

UNIVERZITA PARDUBICE
DOPRAVNÍ FAKULTA JANA PERNERA

Univerzální provozní senzitometr
Bc.Jasmin Draganovi

DIPLOMOVÁ PRÁCE
2009

Univerzita Pardubice
Dopravní fakulta Jana Pernera
Katedra elektrotechniky, elektroniky a zabezpečovací techniky v dopravě
Akademický rok: 2008/2009

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: Bc. Jasmin DRAGANOVIČ
Studijní program: N3708 Dopravní inženýrství a spoje
Studijní obor: Dopravní infrastruktura-Elektrotechnická zařízení
v dopravě
Název tématu: Univerzální provozní senzitometr

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

1. Rešerše dostupných nebo v minulosti používaných nebo vyvinutých senzitometrů
2. Rozbor zjištěných poznatků, navržení koncepce přístroje
3. Návrh osvitové části zařízení dle zvolené koncepce, potřebné výpočty
4. Návrh mechanického provedení pohonu a řízení motoru, potřebné výpočty
5. Volba a návrh řízení výsledného zařízení
6. Zhotovení prototypu

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování diplomové práce: **tištěná**

Seznam odborné literatury:

1. Urban, M, Ing. Sklenář, Z., Základy senzitometrie pro I.- III. Ročník SOU, Praha, Státní pedagogické nakladatelství, 1987, 251s
2. ČSN ISO 5799 (66 6624) Fotografie - Přímě exponovaný lékařský a dentální radiografický film - Stanovení citlivosti ISO a průměrného gradientu ISO
3. ČSN ISO 6 (66 6625) Fotografie - Systémy negativního filmu a jeho zpracování pro černobílou obrazovou fotografii - Stanovení citlivosti podle ISO

Vedoucí diplomové práce:

Ing. Ondřej Černý

Katedra elektrotechniky, elektroniky a zabezpečovací techniky v dopravě

Datum zadání diplomové práce: **5. prosince 2008**

Termín odevzdání diplomové práce: **25. května 2009**

prof. Ing. Bohumil Culek, CSc.
děkan

L.S.

prof. Ing. Vladimír Šejbal, CSc.
vedoucí katedry

V Pardubicích dne 18. února 2009

Prohlašuji:

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

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

Souhlasím s prezentováním a zpřístupněním své práce v Univerzitní knihovně.

V Praze dne 25. 5. 2009

Jasmin Draganovi

Podkování

Chtěl bych touto cestou podkovat za cenné připomínky a podněty vedoucímu diplomové práce Ing. Ondřejem Černému. Dále děkuji Ing. Jiřímu Havelkovi za podnětné rady při tvorbě programu.

ANOTACE

Diplomová práce popisuje návrh a konstrukci provedení univerzálního senzimetru pro přesnou expozici fotocitlivých materiálů v širokém rozsahu citlivostí (černobílé filmy a papíry, barevné filmy a papíry, medicínské rentgenové filmy určené pro primární expozici sv tlem). Vysokou pružnost přístroje zajišťuje nový druh štrbinové závěrky s konstantními rozměry štrbin a variabilní rychlostí pohybu.

Senzimetr může být použit pro většinu úloh při vývoji a kontrole ve fotografickém průmyslu, stejně jako při kontrole zpracování.

KLÍČOVÁ SLOVA

senzimetr; závěrka; modulátor světla; světelný zdroj; filtr; expozice; senzitogram; štrbinová závěrka

TITLE

General purpose process sensitometer

ANNOTATION

This work describes the multi-purpose sensitometer for exact exposition of photosensitive material of wide range of speed (B&W films and papers, color films and papers as well as medical X-ray film dedicated for light exposures). High flexibility is caused by the new-type of focal-plane shutter with variable speed of moving.

The sensitometer can be used for most research- and checking tasks in photographic industry as well as for processing control.

KEYWORDS

senzimeter; shutter; light modulator; light source; filter; exposure; sensitogram; focal-plane shutter

OBSAH

ÚVOD	9
1. Senzitometrie	11
2. Pohled senzimetru	12
2.1 Univerzální (laboratorní) senzimetrie.....	12
2.2 Senzimetrické systémy pro kontrolu zpracování	12
2.3 Provozní senzimetrie	13
2.4 Speciální senzimetrie	14
3. Řešení senzimetru	15
4. Fotografické citlivé vrstvy	18
4.1 Senzimetrie.....	19
4.2 Senzimetrická charakteristika	19
4.3 Kontrola zpracování	20
4.4 Zernání	21
4.5 Závoj, maximální hustota	22
4.6 Citlivost, gradient	22
5. Princip senzimetru.....	23
5.1 Senzimetrie	23
5.2 Rozbor vlastností	24
6. Cíl práce.....	27
6.1 Obecné	27
6.2 Světelný zdroj.....	27
6.3 Filtry	28
6.4 Závěska	28
6.5 Uživatelský interface	28
7. Návrh řešení.....	30
7.1 Konceptce senzimetru	30
7.2 Celkové konstrukční uspořádání	30
8. Mechanická část senzimetru	32
8.1 Lineární kluzné vedení	32
8.2 Držák filtrů v etn držáku žárovky a závěsky	34
8.1.1. Mechanická část filtrů	34
8.3 Závěska	38
8.4 Modulátor.....	38
9. Část elektro	41
9.1 Napájení.....	41
9.2 Ovládání.....	42
9.3 Osvětlovací systém	44
9.4 Světelný zdroj.....	45
9.5 Řízení proudu expoziční žárovky	46
9.6 Řízení pohybu expozičního vozíku	50
9.7 Procesorová deska	54
10. Programové vybavení	58
10.1 Popis programu senzimetru	58
10.1.1 Soubor senzit.c	61
10.1.2 Soubor serial.c	63
10.1.3 Soubor ser_senz.c	63
10.1.4 Soubor lcd.c.....	64
10.1.5 Soubor ppr_senz.c.....	64

10.2	Koncepce ovládacího programu pro senzitometr	65
ZÁV R	67
SEZNAM POUŽITÉ LITERATURY	70
SEZNAM WWW STRÁNEK (8.5.2007)	71
SEZNAM TABULEK	72
SEZNAM OBRÁZK	73
SEZNAM ZKRATEK	74
SEZNAM P ÍLOH	75

ÚVOD

V poslední době se dalšího rozvoje fotografického průmyslu v ČR i ve světě dotýká digitalizace. V oblastech, kde převládá komerce a přístroje se vyrábí již ve velkých sériích (amatérská fotografie) už prakticky úplně vytlačila klasickou (analogovou) fotografii. Existuje přesto řada odvětví, kde klasická fotografie na film a digitální fotografie mohou ještě dlouhou dobu koexistovat. Jde zejména o obory, kde digitální snímek obrazu velkých formátů jsou ekonomicky nevýhodné (medicinální radiografie) nebo nepříliš vhodné pro aplikaci v provozních podmínkách továren, staveb atd. (průmyslová radiografie). I v klasické obrazové fotografii zůstávají oblasti, kde klasický film je nenahraditelný. Takovými oblastmi je například dálkové letecké snímání, kartografie a polygrafie.

Akoliv někdy ekonomicky slabí výrobci fotografických materiálů již ukončili činnost, zůstává několik firem, které se zaměřily na užší oblast ekonomicky výhodných fotografických materiálů, které před sebou mají ještě dlouhou budoucnost. Pokračující výroba s sebou nese nutnost kontroly technologického procesu ve všech fázích výroby. Jednou z potřebovaných metod je senzimetrie. Vzhledem k tomu, že ve všech hlavních fotochemických firmách se další vývoj technologie a kontrolních metod utlumil, ukazuje se, že pro další použití v oboru jsou k dispozici jen dosluhující přístroje. Má-li se výroba udržet, je potřeba přistoupit jejich zásadní modernizaci s využitím nových principů a moderní součástkové základny. Příspěvkem do této oblasti má být i tato diplomová práce.

V práci je nastíněna koncepce senzimetru se štrbinovou závěrkou, která vychází ze senzimetru vyvíjených v 80. letech ve VÚZORT, které nebyly nikdy realizovány. Tento princip postupné expozice senzitogramu je v návrhu zmněn tak, že rámek s exponovaným materiálem je pevný, zatímco se pohybuje expozitivní štrbina se světelným zdrojem a filtry. Celková koncepce je přizpůsobená moderní součástkové základně.

Toto uspořádání přináší konstrukční jednoduchost, jednoduché a ergonominické ovládání vhodné pro práci ve tmě, možnost naprogramování expozitivních parametrů z velkého množství možných. Přístroj je kompaktní, se spotřebou minimalizovanou vzhledem k mnohem lepšímu využití světla. Exponované senzitogramy jsou velké a snadno měřitelné běžnými denzimetry. Do přístroje je možné upnout prakticky

libovolné absorpční modulátory do celkového rozměru 32 x 200 mm, stupňové nebo plynulé.

Nevýhodou popsaného řešení je postupná expozice senzigramu; celková expozice trvá déle. Tento nedostatek je vyvážen jednoduchostí a univerzálností. Konstrukce přístroje byla dovedena do stadií ověřovacího vzorku; zatím tedy není možno k práci připojit podmínky z praktického používání. Zdá se však, že přístroj bude přínosný pro oblast, pro kterou byl určen a předpokládá se jeho výroba.

1. Senzitometrie

Poátky senzitometrie jako vdy sahají do konce 19. století, kdy fotografická výroba opustila laboratorní prostředí a stala se světoobecným průmyslovým oborem. Současnost průmyslovou výrobou vyvolala potřebu standardizace a kontroly produkce.

Senzitometrie je v oblasti fotografických materiálů nejdůležitější kontrolní metodou, nebo vlastně modeluje vlastní fotografický proces. Již v prvních letech existovaly zkoušky expozice, které pomocí odstupované expozice zkušebních proužků vedly k nalezení správné expozice a v průběhu let se vyvinuly v exponometrii.

Kromě toho bylo nutno stanovit správnou citlivost fotografických materiálů, což bylo nutné jednak pro klasickou fotografii, ale potřeba ještě více vzrostla s rozvojem kinematografie.

První senzitometrické postupy využívaly optické lavice, kde se vhodného osvětlení dosahovalo změnou vzdálenosti fotomateriálu od světelného zdroje. Později vznikly absorpční modulátory expozice, které se v nezměnném principu používají dodnes. Definované expozice se dosahovalo v tšinou padací závěskou, s rozvojem elektroniky pak vznikaly různé druhy elektronicky řízených závěsek. Současnost s tím dostávala optická lavice kompaktní podobu a změnila se v senzitometr.

Pomineme-li optické lavice, jejichž éra skončila se začátkem 2. světové války, moderní senzitometry můžeme rozdělit do následujících typů, které jsou uvedeny níže.

2. Pohled senzimetru

2.1 Univerzální (laboratorní) senzimetrie.

Dlouhou dobu byly jedinými přístroji, které sloužily ve všech odvětvích. Vyznačovaly se stabilní konstrukcí se žárovkovým světelným zdrojem s nastavitelnými parametry, později napájeným ze stabilizátoru, závěrkou s jediným zásemením, zejména padací, později elektronicky řízenou s možností volby několika expozičních časů. Filtry zakládány do světelného paprsku ručně, později servosystémy, absorpčním modulátorem pro vytvoření odstupované expozice s možností nastavení osvětlení změnou vzdálenosti modulátoru od světelného zdroje a vhodného držáku fotografického materiálu.

Nevýhodou všech takto koncipovaných přístrojů byly protichůdné požadavky na velikost měřené plochy potřebné pro pohodlnou obsluhu a požadavku na rovnoměrnou expozici celé plochy. Tato skutečnost vedla k prodlužování vzdálenosti modulátoru od světelného zdroje, což dále vedlo k potřebě výkonných světelných zdrojů, jejich energetické náročnosti, krátké životnosti na nadměrném výkonu tepla.

2.2 Senzimetrické systémy pro kontrolu zpracování

Tyto přístroje vznikly v období, kdy byly kladeny vysoké požadavky na standardnost zpracování, na trhu se objevily první vyvolávací automaty, které bylo nutno kontrolovat a seřizovat. Zejména v oblasti rentgenové diagnostiky, kdy bylo riziko nadměrné expozice pacienta, vznikly poměrně dokonalé, i když v principu jednoduché expoziční systémy, založené na velkoplošném luminiscenčním fóliovém světelném zdroji. Spektrální vlastnosti luminiscenčních fólií je ale třeba odlišovat pro úzkopásmovou expozici, tedy zejména pro simulaci expozice rentgenových filmů určených pro práci se zesilovacími foliemi a pro kontrolu zpracování těchto filmů.

- Fantómy: snímky standardního pedmetu (skupin pedmetů). V radiografii zejména snímky hliníkových stupovitých klínů nebo různých tlustých hliníkových destiček zalitých ve společném tělese. V obrazové fotografii tento princip využívají snímky šedých nebo barevných tabulek.

- P edexponované testy: jedná se vlastn o totéž, ale expozice standardního p edm tu je p enechána výrobci fotografického materiálu, kde je zpravidla v tší možnost zajistit standardnost expozice. Nevýhodou je omezená životnost t chto test , nebo latentní obraz postupn degraduje a rovn ž p i skladování stoupá s ásem závoj.
- Senzitivity s fóliovým sv telným zdrojem: výhodou je kompaktní provedení, velký rozm r modulátoru. Nevýhodou je jen omezená využitelnost.

2.3 Provozní senzitivity

- Senzitivity s bodovým sv telným zdrojem: zdrojem sv tla je žárovka, v pom rn velké vzdálenosti je modulátor a exponovaný materiál. Vzdálenost je kompromisem mezi rovnom rností osv tlení modulátoru (z praktických d vod je požadavek na co nejv tší modulátor) a snadnou realizovatelností. Parametry expozice lze snadno vypo ítat. Klasickým p ípadem je optická lavice, senzitivní FSR-4 (kompromisní ešení s 5 mm širokým polem modulátoru a st ední vzdáleností od zdroje 500 mm) a senzitivní 2M. Požadavek na velký rozm r pole modulátoru byl u senzitivní 2M zajišt n modulátorem, který spolu s exponovaným materiálem tvo í sou ást válcové plochy se st edem v míst vlákna žárovky. Tak mají jednotlivá pole modulátoru (p esn ji e eno osa modulátoru) stejnou vzdálenost od sv telného zdroje. U všech typ senzitivní je použita padací záv rka mechanická nebo elektronicky ízená, p ichází v úvahu ješt rota ní záv rka.
- Senzitivity se št rbinovou záv rkou: umož ůjí kompaktní stavbu, protože sv telný zdroj osv tluje z pom rn malé vzdálenosti jen expozi ní št rbinu, p ed kterou se pohybuje vozík s modulátorem a exponovaným materiálem. Pohyb vozíku je lineární (senzitivní Kodak model 6, VÚZORT MKS-1, KS-1) nebo rota ní (Kodak model 7, systém s „gumovou kšandou“).
- Senzitivity zvláštní. Do této skupiny za azují p ístroje, které není možno za adit do n které z p edchozích kategorií, a to senzitivní s neselektivní modulací. Zde je místo obvyklého absorp ního

modulátoru, jehož hlavní nevýhodou je spektrální selektivita (pokud je modulátor vyráběn obvyklým fotografickým způsobem, případně nanášením grafitové suspenze na sklo, případně tmelením z irího a erného skla). Technologie výroby jiných vhodných modulátorů nejdražší součástí senzimetru, má-li vyhovět všem fyzikálním požadavkům. Tuto nevýhodu odstraní prof. Jaroslav Bouček (jedna z nejvládnějších kapacit z oboru senzimetrie v Československu minulého století) zavedením neselektivního (dírkového) modulátoru do světelného paprsku. Modulace osvětlení jednotlivých polí je dosahována synchronním vkládáním plechových přepážek, které jsou provrtány dírkovým rastrem, jehož propustnost je dána poměrem plochy vyvrtaných dírek celkové osvětlené ploše. Z praktických důvodů (synchronizace dírkovaného kotouče s posuvem filmu) našel uplatnění jen ve filmových laboratořích Barrandov pro kontrolu filmových kopií (senzimetr je vlastně součástí filmové kopírky na perforovaný film).

2.4 Speciální senzimetrie

Jde o speciální přístroje pro kinematografii, v podstatě upravené kinematografické kopírky s možností modulace, spektrosenzimetrie se spektrálně proměnným zářením s monochromátorem pro stanovení spektrální citlivosti a speciální granulometrické senzimetrie pro expozici homogenních ploch pro měření zrnitosti, tedy v případech, kdy by kontaktní expozice zrnité struktury absorpčního modulátoru vedla k nepřesným měřením.

Níže je uveden přehled senzimetrie, z nichž komerčně dostupné jsou jen senzimetrie pro kontrolu zpracování.

3. Rešerše senzitometr

- **FSR-4 (FSR-41):** ruský, kolem roku 1957. Robustní konstrukce, padací závěrka s asem 1/20 s pro filmy a ruční ovládání (B) pro papíry. P vodní wolframová žárovka 12 V s vinutým kuželovým vláknem, při rekonstrukci nahrazena halogenovou s elektronickou stabilizací. Nastavitelná vzdálenost od modulátoru (pro nastavení osvětlení). Konverzní a barevné filtry. Modulátor p vodní grafitový, při rekonstrukci nahrazen filmovým. 21 polí po 5 mm.
- **2M:** zachovává stejnou koncepci i geometrii, ale vybaven interní stabilizací proudu žárovkou, elektronickou závěrkou (0,05 - 40 s), optickou a akustickou signalizací, okénkovým diskriminátorem pro kontrolu žhavení žárovky. Modulátor 21 polí po 5 mm nebo 21 polí po 9,5 mm, ohnutý s rádiusem 500 mm.
- **Joyce 2L:** na principu kopírky; 2 žárovky, válcová závěrka se štrbinou. 21 polí po 9,5 mm nebo plynulý modulátor. Prakticky nemožno metrologicky nastavit (homogenita pole), žárovky nemají ustálený světelný tok (zapínají se těsně před otevřením závěrky).
- **Kodak model 6:** bubnový senzitometr, materiál se vkládá do bubnu s modulátorem, který se otočí před osvětlením štrbinou.
- **Kodak model 7:** stejná koncepce, ale s lineárním posunem filmu před štrbinou.
- **MKS-1 (VÚZORT):** stejná koncepce jako Kodak 7, pohon strunou, těžký vozík, výměnná štrbina (čas 0,1 - 0,05 - 0,01 s), šedé a barevné filtry. Pohon strunou spolu s hmotností vozíku vytváří mechanický oscilátor, kmity se projeví nerovnoměrnou expozicí (proužky) při krátkých časech. Halogenová žárovka napájená z externího stabilizátoru.
- **KS-1 (VÚZORT):** stejná koncepce, pohon vozem stejným motorem se snímáním skutečné rychlosti (optický rastr). Žárovka s regulací, změna barevné teploty, čas 0,01 - 10 s, elektronické ovládání a indikace filtru a expozice. Byl realizován jen jako prototyp, poruchovost zejména servosystému pro pohon motoru.

- **X-Rite 333:** bateriový senzitivní modulátor s 21 poli, pro modro - nebo zelenocitlivé rtg. filmy, svitkové filmy a kinofilmy. Parametry: asová stabilita $\pm 0,02 \log H/\text{rok}$, opakovatelnost $\pm 0,04 \log H$, teplotní citlivost $\pm 0,02 \log H$ v rozmezí 10 – 43 °C, napájení 9V alkalická baterie, vlnová délka maxima $465 \pm 10 \text{ nm}$ pro modré sv tlo, $520 \pm 10 \text{ nm}$ pro zelené sv tlo, expozi ní doba 0,1 s, rozm ry 7,6 x 17,8 x 3,8 cm, hmotn ost 0,6 kg.
- **X-Rite 334:** zdokonalená verze p edcházejícího typu; liší se tím, že na spodní stran sk í ky je vícepolohový p epína , který umož ũje reprodukovateln m nit expozici a tím p izp sobit expozici použ ítemu typu filmu tak, aby bylo dosaženo správné expozice u pole (zpravidla 11. u klínu s 21 poli) slouž ícího k ode tení relativní citlivosti (speed index).
- **X-Rite 383:** bateriový sensitometr se dv ma barvami expozi ního sv tla a jednostrannou expozicí koncipovaný pro kontrolu zpracování kinematografických a rentgenových film . Sv telný zdroj simuluje zá ení modrých a zelených zesilovacích fólií. Životnost 9V alkalické baterie je cca 1 rok. Volba barvy expozi ního sv tla je pomocí posuvného p epína e na elním panelu. Expozi ní doba je stavitelná DIP -p epína i p ístupnými otvorem na spodní stran samostatn pro modré i zelené sv tlo. Expozice je stavitelná v 7 stupních, p itom normální expozice (stupe . 3) je zvolena tak, aby bylo dosaženo na 11. stupni u materiálu optické hustoty p ibližn 1,0 D (+ hustota podložky a závoje). P i používání automatického denzitometru X-Rite 380/381 je t eba expozici zvolit tak, aby byl viditelný rozdíl v hustotách každého exponovaného pole, gradient mezi poli 7 a 15 min. 0,7 a expozice na formát má být provedena tak, aby na obou koncích senzitogramu byl asi 3 cm ístý (neexponovaný) zavád cí úsek. Modulátor má 21 polí, klínová konstanta je 0,15. P i expozici jednostranných materiál se materiál vkládá vrstvou dol .
- **X-Rite 384:** stejný jako typ X-Rite 383, ale pro oboustrannou expozici, tj. expozici obou stran najednou jediným stiskem expozi ního tla ítká.
- **Senzitivní X-RITE 396:** jednostrann exponující senzitivní s volitelným spektrem osvitu pro modrocitlivé i zelenocitlivé filmy.

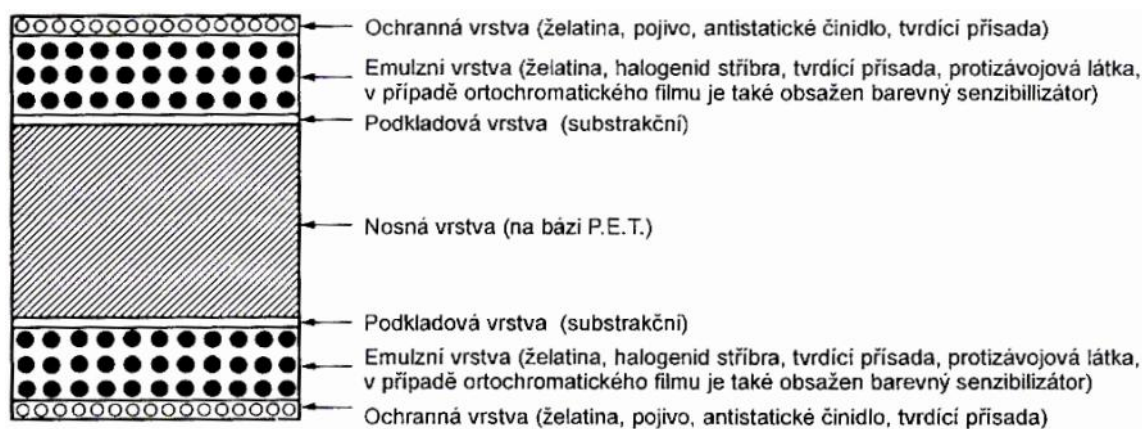
Kompaktní, přesný s vysokou přesností a stabilitou. Nahrazuje předchozí typy 334 a 393.

- **Qualix-X:** jedná se o soupravu přesného senzimetru pro modrý a zelený proces, denzitometru bez vlastního světelného zdroje a digitálního teploměru. Souprava je určena především k nastavování a kontrole vyvolávacích medicinálních automatů. Senzimetr je tvořen skříňkou, na jejíž horní straně je senzimetrický modulátor pod příklopným víkem. Po založení filmu pod víko a jeho přitisknutí se materiál pružně přitiskne k modulátoru a spustí expozice. Modulátor je elektricky přepínatelný do dvou barev odpovídajících modrému a zelenému záření zesilovacích fólií. Přístroj má akustickou indikaci jedním nebo dvojitým pípnutím podle zvoleného světelného režimu. Indikuje se konec expozice. Napájení je bateriové.

4. Fotografické citlivé vrstvy

Jako nositel informace u medicínálních a defektoskopických rentgenových filmů se používá fotografická vrstva. Její hlavní součástí je suspenze halogenidů (Cl, Br, I) stříbrných (světlocitlivé mikrokrystaly halogenidu stříbra) v nosném koloidu (želatina). Halogenidy stříbra použité v rentgenových filmech jsou především bromid stříbrný (AgBr) s malým množstvím jodidu stříbrného (AgJ). Mění se ve své charakteristice podle jeho složení, velikosti částic a jejich distribuce. Lze vytvořit obraz, který vzniká po dopadu záření, lze zviditelnit vyvolávacím procesem. Chemickým působením (redukci) během vyvolávacího procesu se mění osvětlený halogenid stříbra na částky kovového stříbra, které absorbují procházející světlo při prohlížení nebo kopírování. Nevyužitý halogenid stříbrný se odstraní z vrstvy působením do rozpustného komplexu a vypráním.

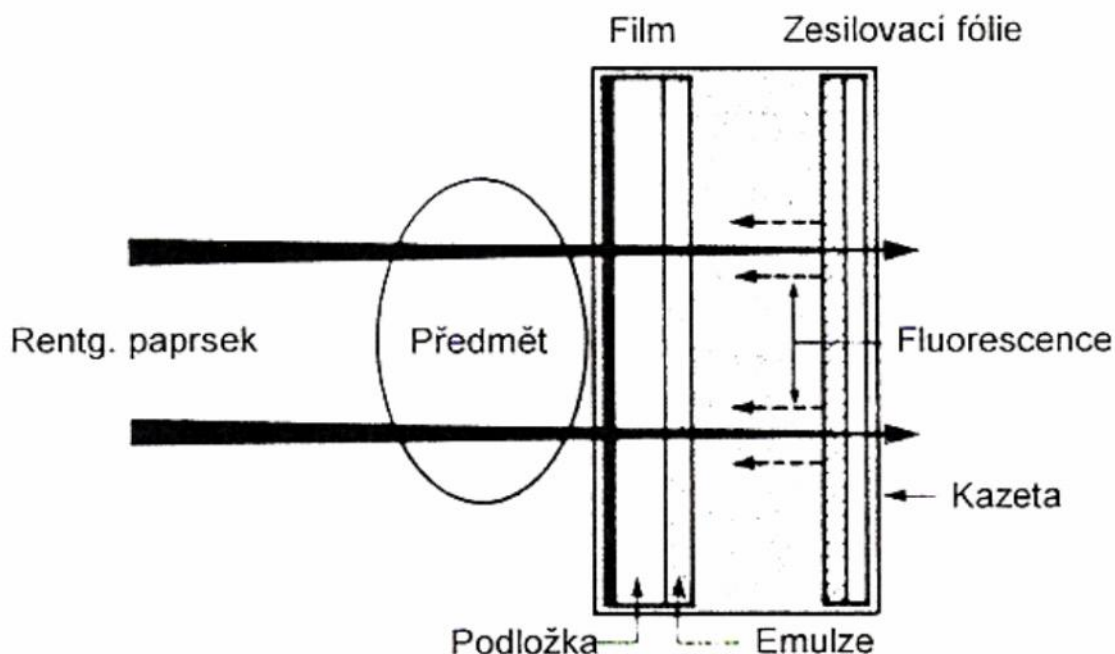
Nosná vrstva, nebo-li podložka pro rentgenový film je na bázi polyethylentereftalátu (polyester, PET – platí pouze pro rentgenové filmy). Její tloušťka je asi 180 mikrometrů a musí vynikat mechanickou pevností, rovinností a rozměrovou stabilitou. Na nosnou vrstvu se nanáší podkladová vrstva nebo-li substrakční, jejichž cílem je soudržnost emulze s podložkou. Poté je nanášena emulzní vrstva. Obsahuje želatinu, výše zmíněný halogenid stříbrný, tvrdící přísadu a protizávojovou látku.



Obr. 4.1 Rentgenový film (se dvěma emulzními vrstvami)

Želatina, ve které je rozptýlen halogenid stříbra, se vyrábí z kostí a kůže zvířat. Tvrdící přísada, která je jak v emulzní, tak v ochranné vrstvě, reaguje s želatinou a vytvrzuje tak každou vrstvu a tím chrání proti poškrábání a otiskům. Ochranná vrstva obsahuje kromě tvrdící přísady také přísady zabráňující lepení filmu,

antistatické inidlo a protizávoje inidlo, které snižuje úroveň závoje a hem stárnutí.



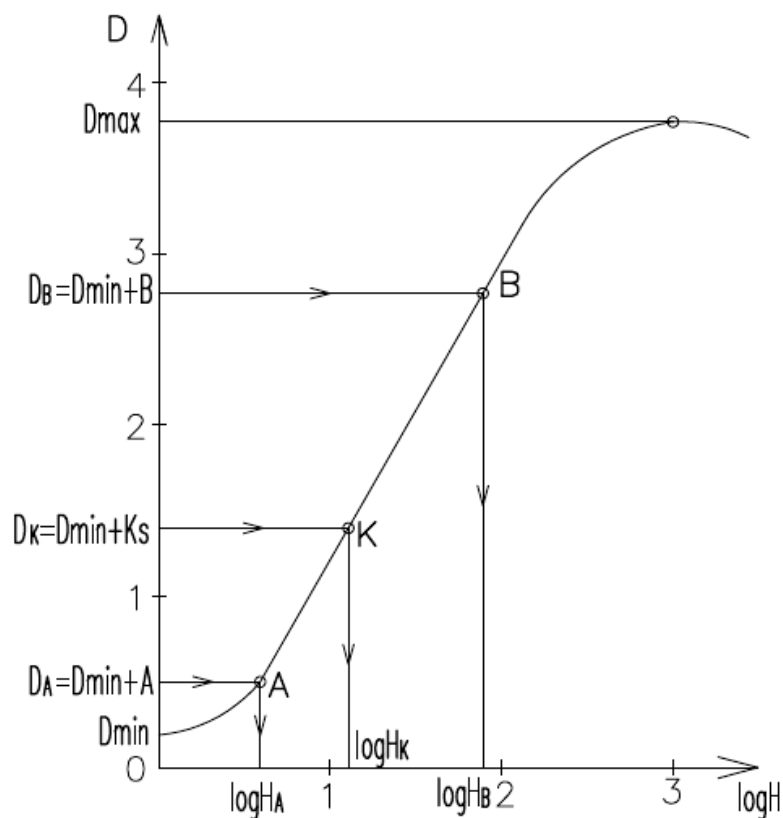
Obr. 4.2 Princip rentgenování (film s jednou emulzí)

4.1 Senzitometrie

Slovo vzniklo z pojmu „senzitivita - citlivost“ a „metrie – měření“. Senzitometrie je vada, která se zabývá kvantifikací účinku záření na fotografickou vrstvu a účinku záření po fotochemickém zpracování. Vznikla z potřeby správného určení expozice za různých světelných podmínek.

4.2 Senzitometrická charakteristika

Senzitometrická charakteristika fotografického materiálu udává závislost optické hustoty na logaritmu expozice (součinu osvětlení a expoziční doby). Je základní charakteristikou fotografického materiálu. Získá se měřením vyvolaného standardního testu (senzitogramu), který se získá expozicí skrze senzitometrický modulátor (klín) v senzitometru za známých expozičních podmínek (intenzita a barevná teplota expozičního světla, expoziční doba). Závisí na spektrálním průběhu zdroje světla, na senzibilizaci emulze, druhu emulze a jejím stáří, na její tloušťce, teplotě vyvolávacích lázní, koncentraci vývojky, době vyvolávacího procesu atd.



Obr. 4.3 Senzitometrická charakteristika

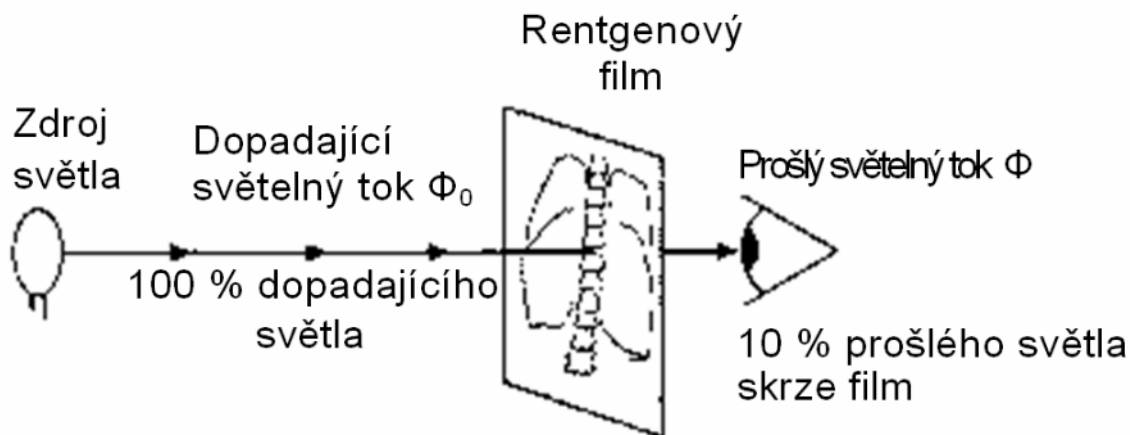
4.3 Kontrola zpracování

Z důvodu jednoduchosti je možné kontrolu zpracování omezit na měření několika polí exponovaných za standardních podmínek, nemít tedy úplnou senzimetrickou charakteristiku zkonstruovanou z mnoha bodů. To umožní zmenšit rozměry exponované plochy v senzimetru a podstatně zjednodušit jeho koncepci. Rovněž denzitometrické vyhodnocení se podstatně zjednoduší. Měření pole se volí zpravidla tyto: jedno odpovídající hustotě závoje (D_{\min}), jedno odpovídající maximálnímu zernání (D_{\max}), expozice dalších dvou polí jsou voleny tak, aby jedna byla poblíž expozice pro kritérium citlivosti (měří se relativní citlivost) a další směrem k vyšším hustotám tak, aby zernání ještě leželo v lineární části senzimetrické charakteristiky. Rozdíl hustot těchto dvou polí charakterizuje relativní gradient. V extrémním případě ani není třeba z těchto bodů cokoli počítat; jsou-li vhodně zvoleny na senzimetrické křivce, dostatečně ji charakterizují a měření se může zredukovat na toleranci měření optických hustot. Za předpokladu stejného materiálu lze polohu kritériálních bodů nastavit vhodnou expozicí; systém ovšem

nemusí vyhovovat při velkých rozdílech v citlivostech několika materiál, které je třeba porovnávat.

