

Model reálného zařízení realizovaný na bázi mikropočítače ATmega16

Model of real establishment realized on base microcomputer
ATmega16

Bc. Martin Krajča

Diplomová práce
2007



Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
Ústav automatizace a řídicí techniky
akademický rok: 2006/2007

ZADÁNÍ DIPLOMOVÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení: **Bc. Martin KRAJČA**
Studijní program: **N 3902 Inženýrská informatika**
Studijní obor: **Automatické řízení a informatika**

Téma práce: **Model reálného zařízení realizovaný na bázi mikropočítače ATMEGA16**

Zásady pro vypracování:

- 1) Provedte návrh mikropočítačové jednotky na bázi procesoru ATMEGA16 s podpůrnými obvody umožňující měření analogových i binárních veličin a ovládající binárních veličin.
- 2) Všechny vstupy a výstupy vyvedte:
 - a) na konektor, na který bude možno připojit panel s LED diodami, tlačítky a potenciometry, které budou simulovat reálné zařízení.
 - b) na konektor, který bude připojen do PLC a bude předávat simulované vstupy do počítače a naopak.
- 3) Realizujte panel připojitelný k mikropočítači, který bude simulovat úlohu zadanou vedoucím práce.
- 4) Propojte Vámi vytvořený mikropočítač simulující zadanou úlohu s PLC.
- 5) Vytvořte program pro PLC Teco i Saia, který bude řídit simulované zařízení.
- 6) Zpracujte podrobnou dokumentaci k Vámi vytvořenému mikropočítači.
- 7) Popište zásady pro tvorbu dalších simulačních panelů připojitelných k Vámi řešenému mikropočítači.

Rozsah práce:

Rozsah příloh:

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

Seznam odborné literatury:

1. Martinásková, M., Šmejkal, L.: Řízení programovatelnými automaty, Vydavatelství ČVUT, Praha, 1998.
2. Martinásková, M., Šmejkal, L.: Řízení programovatelnými automaty II, Vydavatelství ČVUT, Praha, 2000.
3. Šmejkal, L., Martinásková, M.: PLC a automatizace, Nakladatelství BEN – technická literatura, Praha, 1999.
4. Firemní literatura k programovatelnému automatu Teco 600.
5. Firemní literatura k produktům firmy Saia Burgess.
6. Matoušek D.: Práce s mikrokontroléry ATMEL AVR – ATmega16, Nakladatelství BEN – technická literatura, Praha, 2006.
7. Váňa V.: Atmel AVR programování v jazyce C, Nakladatelství BEN – technická literatura, Praha, 2003.
8. Matoušek D.: Práce s mikrokontroléry Atmel AVR – uP a praxe, Nakladatelství BEN – technická literatura, Praha, 2003.

Vedoucí diplomové práce:

Ing. Tomáš Sysala, Ph.D.

Ústav automatizace a řídicí techniky

Datum zadání diplomové práce:

13. února 2007

Termín odevzdání diplomové práce:

24. května 2007

Ve Zlíně dne 13. února 2007

prof. Ing. Vladimír Vašek, CSc.
děkan



prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Diplomová práce se zabývá návrhem a realizací modelu reálného zařízení na bázi mikropočítače ATmega16. Model umožňuje simulovat naprogramované reálné zařízení, jenž následně můžeme řídit prostřednictvím programovatelných automatů SAIA a TECO. Model reálného zařízení je navržen jako univerzální snadno rozšiřitelný modul. Diplomová práce obsahuje teoretickou část, která popisuje použité hardwarové a softwarové vybavení a praktickou část, která popisuje návrh desek plošných spojů a návrh programů pro ATmega16, PLC SAIA a PLC TECO.

Klíčová slova:

Mikropočítač, ATmega16, PLC, SAIA, PCD2, TECOMAT, TECO606, Eagle, PonyProg, Codevision, Mosaic, PG5 Controls Suite, autokláv, AD převodník, deska plošného spoje, DPS

ABSTRACT

Graduation theses dealings with design and realization model of real establishment on base microcomputer ATmega16. Model makes it possible to simulate programmed real establishment, which subsequently we can control through programmable logical controller SAIA and TECO. Model of real establishment is designed like universal easily extendable unit. Graduation theses includes theoretic part, which describes used hardware and software equipment and practical part, which describes design of boards printed circuit and design programs for ATmega16, PLC SAIA and PLC TECO.

Keywords:

microcomputer, ATmega16, PLC, SAIA, PCD2, TECOMAT, TECO606, Eagle, PonyProg, Codevision, Mosaic, PG5 Controls Suite, autoclave, pressure vessel, AD converter, printed circuit boards, PCB

Děkuji vedoucímu diplomové práce panu Ing. Tomášovi Sysalovi, Ph.D. za odborné vedení, rady a připomínky, které mi při řešení mé práce poskytoval.

Dále děkuji panu Mgr. Milanovi Adámkovi Ph.D. za pomoc při realizaci plošných spojů.

Panu Josefovi Kubáčkovi za poskytnutí prostor a pomůcek k realizaci hardwarové části diplomové práce.

V neposlední řadě děkuji svým rodičům za psychickou podporu.

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

Ve Zlíně

.....
Podpis diplomanta

OBSAH

ÚVOD	8
I TEORETICKÁ ČÁST	9
1 HARDWAROVÉ VYBAVENÍ	10
1.1 PLC SAIA.....	10
1.2 PLC TECOMAT	11
1.2.1 Popis PLC Tecomat.....	11
1.2.2 Charakteristika PLC Tecomat	11
1.2.3 Systémové služby	12
1.3 MIKROPOČÍTAČ ATMEGA16.....	14
1.3.1 Parametry mikropočítače ATmega16.....	14
2 PROGRAMOVÉ VYBAVENÍ	19
2.1 CODEVISION AVR	19
2.2 PONYPROG2000.....	20
2.3 PROGRAM MOSAIC	20
2.4 PROGRAMOVACÍ KOMPLET - SAIA® PG5 CONTROLS SUITE	23
2.5 PROGRAM EAGLE	24
II PRAKTICKÁ ČÁST	26
3 NÁVRH MODELU REÁLNÉHO ZAŘÍZENÍ	27
3.1 VÝBĚR A NÁVRH MODELU	27
3.1.1 Princip autoklávu LAGARDE	28
3.1.2 Návrh modelu autoklávu	29
3.1.3 Seznam všech vstupů a výstupů	29
3.1.4 Rozvržení vstupně výstupních portů mikropočítače	30
3.2 NÁVRH DESKY PLOŠNÉHO SPOJE.....	31
3.2.1 Celkový náhled na model	31
3.2.2 Testování zapojení	32
3.2.3 Základní deska s mikropočítačem	32
3.2.4 Komunikační modul.....	35
3.2.5 Vizualizační modul	39
3.2.6 Programovací modul	41
3.3 NÁVRH PROGRAMU REÁLNÉHO MODELU AUTOKLÁVU	42
3.3.1 Stanovení podmínek a popis chování modelu.....	42
3.3.2 Program modelu autoklávu	44
3.4 NÁVRH ŘÍDÍCÍHO PROGRAMU PLC	45
3.4.1 Zadání pro PLC TECOMAT 606.....	45
3.4.2 Program určený pro PLC TECOMAT	47
3.4.3 Zadání pro PLC SAIA PCD2	47
3.4.4 Program určený pro PLC SAIA.....	49

4	ROZŠÍŘITELNOST SIMULAČNÍHO MODULU REÁLNÉHO ZARÍZENÍ	52
4.1	PROGRAMOVÁNÍ ATMEGA16.....	52
4.1.1	Průvodce nastavení CodeVisionAVR	52
4.1.2	Průvodce nastavení PonyProg2000	54
4.2	NÁVRH PLOŠNÉHO SPOJE	55
	ZÁVĚR	57
	ZÁVĚR V ANGLIČTINĚ	58
	SEZNAM POUŽITÉ LITERATURY.....	59
	SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK	60
	SEZNAM OBRÁZKŮ	61
	SEZNAM TABULEK.....	63
	SEZNAM PŘÍLOH.....	64

ÚVOD

Programovatelné automaty jsou páteří průmyslové automatizace již takřka dvě desetiletí. Původně byly určeny k programovému řešení jednoduchých logických obvodů, dnes je jejich použití mnohem širší. Umožňují provádět kromě základních logických funkcí i matematické operace, přesuny bloků dat, zpracovávat spojité signály... Často jsou součástí většího řídicího celku, tzv. distribuovaného řídicího systému, jehož jednotlivé součásti jsou propojeny soustavou sítí.

Použití PLC je velmi široké, od jednoduchých zařízení realizujících logické funkce např. při řízení kotelen a klimatizačních jednotek, přes aplikace ve sklářském a potravinářském průmyslu, až po PLC zabudované jako subsystém v CNC systémech pro řízení obráběcích strojů.

S rozrůstajícím se odvětvím programovatelných automatů rostou i nároky na znalosti studentů. Pro výuku studentů je k dispozici simulované PLC vytvořené vývojovým programem Mosaic (není nutné reálné PLC). Simulované PLC se ale nedá v mnoha ohledech vůbec srovnávat s reálným PLC připojeným k reálné soustavě. Naopak u řízení reálné soustavy prostřednictvím PLC by mohlo dojít vlivem nekorektně vytvořeného regulačního softwaru v nejhorším případě i k jejímu zničení. Proto bylo nutné vyvinout simulační model na kterém by se mohli studenti učit programovat PLC a zdokonalovat své znalosti v tomto odvětví. Simulační model, který by byl schopný co nejvíce se přiblížit chování reálných soustav.

Cílem diplomové práce tedy je navrhnout a zrealizovat simulační model, který je připojitelný k PLC a na kterém by šla tudíž provádět simulace vytvořených programů. Model by měl být univerzální, snadno a libovolně rozšiřitelný o různé moduly reálných zařízení.

V diplomové práci bude podrobně popsán postup realizace modelu na bázi mikro počítače ATmega16 včetně návrhu desek plošných spojů a návrhu programů pro mikro počítač, PLC SAIA a PLC TECO.

I. TEORETICKÁ ČÁST

1 HARDWAROVÉ VYBAVENÍ

V kapitole HARDWAROVÉ VYBAVENÍ budou obecně popsány dva programovatelné automaty (PLC) a to PLC SAIA a PLC TECOMAT. Poté bude podrobně popsán mikropočítač ATmega16 včetně důvodu volby právě tohoto mikropočítače.

1.1 PLC SAIA

Stavebnicový řídicí systém SAIA je určen pro oblast automatizovaných systémů. Ať už se jedná o spojitě nebo diskrétní řízení. Dá se ekonomicky výhodně nasazovat od malých aplikací jako je např. řízení obráběcích strojů, výrobních linek přes řízení technologie budov a řízení výměnných stanic až po řízení celých závodů např. pivovarů nebo energetických bloků.

Mezi největší přednosti řídicího systému SAIA patří jeho komunikační schopnosti. Není problém zapojit řídicí podstanice SAIA do lokální technologické sítě, nebo do podnikové informační sítě, popřípadě přenášet data prostřednictvím globálních informačních médií (internet) či prověřit funkčnost aplikace například pomocí SMS správ v síti GSM.

K dispozici je široký sortiment programovatelných automatů Saia®PCD a to od kompaktního automatu Saia®PCS1, přes nejmenší modulární automat řady Saia®PCD1 až po nejpoužívanější klasickou řadu s plochou konstrukcí Saia®PCD2.

V našem případě se tedy jedná o kompaktní automat Saia®PCD2.

Automat PCD2 je tvořen harmonickou kombinací operačního systému, CPU, vstupních/výstupních modulů (I/O moduly), síťových karet a programovacích nástrojů.

Hlavní částí systému je základní procesorová jednotka (CPU). K dispozici jsou 4 standardní řešení poskytující široké spektrum výkonů a funkcí (modely M110, M120, M150, M170). Do každé základní jednotky PCD2 je možné vložit až 8 libovolných I/O modulů. V případě potřeby jsou k dispozici taktéž rozšiřovací jednotky pro dalších 4 anebo 8 I/O modulů. [10].



Obr. 1 - PLC SAIA PCD2

1.2 PLC Tecomat

1.2.1 Popis PLC Tecomat

Programovatelné automaty řady Tecomat jsou určeny pro řízení technologií v nejrůznějších oblastech průmyslu, potravinářství, dopravě, energetice, výrobě stavebních hmot, apod. Jsou vyráběny ve dvou odlišných provedeních, a to v kompaktním provedení (TC400, TC500, TC600) a v modulárním provedení (NS950). Kompaktní provedení PLC se vyznačuje snadnou montáží a příznivou cenou a je určeno pro aplikace menšího rozsahu podle počtu vstupů a výstupů. Modulární provedení PLC (provedení, u něhož je možné sestavit z daných periferních jednotek nejrůznější konfigurace samostatných modulů) má k dispozici několik typů napájecích zdrojů, různé typy centrálních řídicích jednotek (CPU), různé typy instalačních rámců a více než 80 typů periferních jednotek.

1.2.2 Charakteristika PLC Tecomat

PLC Tecomat vykonávají uživatelský program cyklicky za použití více smyčkového řízení. Operace se provádějí nad zásobníkem (8 úrovní). K ukládání uživatelských dat slouží zápisníková paměť, která obsahuje také systémové registry a obrazy vstupů a

výstupů. PLC umožňuje blokování výstupu a to jednak služebním vstupem, příkazem po sériovém kanálu anebo automaticky po závažné chybě. Hardware zajišťuje kontrolu procesoru (watchdog), hlídání napájecího napětí (power fail), ochranu dat při výpadku napájení, zabezpečuje sériovou komunikaci a přenos dat po I/O sběrnici. Software kontroluje platnost uživatelského programu, hlídá dobu cyklu a správnost uživatelského programu. Pro komunikaci slouží sériové rozhraní které umožňuje programování, vizualizaci a výměnu dat mezi PLC.

1.2.3 Systémové služby

PLC Tecomat mají vestavěné různé systémové služby, které zvyšují komfort programování, obsluhy a zkracují dobu nutnou pro nasazení systému v průmyslových aplikacích. Výhodným prostředkem usnadňujícím programování je soubor systémových registrů S v zápisníkové paměti, ve kterých jsou realizovány následující služby:

- příznaky výsledků aritmetických operací
- příznaky výsledků logických operací
- příznaky stavu systému
- doba minulého cyklu
- čítač cyklů
- čas systému (hodiny, minuty, sekundy, desítky milisekund)
- datum (rok, měsíc, den, den v týdnu)
- časové střídače
- čítače po 100 ms, 1 s a 10 s
- náběžné a sestupné hrany časových střídačů
- řídicí masky uživatelských procesů
- interní kód chyby

Důležitou službou je ochrana dat a programu při vypnutí napájení PLC automatu. Zálohované registry R (remanentní zóna) mají hodnotu dosaženou po poslední ukončené smyčce uživatelského programu (zásada konzistence dat).

Lokalizace závad je velmi účinnou službou. Závady jsou rozděleny do dvou kategorií podle závažnosti. Kritické závady způsobují řízené zastavení činnosti PLC. Ostatní závady dávají uživatelskému programu na vědomí svůj výskyt v registru S, čímž umožní adekvátní reakci uživatelského programu na vzniklé závady.

Systemy Tecomat umožňují svou strukturou uživatelského programu více smyčkové řízení. Uživatelský program se skládá z uživatelských procesů, jejichž vykonání je v dané smyčce programu podmíněno určitou logikou. [9].



Obr. 2 - PLC TC600

1.3 Mikropočítač ATmega16

Při výběru mikropočítače jsem vycházel z několika požadavků na jeho vlastnosti:

- možnost programování mikropočítače ve vyšším programovacím jazyce (C++)
- aby mikropočítač obsahoval AD převodník
- aby měl mikropočítač k dispozici alespoň 30 binárních vstupů/výstupů
- cenová dostupnost samotného mikropočítače a programátoru

Všechny tyto požadavky splňuje mikropočítač ATmega16. Mikropočítač lze programovat například prostřednictvím programu AVRstudio nebo CodevisionAVR. Oba tyto programy využívají programovací jazyk C++. Mikropočítač ATmega16 má k dispozici 10bitový AD převodník. K dispozici je i 32 binárních vstupů/výstupů. Mikropočítač lze programovat prostřednictvím jednoduchého sériového programátoru STK200, jehož pořizovací cena nepřesáhla 150kč. Cena ATmega16 nepřesáhne 70kč.

ATmega16 tedy splňuje všechny požadavky pro volbu vhodného mikropočítače.

1.3.1 Parametry mikropočítače ATmega16

ATmega16 je nízko příkonový CMOS 8-mi bitový mikropočítač, založený na rozšířené AVR RISC architektuře. AVR jádro obsahuje bohatý instrukční soubor s 32 pracovními registry. Všechny 32 registrů je přímo připojeno na ALU (Arithmetic Logic Unit).

Rozšířená RISC architektura :

- o 131 výkonných instrukcí - většina instrukcí je vykonána během jednoho hodinového cyklu.
- o Výpočetní výkon dosahuje hodnoty až 16 MIPS při hodinovém kmitočtu 16 MHz.
- o Čip obsahuje funkce násobení, které jsou vykonávány během dvou hodinových cyklů.

Paměť typu FLASH, EEPROM a SRAM :

- o Obsahuje 16kB paměti FLASH, která je typu ISSP (In-System Self-Programmable). Životnost paměti je udávána na 10,000 zapisovacích/přepisovacích cyklů

- Na čipu je vyhrazena část programové paměti pro zaváděcí program, tzv. Boot Code Section, kterou je možno ochránit zamykacím bitem (Lock Bits).
- Paměť typu EEPROM má velikost 512 bajtů.
- Interní paměť typu SRAM má velikost 1k bajt.
- Obvod obsahuje programovatelné propojky (Programming Lock) pro ochránění softwarového obsahu obvodu.

