tomto módu může pracovat jen brána PA přičemž PC slouží jako řízení přenosu. Jednoduché zapojení.
2. Obvod Z80-PIO – tento obvod obsahuje dvě 8-bitové brány A a B. Obě brány obsahují dva řídicí bity pro korespondenční komunikaci. Může pracovat ve 4 módech: vstup, výstup obousměrně nebo řídicí bity. Má složitější zapojení. Je však přizpůsoben CPU Z80.
3. Obvod 8243 – rozšiřuje 4bity na 4 4-bitové brány. Umožňuje funkce čtení, zápis, funkci AND a OR.
4. Registr 74374 nebo 74574 – lze připojit na sběrnici, nerozšiřuje bity a může pracovat jen v jednom směru. Výhodou je malý počet vývodů a tedy i malé pouzdro. Výhodou malá spotřeba.(CMOS technologie)

Z těchto možností jsem zvolil obvod 8255. Špatně jsem promyslel realizaci paralelního portu a po zapojení se ukázaly dva problémy: obvod měl příliš velkou spotřebu a 2. port je u tohoto mikropočítače naprosto zbytečný. Avšak ani ostatní možnosti nejsou nejideálnější.

Obvod 8255 je připojen na společnou sběrnici. Z něj jsou vyvedeny dva vnější porty 1 a 2. Port jedna má přiřazenu bránu PA a nižší polovinu brány PC. Port 2 má přidělenou bránu PB a vyšší čtyři bity brány PC. Ke každému portu je ještě přidána zem a napájení 5V. Toto uspořádání je výhodné pro obousměrný korespondenční přenos. Taky je vhodné pro zapojení tiskárny.