4.4 Zernání

Pro samotné vysvětlení pojmu „míra zernání fotografické vrstvy“ si představme pohlcující vrstvu, na kterou dopadne světelný tok Φ_0 a po průchodu z ní vystupuje světelný tok Φ . Tomuto poměru Φ / Φ_0 se říká koeficient prostupu. Logaritmus tohoto poměru je mírou zernání – optickou hustotou (obrázek 6). Dopadá-li na fotografickou vrstvu světelný paprsek, je vrstvou absorbován a rozptylován. Proto je jeho intenzita nižší. Mírou zeslabení je logaritmus (dekadický) poměru intenzity dopadajícího a prošlého paprsku. Například prochází-li 1/100 dopadajícího světla, je optická hustota $D = 2$, při průchodu 1/1000 dopadajícího světla je optická hustota $D = 3$ atd. Optická hustota (densita) se měří densitometrem v procházejícím světle (filmy) nebo odraženém světle (papíry).



Obr. 4.4 Optická hustota

Velikost zernání fotografické vrstvy je dána jejími vlastnostmi, množstvím záření, které na ni působilo, ale také způsobem zpracování. Pro viditelné záření je možno pro expozici použít název osvit (H). Osvit (H) je dán součinem osvětlení (E) fotografické vrstvy a doby (t), po kterou světlo na vrstvu působí: $H = E \times t$. Jednotka osvitu je luxsekunda (lx·s). Z daného vzorce vyplývá, že je totožné, zda film osvítime po dobu 1 s světlem o 10 lx, nebo 1 lx po dobu 10 s, dležitý je konstantní součin $H = E \times t$. Tomuto jevu se říká reciproční zákon. Při velkých rozdílech v expozici tento vztah přesně neplatí, což dokázal Schwarzschild, po kterém je pojmenován

exponent z následujícího vzorce vyjadujícího odchylku od recipročního zákona: $D = f(E \times t^p)$, kde p je tzv. Schwarzschildův exponent, který dosahuje hodnot 0,7 až 0,95.

Pro rentgenové záření je zernání závislé na ozáření E_e a času t , po který záření působí: $H_e = E_e \times t$.

4.5 Závoj, maximální hustota

Neexponované místo senzitogramu vykazuje minimální zernání, které má být co nejnižší (závoj, D_{min}). Maximální zernání je dáno především obsahem Ag ve vrstvě a dosahuje maxima (saturace) a nemění se při stoupající expozici. Má být co nejvyšší (vysoký dynamický rozsah materiálu).

4.6 Citlivost, gradient

Citlivost je základní vlastností fotografického materiálu, která je definována převrácenou hodnotou expozice potřebné pro dosažení prahového, normou definovaného zernání. Vyjaduje se jako aritmetická nebo logaritmická citlivost.

Například materiál o citlivosti $S = 200$ je 2x citlivější než o citlivosti 100 (aritmetická citlivost). Vyjadeno v logaritmické citlivosti materiál o citlivosti 20 DIN je 2x citlivější než 17 DIN (protože 1/10 poměru citlivostí je $\log 2 = 0,3$). U rentgenových materiálů se používá zpravidla aritmetická citlivost, u snímkových (negativních) materiálů aritmetická nebo logaritmická.

$$S = \frac{1}{\log H_K}$$

Gradientem se nazývá podíl přírůstku optické hustoty D odpovídající přírůstku logaritmu osvitů $\log H$.

$$G = \frac{D_B - D_A}{\log H_B - \log H_A}$$

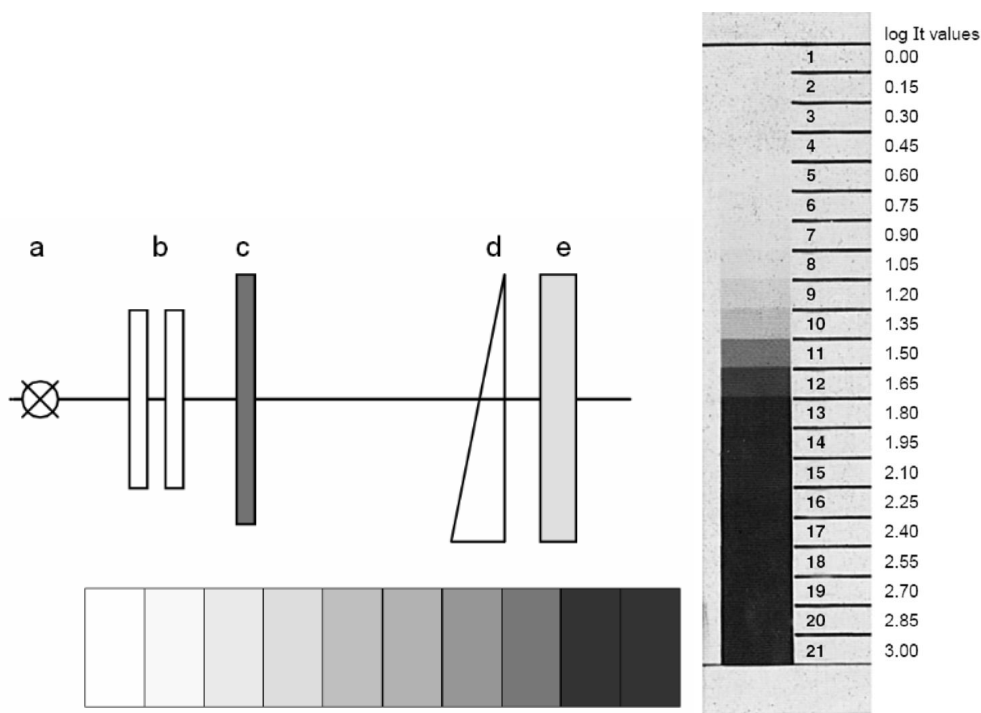
Z Obr. 4.3 je patrné, že kritéria K_s (citlivost) a A a B (gradient) jsou pro každý typ materiálu definovány jinak příslušnou normou.

5. Princip senzimetru

5.1 Senzimetrie

Senzimetr je expoziční zařízení, které slouží pro definovanou expozici fotografických materiálů (film a papír). Po zpracování je vyvolaný materiál – senzitogram proměřen denzitometrem a z naměřených údajů se vypočítávají senzimetrické parametry (závoj, maximální zernání, gradient, citlivost).

Senzimetr je základním přístrojem, který slouží pro kvalitativní kontrolu fotografických materiálů jak při jejich vývoji, tak při výrobě a kontrole zpracování.



Obr. 5.1 Princip senzimetru

Skládá se z následujících částí:

- a) Světelný zdroj se známou světlostí a teplotou chromatičnosti. Toho se dosahuje napájením žárovky známým konstantním proudem, jehož hodnota byla ověřena při kalibraci. Vyšší barevné teploty (senzimetrické denní světlo) lze dosáhnout zařazením namodralého konverzního filtru. Stejně lze filtry simulovat i spektrální složení záření modrých a zelených zesilovacích fólií používaných spolu s rentgenovými filmy.

- b) Mezi světelný zdroj a modulátor se umístí závěrka umožňující expozici. Závěrka může být padací, štrbinová, elektronická.
- c) Spektrální filtr.
Filtr může být dvojího druhu: konverzní (namodralý), který ze žárovkového světla dává tzv. denní světlo, tj. odstraní modrou složku světla, kterou není možno dosáhnout zvyšováním proudu žárovky (vlákno by se při potřebném proudu přepálilo) a nebo barevný (pro zeleno a modrocitlivé filmy).
- d) V určité vzdálenosti, kterou lze ovlivovat velikost osvětlení v exponiční rovině, je modulátor (v tvaru stupňovitý klín, zpravidla s 21 poli o známé optické hustotě). Density jsou odstupňovány v logaritmické řadě. Do kontaktu s modulátorem se přikládá hodnocený filmový materiál.
- e) Film.

5.2 Rozbor vlastností

Na senzitivity jsou kladeny poměrně protichůdné požadavky: jiné jsou na přístroje, které mají sloužit při výrobě fotografických materiálů (film a papír), především pro odměřování jejich citlivosti při uvolnění materiálu k prodeji, jiné požadavky jsou kladeny pro kontrolu zpracování. Mají-li senzitivity splňovat požadavek univerzality, tedy použití jediného přístroje pro veškerý vyráběný materiál od nízkocitlivých kopírovacích papírů až po vysoce citlivé snímací filmy, je potřeba, aby bylo možno zajistit široký rozsah expozic, minimálně v rozsahu 10 sekund až 0,01 sekund, tedy nejméně v rozsahu tří řádů. Přitom na provozní senzitivity jsou kladeny další požadavky, zejména jednoduchá obsluha a snadná manipulace s exponovaným materiálem ve tmě, možnost volby různých světelných zdrojů, zejména tzv. senzitivní denní světlo ($T_c = 5400\text{ K}$), senzitivní žárovkové světlo ($T_c = 2850\text{ K}$ a 3200 K), barevné světlo simulující spektrální složení světla emitovaného modrými (λ_{max} asi 470 nm) a zelenými (λ_{max} asi 510 nm), možnost vkládání neselektivních šedých zeslabovacích filtrů atd. Z uvedeného vyplývá, že skutečně univerzální senzitivní přístroj je poměrně komplikovaný přístroj, jehož vlastnosti zůstávají pro jednoduché senzitivní úlohy jako je kontrola zpracovatelského procesu, nevyužity.

Z tohoto důvodu se v době, kdy se ve zdravotnictví začalo ve větší míře uvažovat o nějaké způsobu kontroly zpracování rentgenových medicijních filmů, začaly používat nejprve standardní snímky (fantomy), předexponované testovací snímky s možností jednoduchého vizuálního porovnávacího vyhodnocování bez nutnosti poizování přístroj. V posledních letech se jednoduché senzitivity pro kontrolu zpracování staly prakticky standardem ve vybavení většiny radiologických pracovišť.

Tyto senzitivity, jejichž popis je uveden výše, dále však nesplňují požadavky kladené na senzitometr pro výrobní účely, nebo jsou zjednodušeny pro jednu konkrétní aplikaci; a to zejména v následujícím:

- svítelný zdroj je tvořen elektroluminiscenční fólií, která při položení vysokého (100 – 250 V) střídavého napětí vhodné frekvence (50 Hz až 2 kHz) poskytuje podle typu použitého luminoforu spektrálně poměrně vhodné modré nebo zelené záření (případně nepřímé pro oba režimy) odpovídající používaným zesilovacím fóliím. Zdroj není vhodný pro materiál vyžadující expozici jiným než spektrálně úzkopásmovým svítelným zdrojem (senzitivitické denní nebo žárovkové světlo), což jsou vlastně všechny běžné fotografické filmy a papíry s výjimkou rentgenových
- použitím jednoduchého mechanicky ovládaného jednoduchým uspořádáním s jediným pevným expozičním zářivkem a velkou plochou fóliového zdroje umožňuje kompaktní konstrukci s malými rozměry a dobrým využitím svítelné energie, kdy exponovaný film je prakticky v kontaktu s modulátorem. Nerovnoměrnost jasu luminiscenční fólie nemusí být omezující pro relativní senzitivitii, nebo stejný senzitometr slouží k přípravě standardního senzitivogramu, ke kterému se kontrolní měření vztahují. Pro absolutní senzitivitii takové uspořádání nevyhovuje, zejména proto, že senzitivitickou expozici není možno vypočítat (není známa svítivost zdroje respektive povrchový jas, není vyhovující rovnoměrnost, není jasný dosvit fólie po přerušení napájení)

- bateriové napájení u těchto přístrojů vyžaduje nějakou podobu stabilizace, aby byl kompenzován úbytek svítivosti u vybitých baterií. Pro provozní použití, kdy je vyžadován velký počet expozic, není bateriové napájení ekonomicky zvládnutelné

6. Cíl práce

Firmy vyrábějící fotografické materiály si vyvíjejí kontrolní zařízení pro výrobní procesy samy. Na trhu univerzální senzitivity nejsou k dispozici. Dostupné jsou prakticky jen jednoduché přístroje pro kontrolu zpracovatelského procesu medicijních rentgenových filmů. V ČR existují a používají se přístroje ve stáří 20 a více let, které nesplňují předpoklady pro moderní kontrolní přístroje a požadavek univerzálnosti je u nich splaven jen částečně. Proto je cílem této práce pokusit se navrhnout univerzální provozní senzitivu, který by v co největší míře splňoval následující požadavky:

6.1 Obecné

Kompaktní přístroj vhodný pro jednoduchou obsluhu (práce ve tmě). Přístroj musí obsahovat pro expozici materiál v šířce 35 mm resp. ve formátech. Expozice přes senzitivní modulátor. S ohledem na pohodlné měření velikost pole modulátoru min. 5x15 mm, lépe 9,5x20 mm. Možnost instalace modulátorů dostupných na trhu. Klínová konstanta modulátoru $k = 0,15$. Zvolit modulátor co nejlépe neutrálně šedý. Přístroj by měl vyhovět nejméně pro černobílé materiály, lépe i pro barevné materiály (filmy a papíry)

6.2 Světelný zdroj

- senzitivní žárovkové světlo (teploty chromatičnosti $T_c = 2850$ K a 3200 K)
- senzitivní denní světlo ($T_c = 5400$ K), realizovat žárovkovým zdrojem a konverzním filtrem
- modré světlo odpovídající emisnímu maximu modrých zesilovacích fólií (asi 410 nm)
- zelené světlo odpovídající emisnímu maximu zelených zesilovacích fólií (asi 560 nm)

Žárovkové světlo pro materiály určené pro kopírování a zvětšování (papíry, fototechnické filmy, kinopozitivní filmy apod. a pro barevné papíry, denní světlo pro snímací materiály (černobílé a barevné negativní filmy), modré a zelené světlo pro expozici odpovídajících rentgenových filmů určených pro použití s fóliemi.

Světelný zdroj bude realizován vhodnou žárovkou s regulátorem řízeným PWM z řídicího procesoru, případně kombinovaným zdrojem (žárovka – LED diody).

6.3 Filtry

Realizovat např. karuselem nebo podobným zařízením s nejméně 4 pozicemi

- bez filtru
- konverzní filtr 5 400 K
- šedý zeslabovací filtr $D = 1$
- rezerva

Platí v případě, že bude jako zdroj barevných světel pro simulaci rentgenové expozice s fóliemi použit další zdroj s LED. Pokud to nebude reálné, použít další pozice s modrým a zeleným filtrem.

Karusel se servoelektrickým ovládním řízeným z procesoru.

6.4 Závěrka

Pro umožnění protichůdných požadavků (velký rozměr senzigramu; požadavek na kompaktnost) realizovat závěrku jako štěrbinovou (posun štěrbiny podél exponované plochy souasně se světelným zdrojem). To přinese lepší využití světla a s tím související snížení potřebného výkonu. Změny expozice bude dosaženo změnou posuvné rychlosti, případně v kombinaci se změnou šířky štěrbiny. Pro posun bude použito lineární vedení, servomotor nebo krokový motor s regulací z procesoru. Požadovaná expoziční doba volně programovaná v rozsahu minimálně 0,05 s – 10 s (lépe 0,01 – 40 s)

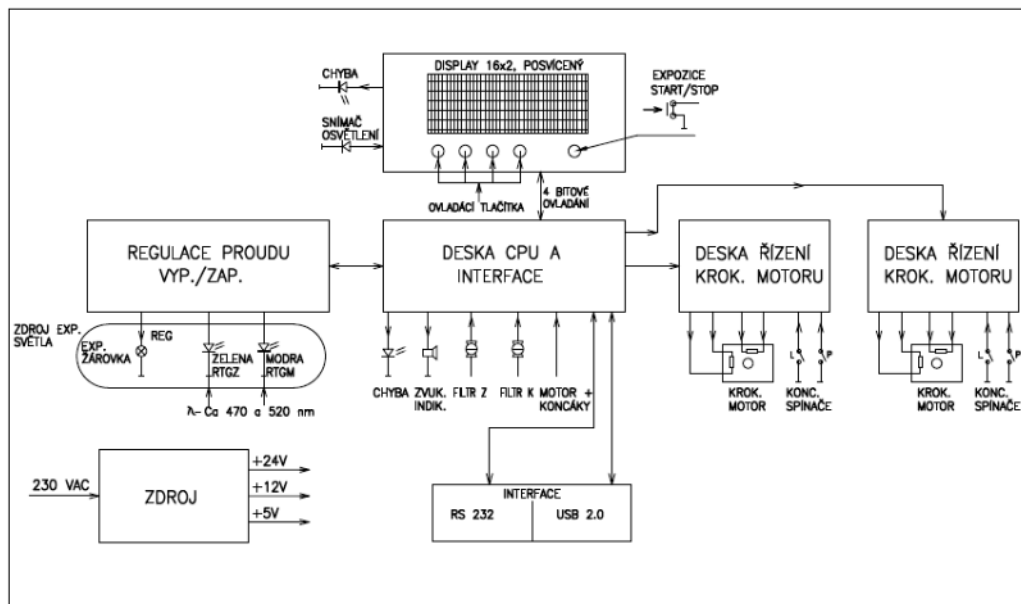
6.5 Uživatelský interface

Vybavení podsvíceným znakovým displejem (např. 2 x 16) a tlačítkovým ovládním. Pro každý typ exponovaného materiálu vybrat tlačítka označení, v druhém řádku zobrazit expoziční parametry (označení světelného zdroje nebo Tc, filtr, expoziční doba). Tabulku expozičních parametrů bude možné nahrávat do paměti procesoru prostřednictvím USB nebo RS232. Minimálně 10 kombinací expozice (materiálových typů). Možnost změny a upgrade firmware procesoru. Po výběru expoziční kombinace možnost přepnutí z programovacího do expozičního

režimu (zabrání nežádoucího nastavení expozice ve tmě). Vlastní ovládní jedním tláčením (start/stop). Akustická indikace ukončení expozice.

Možno doplnit ovládním podsvícení displeje v závislosti na osvětlení ve fotokomóře (zabrání nežádoucí expozici rozbaleného materiálu indikačními kontrolkami a displeji přístroje).

7. Návrh řešení



Obr. 7.1 Blokové schéma

7.1 Koncepce senzimetru

- Senzimetř je určen pro jednostrannou expozici fotografických filmů a papírů ve standardní šířce 35 mm. Je preferováno provedení se stojícím exponovaným vzorkem, které umožní expozici senzigramu v okraji plochých filmů nebo na pátku filmové role.
- S ohledem na pohodlnost měření byl zvolen stupňovitý modulátor s 20 poli po 10 mm, případně s 21 poli po 9,5 mm. Příčný rozměr exponovaného pole je 20 mm. Senzigram bude exponován v ose filmového pásu 35 mm. Tyto modulátory jsou ve skleněném nebo filmovém provedení relativně dostupné.
- Exponovaný film se přitiskne do kontaktu s modulátorem pomocí příklopného víka s pružnou světlo absorbující podložkou přiléhající k zadní straně filmu (bez emulze).

7.2 Celkové konstrukční uspořádání

Při konstrukci byla použita typizovaná skříňka, zakládání filmového materiálu zepedu, v horní polovině předního panelu rámeček pro senzigram s modulátorem (21 x 9,5 mm apod.), ve spodní polovině ovládací prvky a displej.

Pro jednotlivé druhy citlivých materiálů předprogramované parametry (barevná teplota, druh světelného zdroje, proud žárovky, filtry, expoziční doba), obsluha bude volit jen označení druhu na displeji. Ovládání expozice tlačítkem start/stop bude možno použít k přerušení u dlouhých expozic. Použité světelné indikační prvky a podsvětlení se centrálně vypnou po zhasnutí v temné komoře na základě signálu z idla. Použité parametry se do přístroje přenesou po sériovém kanálu (na každém pracovišti se mohou zpracovávat jiné materiály). Do budoucna se uvažuje s přenosem USB.

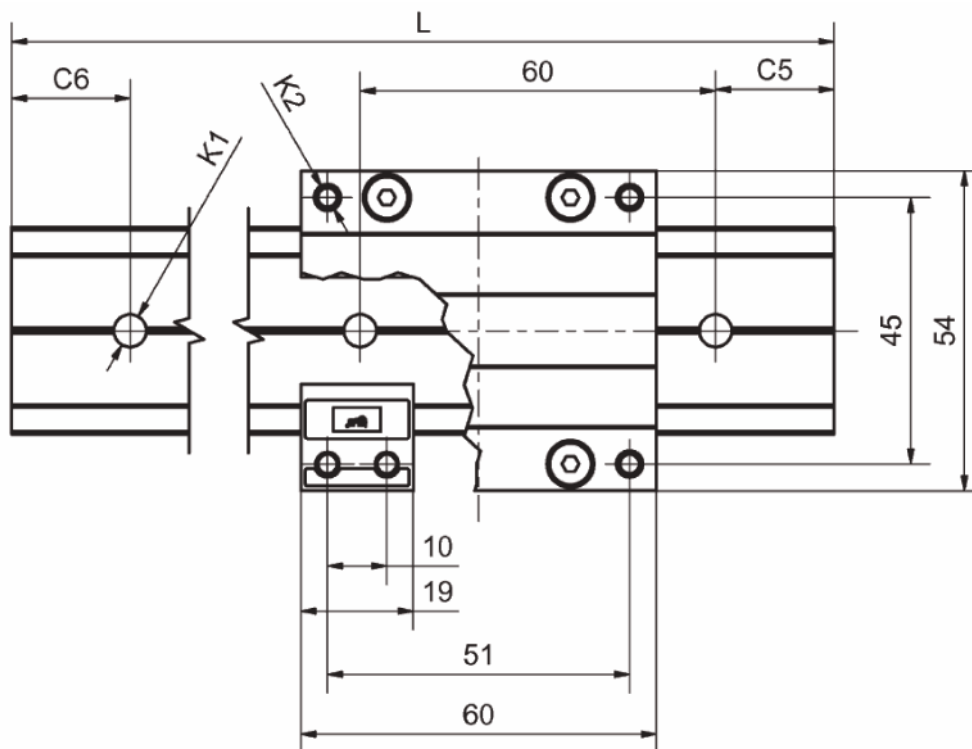
8. Mechanická část senzimetru

Mechanicky se senzitometr skládá z několika celků, a to:

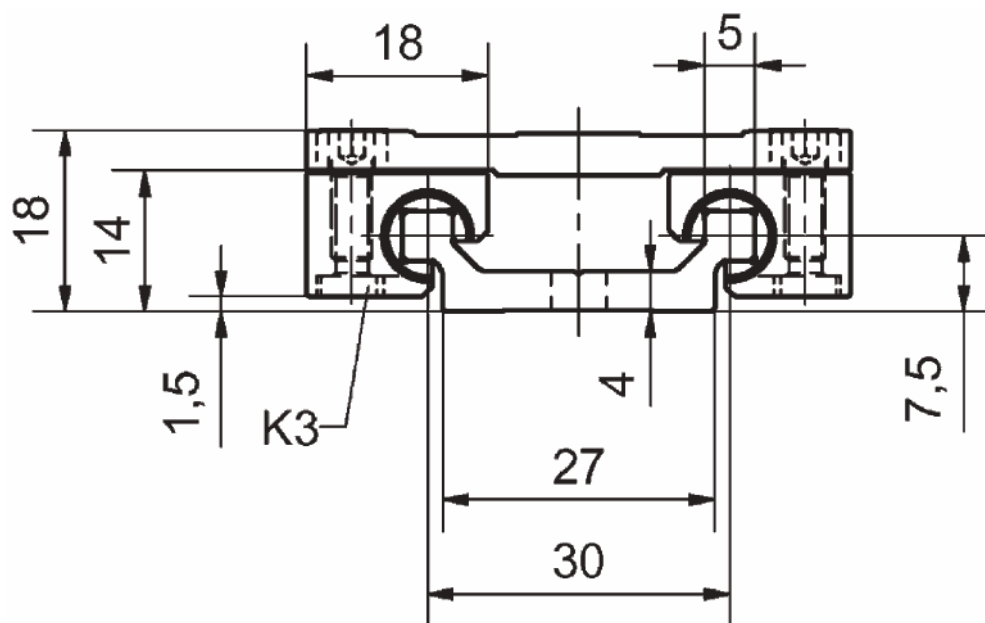
- lineární karusel v etn pohonu
- držáku filtr v etn držáku žárovky a záv rky
- ráme ek pro senzitogram s modulátorem a p ídržným víkem pro film , nebo fotopapír

8.1 Lineární kluzné vedení

Lineární karusel je společně se světelným zdrojem a záv rkou (št rbinou) umíst ěn na lineární kluzné vedení DryLin W od fi rmy Hennlich Industrietechnik spol. s r. o.. Jedná se o vysokovýkonné bezúdržbové vedení umož űující vysokorychlostní pohyb až do 30 m/s s minimálním kluzným odporem. Profilovaná kolejnice je vyrobena z eloxovaného hliníku. Kluzný vozík je též vyrobený z eloxovaného hliníku a je osazen tve ící kluzných element , která jsou uspo řádána v párech proti sob ě , společně pracujících jako dv ě vodící ložiska. Každý z nich je možné samostatn ě ru ě nastavit a tím vymezit v ě le v ě vedení a nastavit t ečí sílu.



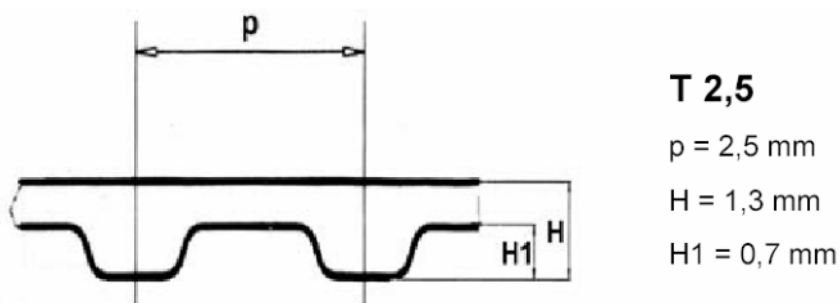
Obr. 8.1 Lineární vedení – pohled shora



Obr. 8.2 ez lineárním vedením

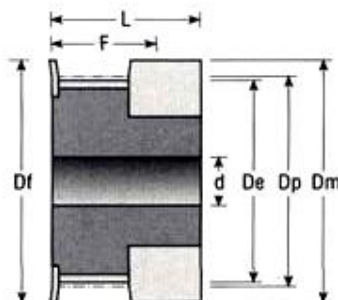
Krajní polohy vozíku (bezpe nostní omezení pohybu) a krajní pracovní polohy jsou snímány mikropsína i. Protože žárovková sk í je p ipevn na na sestavu filtr , jsou signály t chto idel vyvedeny pohyblivým p ívodem, obdobn jako napájení žárovky.

Pro p enos pohybu z motoru na lineární vedení byl použit polyuretanový ozubený emen. Je ideální pro použití v oblastech miniaturních pohon , vysoce p esných polohovacích a t ídicích mechanism , malých ru ních strojk . Vynikající spojení vysoce odolného polyuretanu a délkov stálého ocelového tažného vlákna dává emen m vynikající výkonnostní p edpoklady. Jsou odolné v i ot ru a bez nutnosti údržby.



Obr. 8.3 Polyuretanový ozubený emen a jeho rozm ry

Motor byl osazen ozubenou emenicí T2,5 ze hliníkové slitiny. Emenice má 16 zubů a při jedné otáčce motoru posune lineární vedení o 38,75 mm. S ohledem na rychlostní potřebu posunu a možnostech navrhovaného krokového motoru je toto dostačující.



Obr. 8.4 Ozubená emenice T2,5

Zub počet	Šířka	Označení / Typ	Mat.	De	Df	Dm	F	L	d	m
				mm	mm	mm	mm	mm	mm	kg
16	6	16 T 2,5/16-2	Al	12,20	16,00	16,00	9,00	16,00	5,00	0,005

Tabulka 1 – Katalogové údaje ozubené emenice

8.2 Držák filtr v etn držáku žárovky a závorky

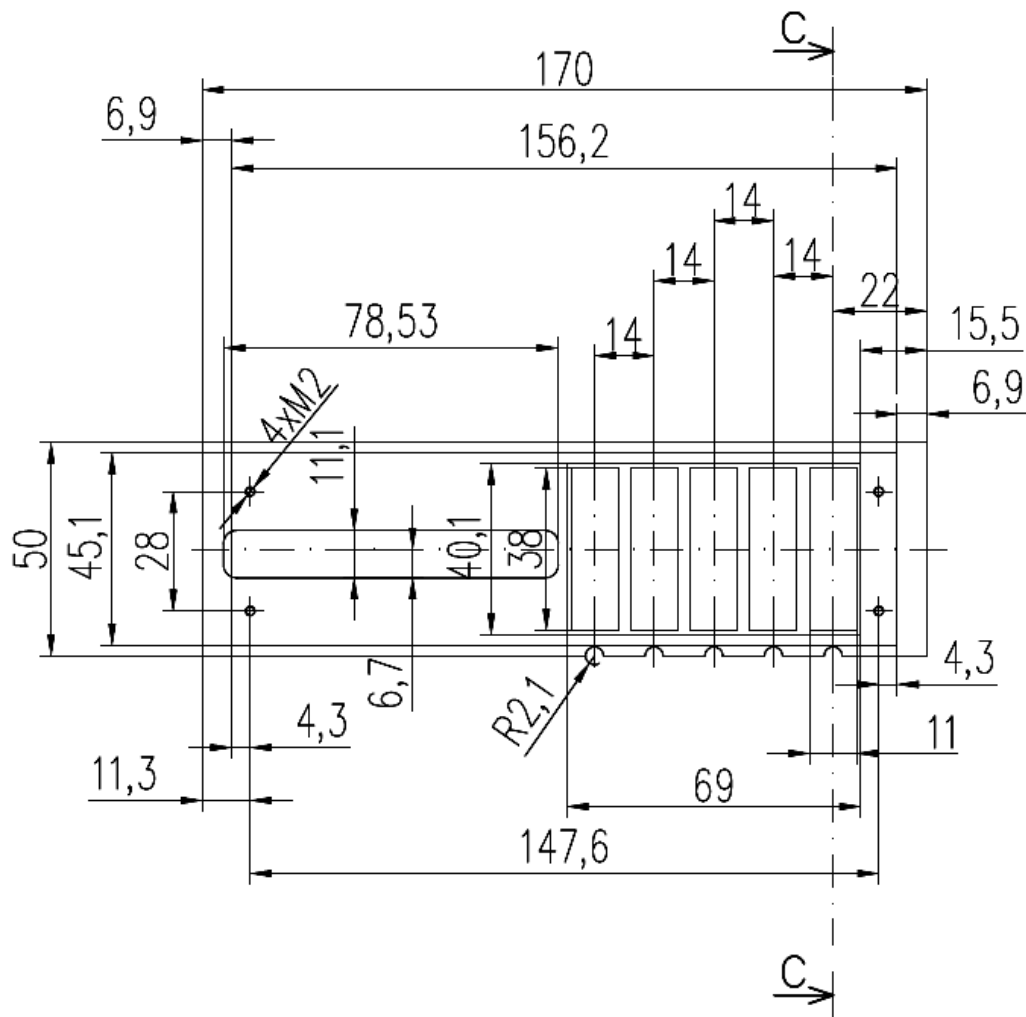
8.1.1. Mechanická část filtru

Aby senzitometr splňoval podmínku univerzality, musí být osazen filtry. Filtry jsou zabudovány do jednoho rámečku po potě, a to v pořadí:

- a) Filtry šedé zeslabovací
 - D = 0,6
 - D = 1,2
- b) Filtry barevné
 - konverzní filtr 5400 K
 - modrý filtr
 - zelený filtr

V případě, že by se senzitometr osadil kromě žárovkového svítidla také LED diodami (modrou a zelenou v odpovídajícím spektru kopírující spektrum zesilujících fólií) budou z rámečku vyjmuty dva filtry, a to modrý a zelený a nahrazeny šedými zeslabovacími filtry pro větší univerzality.

Držák filtru je vyroben s ohledem na univerzálnost použití, což zaručuje vložení jakéhokoliv filtru o maximálních rozměrech 13 x 40 x 10 mm. Na spodní straně držáku jsou vyhloubeny drážky pro aretační kulíčky. Jejich funkce se zprvu zdála zbytečná, jelikož posun zajišťuje krokový motor, ale po důkladném zvážení bylo usouzeno, že je lepší správnou polohu zajistit mechanickou aretací, než-li v klidovém stavu napájet krokový motor malým proudem.

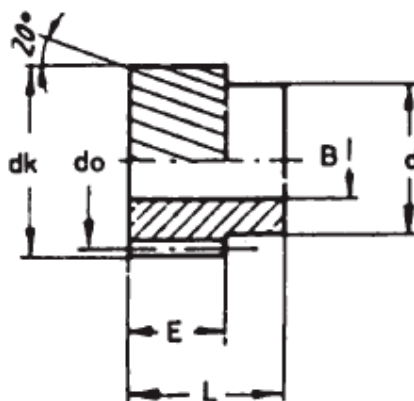


Obr. 8.5 Držák filtru

Instalace filtru se provádí nejmenším krokovým motorem SX16-0301 od firmy Microcon. Jeho statický moment je 0,11 Nm, což vzhledem na minimální třecí odpor je dostačující. Nejvyšší odpor klade mechanická aretace při rozjezdu. Na hřídel krokového motoru je narážena omešnice její parametry jsou v tabulce 2.