Rozhraní JTAG :

- Je kompatibilní se standardem IEEE std. 1149.1.
- Široká podpora ladění programového vybavení na čipu obvodu.
- Pomocí tohoto rozhraní JTAG je možno programovat paměti FLASH, EEPROM, propojky (Fuses) a zamykací (Lock) bity obvodu.

Periferní obvody :

- Dva 8mi bitové časovače/čítače s oddělenou před děličkou a komparačním módem.
- Jeden 16ti bitový časovač/čítač s oddělenou před děličkou, komparačním módem a zachytávacím módem.
- RTC (Real Time Counter) se samostatným oscilátorem.
- Čtyři PWM kanály.
- Osmi kanálový, 10ti bitový analogově digitální převodník (ADC).
- Programovatelný sériový kanál USART.
- Sériové rozhraní SPI s podporou nadřizového/podřizového (Master/Slave) obvodu.
- Univerzální sériové rozhraní s detektorem Start podmínky.
- Programovatelný časovač pro hlídací obvod (Watchdog) se samostatným oscilátorem na čipu.
- Na čipu obvodu je analogový komparátor.

Speciální vlastnosti mikropočítače :

- Obvod je vybaven programovatelným detektorem poklesu napájecího napětí BOD (Brown-out Detection) a funkcí resetu při zapnutí napájecího napětí (Power-on Reset).
- Obvod obsahuje interní kalibrovaný oscilátor.
- Je možno vyvolat přerušení na základě interního nebo externího podnětu.
- Obvod disponuje pěti módy se sníženým odběrem : Idle, redukce šumu pro ADC (ADC Noise Reduction), Power-save, Power-down a Standby.

Vstupně / výstupní obvody a pouzdro :

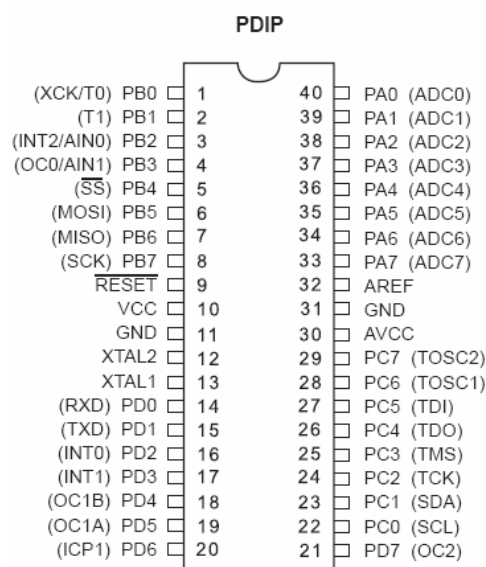
- Mikro počítač obsahuje 32 vstupně/výstupních vývodů.
- Obvod je dodáván v pouzdře se 40 vývody PDIP

Rozsah rychlostí :

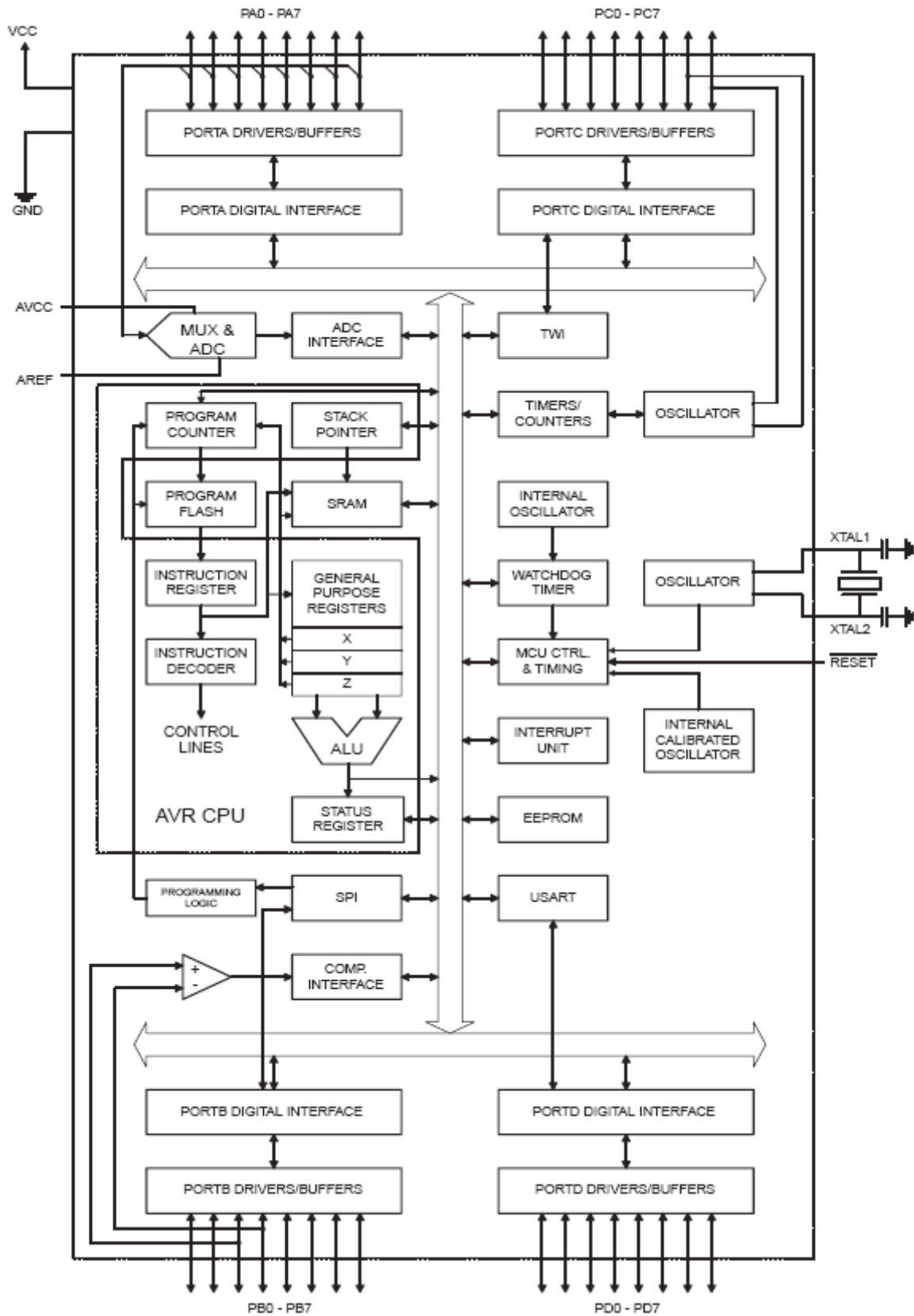
- Obvod ATmega16 pracuje v rozsahu hodinové frekvence 0 až 8 Mhz, při napájecím napětí od 2,7V až do 5,5V, případně v rozsahu hodinové frekvence 0 až 16 Mhz, při napájecím napětí od 4,5V až do 5,5V.

Rozsah teploty :

- Průmyslové provedení mikro počítače je schopno pracovat v rozsahu teplot od -40°C až do +85°C. [12]



Obr. 3 - Schéma mikro počítače ATmega16



Obr. 4 - Blokové schéma mikropočítače ATmega16

Popis pinů:

GND – uzemnění

VCC – konektor pro připojení napájecího napětí

PORTA (PA0-PA7) – může sloužit jako A/D převodník, nebo jako 8bitový vstupně/výstupní port (pokud není A/D převodník používán)

PORTB (PB0-PB7) – může sloužit jako 8bitový vstupně/výstupní port, používá se pro programování prostřednictvím programátoru STK200 (porty PB5(MOSI), PB6(MISO), PB7(SCK))

PORTC (PC0-PC7) – může sloužit jako 8bitový vstupně/výstupní port, JTAG programátor se připojuje k pinům PC5(TDI), PC3(TMS) a PC2 (TCK)

PORTD (PD0-PD7) – může sloužit jako 8bitový vstupně/výstupní port

RESET – vstup pro reset (schéma zapojení je uvedeno v kapitole základní deska s mikropočítačem)

XTAL1 – vstup do invertujícího oscilátoru a vstup do operačního obvodu vnitřních hodin

XTAL2 – výstup z invertujícího oscilátoru

AVCC – je napájecí pin pro PORTA a A/D převodník (zpravidla se připojuje na VCC)

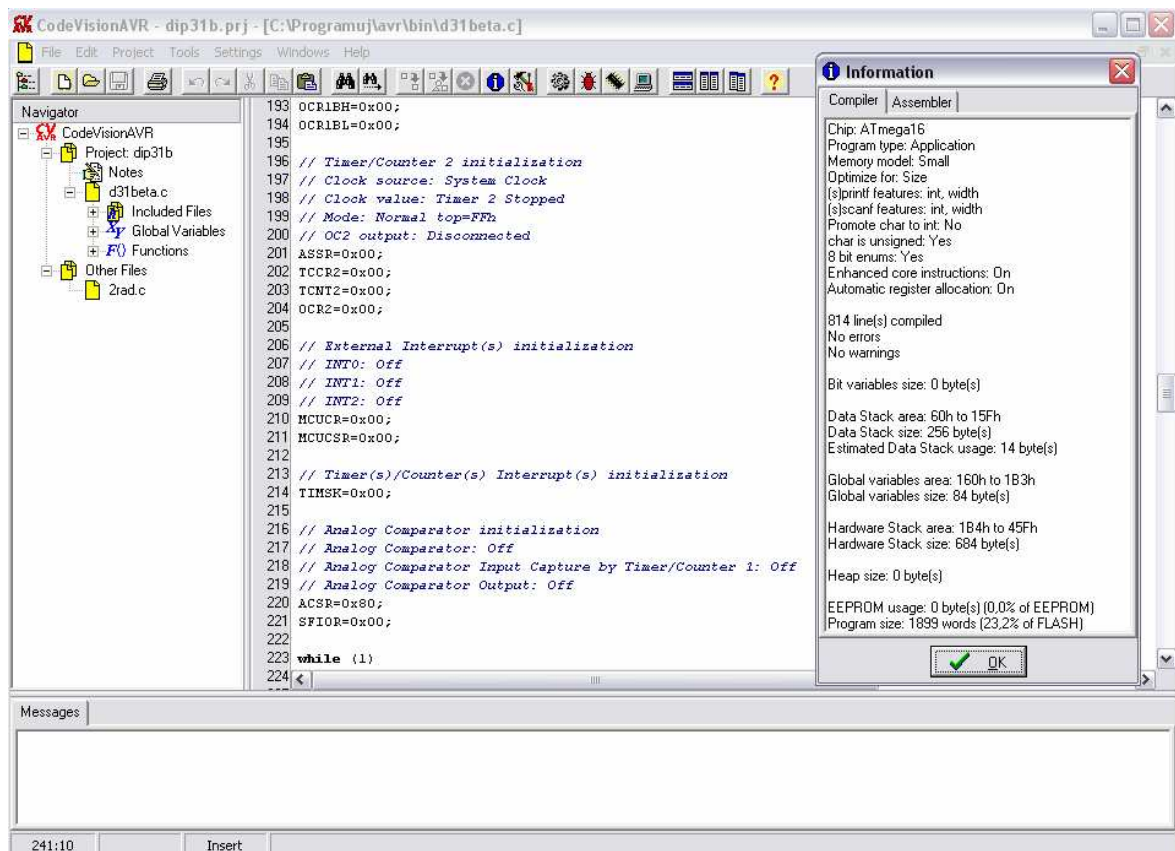
AREF – AREF je analogový referenční pin pro A/D převodník

2 PROGRAMOVÉ VYBAVENÍ

2.1 CodeVision AVR

Program CodeVision AVR je ideálním nástrojem pro programování mikročítačů. Program disponuje velmi přívětivým prostředím a je vhodným nástrojem pro začínající programátory. K dispozici je navíc funkce Wizard, kde stačí nastavit parametry mikročítače, včetně všech vstupů, výstupů a připojených zařízení (LCD), která automaticky vygeneruje kód pro inicializaci mikročítače, což ušetří spoustu času a problémů s programováním. Poté stačí již jen zadávat program do těla vygenerovaného kódu.

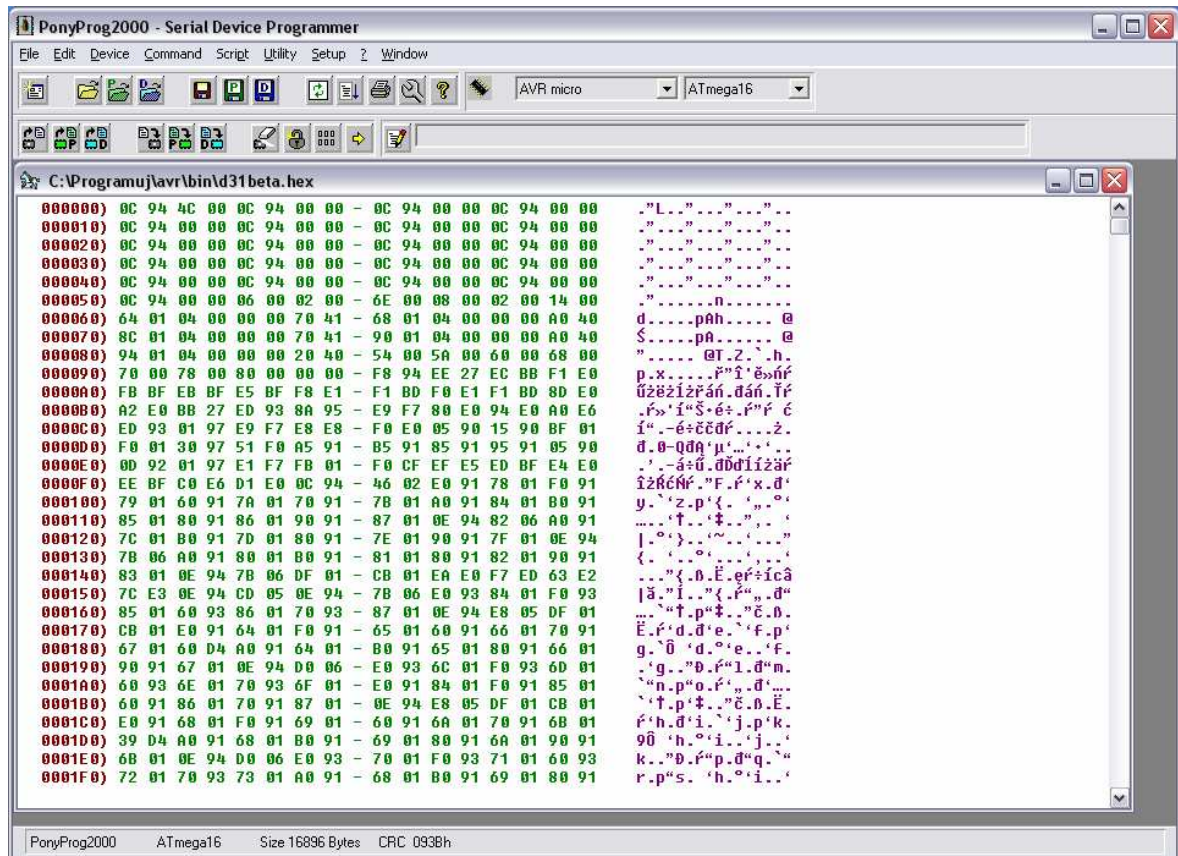
CodeVision AVR navíc dokáže již vytvořený program nahrát přímo do mikročítače. Ale vzhledem k tomu, že používáme programátor STK200 a k programování v CodeVision AVR je nutný programátor STK500, je pro nás tato funkce nedostupná. Z tohoto důvodu používáme externí program PonyProg pro nahrávání kódu do mikročítače.



Obr. 5 - CodeVision AVR – programátor mikročítačů AVR

2.2 PonyProg2000

Program PonyProg2000 slouží k nahrávání již vytvořeného programu do mikropočítače. K dispozici je nastavení spousty programátorů a typů mikropočítače včetně našeho programátoru STK200 a mikropočítače ATmega16. Stručný návod je popsán v kapitole „Stručný průvodce PonyProg“.



Obr. 6 - PonyProg2000

2.3 Program Mosaic

MOSAIC je integrované vývojové prostředí, které umožňuje vytvářet aplikační programy pro PLC TECOMAT. Prostedí umožňuje programování v jazyce instrukcí (mnemokód), systémy s 32 bitovými procesory (TECOMAT TC650 a TC700) lze programovat také v jazycích podle IEC EN 61131-3 (IL, ST, LD, FBD). Součástí prostředí MOSAIC je i řada nástrojů usnadňujících vývoj a ladění aplikací. Prostedí zachovává kompatibilitu se starším dosovským prostředím xPRO a umožňuje pracovat se zdrojovými programy vytvořenými v tomto prostředí.