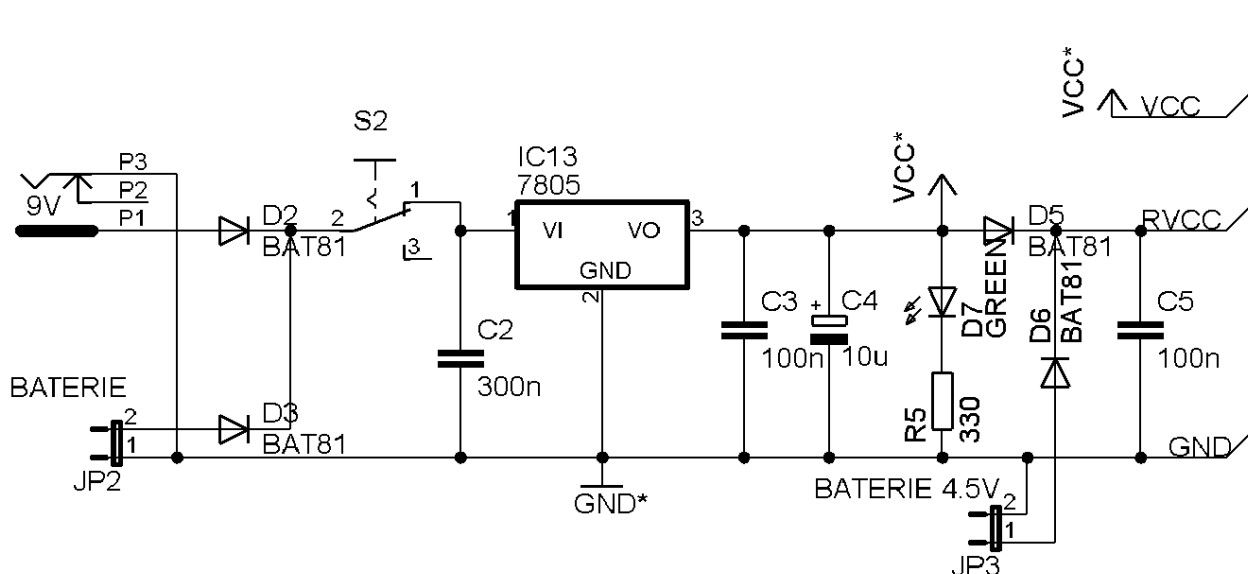
Obr. 11 – Zapojení paralelního portu

### 2.7.1 Tiskárna Barevný Plotter MZ-1P16

K mikropočítači SHARP MZ-800 jsou dodávány dva druhy tiskáren. Barevný mini-plotter MZ-1P16 nebo bodová tiskárna MZ-80P5. Jelikož patřím mezi jednoho ze šťastných vlastníků tohoto barevného plotteru, rozhodl jsem se ho oprášit a použít se svým mikropočítačem. Má hodně malé rozměry a proto ho lze lehce přenášet. Další výhodou je napájení 5V. Má v sobě předdefinovanou tabulku znaků ve formátu ASCII. Proto na něj lze jednoduše posílat znaky. Může tisknout až čtyřmi barvami. Používá však nestandardní šířku papíru. Komunikuje přes paralelní jednosměrné rozhraní. Na konektoru jsou vyvedeno 8 datových vodičů, dva výstupní (signál potvrzení dat a restart) a dva vstupní (zda je tiskárna připojena a zda pracuje).

### 2.8 Napájecí blok

Celý mikropočítač je napájen z vnějšího zdroje 9V. Stabilizátorem je vytvářeno hlavní napájení 5V. V mikropočítači je zapojena 9V baterie pro případ absence vnějšího zdroje. Napájení přes stabilizátor není nejvhodnější řešení, ale asi nejjednodušší po zenerové diodě. Odběr mikropočítače je asi 250mA což znamená, že s baterií o kapacitě 300mAh vydrží asi 1 hodinu. (pravdivost tohoto tvrzení nevyzkoušena v praxi). V mikropočítači je ještě jedna baterie určená k napájení paměti a hodin reálného času. Odběr těchto obvodů je tak malý, že by měla baterie vydržet hodně dlouho. Pro toto napájení jsem zvolil tři 1,5V baterie. Měly by dávat tedy 4,5V, ale při měření jsem naměřil 5V. V době přítomnosti hlavního napájení, je zálohovací napájení posíleno hlavním napájením přes schotkyho diody, které sráží malé napětí. Zelená dioda signalizuje zapnuté napájení. K mikropočítači jsem zakoupil univerzální AC adaptér na max. 500mA.



Obr. 12 – Schéma napájecího bloku

### 2.9 Programovatelný čítač 8253

Už od počátku zapojování jsem chtěl využít výhod přerušovacího systému mikroprocesoru. Procesor má dva vstupy pro žádost o přerušování: maskovatelné INT/ a nemaskovatelné NMI/ (Non-Maskable Interrupt). Jako nejlepší možnost generování přerušování na vstup INT/ jsem chtěl použít obvod 8253. Jde o programovatelný čítač, který v sobě sdružuje 3 16-bitové čítače, které mohou pracovat v několika módech. Hlavní důvody, proč jsem do konečného návrhu obvodu nezahrnul, jsou velké rozměry, velká spotřeba a složitost zapojení (více děr a průchodek). Nabízelo se tedy jiné řešení v podobě stálého generování maskovatelného přerušování odvozeného z frekvence sběrnice. To by pro generování pevné frekvence 318Hz znamenalo přidání jen jednoho čítače 74393. Avšak pro synchronizaci s CPU by bylo potřeba dalších obvodů, neboť CPU potvrzuje přejetí žádosti o přerušování současnou aktivací signálů IORQ/ a M1/. Proto jsem z důvodů jednoduchosti musel výhod přerušovacího systému oželeť.

