

Výukový modul pro předmět mikropočítače: grafický LCD displej

Educational module for microcontrollers course:
graphic LCD display

Jiří Sedlák

Bakalářská práce
2008



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně

Fakulta aplikované informatiky

Ústav aplikované informatiky

akademický rok: 2007/2008

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Jiří SEDLÁK
Studijní program: B 3902 Inženýrská informatika
Studijní obor: Informační technologie

Téma práce: Výukový modul pro předmět mikropočítače: grafický LCD displej

Zásady pro vypracování:

1. Prostudujte hardwarové vlastnosti vývojového kitu M68EVB908GB60 a navrhnete způsob připojení modulu grafického displeje.
2. Provedte hardwarový návrh modulu grafického displeje používající STN displej MG12864 s rozlišením 128x64 bodů.
3. Realizujte prototyp modulu a ověřte jeho funkci na vývojovém kitu.
4. Vytvořte podpůrné programové vybavení pro obsluhu modulu ve formě knihovny podprogramů v jazyce symbolických adres a v C jazyce včetně demonstračního programu.

Rozsah práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. BURKHARD, Mann. C pro mikrokontroléry. Praha : BEN – technická literatura, 2003. 280 s. ISBN 80-7300-077-6.
2. Freescale Semiconductor. CPU08 Central Processor Unit Reference Manual., 2001. Dostupný z WWW: www.freescale.com
3. Freescale Semiconductor. HCS08 Family Reference Manual, Rev.1., 2003. Dostupný z WWW: www.freescale.com
4. Freescale Semiconductor. MC9S08GB/GT Data Sheet, Rev.2.3., 2004. Dostupný z WWW: www.freescale.com
5. HRABOVSKÝ, Miroslav, JURÁNEK, Antonín. EAGLE pro začátečníky : Návrhový systém pro plošné spoje. Praha : BEN – technická literatura, 2007. 192 s. ISBN 80-7300-213-2.
6. Intech LCD Group Ltd. GDM12864A: Liquid Crystal Display Module, 1998. Dostupný z WWW: www.datasheetcatalog.com
7. MATOUŠEK, David. Práce s inteligentními displeji LCD : 1. díl. Praha : BEN – technická literatura, 2006. 224 s. ISBN 80-7300-121-7.
8. VÁŇA V.: Začínáme s mikrokontroléry Motorola HC08 Nitron. Praha: BEN – technická literatura, 2003. 96 s. ISBN 80-7300-124-1.

Vedoucí bakalářské práce:

Ing. Petr Dostálek

Ústav automatizace a řídicí techniky

Datum zadání bakalářské práce:

20. února 2008

Termín odevzdání bakalářské práce:

5. května 2008

Ve Zlíně dne 20. února 2008

prof. Ing. Vladimír Vašek, CSc.

děkan



doc. Ing. Ivan Zelinka, Ph.D.

ředitel ústavu

ABSTRAKT

Cílem této bakalářské práce, bylo vytvořit výukový modul s LCD grafickým displejem. V teoretické části je popsána problematika mikropočítačů, zobrazovacích jednotek s malou hustotou informací a algoritmu pro rasterizaci jednoduchých objektů. Dále je v teoretické části popsán návrhový systém EAGLE, pro tvorbu návrhu schémat a plošných spojů.

Ve druhé části je popsán návrh a realizace celého modulu s LCD grafického displejem ATM12864D. Na závěr je popsána knihovna funkcí pro práci s tímto modulem.

Klíčová slova: mikropočítač, LCD, EAGLE, Freescale

ABSTRACT

The aim of this bachelor thesis was to create education module with LCD graphic display. In theoretical part microcontrollers, display devices with low information density and algorithm for rasterization of simple object are described. In next theoretical part EAGLE, which is design software for designing cabiling diagrams and printed circuit boards, is descibed.

In the second part design and creation of LCD graphic display ATM12864D module is described. The final part is about library of functions for operation this module.

Keywords: microcontroller, LCD, EAGLE, Freescale

Rád bych poděkoval mému vedoucímu práce, kterým je Ing. Petr Dostálek za jeho rady, připomínky a pomoc při řešení této práce.

..

Prohlašuji, že jsem na bakalářské práci pracoval samostatně a použitou literaturu jsem citoval. V případě publikace výsledků, je-li to uvolněno na základě licenční smlouvy, budu uveden jako spoluautor.

Ve Zlíně

.....
Podpis diplomanta

OBSAH

| | |
|--|-----------|
| ÚVOD | 9 |
| I TEORETICKÁ ČÁST | 10 |
| 1 JEDNOČIPOVÉ MIKROPOČÍTAČE | 11 |
| 1.1 HISTORIE MIKROPOČÍTAČŮ..... | 11 |
| 1.2 DĚLENÍ MIKROPOČÍTAČŮ..... | 12 |
| 1.2.1 Podle architektury | 12 |
| 1.2.2 Podle Instrukční sady | 14 |
| 1.3 VÝVOJOVÝ KIT M68EVB08GB60 | 14 |
| 1.3.1 Napájení | 16 |
| 1.3.2 Svorkovnice TB1 | 16 |
| 1.4 MIKROPOČÍTAČ MC9S08GB | 16 |
| 1.4.1 Základní vlastnosti | 16 |
| 1.4.2 Registry | 17 |
| 1.4.3 Paměť | 20 |
| 1.4.4 Porty | 20 |
| 2 ZOBRAZOVACÍ JEDNOTKY | 21 |
| 2.1 VLASTNOSTI ZOBRAZOVACÍCH JEDNOTEK | 21 |
| 2.2 ZOBRAZOVACÍ JEDNOTKY S MALOU HUSTOTOU ZOBRAZOVANÉ INFORMACE..... | 21 |
| 2.3 VAKUOVÁ KATODO LUMINISCENČNÍ JEDNOTKA – VFD..... | 22 |
| 2.3.1 Konstrukce VFD | 22 |
| 2.3.2 Vlastnosti VFD..... | 23 |
| 2.4 ZOBRAZOVACÍ JEDNOTKY S LED DIODAMI | 23 |
| 2.4.1 Uspořádání 7-segmentového displeje | 23 |
| 2.4.2 Vlastnosti jednotek s LED diodami | 24 |
| 2.5 ZOBRAZOVACÍ JEDNOTKY S KAPALNÝMI KRYSTALY – LCD..... | 24 |
| 2.5.1 Definice kapalných krystalů..... | 24 |
| 2.5.2 Struktura kapalných krystalů | 25 |
| 2.5.3 Způsob sledování znaků..... | 25 |
| 2.5.4 TN (Twist Nematic) | 26 |
| 2.5.5 Princip zobrazovací jednotky LCD..... | 26 |
| 2.5.6 Zdokonalené principy LCD..... | 27 |
| 2.6 MULTIPLEXNÍ BUZENÍ MATICOVÝCH ZOBRAZOVACÍCH JEDNOTEK | 28 |
| 3 RASTERIZACE JEDNODUCHÝCH OBJEKTŮ | 29 |
| 3.1 ÚSEČKA..... | 29 |
| 3.1.1 DDA algoritmus | 30 |
| 3.1.2 Bresenhamův algoritmus..... | 30 |
| 3.2 KRUŽNICE | 31 |
| 3.2.1 Bresenhamův algoritmus..... | 32 |
| 4 EAGLE | 33 |

| | | |
|-----------|---|-----------|
| 4.1 | CONTROL PANEL | 33 |
| 4.1.1 | Založení projektu | 34 |
| 4.2 | EDITOR SCHÉMAT | 35 |
| 4.3 | EDITOR NÁVRHU DPS | 36 |
| II | PRAKTICKÁ ČÁST | 38 |
| 5 | NÁVRH MODULU | 39 |
| 5.1 | LCD MODUL ATM12864D | 39 |
| 5.1.1 | Vlastnosti | 39 |
| 5.1.2 | Konstrukce displeje | 42 |
| 5.1.3 | Instrukce pro řízení Displeje | 42 |
| 5.1.4 | DDRAM (Display Data RAM) | 44 |
| 5.2 | NÁVRH SCHÉMATU | 45 |
| 5.2.1 | Napájení | 45 |
| 5.2.2 | Řídící logika | 46 |
| 5.3 | DESKA PLOŠNÉHO SPOJE | 47 |
| 5.3.1 | Návrh DPS | 47 |
| 5.3.2 | Výroba DPS | 47 |
| 5.3.3 | Osazení plošného spoje | 48 |
| 5.4 | PROPOJENÍ MODULU A VÝVOJOVÉHO KITU | 49 |
| 6 | PROGRAMOVÉ VYBAVENÍ | 50 |
| 6.1 | PROMĚNNÉ | 50 |
| 6.1.1 | DDRAM | 50 |
| 6.1.2 | ABECEDA | 50 |
| 6.2 | UŽIVATELSKÉ FUNKCE | 50 |
| 6.2.1 | DISP_INIT | 50 |
| 6.2.2 | ERASE_ALL | 51 |
| 6.2.3 | DISP_DDRAM | 51 |
| 6.2.4 | DISP_POINT | 51 |
| 6.2.5 | DRAW_LINE | 51 |
| 6.2.6 | DRAW_CIRCLE | 52 |
| 6.2.7 | DISP_CHAR | 52 |
| 6.2.8 | DISP_STRING | 53 |
| 6.3 | VNITŘNÍ FUNKCE | 53 |
| 6.3.1 | POSLI_DISP | 53 |
| 6.3.2 | CIRCLE_POINTS | 53 |
| 6.3.3 | ABECEDA_SAVE | 53 |
| | ZÁVĚR | 54 |
| | CONCLUSION | 55 |
| | SEZNAM POUŽITÉ LITERATURY | 56 |
| | SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK | 58 |
| | SEZNAM OBRÁZKŮ | 59 |
| | SEZNAM TABULEK | 60 |

SEZNAM PŘÍLOH.....61

ÚVOD

V dnešní době, se stále častěji setkáváme s různými elektronickými zařízeními, které nám předávají informace, pomoci zobrazovacích jednotek.

Na počátku rozvoje této spotřební elektroniky, se informace zobrazovaly pomocí žárovek a digitronů. S nástupem polovodičové techniky, se začaly využívat LED diody uspořádané do segmentů a matic. Posléze se začaly využívat LCD displeje.

V dnešní době už je LCD spíše standardem než výjimkou a proto vznikl požadavek na vytvoření výukového modulu s grafickým LCD displejem. Tato práce se zabývá právě vytvořením takového modulu.

V teoretické části se budeme zabývat vývojovým kitem s mikroprocesorem rodiny HCS08, jeho základními vlastnostmi, registry, paměti a porty. Pak budou následovat zobrazovací jednotky s malou hustotou informací, na to namazující algoritmy pro pasterizaci jednoduchých objektů. Na závěr teoretické části se podíváme na program EAGLE, který slouží k návrhu schémat a desek plošných spojů.

Praktická část je rozdělena na dvě části. V první části se zabýváme seznámením s technickými parametry LCD displeje ATM12864D, poté pokračujeme návrhem schématu a tvorbou plošného spoje v programu EAGLE. Na závěr je popsán postup pro výrobu plošného spoje a jeho osazení. Ve druhé části, která se zabývá programovým vybavením modulu. Jsou popsány proměnné jejich význam a velikost, kterou zabírají v paměti, dále jsou popsány funkce přístupné uživateli a na závěr funkce vnitřní, tedy nepřístupné.

I. TEORETICKÁ ČÁST

1 JEDNOČIPOVÉ MIKROPOČÍTAČE

1.1 Historie mikropočítačů

První integrovaný obvod, který můžeme nazvat mikroprocesor, navrhl pracovník firmy INTEL E. Hoff při práci na vývoji obvodů pro stolní kalkulač. Napadlo jej přenést na programovatelnou strukturu univerzálního procesoru řešení úloh, které se původně předpokládalo řešit sekvenčním automatem. Výsledný obvod byl nazván mikroprocesor a označen 4004. Měl šířku dat 4 bity, instrukční soubor orientovaný hlavně na práci s čísly BCD a na logické funkce, měl rychlost 108kHz a obsahoval 2 300 tranzistorů. Vývoj pokračoval dále za účasti Federico Faggina a Masatoshi Shimmy (zakladatelů firmy ZILOG) a byla vytvořena mikropočítačová sada MCS4, která byla inzerátem nabídnuta k obecnému použití. Úspěch byl překvapující, za 3 měsíce bylo prodáno na 85 000 sad a prokázalo se, že doba dozrává pro zavedení programovatelných automatů typu mikropočítač. Začátkem roku 1974 pak firma INTEL přišla s typem 8080, (počet tranzistorů: 6 000, rychlost: 2 MHz), který se stal na delší dobu jakýmsi světovým standardem. V ČSR se vyráběl pod označením MHB8080A. Asi rok po zavedení typu 8080 uvedla na trh firma MOTOROLA mikroprocesor MC6800 a v roce 1976 firma ZILOG mikroprocesor Z80-CPU. Toto byli hlavní představitelé osmibitových mikroprocesorů. Od roku 1978 jsou středem pozornosti 16bitové mikroprocesory. Mikroprocesor 8086 firmy INTEL uvedený na trh v roce 1978 byl vyroben technologií HMOS s 29 000, rychlost: 5 MHz, 8 MHz, 10 MHz. S ročním zpožděním nabídla firma ZILOG mikroprocesor Z8000 a firma MOTOROLA mikroprocesor MC68000, který se navenek jeví jako 16bitový, ale vnitřní strukturu má z velké části 32bitovou. Koncem 70. let nevznikaly jen 16bitové mikroprocesory, ale i monolitické mikropočítače, které představují samostatnou větev vývoje od 8bitového obvodu 8048 až po 16bitové jako např. 80186. Vývoj samozřejmě pokračuje stále kupředu přes transputery a procesory typu RISC. V současné době jsou v počítačích používány 64bitové procesory.

Největšího rozšíření z hlediska aplikací doznaly mikrokontroléry dvou významných výrobců mikroprocesorů - firem Motorola a Intel (podobně jako v technice stolních mikropočítačů). Spektrum výrobců jednočipových mikropočítačů je však velmi široké, mimo již jmenované firmy jsou to například americké Texas Instruments, National Semiconductors, evropské Philips, Siemens, japonské OKI, NEC, Toshiba, Hitachi apod.