Vlastnosti programu MOSAIC:

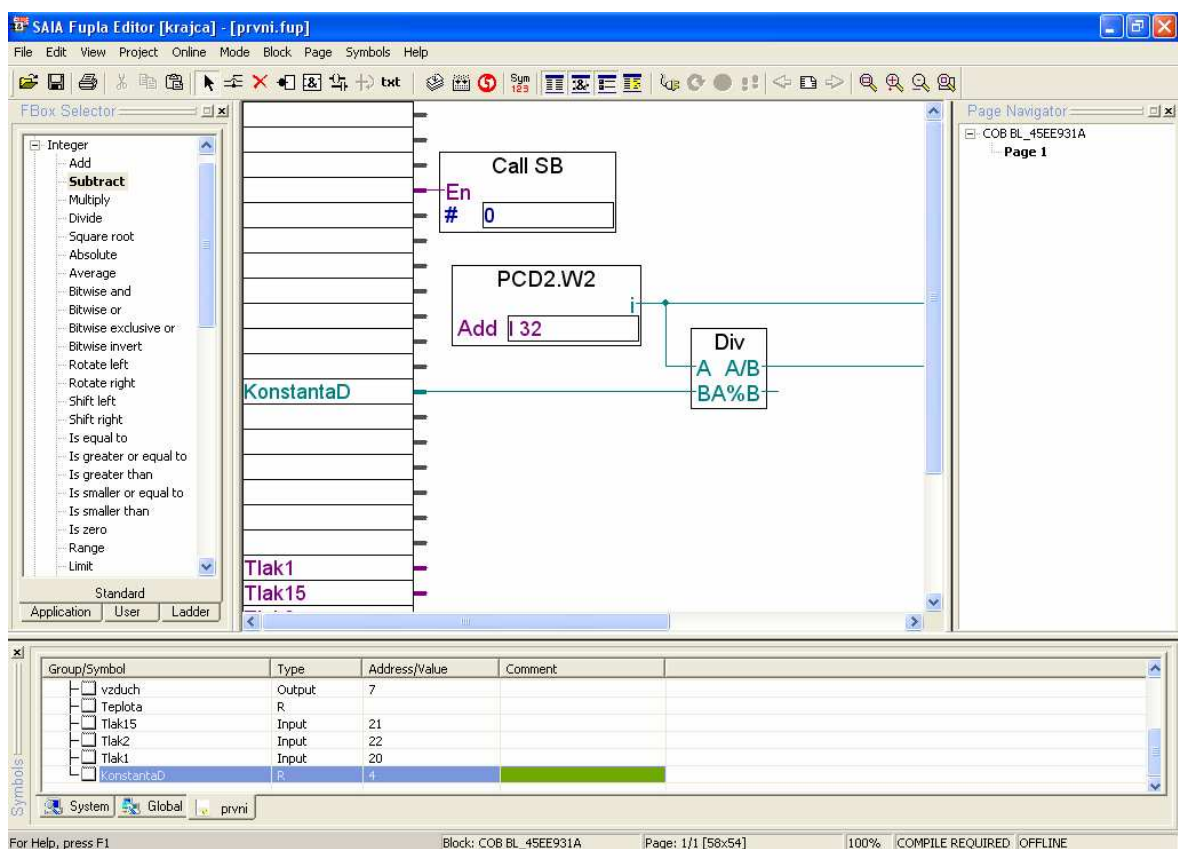
- Pracuje pod Windows 2000/XP
- Podporuje programování v jazycích ST (strukturovaný text), IL (jazyk instrukcí), LD (reléové schéma) a FBD (funkční bloky) podle normy IEC 61131-3
- IEC manažer pro grafickou deklaraci všech prvků programu PLC - datových typů, proměnných, funkcí, funkčních bloků i programových jednotek; možnost deklarace vlastních knihoven
- Inspektor POU - nástroj pro ladění programu PLC, sleduje a zobrazuje stav vybraných proměnných, umožňuje používat ladicí body
- Simulátor PLC - dovoluje ladit programy bez nutnosti připojení reálného hardwaru, simulovat lze všechny typy PLC TECOMAT a TECOREG; k simulátoru lze připojit i vizualizační software RELIANCE a ladit celou aplikaci na jednom PC
- PanelMaker - nástroj na tvorbu dialogů pro operátorské panely ID-07, ID-08 a PLC řady TC500 a TR200; program pro panel je součástí programu pro PLC
- PanelSim - simulátor operátorských panelů dovoluje zkusit dialogy vytvořené PanelMakerem bez připojení skutečného panelu, funguje jak s reálným, tak i simulovaným PLC
- PIDMaker - nástroj pro ladění a návrh PID regulátorů; nabízí interaktivní náhled na průběh regulace, usnadňuje správné nastavení parametrů regulátoru a generuje programový kód. Součástí je simulace jednoduchých soustav do třetího řádu s dopravním zpožděním.
- GraphMaker - nástroj pro podporu ladění a diagnostiku řízeného systému umožňuje zobrazení průběhů vybraných proměnných offline i v reálném čase. Dva sledovací kurzory, nastavitelná perioda vzorkování, umožňuje ukládání dat na disk i export do DB programů. Funkce digitálního osciloskopu (16 kanálů) a logického analyzátoru.
- Softwarová konfigurace PLC - konfigurační nástroj umožňující výběr typu PLC a definici konkrétní sestavy včetně nastavení parametrů jednotlivých modulů. Rovněž umožňuje načíst aktuální konfiguraci z připojeného PLC.

2.4 Programovací komplet - Saia®PG5 Controls Suite

Saia®PG5 je ústředním prvkem kompletu „Saia®PG5 Controls Suite“ jenž slouží jako programovací nástroj pro programátory PLC. Při programování v IL může být použit jako vývojový nástroj pro dedikované automaty, komunikační ovladače i IT funkce. Při tvorbě aplikačních programů pomáhají jeho grafické aplikační moduly uživatelům snadno implementovat i ty nejsložitější automatizační úlohy, aniž by bylo nutné je programovat v jazycích KOPLA, IL či Graftec.

Právě tímto způsobem se programuje většina aplikací pro Saia®PCD. Existující knihovny od společnosti Saia-Burgess a systémových partnerů poskytují výkonnou a komplexní základnu zejména pro projekty v oblasti automatizace infrastruktury. S pomocí programovacího nástroje Saia®FBox-Editor mohou být vyvíjeny vlastní grafické moduly, odpovídající specifickým potřebám určitých aplikací.

Všechny automaty Saia® PCD se programují pomocí kompletu Saia® PG5 Controls Suite. [10].



Obr. 8 - Saia®PG5 Controls Suite

2.5 Program Eagle

Editor plošných spojů EAGLE je uživatelsky přívětivý a výkonný nástroj pro návrh desek plošných spojů (DPS, PCB). Název EAGLE je zkratka, pocházející z původního názvu Easily Applicable Graphical Layout Editor.

Program se skládá ze tří hlavních modulů

- Editor spojů
- Editor schémat
- Autorouter

které jsou ovládány z jednoho uživatelského prostředí. Proto není třeba konvertovat netlisty mezi schémata a deskami.

Editor spojů

- největší rozměr výkresu 1.6 x 1.6m (64 x 64 inch)
- rozlišení 1/10.000 mm (0,1 mikronu)
- až 16 signálových vrstev
- klasické i SMD součástky
- dodává se s plnou sadou knihoven součástek
- snadné vytváření vlastních součástek v plně integrovaném editoru knihoven
- funkce vpřed/vzad pro LIBOVOLNÝ editační příkaz, do libovolné hloubky
- skriptové soubory pro dávkové zpracování příkazů
- poměření ploch
- funkce kopírovat a vložit pro kopírování kompletních částí výkresu
- kontrola pravidel návrhu

Editor schémat

- až 99 listů jednoho schématu
- kontrola elektrických pravidel zapojení
- prohazování hradel a pinů

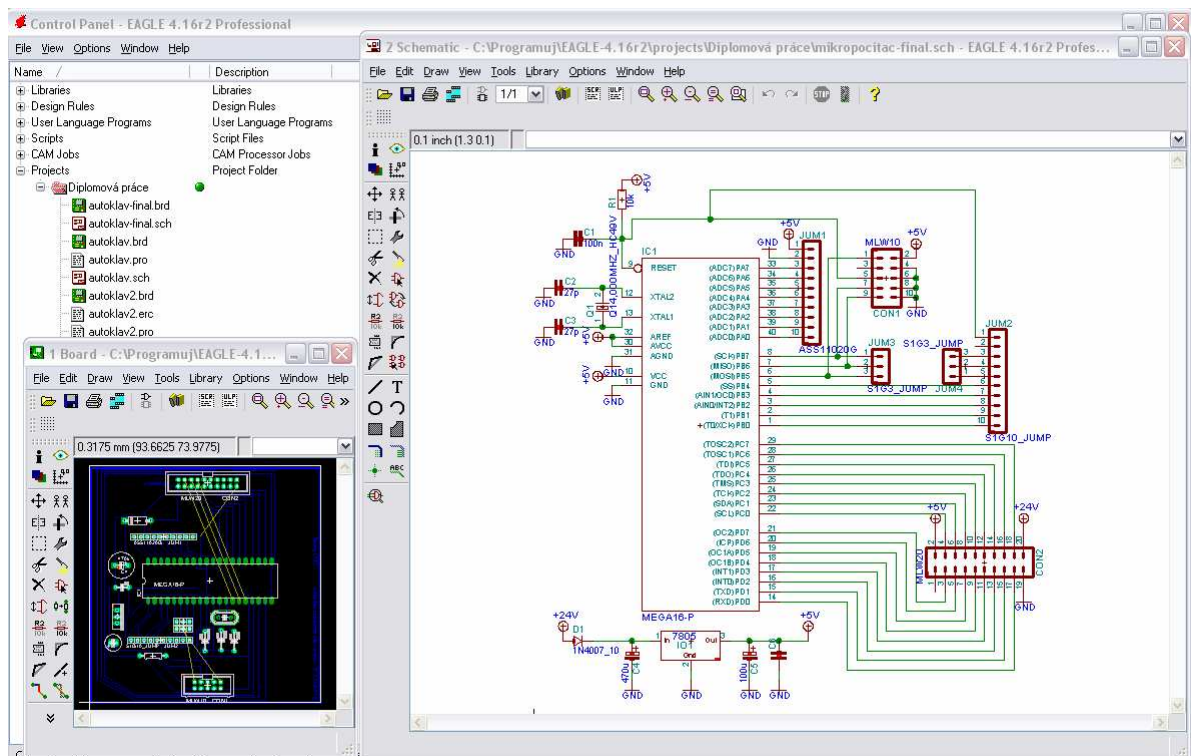
- vytvoření desky ze schématu jediným příkazem

Autorouter

- ripup&retry router
- až 16 signálových vrstev
- strategie propojování nastavitelná uživatelem pomocí váhových faktorů

Požadavky na systém

- PC Pentium (nebo lepší)
- Windows 95/98/NT4/2000 nebo Linux (kernel 2.x nebo vyšší)
- myš
- alespoň 32MB RAM
- pevný disk s alespoň 20MB volného místa [11].



Obr. 9 - EAGLE – tvorba DPS a elektronických schémat

II. PRAKTICKÁ ČÁST

3 NÁVRH MODELU REÁLNÉHO ZAŘÍZENÍ

Při návrhu reálného modelu jsem částečně vycházel z výukového systému EDUtec. Ale vzhledem k požadavkům na vlastnosti modelu (počet vstupů a výstupů, AD a DA převodníky) a použitém hardwaru, jsem byl nucen navrhnout zcela nový simulační model.

3.1 Výběr a návrh modelu

Jako model reálného zařízení jsem zvolil autokláv. Jde o vytápěné, uzavíratelné tlakové zařízení určené ke sterilizaci či dekontaminaci materiálů v mikrobiologii, farmacii, potravinářství V mém případě se jedná o parní sterilizátor firmy LAGARDE (viz.foto).



Obr. 10 - Autokláv

Ovládací panel vypadá následovně (viz.foto).



Obr. 11 - Ovládací panel autoklávu

Zvolený autokláv se nachází ve firmě InMediec s.r.o. sídlem v Luhačovicích. Tato firma se zabývá výrobou infuzních roztoků v plastových vracích. Autokláv slouží ke sterilizaci těchto roztoků.

3.1.1 Princip autoklávu LAGARDE

Princip autoklávu LAGARDE spočívá ve sterilizaci výrobku při teplotě 120°C a při tlaku 2 barů (200 kPa). Postup sterilizace je následovný:

1. do autoklávu se jednou stranou vsunou čtyři vozíky s připravenými infuzními vaky
2. uzavřou se a zajistí oboje dveře a spustí se sterilizační program
3. v první fázi programu se začne napouštět pára (zahřívá prostředí uvnitř autoklávu), voda (kvůli lepší akumulaci tepla), a vzduch (zvyšuje tlak).

4. ve druhé fázi probíhá samotná sterilizace, kdy se teplota ustálí na 120°C a tlak na 2 barech. Průměrná doba sterilizace je 50 minut. V průběhu druhé fáze se připouští jen pára a vzduch.
5. ve třetí fázi, a tudíž v poslední, dochází k chlazení. Vozíky uvnitř autoklávu jsou kropeny vodou. Ve třetí fázi je spuštěno tedy kropení a navíc se snižuje tlak otevíráním vypouštěcího ventilu.
6. Po ochlazení na požadovanou teplotu a snížení tlaku, jsou otevřeny bezpečnostní západky a vaky jsou vyjmuty z autoklávu. (celý proces trvá 1 hodinu a 30 minut).

3.1.2 Návrh modelu autoklávu

Z důvodu omezení vstupů a výstupů mikropočítače, jsem musel částečně zjednodušit model. První zjednodušení spočívá z vypuštění procesu napouštění a vypouštění vody. Ovšem na funkci modelu nemá tato absence žádný vliv. Druhé zjednodušení spočívá ve vypuštění analogového zobrazení hodnoty tlaku v autoklávu. Vzhledem k tomu, že DA převodník zabere celý port mikropočítače (8 pinů), z tohoto důvodu jsem byl nucen použít binární hodnoty místo analogových. Konkrétně jde o zobrazení tří hodnot tlaku (1 bar, 1.5 baru a 2 bary). Na výsledné simulaci nemá toto zjednodušení velký vliv. Navíc bylo tímto zjednodušením získáno 5 pinů mikropočítače.

3.1.3 Seznam všech vstupů a výstupů

Abychom mohli přiřadit jednotlivým pinům mikropočítače ATMEGA16 jejich funkci, musíme si nejdřív udělat souhrnnou tabulku všech použitých vstupů a výstupů.

PLC

Tabulka 1. Rozpis vstupů a výstupu PLC

Vstupy	Výstupy
Teplota (AD převodník)	Vzduch
Tlak 1 bar	Pára
Tlak 1,5 baru	Vypouštěcí ventil
Tlak 2 bary	Chlazení

Vizualizační panel

Tabulka 2. Rozpis vstupů a výstupů mikropočítače

Vstupy	Výstupy
Vzduch	<p>zapojeno pouze tlačítko RESET (nejde ale o standardní vstup mikropočítače)</p> <p>Vizualizační modul lze rozšířit i o potenciometry jenž lze následovně připojit na vstupy AD převodníků mikropočítače.</p>
Pára	
Vypouštěcí ventil	
Chlazení	
Tlak 1 bar	
Tlak 1,5 baru	
Tlak 2 bary	
Chyba - vysoká teplota	
Chyba - nízká teplota	
Chyba - vysoký tlak	
Chyba - nízký tlak	
Animace	

3.1.4 Rozvržení vstupně výstupních portů mikropočítače

Jak už bylo zmíněno výše, mikropočítač ATmega16 má k dispozici 4 vstupně/výstupní porty. Což je 4x8 binárních vstupů/výstupů. Vstupy a výstupy jsou brány z pohledu mikropočítače. Porty A a B jsou určeny pro připojení vizualizačního modulu a porty D a C jsou určeny pro propojení s PLC.

Tabulka 3. Rozpis jednotlivých pinů mikropočítače a jejich použití

Vstupy		Výstupy	
Využití	PIN	Využití	PORT
vzduch (PLC)	D.0	vzduch (vizualizace)	A.0
pára (PLC)	D.1	vypouštěcí ventil (viz.)	A.1
vypouštěcí ventil (PLC)	D.2	animace (vizualizace)	A.2
chlazení (PLC)	D.3	animace (vizualizace)	A.3
		animace (vizualizace)	A.4
		animace (vizualizace)	A.5
		animace (vizualizace)	A.6
		pára (vizualizace)	A.7
		chlazení (vizualizace)	B.0
		nízký tlak (vizualizace)	B.1
		nízká teplota (vizualizace)	B.2
		vysoký tlak (vizualizace)	B.3
		vysoká teplota (vizualizace)	B.4
		tlak 1 bar (vizualizace)	B.5
		tlak 1.5 baru (vizualizace)	B.6
		tlak 2 bary (vizualizace)	B.7
		tlak 1 bar (PLC)	D.4
		tlak 1.5 baru (PLC)	D.5
		tlak 2 bary (PLC)	D.6
			D.7
		DA převodník (PLC)	C

3.2 Návrh desky plošného spoje

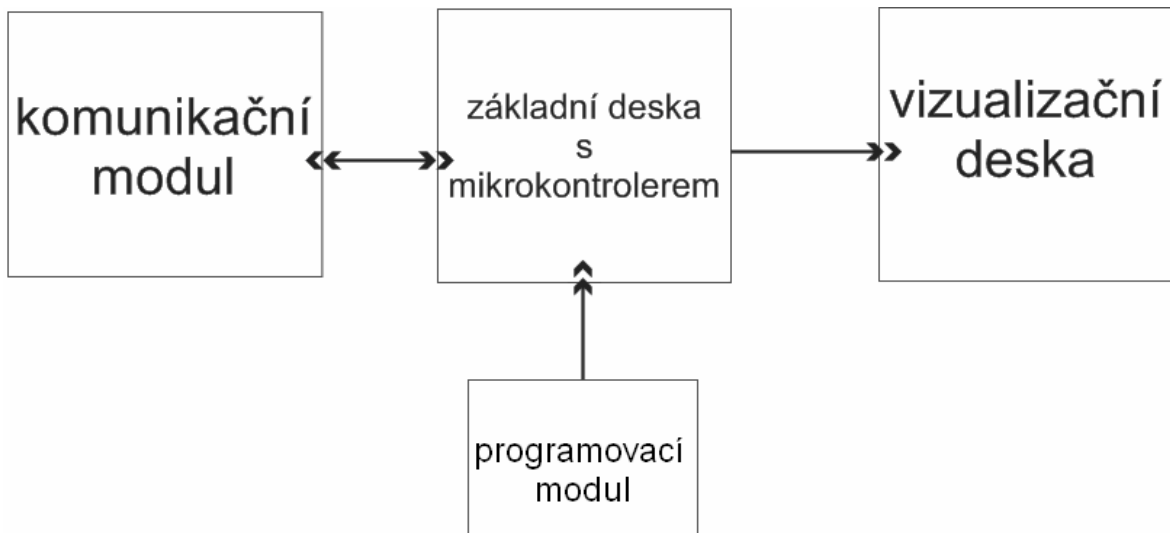
3.2.1 Celkový náhled na model

Vzhledem k velkému množství použitých součástek, velikosti mikropočítače ATmega16 a omezených rozměrů desky plošného spoje, jsem byl nucen rozdělit model na tři fyzické části.