Typ	Modul	Počet zubů	B	d	dk	do	L	E
SN 0324 MS	0,3	24	3	7	8,2	7,6	9	5

Tabulka 2 – Rozměry čerpadla



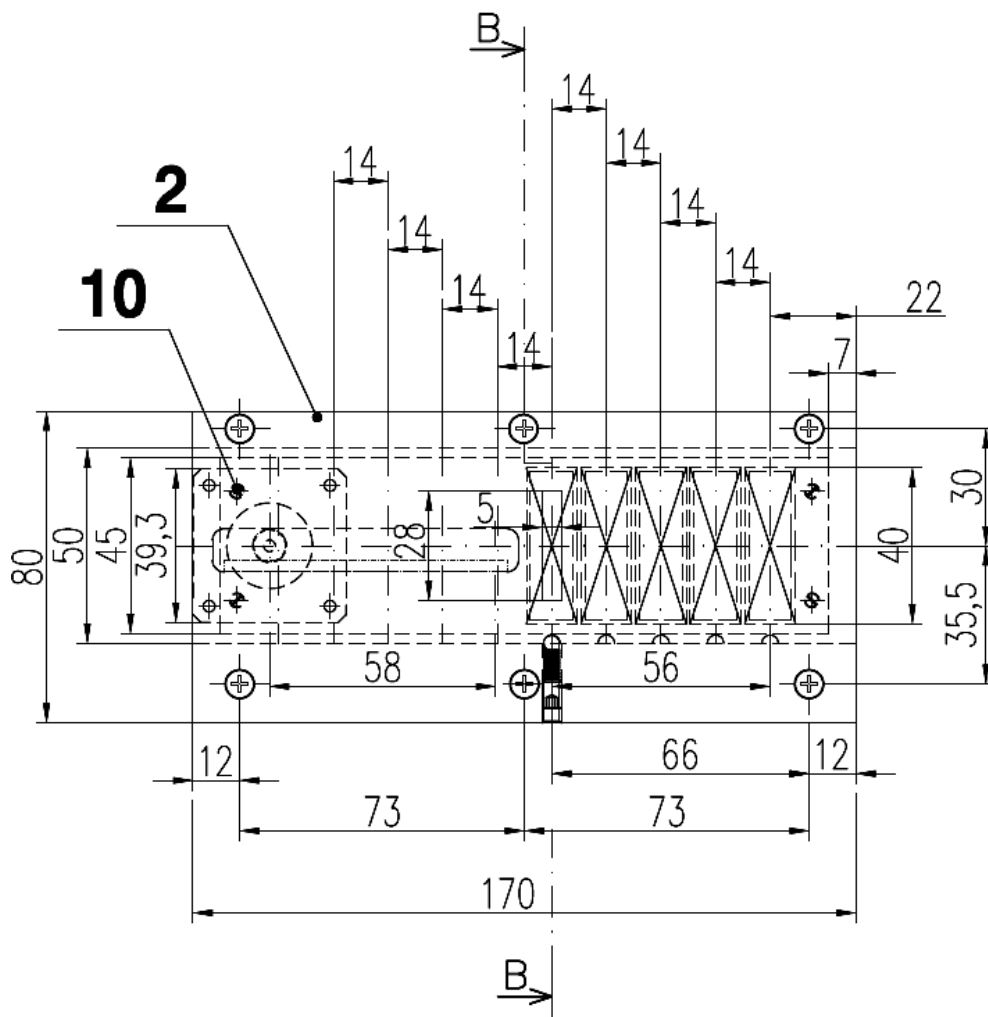
Obr. 8.6 Čerpadlo pohony rámečku filtru

V rámečku v jeho levé krajní poloze je vyfrézovaná drážka, do které je zabudován ozubený hřídel. Ozubený hřídel s šikmým ozubením je vyrobený z mosazi MS 58 a je frézován. Má šikmé ozubení a levý sklon zubů. Úhel záběru je 20°. Jako protikus k tomuto hřídelu na straně motoru je ozubené kolo s šikmým ozubením m-0,3.

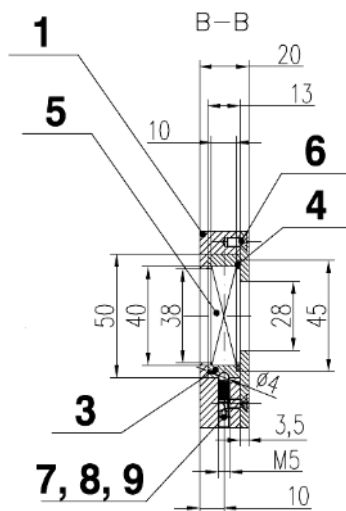


Obr. 8.7 Reálná podoba

Rámečky osazené filtry se budou pohybovat ve speciálním vedení, které jsou vyrobeny z lehké slitiny. Ve spodní straně vedení je umístěn aretní mechanismus zajišťující správnou pozici filtru. Skládá se z kuličky o průměru 4 mm, přítlačné pružiny a zapuštěného imbusového šroubu M4 x 8 mm. Imbusovými šrouby se nastavuje síla přítlačku, která je závislá na stavu opotřebení dosedacích ploch (zápichy na spodní straně rámečku filtru) a celkového tloušťkového odporu vedení. Skutečná poloha filtru je snímána mikroskopem.



Obr. 8.8 Konstrukce lineárního karuse lu – elní pohled



Obr. 8.9 Lineární karusel v ezu

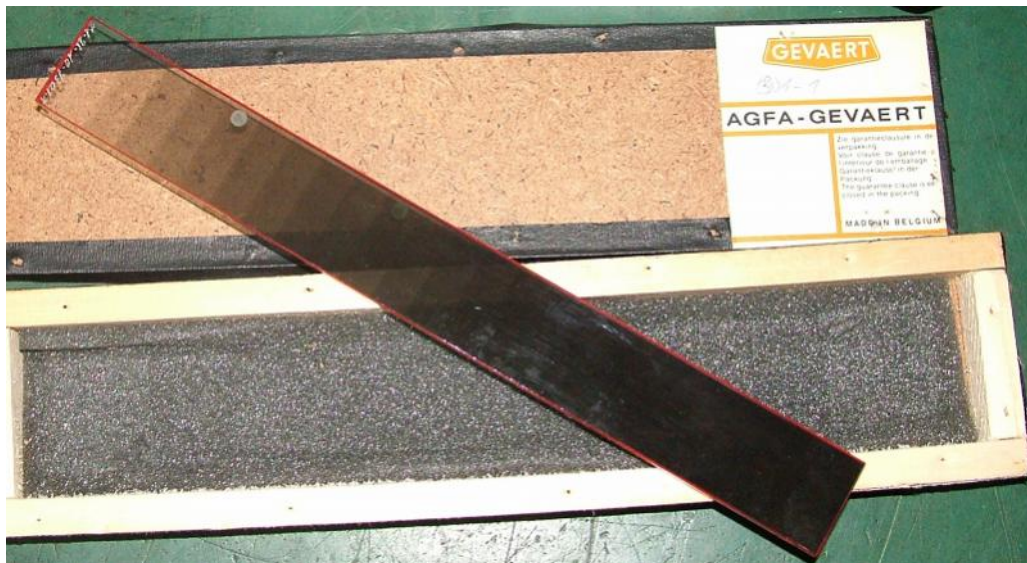
8.3 Závěrka

Závěrka je jednou z nejkomplicovanějších součástí senzimetru vzhledem k tomu, že pro pohodlné měření je požadován velký rozměr senzigramu při rozumných rozměrech senzimetru. V současně používaných senzimetrech při použití bodového zdroje světla (žárovka) bylo potřeba pro zachování rovnoměrného osvětlení modulátoru velkých mechanických rozměrů přístroje, a to vedlo k požadavku na velký úžinný otvor závěrky. Pro umožnění protichůdných požadavků, a to velký rozměr senzigramu a požadavek na kompaktnost přístroje, je závěrka realizována jako štrbinová. Posun štrbiny podél exponované plochy současně se světelným zdrojem a filtry je prováděn co možná nejplynuleji. To umožňuje lepší využití světla a s tím související snížení potřebného výkonu. Aby frekvence kroku nebyla pozorovatelná jako nehomogenita zernání, je šířka expoziční štrbiny zvolena mnohonásobně větší než velikost kroku. Změny expozice bude dosaženo změnou posuvné rychlosti, případně v kombinaci se změnou šířky štrbiny. Pro posun je použito lineární vedení, krokový motor s regulací z procesoru. Požadovaná expoziční doba je volně programovaná v rozsahu minimálně 0,05 s – 10 s (lépe 0,01 – 40 s). Expozici je vhodné spojit s potřebným mechanickým přitisknutím exponovaného materiálu k modulátoru.

8.4 Modulátor

Modulátor slouží k vytvoření stupnice definovaných šedých plošek na exponovaném materiálu - senzimetrického klínu. Zpravidla se používají modulátory stupňové s klínovou konstantou 0,15. U grafických materiálů s velkým kontrastem může být užitečné použít modulátor s konstantou 0,1. Běžné densitometry pro transparentní materiály mívají rozměrné plošky 3 mm, pro pohodlné měření tedy vychází šířka jednoho stupně 5 mm, s výhradami 4 mm. Na spektrální selektivitu modulátoru vzhledem k úzkopásmovým světelným zdrojům nejsou kladeny velké požadavky, rovněž rovnoměrnost modulátoru vzhledem k oblasti použití a zpodobnění (relativní porovnává ní senzigram exponovaných jedním senzimetrem) není kritická. Prakticky je možno použít libovolný modulátor.

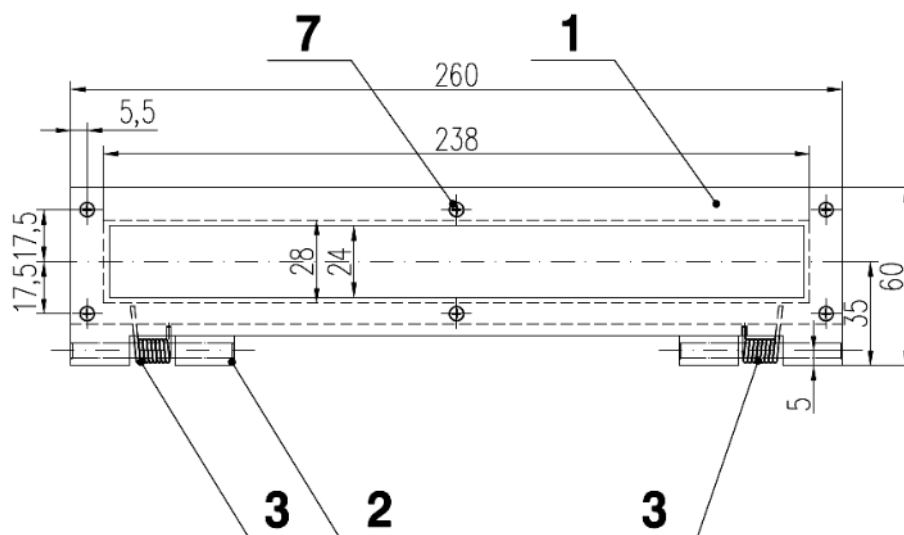
Pro zkušební senzitivometr byl zvolen stupňovitý modulátor od firmy AGFA. Jeho klínová konstanta je 0,15 a má 21 polí po 9,5 mm. Fyzické rozměry vybraného modulátoru jsou 28 x 238 x 4 mm.



Obr. 8.10 Modulátor AGFA

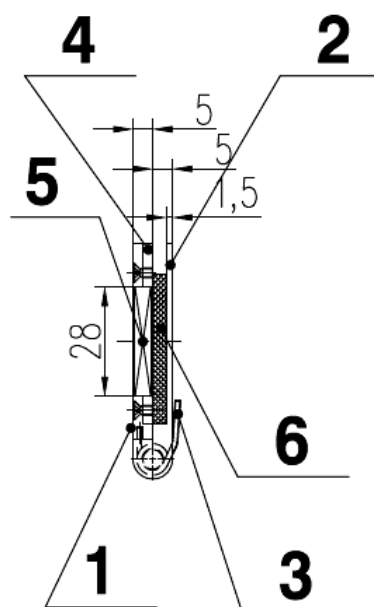
Modulátor je vhodné spojit se šablonou s označením po adových ísel exponovaných polí nebo značkami usnadujícími orientaci v naexponovaném senzitivogramu a chránit ho proti mechanickému poškození; materiál musí být v dobrém kontaktu, což lze docílit pružnou vložkou v přítlačném elementu (molitanu). Pro jednoduchost je vhodné mechanicky spojit dosažení dobrého kontaktu s odstartováním expozice, s čímž se do budoucna počítá.

Při konstrukci senzitivometru bylo třeba zajistit, aby senzitivogram mohl být naexponován kromě obvyklé šířky 35 mm rovněž na okraj formátu rentgenového filmu. To je důležité pro zpracování v medicíně, kde malé formáty klasického senzitivogramu se obtížně transportují v kovovém pohonném systému.



Obr. 8.11 elní pohled sestavy držáku modulátoru a filmu

Na Obr. 8.11 je znázorněn elní pohled sestavy držáku modulátoru a filmu. Modulátor je vložen do dvou rovnocenných vyfrézovaných drážek, které jsou navzájem prošroubované šroubky M 2,5 mm. Venkovní p ítlaké víko je k hlavnímu držáku p ípevn no p es panty, které jsou vybaveny p ítlakým i pružinami. Jejích síla p ítlaku je dostate ná, aby udržela film o v tších rozm rech. Na vnit ní stran je nalepena vrstva p nového polyuretanu, která zajiš uje dokonalý p ítlak emulzní strany k modulátoru a zárove absorbuje sv tlo, p ípadn prošlé skrz vrstvy materiálu.

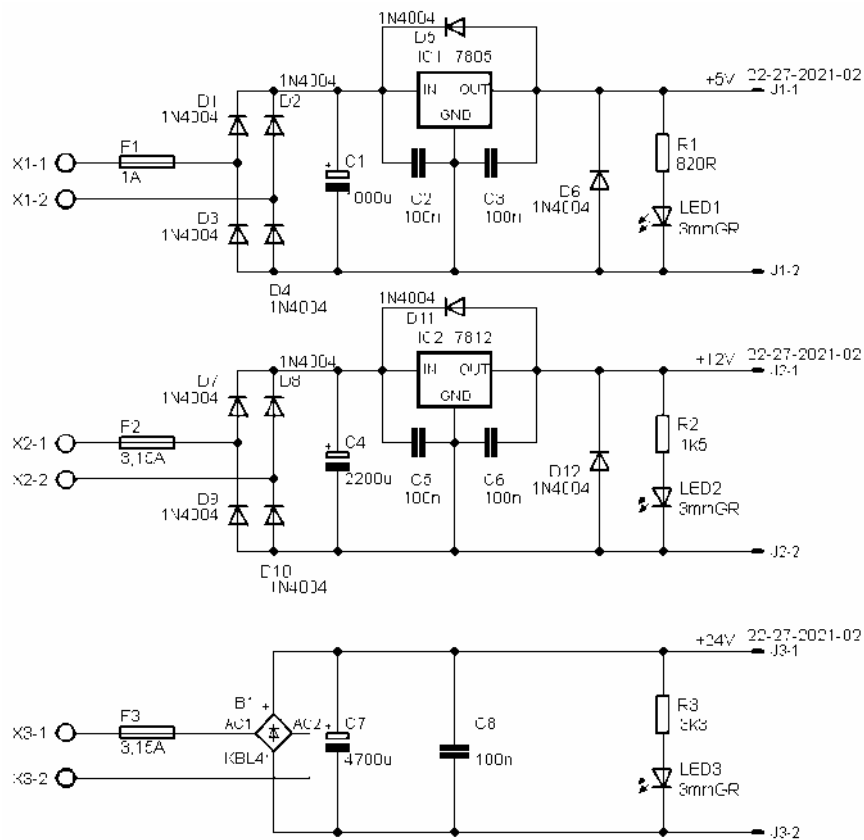


Obr. 8.12 Bo ní ez držáku

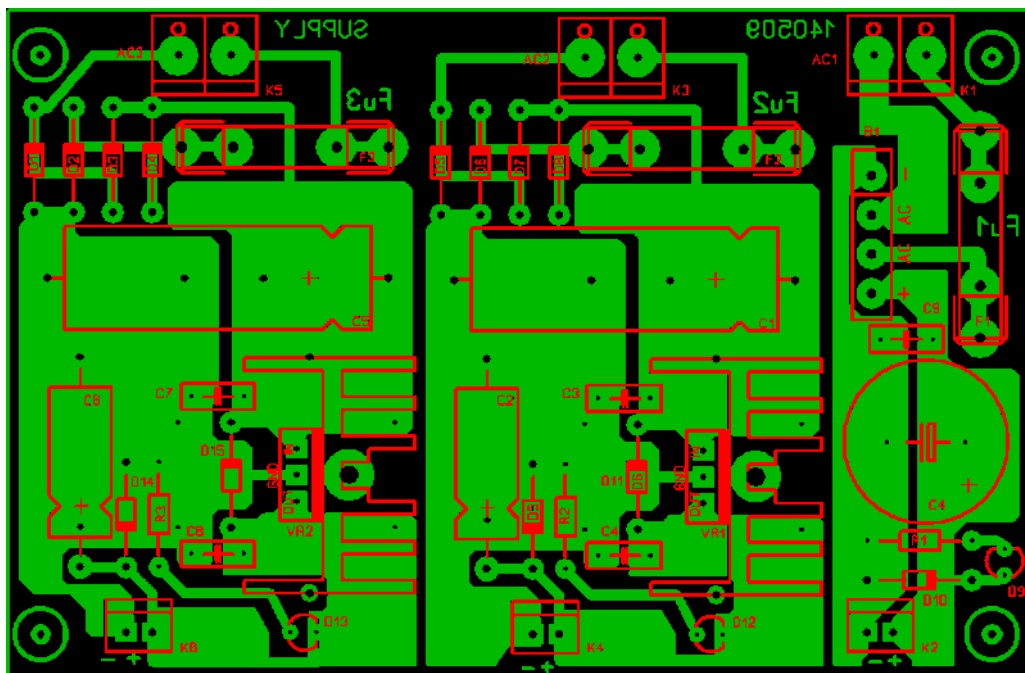
9. část elektro

9.1 Napájení

Pro napájení celého senzimetru byl navržen zdroj, který zajišťuje všechna potřebná napětí, a to jak pro motory, tak pro elektroniku. Napájecí transformátor je umístěn mimo plošný spoj. Pro napájení řídicích obvodů a obvodu displeje se používá 5 V, které zajišťuje stabilizátor 7805 v horní části schématu na Obr. 9.1. Uprostřed téhož schématu je stabilizátor 7812, který zajišťuje 12 V pro napájení žárovky. Ve spodní části je pro napájení krokových motorů použit nestabilizovaný zdroj, který je pro tyto účely dostačující.



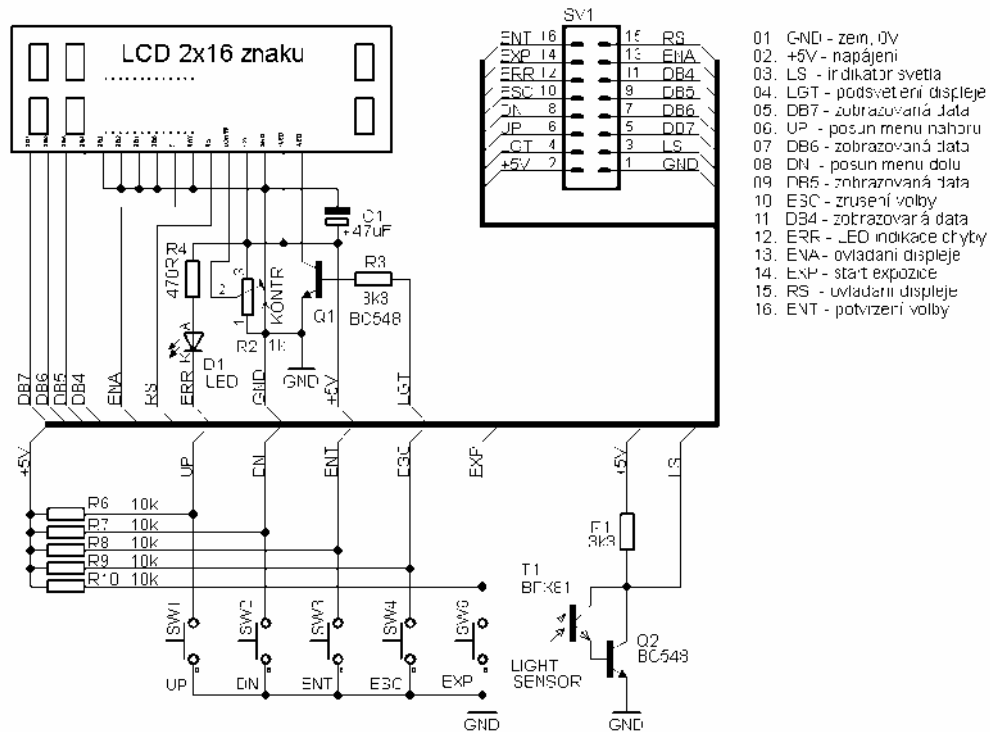
Obr. 9.1 Schéma napájecího zdroje



Obr. 9.2 plošný spoj napájecího zdroje

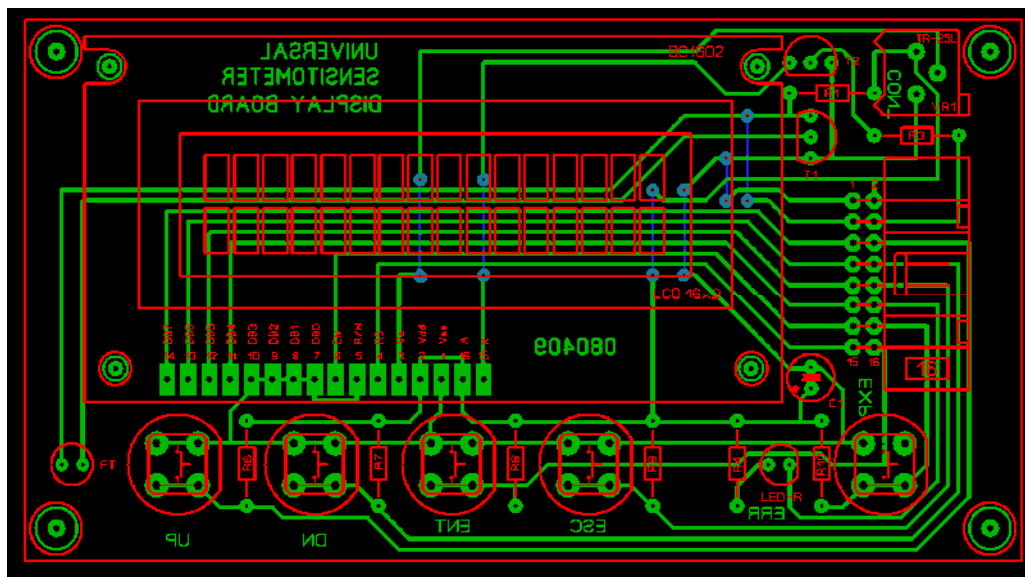
9.2 Ovládání

Senzitometr je vybavený podsvětleným znakovým displejem (2 x 16) a tlačítkovým ovládáním. Pro každý typ exponovaného materiálu je možné vybrat tlačítkové označení, které se bude znázorňovat v prvním řádku, v druhém řádku budou zobrazeny expoziční parametry (označení světelného zdroje nebo Tc, filtr, expoziční doba). Tabulku expozičních parametrů je možné nahrávat do paměti procesoru prostřednictvím RS232. Do budoucna se uvažuje i s USB připojením (což znázorňuje blokové schéma). Minimálně 10 kombinací expozičních (materiálových) tisků. Koncept senzimetru umožní změny a upgrade firmwaru procesoru, což usnadní použití i pro novější typy materiálů. Po výběru expoziční kombinace možnost přepnutí z programovacího do expozičního režimu (zabránil nežádoucího přestavení expozice ve tmě).



Obr. 9.3 Ovládací deska s LCD displejem

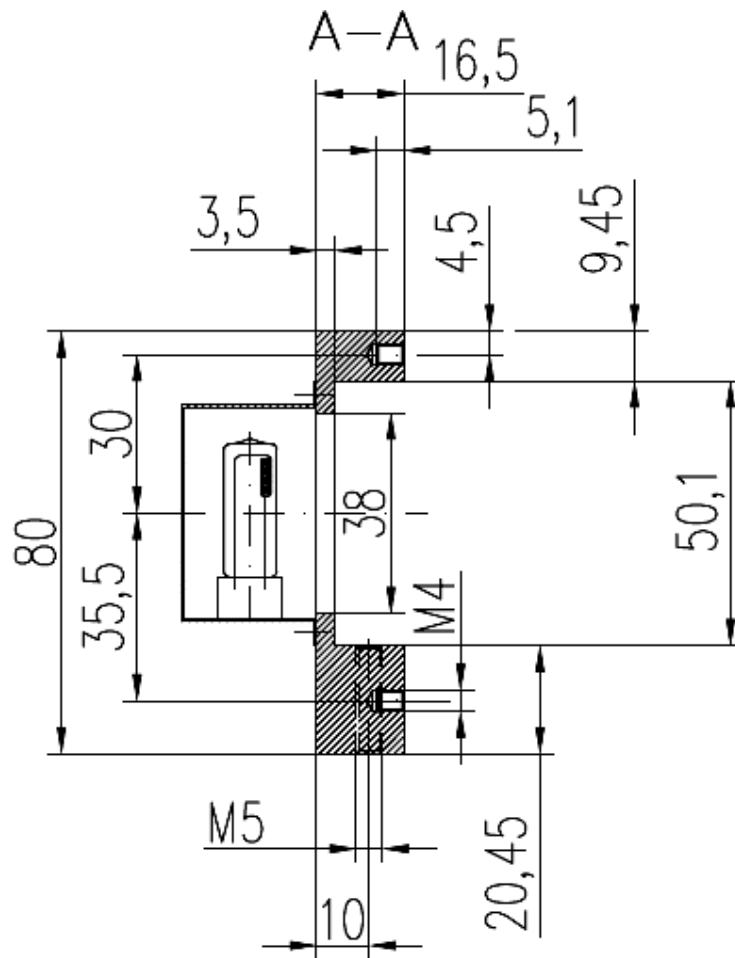
Vlastní ovládání expozice senzitogramu se zahajuje jedním tlačítkem start/stop – SW6. Po ukončení expozice se ozve akustická indikace. S ohledem na podmínky využití je senzitometr osazen fototranzistorem – T1 (BPX81), který nepřímo ovládá podsvětlení displeje v závislosti na osvětlení ve fotokomorbě (zabraňuje nežádoucí expozici rozbaleného materiálu indikacemi kontrolkami a displeji přístroje). LED dioda D1 slouží jako indikace chybové hlášky generované z procesoru. Trimmer VR1 slouží k nastavení kontrastu displeje. Z důvodů rozmanitosti podmínek použití bude nastaven přímo v místě používání s ohledem na pracovní podmínky.



Obr. 9.4 Plošný spoj ovládací desky s LCD displejem

9.3 Osv tlovací systém

Výhodou štrbinového osvětlovacího systému je zajištění rovnoměrného konstantního osvětlení poměrně velké plochy modulátoru. Na rozdíl od současně používaných přístrojů s bodovým zdrojem světla, kde se film a modulátor osvětlují ze značné vzdálenosti a kde je proto potřeba výkonnější světelný zdroj, zde bude použita pohyblivá osvětlená štrbina a stojící film a modulátor. Pohyb štrbinového osvětlovacího systému, jak je již zmíněno výše, zajišťuje krokový motor s náhonem na osvětlovací vozíku (se světelným zdrojem a osvětlenou štrbinou) pomocí ozubeného mechanismu. Žárovka, deternální filtr, konverzní a barevný filtr budou upevněny společně na pohyblivém vozíku. Výhoda tohoto řešení spoívá v tom, že žárovka i štrbina budou blízko modulátoru, lepší využití světla, žárovka bude s nízkým výkonem, což zaručí jednoduchou regulaci proudu. Toto řešení umožní vyzkoušet kromě žárovkového zdroje také LED světelný zdroj alespoň pro expozici rtg. film (simulace modré a zelené zesilovací fólie modrou a zelenou LED diodou).



Obr. 9.5 Umístění žárovky

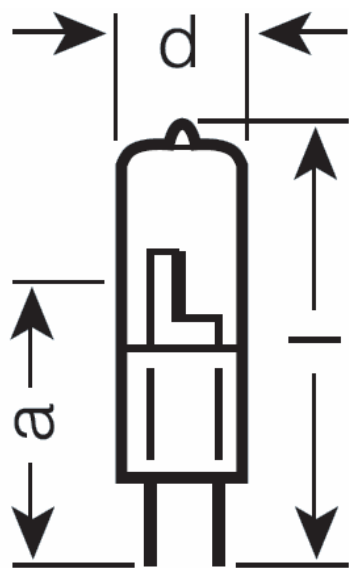
9.4 Sv telný zdroj

Navrhovaný senzitometr je obsazen:

- senzitometrické žárovkové sv tlo (teploty chromati nosti $T_c = 2850 \text{ K}$ a 3200 K)
- do budoucna se uvažuje o modrém sv tle odpovídající emisnímu maximu modrých zesilovacích fólií (asi 410 nm)
- a také o zeleném sv tle odpovídající emisnímu maximu zelených zesilovacích fólií (asi 560 nm)

Sv telný zdroj je realizován halogenovou žárovkou 10 W s regulátorem řízeným PWM z řídicího procesoru, případně se do budoucna počítá s kombinovaným zdrojem (žárovka – LED diody modrá a zelená).

Použitá halogenová žárovka 10 W je od firmy Osram pod označením HALOSTAR STARLITE 64415 S umožňuje provoz s ochranným krytem. Je stmívatelná a její teplota chromatičnosti je 3000 K. Průměrná životnost je cca 4 000 h a má axiální vlákno s optimálním směřováním světla.



Obr. 9.6 Halogenová žárovka 10 W

Osvětlovací žárovka je napájena regulovatelným konstantním proudem, který bude zvolen s ohledem na potřebnou teplotu chromatičnosti. Teploty chromatičnosti 5 400 K bude dosaženo pomocí konverzním filtrem. Žárovka je světlostně uzavřena v pouzdře z hliníkového plechu a v případě, že provozní zkoušky ukáží potřebu chlazení, bude pouzdro doosazeno vyfrézovaným chladičím profilem. Přívod napájení k žárovkové sklenici je řešen pohyblivým páskovým kabelem.

Žárovkové světlo bude pro materiály určené pro kopírování a ztvrdování (papíry, fototechnické filmy, kinopozitivní filmy, pro barevné papíry a denní světlo pro snímací materiály – černobílé a barevné negativní filmy), modrý a zelený filtr (nebo modré a zelené světlo) pro expozici odpovídajících rentgenových filmů určených pro použití se zesilujícími fóliemi.

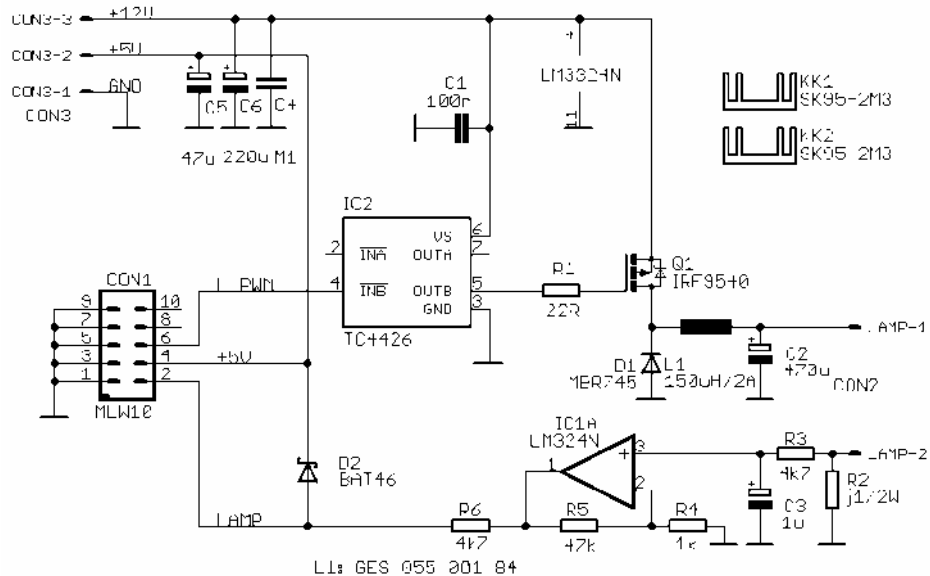
9.5 řízení proudu expoziční žárovky

Pro správnou expozici měřeného materiálu je nutná konstantní svítivost světelného zdroje. Tímto zdrojem zde je halogenová žárovka na 12 V s příkonem asi 10 W. Svítivost takového zdroje závisí na efektivní hodnotě proudu touto žárovkou

procházejícím. Tento proud je tedy nutné vhodně stabilizovat, tedy i měřit. Pro měření proudu je nejjednodušší použít odporový snímač zapojený proti zemi a jeho napětí vhodně zesílit pro další zpracování. Měření proudu žárovkou může být spojitě nebo impulzně. Oba způsoby mají své výhody i nevýhody.

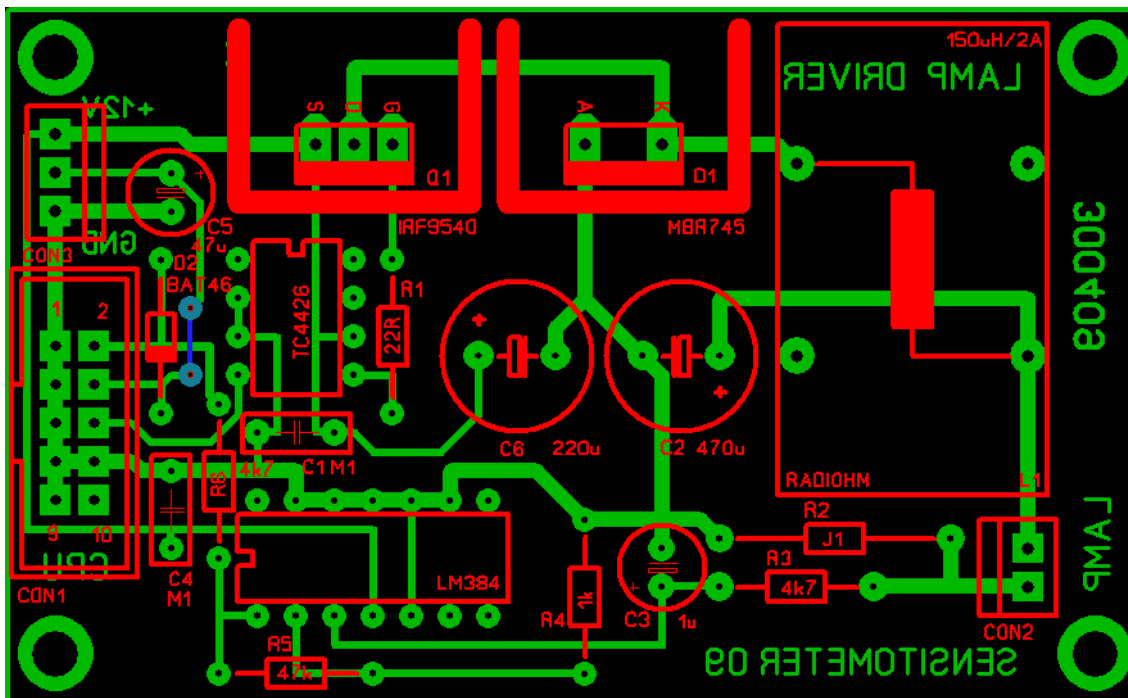
Spojitě měření proudu by mohlo být realizováno např. DA převodníkem, jehož výstupní napětí by po výkonovém zesílení mohlo být přivedeno na žárovku. V tomto případě je proud žárovkou stejnosměrný a jeho měření naprosto bez problémů. Nevýhoda je, že nepotřebná energie se mění v teplo a zahřívá za měření.