Charakteristické pro historický vývoj mikrokontroléru však je, že se zde neprojevuje snaha o drastické zvýšení rychlosti a kapacity paměti jako u osobního počítače. Je to dáno zejména faktem, že nasazením řídicího mikropočítače v dané aplikaci musí dojít k podstatnému zjednodušení výroby a tím ke snížení ceny výsledného výrobku. Stačí-li proto například k vykonání specifického řídicího programu operační paměť 64 bytů, není výrobce ničím motivován, aby použil mikrokontrolér s pamětí 128 bytů, protože by tím jen zdražil výsledný výrobek. Obecně platí, že se pro aplikaci použije ten nejjednodušší mikrokontrolér, který pro danou aplikaci vyhovuje. Tato motivace je proto také důvodem k tomu, že výrobci jednočipových mikropočítačů nabízejí velmi široké spektrum konfigurací mikrokontrolérů, lišící se i drobnými změnami v kapacitě paměti, počtu vstupů/výstupů, použitelných periférií apod. Přitom záměr výrobce aplikace je co nejjednodušší (a tím nejlevněji vyrobitelná a nejspolehlivější) sestava potřebného řídicího hardware, nejlépe složená pouze z vlastního mikrokontroléru. [6]

1.2 Dělení mikropočítačů

Mikropočítače můžeme rozdělit pomocí dvou obecných parametrů:

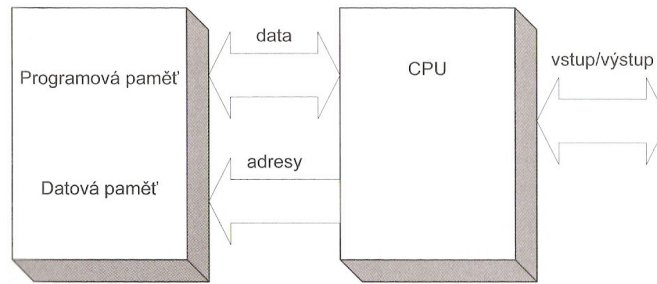
- architektury (von Neuman, Harvardská)
- instrukční sady (RISC, CISC)

1.2.1 Podle architektury

Pokud jde o architekturu jako výkonnostní charakteristiku, používají se doposud hlavně von Neumannova architektura a Harvardská architektura, v poslední době také kombinace obou architektur.[3]

Von Neumanova architektura

Von Neumanova architektura se vyznačuje tím, že datová a programová paměť používají společné vedení (adresní a datová sběrnice).

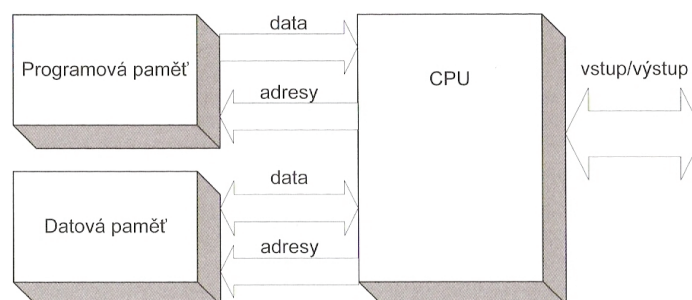


Obr. 1. Architektura von Neuman

Výhodou je hospodárnost využívání sběrnice systémů, jehož výsledkem je snížení počtu vývodů mikročítače při použití externí paměti. Nevýhodou je ovšem neodvratné snížení rychlosti zpracování, protože programová i datová paměť musí sdílet tatiž adresní i datová vedení. Z paměti je možno číst jen buď příkazy nebo data. Datová a programová paměť pokrývá tutěž oblast paměti, o volbě paměti rozhodují řídicí linky.

Harvardská architektura

Harvardská architektura je z obou těchto architektur starší. Má oddělené sběrnice systémy pro programovou a datovou paměť. Díky tomu je možno současně přistupovat k datům i k příkazům, což je základním předpokladem pro vysokou rychlost zpracování. Typická je i rozdílná bitová šířka datové a programové paměti. Programová paměť je navržena tak, že se jedním přístupem může číst příkaz i operand.



Obr. 2. Harvardská architektura

Z hlediska hospodárnosti existují i modifikované Harvardské architektury. Interně je mikročítač navržena podle Harvardské architektury a vzhledem k vnějšku, pro přístup do externí paměti, podle von Neumanovy architektury. Tím se ušetří náklady na výrobu čipu snížením počtu potřebných vývodů. [3]

1.2.2 Podle Instrukční sady

Pro druhou výkonnostní charakteristiku, rozsah příkazů a počet hodinových cyklů na jeden příkaz, je rozhodující interpretace příkazů, RISC nebo CISC:

CISC (Complex Instruction Set Computer)

Je to tradiční interpretace příkazů. Existuje mnoho výkonných příkazů, takže zadáním jediného příkazu je možno realizovat složité operace, čímž se ušetří paměťové místo. To zní zpočátku velmi dobře, ale tyto příkazy vyžadují složitý interpret příkazů, který příslušný příkaz zpracovává během několik hodinových cyklů.

RISC (Reduced Instruction Set Computer)

Ze zkušeností s architekturou CISC se přešlo k RISC, silně redukované sadě instrukcí obsahuje většinou až 50 instrukcí. Všechny mají stejnou bitovou šířku, jsou méně složité a vyžadují méně složitý interpret příkazů ve srovnání s CISC. Tím je možno příkazy podstatně efektivněji dekodovat a zpracovávat, zpravidla v jednom hodinovém cyklu. Tím se neobyčejně zvyšuje rychlost zpracování, ale bohužel také velikost programu při téže prováděné funkci. [3]

1.3 Vývojový kit M68EVB08GB60

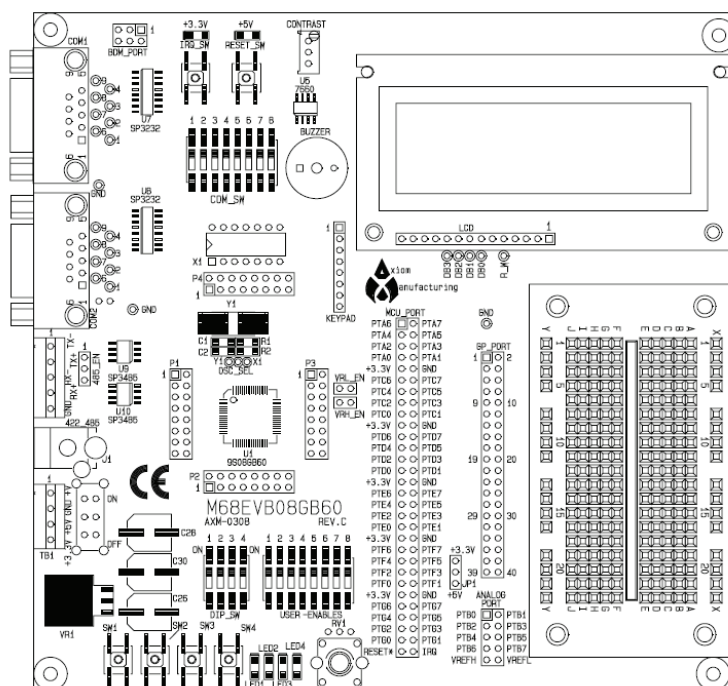
M68EVB08GB60 je vyhodnocovací nebo také vývojová deska pro mikropočítač M9S08GB60. Vývoj aplikací je rychlý a snadný s přiloženým sériovým kabelem, vzorovým softwarovými nástroji, příklady a ladícím programem. Dále je vybaven kontaktním nepájivým polem a systémovým konektorem pro přístup k portům mikropočítače viz obrázek (Obr.11). [5]

Parametry kitu:

- M9S08GB60 mikropočítač
- 32 kHz až 4 MHz oscilátor
- nastavitelný generátor hodinového signálu
- stabilizované nepájecí napětí 3,3V a 5V
- COM1 Sériovou linku s konektorem DB9 – SCIO0 Sériový port
- COM2 Sériovou linku s konektorem DB9 nebo RS422/455 – SCIO1 Sériový port

- Signalizace napájení , vypínač napájení
- Uživatelské periferie
 - 4 led diody (PTF0-3)
 - 4 DIP přepínače (PTB4-7)
 - 4 tlačítka (PTA4-7)
 - LCD modul 2x16 znaků (PTG3-7, PTE6-7)
 - Bzučák (PTD0)
- MCU Port konektor s digitálními vstupy a výstupy
- Analog Port konektor s analogovými vstupy a výstupy
- Kontaktní nepřájivé pole 5,08 cm x 12,70 cm

Kit je dodáván s operačním monitorem, který umožňuje uživateli přístup přes sériové rozhraní, pro ladění a vývoj kódu.



Obr. 3. Vývojový kit M68EVB08GB60

1.3.1 Napájení

Vstupní napětí je realizováno externím zdrojem připojeným na napájecím konektor J1 nebo svorkovnicí TB1. Napájení je možno vypnout pomocí vypínače ON_OFF. Regulátor napětí je chráněn proti přepólování diodou D3 a proud je omezen pojistkou FZ1. Vstupní napětí je regulováno na +5V pomocí stabilizátoru VR1 a toto napětí je opět regulováno na +3,3V pomocí stabilizátoru VR2. Zapnuté napájení +5V a +3,3V je indikováno LED diodami.[5]

1.3.2 Svorkovnice TB1

TB1 zajišťuje přístup k napájecímu napětí +VIN, k elektrické zemi GND a stabilizovaným napětím +5V a +3,3V. Napájecí napětí +VIN se nedá vypnout ON_OFF vypínačem a není omezen pojistkou FZ1. Stabilizovaná napětí +5V a +3,3V mohou sloužit pro napájení externích obvodů maximálně do 50 mA. Napájet desku přes tyto dvě napětí z externího zdroje není vhodné.[5]

1.4 Mikropočítač MC9S08GB

Je členem rodiny vysoce výkonných a levných 8-bitových mikropočítačů HCS08. Všechny mikropočítače této rodiny jsou vybaveny mikroprocesorem HCS08. Jsou dostupné s různými velikostmi paměti, typu paměti, periferiemi a pouzdry[4]. Mikroprocesor HCS08 je postaven na architektuře von Neuman a má instrukční sadu typu CISC.

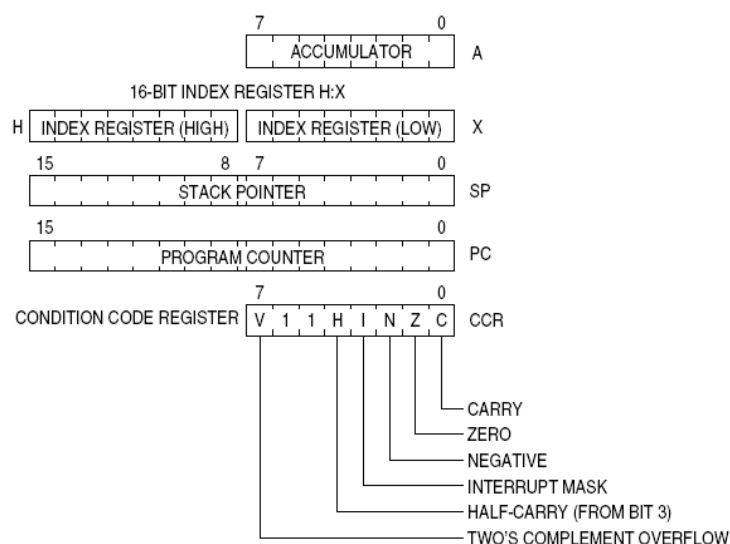
1.4.1 Základní vlastnosti

- 8 bitová centrální procesní jednotka HCS08, taktovací frekvence max. 40MHz
- 60KB FLASH paměti
- 4KB RAM paměti
- 56 vstupně/výstupních linek na 7 portech (porty A-G)
- 5 kanálový TPM2 časovač
- 3 kanálový TPM1 časovač
- 1x Synchronní sériové periferní rozhraní (SPI)
- 1x I2C rozhraní

- 2x Asynchronní sériové komunikační rozhraní (SCI)
- Interní generátor hodinového kmitočtu s FLL obvodem (32KHz – 20MHz frekvence sběrnice)
- 8 kanálový, 10 bitový A/D převodník
- COP watchdog systém s nastavitelnou časovou prodlevou na 218 nebo 213 cyklů sběrnice
- Systém kontroly napájecího napětí umožňující detekovat pokles napětí pod stanovenou mez
- Podpora režimů se sníženou spotřebou umožňující nasazení v bateriově napájených aplikacích
- BDM rozhraní pro pokročilé ladění a programování aplikací přímo v aplikaci

1.4.2 Registry

Procesor má pět registrů, které nejsou součástí paměti viz obrázek(Obr.12).



Obr. 4. Registry HC08

Akumulátor (A - Accumulator)

Akumulátor je 8-bitový univerzální registr. Jeden operand se nahraje do ALU (aritmeticko logická jednotka), ten se spojí s akumulátorem, provede se výpočet a výsledek operace se

uloží opět do akumulátoru. Akumulátor může být naplněn nebo uložen z/do paměti za použití různých způsobů adresování. Reset nemá vliv na obsah akumulátoru.[4]

Index Registr (H:X)

Tento registr je složen ze dvou 8 bitových registrů H a X, které spolu pracují jako jeden 16 bitový adresní ukazatel, kde H je vyšší bajt adresy a X nižší. Všechny adresové instrukce používají plnou 16 bitovou hodnotu. Avšak kvůli zpětné kompatibilitě se starším procesorem HC05, některé instrukce používají pouze dolní byte X.

Spousta instrukcí používá X jako univerzální registr vedle akumulátoru, může být použit k uchování dat. S registrem X se mohou provádět i další operace jako nulování, inkrementace, dekrementace, rotace, porovnání, negace nebo posun. Instrukce pro přesun dat umožňují provádět transfer se zásobníkem, kde se provedou aritmetické nebo logické operace.

Z důvodu kompatibility z HC05, je během restartu provedeno nastavení registru H na \$00. Registr X však reset nijak neovlivní. [4]

Ukazatel zásobníku (SP – Stack Pointer)

Tento 16 bitový adresový registr ukazuje na další dostupnou pozici v zásobníku. Zásobník je paměť typu LIFO (Last in –First out). Zásobník může být kdekoliv v celém adresovém prostoru, kde se nachází paměť RAM. Jeho velikost může být nastavena na maximální velikost RAM. Zásobník je používán k uložení návratové adresy při volání podprogramu a také uložení stavových registrů při přerušení. Může být také využit pro ukládání lokálních proměnných nebo předávání parametrů funkcím.

Pomocí instrukce AIS můžeme k registru SP přičíst 8 bitové číslo se znaménkem. Toho je často využíváno při alokaci nebo uvolnění lokálních proměnných.

SP je po restartu nastaven na hodnotu \$00FF z důvodů zpětné kompatibility. Programově můžeme nastavit adresu zásobníku kdekoliv ve volné RAM.[4]

Programový čítač (PC - Program Counter)

Programový čítač je 16 bitový registr, který obsahuje adresu další instrukce nebo operandu, který se má vykonat.

Během běžného vykonávání programu se čítač automaticky inkrementuje na adresu dalšího operandu nebo instrukce, který se má vykonávat. Skok, přerušení, volání podprogramu nebo návrat se nahraje příslušná adresa skoku nebo návratu.

Během restartu se do programového čítače nahraje vektor na adrese \$FFFE a \$FFFF. Vektor ukládá adresu první instrukce, která se má po restartu provést.[4]

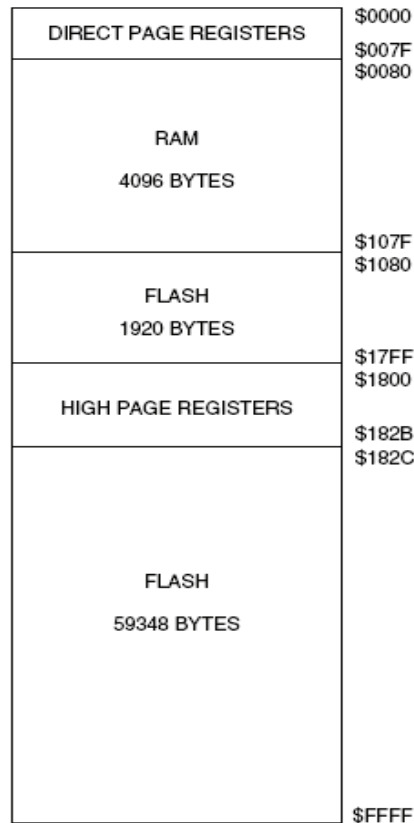
Stavový registr (CCR - Condition Code Register)

8 bitový stavový registr obsahuje masku přerušení I a pět příznaků, které indikují výsledky právě vykonané instrukce. Bity 5 a 6 jsou permanentně nastaveny na 1.