1. Komunikační modul mezi PLC a mikropočítačem ATmega16
2. Deska s mikropočítačem ATmega16
3. Vizualizační modul

Komunikační modul se propojuje se základní deskou (deska s mikropočítačem) prostřednictvím 20 žilového sběrnicevého kabelu. Vizualizační modul se propojuje se základní deskou prostřednictvím vyvedených konektorů na základní desce.

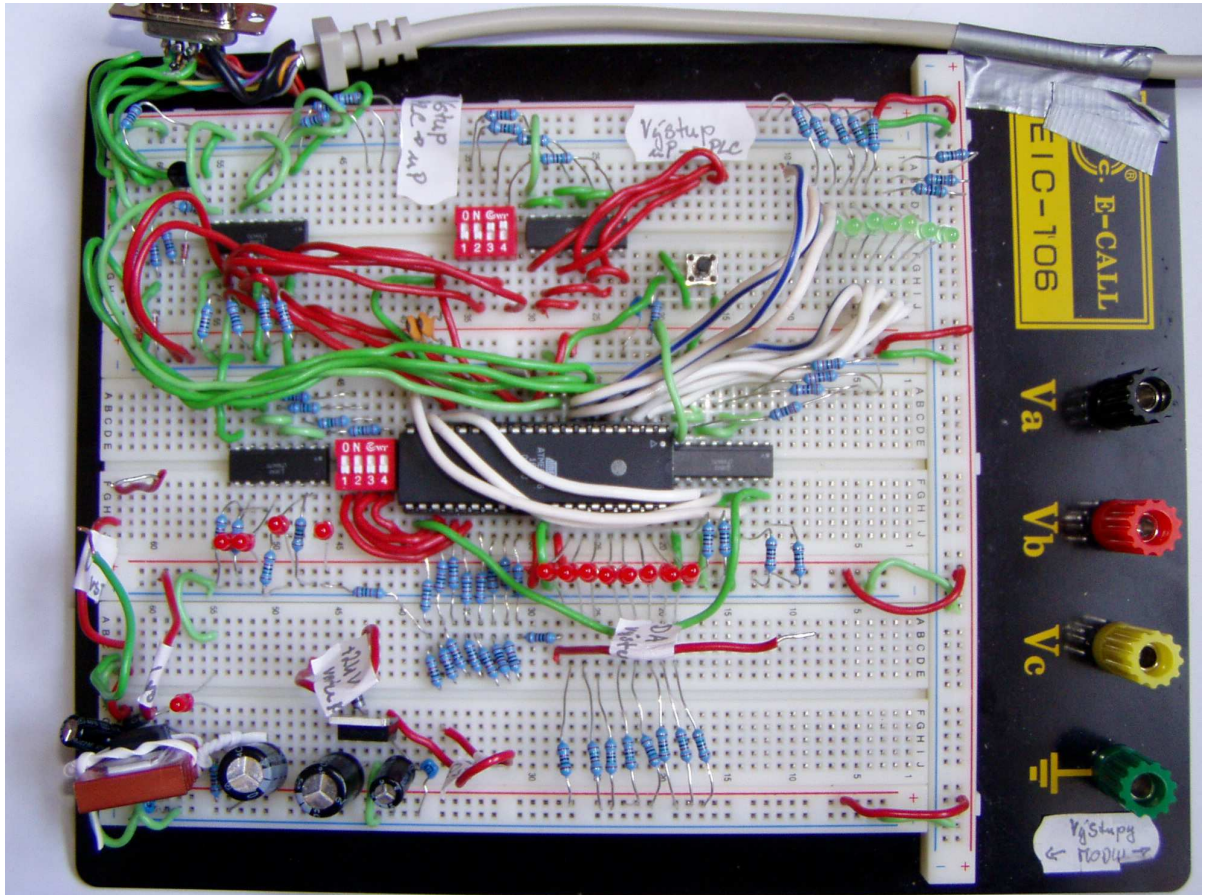
Toto rozdělení nese samozřejmě i výhodu a to, že se celá sestava stává univerzální. Změnou programu v mikropočítači a změnou vizualizačního modulu dostaneme úplně jiný model reálného zařízení, při minimálních nákladech a minimálních změnách hardwaru. Což bylo taky původním záměrem projektu.



Obr. 12 - Blokové schéma zapojení reálného modelu

3.2.2 Testování zapojení

Před návrhem plošných spojů, jsem si nejdříve ověřil celé zapojení na nepájivém kontaktním poli. Ověřil jsem si i komunikaci s PLC a software mikropočítače ATmega16. Po otestování jsem se teprve mohl uchýlit ke tvorbě plošných spojů.

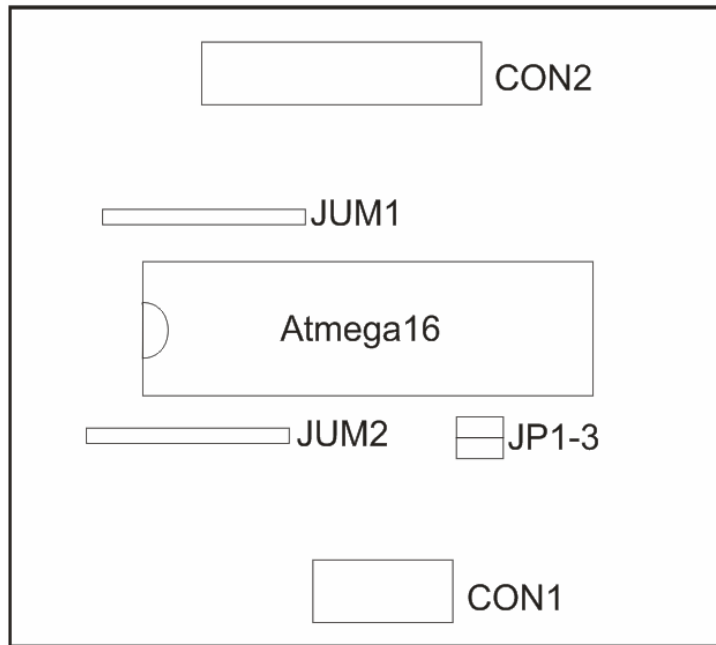


Obr. 13 - Testovací nepájivé kontaktní pole

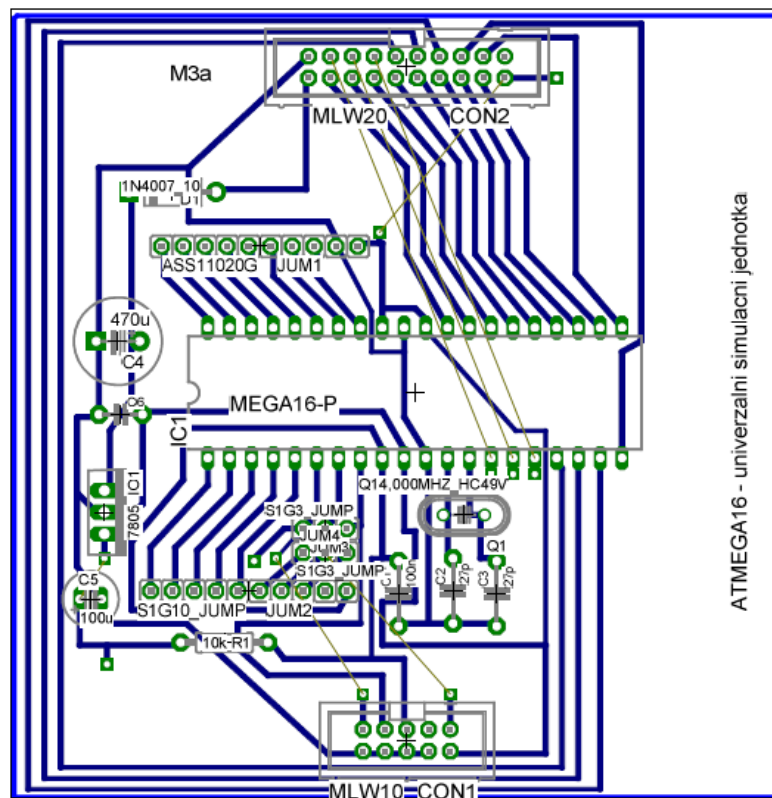
3.2.3 Základní deska s mikropočítačem

Základem celého modelu je mikropočítač ATmega16. Abychom mohli provozovat mikropočítač potřebujeme mu udat frekvenci. Pro tento účel jsem zvolil 14MHz oscilátor. Ze základní desky jsou vyvedeny dva porty. Jeden 20pinový (CON2), jenž je určen pro připojení komunikačního modulu a druhý port 10pinový (CON1), který je určen pro propojení programovacího modulu. Konektory JUM1 a JUM2 slouží pro připojení vizualizačního modulu. Konektor JUM1 zastupuje celý PORTA a JUM2 zastupuje celý PORTB mikropočítače. Jumpery JP1 až JP3 jsou určeny pro odpojení tří vizualizačních LED. Pokud budeme chtít programovat mikropočítač se zapojeným vizualizačním

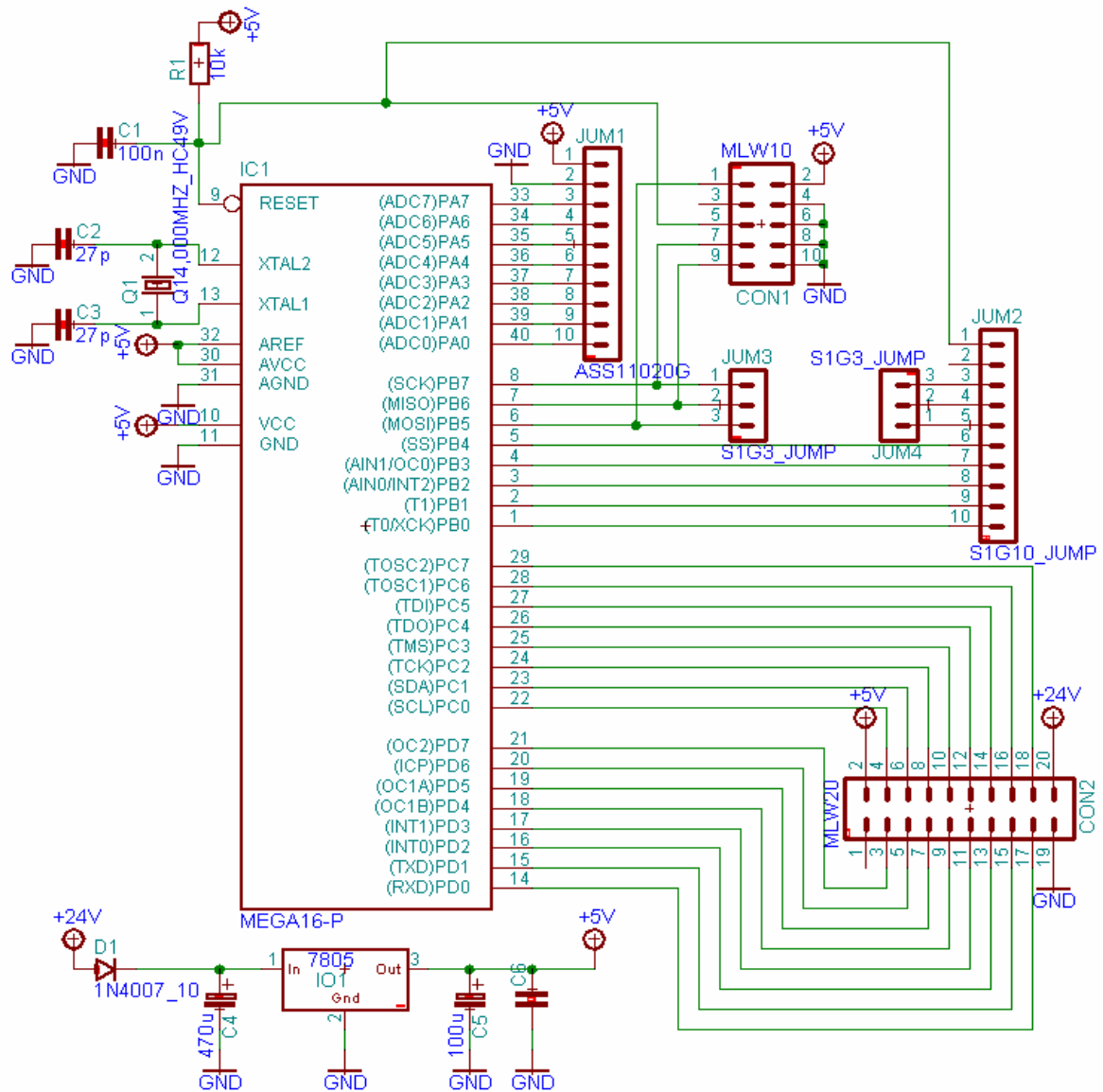
modulem, budeme muset rozpojit tyto jumpery (při programování by jinak mohly vzniknout hazardní stavy, a mohlo by dojít k špatnému vyslání kódu do mikropočítače). V jiných případech musí být jumpery vždy zapojeny.



Obr. 14 - Rozvržení základní desky s mikropočítačem



Obr. 15 – Deska plošného spoje s osazenými součástkami



Obr. 16 – Elektronické schéma základního modulu s mikro počítačem

Napájení celé soustavy je zajištěno 24V z PLC. Na základní desce je poté napětí z PLC stabilizováno na 5V a rozvedeno do všech tří modulů.

Tabulka 4. Rozvržení 20-ti pinového portu na základní desce mikro počítače

1		+5V	2
3	PORT D.7	PORT A.0	4
5	PORT D.6	PORT A.1	6
7	PORT D.5	PORT A.2	8
9	PORT D.4	PORT A.3	10
11	PORT D.3	PORT A.4	12
13	PORT D.2	PORT A.5	14
15	PORT D.1	PORT A.6	16
17	PORT D.0	PORT A.7	18
19	GND	+24V	20

Tabulka 5. Seznam použitých součástek

Označení	Krátký popis součástky	počet ks
R1	10k odpor	1
C1	100n keramický kondenzátor	1
C2-C3	27p keramický kondenzátor	2
IC1	mikropočítač ATmega16	1
CON1	konektor MLW10	1
CON2	konektor MLW20	1
JUM1,2	10 pinové konektorové kolíky	2
JUM3,4	3 pinové konektorové kolíky	2
D1	dioda 1N4007	1
C4	470u elektrolytický kondenzátor	1
C5	100u elektrolytický kondenzátor	1
C6	47n keramický kondenzátor	1
IO1	stabilizátor napětí 7805	1

3.2.4 Komunikační modul

Komunikační modul obsahuje 4 vstupně/výstupní devíti pinové canon porty pro připojení PLC. Každý canon port obsahuje 4 datové piny a dva napájecí piny (+24V a GND).

Tabulka 6. Rozvržení CANON portu na komunikačním modulu

1	2	3	4	5
GND	x	x	+24V	x
DATA.1	DATA.2	DATA.3	DATA.4	
6	7	8	9	

Vstupy a výstupy lze libovolně kombinovat až do úrovně jednotlivých pinů a to prostřednictvím přepínačů (SW1-SW4). Celkový počet je potom 8 vstupně/výstupních pinů.

K dispozici jsou další 2 tentokrát pouze výstupní devíti pinové canon porty pro připojení k PLC (8 výstupních pinů). Tyto dva porty můžeme využít jen za předpokladu, že nebudeme používat DA převodník.

K dispozici je tedy ještě 8bitový DA převodník v rozsahu 0-5V, který je sestaven pouze z odporů. Jeho vývod je označen na obrázku jako DA out.