Impulzní měření je v případě procesoru PIC16F877A jednoduše řešitelné, nebo tento procesor obsahuje jednotku pro generování široce modulovaných pulzů (PWM). Tyto pulzy lze po úpravě spínat výkonový MOSFET tranzistor spínající proud do žárovky. Pro rozlišení 10 bitů může být frekvence PWM asi 20 kHz. Tato frekvence je více než dostatečná z hlediska tepelné setrvačnosti vlákna. Poněkud horší je, pokud chceme měřit efektivní hodnotu proudu žárovkou. Jednoduchým zesílením proudových impulzů a jejich jednoduchou filtrací dostaneme střední hodnotu proudu, nikoliv efektivní. Konstrukce zesilovače pro získání efektivní hodnoty je komplikovanější. Proto se zdá výhodnější proud žárovkou vyhladit. Pro vyhlazení proudu lze použít tlumivku, kondenzátor a diodu. V tomto případě se vlastně jedná o STEP-DOWN měření.



Obr. 9.7 Schéma měření proudu expoziční žárovky

V použitém zapojení je použit pro zesílení a úpravu signálu PWM z mikroprocesoru obvod TC4426. Tento obvod vstupní TTL signál upraví na úroveň 12 V pro spínání výkonového MOSFET tranzistoru IRF9540. Obvod TC4426 má na výstupu buď schopný dodat dostatečný proud pro rychlé nabití i vybití vstupní kapacity tranzistoru. Použitý tranzistor IRF9540 je typu PMOS, takže spíná napětí do zátěže. Filtrací obvod je tvořen rekuperační diodou, akumulací indukčnosti a kapacitou. Rekuperační dioda je typu Schottky pro její rychlost spínání a malé propustné napětí.



Obr. 9.8 Plošný spoj řízení proudu expoziční žárovky

Pro výpočet hodnoty indukčnosti lze použít, že se vlastně jedná o akumulaci v ní se sestupným napětím. Pro tento typ měření je výstupní napětí úměrné době sepnutí tranzistoru v rámci periody PWM. Pro minimální hodnotu indukčnosti tlumivky platí

$$L = \frac{U_2 \cdot T_b}{2 \cdot I_z}$$

kde U_2 je výstupní napětí, T_b je část periody, kdy je tranzistor rozepnut a I_z je proud zátěže.

V našem případě je zářivka, což lze považovat za rezistor. Výše uvedený výraz se tedy dá upravit na

$$L = \frac{R \cdot T_b}{2}$$

kde R je odpor žárovky. Pro náš konkrétní případ R = 14 Ω. Dobu T_b stanovíme z frekvence PWM – 20 kHz a budeme předpokládat vypnutí po dobu 25% periody. Potom T_b = 12,5 μs. Po dosazení dostaneme L = 87,5 μH.

Toto je minimální hodnota, proto použijeme tlumivku s indukčností 150 μH.

Výpočet hodnoty filtrační kapacity pro tento typ měniče bohužel není nikde řešen. Pro její odhad použijeme vztah pro výpočet napětí při vybíjení kondenzátoru přes rezistor.

$$u_c = U \cdot e^{\left(\frac{-t}{RC}\right)}$$

Z tohoto vztahu vyjádříme C a dostaneme

$$C = \frac{t}{R \cdot \ln\left(\frac{U}{u}\right)}$$

Po dosazení t = 12,5 μs a pokles napětí o 1 % dostaneme C = 89 μF. Pro jistotu použijeme větší hodnotu 470 μF. Kondenzátor by měl být typ s nízkým sériovým odporem.

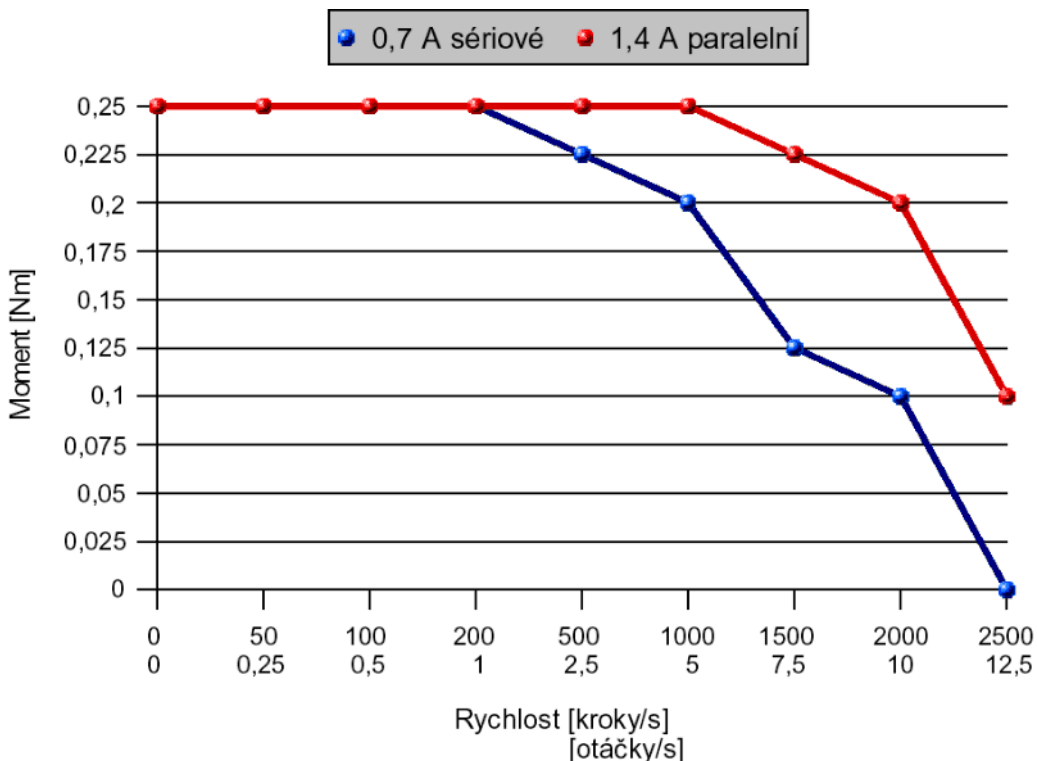
Mění proud vláknem žárovky je sejmutím na bočníku a zesílením na napájecí úrovni vhodnou pro měření mikroprocesorem. Jako snímací prvek je použit rezistor s odporem 0,1 Ω. Napětí na něm tedy může s ohledem na použitou žárovku dosáhnout asi 100 mV. Rezistor a kondenzátor na vstupu operačního zesilovače slouží pro filtraci vstupního signálu. Pro zesílení byl použit operační zesilovač LM324. Tento OZ může být napájen nesymetricky a je schopen pracovat od nulového vstupního napětí a na výstupu také může mít téměř nulové napětí. Napájen je z 12 V, nebo jeho výstupní napětí nemůže dosáhnout hodnoty kladného napájecího napětí. Zapojení OZ je jako neinvertující zesilovač se zesílením 48. Tím je pokryt vstupní rozsah A/D převodníku (5 V) použitého mikroprocesoru. Sériový rezistor na výstupu OZ s Schottkyho diodou na napájení 5 V slouží jako ochrana vstupu převodníku.

9.6 ízení pohybu expozi ního vozíku

Pro pohyb expozi ního vozíku byl zvolen krokový motor SX 16 -0503, který je dodávaný v p írubách NEMA17. S využitím moderní technologie se vyzna ují vysokými momenty 0,3 Nm p í zachování velmi malých rozm r . Standardní délka kroku je 1,8 ° s možností dalšího elektronického zmenšení.

Řada SX - p íruba NEMA17							
Typ	Statický moment (Nm)	Jmenovitý proud (A)	Indukčnost (mH)	Odpor Ω	Zbytkový moment (Nm)	Moment setrvažnosti rotoru (gcm ²)	Hmotnost (kg)
	bipol. napájení, celokrok, jmen. proud v obou fázích	sériové / paralelní zapojení	sériové / paralelní zapojení	sériové / paralelní zapojení			
SX16-0301	0,11	0,35 / -	30 / -	26 / -	0,005	11	0,12
SX16-0402	0,21	0,4 / -	32 / -	30 / -	0,012	20	0,18
SX16-0402NEW	0,21	0,4 / -	32 / -	30 / -	0,012	20	0,18
SX16-0502	0,23	0,5 / -	14 / -	14,4 / -	0,018	24	0,2
SX16-0503 / SX16-0503D	0,3	0,5 / 1	40 / 10	24 / 6	0,018	24	0,24

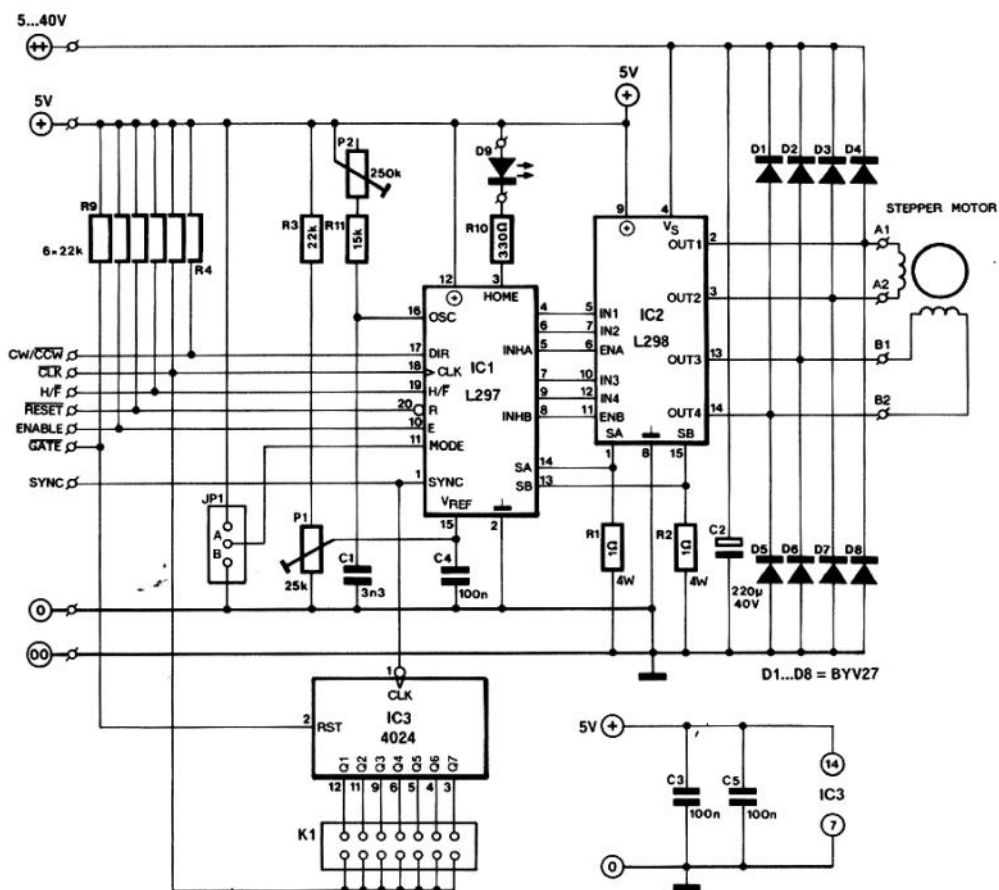
Obr. 9.9 Technické parametry krokových motor řady SX16



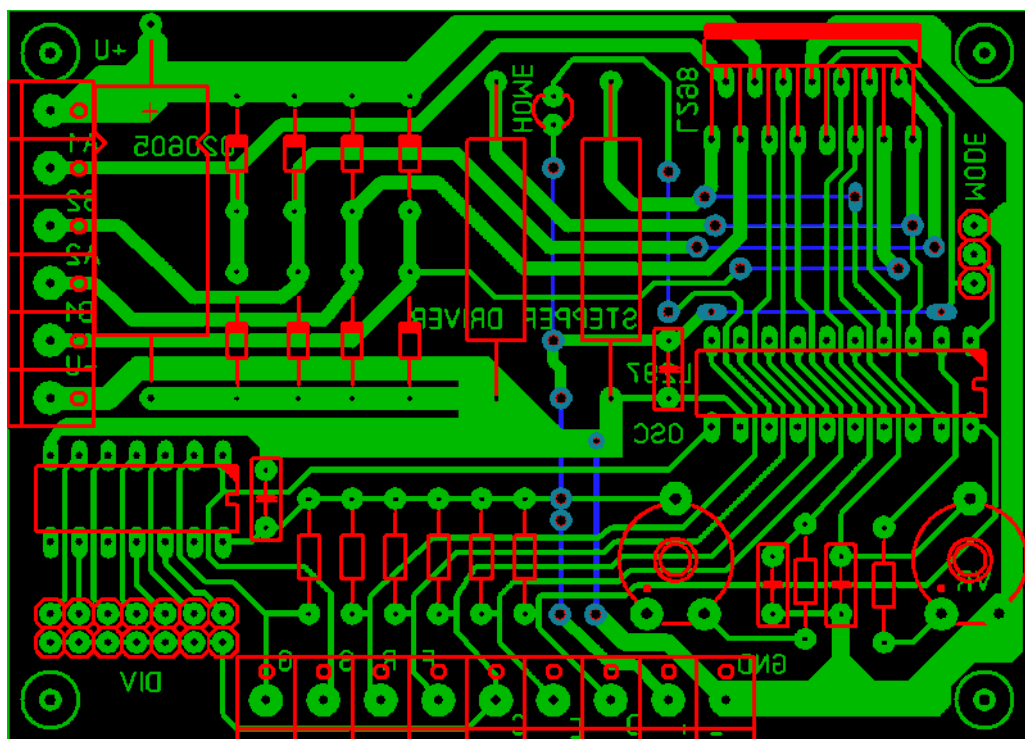
Obr. 9.10 Momentová křivka motoru SX 16-0503 p í nap. nap tí 35 V

Krokový motor bude ízen výkonovým driverem s obvody L298 a L297 v klasickém zapojení podle doporu ení výrobce. Obvod L297 (SGS Thomson) od procesor PIC16F877 (Microchip) p ebere informaci o sm ru otá ení, typu

krokování a povel k posunu o 1 krok a sám automaticky generuje ovládací signály pro dvoufázový bipolární motor. Integrovaný obvod L298 tvoří dvojitý H-m steek, který nabízí celkem 16 spínacích kombinací, z toho jich využijeme pouze 4. Abychom zjistili, že krokový motor SX 16-0503 je co do svých parametrů ta správná volba, byla navržena ovládací deska. Bylo použito zapojení publikované v německém časopisu Elektor 7/8 z roku 1988. Deska je navržena tak, aby se řízení otáček motoru a tím i samotné měření dalo provádět bez připojení na procesor. Deska je osazena kromě obvodů L298 a L297, také diodkou kmitočtu HCF 4024. Nastavení žádaného kmitočtu se provádí na konektoru K1 propojkami a pomocí trimru P2 sloužícímu k jemnému nastavení. Při zkouškách se mělo ověřit, zda je motor dostatečně výkonný v rozsahu otáček, ve kterých bude používán.

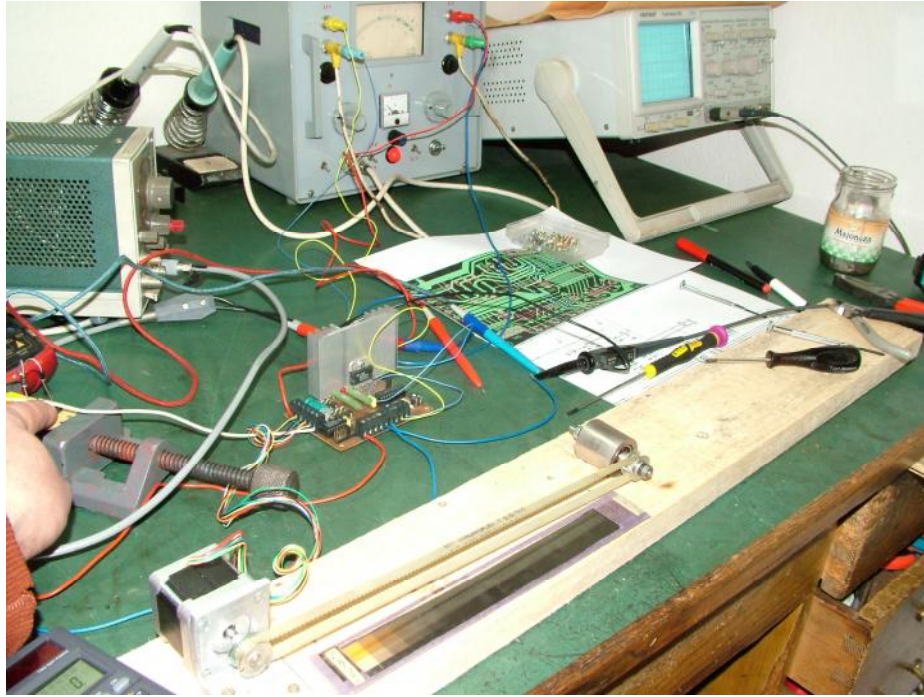


Obr. 9.11 Schéma ovládací desky krokového motoru



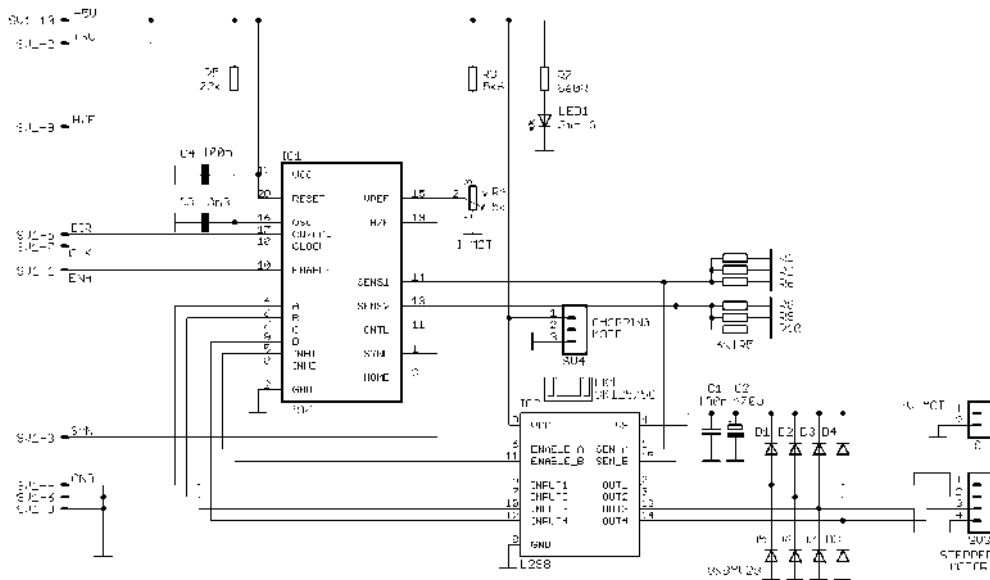
Obr. 9.12 Plošný spoj ovládací desky krokového motoru

Maximální kmitočet, kterým bylo možné motor ovládat bez nabíhové rampy, se kterou se ovšem počítá, je 1,8 kHz při napájecím napětí 35 V_{ss}. Při použití nabíhové rampy je motor možné ovládat s přihlédnutím ke klesajícímu momentu až do 5 kHz. Při této hodnotě klesne kroutící moment při paralelním zapojení o více než polovinu na cca 0,1 Nm. Jelikož je požadovaná expoziční doba minimálně 0,05 s je pro naexponování filmu potřeba, aby se lineární vedení umístilo z jednoho konce modulátoru na druhý minimálně za 1,05 s. Při jedné otáčce motoru se lineární vedení posune o 38,75 mm, což znamená, že na ujetí dráhy 200 mm je potřeba 5,16 ot. motoru. Dle technického listu motoru SX 16-0503, na 5,16 ot. potřebuje 1032 kroků. Zkouška ukázala že tato hodnota pro motor nepředstavuje žádný problém, jelikož od 1 000 kroku/s teprve začne klesat kroutící moment. V případě, že by byla požadována vyšší rychlost, bylo stálo za úvahu použít silnější motor s menší otáčkou v tímto průměru.

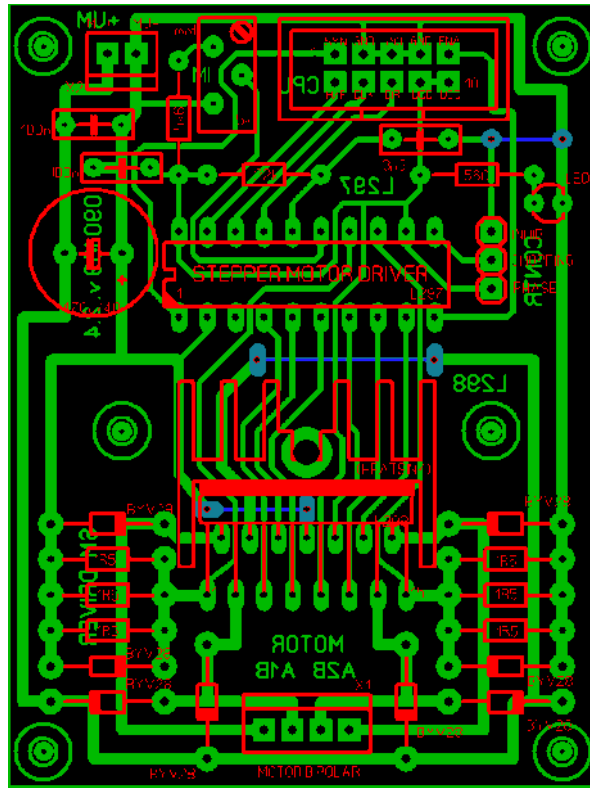


Obr. 9.13 Zkouška pohonu

Na Obr. 9.14 a Obr. 9.15 je definitivní schéma a plošný spoj na ovládání krokových motorů. Jsou dva, a to, jak již bylo uvedeno, jeden na pohyb celého líniového karuselu a druhý na volbu vhodného filtru. Jelikož se jedná o krokové motory, jejichž odbočky se sice liší, ale nikoliv významně (v ádech), bylo použito stejné provedení pro oba, a to z důvodu zjednodušení a zastupitelnosti.



Obr. 9.14 Schéma ovládání krokového motoru

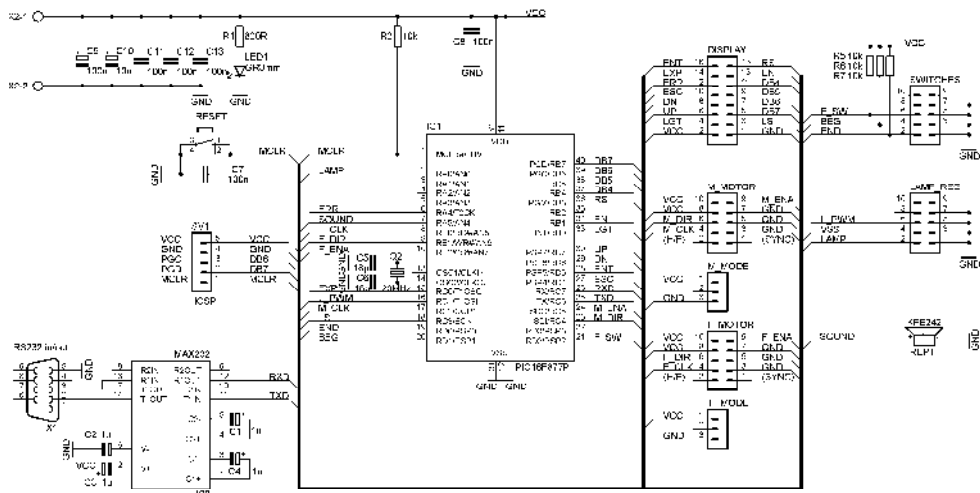


Obr. 9.15 Plošný spoj ovládání krokového motoru

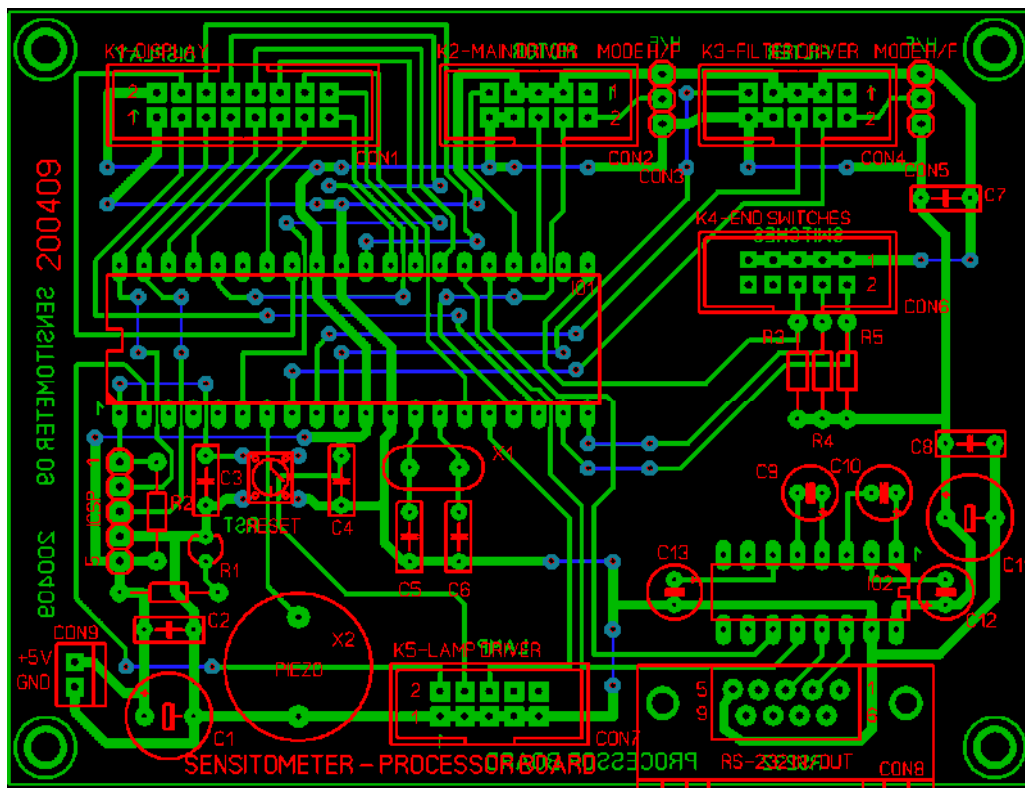
9.7 Procesorová deska

Pro řízení celého senzimetru je použit procesor PIC16F877 (Microchip), a to z následujících důvodů :

- Možnost programování v aplikaci prostřednictvím předem nainstalovaného programovacího driveru (bootloader), který umožní programování po stejném sériovém kanále, který je použit pro přenos dat. Tato koncepce umožňuje zjednodušení vývojových prací (není nutno používat externí programátor ani programovací přípravky pro ICSP programování) a umožňuje v budoucnu snadný upgrade použitého firmwaru.
- Hardwarově vyřešená sériová komunikace s řídícím počítačem. Jednoduše je možno využít obousměrnou komunikaci pro ovládání z PC.
- Dostatečný počet vstupních/výstupních portů, které jsou využity pro řízení transportního krokového motoru, pro připojení maticového displeje a řízení proudu světelného zdroje.



Obr. 9.16 Schéma procesorové desky



Obr. 9.17 Plošný spoj procesorové desky

Na místě řídicího procesoru celého senzimetru byl použit jednoipový mikroprocesor PIC16F877. Tento procesor je dosti výkonný, má dostatečnou paměť programu a obsahuje vhodné periférie. Paměť programu je typu FLASH a její délka 8 K slov je pro daný úkol dostatečná. Velkou výhodou příslušenství je možnost programování tohoto procesoru přímo v aplikaci po sériové lince RS232. Paměť RAM má délku 368 bytů. Datová paměť EEPROM 256 bytů zde je téměř zcela využita.

Pro taktování procesoru je použit krystal s frekvencí 20 MHz. Kondenzátory C5 a C6 mají hodnotu dle doporučení výrobce. Taktovací frekvence je nejvyšší pro daný procesor, takže umožňuje jeho nejvyšší výkon a i komunikační rychlost 9600 Baud je dobře nastavitelná. Tlačítko RESET na desce se použije při programování tohoto procesoru po sériové lince RS232. Konektor SV1 je pro možnost sériového programování procesoru v desce. Toto zde však bylo použito pouze pro nahrání bootloaderu (zaváděče) do procesoru a aplikací program již byl nahráván přes rozhraní RS232.

Komunikace s nadešeným PC je velice jednoduchá (i při nahrávání programu), takže žádné další bity nejsou potřeba. Na správné napávací úrovni standardu RS232 jsou logické signály upraveny obvodem MAX232. Tento obvod z jednoho napájecího napětí 5 V vytváří na principu nábojové pumpy kladné i záporné napětí pro úroveň RS232 a signály příslušenství zesiluje. Tento procesor již obsahuje jednotku USART, takže obsluha této linky je jednoduchá.

Další použitá vnitřní periférie je jednotka komparace. S její pomocí je možné přesně generovat výstupní frekvenci, což je použito pro buzení krokového motoru pojezdu expoziční žárovky. Z tohoto důvodu je tento výstupní signál (M_CLK) připojen na RC2. Ostatní řídicí signály pro tento motor M_DIR (směr rotace) a M_ENA (povolení buzení motoru) jsou připojeny na obecně použitelné výstupy procesoru. Řídicí výstupy a napájení logiky budíce krokového motoru je vyvedeno na konektor M_MOTOR. Pro tento motor je na desce propojka pro volbu celokroku nebo polokroku motoru.

Pro krokový motor posunu filtru jsou výstupní signály také vyvedeny na samostatný konektor F_MOTOR. Na procesoru jsou tyto signály připojeny na obecně použitelné výstupy. Důvodem je, že budicí frekvence pro tento motor je nízká, takže není problém ji generovat z celého programu.

Deska procesoru dále řídí svítivou žárovku. Pro řízení proudu je použita vnitřní jednotka pulsní kóde modulace. Tato jednotka může mít opakovací frekvenci asi 20 kHz a rozlišení 10 bitů. Signál je vyveden na portu RC1. Skutečný proud žárovkou je měřen na základě úbytku napětí na odporu. Tento signál je na desce řízení žárovky vhodně zesílen pro zpracování AD převodníkem, který je další použitou vnitřní periférií mikroprocesoru. Napájecí signál vstupuje do portu RA0. Referenční napětí pro převodník je odvozeno z napájecího napětí.

Pro styk s obsluhou je použit LCD displej s pěti tlačítky. Displej má 2 x 16 znaků. Komunikace s procesorem je po 4 datových bitech a 2 řídicích bitech. Řídicí bit pro možnost čtení informací z displeje byl pro nedostatek vývodů v konektoru vynechán, jeho funkce není nezbytná. S ohledem na nutnost pracovat ve tmě, je podsvícení displeje ovládané procesorem. Všechny signály jsou připojeny na obecně použitelné vstupy a výstupy procesoru.

Na desce je ještě umístěn akustický měnič pro signalizaci provozních stavů bez použití svítla.

Celá deska je napájena stabilizovaným napětím 5 V.

10. Programové vybavení

10.1 Popis programu senzitometru

Celý program je napsán v jazyce C a je použit p eklada BoostC. Program se skládá z p ti modul v jazyce C, jsou to senzit.c, serial.c, lcd.c, ppr_senzit.c, ser_senz.c. Všechny jsou uvedeny v projektu. Dále se používá n kolik hlavi kových soubor . Soubor senzit.h obsahuje definice vstupních a výstupních bit a použitých konstant. Soubor serial.h obsahuje konstanty pro sériovou linku a hlavi ky funkcí pro sériový p enos. Soubor inic_hw.h obsahuje obecn použitelné konstanty a makra pro snadn jší inicializaci HW použitého mikroprocesoru. Soubor lcd.h obsahuje konstanty pro LCD displej a hlavi ky funkcí pro jeho obsluhu.

Program obsluhuje navržený HW a eší úlohy spojené s provozem senzitometru. Pro provoz senzitometru je nutné ešit n kolik úloh. Po zvážení požadavk na sou asnost úloh jsem dosp l k záv ru, že bude jednodušší nepoužít p erušovací systém mikroprocesoru. Tímto se zjednoduší odlad ní programu.

Pro komunikaci s obsluhou je použit podsvícený LCD displej 2 x 16 znak . Jedná se o standardní modul s adí em Hitachi HD44780. Pro jeho ízení se používá datová sb rnice o ší ce 8 nebo 4 bity a 3 ídicí bity. Pro úsporu ovládacích bit byla použita 4 bitová datová sb rnice a 2 ídicí bity. Tímto se komunikace s displejem trochu zkomplikovala, data se posílají na dv ásti a místo na tení zp tného hlášení o ukon ení vnit ních operací displeje se eká po dobu uvedenou v technické dokumentaci.

Obsluha komunikuje s p ístrojem pomocí 5 tla ítek. Tato tla ítka jsou p ipojena na jednotlivé vstupy mikroprocesoru. Obslužný program se musí vypo ádat s mechanickými zákmity p i spínání t chto tla ítek.