- **V – Two's Complement Overflow Flag** Procesor nastaví příznak přetečení, když nastane přetečení dvojkového doplňku. Tento příznak využívají instrukce BGT, BGE, BLE a BLT.
- **H – Half-Carry Flag** Tento příznak se nastaví, když dojde k přenosu v akumulátoru z 3 na 4 bit. Příznak se využívá při operacích v BCD kódu. Příkaz DAA podle něj reviduje součet pomocí ADD (standardní binární součet) na součet v kódu BCD.
- **I — Interrupt Mask Bit** Pokud je nastavena maska přerušení, všechny maskovaná přerušení jsou zakázána. Přerušení se povolí, když je maska nastavena na 0. Když nastane přerušení, tak se maska přerušení automaticky nastaví na 1, po uložení registru na zásobník.
- **N — Negative Flag** Procesor nastaví tento příznak, když aritmetická, logická nebo přesouvací operace má záporný výsledek. To znamená, že je nastaven 7 bit. Změnu příznaku také změní prosté načtení nebo uložení 8 nebo 16 bitové hodnoty pokud je nastaven nejvýznamnější bit na 1.
- **Z — Zero Flag** Příznak se nastaví pokud je provedena aritmetická, logická nebo přesouvací operace s nulovým výsledkem. I prosté načtení hodnoty čísel \$00 nebo \$0000 má za následek vyvolání tohoto příznaku.
- **C — Carry/Borrow Flag** Pokud nějaká instrukce způsobí přenos z nejvyššího bitu akumulátoru nebo když operace dělení potřebuje výpůjčku, tak se bit nastaví na jedna. Některé instrukce jako bit test nebo skok, případně rotace, také mohou způsobit změnu bitu. [4]

1.4.3 Paměť

Na obrázku (Obr. 13) můžeme vidět paměťovou mapu mikropočítače MC9S08GB60. Skládající se z paměti RAM, FLASH paměti programu a kontrolní/příznakové registry vstupů a výstupů.



Obr. 5. Mapa paměti

1.4.4 Porty

Mikropočítač má 7 vstupně výstupních portů dohromady tedy 55 univerzálních pinů (plus jeden pin je pouze výstupní). Mnoho těchto pinů je sdíleno s integrovanými periferiemi jako například časovač, A/D převodník nebo přerušeni od klávesnice.

Pokud není pin sdílen s některou periferií, přejde opět do univerzálního vstupně výstupního režimu. Každý port má stavové registry, které nastavují zda bude port vstupní nebo výstupní, zda se mají aktivovat pullup rezistory. Po restartu jsou všechny piny nastaveny na vysokou impedanci jako vstupy bez vnitřních pullup rezistorů.

Jednotlivé porty se značí písmeny A-G a bity se značí čísly 0-7, takže přesné určení pinu se zapíše například takto PTA0.[4]

2 ZOBRAZOVACÍ JEDNOTKY

Zobrazovací jednotky jsou dnes nedílnou součástí většiny elektronických zařízení a systémů. Předávají určitou vizuální informaci mezi procesem a uživatelem elektronického zařízení. Můžeme je rozdělit na dvě základní skupiny: [2]

- S malou hustotou informací
- S velkou hustotou informací.

2.1 Vlastnosti zobrazovacích jednotek

Viditelnost – je posouzení snadnosti a správnosti čtení informací. Viditelnost je ovlivněna různými činiteli, jako jsou jas, kontrast, barva.

Spotřeba – je dána spotřebovanou energií a velikostí potřebného napájecího napětí, aby bylo možné porovnávat spotřebu různých displejů, je spotřeba udávána pro určitý zobrazovaný znak, zpravidla číslici 8.

Rychlost odezvy – je čas potřebný k výměně nového údaje za starý. Doba nutná k tomu, aby byl na displeji odstraněn starý a zobrazen nový údaj.

Životnost – provoz displeje je považován za ukončený, když jas v porovnání s počátkem klesne na polovinu

Zorný úhel – maximální úhel, pod kterým je možné s dobrou čitelností sledovat zobrazení na displeji.

Princip zobrazování:

- Aktivní – generující optické záření a proto je lze vidět i bez osvětlení.
- Pasivní – vyžadují cizí osvětlení, samy negenerují optické záření, ale řídí jeho odraz nebo průchod

2.2 Zobrazovací jednotky s malou hustotou zobrazované informace

Zobrazované symboly lze vytvořit jedním ze tří základních způsobů:

- **Zobrazovací jednotky s předem připravenými znaky** – jejich představiteli jsou indikační číslicová výbojky plněné zředěným plynem známé jako digitrony. Jsou

tvořeny soustavou katod, které mají tvar zobrazovaných znaků a jednou anodou tvořenou drátovou mřížkou

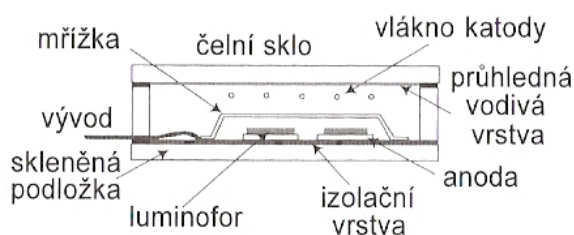
- **Segmentové zobrazovací jednotky** – jsou využívány převážně pro zobrazení číslic, mohou zobrazovat i některé speciální symboly. Příslušné znaky se zobrazí rozsvícením odpovídajících segmentů.
- **Maticové zobrazovací jednotky** – umožňují zobrazit číslice i písmena případně další znaky. Znaky jsou tvořeny body. Které jsou sestaveny do matice složené z řádků a sloupců. Maticové displeje obsahující dostatečný počet bodů jsou používány také jak grafické displeje.

2.3 Vakuová katodo luminiscenční jednotka – VFD

Vakuové luminiscenční zobrazovací jednotky byly vyvinuty v 60. letech minulého století v Japonsku. V současné době jsou vyráběny jak display složené ze segmentů, tak z bodových matic a také různobarevné grafické display pro široký rozsah aplikací. [1]

2.3.1 Konstrukce VFD

Jednotky pracují na obdobném principu jako vakuové obrazovky. Zobrazovací jednotka obsahuje tři elektrody, anodu, mřížku a katodu, umístěné ve skleněném pouzdru o vysokém vakuu. Princip uspořádání VFD je uveden na obrázku (Obr.1)



Obr. 6. Uspořádání VFD

Katoda je tvořena tenkými wolframovými drátky potaženými oxidy kovů alkalických zemin, které emitují elektrony. Zpravidla je přímo žhavená.

Mřížka je tvořena tenkou kovovou sítí. Která řídí a rozptyluje elektrony emitované z katod.

Anody mají tvar segmentů, určitých znaků nebo bodů. Jsou vytvořeny z vodivého materiálu pokrytého fluorescenčním povlakem.

Elektrony emitované katodou, jsou urychlovány kladným napětím mřížky . Po přivedení kladného napětí také na segment anody, narazí tyto elektrony na fluorescenční vrstvu a způsobí, že daný segment se rozsvítí. Na segment, který má zůstat zhasnutý se přivede záporné napětí. Je-li na mřížku přivedeno záporné napětí, jsou elektrony odpuzovány a nemohou se dostat k anodám.

Zobrazovací jednotky jsou zpravidla dodávány se zabudovanou řídicí elektronikou. Signály k ní je možné přivádět pomocí sériového nebo paralelního rozhraní. [1]

2.3.2 Vlastnosti VFD

- Velmi jasné zobrazení, zpravidla zelené, je možné použít i další barvy
- Široký pozorovací úhel
- Rozsah provozních teplot -40°C až $+85^{\circ}\text{C}$
- Snadné vytváření různých speciálních znaků
- Velká životnost
- Provozní napětí pro mřížku a anodu podle typu 10 V až 60 V
- Napájení katody 1,5 V až 5V dle typu

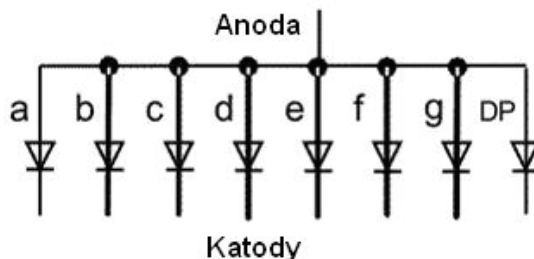
2.4 Zobrazovací jednotky s LED diodami

Tyto zobrazovací jednotky jsou tvořeny LED diodami buď v segmentovém zapojení nebo v maticovém uspořádání. Segmentové se vyrábějí jedno až 4 místné. Stejně, jako LED diody, jsou i displeje jsou vyráběny v různých barvách, nejčastěji však zelené a červené.

Informační panely pozorovatelné z velké vzdálenosti, jsou konstruovány s diskrétními LED diodami. [1]

2.4.1 Uspořádání 7-segmentového displeje

Každý ze sedmi segmentů je tvořen jednou nebo několika LED diodami. Pro zjednodušení ovládacích obvodů mají diody navzájem propojeny katody nebo anody. Záleží to na druhu použitého logického obvodu. Obvody typu TTL pracuje se společnou anodu (Obr.2), CMOS pracuje se společnými katodami. Výška znaku je podle typu indikátoru 3 až 16mm. Doba odezvy je řádově deset nanosekund. [2]



Obr. 7. Uspořádání segmentového displeje

2.4.2 Vlastnosti jednotek s LED diodami

- Nízké napájecí napětí kolem 3V
- Poměrně nízká cena
- Zobrazovací jednotky jsou vyráběny v mnoha různých rozměrech
- Barva a svítivost se mění s teplotou a s velikostí napájecího proudu
- Jsou vyráběny i displeje s velmi vysokým jasem

2.5 Zobrazovací jednotky s kapalnými krystaly – LCD

Zobrazovací systémy s tekutými krystaly mají široké využití. Od miniaturních typů v náramkových hodinkách se rozvoj LCD displejů soustředil na obrazovky přenosných počítačů a na náhradu obrazovek stacionárních počítačů. Na významu nabývá i prostorové zobrazení. [1]

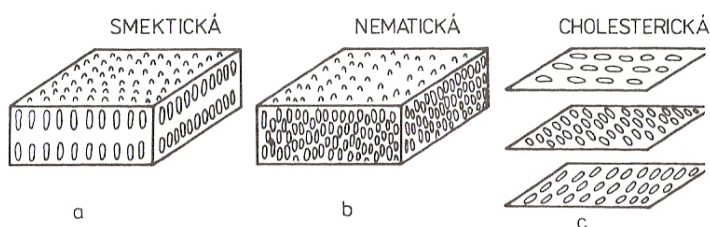
2.5.1 Definice kapalných krystalů

Kapalné krystaly jsou téměř průzračné substance, které vykazují vlastnosti pevné hmoty i kapaliny. Je pro ně příznačné, že v určitém rozmezí teplot u nich existuje stabilní mezifáze, ve které jsou molekuly v kapalině uspořádány obdobně jako v krystalu pevné látky. Vazby molekul však nejsou příliš silné, molekuly se mohou pohybovat. Pohyb molekul může být vyvolán mechanickými vlivy, ale také vhodně polarizovaným elektrickým nebo magnetickým polem. Změna uspořádání molekul vyvolaná některým z uvedených vlivů má za následek změnu optických vlastností, která se projevuje změnou propustnosti optického záření.[1]

2.5.2 Struktura kapalných krystalů

Krystaly mohou být uvnitř kapaliny uspořádány třemi základními způsoby:

- Smekticky – všechny krystaly jsou stejně dlouhé, leží v rovnoběžných vrstvách a mají navzájem rovnoběžné osy (Obr.3a).
- Nematicky – krystaly jsou různě dlouhé, navzájem nejsou rovnoběžné, vyplňují celý prostor (Obr.3b).
- Cholestericky – krystaly jsou uloženy ve vrstvách. Krystaly v jedné vrstvě mají rovnoběžné osy, avšak každá vrstva má směr os krystalů proti sousední vrstvě o určitý úhel posunutý (Obr.3c).



Obr. 8. Struktury kapalných krystalů

V současné době mají největší význam tekuté krystaly s nematickou strukturou. [2]

2.5.3 Způsob sledování znaků

Zobrazovací jednotky s tekutými krystaly jsou pasivní zobrazovací prvky které neemitují světlo, ale používají světlo z jiných zdrojů. Z hlediska způsobu sledování můžeme LCD displeje rozdělit na tři skupiny:

- **Reflexní** – aby je bylo možné sledovat, vyžadují vnější osvětlení ze strany pozorovatele. Pro osvětlení je používáno světlo z okolního prostředí. Zobrazení je kvalitní venku za dne nebo v dobře osvětlených prostorách. Čelní optické záření je dopadající na displej, je k pozorovateli odraženo pomocí zrcadla, které je umístěno na zadní straně displeje.
- **Transmisní** – aby bylo možno sledovat zobrazovanou informaci, je nutné umístit zdroj osvětlení za displej. Aby bylo možno displej prosvítit, je zadní část LCD průhledná. Displeje pracují nejlépe při horším okolním osvětlení.

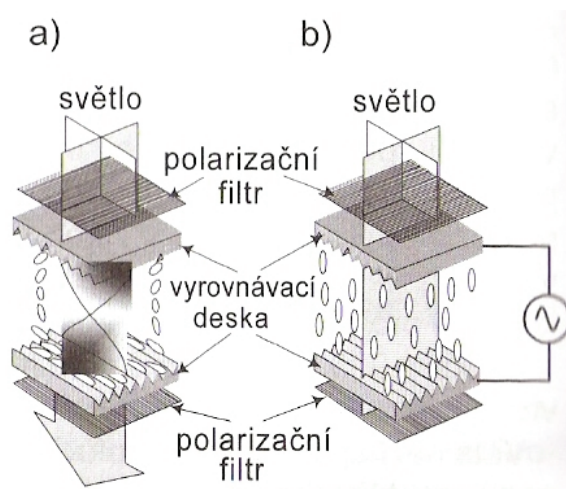
- **Transflexní** – je kombinaci obou předchozích metod. Zadní osvětlení může být dle podmínek zapnuto nebo vypnuto. Při dobrých světelných podmínkách okolí se tak snižuje spotřeba. [1]

2.5.4 TN (Twist Nematic)

Jestliže jsou tekuté krystaly vloženy mezi dvě drážkované elektrody, jejich drážky jsou vzájemně pootočené o 90° , v blízkosti drážek se krystaly nasměrují jejich směrem. Krystaly jsou v tomto případě v pootočeném (twistovém) uspořádání. [1]

2.5.5 Princip zobrazovací jednotky LCD

Zobrazovací jednotky se vytvářejí ve tvaru dvou rovnoběžných skleněných destiček. Mezi destičkami je uzavřena kapalina v nematické fázi. Na vnitřní přední straně jsou napařeny elektrody ve tvaru bodu nebo znaku. Na zadní straně je společná elektroda. Na vnějších stranách skel jsou nanášeny polarizační filtry, které polarizují procházející světlo, vůči sobě jsou pootočené o 90° . Pokud není přiloženo napětí, tak světlo prochází, protože je polarizováno filtry i krystaly (Obr.4a). Když přiložíme na oddělenou elektrodu napětí, většinou střídavé dá se tak předejít parazitním jevům (například elektrolyza), tak dojde k vertikálnímu natočení krystalů. Nyní jsou krystaly orientovány stejným směrem kolmo na elektrody a proto nedojde k natočení polarizované vlny. Protože je světlo polarizováno pouze prvním filtrem, který je proti druhému pootočen. Tak světlo nebude procházet (Obr.4b).