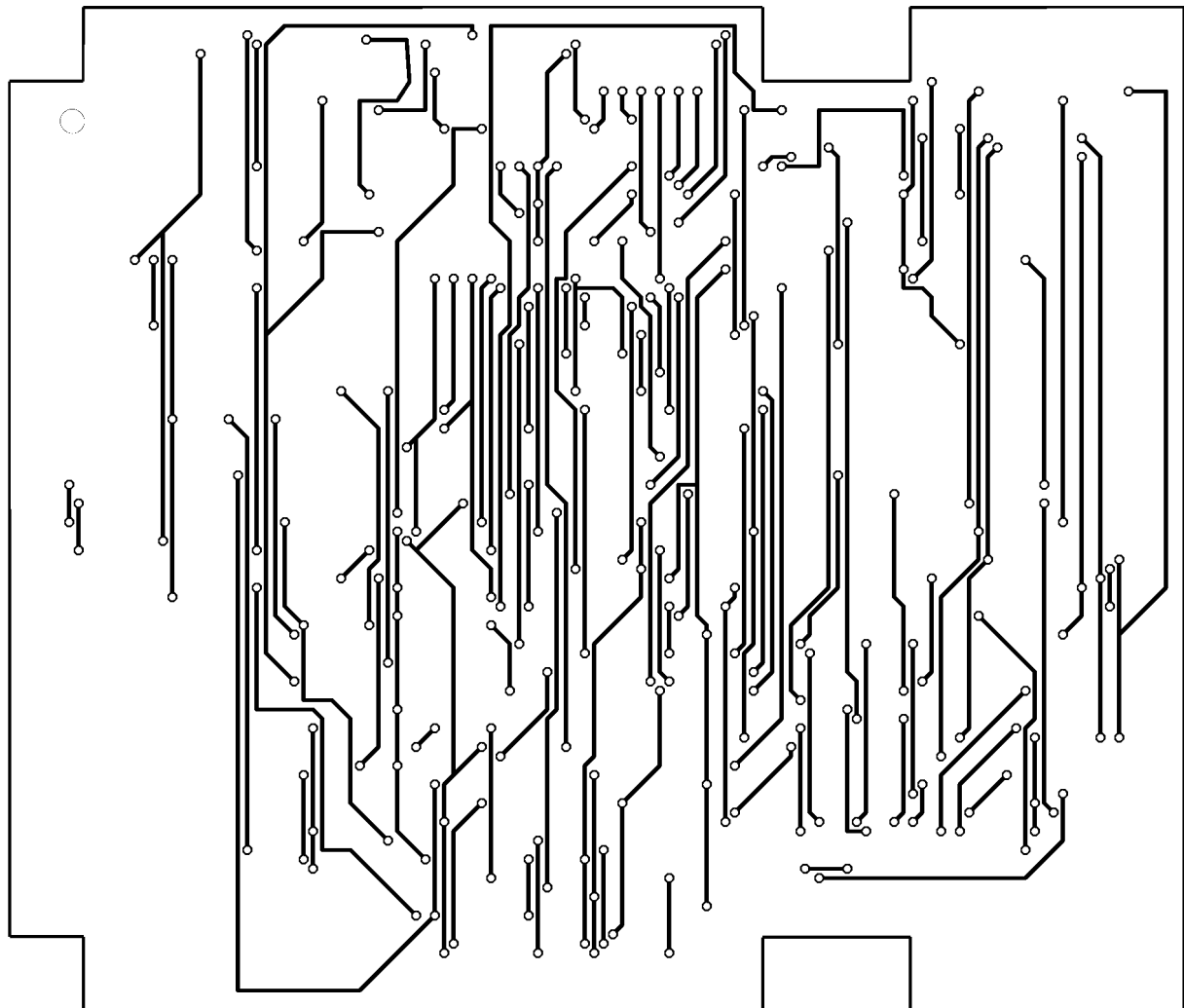
## 3 Oživení

Jednotlivé bloky jsem zapojoval nejprve na nepájivém poli. Pak jsem je vytvářel samostatně jako jednotlivé moduly a u každého jsem přidal konektor připojený na společnou sběrnici prostřednictvím 40-žilového kabelu. Bylo však nevhodné dát tyto moduly do krabičky a navíc se tam nevešly. Celý počítač jsem finálně navrhnul do jednoho schématu a na jednu desku. Pod každý IO jsem zapájel patici, abych mohl testovat správné propojení cest bez ovlivňování integrovanými obvody. Několik cest nebylo spojeno vlivem špatného osvětlení a několik vlivem špatného zapájení průchodek. Po kontrole jsem postupně zapojoval obvody. Nejdříve CPU, paměť a obvody pro výběr paměti. Procesor pracoval, avšak problém byl s pamětí. Jak jsem pak zjistil nebyl průchozí výběrový signál pro EPROM a proto byly v době čtení z EPROM datové vodiče ve vysoko-impedančním stavu a tedy probíhal náhodný program. Po opravě již program probíhal správně. Dále jsem připojil displej a klávesnici. U registru klávesnice jsem narazil na logickou chybu v podobě přehozených bitů D2 a D5. Dále jsem zapojil zvukový obvod a obvod 8255. Doposud jsem vše testoval s přímým napájením 5V spojeným i s zálohovacím napájením. Nyní jsem zapájel napájecí blok. Opět se objevila chyba ze schéma. Jedna nožička spínače nebyla propojena. Nakonec jsem zapojil i obvod hodin. Zde jsem zjistil, že jsem omylem zaměnil hlavní napájení s napájením zálohovacím. Pro zkušební programování jsem používal simulátor EPROM od firmy EasySoft. Nedostal jsem se k výrobě vlastního simulátoru. Nakonec jsem odzkoušený program chtěl přenést do připravené EPROM, ale zde se objevil velký problém. Měl jsem 2 čipy EPROM 27256 a 2 čipy EPROM 27128. 2 čipy EPROM 27256 a jeden 27128 nebyly dobře smazány a zbylý čip 27128 se mi nepovedlo naprogramovat (snad jiné programovací napájení). Proto jsem musel použít čip EPROM 271001 o kapacitě 128kB a vytvořit redukci, protože tato EPROM má jiné pouzdro. Následně jsem zařízení vložil do krabičky, zapojil displej, klávesnici a zašrouboval. Zjistil jsem však, že u klávesnice jsou poškozené kontakty a nepracují všechna tlačítka.