S nad azeným PC se komunikuje po sériové lince RS232. Použitý procesor tuto komunikaci svým vnit ním modulem podporuje. Je tedy pouze nutné správn nastavit bity ídicích registr . Protože není použito p erušení, je nutné p i tení této linky dostate n rychle znaky odebírat. To je spln no, nebo po zjišt ní p íjmu znak se procesor v nuje pouze jejich tení. P íjímat zprávu z nad azeného PC je však možné pouze v klidu za ízení, tzn. vozík je v klidu a obsluha nema ká žádné tla ítko.

Za řízení obsahuje 2 krokové motory. Jejich proud je řízen externím modulem s běžnými obvody L297 a L298. Řídící signály jsou 3, povolení buzení motoru, směr pohybu a pohybové pulsy. Pro vyšší rychlosti pohybu může být obtížnější realizovat pohybové pulsy. Pro správný chod motoru je třeba co nejpřesněji dodržet jejich časování. Na pohon filtr nejsou kladeny vysoké nároky na rychlost. Proto je možné řídicí pulsy generovat zcela programově. Řídící signály jsou tedy připojeny na vhodné volné výstupy mikroprocesoru. Pokud jiná je situace pro motor pohonu vozíku. Zde potřebuje vyšší výstupní frekvence není vyloučena. Proto je signál pohybových pulsů připojen na RC2. Tento výstup může mít vnitřní připojenou jednotku komparace. Právě tato jednotka se dobře hodí na generování výstupní frekvence. Základem je 16 bitový časovač T1, který je nastaven na takt o délce 1,6 μs a volně běží. Další důležitého registr je CCPR1, do něhož je možné zapsat číslo, se kterým se vnitřní porovnává hodnota T1. Pokud dojde ke shodě, je na výstup RC2 zapsána předvolená hodnota 1 nebo 0. Vnitřně se nastaví flag o vzniku shody. Toto musí zjistit obslužný program, připočítat další dobu k CCPR1 a předvolit výstupní hodnotu po nové komparaci. Výhoda je, že obslužný program nemusí okamžitě reagovat, má čas podle délky generovaného pulsu a výstupní pulsy jsou přesto dlouhé. S provozem krokového motoru na vyšší frekvenci také souvisí, že je nutné motor pozvolna rozběhnout. Znamená to periodu pulsů pozvolna zkracovat, nejlépe tak, že zrychlení motoru je konstantní. Výpočet dob period není zcela jednoduchý. Je možné použít rekurentní vztah dle patentu ing. Eduarda Vereše

$$t_n = t_{n-1} - k \cdot t_{n-1}^3$$

kde t je doba periody a hodnota k je konstanta pro zrychlení. I výpočet podle tohoto vzorce by pro daný procesor byl dosti časově náročný, takže se jeví jednodušší, hodnoty rampy vypočítat při tvorbě programu, uložit je do paměti programu a při chodu programu je pouze vybírat. Pokud by tedy bylo třeba dosáhnout vyšší frekvenci, začne se na začátku tabulky a pro každý následující krok se vybere následující hodnota. Hodnoty se vybírají, dokud se nedosáhne cílová doba periody, se kterou se dále pokračuje. Při zastavení motoru by se mělo postupovat podobně, ale v této aplikaci není přesné udržení počtu kroků podstatné, a tak se zastaví bez zpomalení. Případná ztráta kroku se opraví při návratu na koncový spínač.

Poslední úloha je řízení proudu žárovkou. Tato úloha má dvě podúlohy.

Proud je řízen jednotkou šířkové modulace (PWM). Tato jednotka může být vnitřně připojena na port RC1. Jako zdroj časových značek je použit 8 bitový takt T2. Ten je konfigurován jako časovač, je možné použít jeho předělení. Protože se porovnává s 10 bitovou hodnotou, je jeho 8 bit interně rozšířeno o 2 bity předělení. Jeho taktování je možné zkrátit nastavením registru PR2, se kterým se aktuální hodnota T2 porovnává. Po dosažení shody se T2 nuluje a takt se odpočítá. Pro opakovací periodu PWM tedy platí:

$$PWM\ Period = [(PR2) + 1] \cdot 4 \cdot TOSC \cdot (TMR2\ Prescale\ Value),$$

kde TOSC je perioda řídicího oscilátoru procesoru. V našem případě PR2 = 255, TOSC = 1/20000000, takže dostaneme PWM Period = 51,2 μs, což odpovídá frekvenci 19531 Hz. Pro rozlišení v bitech potom platí:

$$rozlišení [bit] = \frac{\log\left(\frac{F_{OSC}}{F_{PWM}}\right)}{\log 2}$$

V našem případě je rozlišení 10 bitů. Vyšších 8 bitů se vkládá do registru CCP2L a 2 nižší jsou v CCP2CON<5:4>. Oběrstvovat lze kdykoliv, hodnota je použita až na začátku nového cyklu.

Chceme-li řídit, musíme měřit. K měření proudu použijeme vnitřní převodník A/D. Tento převodník může mít více vstupů. V našem případě jej nakonfiguruje na vstup RA0 a použítí napájecího napětí a zem jako reference. Dále je třeba nastavit vnitřní takt převodníku s ohledem na frekvenci oscilátoru. Z toho vychází doba převodu asi 20 μs. Ukončení převodu je signalizováno vnitřním bitem. Pokud by bylo nutné přepínat kanály, je nutné po přepnutí počkat asi také 20 μs, což ale nepotřebujeme. Měření a řízení probíhá i při chodu vozíku, což není časový problém.

Žárovka se zapíná před rozjezdem vozíku, kdy její svět neexponuje materiál. První hodnota PWM se vypočítá podle pevně zjištěných úměrů mezi žádanou hodnotou proudu a hodnotou PWM. Po nastavení této hodnoty se testuje proud, zda se mění. Pokud je změna malá, předpokládá se, že již žárovka svítí a přistoupí se k vlastní regulaci. Ta spočívá ve snížení i zvýšení hodnoty PWM podle znaménka odchylky. Takto se periodicky postupuje až do dosažení tolerovatelné odchylky. Při chodu motoru se reguluje podobně. Právě použití jednotky komparace umožňuje bez problémů sdílet tyto dvě činnosti současně.

10.1.1 Soubor senzit.c

Tento soubor obsahuje hlavní program. Začíná se definicí vložených souborů. Systémové soubory jsou v lomených závorkách. *Pragma DATA* umožňuje zápis dat do paměti programu. Hodnota *DATA_ROZBEH* je adresa, odkud jsou uloženy doby periody pro rozběh, viz výše. Čísla znamenají doby period v násobcích 1,6 μ s. Pro nastavení základních konfiguračních bitů použitého procesoru je také *pragma DATA*, ale ukládá se na speciální adresu dostupnou pouze při programování. Další *pragma* se používá pro určení hodinové frekvence pro překladač, podle ní vypočítává napájecí zdrojovací smyčky. Ještě před hlavním programem je definice globálních proměnných. Tyto proměnné jsou dostupné ze všech modulů a podprogramů, jsou stále v paměti.

Hlavní program v jazyce C musí vždy být a jmenovat se *main*. Po restartu a vykonání globálních inicializací jej procesor začne vykonávat. Hlavní program začíná konfigurací směrů jednotlivých bitů portů A až E. Hodnota odpovídajícího bitu 1 znamená vstup. Pro výstupy je nutné nastavit počáteční hodnoty. Vnitřní periférie je také nutné inicializovat vhodným nastavením registrů. Jedná se o A/D převodník pro převod kanálu RA0, čítač T0 na přetečení po 13,1 ms, T1 na tik 1,6 μ s a přetečení po 105 ms, čítač T2 pro generátor pulsů pro PWM s opakovací frekvencí 19531 Hz a rozlišením 10 bitů. Poslední vnitřní inicializovanou periférií je sériová linka, inicializuje se jako USART, přenosová rychlost 9,6 kB, délka 8 bitů, 1 stopbit.

Tento procesor dovoluje využít mnoho druhů přerušení od různých periférií. V tomto programu však není žádné přerušení použito nebo to není nutné. Zjednoduší se tím odladění programu. Zakázat je možné i jednotlivě, ale jsou zakázána globálně.

Následuje inicializace displeje a výpis úvodního hlášení. K tomu se volají příslušné funkce.

Dosud uvedené příkazy se vykonají pouze jednou po startu mikroprocesoru. Následuje smyčka, ve které je volána funkce návratu vozíku do výchozí polohy. Akce je uvozena hlášením na displeji. Výchozí poloha je testována koncovým spínáním. Pokud jí je dosaženo, je to uvedeno na displeji. Pokud ne, je i toto oznámeno a po stisku ENTER se akce opakuje. Pokud se před ENTER stiskne a drží ESC, inicializace je ukončena i se špatným výsledkem. Po vozíku se inicializuje poloha filtru a senzitometr je připraven k činnosti. Na displeji se objeví první expoziční profil.

Přiblížení innosti program v reálném čase běhá v nekonečné smyčce. Tato smyčka obsahuje několik testů, které povolují další innosti.

Prvně se testuje, zda se má motor vozíku pohybovat vpřed. Pokud ano, testuje se jednotka komparace, zda je třeba upravit nová doba, případně se upravit. Tato funkce také připomíná kroky ke globální proměnné. Testuje se ujetá dráha a pokud je její konec, zhasne se žárovka a jednotka komparace upravit na chod zpět. Změní se řídící proměnná z chodu vpřed na vzad. Nekončí-li dráha, testuje se čas pro regulaci proudu žárovky, případně se regulace vykoná. Čas je odvozen z přetečení asova T_0 .

Pokud se motor vozíku má pohybovat vzad, opět se testuje a řídí jednotka komparace. Po dosažení požadovaného kroku zpět se motor vypne a na displeji se zobrazí hlášení o ukončení expozice. Stisk tlačítka ENTER toto hlášení přemazá zobrazením parametrů povoleného profilu. Řídící proměnné chodu vpřed i vzad se nulují. Pokud je pohyb motoru vpřed i vzad, následně uvedené testy se nemohou vykonat, nebo jsou přeskočeny.

Není-li tedy pohyb motoru, testuje se přímým sériovým lince. Pokud byl přijat znak, volá se funkce na zpracování přijímaného dat zce.

Testuje se hodnota proměnné *start*. Je-li nastavena, zahájí se úkony k vykonání další expozice. Testuje se výchozí poloha vozíku, pokud není, je posunut. Filtr se také přesune do výchozí polohy. Žárovka se rozsvítí na zadaný proud. Jeho nastavení se testuje. Filtr se přesune do pracovní polohy dle povoleného profilu. Upravit se povolená hodnota expozice a nastaví jednotka komparace. Nastaví se řídící proměnná chodu vpřed.

Poslední test se vykoná po přetečení T_0 . Funkce *tlacitka()* vyhodnocuje stisk všech 5 tlačítek a ukládá je do globální struktury. Stisk jednotlivých tlačítek má různý význam.

Stisk tlačítka DOWN zobrazí na displeji následující expoziční profil. Expozičních profilů je 10 uložených v paměti EEPROM. Zobrazí se vždy jeho číslo a po 1 s jeho hodnoty, tzn. název, doba expozice, barevná teplota a číslo použitého filtru. Po posledním záznamu se zobrazí zase první.

Stisk tlačítka UP zobrazí na displeji předcházející expoziční profil. Po prvním záznamu se zobrazí zase poslední.

Stisk tlačítka ENTER zobrazí na displeji číslo profilu a na dalším řádku hodnotu periody pro motor pohonu vozíku. Tato hodnota je uvozena písmenem T a je v násobcích 12,8 μ s, takže tak, jak je uvedena v PC . Další hodnota je uvozena I a jedná se o žádanou hodnotu proudu žárovky v jednotkách p evodníku.

Stisk tlačítka ESC vrátí zobrazení zpět po ENTER.

Stisk tlačítka START nastaví proměnnou *start*.

Tímto hlavní program končí.

10.1.2 Soubor serial.c

Serial.c obsahuje pouze základní funkce pro obsluhu sériové linky. Je převzat z internetu a jeho autoři jsou uvedeni v jeho záhlaví. K tomuto souboru se váže soubor serial.h. Je zde uvedeno poměrně mnoho funkcí. Z nich je použito minimum. To však nevadí, nebo nepoužité funkce nejsou do programu linkovány.

Doplněna byla pouze funkce serial_poll(). Tato funkce testuje chybu příjmu, v tom případě restartuje příjima a vrátí odpovídající číslo CH_PRIJ. Dále testuje přijatý znak, pokud je, vrátí jej, jinak vrátí odpovídající číslo BEZ_ZN. Tato čísla jsou dodefinována v serial.h. Tato funkce je potřeba, neboť na znak nečeká a program může vykonávat jinou činnost.

10.1.3 Soubor ser_senz.c

V tomto souboru jsou funkce pro zpracování akcí souvisejících s příjmem po sériové lince.

Funkce *cti_zpravu_ser()* se používá pro příjem po sériové lince. Po vyvolání testuje příjem znaku, pokud je přijatý, testuje se, zda je inicializace pro telegram. Pokud ano, čeká se na příjem dalších znaků a zpráva se uloží do globální proměnné pro další zpracování. Pokud znak nebyl přijat nebo není inicializace je vráceno příslušné hlášení.

Pokud se jedná o telegram, podle úvodního znaku se rozpozná o jaký se jedná. Pokud se jedná o expoziční profil, je zpráva dekódována a uložena do EEPROM. Pro ukládání do EEPROM se číselné údaje přivedou pro úsporu místa z čísla na číslo.

10.1.4 Soubor lcd.c

Tento soubor obsahuje základní funkce pro obsluhu displeje. Je převzat z internetu a jeho autor je uveden v jeho záhlaví. K tomuto souboru se váže soubor lcd.h, kde je i konkrétní pí azení výstup procesoru jednotlivým řídicím signál m. Základní funkce jsem rozšířil o další užitečné funkce, které umožní zápis na požadované místo na displeji.

10.1.5 Soubor ppr_senz.c

Toto je poslední popisovaný soubor. Obsahuje různé funkce použité v hlavním programu.

První jsou funkce pro obsluhu tlačítek. Funkce se snaží eliminovat záškrtí tlačítek při spínání. Docílí se toho testováním stisku po dobu několika desítek ms, stejně tak rozepnutí. Funkce je schopna rozpoznat i dlouhý stisk. Výsledek se zaznamená do globální proměnné. Pro správnou činnost je nutné tuto funkci periodicky volat, což činí hlavní program s periodou přibližně T0.

Dále jsou funkce pro přípravu a obsluhu komparací jednotky pro generování pulsů pro krokový motor vozíku. Tyto činnosti již byly vysvětleny výše.

Velmi krátké jsou funkce pro zápis hodnoty do jednotky PWM. Jedná se vlastně pouze o přepis do správných registrů. Funkce jsou definovány jako inline, což znamená, že se do kódu vloží přímo.

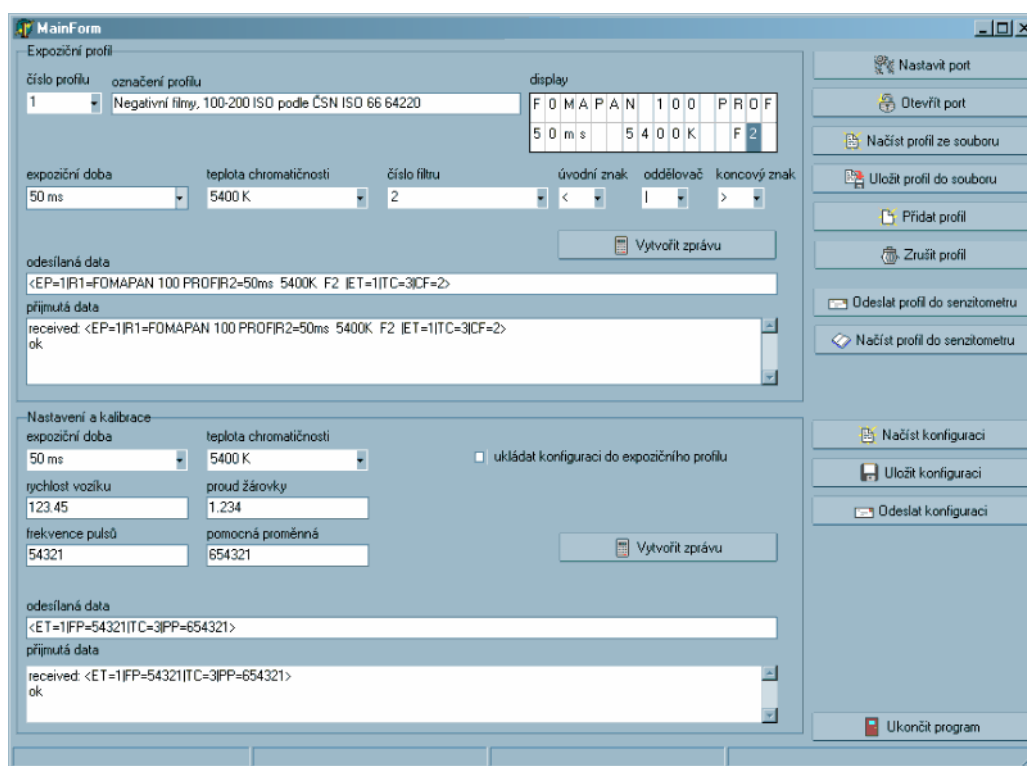
Funkce *zobr_zaznam(byte index)* na displej zobrazí aktuální expoziční profil. Hodnoty vybere z EEPROM, na které převede z číselné reprezentace na metrac a zobrazí na displej. Jako vedlejší činnost také naplní pracovními hodnotami globální proměnnou, podle které se potom exponuje a řídí proud žárovky.

Další dvě funkce řídí pohyb vozíku a filtru. Společně mají, že se pro pohyb používá generování pulsů v programu. Jedná se o pomalejší pohyb, takže to je bez problémů.

Posledních několik funkcí se zabývá měřením A/D převodníku a regulací proudu žárovkou. Tato regulace již byla vysvětlena výše.

10.2 Koncepce ovládacího programu pro senzitometr

Program pro komunikaci se senzitometrem byl vytvořen v jazyce Delphi 6 professional s využitím několika volnějších externích komponent. Popisovaná verze programu slouží především pro odlaďování komunikace; předpokládá se, že konečná verze bude zjednodušená na základní nutné procesy, podstatné z hlediska obsluhy.



Obr. 10.1 Ovládací program pro senzitometr

Ovládací program je vlastně jednou zložitým komunikačním terminálem, který zajišťuje obousměrnou komunikaci mezi řídícím počítačem a senzitometrem po sériovém kanálu RS232.

Program zajišťuje dvě základní úlohy:

- Nastavení expozičních profilů pro jednotlivé materiály, které mají být exponovány a výběr, mezi kterými má být dostupná obsluha.
- Kalibraci expoziční doby změnou rychlosti posuvu expozičního vozíku a svítivosti expoziční žárovky.

Tomu odpovídají 2 oblasti hlavního okna. Přístup do kalibračního režimu je chráněn heslem. Předpokládá se, že nastavení provádí metrolog.

U expozičních profilů se volí číslo profilu (1 z max. 10). Zde je k dispozici popis (jen pro informaci), simulovaný display, kde je možné napsat příslušné zobrazované texty a hned je viditelná kombinace, ze kterých se volí předdefinované jmenovité exp. parametry (exp. doba, teplota chromatičnosti a číslo filtru). Dále se volí koncový a úvodní znak a případná oddělování.

Do senzimetru se přenáší tedy text pro zobrazení na displej a požadovaná čísla parametrů z předem definovaných seznamů.

U nastavení a kalibrace se zobrazí vlastní dvě tabulky, pro expoziční dobu a teplotu chromatičnosti; u filtru není co kalibrovat, přenáší se přímo jeho požadované číslo. Pro každou předvolenou expoziční dobu (např. 50 ms) se zobrazí rychlost posunu (vypočtená v PC) a frekvence pulsu pro krokový motor. Tato se pak přenesou do senzimetru.

Podobně u teploty chromatičnosti se zobrazí proud žárovky (jen pro informaci, hodnota se získá při kalibraci žárovky) a pomocná hodnota, která se v PIC-u uloží a se kterou se pak porovnává úbytek na seriovém odporu.

Tedy při odeslání do senzimetru se přenášejí hodnoty těchto dvou proměnných, vždy pro každou expoziční dobu z předvolených, případně pro každou teplotu chromatičnosti z předvolených. Tyto hodnoty a proměnné z expozičních profilů se interně ukládají v E2PROM a při startu expozice se na to, zvolí se příslušný profil a proměnné se využijí pro řízení rychlosti a svítivosti.

Po kliknutí na tlačítko „Vytvořit zprávu“ se sestaví zpráva (telegram) o dané struktuře s vloženými znaky, odešle do senzimetru a zpět na to. Odeslaný a přijmutý string se porovnají a při shodě se vypíše „ok“, jinak chybové hlášení.

Program se ovládá tlačítky na pravé straně okna (otevírání a ukládání souborů s profily a kalibračními údaji, přidávání a mazání profilů, nastavení a otevření/zavření portu atd.).

Ve stavovém řádku se zobrazují parametry přenosu (např. COM1:9600,8,n,1), chybová hlášení, event. další informace.

ZÁV R

Práce předkládá návrh koncepce univerzálního senzimetru vhodného pro použití ve fotografickém průmyslu při výrobě a kontrole zpracování. Z předložené řešení, kde jsou popsána existující komerčně dostupná řešení senzimetru vyplývá, že navrhované řešení je zcela nové a pokud je známo, popsaná koncepce přístroje není známa.

Podrobněji byl diskutován základní uzel nové koncepce senzimetru, kterým je šťrbínová expozice s pohyblivým světelným zdrojem.

Hlavní přednosti navrhovaného řešení jsou následující:

- a) Úspora dávkování se šťrbínovou závěrkou, tj. s osvětlenou šťrbínou, pohyblivou konstantní rychlostí v těsné blízkosti absorpčního modulátoru a exponovaného filmu umožňuje použít relativně velkého modulátoru, jehož rozměrná pole jsou velmi vhodná pro pohodlné měření senzigramu. Tento princip zajišťuje, na rozdíl od senzimetru s bodovým světelným zdrojem, rovnoměrné osvětlení exponovaných polí při malé vzdálenosti světelného zdroje od exponovaného materiálu.
- b) Použití šťrbínové závěrky je všem známo a bylo dovedeno k nejméně dvěma komerčně dostupným řešením (senzimetry Kodak Model 6 a Model 7) a rovněž prototypovým senzimetrem VÚZORT MKS-1 a KS-1. Jmenované přístroje Kodak byly, alespoň teoreticky, dostupné v 80. letech, ale prakticky tomu tak nebylo, nebo politikou firmy Kodak nebylo poskytovat přístroje tvořící součást výrobní technologie konkurenčnímu československému fotografickému průmyslu. To bylo také důvodem vývoje vlastního řešení VÚZORT. Rozpadem přístrojové sekce VÚZORT v polovině 90. let skončil vývoj senzimetru v ČR.
- c) Princip pohyblivé šťrbíny a nepohyblivého exponovaného materiálu je vlastně použit u klasických zrcadlovek, ale dosud nikde není popsáno jeho použití v senzimetrii.
- d) Malá vzdálenost světelného zdroje umožňuje z principu dosáhnout vhodného osvětlení pro expozici s využitím podstatně méně výkonné osvětlovací žárovky, přičemž vhodných expozičních podmínek je dosaženo v širokém rozsahu citlivostí exponovaného fotografického materiálu, tedy jak pro vysoce citlivé negativní filmy, tak i pro nízkocitlivé kopírovací černobílé papíry. Přizpůsobení

expozice citlivosti fotomateriálu je dosahováno v širokém rozsahu regulací rychlosti expoziční štrby. Dalšího rozšíření do oblasti vysokých citlivostí je možno dosáhnout zaazením zeslabovacího neutrální šedého filtru. Posuvná kulisa s filtry umožňuje i aplikaci konverzních a pásmových barevných filtrů.

- e) Použití žárovky s nižším výkonem přináší významnou energetickou úsporu, jednodušší chlazení regulačních prvků i samotné svítlové sklenice. Základním fotometrickým měřením bylo zjištěno, že požadovaného osvětlení 480 lx v rovině materiálu je možno dosáhnout halogenovou žárovkou o výkonu 10 W a méně, tedy 15x menším než u běžných senzimetrických zdrojů. Žárovky jsou podžhávány a pracují s nižším výkonem.

V dalším byla navržena konstrukce pohyblivé části závěrky, která je tvořena lineárním vedením s profilovou vodicí kolejničkou, kluzným pohyblivým vozíkem, který nese svítlovou lampovou sklenici, která nese na straně převrácené k modulátoru expozice posuvnou kulisu s filtry a expoziční štrbinou. Expozici je možno tedy ovlivňovat:

- lineární posuvnou rychlostí expoziční štrby umožněním řízení pohonného motoru
- šířkou expoziční štrby (pevnou, případně vyměnitelnou)
- optickou hustotou zaazeného zeslabovacího filtru
- svítivostí osvětlovací žárovky

Svítivost žárovky se předpokládá konstantní, a to zejména pro souvislost svítivosti a teploty chromatičnosti a napájecího proudu (zvýšením teploty vlákna žárovky zvýšením napájecího proudu se sice zvýší svítivost, současně se ale změni vyzařované spektrum – posune se spektrální maximum k modré části spektra). Proto se uvažuje o žhavení konstantním nastavitelným proudem tak, aby bylo dosaženo normovaných hodnot senzimetrického „žárovkového“ (2 850 K) a „kopírovacího“ (3 200 K) svítiva. Tetí normované složení svítiva „denní“ (5 400 K) je nutno dosáhnout obvyklým slabomodrým konverzním filtrem.

Jaké jsou nevýhody navrhovaného řešení?

Pohyblivý svítivý zdroj je nutno napájet; rovněž je nutno napájet pomocný motor, který přestavuje kulisu s filtry a vyvést do řídicí jednotky signály ideálně indikující správnou polohu filtru, nebo celá osvětlovací jednotka je pohyblivá.

Tento problém je běžně řešen u jehli kovových a inkoustových tiskáren s pohyblivou tiskovou hlavou, stejně jako u CNC strojů. Stejně řešení, tj. tažený plochý kabel, bude použit u senzitometru.

Použití krokového motoru pro pohon je vhodné pro snadnou regulaci v širokém rozsahu rychlostí potřebných pro expozici bez nutnosti snímání skutečné posuvné rychlosti, ale má rovněž dvě nevýhody:

- Pokles krouticího momentu u vysokých rychlostí je omezující u krátkých expozicích a potřebných pro expozici vysoce citlivých materiálů; snížení expozice je pak vhodné docílit zaazením zeslabovacího filtru.
- Aby krokování posouvané expozicní štrby nebylo na exponovaném materiálu pozorovatelné jako pípná pruhovitost, je nutné aby šířka expozicní štrby byla mnohonásobně větší než je velikost kroku posunu do roviny posunu.

Oba problémy omezují použití senzitometru u požadovaných krátkých expozic. Proto je nutné zvolit krokový motor s poměrně velkou rezervou výkonu tak, aby zajistil spolehlivý posun expozicní štrby vysokými rychlostmi. U dlouhých expozic tento problém není, omezujícím faktorem je spíše pohodlná expozice, nebo jednotlivá pole senzitogramu se vlastně exponují postupně, což přináší podstatné prodloužení doby potřebné k naexponování celého senzitogramu.

Závěrem je třeba říci, že popisovaný zkušební vzorek senzitometru slouží zejména k ověření funkčnosti jednotlivých funkčních uzlů. Podle výsledků praktických zkoušek může být definitivní verze přístroje v některých částech odlišná.

Byly ale vytvořeny předpoklady pro výrobu přístroje, jehož absence je ve fotoprůmyslu výrazně pocítována a jehož dovedení k realizaci by mohlo být komerčně úspěšné i v současné době charakterizované poklesem produkce klasických fotografických materiálů.

SEZNAM POUŽITÉ LITERATURY

- [1] Ing. Záhlava, V., CSc., Metodika návrhu plošných spojů, Praha, Vydavatelství VUT, 2000, 81s, ISBN 80-01-02193-9
- [2] Kovář, D., Kovářová, I., Kačuch, J., EMC – z hlediska teorie a aplikace, Nakladatelství BEN – technická literatura, 1. vydání, Praha, 2006, 216s, ISBN 80-7300-202-7
- [3] Ing. Abel, M., Plošné spoje se SMD, návrh a konstrukce, Nakladatelství Platan, 1. vydání, Pardubice, 2000, 203s, ISBN 80-902733-2-7
- [4] Ing. Šandera J., Ph.D., Návrh plošných spojů pro povrchovou montáž, Nakladatelství BEN – technická literatura, 1. vydání, Praha, 2006, 269s, ISBN 80-7300-181-0
- [5] Urban, M, Ing. Sklenář, Z., Základy senzimetrie pro I. - III. Ročník SOU, Praha, Státní pedagogické nakladatelství, 1987, 251s
- [6] SN ISO 5799 (66 6624) Fotografie - Přímě exponovaný lékařský a dentální radiografický film - Stanovení citlivosti ISO a průměrného gradientu ISO
- [7] SN ISO 6 (66 6625) Fotografie - Systémy negativního filmu a jeho zpracování pro černobílou obrazovou fotografii - Stanovení citlivosti podle ISO
- [8] SN ISO 5-1 (66 6601), Fotografie – Měření hustot – část 1: pojmy, značky a vysvětlivky
- [9] SN ISO 5-2 (66 6601), Fotografie – Měření hustot – část 2: Geometrické podmínky pro měření prostupové hustoty
- [10] SN ISO 5-2 (66 6601), Fotografie – Měření hustot – část 3: Spektrální podmínky
- [11] Kolektiv, Principy a praxe radiální ochrany, Praha, Státní úřad pro jadernou bezpečnost, 2000, 619s, ISBN 80-238-3703-6

SEZNAM WWW STRÁNEK (20.5.2009)

- [1] http://www.foma.cz/foma/produkt/RadioDetail.asp?ProduktID=149&seznam=vybav_RDG
- [2] <http://www.microcon.cz/>
- [3] http://ouwehand.net/~peter/lcd/lcd1.shtml#_8051_lcd_sch
- [4] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2047
- [5] http://www.microchip.com/stellent/groups/designcenter_sg/documents/device_doc/en027001.pdf
- [6] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2121
- [7] <http://lika.be/wp/2005/08/boostc-compiler-libraries/>
- [8] <http://www.teatechnik.cz/>
- [9] <http://www.teatechnik.cz/katalog/ozubeni-prevody/ozubene-hrebeny/sikmym-ozubenim-m-0-3-0-5/>
- [10] http://www.vmk-rtg.cz/pdf/katalog/kl_18311.pdf
- [11] <http://ouwehand.net/~peter/lcd/lcd0.shtml#hd44780>
- [12] http://www.microchip.com/stellent/idcplg?IdcService=SS_GET_PAGE&nodeId=2590¶m=en529494
- [13] http://hw.cz/docs/lcd_ig_displaje/lcd_ig_dip.html
- [14] http://www.regulacni-pohony.cz/frm_krok.html
- [15] <http://www.regulace.cz/CZ/KM/index.html>
- [16] <http://www.ges.cz/>
- [17] <http://www.gme.cz/cz/index.php>
- [18] <http://prochazka.profitux.cz/ctyrbytova-komunikace-lcd-display-s-radice-m-hd4>
- [19] <http://www.acdc-audio.cz/minimotoryp.html>
- [20] <http://www.quasarelectronics.com/404.shtml>
- [21] <http://www.transmotec.com/dc-motors/?gclid=CKrdmqKNzpoCFQ0FZgod-xhI2Q>