Všechny vstupy i výstupy mikropočítače jsou připojeny skrz optočleny (optoelektronické oddělovače). Vstupy mikropočítače jsou tím chráněny proti napěťovým rázům z vnějšího zdroje. Výstupy z mikropočítače jsou též připojeny prostřednictvím optočlenů. Tím docílíme toho, že výstupním 5V napětím z mikropočítače můžeme spínat např. napětí 24V

Tabulka 7. Rozpis jednotlivých pinů CAN1 a CAN2 a přepínačů SW1 a SW2

CAN1/SW1		poloha přepínače			
		0	1		
Výstupní			ATmega16	PLC	
pin konektoru	/číslo přepínače	6/1	x	A.0	Y0.0
		7/2	x	A.1	Y0.1
		8/3	x	A.2	Y0.2
		9/4	x	A.3	Y0.3

CAN2/SW2		poloha přepínače			
		0	1		
výstupní			ATmega16	PLC	
pin konektoru	/číslo přepínače	6/1	x	A.4	Y0.4
		7/2	x	A.5	Y0.5
		8/3	x	A.6	Y0.6
		9/4	x	A.7	Y0.7

Tabulka 8. Rozpis jednotlivých pinů CAN3 a CAN4 a přepínačů SW3 a SW4

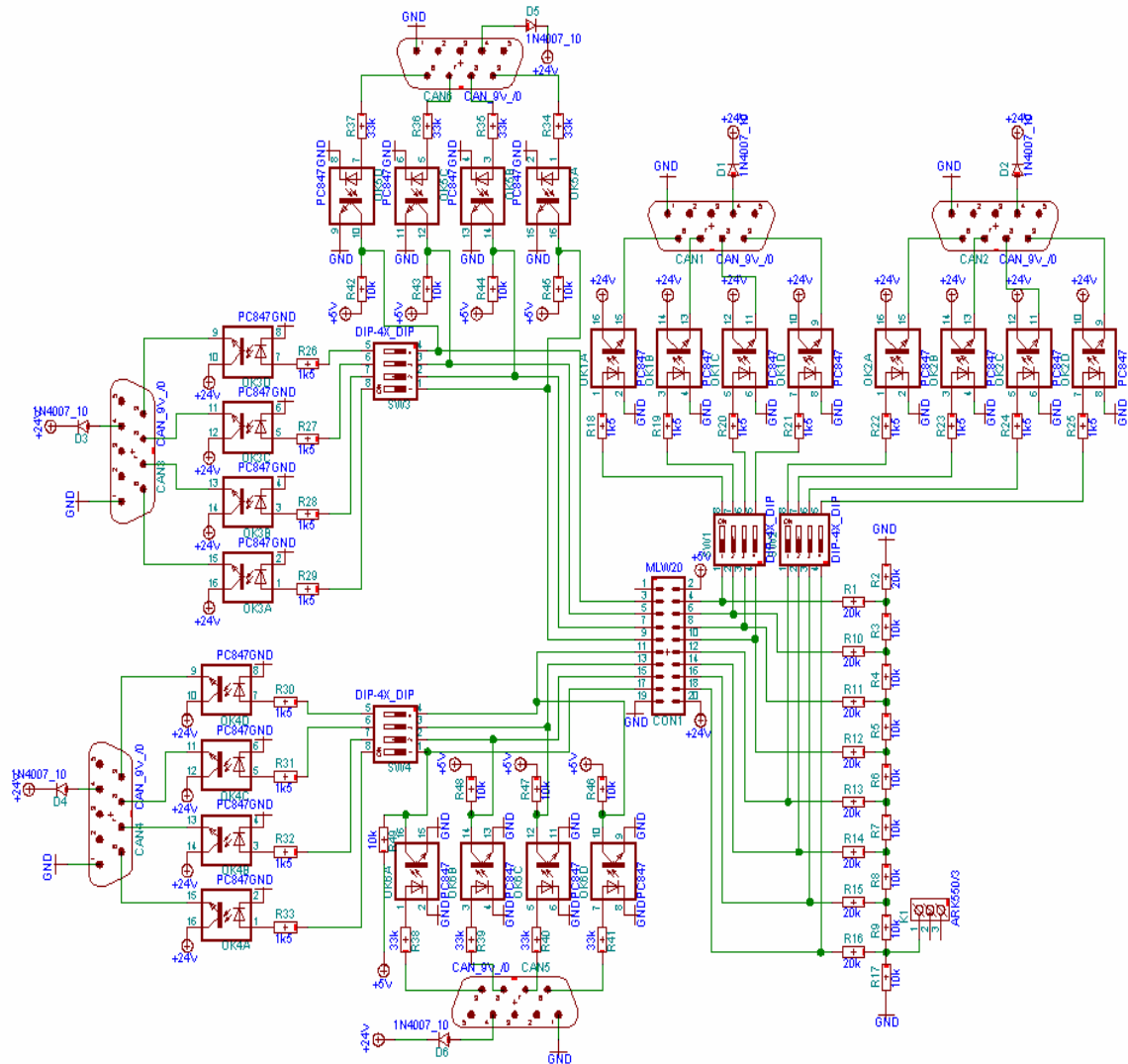
CAN3/SW3		poloha přepínače			
		1	0		
Výstupní			ATmega16	PLC	
pin konektoru	/číslo přepínače	6/1	x	D.4	Y0.0
		7/2	x	D.5	Y0.1
		8/3	x	D.6	Y0.2
		9/4	x	D.7	Y0.3

CAN4/SW4		poloha přepínače			
		1	0		
výstupní			ATmega16	PLC	
pin konektoru	/číslo přepínače	6/1	x	D.0	Y0.4
		7/2	x	D.1	Y0.5
		8/3	x	D.2	Y0.6
		9/4	x	D.3	Y0.7

Tabulka 9. Rozpis jednotlivých pinů CAN5 a CAN6 a přepínačů SW4 a SW3

CAN5/SW5		poloha přepínače			
		0	1		
Vstupní			ATmega16	PLC	
pin konektoru	/číslo přepínače	6/1	x	D.3	X0.0
		7/2	x	D.2	X0.1
		8/3	x	D.1	X0.2
		9/4	x	D.0	X0.3

CAN6/SW6		poloha přepínače			
		0	1		
vstupní			ATmega16	PLC	
pin konektoru	/číslo přepínače	6/1	x	D.7	X0.4
		7/2	x	D.6	X0.5
		8/3	x	D.5	X0.6
		9/4	x	D.4	X0.7



Obr. 19 - Elektronické schéma komunikačního modulu

Tabulka 10. Rozpis použitých součástek

Označení	Krátký popis součástky	počet ks
R1,2,10-16	20k odpor	8
R3-R9,R17	10k odpor	16
R42 – R49	10k odpor	16
R18 – R25	1k5 odpor	16
R26 – R33	1k5 odpor	16
R33 - R41	33k odpor	8
CON1	konektor MLW20	1
SW1 - SW4	DIP4xR - 4 tlačítkové přepínače	4
OK1 - OK6	PC847 optočleny	6
D1 - D6	dioda 1N4007	6
CAN1 - CAN6	konektor CAN9V	6
K1	konektor ARK550	1

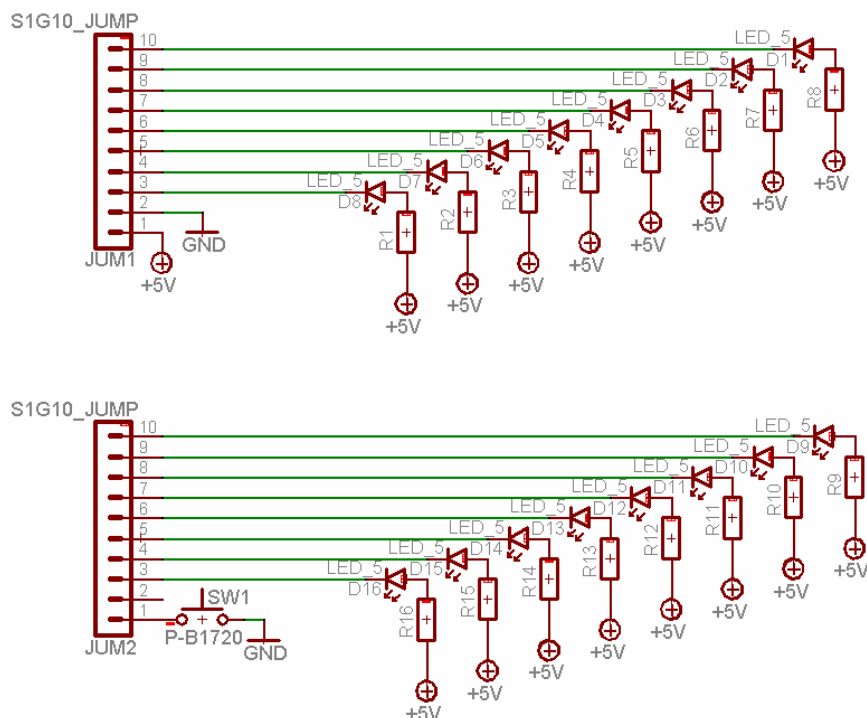
3.2.5 Vizualizační modul

Pro vizualizaci jsem použil 1,8V nízko odběrové LED a k nim připojil 1k5 odpory, abychom tyto diody nespálili výstupním napětím z mikropočítače (5V). Celkem můžeme k vizualizaci použít 16 diod. Je vyveden i jeden pin pro resetování tlačítko. Diody rozsvěcujeme tím, že vyšleme na daný pin logickou nulu (provedeme uzemnění pinu).

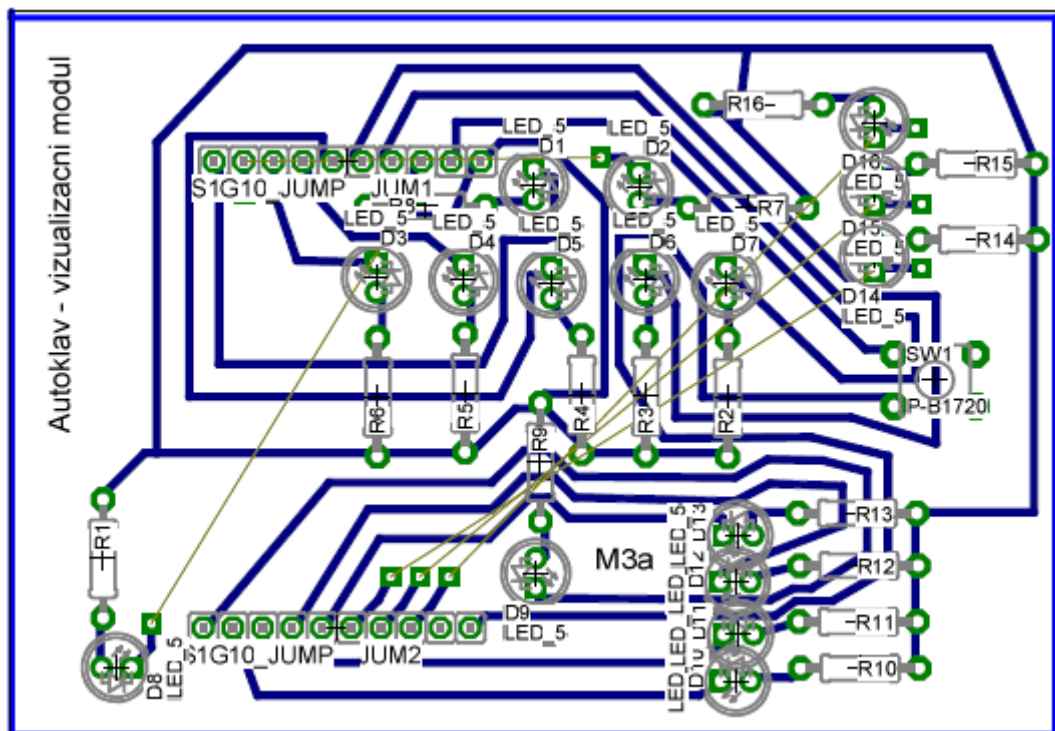
Příklad.: Chceme-li rozsvítit diodu nacházející se na sedmém pinu portu A. Použijeme následující příkaz: `PORTA.7=0;`

Tabulka 11. Rozpis pinů vizualizačního modulu

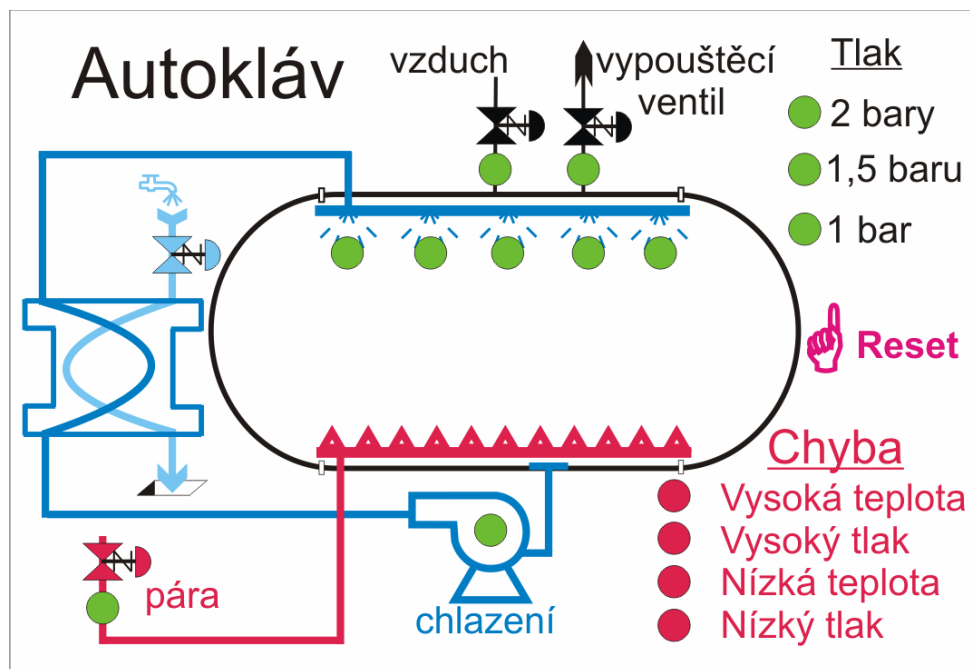
JUM_1		JUM_2	
+5V	1	Reset	1
GND	2	x	2
PORT A.7	3	PORT B.7	3
PORT A.6	4	PORT B.6	4
PORT A.5	5	PORT B.5	5
PORT A.4	6	PORT B.4	6
PORT A.3	7	PORT B.3	7
PORT A.2	8	PORT B.2	8
PORT A.1	9	PORT B.1	9
PORT A.0	10	PORT B.0	10



Obr. 20 - Elektronické schéma vizualizačního modulu



Obr. 21 - Deska plošného spoje vizualizačního modulu s osazenými součástkami



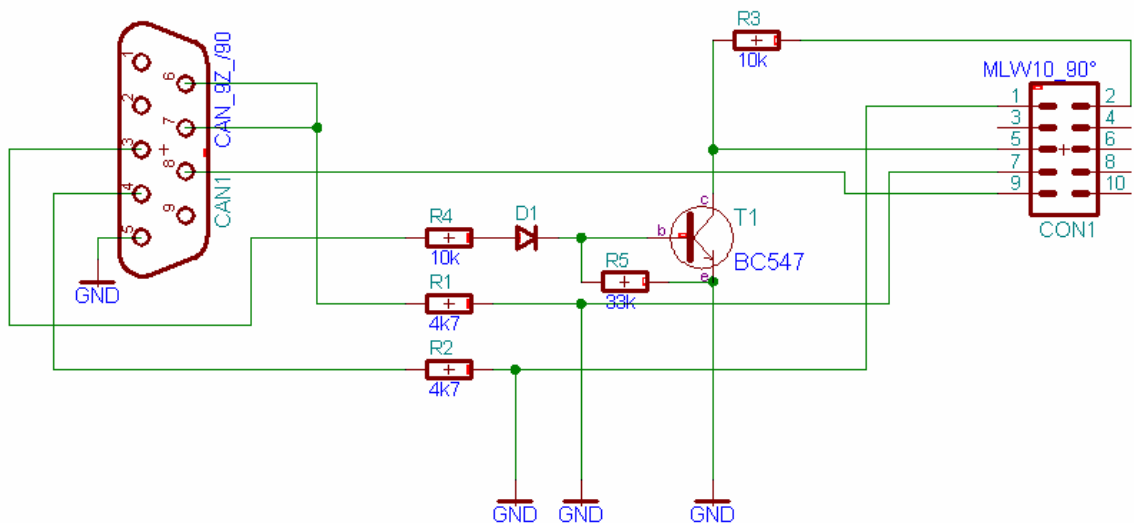
Obr. 22 – Zobrazovací schéma vizualizačního modulu

Tabulka 12. Seznam použitých součástek

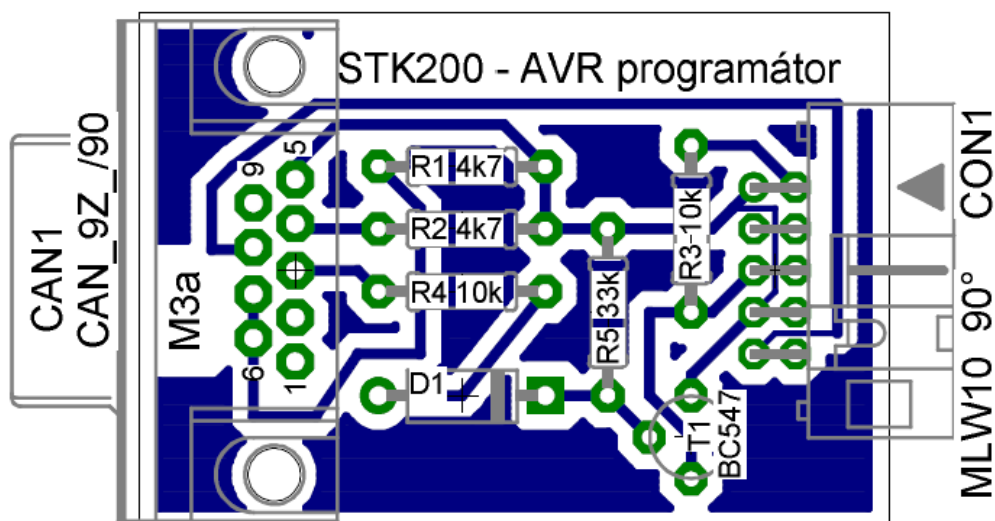
Označení	Krátký popis součástky	počet ks
R1 - R16	1k5 odpor	16
D1 - D16	1,8V LED o průměru 3mm	16
SW1	mikrospínač	1
JUM1-JUM2	10 pinové konektorové kolíky	2

3.2.6 Programovací modul

Abychom mohli snadno a rychle programovat mikropočítač ATmega16 je pro tento účel ze základní desky vyveden port pro připojení programovacího modulu. Ten je následovně připojen k PC prostřednictvím sériové linky. Programátor je velmi jednoduchý a na internetu ho najdeme pod označením: sériový programátor STK200. Návod pro zapojení a nastavení programátoru najdete v kapitole programování ATmega16.



Obr. 23 – Elektronické schéma programovacího modulu



Obr. 24 – Deska plošného spoje komunikačního modulu se součástkami

Tabulka 13. Seznam součástek

Označení	Krátký popis součástky	počet ks
CAN1	konektor CAN9V	1
CON1	konektor MLW10	1
R1,R2,R4	4k7 odpory	3
D1	dioda 1N4007	1
R5	33k odpory	1
T1	NPN tranzistor BC547	1
R3	10k odpor	1

Ve finální verzi simulačního modulu jsem programovací modul vypustil a implementoval ho přímo do sériového kabelu.

3.3 Návrh programu reálného modelu autoklávu

Abychom mohli vytvořit program pro řízení dané úlohy musíme si nejdříve stanovit podmínky řízení.