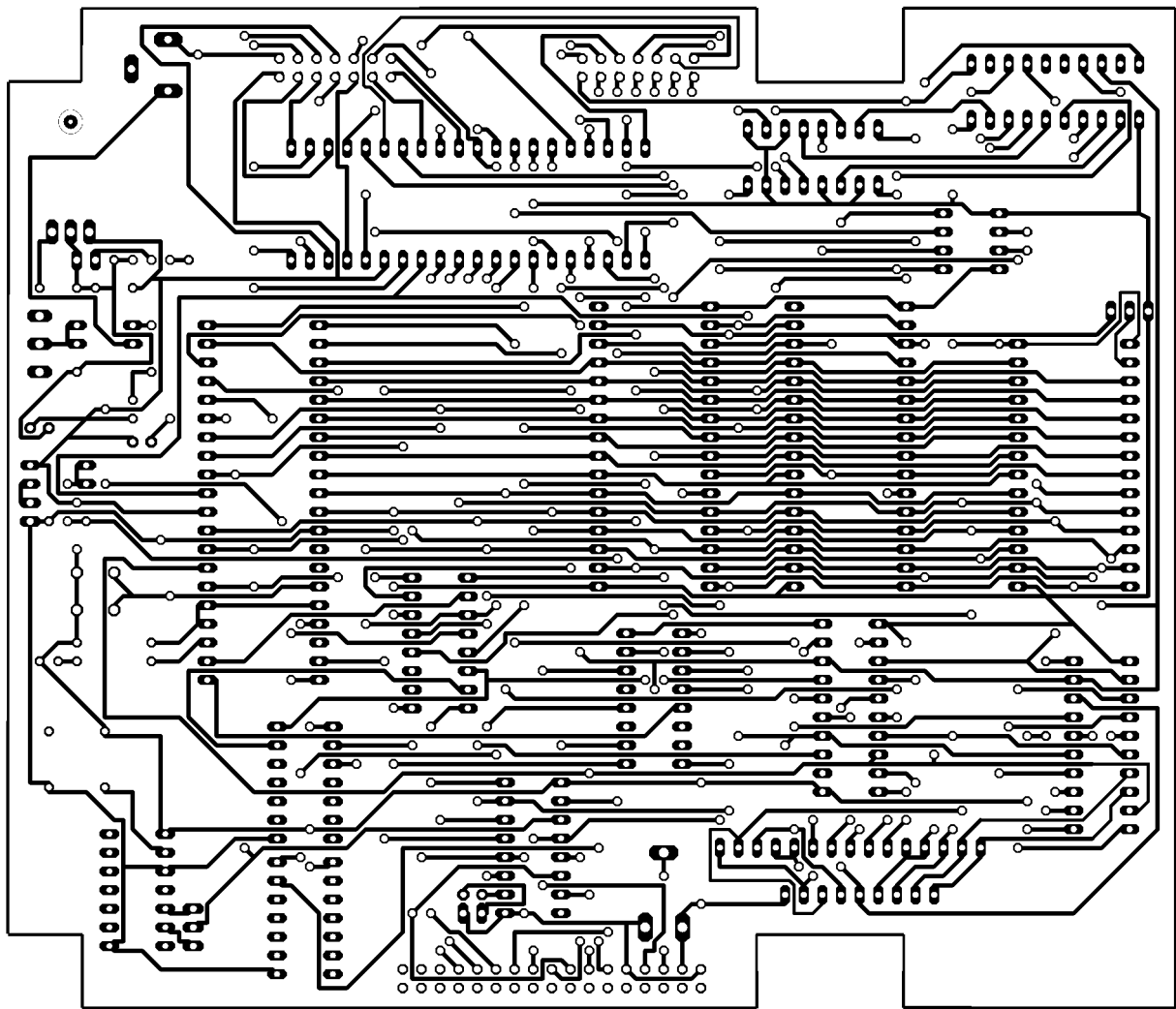
Při zapojování jsem se setkal především s chybami způsobenými nekvalitním zpracováním desky a s několika logickými chybami.