Obr. 9. Průchod světla skrz LCD

2.5.6 Zdokonalené principy LCD

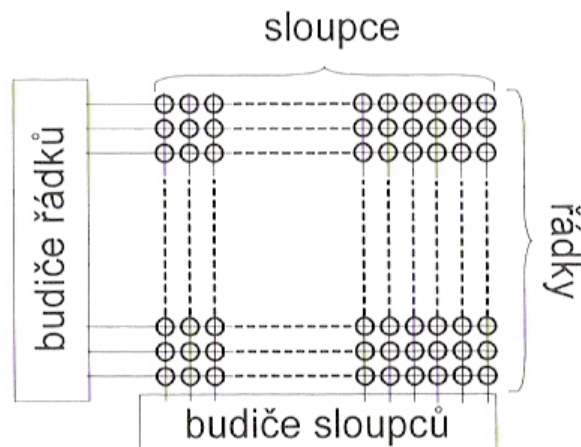
Ve snaze zlepšit kontrast a další optické vlastnosti byly vyvinuty další typy jednotek:

- **HTN (Hyper Twist Nematic)** – U tohoto typu je zvětšeno natočení krystalů z 90° na 110° . Displej vykazuje lehce žlutozelené nebo nazelenale šedé pozadí a téměř černé zobrazení segmentů. Hlavní nevýhodou tohoto uspořádání je zhoršení časové odezvy a tím i zamezení použití v prostředí s velmi nízkými teplotami. Vzhledem k dobrému poměru cena / technické vlastnosti je displej HTN výhodný pro aplikace v telekomunikacích a v přístrojové technice.
- **STN (Super Twist Nematic)** – struktura STN je založena na zvětšení úhlu natočení na hodnotu větší než 200° zpravidla 240° . To umožňuje další zvýšení kontrastu zobrazení a zvětšení pozorovacího úhlu. Nevýhodou je dvojlom paprsků, který zapříčiňuje změnu barvy pozadí na žlutozelenou a barvu znaků na tmavě modrou, případně na tmavě zelenou. Barva pozadí může být změněna na šedivou použitím speciálního filtru. Tento displej má o něco vyšší cenu. Je používán v náročnějších provedeních displejů.
- **DSTN (Double Super Twist Nematic)** – Problémy s barvami typu STN řeší struktura DSTN, která je založena na dvouvrstvé struktuře STN. Není-li na vývodu určitého bodu přivedeno napětí, je fázový posun první vrstvy kompenzován druhou vrstvou. Bod má černou barvu. Po přivedení napětí není činnost druhou vrstvou ovlivňována a zobrazení bude mít bílou barvu. Výsledkem je další zvýšení kontrastu. Protože jsou obě vrstvy ze stejného materiálu, je displej stabilní v širokém rozsahu teplot.
- **TSTN (Triple Super Twist Nematic)** – Technologie TSTN používá pro zobrazení jednoho bodu trojici buněk STN. To umožňuje v grafických displejích zvýšit počtu úrovní šedé, případně konstrukci barevných obrazovek, kdy každá buňka STN v trojici tvoří jeden zobrazovaný bod odpovídá jedné z barev RGB.
- **FSTN (Film Compensated Super Twist Nematic)** – Struktura FSTN přidává ke struktuře STN optické zpoždovací fólie, které mohou být umístěny z jedné nebo obou stran displeje mezi sklem a polarizačním filtrem. Účelem fólie je kompenzace účinku dvojlomu, který se projevuje v zobrazení změnou barev. To umožňuje

výrobu černobílých displejů s vysokým kontrastem a velkým pozorovacím úhlem.[1]

2.6 Multiplexní buzení maticových zobrazovacích jednotek

Maticové displeje jsou tvořeny soustavou bodů uspořádaných do mřížky. Mřížka bodů je uspořádána do řádků a sloupců, jako na obrázku (Obr.5), kde jsou jednotlivé body znázorněny kolečky.



Obr. 10. Multiplexní buzení maticového displeje

Pro zobrazení určitého bodu se musí vybrat příslušný řádek a sloupec. Budič řádků postupně připojuje jednotlivé řádky k napájecímu napětí. Budič sloupců připojuje sloupce k napětí, podle toho které body v řádku mají být zobrazeny. Tento proces se opakuje dokud se nedosáhne posledního řádku, poté celý proces začíná opět od prvního řádku. Budiče řádků a sloupců jsou ovládány pomocí mikro počítače případně, integrovanou řídicí logikou.

3 RASTERIZACE JEDNODUCHÝCH OBJEKTŮ

Rasterizací se nazývá transformace základních grafických objektů(úsečka, kružnice...) do posloupnosti obrazových bodů. Při práci se používají výstupní zařízení, které většinou pracují s rastrem. Základní prvkem rastru je bod, ze kterého jsou sestaveny všechny grafické prvky.

Při rasterizaci objektů vzniká určitá chyba, která vzniká skutečností, že bod je nejmenší zobrazovací jednotka a souřadnice zobrazovaných bodů, které nemají celočíselnou hodnotu, se musí zaokrouhlovat.[10]

3.1 Úsečka

Úsečka patří mezi základní grafické prvky. V analytické geometrii obsahuje úsečka nekonečně mnoho bodů, na rastrovém monitoru se skládá z konečného počtu bodů. Naproti tomu, že je poměrně jednoduchá, její konstrukci se většinou věnuje velká pozornost. Pomocí úseček se totiž vykreslují složitější grafické objekty, a proto je naší snahou vykreslovat úsečku co nejefektivněji.

Nejčastěji se úsečka zadává pomocí svých koncových bodů $[x_1, y_1]$ a $[x_2, y_2]$, případně jedním bodem $[x_1, y_1]$ a vektorem rozdílů souřadnic $(\Delta x, \Delta y) = (x_2 - x_1, y_2 - y_1)$.

Analytické vyjádření přímky, která není rovnoběžná s osou y , vyjadřujeme v tvaru

$$y = mx + q \quad (1)$$

($m, q \in \mathbb{R}$), kde m je směrnice přímky a q je posun na ose y . Koncové body úsečky určují přímku s parametry m a q :

$$m = \frac{(y_2 - y_1)}{(x_2 - x_1)} \quad (2)$$

$$q = \frac{(x_2 \cdot y_1 - x_1 \cdot y_2)}{(x_2 - x_1)} \quad (3)$$

Rastrový rozklad úsečky je založený na vzorkování s konstantním krokem podle osy x a nebo y . Závisí to na sklonu úsečky reprezentovaného směrnici m .

Pokud je $|m| < 1$, tak úsečka má sklon k ose x menší jak 45 stupňů, a proto vzorkujeme podle osy x s krokem 1. Jestliže je $|m| > 1$, vzorkujeme podle osy y . Úsečku se směrnici 1

nazýváme diagonálou a vzorkujeme ji podle libovolné osy. Osu, podle které vzorkujeme, nazýváme řídicí (hlavní) osou. Druhou osu nazveme vedlejší osou.[8]

3.1.1 DDA algoritmus

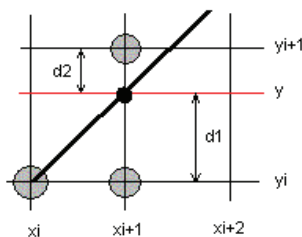
DDA (Digital Differential Analyzer) algoritmus je znám také jako jednoduchý přírůstkový algoritmus. Tento algoritmus je založený na pravidelném přírůstku Δx pro souřadnici x a Δy pro souřadnici y , při čem začíná v počátečním bodě úsečky. Vzhledem k tomu, že vzorkujeme v rastru, bývá přírůstek na řídicí ose, v případě, že úsečka má sklon bližší k ose x je řídicí osa x , rovný jednomu bodu.

V jednom cyklu budeme postupně připočítávat k souřadnici na hlavní ose jednotku a k souřadnici na vedlejší ose vypočítaný přírůstek. Ten má v případě, že hlavní osou je x , hodnotu směrnice m , v případě osy y $1/m$. Tyto hodnoty jsou reálné a při vykreslování je zaokrouhluje. DDA algoritmus při výpočtu nových souřadnic využívá hodnoty předcházejícího bodu. Tento typ algoritmu se označuje jako iterační.[8]

3.1.2 Bresenhamův algoritmus

V DDA algoritmu jsme používali reálná čísla, která jsme potom zaokrouhlovali. V následujícím algoritmu budeme využívat pouze celočíselné operace. Celý algoritmus spočívá v hledání nejbližších bodů ke skutečné úsečce pouze pomocí celočíselné aritmetiky.

Představme si část rastru, kde již byl nakreslen pixel o souřadnicích $[x_i, y_i]$. Dva další možné pixely leží na pozicích $[x_{i+1}, y_i]$. Rozdíly mezi souřadnicí y středů uvedených pixelů a skutečnou souřadnicí y na úsečce v bodě x_{i+1} jsou na obrázku (Obr.14) označeny jako d_1 a d_2 . [8]



Obr. 11. Výběr pixelů úsečky

Vyjdeme-li z obecné rovnice přímky $y = mx + b$, kde m je směrnice a b posun na ose y , získáme souřadnici y dosazením jako $y = m(x_i+1) + b$. Pak lze odvodit

$$d_1 = y - y_i = m(x_i + 1) + b - y_i \quad (4)$$

$$d_2 = y_i + 1 - y = y_i + 1 - m(x_i + 1) - b \quad (5)$$

Vyjádříme rozdíl těchto dvou vzdáleností jako:

$$\Delta d = d_1 - d_2 = 2m(x_i + 1) - 2y_i + 2b - 1 \quad (6)$$

Podle proměnné Δd můžeme určit, který ze dvou možných pixelů leží blíže skutečné úsečce. Záporná hodnota Δd znamená, že blíže leží pixel o souřadnici y , kladná hodnota určuje, že bližší pixel má souřadnici y_i+1 . Pro stanovení, který bod se má vykreslit, není důležitá skutečná hodnota Δd , ale její znaménko.[8]

3.2 Kružnice

Kružnice je nejčastěji definována středem v bodě $[x_c, y_c]$ a poloměrem r nebo souřadnicemi tří různých bodů.

Při kresbě kružnice v rastru můžeme využít již existujících metod pro kresbu úsečky a nahradit kružnici lomenou čarou. Tento nepřesný, ale rychlý postup, je doporučován při interaktivním zadávání kružnice. Přesná výsledná kresba pak pracuje s jednotlivými pixely kružnice. Vyjádříme – li kružnici v parametrickém tvaru

$$\begin{aligned} x &= x_c + r \cos \alpha \\ y &= y_c + r \sin \alpha \end{aligned} \quad (7)$$

můžeme jedním cyklem parametru α ($\alpha \in \langle 0, 2\pi \rangle$) získat posloupnost bodů pravidelně rozmístěných na kružnici. Tyto body nahrazují lomenou čarou kružnici. Velikost přírůstku úhlu alfa by měla odpovídat velikosti kružnice. Větší kružnice je vhodné nahrazovat větším počtem úseček. Tento způsob získávání bodů na kružnici je pomalý, jelikož pro každý bod je nutné vyhodnotit dvě goniometrické funkce. Lepší je využití transformací, při níž jsou souřadnice jednoho bodu na kružnici opakovaně podrobovány otočení kolem středu kružnice. Výpočet $\sin \alpha$ a $\cos \alpha$, nutných k transformaci otáčení, stačí provést pouze jednou. V algoritmu se potom používají pouze násobení a sčítání.[9]

Nevýhodou tohoto algoritmu je jeho iteračnost, protože se chyba vzniklá výpočtem kumuluje a u velkých kružnic se pak poslední bod kružnice nemusí setkat s prvním. Místo kružnice pak vznikne spirála.[10]

3.2.1 Bresenhamův algoritmus

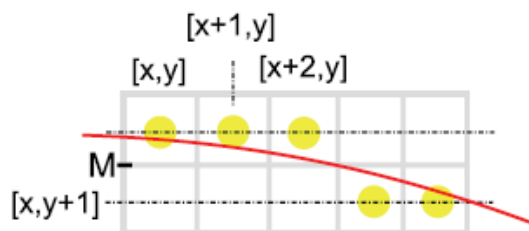
Víme, že kružnice se dá rozdělit na 360 stupňů. Při pohledu na kružnici můžeme vidět, že kružnice je symetrická vzhledem k ose x , ose y a také k osám kvadrantů (funkce $x=y$ a $x=-y$). Pro kružnici se středem v počátku souřadnicové soustavy platí, že pokud bod (x, y) je bodem kružnice, potom také (y, x) , $(y, -x)$, $(x, -y)$, $(-x, -y)$, $(-y, -x)$, $(-y, x)$ a $(-x, y)$ jsou body kružnice. Proto nám na určení všech bodů kružnice stačí vypočítat jen 1/8 kružnice, tj. oblouk v rozsahu 0 až 45°. Každý vypočítaný pixel potom vykresluje pro každý oktant rastrové roviny.[9]

Podobně jako u Bresenhamova algoritmu pro rasterizaci přímky se jedná o celočíselný algoritmus.

Proměnné u tohoto algoritmu nabývají na počátku hodnot: $D_0 = 1.25 - r$ (případně $D_0 = 1-r$). Algoritmus rozhoduje vykreslení bodu dle znaménka výsledku D_n , který určuje, zda bod leží pod středem či nad středem M viz. obrázek (Obr.15). Pokud platí, že $D_n < 0$, je výpočet veden podle vzorce (8). Pokud tomu tak není a platí tedy $D_n > 0$ jsou použity vzorce (9).[11]

$$\begin{aligned} D_{n+1} &= D_n + 2x + 3 \\ y_{n+1} &= y_n \\ x_{n+1} &= x_n + 1 \end{aligned} \quad (8)$$

$$\begin{aligned} D_{n+1} &= D_n + 2x - 2y + 3 \\ y_{n+1} &= y_n - 1 \\ x_{n+1} &= x_n + 1 \end{aligned} \quad (9)$$



Obr. 12. Část rastru kružnice

4 EAGLE

Editor plošných spojů EAGLE je uživatelsky přívětivý a výkonný nástroj pro návrh desek plošných spojů (DPS, PCB). Název EAGLE je zkratka, pocházející z původního názvu Easily Applicable Graphical Layout Editor. Skládá se ze tří modulů a to spojů, schématu a autorouteru. Podporuje platformy Windows, Linux a MAC. Je dostupný ve třech verzích.

Light verze, která je pro nekomerční využití zdarma. Má omezení desky plošného spoje 8 x 10 cm, na pouze dvě signálové vrstvy a jeden list schématu.

Standart odlehčená profesionální verze, je omezena na rozměr desky plošného spoje 10 x 16 cm, 4 signálové vrstvy, 99 listů schématu.