SEZNAM TABULEK

Tabulka 1 – Katalogové údaje ozubené emenice	34
Tabulka 2 – Rozměry emenice	36

SEZNAM OBRÁZK

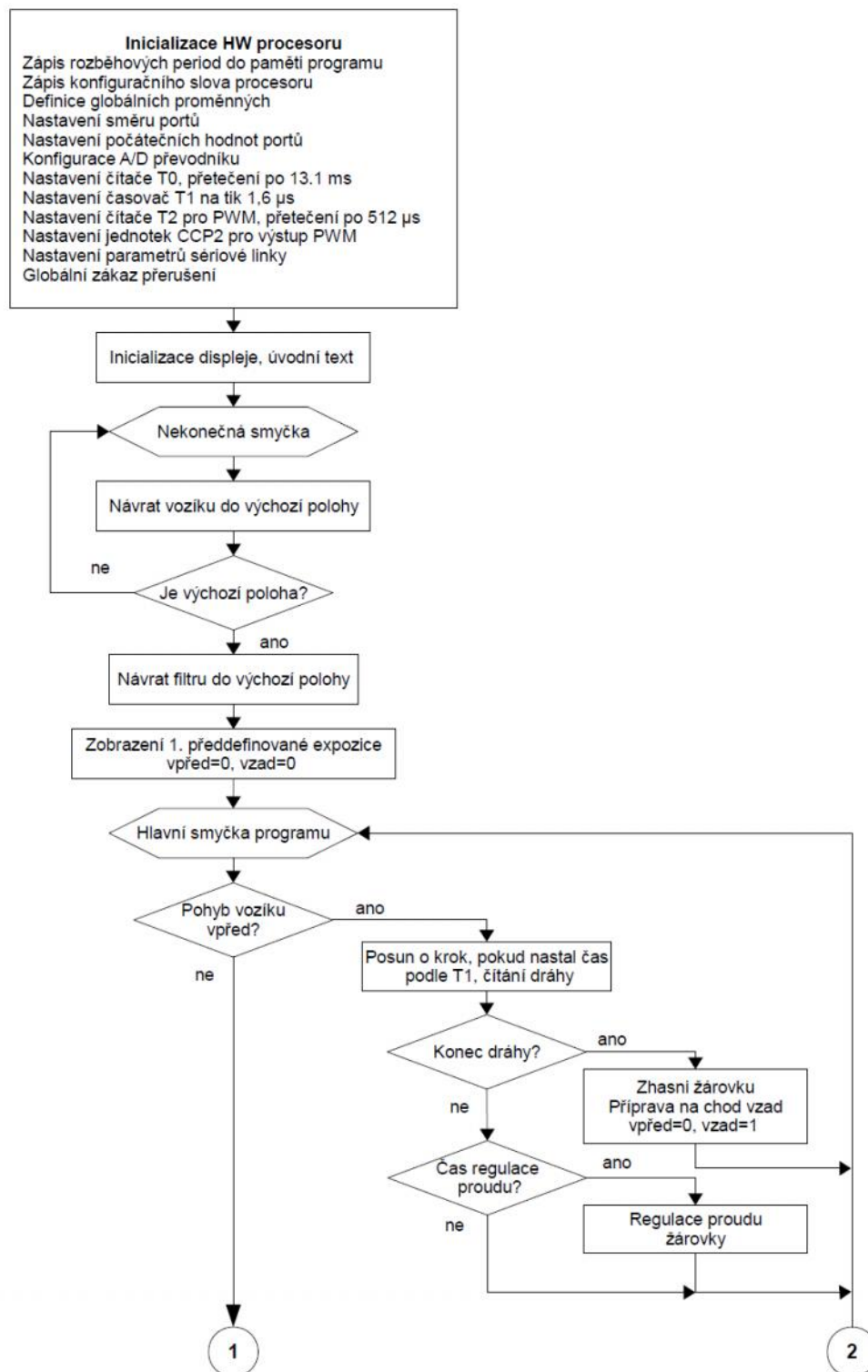
Obr. 4.1 Rentgenový film (se dvěma emulzními vrstvami)	18
Obr. 4.2 Princip rentgenování (film s jednou emulzí)	19
Obr. 4.3 Senzitometrická charakteristika	20
Obr. 4.4 Optická hustota	21
Obr. 5.1 Princip senzimetru	23
Obr. 7.1 Blokové schéma	30
Obr. 8.1 Lineární vedení – pohled shora	32
Obr. 8.2 Úhel lineárním vedením	33
Obr. 8.3 Polyuretanový ozubený čep a jeho rozměry	33
Obr. 8.4 Ozubená čepnice T2,5	34
Obr. 8.5 Držák filtr	35
Obr. 8.6 Čepnice pohonu rámečku filtr	36
Obr. 8.7 Reálná podoba	36
Obr. 8.8 Konstrukce lineárního karuselu – horní pohled	37
Obr. 8.9 Lineární karusel v úhlu	37
Obr. 8.10 Modulátor AGFA	39
Obr. 8.11 Horní pohled sestavy držáku modulátoru a filmu	40
Obr. 8.12 Boční úhel držáku	40
Obr. 9.1 Schéma napájecího zdroje	41
Obr. 9.2 plošný spoj napájecího zdroje	42
Obr. 9.3 Ovládací deska s LCD displejem	43
Obr. 9.4 Plošný spoj ovládací desky s LCD displejem	44
Obr. 9.5 Umístění žárovky	45
Obr. 9.6 Halogenová žárovka 10 W	46
Obr. 9.7 Schéma řízení proudu expoziční žárovky	47
Obr. 9.8 Plošný spoj řízení proudu expoziční žárovky	48
Obr. 9.9 Technické parametry krokových motorů SX16	50
Obr. 9.10 Momentová charakteristika motoru SX 16-0503 při nap. napětí 35 V	50
Obr. 9.11 Schéma ovládací desky krokového motoru	51
Obr. 9.12 Plošný spoj ovládací desky krokového motoru	52
Obr. 9.13 Zkouška pohonu	53
Obr. 9.14 Schéma ovládací desky krokového motoru	53
Obr. 9.15 Plošný spoj ovládací desky krokového motoru	54
Obr. 9.16 Schéma procesorové desky	55
Obr. 9.17 Plošný spoj procesorové desky	55
Obr. 10.1 Ovládací program pro senzimetru	65

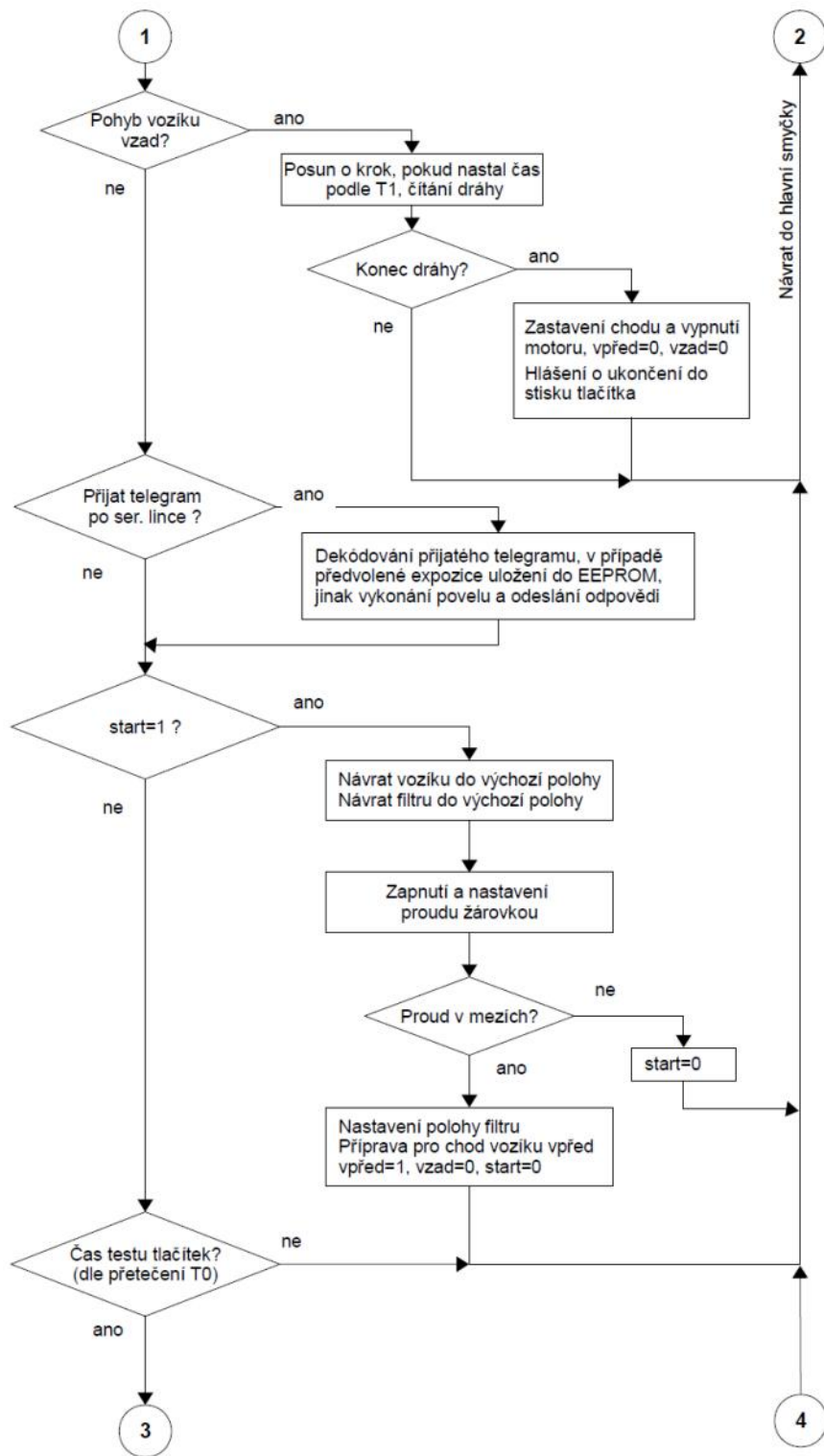
SEZNAM ZKRATEK

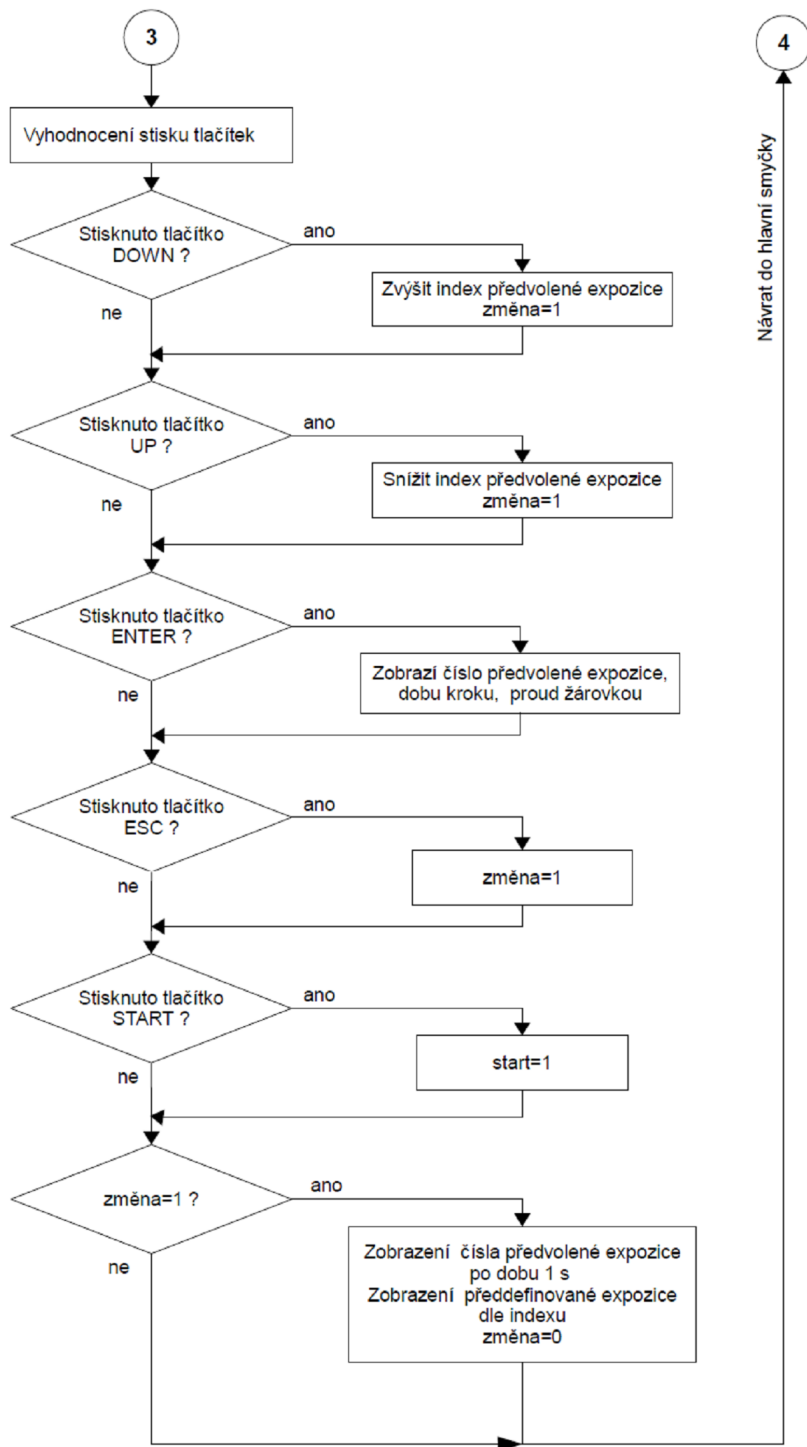
CR	Computed radiology
DR	Direct radiology
R	eská republika
NDT	Non Destructive Testing
RTG	Rentgenové záření
PC	Personal computer
A/D	Analogov digitální převodník
RAM	Random Acces Memory
EEPROM	Electrically Erasable Programmable Read Only Memory
FLASH	Elektricky programovatelná paměť typu RAM
MSB	Most significant byt
LSB	Lowest significant byt
LCD	Liquid Crystal Display
LED	Light-Emitting Diode
PET	Polyetylentereftalát
TTL	Transistor-Transistor-Logic
HW	Hardware

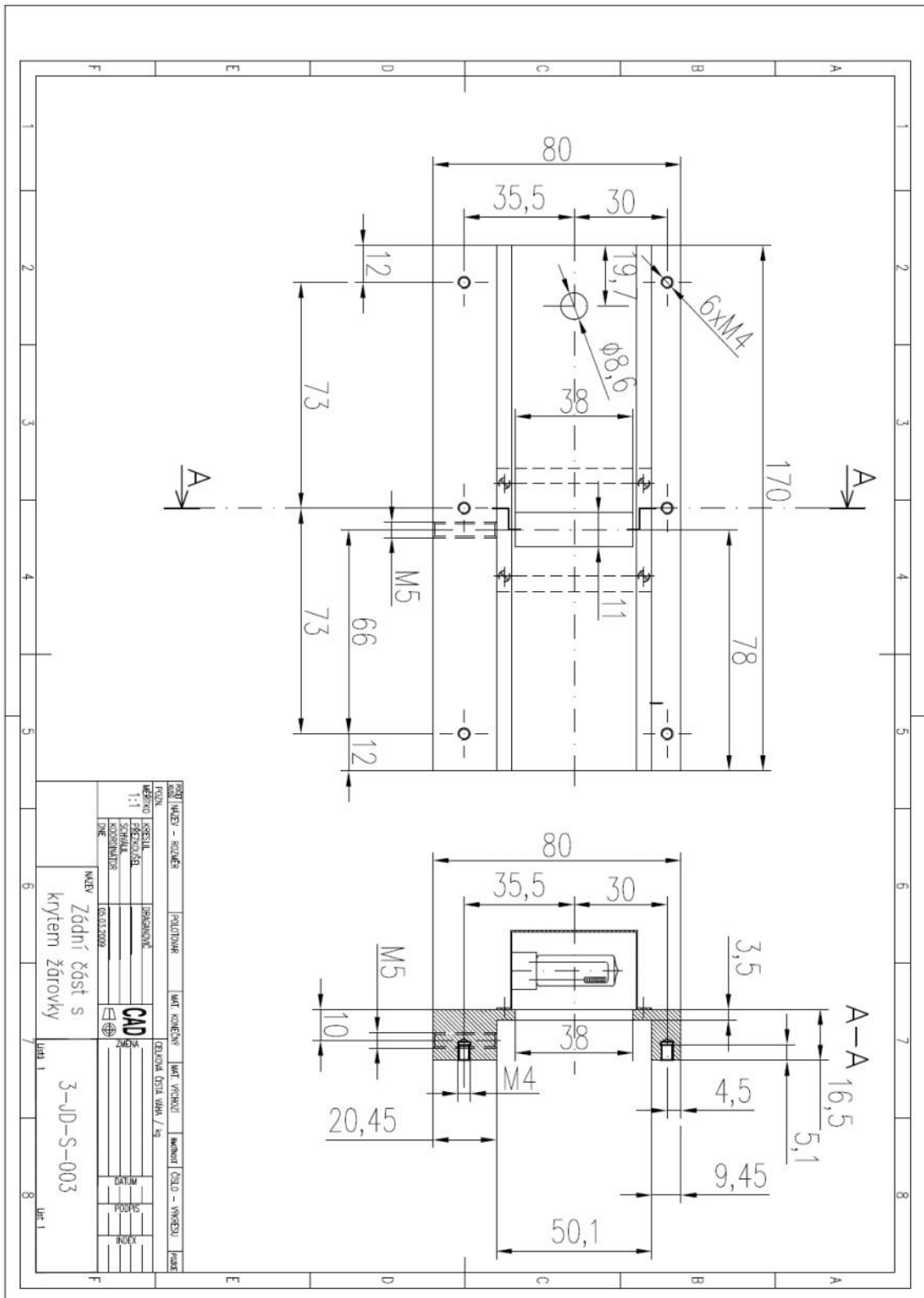
SEZNAM P ÍLOH

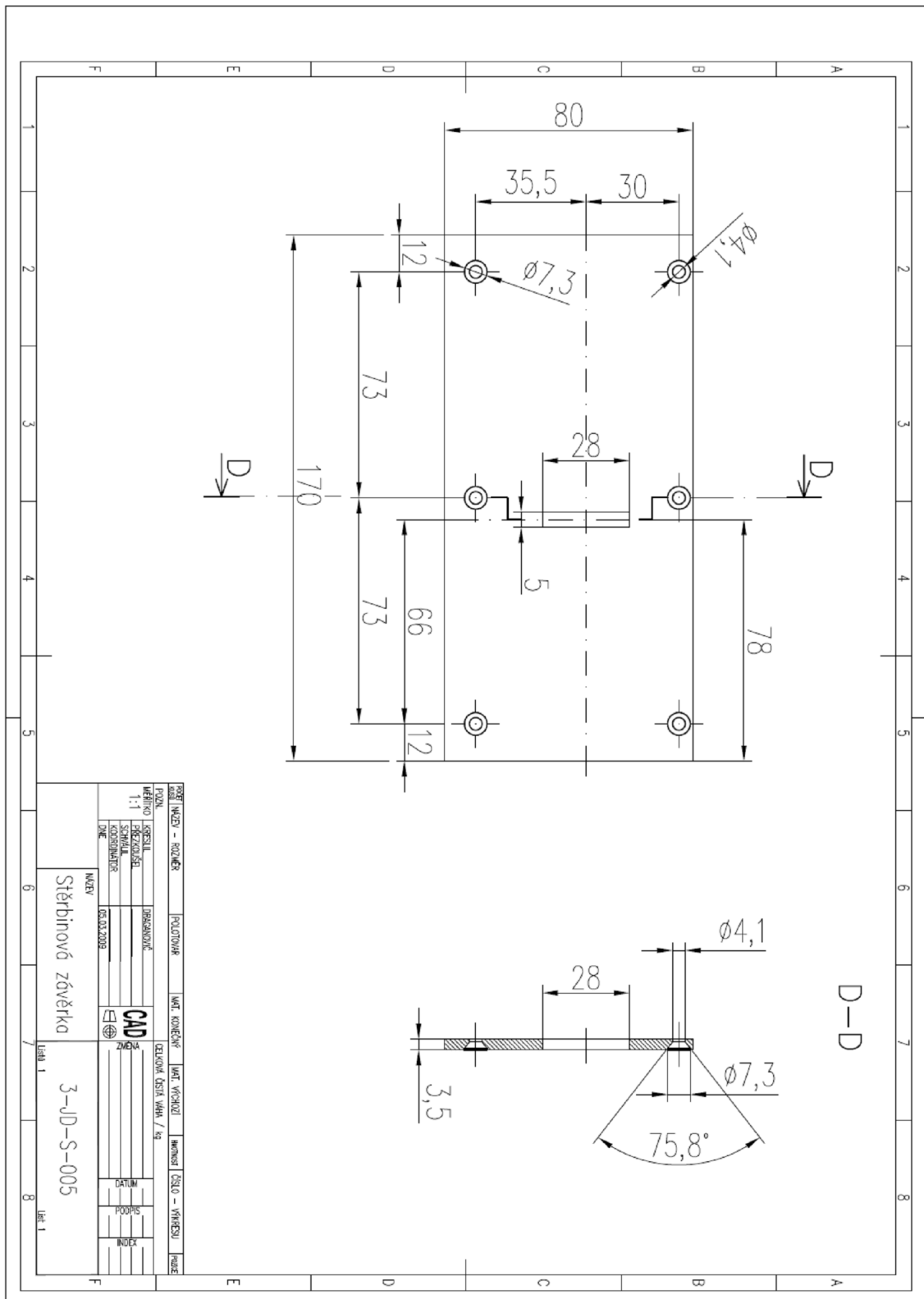
P ÍLOHA .1	V ývojový diagram hlavního programu
P ÍLOHA .2	V ýkresy mechanické ásti
P ÍLOHA .3	V ýpis programu na stran senzimetru; p íloha na CD



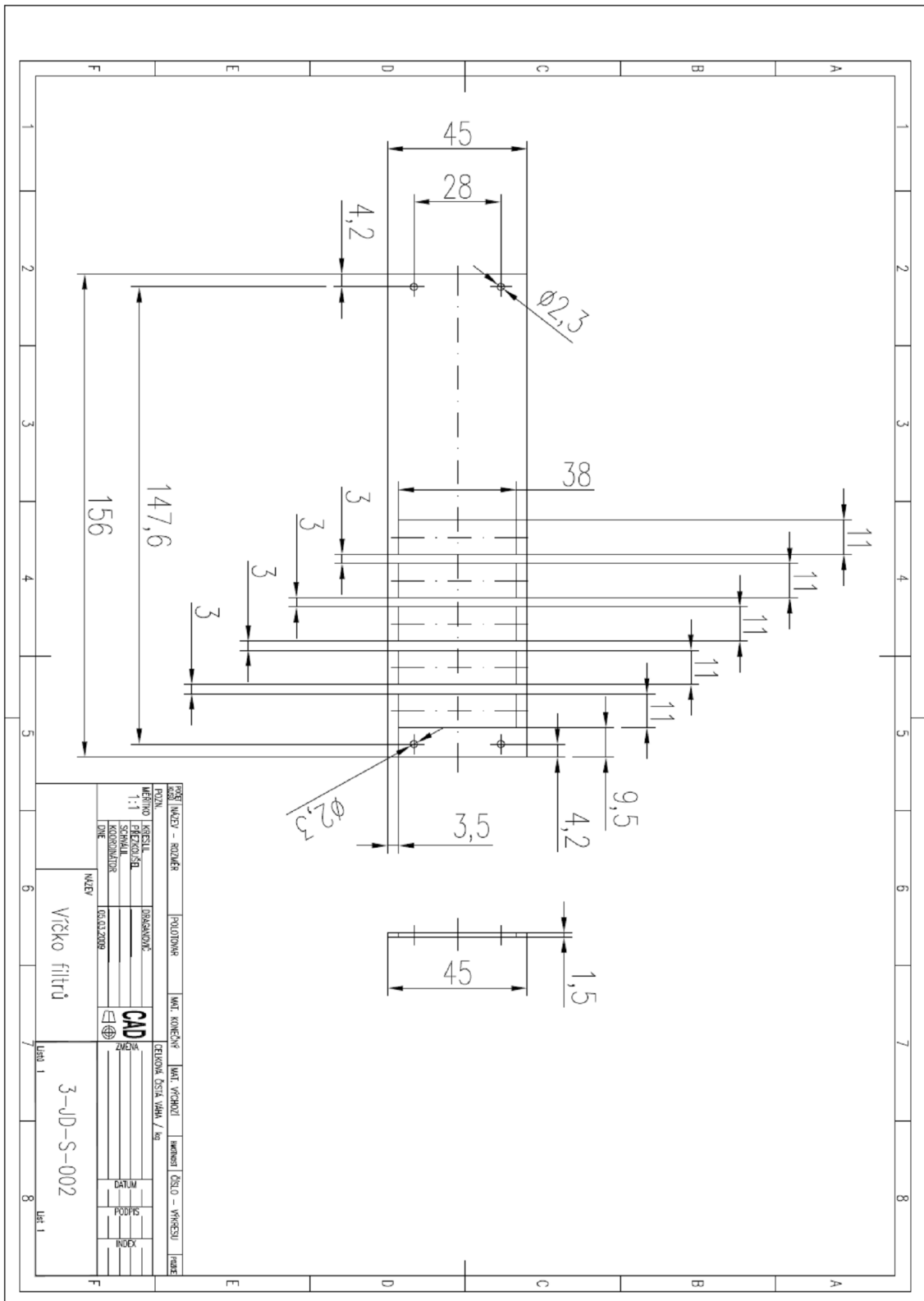


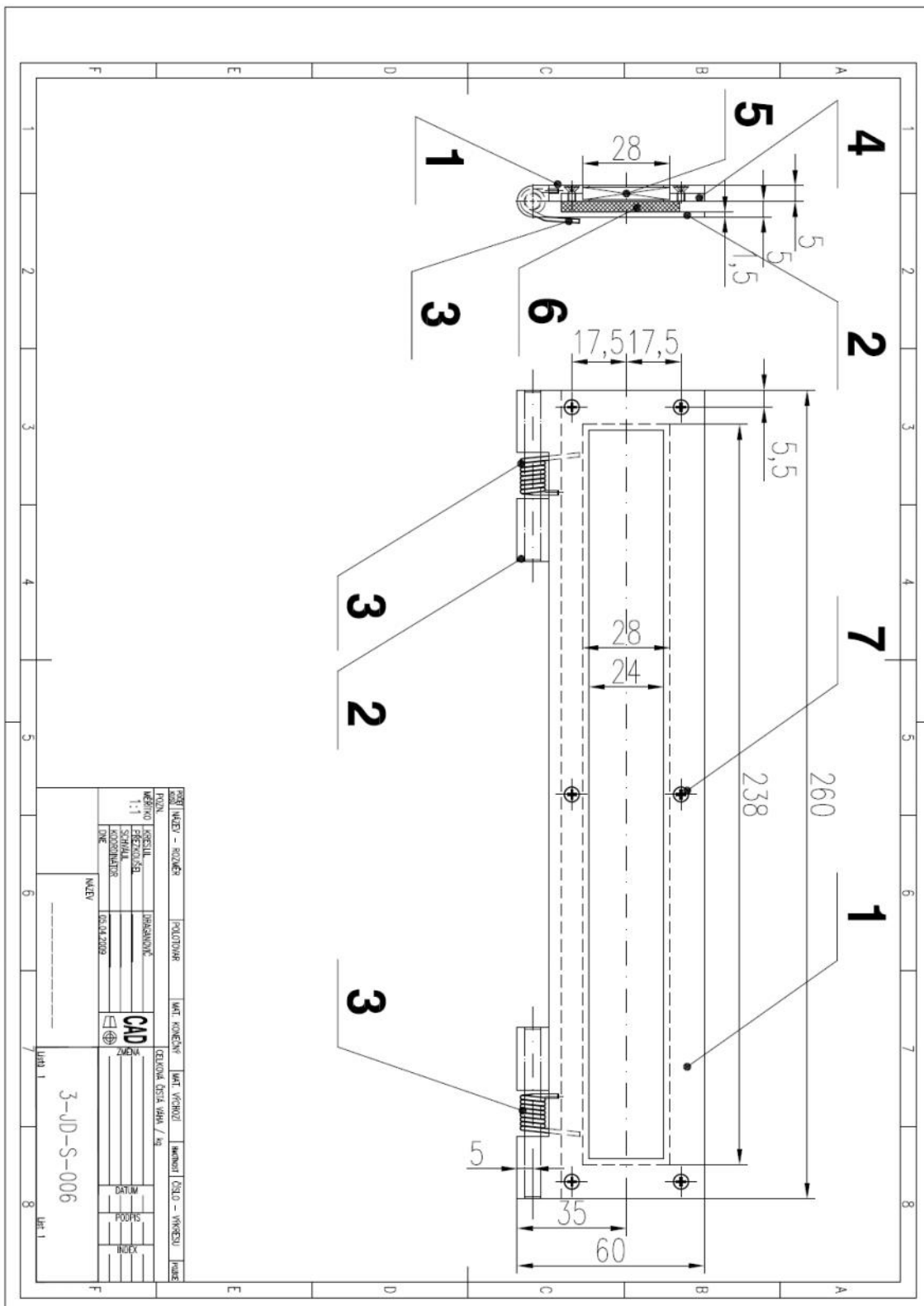


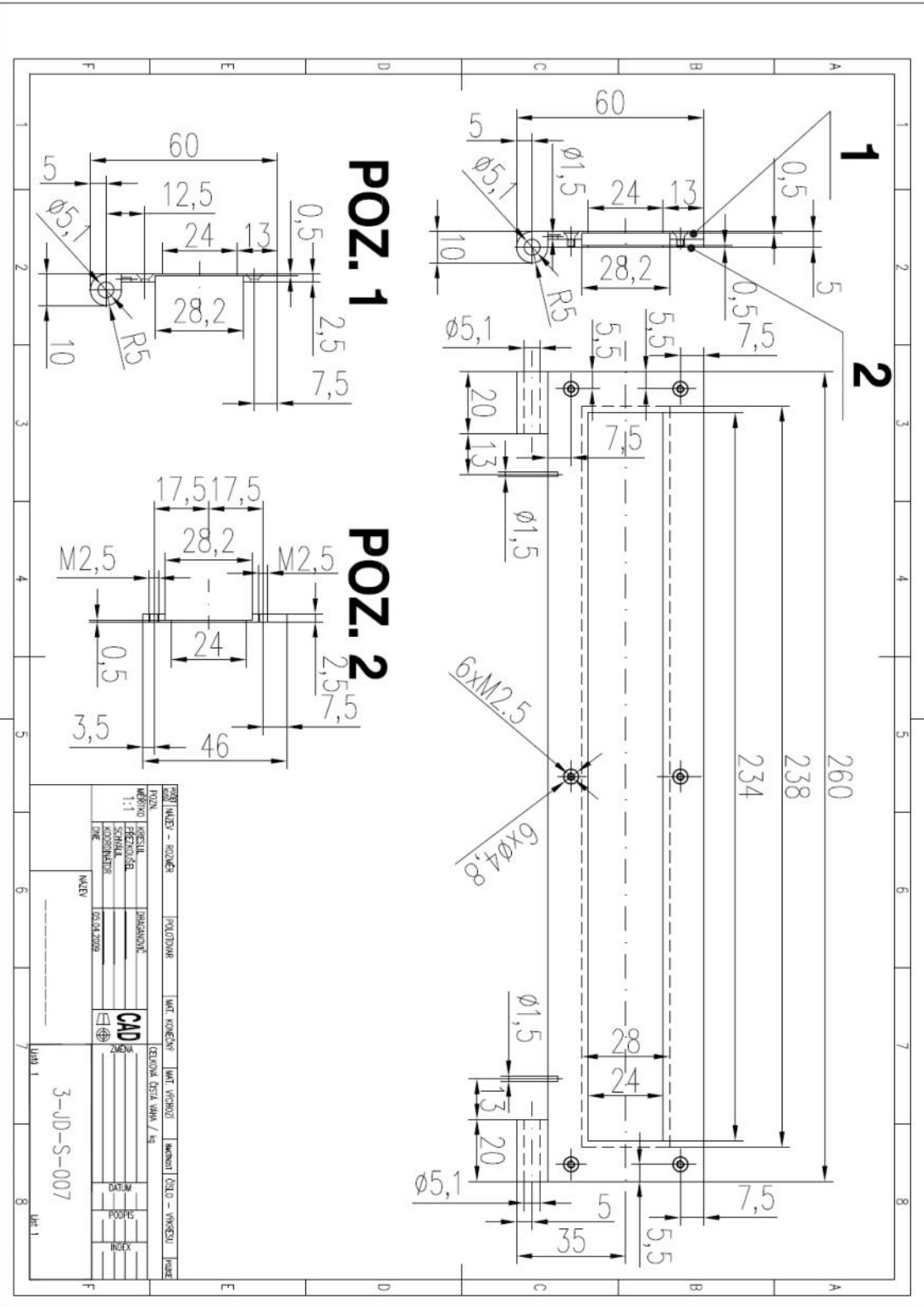


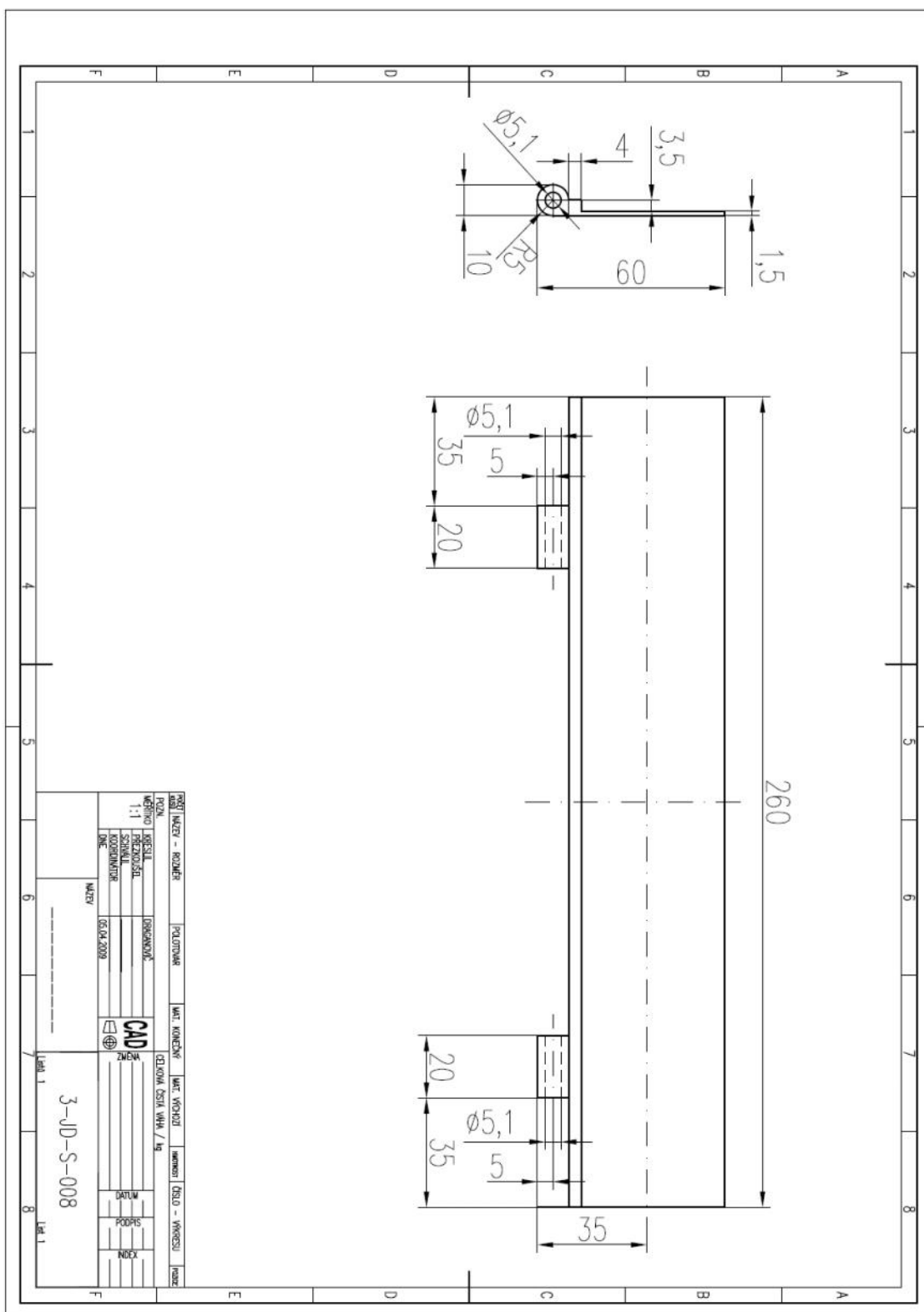


PROJEKT NÁZEV - ROZMĚRY		PROJEKTOVANÝ	UVAŽ. KONSTRUKTOR	UVAŽ. VYKONATEL	UVAŽOVANÝ	UVAŽOVANÝ	UVAŽOVANÝ	UVAŽOVANÝ
TYP DOKUMENTU		1:1	1:1	1:1	1:1	1:1	1:1	1:1
MĚŘÍTKO		1:1	1:1	1:1	1:1	1:1	1:1	1:1
SCHÉMA		1:1	1:1	1:1	1:1	1:1	1:1	1:1
KOORDINACE		1:1	1:1	1:1	1:1	1:1	1:1	1:1
DNE		05.03.2009	05.03.2009	05.03.2009	05.03.2009	05.03.2009	05.03.2009	05.03.2009
NÁZEV		Stěrbinová závěrka			3-JD-S-005			
LH. 1		LH. 1			LH. 1			