## 4 Deska plošného spoje

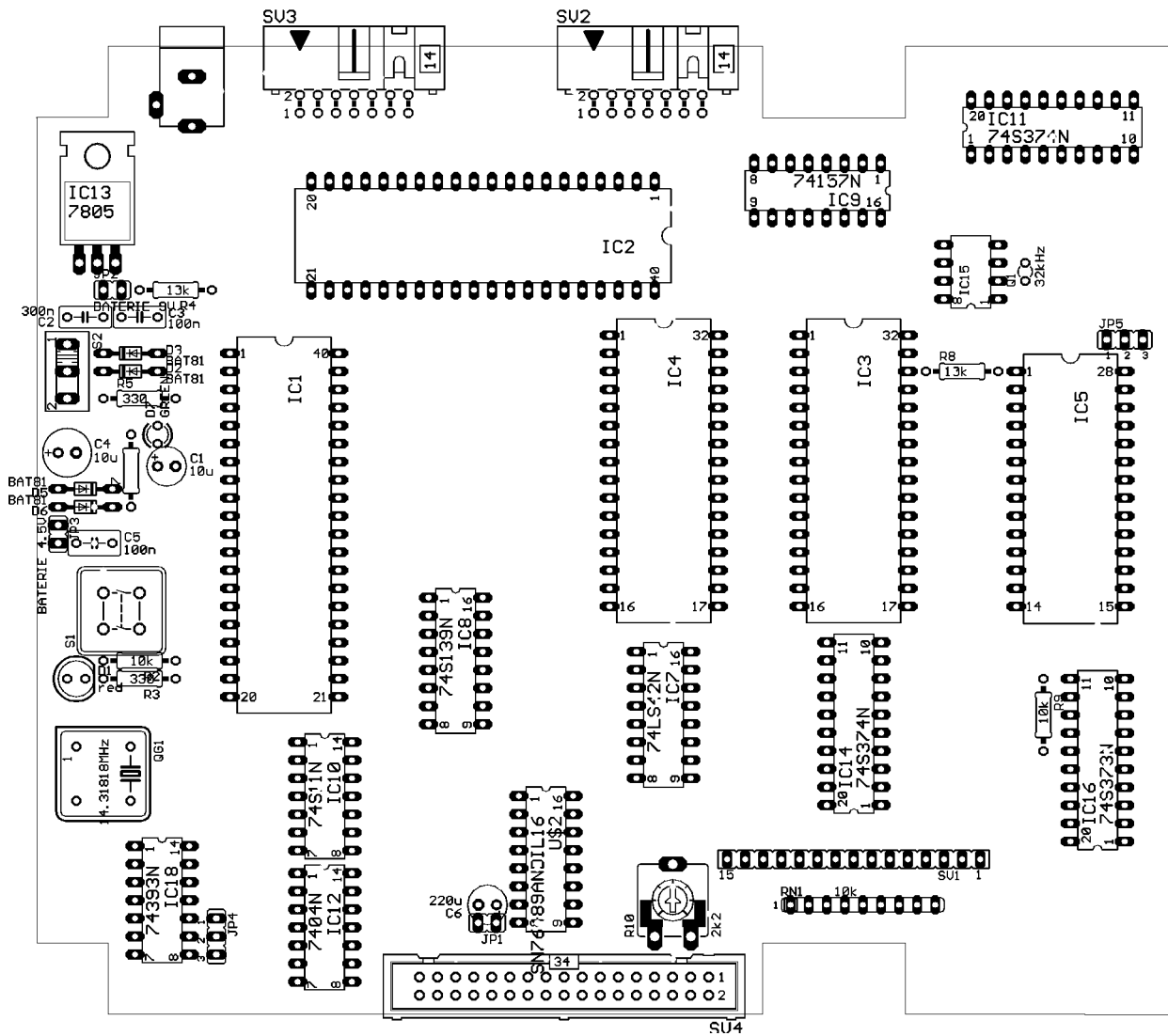
Mikropočítač nelze realizovat na jednostranné desce, a proto jsem musel použít oboustrannou. Pokoušel jsem se vytvořit ručně oboustrannou desku, ale narazil jsem na několik potíží. Největší problém jsou průchodky. Vytvořit je ručně dá dost práce a esteticky to nevypadá pěkně. Pokusil jsem se takovou desku vytvořit foto-cestou. I když jsem se trefil z obou stran, narazil jsem při pájení na problém s průchodkami v nožičkách integrovaných obvodů. S pistolovou pájkou se pod obyčejnou patici prostě nedostanete. Problém jsem zkoušel vyřešit zakoupením precizních patic, ale ty jsou drahé a pájení stále nebylo nejjednodušší. Proto jsem přizpůsobil návrh desky tak, aby vedly cestičky k nožičkám pouze ze strany spojů. Došlo k rapidnímu nárůstu prokovek, ale bylo to ještě v mezích normy. Aby se daly průchodky ručně zapájet musel jsem zvětšit jejich průměr. Původně jsem plánoval nechat výrobu na specializované firmě, ale nakonec s ohledem na zbývající čas a cenu, jsem vytvořil desku ručně. Na desce bylo asi 800 děr a 300 průchodek.