Professional nejvyšší verze programu EAGLE je omezena na maximální rozměr desky 1,6 x 1,6 m, 16 signálových vrstev, 99 listů schématu.

4.1 Control Panel

Po spuštění programu se zobrazí Control Panel. Tento panel nám umožní provést základní nastavení a zakládat případně otevírat projekty a jejich součásti.

Na obrázku (Obr.6.) můžeme vidět několik základních složek. Nás budou zajímat pouze některé a to libraries, project, scripts. Kde jsou uloženy hlavní součásti programu.

Libraries – zde jsou uloženy všechny knihovny programu. Můžeme nastavit, které knihovny součástek chceme použít případně je otevřít pro úpravu.

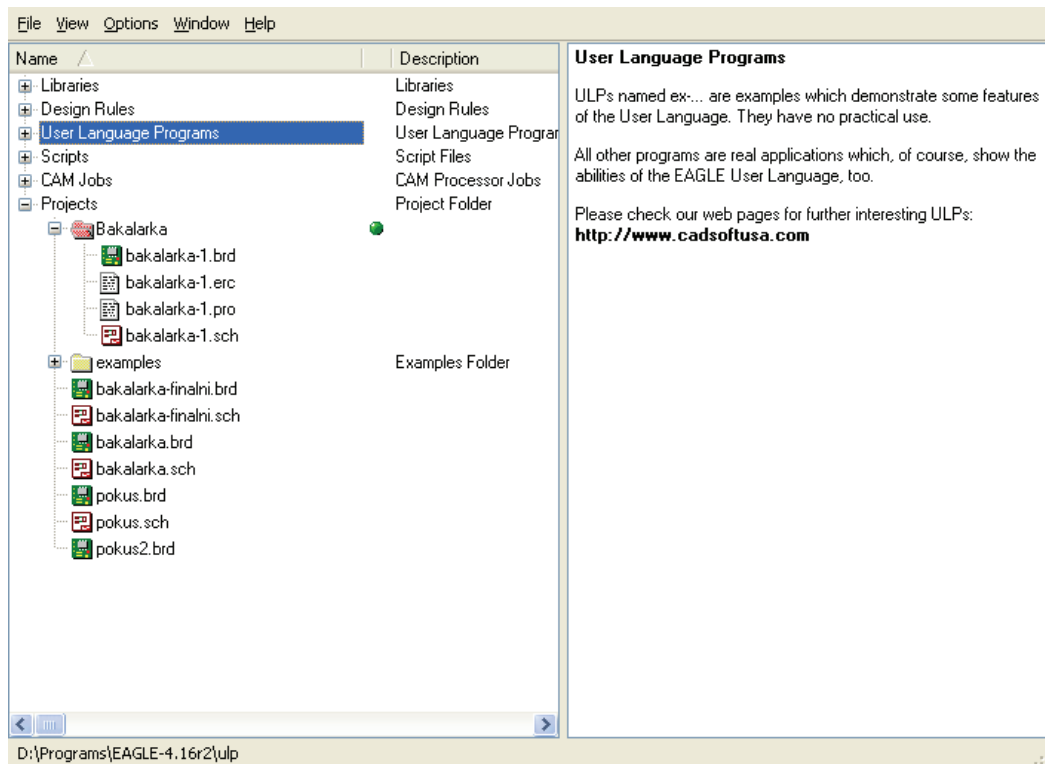
Scripts – tady jsou skripty hlavního nastavení programu. Změnu nastavení můžeme provést pomocí rozhraní programu nebo přímo v těchto skriptech.

Projects – pro nás asi nejpoužívanější složka. Zde jsou uloženy projekty, ale i jednotlivé soubory.

Dále nás budou zajímat dvě položky menu Options a File.

Options – nastavení programu pomocí uživatelsky přívětivějšího rozhraní, než přímého zápisu do skriptu. Jsou zde tři položky directories, backup a user interface. Nabídka directories nastaví složky do kterých se mají automaticky ukládat jednotlivé druhy souborů. Backup nastavení automatického ukládání a počtu záloh. User interface slouží k nastavení uživatelského rozhraní.

File – slouží k otevírání ukládání a vytváření souborů a projektů.



Obr. 13. Control Panel

4.1.1 Založení projektu

Projekt se zakládá v kontrolním panelu a to dvěma způsoby. První je přes hlavní menu a druhý přes složku project přímo.

V hlavním menu vybereme Files a dále pokračujeme na new project. Vytvoří se nám prázdný projekt z názvem New_Project_1. To tohoto projektu musíme dále přidat schéma obdobným způsobem, jako jsem vytvořili projekt.

Založení projektu přes složku projekt je obdobné, akorát klikneme pravým tlačítkem myši na složku projects a v nabídce vybereme new project. Obdobným způsobem přidáme schéma.

Pokud máme vytvořen projekt můžeme přistoupit k tvorbě schématu.

4.2 Editor schémat

Jak už jeho název napovídá slouží pro tvorbu schémat. Poté se nám objeví editor schémat, ve kterém budeme navrhovat potřebná schéma.

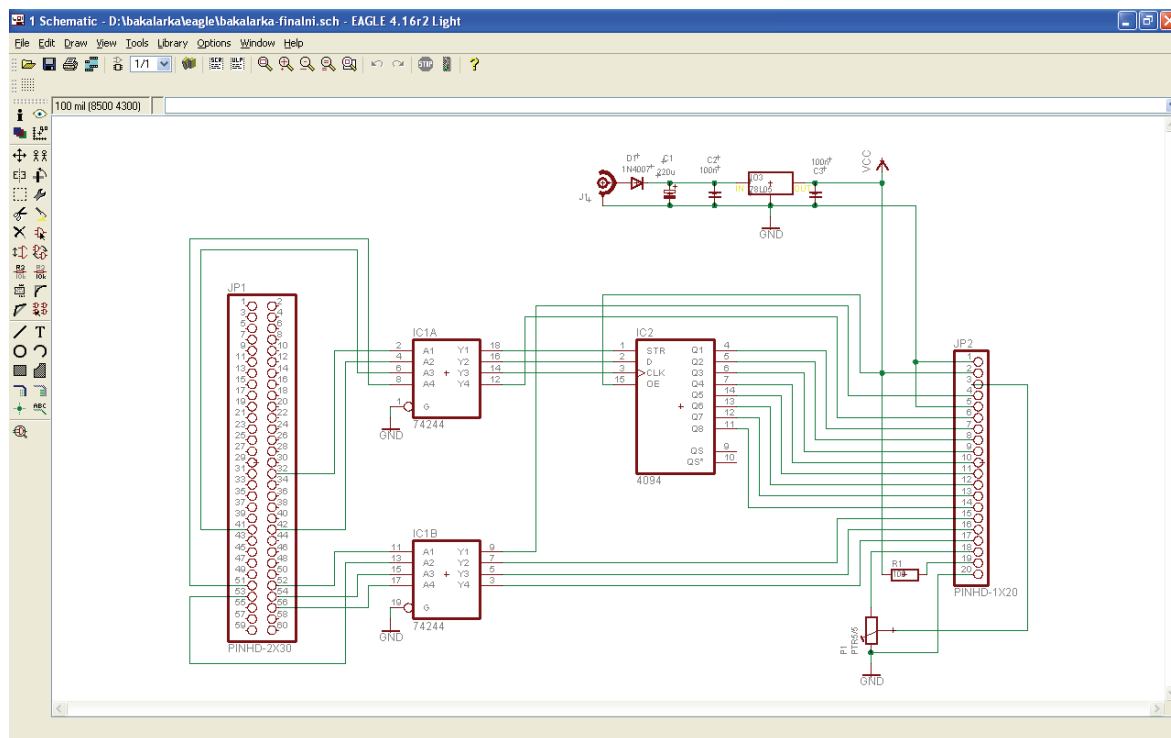
Jako první můžeme nastavit mřížku, můžeme nechat i původní, ale ta je poměrně velká a při velkém počtu součástek a propojovacích čar, bude schéma zbytečně velké. Nastavení se provede pomocí ikony Grid. Otevře se nabídka, kde můžeme zvolit zda chceme mřížku zobrazit, v jaké formě (body, mřížka) a nakonec velikost mřížky.

Nyní můžeme začít s tvorbou schématu. Součástky jsou uloženy v knihovnách. Nabídku knihoven lze vyvolat ikonou ADD. Zobrazí se nabídka se všemi aktivními knihovnamí. Knihovny jsou poměrně obsáhlé, nalezení vhodné součástky zabere určitý čas, musíme volit pečlivě i typ pouzdra součástky, protože potom bude využita při tvorbě desky plošného spoje. V případě, že jsme našli potřebnou součástku vybereme ji. Poté ji vhodně umístíme a opakujeme dokud nebudou položeny všechny součástky.

Propojení součástek se provede pomocí ikony NET. Tím se dostaneme do režimu kreslení elektrických spojů. Začíná se kreslit od kliknutí na vývod jedné součástky a končí se dvoj klikem na vývod druhé součástky. A tak pokračujeme do propojení celého schématu.

Nesmíme zapomenout také na označení elektrické země GND a napájení VCC. Je to důležité, protože napájení integrovaných obvodů je ve schématu skryto a pokud budeme chtít při tvorbě plošných spojů využít Autorouter, tak se toto propojení musí provést.

Chyby ve schématu si můžeme zkontrolovat v erc záznamu. Dá se vyvolat kliknutím na ikonu Erc. Pokud je tam napsáno 0 errors a 0 warnings. Můžeme přikročit k návrhu desky plošných spojů.



Obr. 14. Editor schémat

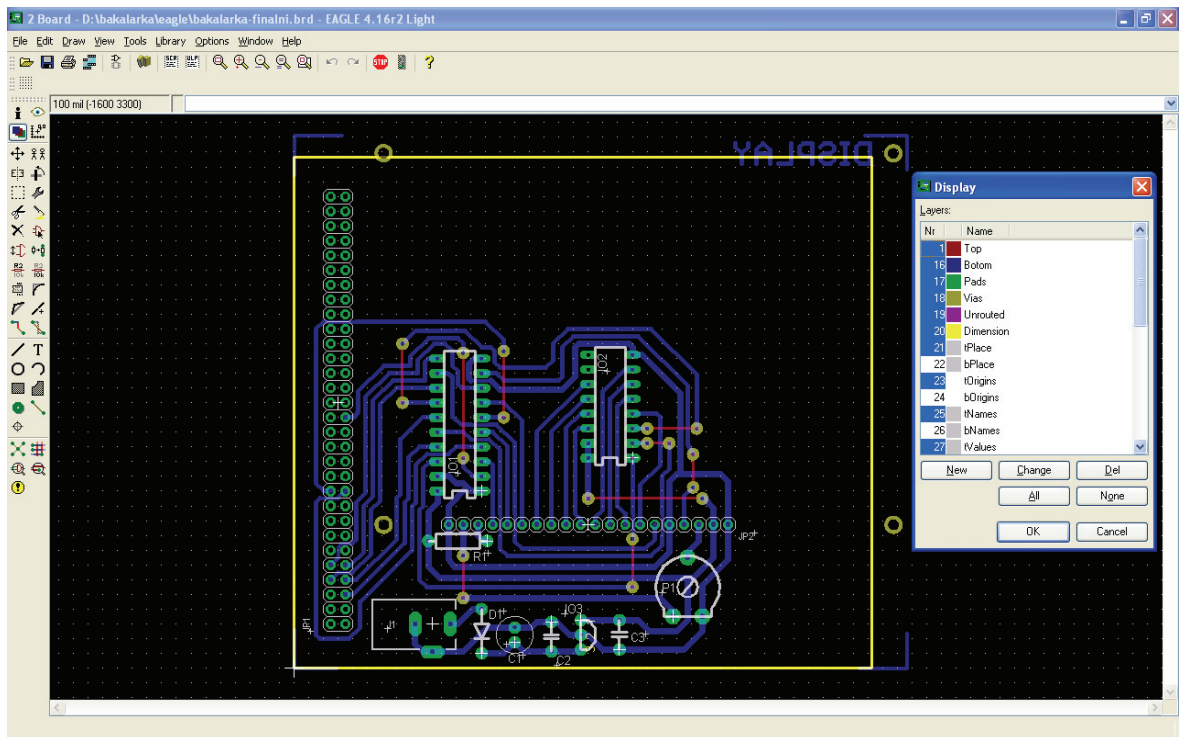
4.3 Editor návrhu DPS

Po aktivaci ikony BOARD ve schématu se automaticky přepneme do editoru návrhu DPS. Po přepnutí se nám zobrazí napravo všechny pouzdra součástek a na levé straně máme ohraničenou plochu (verze light 8 x 10 cm), kde musíme rozmístit součástky. Piny součástek jsou propojeny vzdušnými spoji.

Rozmístění součástek se provádí pomocí ikony Move, můžeme také přemísťovat celé bloky, součástky se taky dají otáčet pomocí pravého tlačítka myši. Rozmístění je zdlouhavá činnost, která se odvíjí, od našich zkušeností a citu, dá se pouze doporučit sledovat vzdušné spoje a umísťovat tak dlouho, dokud se budou co nejméně křížit.

Propojení se provádí pomocí ikony Route, která transformuje vzdušný spoj na spoj elektrický. Ve verzi light máme k dispozici pouze dvě vrstvy a to BOTTOM (zadní modrá barva) a vrstvu TOP (přední červená barva). Spoj můžeme různě ohýbat pravým tlačítkem myši a měnit jeho tloušťku v horní liště. Doporučuje se dodržovat úhel čar 90° a 45°. Jednovrstvá deska se dělá v případě obyčejných součástek ve vrstvě BOTTOM. Pokud používáme součástky SMD využívá se strana TOP.

Pokud chceme již vytvořený spoj zrušit nemaže se příkazem Delete, ale transformuje se zpátky na vzdušný spoj ikonou Ripup.



Obr. 15. Editor DPS

Další možnost propojení po rozmístění součástek je použít Autorouter. Je integrován do editoru a má ikonu Auto. Je konfigurovatelný uživatelem pomocí penalizačních faktorů. Po kliknutí na ikonu Auto se nám zobrazí nabídka, kde máme možnost nastavení jednotlivých penalizací. Po spuštění nám Autorouter vygeneruje ze vzdušných spojů, spoje elektrické. Dále můžeme spoje upravit, případně zrušit elektrické spoje, přemístit součástky a celý proces opakovat.

Po úspěšném propojení všech součástek ještě pojmenujeme desku pomocí ikony text. Používá se to proto, aby nedošlo k otočení vytištěné předlohy při tvorbě desky. Pokud tedy máme dvouvrstvou desku musíme text přenést do obou vrstev. Ve vrstvě BOTTOM bude obrácená.

II. PRAKTICKÁ ČÁST

5 NÁVRH MODULU

Cílem této bakalářské práce je navrhnout a vyrobit modul s LCD grafickým displejem řízený mikropočítačem, který bude sloužit jako výuková pomůcka pro předmět Mikropočítače.