PROJEKCIJA: MATEMATIČKI		POLUPROJEKCIJA: PRVA		MATERIJAL: ČELIČNA OŠIČKA V4A / A4		MERNIŠKI SISTEM: SI		MATERIJALNA KODIRANJE: 3-UD-S-008	
PROJEKCIJA: MATEMATIČKI		POLUPROJEKCIJA: PRVA		MATERIJAL: ČELIČNA OŠIČKA V4A / A4		MERNIŠKI SISTEM: SI		MATERIJALNA KODIRANJE: 3-UD-S-008	
PROJEKCIJA: MATEMATIČKI		POLUPROJEKCIJA: PRVA		MATERIJAL: ČELIČNA OŠIČKA V4A / A4		MERNIŠKI SISTEM: SI		MATERIJALNA KODIRANJE: 3-UD-S-008	
PROJEKCIJA: MATEMATIČKI		POLUPROJEKCIJA: PRVA		MATERIJAL: ČELIČNA OŠIČKA V4A / A4		MERNIŠKI SISTEM: SI		MATERIJALNA KODIRANJE: 3-UD-S-008	
PROJEKCIJA: MATEMATIČKI		POLUPROJEKCIJA: PRVA		MATERIJAL: ČELIČNA OŠIČKA V4A / A4		MERNIŠKI SISTEM: SI		MATERIJALNA KODIRANJE: 3-UD-S-008	
PROJEKCIJA: MATEMATIČKI		POLUPROJEKCIJA: PRVA		MATERIJAL: ČELIČNA OŠIČKA V4A / A4		MERNIŠKI SISTEM: SI		MATERIJALNA KODIRANJE: 3-UD-S-008	
PROJEKCIJA: MATEMATIČKI		POLUPROJEKCIJA: PRVA		MATERIJAL: ČELIČNA OŠIČKA V4A / A4		MERNIŠKI SISTEM: SI		MATERIJALNA KODIRANJE: 3-UD-S-008	
PROJEKCIJA: MATEMATIČKI		POLUPROJEKCIJA: PRVA		MATERIJAL: ČELIČNA OŠIČKA V4A / A4		MERNIŠKI SISTEM: SI		MATERIJALNA KODIRANJE: 3-UD-S-008	
PROJEKCIJA: MATEMATIČKI		POLUPROJEKCIJA: PRVA		MATERIJAL: ČELIČNA OŠIČKA V4A / A4		MERNIŠKI SISTEM: SI		MATERIJALNA KODIRANJE: 3-UD-S-008	
PROJEKCIJA: MATEMATIČKI		POLUPROJEKCIJA: PRVA		MATERIJAL: ČELIČNA OŠIČKA V4A / A4		MERNIŠKI SISTEM: SI		MATERIJALNA KODIRANJE: 3-UD-S-008	

Výpis programu na stran senzimetru:

```

#ifndef _Inic_hw_H_
#define _Inic_hw_H_

//nastaveni T1
#define T1_PRESC_1 0b00000000
#define T1_PRESC_2 0b00010000
#define T1_PRESC_4 0b00100000
#define T1_PRESC_8 0b00110000
#define T1_OSC_EN 0b00001000
#define T1_OSC_DIS 0b00000000
#define T1_SYNC_DIS 0b00000100
#define T1_EXT_CLK 0b00000010
#define T1_INT_CLK 0b00000000
#define T1_ON 0b00000001

#define T1_INIT(x) (tlcon=(x))
#define T1_START tlcon.0=1
#define T1_STOP tlcon.0=0
#define T1_NUL tmr1h=0; tmr1l=0;
#define T1_F_NUL pir1.TMR1IF=0

//nastaveni T0
#define T0_PRESC_2 0b10000000
#define T0_PRESC_4 0b10000001
#define T0_PRESC_8 0b10000010
#define T0_PRESC_16 0b10000011
#define T0_PRESC_32 0b10000100
#define T0_PRESC_64 0b10000101
#define T0_PRESC_128 0b10000110
#define T0_PRESC_256 0b10000111
#define T0_PRE_WD_EN 0b10001000
#define T0_HRANA_HL 0b10010000
#define T0_EXT_CLK 0b10100000

#define T0_INIT(x) (option_reg=(x)) //odpory RB vypnuty,
INT na spadovou hranu
#define T0_NUL tmr0=0

//nastaveni T2
#define T2_PRESC_1 0b00000000
#define T2_PRESC_4 0b00000001
#define T2_PRESC_16 0b00000010
#define T2_ON 0b00000100
#define T2_POSC_1 0b00000000
#define T2_POSC_2 0b00001000
#define T2_POSC_3 0b00010000
#define T2_POSC_4 0b00011000

```

```

#define T2_POSC_5    0b00100000
#define T2_POSC_6    0b00101000
#define T2_POSC_7    0b00110000
#define T2_POSC_8    0b00111000
#define T2_POSC_9    0b01000000
#define T2_POSC_10   0b01001000
#define T2_POSC_11   0b01010000
#define T2_POSC_12   0b01011000
#define T2_POSC_13   0b01100000
#define T2_POSC_14   0b01101000
#define T2_POSC_15   0b01110000
#define T2_POSC_16   0b01111000

#define T2_INIT(x)   (t2con=(x))
#define T2_START     t2con.2=1
#define T2_STOP      t2con.2=0
#define T2_NUL       tmr2=0
#define T2_F_NUL     pir1.TMR2IF=0

//nastaveni CCP1CON
//bity pro PWM cyklus=0
#define CCP_DIS      0
#define CCP_CAP_L    4
#define CCP_CAP_H    5
#define CCP_CAP_4H   6
#define CCP_CAP_16H  7
#define CCP_COM_H    8
#define CCP_COM_L    9
#define CCP_COM_U    10
#define CCP_COM_ST   11
#define CCP_PWM      12

//Vytvori 32 bit long ze 4 8-bit promennych
#define MAKELONG( dst, lobyte, mbytel, mbyteh, hibyte )      asm
movf _##lobyte, W \
                                asm movwf _##dst \
                                asm movf _##mbytel, W \
                                asm movwf _##dst##+1 \
                                asm movf _##mbyteh, W \
                                asm movwf _##dst##+2 \
                                asm movf _##hibyte, W \
                                asm movwf _##dst##+3

//provede swap 4 bitu
#define SWAP4(src)   asm swapf _##src,F

//ulozi low zdroje do pole, cil muze byt ve tvaru nazev+index

```



```

#define LOBYTE_RET( dst,src )      asm movf _##src, W \
                                   asm movwf _##dst

//ulozi high zdroje do pole, cil muze byt ve tvaru nazev+index
#define HIBYTE_RET( dst,src )      asm movf _##src+1, W \
                                   asm movwf _##dst

typedef unsigned char byte;

#endif // _Inic_hw_H_
/*****
* 7-pins LCD interface code
*
* Drives an LCD in 4-bit mode, using the three control
* signals. This has the advantage that the code is
* independent of the LCD, since there are no
* timing loops in the code. The code is blocking
* and used the LCD busy signal to determine if the
* display is ready to accept new data.
*
* (c) Lieven Hollevoet
* most recent version on http://boostc.lika.be
*****/
typedef unsigned char byte;

#include "lcd.h"

// Local variable
bit lcd_found = 0;

// Drive a certain nibble on the LCD data pins, the nibble
// is taken from the lsb's of the input char
void lcd_clock_nibble(char nibble){

    // Since there seems to be no way to extract a bit from a
    byte in BoostC -> do it in asm.
    // It is replicated 4 times here, since a function cannot
    return a bit eiter.

    // Set the d4 depending on the state of nibble bit 0
    asm {
    btfss _nibble, 0
    goto clear_lcd4
    bsf    _lcd_d4, F
    goto  done_lcd4
clear_lcd4:

```

```

    bcf    _lcd_d4, F
done_lcd4:
}

// Set the d5 depending on the state of nibble bit 1
asm {
btfss _nibble, 1
goto  clear_lcd5
bsf    _lcd_d5, F
goto  done_lcd5
clear_lcd5:
bcf    _lcd_d5, F
done_lcd5:
}

// Set the d6 depending on the state of nibble bit 2
asm {
btfss _nibble, 2
goto  clear_lcd6
bsf    _lcd_d6, F
goto  done_lcd6
clear_lcd6:
bcf    _lcd_d6, F
done_lcd6:
}

// Set the d7 depending on the state of nibble bit 3
asm {
btfss _nibble, 3
goto  clear_lcd7
bsf    _lcd_d7, F
goto  done_lcd7
clear_lcd7:
bcf    _lcd_d7, F
done_lcd7:
}

    lcd_toggle_e;

    return;
}

////////////////////////////////////
// Read the busy flag of the LCD and wait to return until
// the LCD is ready.
// Skipped when running in debug mode.
// dotaz nahrazen cekanim
////////////////////////////////////
void lcd_wait_busy()
{

```

```

        // Skip this function if we are running in debug mode.
#ifdef DEBUG
        return;
#endif

        // If we have not detected an LCD in a previous call to
this function,
        // then we just skip it this time, since else the program
will hang later
        // on in this function.
        if (!lcd_found){
            return;
        }

        delay_10us(10);        //cekani 100 us

        // Save the current LCD mode
        bit mode = lcd_rs;

/*
        // The data lines are inputs
        lcd_tris_d4 = 1;
        lcd_tris_d5 = 1;
        lcd_tris_d6 = 1;
        lcd_tris_d7 = 1;

        // Enter command mode
        lcd_cmd_mode;
        lcd_rw      = 1;

        // Put lcd_e high
        lcd_e_hi;

        char counter = 0;

        // Wait for completion of the operation, with a timeout of
~.5 seconds
        // LCD d7 is high if the operation is complete.
        while (lcd_d7 && counter < 0xFF){
            lcd_e_lo;
            lcd_e_hi;
            lcd_e_lo;
            lcd_e_hi;
            delay_ms(2);
            counter++;
        }

        // Check if the previous loop timed out
        if (counter == 0xFF) {
            // If it was a timeout -> disable the flag.

```

```

        lcd_found = 0;
    }

    // And put the lcd_lo again
    lcd_e_lo;
*/

    // Restore the TRIS
    lcd_tris_d4 = 0;
    lcd_tris_d5 = 0;
    lcd_tris_d6 = 0;
    lcd_tris_d7 = 0;

    // Restore LCD RS mode.
    lcd_rs = mode;
    lcd_rw = 0;

    return;
}

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
// Send a byte to the LCD. Don't forget to set the mode
// (data or cmd) before calling this function.
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
void lcd_send_byte(char data){

    char temp = data >> 4;

    lcd_rw = 0;    // Write mode

    // Clock the high nibble
    lcd_clock_nibble(temp);

    // Clock the low nibble
    lcd_clock_nibble(data);

    // Wait until the LCD is finished
    lcd_wait_busy();
}

////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
// Initialisation sequence of the LCD
////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
void lcd_init(){

    // Wait for internal reset
    delay_ms(15);

    // Set all pins as output
    lcd_tris_rs = 0;
    lcd_tris_rw = 0;
}

```

```

    lcd_tris_e = 0;
    lcd_tris_d4 = 0;
    lcd_tris_d5 = 0;
    lcd_tris_d6 = 0;
    lcd_tris_d7 = 0;

    //clear_bit(trisa, 1);
    //clear_bit(porta, 1);

    // Clear the control bits
    lcd_rw = 0;
    lcd_cmd_mode;

    // Init sequence (see datasheet)
    /// attention (clock 0x03 twice)
    lcd_clock_nibble(0x03);
    delay_ms(5);
    lcd_toggle_e;
    delay_10us(15);
    /// 4bit mode (clock 0x03 and 0x02)
    lcd_toggle_e;
    delay_ms(5);
    lcd_d4 = 0;
    lcd_toggle_e;
    delay_10us(15);

    // Assume we have an LCD attached
    lcd_found = 1;

    /// We're in 4 bit mode now, program general settings
    lcd_send_byte(FUNCTION_SET); //4 bit, 2 (4) radky, 5x7
    bodu
    lcd_send_byte(DISP_OFF); //displ. vypnut, kurzor vypnut,
    blikani vypnuto
    lcd_send_byte(DISP_ON); //displ. zapnut, kurzor
    vypnut, blikani vypnuto
    lcd_send_byte(ENTRY_INC); //inkrement adresy, posun
    kurzoru
    lcd_send_byte(DISP_CLR); //mazani
    delay_ms(5); //pokud neni cteni busy
    lcd_send_byte(LINE1); //adresa prvni znak
    prvniho radku

    // Init done, you can start writing characters now
    lcd_data_mode;

    return;
}

```

////////////////////////////////////

```

// Send a command to the LCD
///////////////////////////////////////////////////////////////////
void lcd_send_cmd(char command){
    lcd_cmd_mode;
    lcd_send_byte(command);
}

///////////////////////////////////////////////////////////////////
// Posli data na LCD
///////////////////////////////////////////////////////////////////
void lcd_send_data(char command)
{
    lcd_data_mode;
    lcd_send_byte(command);
}

///////////////////////////////////////////////////////////////////
// Send a line to the LCD
///////////////////////////////////////////////////////////////////
void lcd_printf(const char* line){
    char i = 0;
    lcd_data_mode;
    while(line[i] != 0)
        lcd_send_byte(line[i++]);
}

///////////////////////////////////////////////////////////////////
// Print a character to the LCD
///////////////////////////////////////////////////////////////////
void lcd_printf(char value){
    lcd_data_mode;
    lcd_send_byte(value);
}

///////////////////////////////////////////////////////////////////
// Print a hex value to the LCD
///////////////////////////////////////////////////////////////////
void lcd_print_hex(char value){
    lcd_data_mode;

    char hexChar;
    char i;

    for(i = 0; i < 2; i++)
    {
        if(i == 0)
            hexChar = value >> 4;
        else
            hexChar = value & 0x0F;
        if(hexChar < 10)

```

```

        hexChar = hexChar + '0';
    else
        hexChar = hexChar + ('A' - 10);
    lcd_send_byte(hexChar);
}
}

/////////////////////////////////////////////////////////////////
// Print a 16-bit hex value to the LCD
/////////////////////////////////////////////////////////////////
void lcd_print_hex(short value){

    char value1 = (char)((value >> 8) & 0x00FF);

    lcd_print_hex(value1);

    char value0 = (char)(value & 0x00FF);

    lcd_print_hex(value0);

    return;
}

//nastavi kurzor na pozici x=0 az 19, y=0 az 3 shora
void nast_kurz(byte pol_x, byte pol_y)
{
    switch(pol_y)
    {
        case 0: lcd_send_cmd(0x80+pol_x); break; /* radek
0 */
        case 1: lcd_send_cmd(0x80+0x40+pol_x); break;
/* radek 1 */
        //case 2: lcd_send_cmd(0x80+0x14+pol_x);
break; /* radek 2 */
        //case 3: lcd_send_cmd(0x80+0x54+pol_x);
break; /* radek 3 */
    }
    //sx=pol_x; sy=pol_y;
}

/* zapise znak na pozici x=0 az 19, y=0 az 3 shora */
void pis_znak_xy(byte pol_x, byte pol_y, char znak)
{
    nast_kurz(pol_x,pol_y); /*nastaveni polohy kurzoru */
    lcd_data_mode;
    lcd_send_byte(znak);
}

/* zapise retezec na pozici x=0 az 19, y=0 az 3 shora */
void pis_ret_xy(byte pol_x, byte pol_y, char *ret)

```

```

{ byte i;
  nast_kurz(pol_x,pol_y);      /*nastaveni polohy kurzoru */
  i=0;
  lcd_data_mode;
  while(ret[i]) lcd_send_byte(ret[i++]);
}

// zapise byte na pozici x=0 az 19, y=0 az 3 shora, 3 znaky
void pis_byte3_xy(byte pol_x, byte pol_y, byte cislo)
{ byte i;

  nast_kurz(pol_x,pol_y);      //nastaveni polohy kurzoru
  lcd_data_mode;
  i=cislo/100; cislo%=100; if(i==0) i=' '; else i+='0';
  lcd_send_byte(i);
  i=cislo/10; cislo%=10; i=(i==0)?' ':i+'0';
  lcd_send_byte(i);
  cislo+='0'; lcd_send_byte((byte)cislo);
}

// zapise cislo na pozici x=0 az 19, y=0 az 3 shora, pocet
cisel
void pis_uint_xy(byte pol_x, byte pol_y,unsigned int cislo,
byte poc_dig)
{ signed char i,j=4;
  char ret[5];

  //if(poc_dig>10) poc_dig-=10;
  nast_kurz(pol_x,pol_y); //nastaveni polohy kurzoru
  lcd_data_mode;
  i=cislo/10000; cislo%=10000; i=i+'0'; ret[j--]=i;
  i=cislo/1000; cislo%=1000; i=i+'0'; ret[j--]=i;
  i=cislo/100; cislo%=100; i=i+'0'; ret[j--]=i;
  i=cislo/10; cislo%=10; i=i+'0'; ret[j--]=i;
  ret[0]=cislo+'0';
  //for(j=4;j>0;j--) if(ret[j]=='0') ret[j]=' '; else break;
nahrada levych 0
  for((j=poc_dig-1);j>=0;j--) lcd_send_byte(ret[j]);
}

//smaze displej
void display_smaz(void)
{ lcd_send_cmd(DISP_CLR);
  delay_ms(5); //pokud neni cteni busy
  lcd_data_mode;
}
/*****
* 7-pins LCD interface code
*

```



```

* Drives an LCD in 4-bit mode, using the three control
* signals. This has the advantage that the code is
* independent of the LCD, since there are no
* timing loops in the code. The code is blocking
* and used the LCD busy signal to determine if the
* display is ready to accept new data.
*
* (c) Lieven Hollevoet
* most recent version on http://boostc.lika.be
*****/

#ifndef _LCD_H_
#define _LCD_H_

#include <system.h>

// Define the LCD connections here
#define LCD_E_PORT          PORTB
#define LCD_RS_PORT        PORTB
#define LCD_RW_PORT        PORTB
#define LCD_DATA4_PORT     PORTB
#define LCD_DATA5_PORT     PORTB
#define LCD_DATA6_PORT     PORTB
#define LCD_DATA7_PORT     PORTB

#define LCD_E_TRIS         TRISB
#define LCD_RS_TRIS        TRISB
#define LCD_RW_TRIS        TRISB
#define LCD_DATA4_TRIS     TRISB
#define LCD_DATA5_TRIS     TRISB
#define LCD_DATA6_TRIS     TRISB
#define LCD_DATA7_TRIS     TRISB

#define LCD_E_PIN          1
#define LCD_RS_PIN         3
#define LCD_RW_PIN         2
#define LCD_DATA4_PIN      4
#define LCD_DATA5_PIN      5
#define LCD_DATA6_PIN      6
#define LCD_DATA7_PIN      7

////////////////////////////////////
////
// Don't change below this line
volatile bit lcd_e        @ LCD_E_PORT      . LCD_E_PIN;
volatile bit lcd_rs       @ LCD_RS_PORT     . LCD_RS_PIN;
volatile bit lcd_rw       @ LCD_RW_PORT     . LCD_RW_PIN;
volatile bit lcd_d4       @ LCD_DATA4_PORT  . LCD_DATA4_PIN;

```

```

volatile bit lcd_d5          @ LCD_DATA5_PORT . LCD_DATA5_PIN;
volatile bit lcd_d6          @ LCD_DATA6_PORT . LCD_DATA6_PIN;
volatile bit lcd_d7          @ LCD_DATA7_PORT . LCD_DATA7_PIN;

volatile bit lcd_tris_e @ LCD_E_TRIS      . LCD_E_PIN;
volatile bit lcd_tris_rs @ LCD_RS_TRIS    . LCD_RS_PIN;
volatile bit lcd_tris_rw @ LCD_RW_TRIS    . LCD_RW_PIN;

bit lcd_tris_d4 @ LCD_DATA4_TRIS . LCD_DATA4_PIN;
bit lcd_tris_d5 @ LCD_DATA5_TRIS . LCD_DATA5_PIN;
bit lcd_tris_d6 @ LCD_DATA6_TRIS . LCD_DATA6_PIN;
bit lcd_tris_d7 @ LCD_DATA7_TRIS . LCD_DATA7_PIN;

// LCD command set
#define LINE1          0x80      // set display to line 1
character 0
#define LINE2          0xC0      // set display to line 2
character 0
#define FUNCTION_SET   0x28      // 4 bits, >1 lines,
5x7 Font
#define DISP_ON        0x0C      // display on
#define DISP_ON_C      0x0E      // display on, Cursor on
#define DISP_ON_B      0x0F      // display on, Cursor on,
Blink cursor
#define DISP_OFF       0x08      // display off
#define DISP_CLR       0x01      // clear the Display
#define ENTRY_INC      0x06      // increment-mode, display
shift OFF
#define ENTRY_INC_S    0x07      // increment-mode, display
shift ON
#define ENTRY_DEC      0x04      // decrement-mode, display
shift OFF
#define ENTRY_DEC_S    0x05      // decrement-mode, display
shift ON
#define DD_RAM_ADDR    0x80      // Least Significant 7-bit
are for address

#define VYP_KURZ lcd_send_cmd(0xC); /* displ. zapnut,
kurzor vypnut, blikani vypnuto */
#define ZAP_KURZ lcd_send_cmd(0xD); /* ?displ. zapnut,
kurzor zapnut, blikani vypnuto asi 0xE */
#define MAZ_DISPL lcd_send_cmd(1); /* displ. zapnut,
kurzor zapnut, blikani vypnuto */

// Function definitions
void lcd_init();
void lcd_send_byte(char data);
void lcd_data_mode();

```

```

void lcd_cmd_mode();
void lcd_send_cmd(char command);
void lcd_printf(const char* line);
void lcd_printf(char value);
void lcd_print_hex(char value);
void lcd_print_hex(short value);

void nast_kurz(byte pol_x, byte pol_y);
void pis_znak_xy(byte pol_x, byte pol_y, char znak);
void pis_ret_xy(byte pol_x, byte pol_y, char *ret);

// Macro's
#define lcd_data_mode lcd_rs = 1
#define lcd_cmd_mode lcd_rs = 0
#define lcd_e_hi lcd_e = 1
#define lcd_e_lo lcd_e = 0
#define lcd_toggle_e lcd_e_hi ; lcd_e_lo

#endif // _LCD_H_
#include <system.h>
#include <flash.h>
#include <stdlib.h>
#include <EEPROM.h>
#include "Inic_hw.h"
#include "senzit.h"
#include "lcd.h"

void j_tlacitko(bit h_tlac, const byte index)
{ signed char citac;
  citac=tlac[index].citac;
  if(h_tlac==STISK) {if(citac>=0) citac++; else citac=0;}
  else {if(citac<=0) citac--; else citac=0;}
  if(citac==100) citac=99;
  if(citac==-100) citac=-99;
  if(citac<C_NE_STISK) {tlac[index].stisk=NE_STISK;
tlac[index].stav=NE_STISK;};
  if(citac>C_K_STISK )
  { if(tlac[index].stav==NE_STISK)
tlac[index].stisk=K_STISK; else tlac[index].stisk=NE_STISK;}
  if(citac>C_D_STISK)
  { if(tlac[index].stav==NE_STISK)
tlac[index].stisk=D_STISK; else tlac[index].stisk=NE_STISK;}
  if(tlac[index].stav!=NE_STISK) tlac[index].stisk=NE_STISK;
  tlac[index].citac=citac;
}

//vyhodnocuje stavy tlacitek, meni prislusnou strukturu

```

```

//ocekava se periodicke volani
void tlacitka(void)
{
    j_tlacitko(T_ENT,ENTER);
    j_tlacitko(T_ESC,ESC);
    j_tlacitko(T_UP,UP);
    j_tlacitko(T_DOWN,DOWN);
    j_tlacitko(T_START,START);
}

//pripravi chod KM, nastartuje T1, po CCP1IF prvni krok
void priprava_KM(bit smer)
{
    O_SM_KME=smer; O_EN_KME=KM_EN_E; //smer a povoleni chodu
    delay_ms(5); //pro jistotu pro nabuzeni proudu
    ccplcon=0; //vystup do 0
    expozice.per_KM=flash_read(ADR_ROZBEH0); //pocatecni
    rozbehova perioda
    if(expozice.per_rychlost>expozice.per_KM) //bez zrychleni
        expozice.ind_rampa=255; else expozice.ind_rampa=0;
    tmr1h=0; tmr1l=1; //komparace nenastane hned
    cit_per_HW=PRODL_KROK_1; //komparace do 1. kroku
    ccplcon=CCP_COM_H; //predvolba komparace, vystup do H,
    potom nastaveno CCP1IF
    T1_NUL; T1_START;
    pirl.CCP1IF=0; //shozeni flagu
    expozice.akt_draha=0;
    if(expozice.ind_rampa==255)
    expozice.per_KM=expozice.per_rychlost; //rovnou cilova
}

//pripravi dalsi periodu KM, ridi rozbeh po rampe, pripocte k
delce drahy
//pro krok vraci 1
byte chod_KM(void)
{
    if(pirl.CCP1IF==0) return 0; //neni komparace
    pirl.CCP1IF=0; //shozeni flagu
    if(ccplcon.0==0) //vystup presel do 1, byl krok
        { expozice.akt_draha++;
        if(expozice.ind_rampa!=255)
            {
    expozice.per_KM=flash_read(ADR_ROZBEH0+expozice.ind_rampa);
            if(expozice.per_KM<=expozice.per_rychlost) //cilova
dosazena
                { expozice.per_KM=expozice.per_rychlost;
                expozice.ind_rampa=255; //konec zrychlovani
                }
            else expozice.ind_rampa++;
            cit_per_HW+=expozice.per_KM-CLK_L;
            }
        }
}

```

```

        //if(k<30) pole[k++]=expozice.per_KM;
    }
    else
    {   cit_per_HW+=expozice.per_rychlost -CLK_L;   //cilova
perioda
    }
        ccplcon.0=1;           //hodnota 0 po dalsi shode
    }
    else
    {   cit_per_HW+=CLK_L;
        ccplcon.0=0;           //hodnota 1 po dalsi shode a
krok KM
    }
    return 1;
}

```

```

//zapise hodnotu 10 b do PWM
inline void pis_PWM2_10(unsigned int &hodnota)
{   byte pom;
    ccp2con.4=hodnota.0;       //hodnota<<4;
    ccp2con.5=hodnota.1;
    pom=hodnota>>2;
    ccpr2l=pom;
}

```

```

//zapise hodnotu 8 b (od Msb) do PWM
inline void pis_PWM2_8(byte hodnota)
{   ccp2con.CCP1X=0; ccp2con.CCP1Y=0;
    ccpr2l=hodnota;
}

```

```

//zobrazí zaznam z eeprom na LCD a nacte do struktury
void zobr_zaznam(byte index)
{ char ret[26],ret_pom[6];
  byte adr,i;
  struct Predv_ee p_e;
  unsigned int cislo=0;

  adr=index*25; //adresa
  //nacisti
  for(i=0;i<26;i++) ret[i]=eeprom_read(adr++); //cely zaznam
display_smaz();
  //nazev materialu
  for(i=0;i<16;i++) p_e.nazev[i]=ret[i];
  p_e.nazev[i]=0; //konec retezce
  //cas expozice
  MAKESHORT(cislo,ret+16,ret+17); //cas v ms
}

```

```

        uitoa_dec(ret_pom,cislo,5); //text 5 mist, uvodni nuly
        if(ret_pom[0]!='0') //rozhodnuti, kde bude des. tecka
        {
p_e.exp[0]=ret_pom[1];p_e.exp[1]=',';p_e.exp[2]=ret_pom[2];p_e
.exp[3]=ret_pom[3];p_e.exp[4]=0;}
        else
        {
p_e.exp[0]=ret_pom[0];p_e.exp[1]=ret_pom[1];p_e.exp[2]=',';p_e
.exp[3]=ret_pom[2];p_e.exp[4]=0;}
        //barevna teplota
        MAKESHORT(cislo,ret+18,ret+19);
        uitoa_dec(p_e.t_barva,cislo,4); //text 4 mist, uvodni
nuly
        //cislo filtru
        p_e.filtr=ret[20]; expozice.filtr=ret[20];
        //perioda pro KM v nas. 12.8 us
        MAKESHORT(cislo,ret+21,ret+22);
        p_e.per_rychlost=cislo; expozice.per_rychlost=cislo*8;
//zadava se v jinych jednotkach
        expozice.per_rychlost_E=expozice.per_rychlost;
        //proud zarovkou v jednotkach prevodniku
        MAKESHORT(cislo,ret+23,ret+24);
        p_e.zad_proud=cislo; expozice.zad_proud=cislo;
        p_e.index=index; //zvoleny zaznam

        //zobrazeni
        pis_ret_xy(0,0,p_e.nazev); // prvni radek
        pis_ret_xy(0,1,p_e.exp); // cas expozice
        pis_znak_xy(4,1,'s'); // znak s
        pis_ret_xy(6,1,p_e.t_barva); // barevna teplota
        pis_znak_xy(10,1,'K'); // znak K
        pis_znak_xy(12,1,'F'); // znak F
        pis_znak_xy(13,1,'='); // znak =
        pis_znak_xy(14,1,p_e.filtr+'0'); // cislo filtru
    }

//navrat voziku do vychozi polohy, testuje KS, pokud sepnut,
vraci 1
byte navr_voz_ks(void)
{ unsigned int draha;

        ccplcon=0; //jednotka komparace vypnuta
        O_SM_KME=KM_VZAD_E; //smer vzad (KM_VPRED_E)
        O_KME=0; //vystup portu
        O_EN_KME=KM_EN_E; //povoleni chodu (KM_DIS_E)

        for(draha=0;draha<NAVR_DRAHA_INI;draha++)
            { if(I_K_STRT_VOZIK==JE_POLOHA) {O_EN_KME=KM_DIS_E;
return 1;} //nalezeno, m. vypnut
                O_KME=1;

```

```

        delay_100us(NAVR_PER_INI);
        O_KME=0;
        delay_100us(1);
    }
    O_EN_KME=KM_DIS_E;
    return 0; //nenalezeno, m. vypnut
}

//pohyb filtru
void pohyb_filtr(bit smer,unsigned int draha)
{
    O_KMF=0; //vystup portu
    O_SM_KMF=smer; //smer chodu
    O_EN_KMF=KM_EN_F; //povoleni chodu (KM_DIS_F)
    for(;draha>0;draha--)
        { O_KMF=1;
          delay_100us(PER_CHOD);
          O_KMF=0;
          delay_100us(1);
        }
    O_EN_KMF=KM_DIS_F; //m. vypnut
    return ;
}

//precti kanal a vrati 10 bit hodnotu
unsigned int cti_AD(void)
{
    unsigned int vysledek;
    adcon0.GO_DONE=1; //start prevodu
    while(adcon0.GO_DONE!=1); //cekani na ukonceni asi 20
us
    MAKESHORT(vysledek,adresl,adresh);
    return vysledek;
}

//nastavi zadany proud zarovnou v jednotkach p prevodniku
void nastav_proud(void)
{ int proud_m=0,proud=0,delta;
  byte p_cyklu=0;

  expozice.skut_proud=(expozice.zad_proud*NAS_PROUD)/DEL_PRO
UD; //prvotni nastaveni
  pis_PWM2_10(expozice.skut_proud);
  while(1)
    { delay_ms(D_MEZI_MER);
      proud=cti_AD();
      if(abs(proud-proud_m)<UKLID_I) break; //proud
uklidnen, konec cekani
      proud_m=proud;
      p_cyklu++; if(p_cyklu>D_NAST1) return
PREKR_DOBA;
}

```

```

    }
    p_cyklu=0;
    while(1)
        {
            proud=cti_AD();
            delta=expozice.zad_proud-proud;
            if(abs(delta)<POV_ODCH) return ; //proud je v
mezich
            if(delta>0)          expozice.skut_proud++;          else
expozice.skut_proud--;
            if(expozice.skut_proud>1023          ||
expozice.skut_proud==0)
                {expozice.skut_proud=NELZE_REG; return;}
            pis_PWM2_10(expozice.skut_proud);
            delay_ms(D_MEZI_INT1);
            p_cyklu++; if(p_cyklu>D_NAST2)
                { expozice.skut_proud=PREKR_DOBA; return;}
        }
}

```

//reguluje zadany proud zarovkou v jednotkach prevodniku

```
void reguluj_proud(void)
```

```
{ int proud,delta;
```

```

    proud=cti_AD();
    delta=expozice.zad_proud-proud;
    if(abs(delta)<POV_ODCH) return; //proud je v mezich
    if(delta>0)          expozice.skut_proud++;          else
expozice.skut_proud--;
    if(expozice.skut_proud>1023)      {expozice.skut_proud=1023;
return;} //bez zmeny
    if(expozice.skut_proud==0)      {expozice.skut_proud=0;
return;}
    pis_PWM2_10(expozice.skut_proud);
}