Obr. 12 – Deska plošného spoje – strana součástek



Obr. 13 – Deska plošného spoje – strana spojů



Obr. 14 – Deska plošného spoje – Osazovací nákras

#### 4.1 Rozpiska součástek

Označení	Hodnota	Pouzdro
C1	10uF	E2,5-6
C2	300nF	C050-030X075
C3	100nF	C050-030X075
C4	10uF	E2,5-7
C5	100nF	C050-035X075
C6	220uF	E2,5-6
D1	ČERVENÁ	LED3MM
D2	BAT81	DO34-7
D3	BAT81	DO34-7
D5	BAT81	DO34-7
D6	BAT81	DO34-7
D7	ZELENÁ	LED3MM
IC1	Z80CPUA	DIL40
IC2	8255A	DIL40
IC3	628128P	DIL32
IC4	628128P	DIL32
IC5	27256	DIL28
IC7	74HCT42	DIL16
IC8	74HCT139	DIL16
IC9	74HCT157	DIL16
IC10	74HCT11	DIL14



IC11	74HCT374	DIL20
IC12	74HCT04	DIL14
IC13	7805	TO220H
IC14	74HCT374	DIL20
IC15	DS1302	DIL08
IC16	74HCT373	DIL20
IC18	74HCT393	DIL14
IC19	SN76489AN	DIL16
JP1	JP1E	JP1
JP2	BATERIE 9V	JP1
JP3	BATERIE 4.5V	JP1
JP4	JP2E	JP2
JP5	JP2E	JP2
Q1	32kHz	TC26V
QG1	14.31818MHz	DIL08S
R2	10k	0207/10
R3	330	0207/10
R4	10k	0207/10
R5	330	0207/10
R7	10k	0207/10
R8	10k	0207/10
R9	10k	0207/10
R10	2k2	CA9V
RN1	10k	SIL9
S1	DTE6K	DTE6K
S2	255SB	255SB
SV1	FE15-1	FE15
SV2	ML14LE	ML14L
SV3	ML14LE	ML14L
SV4	ML34	ML34