3.3.1 Stanovení podmínek a popis chování modelu

Model autoklávu nám bude simulovat proces sterilizace. Sterilizace je proces, kdy se ničí všechny bakterie. Autokláv tedy pracuje na principu, kdy za určité teploty a tlaku dochází ke sterilizaci. V našem případě se sterilizují infuzní vaky napuštěné roztokem. V autoklávu je sterilizační teplota nastavena na 120°C. Pokud bychom při zvyšování teploty nezvyšovali i tlak, došlo by k znehodnocení infuzních vaků. Důvodem je to, že při vyšších

teplotách vzniká ve vacích pára a ta roztahuje infuzní vak dokud nepraskne. V následující tabulce jsou uvedeny tlaky a k nim odpovídající teploty (hodnoty jsou získané z měření reálného modelu).

Tabulka 14. Tabulka rozsahu teplot a k nim přiřazených tlaků

Rozsah teplot	Tlak
75°C < t < 85°C	1bar
95°C < t < 105°C	1,5baru
115°C < t < 125°C	2 bary

Sterilizace se skládá ze tří hlavních částí:

1. zahřívání a zvyšování tlaku v prostředí uvnitř autoklávu
2. sterilizace – udržování stanoveného tlaku a teploty po určitou dobu
3. chlazení

Reálný autokláv se chová jakou soustavu druhého řádu. Proto chování zahřívání (chladnutí) a růst tlaku (klesání tlaku) bude popsáno rovnicemi druhého řádu.

$$Y_t = T_{poc} + k_t \left(1 - \frac{1}{T_{1t} - T_{2t}} \left(T_{1t} \cdot \exp^{\frac{-t}{T_{1t}}} - T_{2t} \cdot \exp^{\frac{-t}{T_{2t}}} \right) \right)$$

$$Y_{tl} = k_{tl} \left(1 - \frac{1}{T_{1tl} - T_{2tl}} \left(T_{1tl} \cdot \exp^{\frac{-t_{tl}}{T_{1tl}}} - T_{2tl} \cdot \exp^{\frac{-t_{tl}}{T_{2tl}}} \right) \right)$$

Simulace chování soustavy druhého řádu je vyřešena tak, že všechny hodnoty kromě času jsou nastaveny napevno. Z toho plyne, že pokud připouštím například páru, tak se pohybujeme po časové ose doprava. Ve výsledku se to jeví tak, že se soustava zahřívá. Pokud chladíme soustavu, tak se čas odečítá, tím pádem se pohybujeme po časové ose doleva a ve výsledku se soustava ochlazuje. Podobný princip platí i pro tlak.

Rovnice času pro teplotní průběh:

$$t_i = t_t + \text{krokP}_t - \text{krokV}_t - \text{krokCH} - 0.01$$

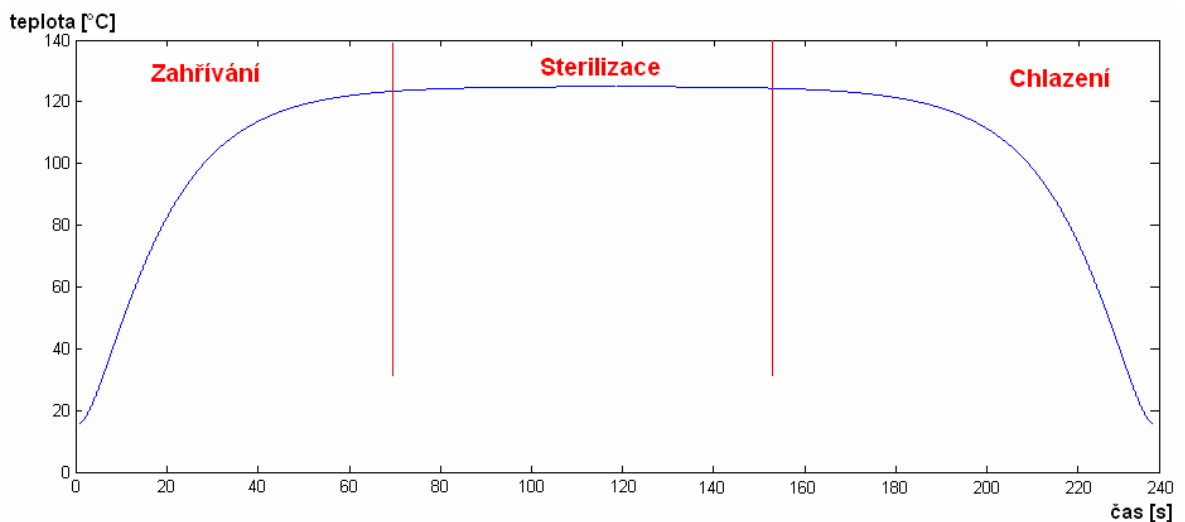
t_t = celkový čas, krokP_t = časová konstanta napouštění páry, krokV_t – časová konstanta napouštění vzduchu, krokCH – časová konstanta chlazení

Z rovnice času je zřejmé, že pokud budou všechny konstanty nulové poté bude docházet k pomalému chladnutí. (tepelné ztráty).

Rovnice času pro průběh tlaku:

$$t_{\text{tl}} = t_{\text{tl}} + \text{krokP}_{\text{tl}} + \text{krokV}_{\text{tl}} - \text{krokVV}$$

t_{tl} = celkový čas , krokP_{tl} = časová konstanta napouštění páry, krokV_{tl} – časová konstanta napouštění vzduchu, krokVV – časová konstanta vypouštění ventilu.



Obr. 25 – Průběh celého procesu sterilizace

3.3.2 Program modelu autoklávu

Nyní již známe vše potřebné k tvorbě programu pro simulaci reálného modelu autoklávu.

Program je vzhledem k jeho rozsáhlosti uveden až v příloze.

Program má celkem 814 řádků, 1899 slov. Velikost programu je 12 288 bajtů. Celkem je zaplněných 23,2% FLASH paměti mikropočítače ATmega16.

3.4 Návrh řídicího programu PLC

Nejprve jsem si vytvořil zadání, podle kterého jsem vytvářel program. Zadání může být použito i pro zadání úlohy studentům.

3.4.1 Zadání pro PLC TECOMAT 606

Vytvořte program v prostředí MOSAIC pro PLC TECOMAT TC606. Programem budete řídit model reálného zařízení autokláv. Jde o vytápěné, uzavíratelné tlakové zařízení určené ke sterilizaci či dekontaminaci materiálů v mikrobiologii, farmacii či potravinářství... Autokláv pracuje ve třech fázích: Zahřívání na požadovanou teplotu, sterilizace a chlazení. První fázi naprogramujte tak, že budete pouštět jen páru a vzduch podle následující tabulky.

Tabulka 15. Tabulka rozsahu teplot a k nim přiřazených tlaků

Rozsah teplot	Teplota v hexa pro TECO	Tlak
$75^{\circ}\text{C} < t < 85^{\circ}\text{C}$	\$500 (82°C)	1bar
$95^{\circ}\text{C} < t < 105^{\circ}\text{C}$	\$640 (102°C)	1,5baru
$115^{\circ}\text{C} < t < 125^{\circ}\text{C}$	\$754 (120°C)	2 bary

Poznámka: Při převodu teploty z desítkové soustavy na šestnáctkovou soustavu vycházejte z faktu, že počáteční teplota je 20°C a k tomu odečtená teplota z AD převodníku je 138₁₆ (v šestnáctkové soustavě)

Nápověda: Pouštějte páru dokud nedosáhnete 80°C, poté vypněte páru a dopouštějte tlak dokud nedosáhnete 1 baru. Ale pozor! Teplota Vám nesmí poklesnout pod 75°C. Tento postup opakujte pro tlaky 1,5 baru a 2 bary.

Jakmile dosáhnete teploty 120°C přecházíte do fáze sterilizace. V této fázi budete udržovat teplotu 120°C ($\pm 2^{\circ}\text{C}$) (754₁₆). Celý proces bude trvat 50 sekund. Po 50 sekundách přecházíte na fázi chlazení. Nechte puštěné chlazení po celou dobu 3.fáze a každých 5 sekund otevřete na 6 sekund vypouštěcí ventil. Celý postup opakujte do té doby dokud nedosáhnete teploty 20,5°C (140₁₆). Poté uzavřete všechny ventily kromě vypouštěcího ventilu. Po celou dobu regulace se nesmí rozsvítit varovné diody signalizující chybu !

Rozpis pinů PLC, popis zapojení konektoru a nastavení přepínačů:

Přepínače SW1,SW2,SW4 nastavte do polohy 0 a přepínač SW3 nastavte do polohy 1.

Výstupy:

Jeden z výstupních konektorů (např. Y0.4-0.7) připojte do komunikační desky ke konektoru VÝSTUP1!

y0.4 – chlazení (snižuje se teplota)

y0.5 – vypouštěcí ventil (snižování tlaku, mírně klesá i teplota)

y0.6 – připouštění páry (zahřívání, mírně stoupá tlak)

y0.7 – vzduch (zvyšuje tlak, mírně klesá teplota)

Vstupy:

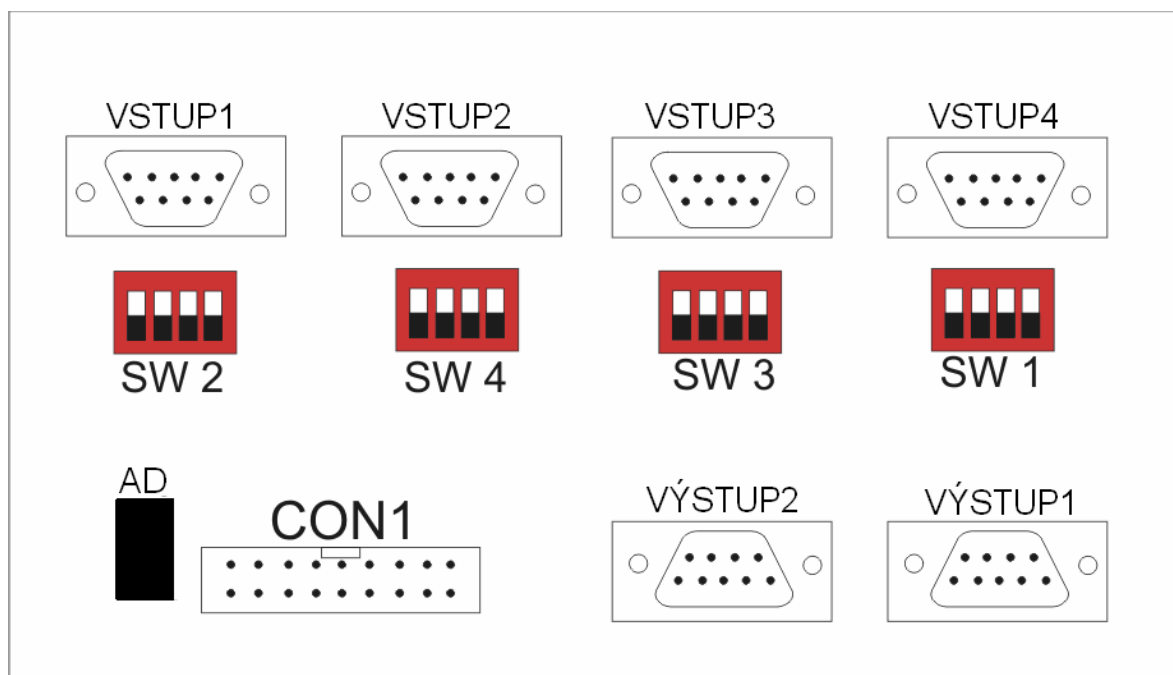
Připojte ke vstupnímu konektoru PLC pod označením AI0 výstup z komunikační desky označený AD!

Jeden ze vstupních konektorů (např. X0.4-0.7) připojte ke konektoru VSTUP3 na komunikační desce!

x0.4 – binární vstup tlaku – 1 bar

x0.5 – binární vstup tlaku – 1,5 baru

x0.6 – binární vstup tlaku – 2 bary



Obr. 26 - Schéma komunikační desky s rozpisem konektorů

3.4.2 Program určený pro PLC TECOMAT

Vzorový program pro PLC TECOMAT jsem navrhoval v prostředí MOSAIC. K programování jsem použil tzv. jazyk seznamů instrukcí IL (Instruction List) Jazyk seznamu instrukcí je nízko úroňový jazyk typu assembler. Tento jazyk patří mezi řádkově orientované jazyky.

Seznam instrukcí se skládá ze sekvence (posloupnosti) *instrukcí*. Každá instrukce (příkaz) začíná na novém řádku a obsahuje *operátor*, který může být doplněn *modifikátory*, a pokud je to pro konkrétní instrukce nutné, tak dále obsahuje jeden nebo více *operandů* oddělených čárkami. Na místě operandů mohou být libovolné reprezentace dat definované pro literály a proměnné. Pro účely identifikace může být před instrukcí uvedeno *návěští*, za kterým následuje dvojtečka. Návěští slouží k označení místa v programu pro instrukce volání resp. skoku. Na posledním místě na řádku instrukce může být uveden komentář. Mezi instrukcemi mohou být vloženy prázdné řádky.

Vzorový odladěný program pro TECOMAT 606 je uveden v Příloze PIII.

3.4.3 Zadání pro PLC SAIA PCD2

Vytvořte program v prostředí Saia® PG5 Controls Suite pro PLC Saia PCD2. Programem budete řídit model reálného zařízení autokláv. Jde o vytápěné, uzavíratelné tlakové zařízení určené ke sterilizaci či dekontaminaci materiálů v mikrobiologii, farmacii či potravinářství...

Autokláv pracuje ve třech fázích: Zahřívání na požadovanou teplotu, sterilizace a chlazení. První fázi naprogramujte tak, že budete pouštět jen páru a vzduch podle následující tabulky.

Tabulka 16. Tabulka rozsahu teplot a k nim přiřazených tlaků

Rozsah teplot	Teplota ve formátu integer	Tlak
$75^{\circ}\text{C} < t < 85^{\circ}\text{C}$	312 (80°C)	1bar
$95^{\circ}\text{C} < t < 105^{\circ}\text{C}$	390 (100°C)	1,5baru
$115^{\circ}\text{C} < t < 125^{\circ}\text{C}$	468 (120°C)	2 bary

Poznámka: Počáteční teplota je 20°C což odpovídá teplotě odečtené z AD převodníku 78.

Nápověda: Pouštějte páru dokud nedosáhnete 80°C, poté vypněte páru a dopouštějte tlak dokud nedosáhnete 1baru. Ale pozor! Teplota Vám nesmí poklesnout pod 75°C. Tento postup opakujte pro tlaky 1,5 baru a 2 bary.

Jakmile dosáhnete teploty 120°C přecházíte do fáze sterilizace. V této fázi budete odpouštět tlak a připouštět páru. Celý proces bude trvat 50 sekund. Po 50 sekundách přecházíte na fázi chlazení. Nechte puštěné chlazení po celou dobu 3. fáze a každých 5 sekund otevřete na 6 sekund vypouštěcí ventil. Celý postup opakujte do té doby dokud nedosáhnete teploty 21°C. Poté uzavřete všechny ventily kromě vypouštěcího ventilu. Po celou dobu regulace se nesmí rozsvítit varovné diody signalizující chybu !

Rozpis pinů PLC, popis zapojení konektoru a nastavení přepínačů:

Přepínače SW1,SW2,SW4 nastavte do polohy 0 a přepínač SW3 nastavte do polohy 1.

Výstupy:

Jeden z výstupních konektorů (např. Y0.4-0.7) připojte do komunikační desky ke konektoru VÝSTUP1!

O 4 – chlazení (snižuje se teplota)

O 5 – vypouštěcí ventil (snižování tlaku, mírně klesá i teplota)

O 6 – připouštění páry (zahřívání, mírně stoupá tlak)

O 7 – vzduch (zvyšuje tlak, mírně klesá teplota)

Vstupy:

Připojte ke vstupnímu konektoru PLC pod označením AI0 výstup z komunikační desky označený AD!

Jeden ze vstupních konektorů (např. X0.4-0.7) připojte ke konektoru VSTUP3 na komunikační desce!

I 20 – binární vstup tlaku – 1 bar

I 21 – binární vstup tlaku – 1,5 baru

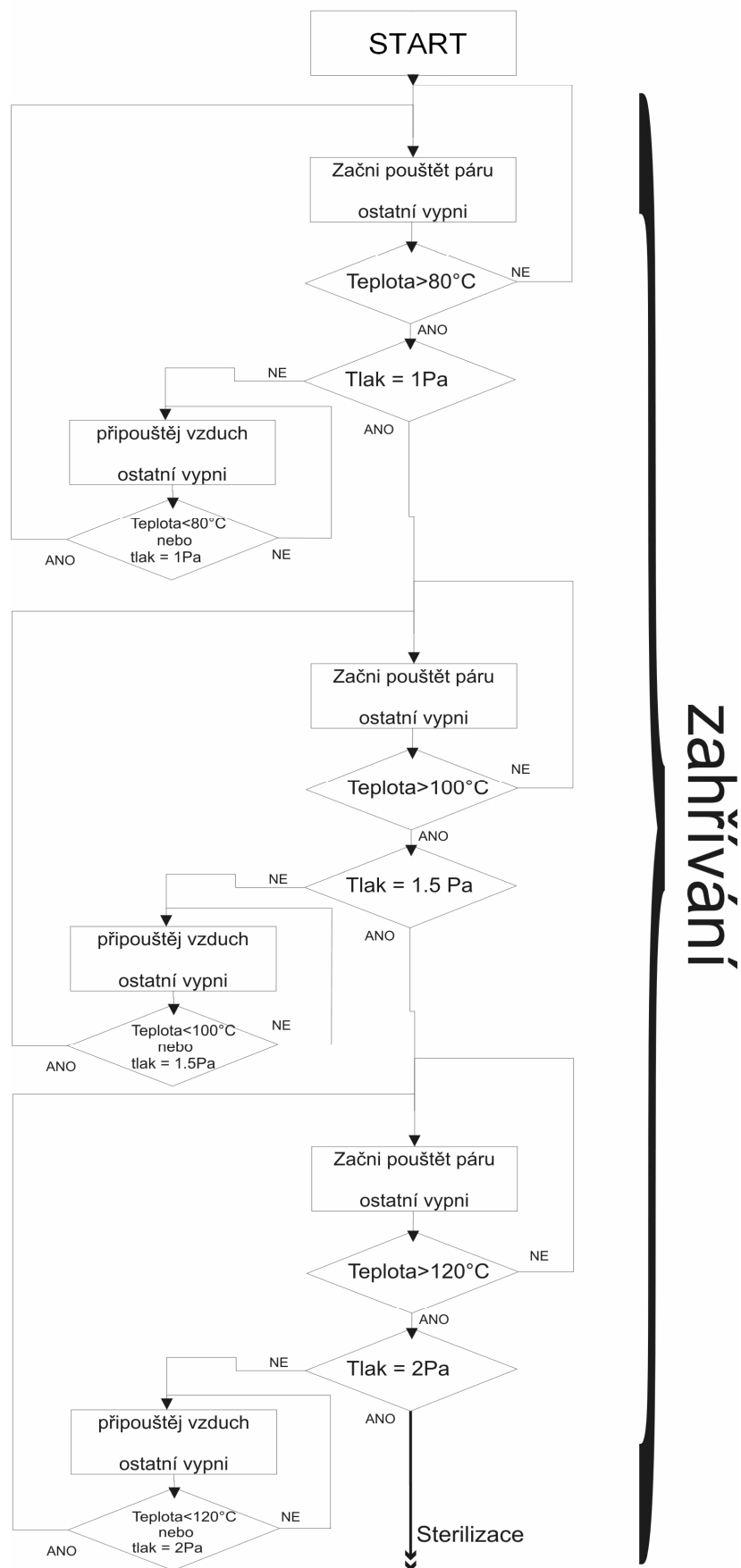
I 22 – binární vstup tlaku – 2 bary

Rozpis konektorů komunikační desky je uveden na obrázku č.26.