5.1 LCD modul ATM12864D

LCD modul ATM12864D je grafický displej rozlišením 128x64 bodů, se zabudovaným podsvícením. [12]

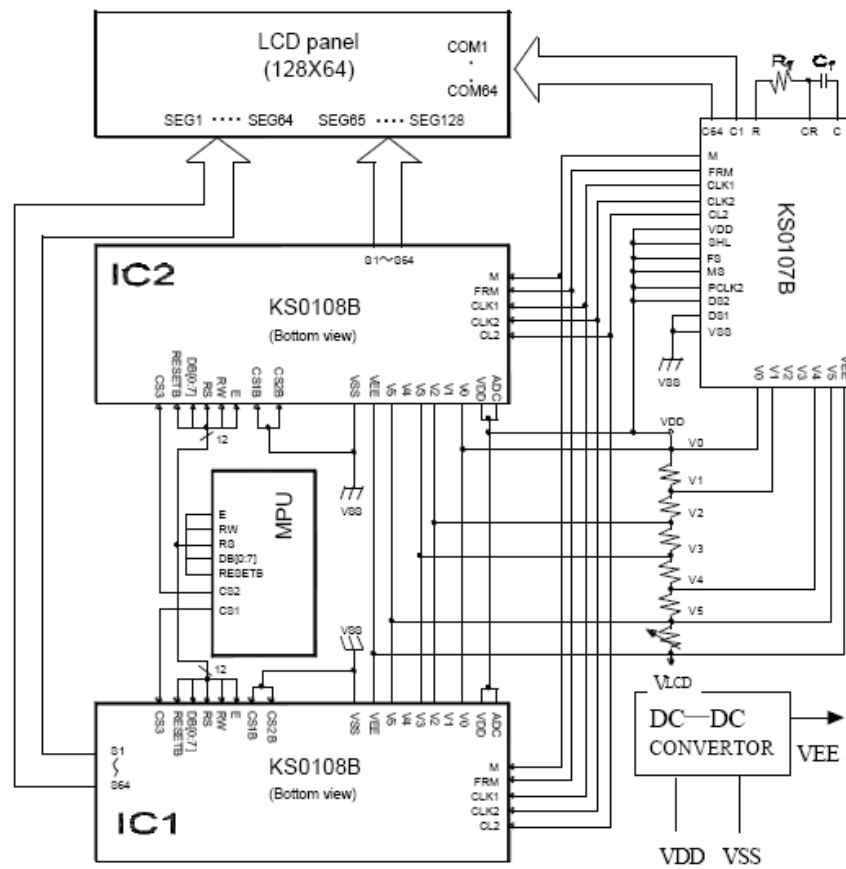
5.1.1 Vlastnosti

- Formát displeje 128x64 maticový grafický
- Velikost bodu 0,48x0,48 mm
- Velikost zobrazovací oblasti 72.0x40.0 mm
- STN žluto zelený
- Rozhraní pro 8bitový mikropočítač
- Nízkou spotřebu
- LED podsvícení
- Pozorovací úhel 6 hodin
- LCD radiče KS0108B(2) KS0107B
- Velikost modulu 93.0x70.0x15.0 mm
- Napájecí napětí 4,5 - 5,5V
- Provozní teplota -20 – 60 °C

Tab. 1. Obsazení vývodů displeje

| Číslo vývodu | Signál | Funkce |
|--------------|-----------------|-------------------------|
| 1 | GND | Zem (0 V) |
| 2 | U _{CC} | Napájecí napětí |
| 3 | U ₀ | Nastavení kontrastu |
| 4 | RS | Příkaz(0), Data(1) |
| 5 | R/W | Čtení(1), Zápis(0) |
| 6 | E | Strobovací signál |
| 7 | DB0 | Data/Příkaz |
| 8 | DB1 | Data/Příkaz |
| 9 | DB2 | Data/Příkaz |
| 10 | DB3 | Data/Příkaz |
| 11 | DB4 | Data/Příkaz |
| 12 | DB5 | Data/Příkaz |
| 13 | DB6 | Data/Příkaz |
| 14 | DB7 | Data/Příkaz |
| 15 | CS1 | Výběr 1. řadiče |
| 16 | CS2 | Výběr 2. řadiče |
| 17 | RST | Reset modulu |
| 18 | U _{EE} | Výstup záporného napětí |
| 19 | A | Anoda podsvícení |
| 20 | K | Katoda podsvícení |

Blokové schéma:



Obr. 16. Blokové schéma modulu

5.1.2 Konstrukce displeje

Displej obsluhují dva řadiče KS0108B a jeden řadič KS0107B viz blokové schéma (Obr.16).

KS0107B má 64 posuvných registrů a 64 kanálový výstup. Slouží pro generování hodinového signálu pro řadiče KS0108B.

KS0108B je LCD řadič se 64 kanálovým výstupem pro maticové LCD zobrazovací systémy. Tato součástka se skládá ze zobrazovací RAM paměti, 64 bitového registru a dekodovací logiky. Vnitřní RAM paměť slouží k ukládání dat a generování bodové matice pro LCD displej, která koresponduje s uloženými daty.

Celý displej je rozdělen na 2 poloviny s rozlišením 64x64 pixelů. Každá z nich má svůj vlastní řadič. Pomocí vývodů CS1 a CS2 se volí práce buď s levým, pravým nebo oběma řadiči (aktivní v log.1). Displej komunikuje po 8-bitové paralelní sběrnici. Volba mezi zápisem a čtením se provádí pomocí vývodu RW a volba mezi instrukcí a daty pomocí vývodu RS. Záporným pulzem na vývodu RST se LCD vypne a vynulují se interní registry. Obsah obrazové paměti se nezmění, takže po zapnutí LCD instrukcí se objeví původní obraz.

Při zápisu instrukce nebo dat je byte ze sběrnice načten při sestupné hraně kladného impulzu na vývodu E, který by měl trvat alespoň 450ns. Data musí být na sběrnici přivedeny ještě před příchodem vzestupné hrany na E a ne těsně před sestupnou hranou, jinak při některých konkrétních kombinacích na datové sběrnici dojde k zápisu náhodného bytu do té části LCD, která není vybraná. Po zapnutí napájení není možné ihned vysílat instrukce. Je třeba generovat časovou prodlevu.

5.1.3 Instrukce pro řízení Displeje

Instrukce pro řízení ovládají řadič KS0108B. Tyto instrukce jsou posílány přes datové vstupy

do řadiče. Následující tabulka (Tab.2.) zobrazuje různé příkazy a nastavení příznakových bitů. [7]

Tab. 2. Instrukce displeje

| Instrukce | RS | R/W | DB7 | DB6 | DB5 | DB4 | DB3 | DB2 | DB1 | DB0 |
|-------------------|----|-----|-------------------------|-----|------------------------------|-----|-----|----------|-----|-----|
| Čtení dat | 1 | 1 | Čtení zobrazovaných dat | | | | | | | |
| Zápis dat | | 0 | Zápis zobrazovaných dat | | | | | | | |
| Čtení stavu | 0 | 1 | B | 0 | OFF | RST | 0 | 0 | 0 | 0 |
| Nastavení Y adres | | 0 | 0 | 1 | Y adresa | | | | | |
| Nastavení řádku | | 0 | 1 | 1 | Nastavení startovacího řádku | | | | | |
| Nastavení X adres | | 0 | 1 | 0 | 1 | 1 | 1 | X adresa | | |
| Displej Zap/Vyp | | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | ON |

Zapnutí/Vypnutí displeje

Data se zobrazí pokud je bit ON nastaven na 1 a zmizí pokud je 0. Ačkoliv nejdou data zobrazena, jsou neustále uchovány v DDRAM. Takže když se displej opět zapne data se opět zobrazí.

Nastavení adresy Y

Adresa Y má délku 6 bitů, to znamená čísla 0-63. Po zaslání této adresy se nastaví příslušná adresa Y. Po přečtení nebo zapsání dat do řadiče se adresa automaticky zvýší o jedničku. Pokud se provede operace na adrese 63, tak dojde k přetečení na 0.

Nastavení adresy X

Adresa X má délku pouze 3 bity, teda rozsah 0-7. Čtení a zápis se budou provádět na příslušném řádku, dokud se nenastaví nová adresa.

Nastavení startovacího řádku

Nastaví registr startovacího řádku na rozsah 0-63. Umožňuje rolování řádků displeje

Čtení stavu

- **Busy (B)** pokud je nastaven na 1, tak řidič je zaneprázdněn a nepřijímá žádné vnější podněty ke zpracování. Když je opět na 0 můžeme pokračovat v posílání dat.
- **ON/OFF (OFF)** když je tento bit na jedničce znamená to, že displej je zapnut. Nula znamená že je vypnut.
- **RESET (RST)** jednička znamená, že displej je právě inicializován. Žádná instrukce nebude akceptována kromě čtení statusu. Když je nulový, displej byl inicializován a je připraven přijímat data. [12]

Zápis zobrazovaných dat

Do DDRAM se zapíší data z datových vstupů DB7-DB0. A adresa Y se zvýší o 1.

Čtení zobrazovaných dat

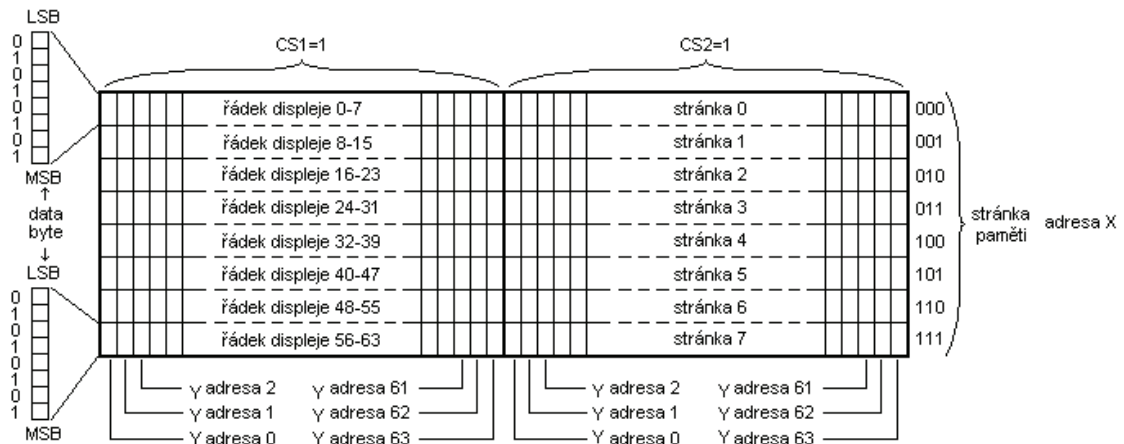
Z DDRAM se přečtou data z datových vstupů DB7-DB0. A adresa Y se zvýší o 1.

5.1.4 DDRAM (Display Data RAM)

Jak už bylo dříve uvedeno v kapitole 5.1.2, displej je rozdělen na dvě poloviny. To znamená, že i DDRAM je uložena na dvě poloviny každá v jednom řadiči.

Příslušný sloupec nastavíme pomocí registru Y. Registr Y je schopen na adresovat 64 sloupců. Takže jsme schopni adresovat jednu polovinu displeje (tedy 0 - 63), abychom byli schopni obsluhovat i druhou polovinu (64-127), tak se musíme přepínat mezi řadiči pomocí CS1, CS2.

Registr X nastavuje řádkovou stránku. To znamená, že nemůžeme adresovat třeba řádek 43 přímo, ale po stránkách. Jedna stránka má 8 bitů tedy jeden bajt. Jak je vidět na obrázku (Obr.17.), tak musíme nastavit adresu X na pátou stránku a provést operaci s celým bajtem.



Obr. 17. Rozložení DDRAM displeje

5.2 Návrh schématu

Modul s grafickým displejem se skládá ze dvou základních částí:

- napájení
- řídicí logika

5.2.1 Napájení

Napájení je realizováno externím zdrojem, protože napájecí napětí přímo na vývojovém kitu je nedostatečné. Vývojový kit dodává jenom 3,3V a maximálně 50 mA, pro napájení logiky displeje a proudový odběr podsvícení, je to málo.

Napájení je realizováno externím zdrojem (9V /1100mA) a je odebíráno ze svorkovnice TB1 umístěné přímo na vývojovém kitu.

Vstup je tvořen napájecím konektorem pro malá napětí J1 o průměru 2.1mm. Je chráněn proti přepólování diodou D1 typu 1N4007. Dále je vyhlazovací elektrolytický kondenzátor C1.

Napětí stabilizuje na 5V stabilizátor 78L05 v klasickém zapojení. To znamená s blokovacími keramickými kondenzátory C2 na vstupu a C3 na výstupu. Napájecí napětí stabilizátoru nesmí přesáhnou 20V a také nesmí klesnou pod 7V. Maximální proudový odběr je 100mA.

Na anodu displeje je přivedeno přes odpor +5V. Odpor zajišťuje redukování proudového odběru podsvícení.

Displej si sám generuje záporné napětí U_{EE} , které je důležité pro nastavení kontrastu(konektor JP2 – 18). Ten je přes odporový trimr napojen na vstup U_0 , který slouží k nastavení kontrastu displeje (JP2 – 3).

5.2.2 Řídící logika

Na modulu jsou dva integrované obvody a to budič 74HC244 a 8-bitový posuvný registr 4094.

Na konektor JP1, který je připojen do desky vývojového kitu, je napojen budič, protože displej pracuje na 5V logice, kdežto mikropočítač pracuje na 3,3V. Potom by mohlo docházet k neurčitým stavům na displeji. Zároveň plní funkci oddělení portů mikropočítače od zbytku modulu.

Vzhledem k tomu, že modul má být použit pro výuku, tak byl požadavek na zachování co nejvíce portů, které jsou používány pro ovládání periférii. Tak byl použit posuvný registr, pro komunikaci dat s displejem. Takže byly místo 8 portů pro data použity jenom 3 pro sériovou komunikaci s posuvným registrem.

Posuvný registr nebo-li SIPO (Serial In-Parallel Out) obvod. Má 4 vstupy, STR – zapisovací vstup, Data, CLK – hodinový signál a OE - nastavení vysoké impedance. Nastavení výstupu vysoké impedance nebudeme používat, tak jsme jej napojili na +5V.

Princip funkce posuvného registru. Na posuvný registr posíláme data, na náběžnou hranu hodinového signálu se zapíše do vnitřní paměti, na pozici Q_1 , při další náběžné hraně se přepíše hodnota Q_1 na pozici Q_2 a na Q_1 se nahrají aktuální data a tak pokračujeme dokud nebudeme mít první nahrané data na Q_8 . Data z paměti zobrazíme na výstupy $Q_1 - Q_8$ pomocí zapisovacího vstupu, na který pošleme kladný impuls.

Výstupy posuvného registru jsou napojeny na datové vstupy displeje. Zbytek portů je připojen na datové vstupy displeje podle tabulky (Tab.1.).

Tab.3. Propojení portů mikropočítače s modulem

| Číslo pinu | Název Portu | Funkce |
|------------|-------------|-----------|
| 56 | PTG3 | RES |
| 53 | PTG4 | CS1 |
| 54 | PTG5 | CS2 |
| 51 | PTG6 | E |
| 52 | PTG7 | R/S |
| 41 | PTF6 | CLK(4094) |
| 42 | PTF7 | D(4094) |
| 32 | PTE7 | STR(4094) |

5.3 Deska plošného spoje

5.3.1 Návrh DPS

Desku plošného spoje jsme navrhovali v program EAGLE Light. Jako první jsme si nakreslili schéma, podle postupu, který je popsán v kapitole 4.2.