## 5 Krabička

V původních požadavcích sice bylo vytvořit mikropočítač co možná nejmenší, ale při shánění vhodné krabičky jsem zjistil, že tento požadavek bude těžké naplnit. V katalogích jsem nenašel krabičku vhodných rozměrů, a proto jsem zakoupil dvě krabičky polovičních rozměrů. Každé jsem vyfrézoval jednu stranu tak, abych je mohl přiložit k sobě a slepit v jednu. Dalším krokem bylo vyřezání otvorů pro klávesnici a displej. Protože se na stabilizátoru uvolňuje teplo a docela hřeje i procesor, vyvrtal jsem do krabičky větrací otvory. Ty se ukázaly vhodné i pro průchod zvuku z reproduktoru. Nakonec jsem vyvrtal otvory pro napájecí konektor, paralelní port, napájecí spínač, regulátor hlasitosti reproduktoru a pro kontrolku napájení. Nákres krabičky se zakótovanými hlavními rozměry je uveden na poslední straně.

## 6 Programová výbava

Po softwarové stránce jsem odzkoušel všechna vstupně-výstupní zařízení. Z důvodů časové tísně však nebyl vytvořen patřičný ovládací program. Pro ideální funkci by byl potřeba operační systém s mnoha funkcemi. Vytvoření takového systému by zabralo několik měsíců. Hlavním požadavkem by bylo vytvořit programovací jazyk, aby bylo možno rozšiřovat systém přímo (sám na sobě) a programovat požadované aplikace v terénu. K vývoji programového vybavení jsem použil assembler AS80 stažený s internetu od Kingswood software. Pro efektivnější práci je však potřeba kompilátor vyššího jazyka. Kompilátor jazyka C jistě existuje ve verzi pro procesor Z80. Již před započatím práce jsem považoval za hlavní výhodu celého výrobku lepší softwarové schopnosti, ale neuvědomil jsem si obtížnost realizace. Protože tento výrobek nebyl vyroben z důvodů maturity (jako nejmenší zlo, ale k vlastnímu užití) hodlám do budoucna na softwaru aktivně pracovat a dovést tuto práci do konce.

## 6.1 Popis portů

Port	Zařízení
00H	Výběr sloupce klávesnice, ovládání RTC(CLK a RST/)
20H	Čtení stavu klávesnice
40H	Zápis instrukce do displeje
41H	Zápis dat do displeje
80H	Ovládání zvukového generátoru 76489
C0H	Ovládání paralelního portu (obvod 8255)
E0H	Registr řízení paměti

## 7 Závěr, zhodnocení a doporučení

Zařízení jsem přes mnohé komplikace zhotovil. Jako největší chybu celého projektu jsem shledal špatné rozvrhnutí časového plánu. I když ve skutečnosti nelze tento plán plně dodržet, protože se při vývoji postupně objevovaly další problémy. Nejvíce času jsem zpočátku ztratil při pokusech o programování paměti FLASH typu 28F. Když jsem promýšlel realizaci dlouhodobé paměti, neměl jsem dostatečné znalosti o všech možnostech a nakonec jsem zvolil snad nevhodnější variantu se zálohovanou pamětí. Až později z katalogových údajů jsem zjistil, rozdíl mezi typy paměti 28 a 29 a mezi typy F a C. Po tomto neúspěchu jsem opět špatně posoudil časovou náročnost jednotlivých fází vývoje a samotné realizace. Dalším problémem bylo vytvoření desky plošného spoje. Desku lze vyrobit vlastnoručně, ale deska je velice nekvalitní a výroba je pracná. Proto doporučuji, pokud máte dost času, si nechat desku zhotovit ve specializované firmě. Nakonec zbývá říci, že po navrhnutí a sestrojení tohoto výrobku, nejsem spokojen s jeho vlastnostmi a rád bych přidal další vylepšení. Toto by se však dalo provádět do nekonečna...

## 8 Literatura

1. Pavel Valášek - Monolitické mikroprocesory a mikropočítače, SNTL Praha 1989
2. Vladimír Dvořák - Mikroprocesor Z80 a programovatelné obvody, ČSVTS Praha 1988
3. Ladislav Zajíček – Bity do bytu, Mladá fronta Praha 1988
4. Amatérské rádio příloha Mikroelektronika, Naše vojsko Praha 1988
5. Martin Váňa, Mikropočítač SHARP MZ-800
6. Owner's manual of Personal Computer SHARP MZ-800, Sharp corporation 1984

Dokumentaci vypracoval ke svému maturitnímu výrobku Jiří Hajda dne 1.4.2002