Pro zjednodušení úlohy a pochopení správné funkce modelu slouží následující vývojový diagram. (viz. Obrázek č.27 a 28)

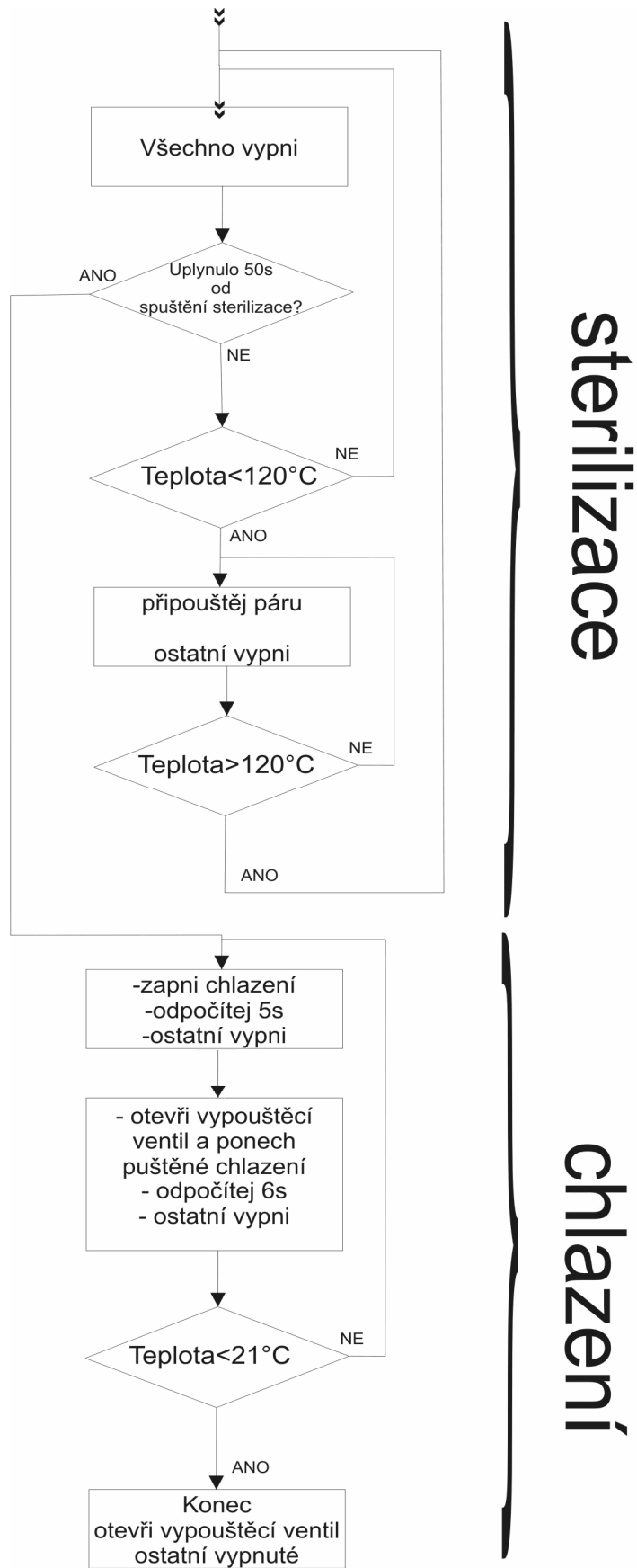
3.4.4 Program určený pro PLC SAIA

Vzorový program pro PLC SAIA jsem navrhoval v prostředí Saia® PG5 Controls Suite. Zde jsem využil sekvenčního programování. Vzorový program je uložený na přiloženém disku.



zahřívání

Obr. 27 – Vývojový diagram - zahřívání



Obr. 28 - Vývojový diagram – sterilizace a chlazení

4 ROZŠÍŘITELNOST SIMULAČNÍHO MODULU REÁLNÉHO ZAŘÍZENÍ

Simulační modul je navržen tak, aby ho bylo možné dále rozšiřovat o další modely reálných zařízení. Základní části modulu, tedy komunikační modul a základní deska osazená mikropočítačem ATMEGA16 zůstávají stejné. Mění se pouze program mikropočítače, (případně celý mikropočítač s novým programem) a vizualizační panel.

Základní deska s mikropočítačem je osazena 10pinovým konektorem určeným pro připojení sériové programovací jednotky STK200, která je součástí celého kitu. Programovací jednotka STK200 se připojuje k PC prostřednictvím sériového kabelu, který je též součástí celého kitu. Aby bylo možno korektně nahrát již vytvořený program do mikropočítače, je nutné vyjmout ze základní desky jumpery JP1 – JP3, tím se vyřadí vizualizační porty: B.7, B.6 a B.5 (TIP: pokud nebude zapojen vizualizační panel, poté jumpery JP1 – JP3 mohou zůstat zapojeny).

4.1 Programování ATMEGA16

K programování mikropočítače ATMEGA16 doporučuji použít program CodeVisionAVR jenž je nahrán na příloženém CD (aktuální verze je k dispozici na adrese: <http://www.codevision.be>). Jedná se sice o placenou verzi, ale je k dispozici i verze CodevisionAVR Evaluation, která je zdarma pouze s omezením velikosti programového kódu (pro náš případ je velikost kódu dostačující). K dispozici jsou i další programovací nástroje jako například WinAVR (stáhnout jej lze na stránkách <http://winavr.sourceforge.net/>).

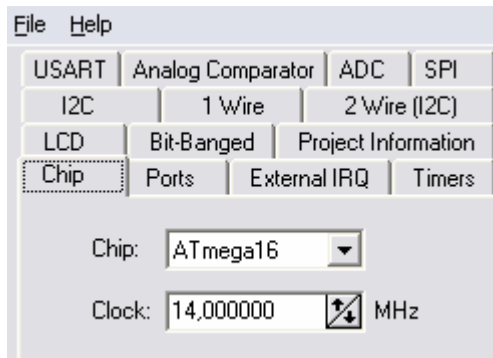
Dále budeme potřebovat software určený k nahrávání programu přímo do mikropočítače ATMEGA16 prostřednictvím sériové programovací jednotky STK200. Software se nazývá PonyProg2000 jenž je nahrán na příloženém CD (aktuální verze je k dispozici na adrese: <http://www.lancos.com/prog.html>).

4.1.1 Průvodce nastavení CodeVisionAVR

Po spuštění programu CodeVisionAVR si vytvořte nový projekt (nabídka File poté New a vyberte File Type: Project a dejte OK). Zobrazí se okno s dotazem, zda-li chcete použít

průvodce pro vytvoření nového projektu (stiskněte Yes). V průvodci je důležité nastavit následující parametry:

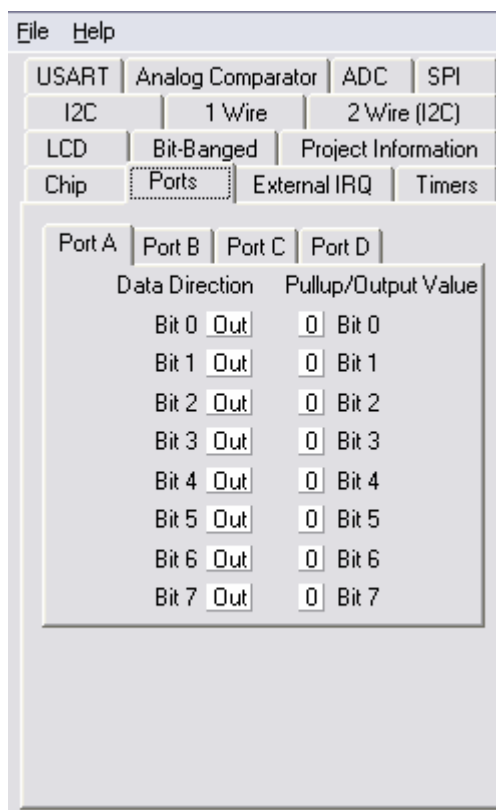
- v záložce Chip nastavit Chip:ATmega16 a Clock:14 MHz.



Obr. 29 – CodeVision AVR – nastavení mikropočítače

- v záložce Ports nastavit Port A a Port B na výstupní porty.

Port C a Port D podle toho, jak mají být využity pro propojení s PLC (vstupní/výstupní)



Obr. 30 - CodeVision AVR – nastavení portů mikropočítače

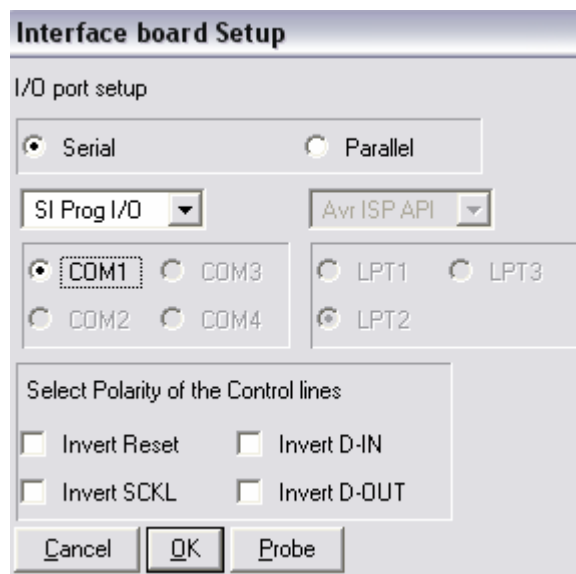
- poté klikněte na File a dejte Generate, save and exit.

- Vytvoří se nové okno, kde máte již před vyplněné nastavení mikropočítače.
- po vytvoření programu klikněte na menu Project a dejte Make. Tím se vytvoří soubory, které posléze nahrajeme do mikropočítače.

4.1.2 Průvodce nastavení PonyProg2000

Po spuštění programu PonyProg2000 zvolte nabídku Setup a poté na Interface Setup. Tam zvolte Serial a SI Prog I/O, vyberte komunikační port COM, kde máte připojený mikropočítač. Pokud kliknete na tlačítko Probe a zobrazí se dialogové okno „Test OK“ poté je nastavení v pořádku. Pokud se zobrazí okno „Test FAILED“, poté není správně připojen mikropočítač. Řešení problému může být následující:

1. zadejte jiný komunikační port (COM1, COM2 ...)
2. zkontrolujte, zda-li jsou správně propojeny kabely a zda-li je správně připojen programátor k mikropočítači
3. zkontrolujte, zda-li je zapojeno napájení mikropočítače



Obr. 31 – PonyProg2000 – nastavení programátoru

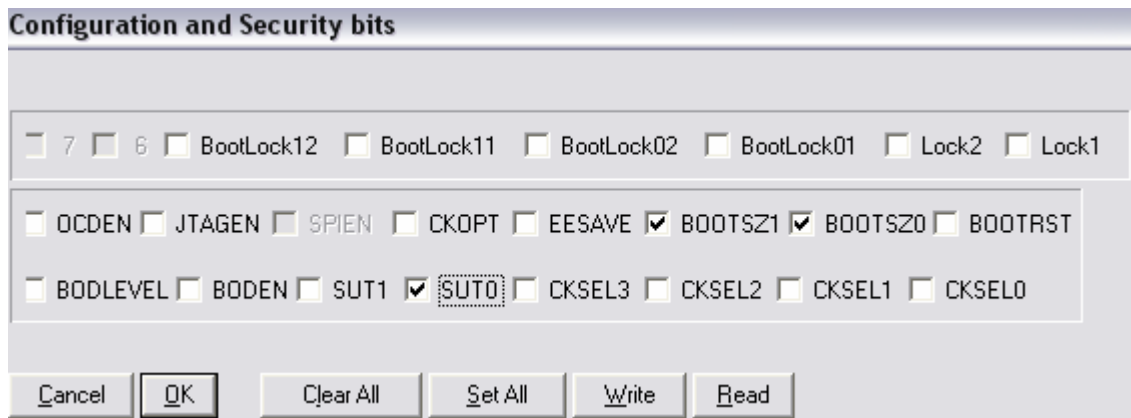
Pokud je vše v pořádku, poté opět v menu Setup klikněte na nabídku Calibration (proběhne kalibrace softwaru).

V horní liště ještě nastavte typ mikropočítače a to následovně: v první kolonce bude AVR Micro a v druhé kolonce ATmega16.



Obr. 32 – PonyProg2000 – výběr mikropočítače

Poslední nastavení je v nabídce Command a v podnabídce Configuration and Security bits. Tím se nastaví lock bity mikropočítače. Nastavte následující kolonky: BOOTSZ1, BOOTSZ0, SUT0 ostatní kolonky nechat prázdné.



Obr. 33 – PonyProg2000 – nastavení bezpečnostních bitů mikropočítače

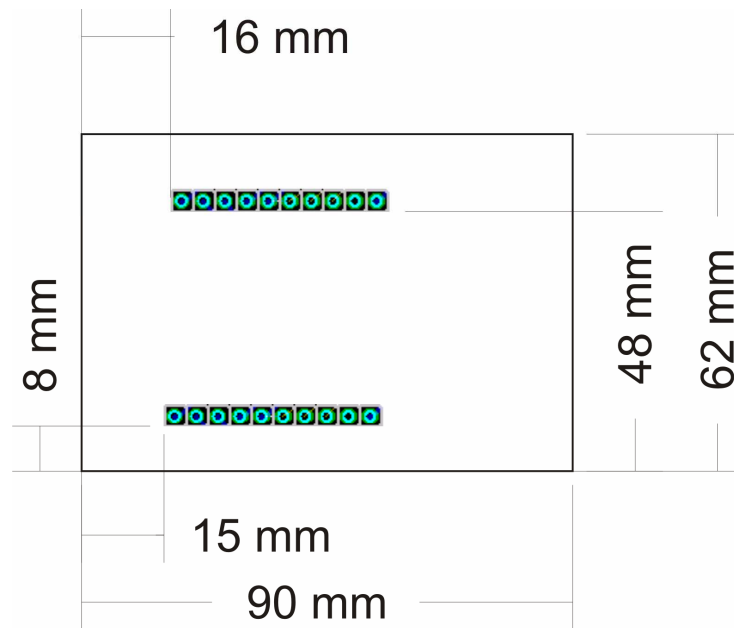
Poznámka: Při dalším zpuštění programu zůstane nastavení uloženo.

Pro nahrání vytvořeného programu do mikropočítače stačí již jen kliknout na nabídku File poté na nabídku „Open Program (FLASH) File ...“. Otevře se okno, kde zvolíte typ souboru (*.hex) a otevřete soubor s programem, který se nachází v adresáři ve kterém se nachází projekt z programu CodeVisionAVR.

4.2 Návrh plošného spoje

K vytváření plošných spojů doporučuji použít program EAGLE jenž je nahrán na příloženém CD včetně všech doposud použitých knihoven (aktuální verze je k dispozici na adrese: <http://www.cadsoft.de/>).

Při návrhu vizualizační desky je nutné zachovat její rozměry a polohu vstupních pinů. Deska má rozměry 90mm na délku x 62mm na výšku. Osazení vstupních pinů na vizualizační desce je znázorněno na následujícím obrázku.



Obr. 34 – Osazení vizualizačního modulu

TIP: Abychom mohli pracovat v metrických jednotkách *mm* v programu Eagle, je nutné si vytvořit skript. To uděláme tak, že vytvoříme prázdný soubor, např. *grid.scr* a vložíme do něj řádek: *Grid mm 0.5 on;* . Poté v nabídce File , Script... otevřeme námi vytvořený soubor. Nyní již můžete pracovat v metrických jednotkách (mm).

U vizualizačního modulu se nemusíme omezovat jen na výstupy. Základní deska s mikropočítačem je zkonstruována tak, že lze na vizualizačním modulu kromě LED použít i potenciometry nebo tlačítka.

Navíc je zde i možnost vytvoření vizualizačního modulu který bude osazen jednodušší mikropočítače (např. I8051), jenž bude provádět složitější vizualizace a komunikovat s mikropočítačem ATmega16 na základní desce. Tím můžeme značně rozšířit počet pinů určených pro vizualizaci.