Při tvorbě plošného spoje jsme museli mít na vědomí omezení veze Light, které je omezeno plochou rozmístění součástek. Vzhledem k tomu, že jsme použili místo součástky displeje pouze konektorový kolík, na který jsme poté umístili display, tak se nás omezení netýkalo. Na propojení jednotlivých součástek jsme, ale nemohli použít Autorouter, protože jsme byli omezeni pouze jednovrstvým plošným spojem. Autorouter nebyl toto zapojení schopen adekvátně propojit, tak jsme byli nuceni se uchýlit k ručnímu propojení. Ruční propojení má však i výhody, protože jsme, byli schopni optimalizovat rozmístění součástek a tím i počet propojek. Konečný rozměr desky byl 106x93 mm.

5.3.2 Výroba DPS

Jako první si vytiskneme předlohu na fólii nebo pauzovací papír laserovou tiskárnou. Vždy je třeba si ověřit, aby předloha nebyla vytištěna zrcadlově obráceně. Výsledný plošný spoj by byl, potom nepoužitelný. Poté si předlohu upravíme na velikost tištěného spoje. Pokud

máme velice tenké spoje, je lepší si koupit Cuprexit s již nanesenou fotocitlivou vrstvou. Vyhnete se tak nepravidelnému nanesení laku.

Poté desku opracujeme na potřebný rozměr, případně o několik milimetrů větší. Předlohu přiložíme na nalakovaný plošný spoj a upevníme je k sobě, nejlépe vložením mezi dvě skleněné desky. Necháme UV lampu rozehrát a poté dáme desku s předlohou osvitit. Osvícení trvá zhruba 5 minut, záleží na použité předloze a fotocitlivém laku. Na plošném spoji se to projeví tak, že osvícené plochy ztmavnou a plochy kde mají být vodivé cesty jsou světlejší.

Po osvícení sejmeme předlohu a desku omyjeme vodou. Desku vložíme do nádoby s vývojkou. Vyvolání se projeví tak, že osvícený lak se odplaví a zbude jenom na spojích.

Do nádoby nalijeme leptací roztok. Vyvolanou desku položíme na hladinu obrazcem dolů. Vyleptání trav zhruba půl hodiny, záleží na kvalitě vyvolání, stáří a síle leptacího roztoku. Během leptání pravidelně kontrolujeme průběh, případně odstraníme štětečkem již odleptané kusy mědi, které se přilepily.

Zbytky laku odstraníme ředidlem. Zkontrolujeme, zda na desce nejsou vodivé cesty, které tam nemají být. Pokud tam jsou, tak je odstraníme. Pro ochranu před korozí a snazší pájení potřebe vyleptané spoje tekutou kalafunou.

5.3.3 Osazení plošného spoje

Po leptání se ještě plošný spoj musí upravit na konečný rozměr. Dále byly vyvrtány otvory pro součástky. Pro lepší vodivost cest a taky ochranu před korozí byl plošný spoj pocínován. Pro pocínování se volí nižší teplota pájení zhruba 270°C, aby se cesty namáhaly co nejméně.

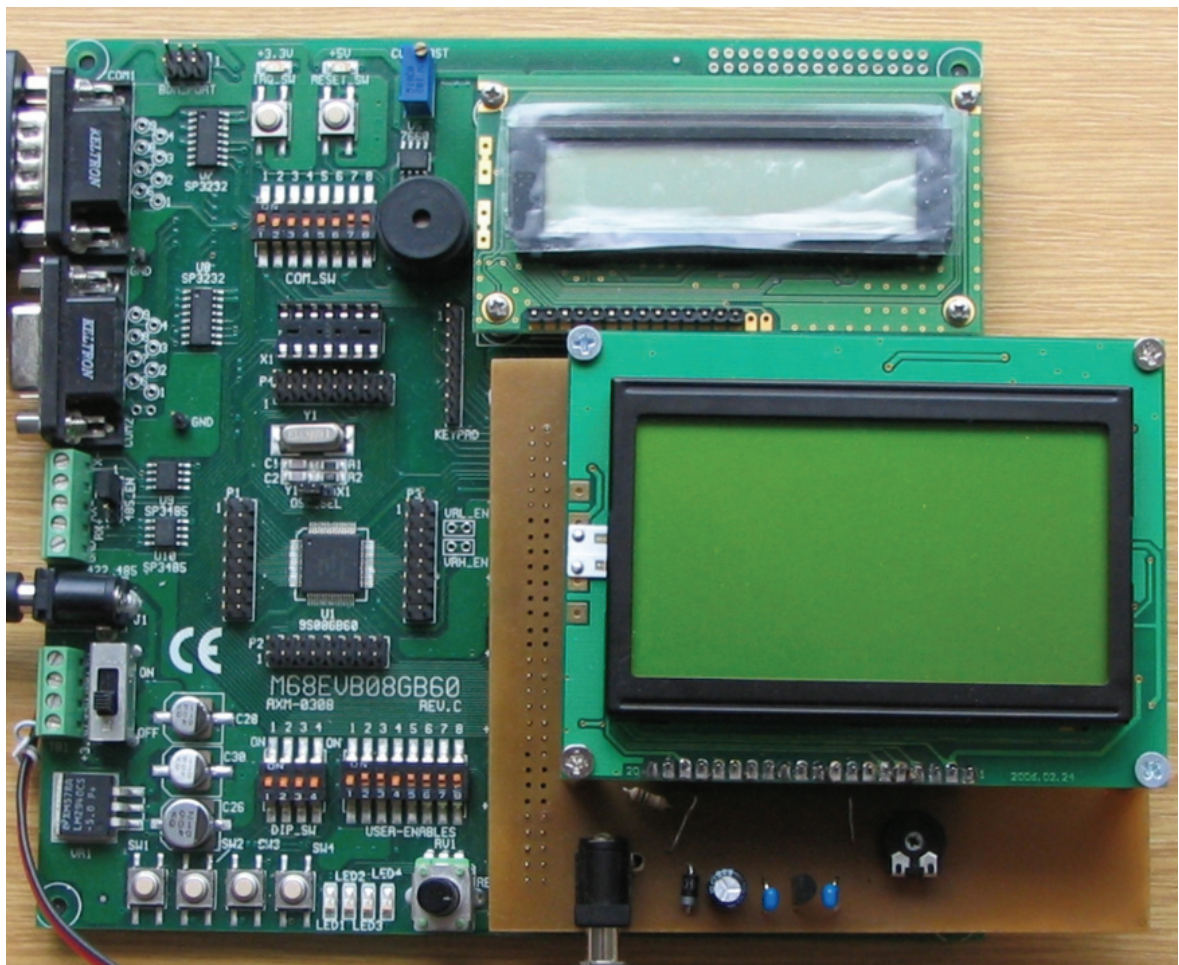
Součástky se osazují od těch nejodolnějších. První konektory, patice integrovaných obvodů, poté následují pasivní součástky odpory, kondenzátory. Jako poslední pájíme polovodičové prvky tranzistory, diody, integrované obvody. Samozřejmě v potaz pro pořadí se musí brát i dostupnost součástky, což znamená, že když máme mít pod paticí zapájen odpor, tak první musíme osadit odpor a až potom patici.

5.4 Propojení modulu a vývojového kitu

Pro propojení modulu s vývojovým kitem, byl použit MCU konektor. Na MCU konektoru jsou vyvedeny všechny porty mikropočítače. Porty mikropočítače jsou od modulu odděleny pomocí budiče 74HC244.

Pro přímou komunikaci s displejem je použito 5 portů. 3 porty jsou použity pro ovládání 8-bitového posuvného registru 4094, jehož datové výstupy jsou připojeny k datovým vstupům displeje. Bylo sice znemožněno čtení z displeje a tím potřeba ukládat obraz displeje do paměti mikropočítače, ale vzhledem k velké paměti RAM 4kB, to nepředstavuje příliš velký problém. Obraz displeje v paměti RAM zabírá 1kB.

Pro propojení byly využity porty PTG3-PTG7 a PTE7, které obsluhují integrovaný znakový LCD displeje. Ten byl vyřazen za pomoci uživatelských přepínačů, konkrétně přepínačem číslo 4. Porty PTF6-PTF7 byly využívány pro sériovou komunikaci COM2.



Obr. 18. Modul připojený na vývojový kit

6 PROGRAMOVÉ VYBAVENÍ

Jako podpora modulu byla vytvořena knihovna funkcí, která umožňuje jednoduché ovládání modulu s LCD grafickým displejem. Knihovna byla vytvořena v jazyku symbolických adres, pomocí vývojového prostředí Freescale CodeWarrior. Obsahuje tři soubory, s názvem grafd_GB60 a koncovkami asm, h a inc.

6.1 Proměnné

Knihovna má dvě proměnné a to DDRAM a ABECEDA.

6.1.1 DDRAM

Z důvodu jednosměrné komunikace s modulem displeje, musela být vytvořena proměnná, která reprezentuje jednotlivé body, které jsou na displeji.

Jak bylo popsáno v kapitole 5.1.4, je displej rozdělen na dvě samostatné části. Proměnná DDRAM však toto rozdělení ignoruje a bere displej jako kdyby byl celistvý. Což znamená, že ukládá jednotlivé stránky (bajty) řádku za sebe. Až dosáhne 128. stránky, tak začne ukládat další řádku. Mapa celého displeje zabírá 1kB paměti RAM.

6.1.2 ABECEDA

Proměnná abeceda slouží k uložení abecedy, protože displej nedisponuje žádnou vnitřní abecedou. Je zde uloženo 95 znaků, každý znak zabírá 5 bajtů. Rozsah znaků v ASCII je od hodnoty 32 do 126. Tato proměnná zabírá 475B paměti RAM.

6.2 Uživatelské funkce

Tyto funkce jsou uživateli přístupné. Slouží k základní manipulaci s displejem.

6.2.1 DISP_INIT

Tuto funkci je nutné použít vždy jako první. Provádí inicializaci displeje. Nastaví komunikační porty jako výstupní a vynuluje je. Dále vypne reset a zapne displej. Provede vymazání jak displeje tak DDRAM. A uloží znaky do proměnné ABECEDA

6.2.2 ERASE_ALL

Slouží k mazání displeje a DDRAM. Mazání displeje je rychlejší než vymazat DDRAM a poslat ji na displej, protože maže oba čipy najednou.

6.2.3 DISP_DDRAM

Přepíše DDRAM na displej. Pokaždé když budete chtít zobrazit výsledky nějaké operace s DDRAM, tak je třeba zavolat tuto funkci.

6.2.4 DISP_POINT

Vstupy: zásobník – sloupec

Registr X - řádek

A – viditelnost bodu

Zapíše bod na souřadnice v zásobníku a registru X do DDRAM. Viditelnost bodu znamená, že v případě 0 v A bude bod z DDRAM vymazán, v případě jiné hodnoty než 0 v A bude bod zapsán. V případě, že budou souřadnice mimo rozsah, funkce se ukončí. Rozsah bodu je pro sloupec 0-127 a pro řádek 0-63.

6.2.5 DRAW_LINE

Vstupy: zásobník - počáteční sloupec, počáteční řádek, konečný sloupec

Registr X - konečný řádek

A – viditelnost přímky

Pomocí algoritmu popsaném v kapitole 3.1.2 zapíše jednotlivé body přímky do DDRAM. Viditelnost má stejné parametry, stejně tak i rozsah bodu počátečního a konečného bodu, jako v případě funkce 6.2.4.

6.2.6 DRAW_CIRCLE

Vstupy: zásobník - sloupec středu, řádek středu

A – poloměr

X – viditelnost kruhu

Pomocí algoritmu popsaném v kapitole 3.2.1 zapíše kružnici do DDRAM. Střed kružnice a poloměr mohou mít libovolné hodnoty, pokud není spočítán bod uvnitř displeje, tak se neprovede zápis do DDRAM. Viditelnost má stejné parametry jako obě předchozí funkce.

6.2.7 DISP_CHAR

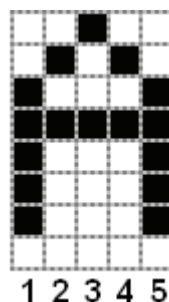
Vstupy: zásobník - sloupec

Registr X - řádek

A – ASCII znak

Pro zápis znaku do DDRAM je displej rozdělen na 8 řádků po 21 písmenech to nám tedy dohromady dá 168 znaků. Takže hodnota adresy sloupce je 0-20 a řádku 0-7. Rozsah zobrazovaných ASCII znaků je 32 až 126. V případě špatného zadání vstupních parametrů se funkce ukončí.

Znaky jsou uloženy v proměnné ABECEDA. Funkce zapíše příslušný znak do DDRAM. Znak má rozměr 5x7 viz obrázek (Obr.19.). Za každý znak je vložen jeden bajt s hodnotou 0, tedy malá mezera. Tudíž rozměr znaku má celkově velikost 6x8 i s mezerami.



Obr. 19. zobrazení znaku A

6.2.8 DISP_STRING

Vstupy: zásobník – sloupec

A - řádek

H:X – adresa řetězce

Funkce zapisuje řetězec znaků uložených na adrese předané v H:X do DDRAM. Rozsah adres umístění prvního znaku je stejný jako ve funkci DISP_CHAR. Řetězec musí být ukončen nulovým znakem. Pokud je řetězec delší, než počet pozic znaků do konce displeje, tak se zobrazí jenom znaky, které se na displej vejdou.

6.3 Vnitřní funkce

Vnitřní funkce knihovny uživatel k nim nemá přístup.

6.3.1 POSLI_DISP

Funkce pro komunikaci z obvodem 4094 a zápis na displej. Funkce posílá data, která byla uložena v registru A. Ještě před zavoláním této funkce musí být vyslány řídicí informace displeji o jaký přenos se jedná, zda-li o data či řídicí instrukce a kterému čipu informace patří. Po odeslání dat do obvodu je proveden zápis do displeje.

6.3.2 CIRCLE_POINTS

Algoritmus pro rasterizaci kruhu počítá jen jednu osminu kružnice se středem v [0,0]. Tato funkce provádí posun do skutečného středu a vykreslení ostatních oktantů. Vykreslení provádí pomocí funkce DISP_POINT.

6.3.3 ABECEDA_SAVE

Provádí naplnění proměnné ABECEDA rasterizovanými znaky ASCII ve velikosti 5x8, v rozsahu ASCII hodnot znaků 32-126.

ZÁVĚR

Hlavním cílem této bakalářské práce bylo nahrnout a vytvořit výukový modul s grafickým LCD displejem do předmětu Mikropočítače, s podpůrnou knihovou funkcí pro jeho ovládání a přiblížit tak studentům práci s tímto moderním zobrazovacím zařízením, které se dnes vyskytuje v mnoha praktických aplikacích.

V teoretické části se tato práce zabývá popisem vývojového kitu s mikropočítačem rodiny HCS08 a to jeho základními vlastnostmi, organizací paměti registrů a portů. Dále popisem zobrazovacích jednotek s malou hustotou informací a na to namazující rasterizací jednoduchých objektů a jejich algoritmy. Na závěr teoretické části je popsán program EAGLE, který slouží k návrhu schémat a desek plošných spojů.

První díl praktické části se věnuje návrhu modulu s grafickým LCD displejem. Jako první je popsán grafický displej, jeho komunikační a grafické možnosti, poté je popsáno schéma zapojení a návrh desky plošného spoje v programu EAGLE. Na závěr je popsána výroba plošného spoje a oživení celého výrobku.