```

```
#include <system.h>
```

```
#include <flash.h>
```

```
#include "Inic_hw.h"
```

```
#include "senzit.h"
```

```
#include "serial.h"
```

```
#include "lcd.h"
```

```
// zapis dat pro rozbeh do pameti programu
```

```
#pragma          DATA          ADR_ROZBEH0,
750,745,740,735,731,727,723,719,715,711,707,703,699,695,691,68
7,683,679,675,671
```

```
#pragma          DATA          ADR_ROZBEH1,
667,664,661,658,655,652,649,646,643,640,637,634,631,628,625,62
2,619,616,613,610
```



```

#pragma          DATA          ADR_ROZBEH2,
607,604,601,598,595,592,589,586,583,581,579,577,575,573,571,56
9,567,565,563,561
#pragma          DATA          ADR_ROZBEH3,
559,557,555,553,551,549,547,545,543,541,539,537,535,533,531,52
9,527,525,523,521
#pragma          DATA          ADR_ROZBEH4,
519,517,515,513,511,509,507,505,503,501,499,497,495,493,491,48
9,487,485,483,481
#pragma          DATA          ADR_ROZBEH5,
479,477,475,473,471,469,467,465,463,462,461,460,459,458,457,45
6,455,454,453,452
#pragma          DATA          ADR_ROZBEH6,
451,450,449,448,447,446,445,444,443,442,441,440,439,438,437,43
6,435,434,433,432
#pragma          DATA          ADR_ROZBEH7,
431,430,429,428,427,426,425,424,423,422,421,420,419,418,417,41
6,415,414,413,412
#pragma          DATA          ADR_ROZBEH8,
411,410,409,408,407,406,405,404,403,402,401,400,399,398,397,39
6,395,394,393,392
#pragma          DATA          ADR_ROZBEH9,
391,390,389,388,387,386,385,384,383,382,381,380,379,378,377,37
6,375,374,373,372
#pragma          DATA          ADR_ROZBEH10,
371,370,369,368,367,366,365,364,363,362,361,360,359,358,357,35
6,355,354,353,352
#pragma          DATA          ADR_ROZBEH11,
351,350,349,348,347,346,345,344,343,342,341,340,339,338,337,33
6,335,334,333,332
#pragma          DATA          ADR_ROZBEH12,
331,330,329,328,327,326,325,324,323,322,321,320,319,318,317,31
6,315,314,313,312

#pragma DATA _CONFIG, _PWRTEN_ON & _BODEN_ON & _WDT_OFF &
_LVP_OFF & _CPD_OFF & _DEBUG_OFF & _HS_OSC & _CP_OFF

//instrukce clrfs STATUS, clrfs PCLATH, linkovani od adresy -rb
2
#pragma DATA 0,0x0183,0x018A

#pragma CLOCK_FREQ 20000000

unsigned int  cit_per_HW @CCPR1L;
struct Expozice expozice;
struct Tlacitko tlac[5];
bit sm_chod, vpred_mv=0, vzad_mv=0, zmena;

```

```

char ser_buf[47]; //pro prijaty telegram
byte vysledek;
byte c_regulace;

inline void zhasni_zarovku(void)
{ ccp2con.4=0;
  ccp2con.5=0;
  ccpr2l=0;
}

void main()
{ signed char i_exp;
  bit start;

  clear_wdt();
  trisa = 0x0F; //konfigurace port A, vstup RA0, RA1, RA2,
RA3 ostatni vystupy
  trisb = 0xF0; //konfigurace port B LCD, RB0 vystup,
ostatni zatim vstupy
  trisc = 0xC9; //konfigurace port C, vstup pro 232 a RC0,
RC3, ostatni vystupy
  trisd = 0xFF; //konfigurace port D, tlacitka, koncaky,
vse vstupy
  trise = 0x0; //konfigurace port E, vystupy pro motor
filtru

  porta = 0x10; //inicialize port A
  portb = 0x00; //inicialize port B
  portc = 0x00; //inicialize port c
  porte = 0x00; //Inicialize port E

  adcon1 = 0x8E; //zarovnaní vpravo, 1 analogovych vstupu
bez reference, pripr. na 32Fosc
  adcon0 =0b10000001; //predvolba kanalu 0, 32Fosc

  intcon=0; //globalni zakaz preruseni

  T0_INIT(T0_PRESC_256); //preddeleni 256 prirazeno T0,
citani vnitrich hodin, RB vypnuty

  //tik 51.2 us, pretečení 13.1 ms, jiz v chodu

  //inicializace T1 citac vnejsich udalosti, 1 tik = 1.6 us,
pretečení po 104,85 ms, zastaven
  T1_INIT(T1_PRESC_8|T1_OSC_DIS|T1_SYNC_DIS|T1_INT_CLK);
  //T1_PRESC_1,2,4,8, T1_OSC_EN, T1_SYNC_DIS, T1_EXT_CLK,
T1_ON

  //nastaveni a start PWM s hodnotou 0
  //pro 20 MHz asi 19.53 kHz, max. 10 bitu

```

```

    pr2=0xFF;          //PWM period = [(PR2) + 1] * 4 * TOSC *
(TMR2 prescale value)
    ccpr2l=0;          //PWM duty
cycle=(CCPR1L:CCP1CON<5:4>)*Tosc*(TMR2 prescale value)
    T2_NUL;
    T2_INIT(T2_PRESC_1|T2_POSC_10|T2_ON); //T2_PRESC_1,4,16,
T2_ON, T2_POSC_1-16
                                //pretecani po 0.512 ms
    ccp2con=CCP_PWM; //povoleni PWM, LSb=0,0

    serial_init(129); //9600 baud (20 MHz)
    PODSVIT_ZAP;
    lcd_init();
    display_smaz();
    lcd_send_cmd(LINE1);
    lcd_printf("Senzitometr V1.0");
    delay_s(1);
    //inicializace polohy voziku
    while(1)
        {
            lcd_send_cmd(LINE1);
            lcd_printf(" Inicializace ");
            lcd_send_cmd(LINE2);
            lcd_printf(" polohy voziku ");
            delay_ms(250);
            if(navr_voz_ks())
                { lcd_send_cmd(LINE2);
                  lcd_printf("probehla          spravne");
ERR_VYP; zmena=0;
                }
            else
                { lcd_send_cmd(LINE2);
                  lcd_printf(" selhala!!!          ");
ERR_ZAP; zmena=1;
                }
            delay_s(1);
            if(zmena)
                { display_smaz();

    lcd_send_cmd(LINE1);lcd_printf("Stiskni klav.ENT");
                                while(T_ENT!=STISK); //cekani na
stisk pro opakovani
                                if(T_ESC==STISK) {ERR_VYP; break;}
//po stisku pred ENT se inicializace ukonci
                                }
                                else break; //konec inicializace
                }

    //inicializace polohy filtru
    lcd_send_cmd(LINE1);
    lcd_printf(" Inicializace ");

```

```

    lcd_send_cmd(LINE2);
    lcd_printf(" polohy filtru  ");
    delay_ms(250); delay_ms(250);
    pohyb_filtr(KM_VZAD_F,NAVR_DRAHA_F);

    lcd_send_cmd(LINE1);
    lcd_printf("Pripraveno k  ");
    lcd_send_cmd(LINE2);
    lcd_printf(" cinnosti  ");
    delay_s(1);

    zobr_zaznam(0);      i_exp=0; //prvni zaznam
    zmena=0;            //neni zmena zobrazeni
    start=0;            //neni povel ke startu
    vpred_mv=0; vzad_mv=0; //motor voziku není v chodu

    while(1) //hlavni smycka programu
    {

        //vozik vpred, expozice
        if(vpred_mv==1)
            {      vzad_mv=0;      //pro jistotu
                  chod_KM();
                  if(expozice.akt_draha>=EXP_DRAHA) //konec
drahy
                        { zhasni_zarovku();
                          ccplcon=0;      //vystup do 0,
zastaveni KM
                              delay_ms(250);
                              expozice.per_rychlost=NAVR_PER;

//navratova perioda
                                  priprava_KM(KM_VZAD_E);
                                  vpred_mv=0;
                                  vzad_mv=1;
                                  continue; //zbytek cyklu se
preskoci
                        }
        if(intcon.T0IF) //pretečení po 13 ms
            { intcon.T0IF=0;
              c_regulace++;
              if(c_regulace>=D_REG) //je cas
na regulaci proudu
                        { c_regulace=0;
                          reguluj_proud();
                        }
            }
        continue; //zbytek cyklu se preskoci
    }
    //vozik vzad, zhasnuto
    if(vzad_mv==1)

```

```

        { vpred_mv=0;
          chod_KM();
          if(expozice.akt_draha>=NAVR_DRAHA) //konec
drahy, vychozi poloha
          { ccplcon=0; //vystup do 0,
zastaveni KM
          O_EN_KME=KM_DIS_E;
            vpred_mv=0;
            vzad_mv=0;
            display_smaz();
            lcd_send_cmd(LINE1);
            lcd_printf("   Expozice   ");
            lcd_send_cmd(LINE2);
            lcd_printf("   dokoncena   ");
            while(T_ENT!=STISK);
            zobr_zaznam(i_exp); start=0;
          }
          continue; //zbytek cyklu se preskoci
        }

//motor pohonu voziku stoji
vysledek=cti_zpravu_ser(); //pokud prijata, vraci
ZPR_PRIJ, jinak BEZ_ZPR
if(vysledek==ZPR_PRIJ) dekod_zpravu();
if(start) //stisknuto START
  { navr_voz_ks(); //test pripadne navrat do
vychozi polohy
    pohyb_filtr(KM_VZAD_F,NAVR_DRAHA_F);
//navrat filtru do vychozi polohy
    nastav_proud();
    if(expozice.skut_proud==PREKR_DOBA ||
expozice.skut_proud==NELZE_REG)
      { display_smaz();
        lcd_send_cmd(LINE1);
        lcd_printf("Chyba regulace: ");
        lcd_send_cmd(LINE2);

        if(expozice.skut_proud==PREKR_DOBA) lcd_printf("Neustalen
proud!");
        else lcd_printf("Nelze nasavit!
");
        while(T_ENT!=STISK);
        zobr_zaznam(i_exp); start=0;
        if(T_ESC!=STISK) continue;
//odstartovani ukonceno
      }

pohyb_filtr(KM_VPRED_F,expozice.filtr*DRAHA_POLE);
//presun filtru do pracovni polohy

expozice.per_rychlost=expozice.per_rychlost_E;

```

```

        priprava_KM(KM_VPRED_E);
        vpred_mv=1;          vzad_mv=0;          start=0;
c_regulace=0;
        continue;
    }

    if(intcon.T0IF)        //pretezeni po 13 ms
    { intcon.T0IF=0;
      tlacitka();
      //posun po polich
      if(tlac[DOWN].stisk>=K_STISK)
        { tlac[DOWN].stav=K_STISK;
          i_exp++;   zmena=1;   if(i_exp>9)
i_exp=0;
        }
      if(tlac[UP].stisk>=K_STISK)
        { tlac[UP].stav=K_STISK;
          i_exp--;   zmena=1;   if(i_exp<0)
i_exp=9;
        }
      if(tlac[ENTER].stisk>=K_STISK)
        { tlac[ENTER].stav=K_STISK;
          display_smaz();
          pis_ret_xy(0,0,"Cislo   zaznamu:");
// prvni radek
          pis_uint_xy(14,0,i_exp+1,2);
          pis_ret_xy(0,1,"T=       I=       ");
// prvni radek
          pis_uint_xy(2,1,expozice.per_rychlost/8,5);
          pis_uint_xy(11,1,expozice.zad_proud,4);
        }
      if(tlac[ESC].stisk>=K_STISK)
        { tlac[ESC].stav=K_STISK; zmena=1;
        }
      if(tlac[START].stisk>=K_STISK)
        { tlac[START].stav=K_STISK; start=1;
        }
      if(zmena)
        { display_smaz();
          pis_ret_xy(0,0,"Cislo   zaznamu:");
// prvni radek
          pis_uint_xy(14,0,i_exp+1,2);
          delay_s(1);
          zobr_zaznam(i_exp); zmena=0;
        }
    }
}

```

```

    }

}

//Inicializace EEPROM Data
#pragma          DATA          _EEPROM, "Pokusna
expo_01", 100, 0, 0x18, 0x15, 2, 0xE8, 3, 0x58, 2 //100 ms, 5400, 2,
1000, 600

    #ifndef _SENZIT_H_
#define _SENZIT_H_

//Ovladaci tlacitka
volatile bit    T_ENT          @PORTD.5;
volatile bit    T_ESC          @PORTD.4;
volatile bit    T_UP           @PORTD.7;
volatile bit    T_DOWN        @PORTD.6;
volatile bit    T_START        @PORTC.0;
#define STISK 0

volatile bit    I_SVETLO      @PORTC.3;
#define JE_SVETLO 0
volatile bit    I_K_STRT_VOZIK @PORTD.1;
volatile bit    I_K_END_VOZIK @PORTD.0;
volatile bit    I_K_END_FILTR @PORTD.2;
#define JE_POLOHA 0

//Ridici vystup pro pisklak
volatile bit    O_AKUST        @PORTA.5;
#define AKUST_ZAP O_AKUST=1;
#define AKUST_VYP O_AKUST=0;
//LED chyba
volatile bit    O_ERR          @PORTA.4;
#define ERR_ZAP O_ERR=0;
#define ERR_VYP O_ERR=1;
//podsviceni
volatile bit    O_PODSVIT     @PORTB.0;
#define PODSVIT_ZAP O_PODSVIT=1;
#define PODSVIT_VYP O_PODSVIT=0;

volatile bit    O_SM_KME      @PORTC.4; //smer chodu KM voziku

```

```

volatile bit O_EN_KME      @PORTC.5; //povoleni chodu KM voziku
volatile bit O_KME        @PORTC.2; //pulsy KM voziku

volatile bit O_SM_KMF     @PORTE.1; //smer chodu KM filtru
volatile bit O_EN_KMF     @PORTE.2; //povoleni chodu KM filtru
volatile bit O_KMF        @PORTE.0; //pulsy KM filtru

//krok motoru L>H, jednotka CCP1

//delka drahy asi 1000 kroku, pole sirka 9,5 mm, delka drahy
//asi 280 mm, sterbina sirka 5 mm
//pro pohyb voziku
#define KM_VPRED_E        1 //smer chodu KM
#define KM_VZAD_E         0 //smer chodu KM
#define KM_EN_E           1 //povoleni chodu KM
#define KM_DIS_E          0 //zakazani chodu KM
#define EXP_DRAHA         1000 //celkova draha chodu KM pri expozici
#define NAVR_DRAHA        1000 //celkova draha chodu KM pri navratu
#define CLK_L             100 //konstantni doba pulsu v L
//v nas. 1.6 us, nutno odecist z doby kroku
#define PRODL_KROK_1      200 //prodleva pred prvnim krokem,
//nas. 1.6 us
#define NAVR_PER           800 //navratova perioda, nas.
//1.6 us
#define NAVR_DRAHA_INI    1000 //celkova draha chodu KM pri
//navratu do vychozi polohy
#define NAVR_PER_INI      10 //navratova perioda v
//100 us
//pro filtr
#define KM_VPRED_F        1 //smer chodu KM
#define KM_VZAD_F         0 //smer chodu KM
#define KM_EN_F           1 //povoleni chodu
//KM
#define KM_DIS_F          0 //zakazani chodu KM
#define DRAHA_POLE        100 //draha chodu KM pro jedno pole
#define NAVR_DRAHA_F      600 //celkova draha chodu KM pri
//navratu
#define PER_CHOD           10 //perioda chodu v 100 us

#define ADR_ROZBEH0       0x1000 //adresa pole s rozbehovymi
//periodami
#define ADR_ROZBEH1       0x1014 //adresa pole s rozbehovymi
//periodami
#define ADR_ROZBEH2       0x1028 //adresa pole s rozbehovymi
//periodami
#define ADR_ROZBEH3       0x103C //adresa pole s rozbehovymi
//periodami

```



```

#define  ADR_ROZBEH4  0x1050    //adresa  pole  s  rozbehovymi
periodami
#define  ADR_ROZBEH5  0x1064    //adresa  pole  s  rozbehovymi
periodami
#define  ADR_ROZBEH6  0x1078    //adresa  pole  s  rozbehovymi
periodami
#define  ADR_ROZBEH7  0x108C    //adresa  pole  s  rozbehovymi
periodami
#define  ADR_ROZBEH8  0x10A0    //adresa  pole  s  rozbehovymi
periodami
#define  ADR_ROZBEH9  0x10B4    //adresa  pole  s  rozbehovymi
periodami
#define  ADR_ROZBEH10 0x10C8    //adresa  pole  s  rozbehovymi
periodami
#define  ADR_ROZBEH11 0x10DC    //adresa  pole  s  rozbehovymi
periodami
#define  ADR_ROZBEH12 0x10F0    //adresa  pole  s  rozbehovymi
periodami

#define  NAS_PROUD    15          //nasobici      koeficient
proudu, max. 63
#define  DEL_PROUD    12          //delici koeficient proudu
#define  D_MEZI_MER   50          //doba  v  ms  mezi  merenimi  po
inicializaci
#define  UKLID_I      50          //ceka  se  na  zmenu  po
sobe
#define  POV_ODCH     5           //povolena odchylka pro
konec inicializace
#define  D_NAST1      20          //maximalni      pocet
mereni uklidneni
#define  PREKR_DOBA   0xFFFF     //doba      pro      nastaveni
prekrocena
#define  NELZE_REG    0xFFFFE    //nelze regulovat
#define  D_MEZI_INT1  10          //doba  v  ms  mezi  regulaci  pri
inicializaci
#define  D_NAST2      100         //maximalni      pocet
regulaci pri inicializaci
#define  D_REG        4           //doba  v  nasobcich
13 ms mezi regulaci pri chodu

struct Expozice
{ unsigned int per_rychlost; //perioda kroku KM v jednotkach
T1 (1.6 us) v ustalenem stavu
  unsigned int per_rychlost_E; //perioda kroku KM v
jednotkach T1 (1.6 us) v ustalenem stavu pro expozici
  unsigned int per_KM; //aktualni perioda
  unsigned int akt_draha;
  unsigned int zad_proud; //zadany proud v jednotkach
prevodniku

```

```

        unsigned int  skut_proud;    //skutečný proud v
jednotkách převodníku
        byte filtr;
        byte ind_rampa;             //index v poli zrychlování, pro 255
se nezrychluje
};

//po potvrzení vyberu se načte z EEPROM
struct Predv_ee
{ char nazev[17]; //název 16 znak ASCII (informace)
  char exp[5];    //expozice 4 znaky ASCII, udáno v s, x,xx
nebo xx,x (informace)
  char t_barva[5]; //barva 4 znaky ASCII (informace)
  char filtr;     //filtr 1 byte (informace i pracovní)
  unsigned int per_rychlost; //per. kroku KM v jednotkách T1
(12.8 us) v ust. stavu
  unsigned int zad_proud;    //v jednotkách převodníku
  byte index;               //index v poli expozičních EEPROM
};

//vyhodnocené stavy tlačítek
#define NE_STISK 0
#define K_STISK 1
#define D_STISK 2
//meze pro vyhodnocení v násobcích testovacího cyklu
#define C_NE_STISK -5
#define C_K_STISK 5
#define C_D_STISK 10

#define ENTER 0
#define ESC 1
#define UP 2
#define DOWN 3
#define START 4

struct Tlacitko
{ signed char stisk, citac;
  byte stav,pom;
};

extern char ser_buf[47];
extern struct Tlacitko tlac[5];
extern struct Expozice expozice;
extern volatile unsigned int cit_per_HW;

void dekod_zpravu(void);

```

```

void zobr_zaznam(byte index);
void tlacitka(void);
void display_smaz(void);
byte cti_zpravu_ser(void);
void pis_byte3_xy(byte pol_x, byte pol_y, byte cislo);
void pis_znak_xy(byte pol_x, byte pol_y, char znak);
void pis_uint_xy(byte pol_x, byte pol_y, unsigned int cislo,
byte poc_dig);
byte navr_voz_ks(void);
void pohyb_filtr(bit smer, unsigned int draha);
unsigned int cti_AD(void);
void nastav_proud(void);
byte chod_KM(void);
void priprava_KM(bit smer);
void reguluj_proud(void);

#include <system.h>
#include <stdlib.h>
#include <EEPROM.h>
#include "Inic_hw.h"
#include "senzit.h"
#include "serial.h"

//pokud prijata, vraci ZPR_PRIJ, jinak BEZ_ZPR
byte cti_zpravu_ser(void)
{ byte znak, index;
  znak=serial_pool(); //dotaz na prijem
  if(znak==CH_PRIJ || znak==BEZ_ZN) return BEZ_ZPR;
  if(znak=='>') { index=0; ser_buf[0]=0;} //inicializacni znak
  else return BEZ_ZPR;
  ser_buf[index++]=znak;
  for(;index<46;index++) //nacitani dalsich znaku
    { T0_NUL; intcon.T0IF=0; //priprava casovace pro test
pretrzeni prijmu
      for(;;) //cekani na dalsi znak
        { znak=serial_pool(); //dotaz na prijem
          if(znak==CH_PRIJ) return BEZ_ZPR;
          if(znak!=BEZ_ZN) break;
          if(intcon.T0IF) return BEZ_ZPR;
        }
      //pretečení po 13 ms
    }
  ser_buf[index]=znak;
  if(znak=='<')
    { ser_buf[++index]=0; return ZPR_PRIJ;}
  //koncovy znak, na konec 0
}
return BEZ_ZPR; //prekrocena delka
}

```

```

//dekodovani zpravy o expozici, ulozi ji do EEPROM
void dekod_expozice(void)
{ char zaznam[26],pom[6];
  byte i,j,adr,filtr_cislo;
  unsigned int pom_int;
  //urceni adresy v EEPROM
  adr=ser_buf[4]-'0';
  adr+=(ser_buf[3]=='1')?10:0;
  adr--; adr*=25; //delka 25 B

  for(i=0,j=6;i<16;i++,j++) zaznam[i]=ser_buf[j]; //nazev
  //expozice
  pom[0]=ser_buf[23]; pom[1]=ser_buf[24]; pom[2]=ser_buf[25];
  pom[3]=ser_buf[26]; pom[4]=ser_buf[27]; pom[5]=0;
  pom_int=atoui_dec(pom); //retezec expozice pro usporu mista
  prevedeny na cislo
  LOBYTE_RET(zaznam+16,pom_int); //cil+index, zdroj
  HIBYTE_RET(zaznam+17,pom_int); //cil+index, zdroj
  //barevna teplota
  pom[0]=ser_buf[29]; pom[1]=ser_buf[30]; pom[2]=ser_buf[31];
  pom[3]=ser_buf[32]; pom[4]=0;
  pom_int=atoui_dec(pom); //retezec teploty pro usporu mista
  prevedeny na cislo
  LOBYTE_RET(zaznam+18,pom_int); //cil+index, zdroj
  HIBYTE_RET(zaznam+19,pom_int); //cil+index, zdroj
  //cislo filtru
  zaznam[20]=ser_buf[34]-'0';
  //doba periody
  pom[0]=ser_buf[36]; pom[1]=ser_buf[37]; pom[2]=ser_buf[38];
  pom[3]=ser_buf[39]; pom[4]=0;
  pom_int=atoui_dec(pom); //retezec teploty pro usporu mista
  prevedeny na cislo
  LOBYTE_RET(zaznam+21,pom_int); //cil+index, zdroj
  HIBYTE_RET(zaznam+22,pom_int); //cil+index, zdroj
  //proud zarovky
  pom[0]=ser_buf[41]; pom[1]=ser_buf[42]; pom[2]=ser_buf[43];
  pom[3]=ser_buf[44]; pom[4]=0;
  pom_int=atoui_dec(pom); //retezec teploty pro usporu mista
  prevedeny na cislo
  LOBYTE_RET(zaznam+23,pom_int); //cil+index, zdroj
  HIBYTE_RET(zaznam+24,pom_int); //cil+index, zdroj

  for(i=0;i<25;i++) //ukladani cele zpravy
    { eeprom_write(adr++,zaznam[i]); //start
      while(eecon1.WR); //cekani na dokonceni
    }
}

void posli_odpoved(void)

```

```

{
    pis_znak_xy(3,1,'Q');
    serial_printf(ser_buf);
}

void dekod_zpravu(void)
{
    switch(ser_buf[1])
    { case 'Z': //zapis
      switch(ser_buf[2])
      { case 'E': //expozice
        pis_znak_xy(0,1,'E');
        dekod_expozice();
        pis_znak_xy(1,1,'D');
        pis_znak_xy(2,1,ser_buf[45]);
        if(ser_buf[45]=='<')
        {ser_buf[1]='P'; posli_odpoved();}
        break;

        case 'F': //filtr
        case 'I': //proud zarovky
        case 'P': //poloha voziku

      }
    }
}

/*****
* Blocking serial interface code for the hardware UART
* of PIC16-range microcontrollers from Microchip
*
* (c) Lieven Hollevoet
* most recent version on http://boostc.lika.be
* Based on code from Yann Hendrikx.
*****/

#include "serial.h"

char buffer[SERIAL_BUFSIZE];
char buffer_pointer;

////////////////////////////////////
// Hardware initialisation
// input argument: the baudrate scaler
////////////////////////////////////
//inicializace RS232, povoleno vysilani, prijem
void serial_init(char brg)
{
    // Enable USART function on PORTC (see PIC18F4620
datasheet page 201)
    trisc.7      = 1;
    trisc.6      = 1;

```

```

// Transmit control register
txsta.TX9      = 0; //TX9 = 0 8 bitu
txsta.TXEN     = 1; //TXEN = 1 povoleni vysilani
txsta.SYNC     = 0; //SYNC = 0 povoleni asynchr. provozu
txsta.BRGH     = 1; //1 vysoka rychlost
// Receive control register
rcsta.SPEN     = 1; //1 enable serial port
rcsta.RX9      = 0; //RX9=0 8 bitu prijem
rcsta.CREN     = 1; //CREN=1 povoleni prijimani
rcsta.ADDEN    = 0; //ADDEN=0 bit 9 je parita
// Baudcon register, if supported by the target
architecture
#ifdef baudcon
    baudcon.BRG16 = 0;
    baudcon.WUE   = 0;
    baudcon.ABDEN = 0;
#endif
    // Baud rate setting
    spbrg         = brg;
}

////////////////////////////////////
// Print a single character to the serial port
// This funtion is skipped in debug mode since it will
// block the debugger (the function waits for a hardware
// signal)
////////////////////////////////////
void serial_printf(char value)
{
#ifdef DBG_COMPILE
    // Skip when debugging
#warning "Debug build, serial_printf will return immediately"
    return;
#endif
    //while((txsta & 1 << TRMT) == 0); // 0 vysilani
nedokonceno
    while(txsta.TRMT==0); // 0 vysilani nedokonceno
    txreg = value;
}

////////////////////////////////////
// Prints a line to the serial port. Ends with a linefeed.
// The function parameter is stored in rom.
////////////////////////////////////
void serial_printf(const char* text)
{
    char i = 0;
    while(text[i] != 0)
        serial_printf(text[i++]);
}

```

```

}

////////////////////////////////////
// Print a decimal value to the serial port
////////////////////////////////////
void serial_print_dec(char number)
{
    if(number > 99) {
        serial_printf(((number / 100) % 10) + '0');
    }
    if(number > 9) {
        serial_printf(((number / 10) % 10) + '0');
    }
    serial_printf((number % 10) + '0');
}

////////////////////////////////////
// Print a decimal value to the serial port
// The number takes multiple characters. If a character
// position is not used, it is filled with 0.
// Positions should be 1, 2 or 3.
////////////////////////////////////
void serial_print_dec(char number, char positions)
{
    if(number > 99) {
        serial_printf(((number / 100) % 10) + '0');
    } else {
        if (positions > 2){
            serial_printf('0');
        }
    }
    if(number > 9) {
        serial_printf(((number / 10) % 10) + '0');
    } else {
        if (positions > 1){
            serial_printf('0');
        }
    }
    serial_printf((number % 10) + '0');
}

////////////////////////////////////
// Print a decimal value to the serial port
////////////////////////////////////
void serial_print_dec(short number)
{
    if(number > 9999) {
        serial_printf(((number / 10000) % 10) + '0');
    }
    if(number > 999) {

```

```

        serial_printf(((number / 1000) % 10) + '0');
    }
    if(number > 99) {
        serial_printf(((number / 100) % 10) + '0');
    }
    if(number > 9) {
        serial_printf(((number / 10) % 10) + '0');
    }
    serial_printf((number % 10) + '0');
}

/////////////////////////////////////////////////////////////////
// Print a hex value to the serial port
/////////////////////////////////////////////////////////////////
void serial_print_hex(char number)
{
    char hexChar;
    char i;
    for(i = 0; i < 2; i++)
    {
        if(i == 0)
            hexChar = number >> 4;
        else
            hexChar = number & 0x0F;
        if(hexChar < 10)
            hexChar = hexChar + '0';
        else
            hexChar = hexChar + ('A' - 10);
        serial_printf(hexChar);
    }
}

/////////////////////////////////////////////////////////////////
// Print a 16-bit hex value to the serial port
/////////////////////////////////////////////////////////////////
void serial_print_hex(short value){

    char value1 = (char)((value >> 8) & 0x00FF);

    serial_print_hex(value1);

    char value0 = (char)(value & 0x00FF);

    serial_print_hex(value0);

    return;

}

/////////////////////////////////////////////////////////////////
// Print a binary value to the serial port
/////////////////////////////////////////////////////////////////

```



```

void serial_print_bin(char number)
{
    char i;
    for(i = 0; i < 8; i++)
    {
        if (number & 10000000b){
            serial_printf('1');
        } else {
            serial_printf('0');
        }
        number <<= 1;
    }
}

////////////////////////////////////
// Wait for a character to read it from the serial port
// This function is skipped when running in debug mode
// since else it blocks the simulator (waits for a
// hardware signal from the UART)
////////////////////////////////////
char serial_getch(){

#ifdef DBG_COMPILE
#warning "Debug build, serial_getch will return 0x00"
    return 0x00;
#endif

    // Check for overrun error. Reset the
    // UART if any.
    if (rcsta.OERR || rcsta.FERR)
    { //serial_printf("OERR");
        rcsta.CREN = 0;
        rcsta.CREN = 1;
    }
    while (!serial_peek());
    return rcreg;
}

//dotaze se na prijaty znak, pokud je, vrati, jinak BEZ_ZN
pripadne CH_PRIJ
char serial_pool(void)
{
    if (rcsta.OERR || rcsta.FERR)
    { //serial_printf("OERR");
        asm
        { movf _rcreg,W ;vyprazdneni buferu, shozeni FERR
        }
        rcsta.CREN = 0;
        rcsta.CREN = 1;
        return CH_PRIJ;
    }
    if(pir1.RCIF) return rcreg; else return BEZ_ZN;
}

```

```

// Read a decimal value from the serial port
char serial_get_decimal(){
    char value = 0;
    char input = 0;

    input = serial_getch();

    while (input != 0x0D){
        serial_printf(input);
        value *= 10;
        value += (input-0x30);
        input = serial_getch();
    }
    serial_print_lf();
    return value;
}

void serial_safe_init(){
    buffer_pointer = 0;
}

char serial_safe_putbyte(char data){
    buffer[buffer_pointer] = data;
    buffer_pointer++;

    if (buffer_pointer == SERIAL_BUFSIZE){
        return 1;
    }

    return 0;
}

char serial_safe_send(){
    char loper;
    char cur;
    char checksum;

    // Safety check
    if (buffer_pointer == 0){
        return 1;
    }

    checksum = 0;

    // Send header
    serial_printf(SERIAL_PRE);
    serial_printf(SERIAL_PRE);

    // Transmit buffer
    for (loper = 0; loper < buffer_pointer; loper++){

```

```

        // Byte to send
        cur = buffer[loper];

        // Check if we need to send the ESCAPE byte
        if ((cur == SERIAL_PRE) || (cur == SERIAL_EOP) ||
(cur == SERIAL_ESC)){
            serial_printf(SERIAL_ESC);
        }

        // Send data
        serial_printf(cur);

        // Update checksum
        checksum += cur;
    }

    // Send checksum and EOP
    if ((checksum == SERIAL_PRE) || (checksum == SERIAL_EOP)
|| (checksum == SERIAL_ESC)){
        serial_printf(SERIAL_ESC);
    }

    serial_printf(~checksum + 1);
    serial_printf(SERIAL_EOP);

    buffer_pointer = 0;
    return 0;
}

signed char serial_safe_receive(){

    buffer_pointer = 0;
    char cur;
    char checksum = 0;
    char loper = 0;
    bit got_escape;

    // Expecting a packet that has the structure
    // <PRE><PRE>[<DATA1><DATA2> ... ]<CRC><EOP>

    while (loper < 2){
        cur = serial_getch();
        if (cur == SERIAL_PRE){
            loper ++;
        } else {
            loper = 0;
        }
    }

    //serial_printf("Passed header...");
    //serial_print_lf();

```

```

// Receive data
do {
    got_escape = 0;

    cur = serial_getch();
    if (cur == SERIAL_ESC){
        cur = serial_getch();
        got_escape = 1;
    }

    buffer[buffer_pointer] = cur;
    buffer_pointer++;
    if (buffer_pointer == SERIAL_BUFSIZE){
        return -1;
    }
} while (cur != SERIAL_EOP || got_escape);

//serial_printf("received: ");
//for (loper = 0; loper < buffer_pointer; loper++){
//    serial_print_hex(buffer[loper]);
//}

// TODO: implement checksum

// Correct the pointer so we can reuse a part of the
packet for the response
buffer_pointer -= 2;
return buffer_pointer;
}

char serial_safe_getbyte(char index){
    return buffer[index];
}
/*****
* Blocking serial interface code for the hardware UART
* of PIC16-range microcontrollers from Microchip
*
* (c) Lieven Hollevoet
* most recent version on http://boostc.lika.be
* Based on code from Yann Hendrikx.
*****/

#ifndef _SERIAL_H_
#define _SERIAL_H_

#include <system.h>

#define SERIAL_PRE        0x0F
#define SERIAL_EOP        0x04
#define SERIAL_ESC        0x05

```

```

#define SERIAL_BUFSIZE 32

#define CH_PRIJ      0xFE //detekovana chyba prijmu
#define BEZ_ZN      0xFF //neni znak
#define ZPR_PRIJ    1     //je zprava
#define BEZ_ZPR     0     //neni zprava

// Function declarations
void serial_init(char brg);

void serial_printf(char value);
void serial_printf(const char* text);
void serial_print_lf();
void serial_print_hex(char number);
void serial_print_hex(short number);
void serial_print_dec(char number);
void serial_print_dec(short number);
void serial_print_dec(char number, char positions);
void serial_print_bin(char number);

char serial_getch();
char serial_get_decimal();
char serial_pool();

// Function declarations for safe serial transmission
void serial_safe_init();
char serial_safe_putbyte(char data);
char serial_safe_send();
signed char serial_safe_receive();
char serial_safe_getbyte(char index);

// Macros for simple functions (to reduce stack usage)
#define serial_peek()      (pir1 & 1 << RCIF)
#define serial_print_lf()  serial_printf(0x0a);
serial_printf(0x0d)

#endif // _SERIAL_H_

```