Druhý díl praktické části se zabývá popisem knihovny, vytvořené pro snadnou obsluhu modulu. Nejprve popisem a proměnných, dále uživatelskými funkcemi a na závěr popisem vnitřních funkcí.

Byl vytvořen prototyp modulu s grafickým LCD displejem a byl odzkoušen, společně s knihovou funkcí na vývojovém kitu.

CONCLUSION

The main aim of this bachelor thesis was to design and create education module with LCD graphic display for Microcontrollers course and its library of functions for control and introducing operating with this device, which is used in many applications to students.

Theoretical part describes a development kit with HCS08 family microcontroller, its features, organization of register memory and ports. This part also describes display devices with low information density and then rasterization of simple objects and their algorithms. In the end of theoretical part designing with EAGLE is explained.

The first part of practical part is about designing LCD graphic display module. As the first thing graphic display and its communication potentialities are explained. Then cabling diagram, design of printed circuit board in EAGLE, creation and activation of module are described.

The second part of practical part is about libraries, created to easy service of module. It starts with description variables, then user functions and finally with description of internal functions.

The prototype of LCD graphic display module was created and tested, together with library of user functions on the development kit.

SEZNAM POUŽITÉ LITERATURY

- [1] DOLEČEK, Jaroslav. *Moderní učebnice elektroniky 3. : Optoelektronika a optoelektrické prvky*. 1. vyd. Praha : BEN - technická literatura, 2005. 160 s. ISBN 80-7300-184-5.
- [2] MAŤÁTKO, Jan. *Elektronika*. 3. rozš. vyd. Praha : IDEA SERVIS, 2002. 327 s. ISBN 80-85970-42-2.
- [3] BURKHARD, Mann. *C pro mikrokontroléry*. 1. vyd. Praha : BEN - technická literatura, 2004. 280 s. ISBN 80-7300-077-6.
- [4] FREESCALE SEMICONDUCTOR *MC9S08GB/GT Data Sheet* [online]. 2004-2008 , 12/3/2004 [cit. 2008-05-12]. Dostupný z WWW: <http://www.freescale.com/files/microcontrollers/doc/data_sheet/MC9S08GB60.pdf>.
- [5] FREESCALE SEMICONDUCTOR. *M68EVB908GB60 User's Manual* [online]. 2004-2008 , 01/12/05 [cit. 2008-05-12]. Dostupný z WWW: <http://www.freescale.com/files/soft_dev_tools/doc/user_guide/M68EVB908GB60UM.pdf>.
- [6] TICHÝ, Milan. *Jednočipové mikropočítače* [online]. 1998 [cit. 2008-05-12]. Dostupný z WWW: <<http://lucy.troja.mff.cuni.cz/~tichy/kap9/jednocpoc.html>>.
- [7] MATOUŠEK, David. *Práce s inteligentními displeji LCD*. 1. vyd. Praha : BEN - technická literatura, 2006. 224 s. ISBN 80-7300-121-7.
- [8] GABRIEL, Michal. *Základy 2D grafiky - úsečka* [online]. 2004 [cit. 2008-05-12]. Dostupný z WWW: <<http://www.pcsvet.cz/art/article.php?id=800>>.
- [9] GABRIEL, Michal. *Základy 2D grafiky - kružnice a elipsa* [online]. 2004 [cit. 2008-05-12]. Dostupný z WWW: <<http://www.pcsvet.cz/art/article.php?id=820>>.
- [10] POKORNÝ, Pavel. *Základy počítačové grafiky*. 1. vyd. Zlín : Univerzita Tomáše Bati, Fakulta technologická, Institut informačních technologií, 2004. 120 s. ISBN 80-73181-61-4.

- [11] ČAPKA, Ladislav. *Akcelerace grafických operací s využitím FPGA*. [s.l.], 2007. 47 s. Vysoké Učení Technické v Brně. Diplomová práce. Dostupný z WWW: <<http://www.fit.vutbr.cz/study/DP/rpfile.php?id=5484>>.
- [12] *User`s guide ATM12864D* [online]. [cit. 2008-05-13]. Dostupný z WWW: <<http://www.hebeiltd.com.cn/lcm.datasheet/ATM12864D.pdf>>.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

| | |
|--------|--|
| VFD | Vacuum Fluorescent Display |
| LED | Light-Emitting Diode |
| TTL | Transistor–Transistor Logic |
| CMOS | Complementary metal–oxide–semiconductor |
| TN | Twist Nematic |
| HTN | Hyper Twist Nematic |
| STN | Super Twist Nematic |
| DSTN | Double Super Twist Nematic |
| TSTN | Triple Super Twist Nematic |
| FSTN | Film Compensated Super Twist Nematic |
| LCD | Liquid Crystal Display |
| DPS | Deska plošného spoje |
| EAGLE | Easily Applicable Graphical Layout Editor |
| TOP | Vrchní strana plošného spoje |
| BOTTOM | Spodní strana plošného spoje |
| CISC | Complex Instruction Set Computer |
| RISC | Reduced Instruction Set Computer |
| ALU | Arithmetic Logic Unit |
| LIFO | Last In – First Out |
| SP | Stack Pointer |
| RAM | Random-Access Memory |
| D/A | Digital to Analog |
| DDA | Digital Differential Analyzer |
| ASCII | American Standard Code for Information Interchange |

SEZNAM OBRÁZKŮ

| | |
|---|----|
| Obr. 1. Architektura von Neuman | 13 |
| Obr. 2. Harvardská architektura..... | 13 |
| Obr. 3. Vývojový kit M68EVB08GB60 | 15 |
| Obr. 4. Registry HC08 | 17 |
| Obr. 5. Mapa paměti | 20 |
| Obr. 6. Uspořádání VFD..... | 22 |
| Obr. 7. Uspořádání segmentového displeje | 24 |
| Obr. 8. Struktury kapalných krystalů..... | 25 |
| Obr. 9. Průchod světla skrz LCD | 26 |
| Obr. 10. Multiplexní buzení maticového displeje | 28 |
| Obr. 11. Výběr pixelů úsečky | 30 |
| Obr. 12. Část rastru kružnice | 32 |
| Obr. 13. Control Panel | 34 |
| Obr. 14. Editor schémat | 36 |
| Obr. 15. Editor DPS..... | 37 |
| Obr. 16. Blokové schéma modulu | 41 |
| Obr. 17. Rozložení DDRAM displeje..... | 45 |
| Obr. 18. Modul připojený na vývojový kit..... | 49 |
| Obr. 19. zobrazení znaku A | 52 |

SEZNAM TABULEK

| | |
|---|----|
| Tab. 1. Obsazení vývodů displeje..... | 40 |
| Tab. 2. Instrukce displeje..... | 43 |
| Tab.3. Propojení portů mikropočítače s modulem..... | 47 |

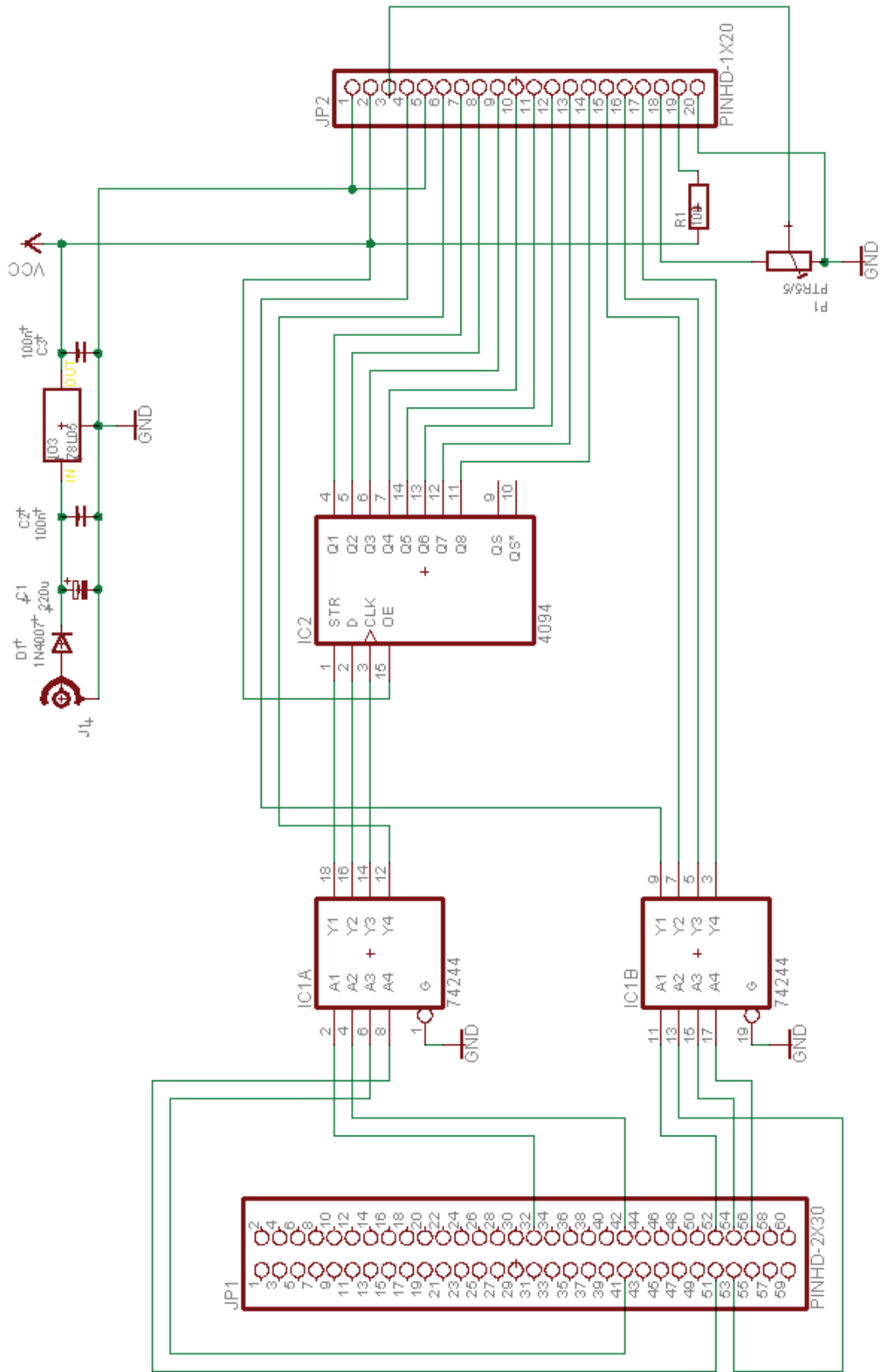
SEZNAM PŘÍLOH

- P I Seznam součástí
- P II Schéma zapojení
- P III Předloha DPS
- P IV Rozmístění součástí
- P V Elektronické přílohy a verze bakalářské práce na CD

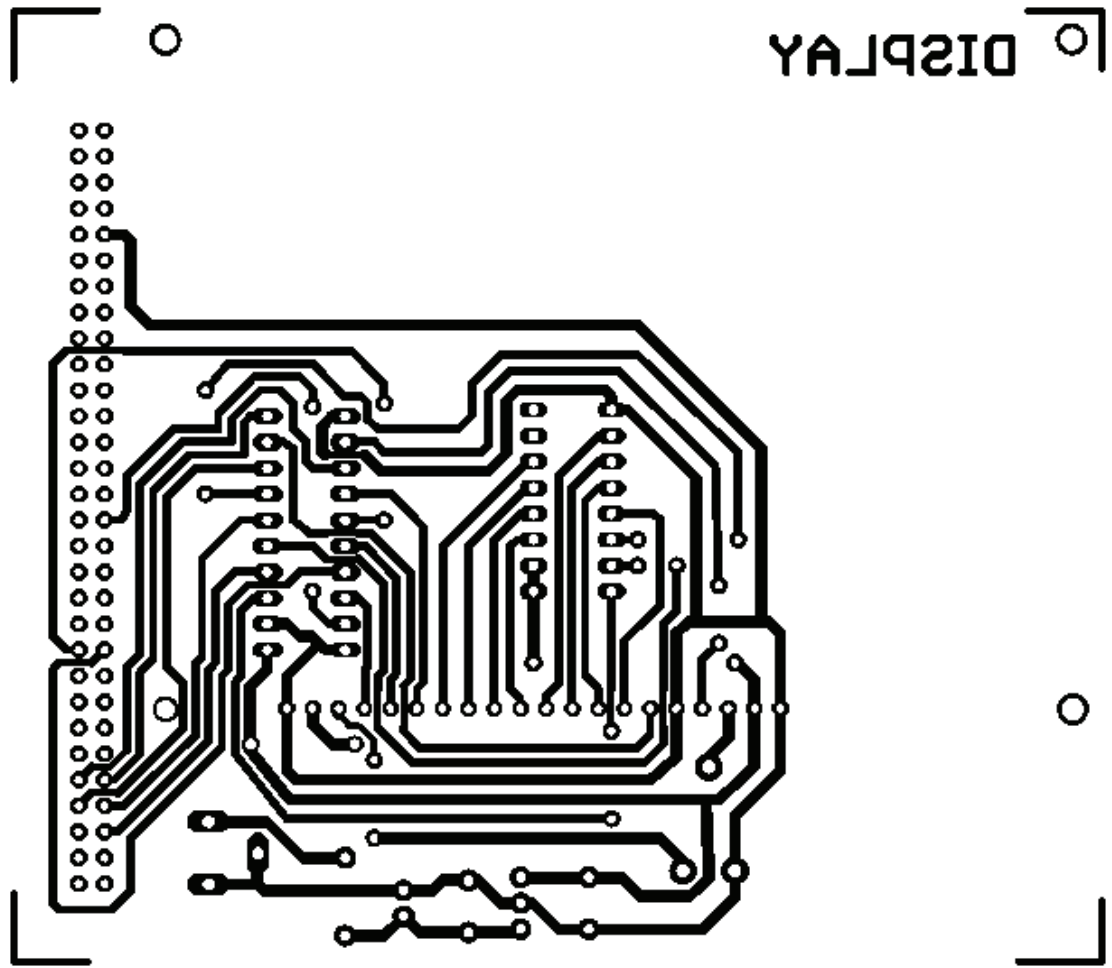
PŘÍLOHA P I: SEZNAM SOUČÁSTEK

| | ks | typ | referenc e |
|----|----|--------------------------|---------------|
| 1 | 2 | 100nF | C2,C3 |
| 2 | 1 | 220 μ F/16V | C1 |
| 3 | 1 | 1N4007 | D1 |
| 4 | 1 | 4094 | IO2 |
| 5 | 1 | 74HT244 | IO1 |
| 6 | 1 | 78L05 | IO3 |
| 7 | 1 | 150 Ω | R1 |
| 8 | 1 | 10k Ω | P1 |
| 9 | 1 | Napájecí konektor | J1 |
| 10 | 1 | Konektorový kolík 2x30 | JP1 |
| 11 | 1 | Konektorový kolík 1x30 | JP2 |
| 12 | 1 | Patice DIL 16 | -- |
| 13 | 1 | Patice DIL 20 | -- |
| 14 | 1 | LCD Display ATM12864D | -- |

PŘÍLOHA P II: SCHÉMA ZAPOJENÍ



PŘÍLOHA P III: PŘEDLOHA DPS



PŘÍLOHA P IV: ROZMÍSTĚNÍ SOUČÁSTEK