ZÁVĚR

Výsledkem mé diplomové práce je plně funkční univerzální model reálného zařízení. Model se skládá ze dvou částí a to z komunikačního modulu a vizualizačního modulu s mikropočítačem.

Komunikační modul, který tvoří propojovací bránu mezi PLC a mikropočítačem, obsahuje čtyři výstupní, dva vstupní konektory a jeden DA převodník. Vstupy a výstupy PLC a mikropočítač jsou navzájem odděleny prostřednictvím optoelektronických oddělovačů.

Vizualizační modul s mikropočítačem je vytvořen formou dvou oddělitelných modulů. Modul s mikropočítačem tedy obsahuje mikropočítač ATmega16 a stabilizátor napětí. Vizualizační modul obsahuje řadu různobarevných LED zobrazujících stav modelu reálného zařízení. Navíc jsou konektory konstruovány tak, že lze modul osadit spínači, potenciometry nebo například tepelnými čidly.

Univerzálnost modelu spočívá v tom, že pouhou změnou vizualizačního modulu a mikropočítače (s rozdílným programem) dosáhneme zcela odlišného modelu reálného zařízení. Navíc je model vybaven konektorem pro programování mikropočítače, pomocí něhož lze snadno a rychle nahrát software do mikropočítače a ihned otestovat jeho funkčnost.

Jako model reálného zařízení jsem si zvolil autokláv. Jde o vytápěné, uzavíratelné tlakové zařízení určené ke sterilizaci či dekontaminaci materiálů v mikrobiologii, farmacii apod. K řízení tohoto modelu jsem naprogramoval software pro PLC TECO606 a PLC SAIA PCD2.

V diplomové práci je kompletně popsán postup tvorby modelu reálného zařízení včetně elektronických schémat, desek plošných spojů a softwarového vybavení pro zvolený mikropočítač.

V diplomové práci je též uveden podrobný postup pro tvorbu dalších simulačních panelů připojitelných k základnímu modulu.

ZÁVĚR V ANGLIČTINĚ

Result of my graduation theses is fully functional universal model of real establishment. Model is consisting from two parts. First part is communication module and second part is visualization module with microcomputer.

Communication module, which forms connecting gate between PLC and microcomputer, contains four outputs, two inputs connectors and one DA converter. Inputs and outputs of PLC and microcomputer are respectively separate through optoelectronic separators.

Visualization module with microcomputer is created by form of two separable modules. Unit with microcomputer includes microcomputer ATmega16 and voltage stabilizer. Visualization module contains series of variegated LED displaying state of model real establishment. In addition connectors are designed so, that it is possible to occupy module by switches, potentiometers or for example thermal sensors.

Universality of this model repose in that, that pure change of visualization module and microcomputer (with different program) we can get quiet different model of real establishment. In addition model is equipped by connector for programmed of microcomputer, by the help of it we can easily and quickly download software into microcomputer and we can immediately test its functionality.

I selected autoclave as a model of real establishment. It is hot press, concludes pressurized establishment which is intended to sterilization or decontamination materials in branch of microbiology, pharmacy etc. I programmed software for PLC TECO606 and PLC SAIA PCD2 to control this model.

In graduation theses is completely described design progress of model of real establishment which including electronic schematics, boards printed circuits and software equipment for selected microcomputer.

In graduation theses is also described detailed progress for production of other simulation boards attachable toward base module.

SEZNAM POUŽITÉ LITERATURY

- [1] Martinásková, M., Šmejkal, L.: Řízení programovatelnými automaty, Vydavatelství ČVUT, Praha, 1998.
- [2] Martinásková, M., Šmejkal, L.: Řízení programovatelnými automaty II, Vydavatelství ČVUT, Praha, 2000.
- [3] Martinásková, M., Šmejkal, L.: PLC a automatizace, Nakladatelství BEN – technická literatura, Praha, 1999.
- [4] Firemní literatura k programovatelnému automatu TECO 600.
- [5] Firemní literatura k produktům firmy Saia Burgess.
- [6] Matoušek D.: Práce s mikrokontroléry ATMEL AVR – Atmega16, Nakladatelství BEN – technická literatura, Praha, 2006. ISBN: 80-7300-066-0
- [7] Váňa V.: Atmel AVR programování v jazyce C, Nakladatelství BEN – technická literatura, Praha, 2003. ISBN: 80-7300-102-0
- [8] Matoušek D.: Práce s mikrokontroléry ATMEL AVR – uP a praxe, Nakladatelství BEN – technická literatura, Praha, 2003. ISBN: 80-7300-174-8
- [9] Programovatelné automaty TECO [online]. [cit. 2007-5-2]. Dostupný z URL: <www.tecomat.cz>.
- [10] Programovatelné automaty SAIA [online]. [cit. 2007-5-2]. Dostupný z URL: <<http://www.sbsys.cz/>>.
- [11] Grafický návrhářský software EAGLE [online]. [cit. 2007-4-28]. Dostupný z URL: <<http://www.cadsoft.de/info.htm>>.
- [12] Atmel – mikropočítač ATmega16 [online]. [cit. 2007-5-1]. Dostupný z URL:<http://www.atmel.com/dyn/products/product_card.asp?part_id=2010>.

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

DPS	Deska plošných spojů
PLC	Programovatelný automat (z anglického Programmable Logical Controller)
uP	Mikropočítač
DA/AD	Digitálně Analogový / Analogově Digitální
I/O	Vstupně / Výstupní (z anglického Input /Output)
LED	dioda emitující světlo (z anglického Light Emiting Diode)

SEZNAM OBRÁZKŮ

Obr. 1 - PLC SAIA PCD2.....	11
Obr. 2 - PLC TC600.....	13
Obr. 3 - Schéma mikropočítače ATmega16.....	16
Obr. 4 - Blokové schéma mikropočítače ATmega16.....	17
Obr. 5 - CodeVision AVR – programátor mikropočítačů AVR.....	19
Obr. 6 - PonyProg2000.....	20
Obr. 7 - MOSAIC – programátor a simulátor PLC TECOMAT.....	22
Obr. 8 - Saia®PG5 Controls Suite.....	23
Obr. 9 - EAGLE – tvorba DPS a elektronických schémat.....	25
Obr. 10 - Autokláv.....	27
Obr. 11 - Ovládací panel autoklávu.....	28
Obr. 12 - Blokové schéma zapojení reálného modelu.....	31
Obr. 13 - Testovací nepájivé kontaktní pole.....	32
Obr. 14 - Rozvržení základní desky s mikropočítačem.....	33
Obr. 15 – Deska plošného spoje s osazenými součástkami.....	33
Obr. 16 – Elektronické schéma základního modulu s mikropočítačem.....	34
Obr. 17 – Rozvržení komunikační desky.....	36
Obr. 18 – Deska plošného spoje komunikačního modulu s osazenými součástkami.....	36
Obr. 19 - Elektronické schéma komunikačního modulu.....	38
Obr. 20 - Elektronické schéma vizualizačního modulu.....	39
Obr. 21 - Deska plošného spoje vizualizačního modulu s osazenými součástkami.....	40
Obr. 22 – Zobrazovací schéma vizualizačního modulu.....	40
Obr. 23 – Elektronické schéma programovacího modulu.....	41
Obr. 24 – Deska plošného spoje komunikačního modulu se součástkami.....	42
Obr. 25 – Průběh celého procesu sterilizace.....	44
Obr. 26 - Schéma komunikační desky s rozpisem konektorů.....	46
Obr. 27 – Vývojový diagram - zahřívání.....	50
Obr. 28 - Vývojový diagram – sterilizace a chlazení.....	51
Obr. 29 – CodeVision AVR – nastavení mikropočítače.....	53
Obr. 30 - CodeVision AVR – nastavení portů mikropočítače.....	53
Obr. 31 – PonyProg2000 – nastavení programátoru.....	54

Obr. 32 – PonyProg2000 – výběr mikropočítače.....	55
Obr. 33 – PonyProg2000 – nastavení bezpečnostních bitů mikropočítače.....	55
Obr. 34 – Osazení vizualizačního modulu.....	56

SEZNAM TABULEK

Tabulka 1. Rozpis vstupů a výstupu PLC	29
Tabulka 2. Rozpis vstupů a výstupů mikropočítače	30
Tabulka 3. Rozpis jednotlivých pinů mikropočítače a jejich použití.....	30
Tabulka 4. Rozvržení 20-ti pinového portu na základní desce mikropočítače	34
Tabulka 5. Seznam použitých součástek	35
Tabulka 6. Rozvržení CANON portu na komunikačním modulu	35
Tabulka 7. Rozpis jednotlivých pinů CAN1 a CAN2 a přepínačů SW1 a SW2	37
Tabulka 8. Rozpis jednotlivých pinů CAN3 a CAN4 a přepínačů SW3 a SW4	37
Tabulka 9. Rozpis jednotlivých pinů CAN5 a CAN6 a přepínačů SW4 a SW3	37
Tabulka 10. Rozpis použitých součástek	38
Tabulka 11. Rozpis pinů vizualizačního modulu	39
Tabulka 12. Seznam použitých součástek	41
Tabulka 13. Seznam součástek	42
Tabulka 14. Tabulka rozsahu teplot a k nim přiřazených tlaků	43
Tabulka 15. Tabulka rozsahu teplot a k nim přiřazených tlaků	45
Tabulka 16. Tabulka rozsahu teplot a k nim přiřazených tlaků	47

SEZNAM PŘÍLOH

PŘÍLOHA P I: DESKY PLOŠNÝCH SPOJŮ

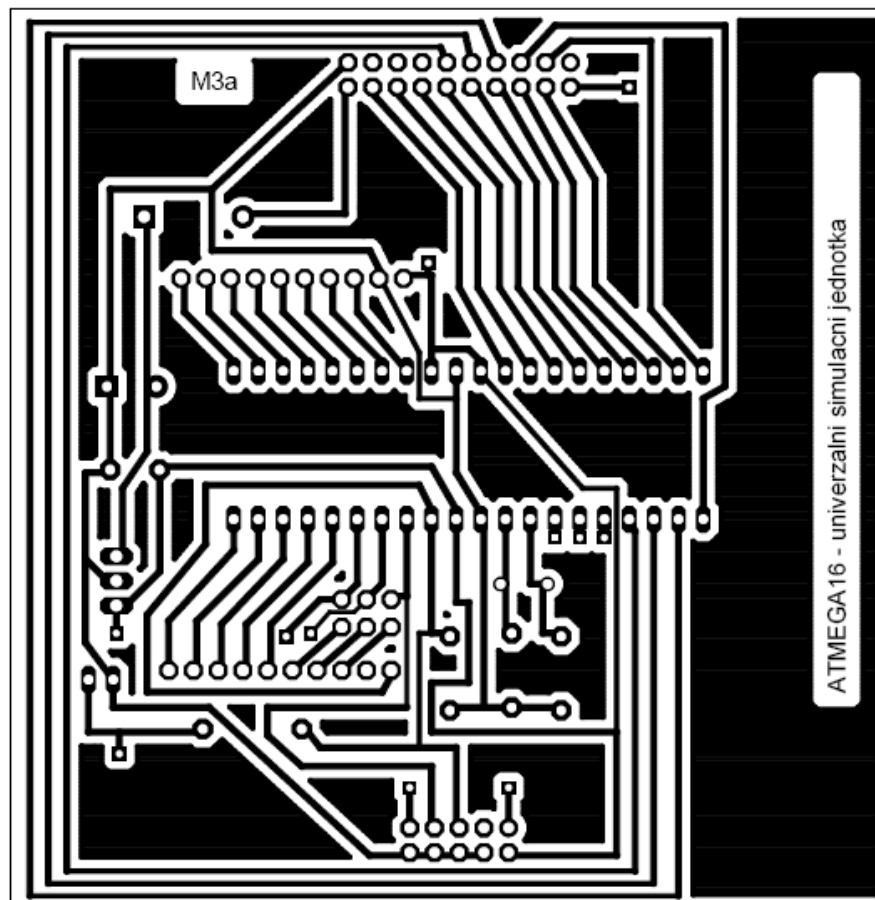
PŘÍLOHA P II: PROGRAM SIMULUJÍCÍ REÁLNÝ MODEL AUTOKLÁVU

PŘÍLOHA P III: PROGRAM V IL PRO TECOMAT606

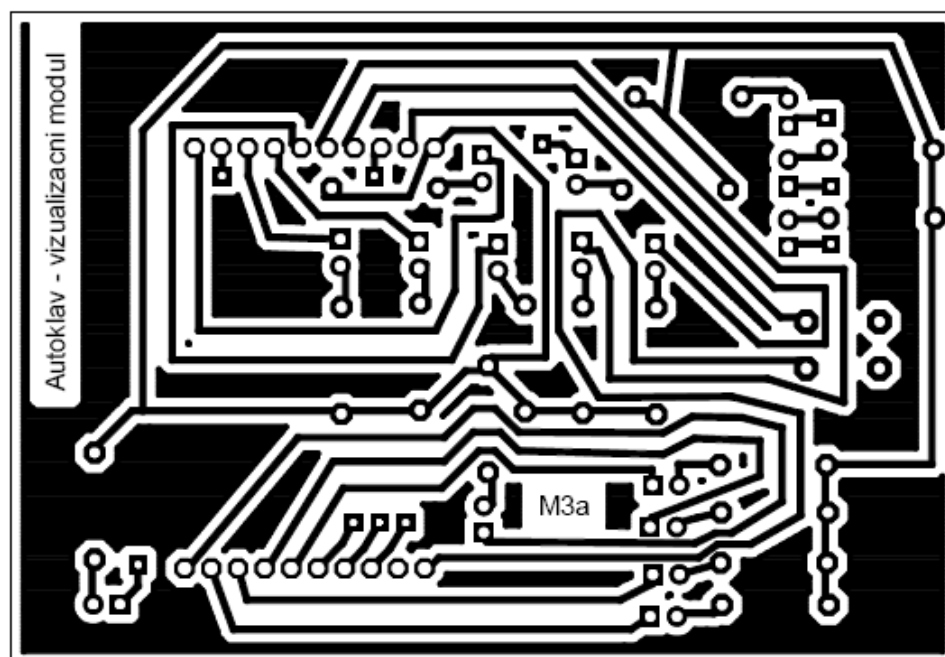
PŘÍLOHA P IV: SEZNAM MNEMOKÓDU PRO PLC TECO600

PŘÍLOHA P V: VÝVOJOVÝ DIAGRAM PRO PLC SAIA PCD2 V PROSTŘEDÍ
GRAFTEC

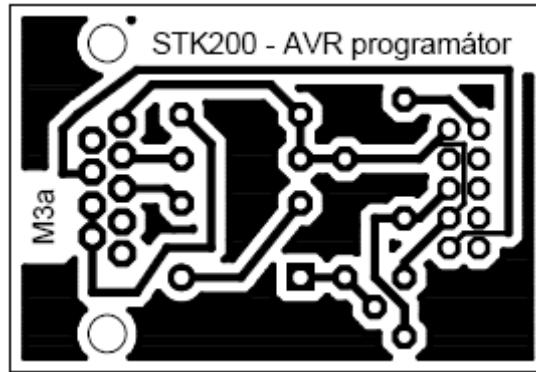
PŘÍLOHA P I: DESKY PLOŠNÝCH SPOJŮ



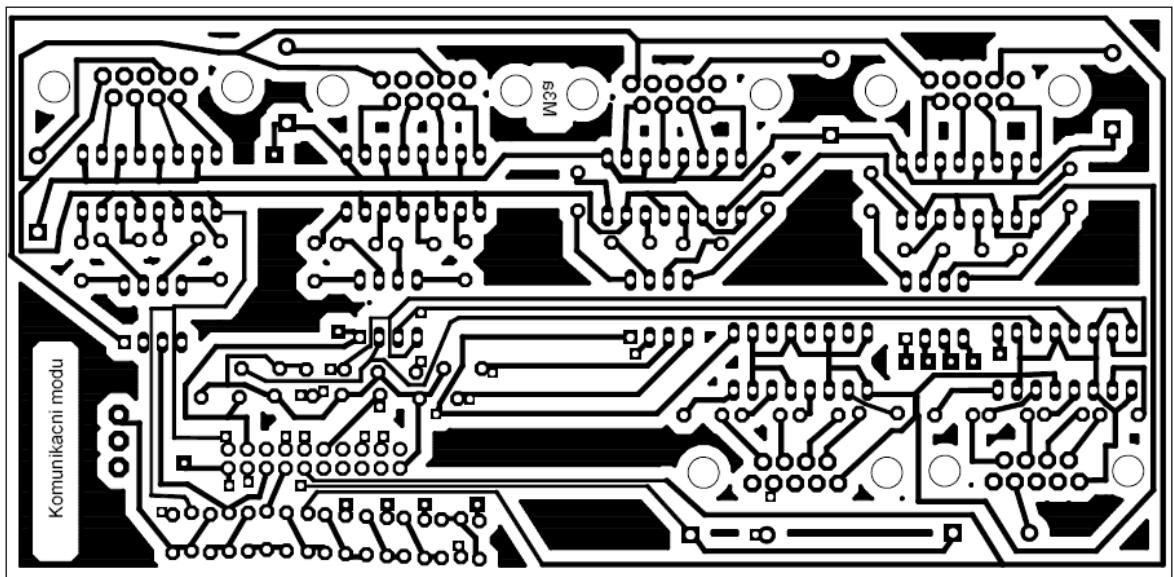
Deska plošného spoje - Základní modul s mikropočítačem ATmega16



Deska plošného spoje - Vizualizační modul



Deska plošného spoje – programovací modul



Deska plošného spoje – komunikační modul mezi PLC a mikropočítačem

PŘÍLOHA P II: PROGRAM SIMULUJÍCÍ REÁLNÝ MODEL

AUTOKLÁVU

Program je určený pro mikrokontroler ATmega16 a je naprogramován v jazyce C++.

```

/*****
This program was produced by the
CodeWizardAVR V1.24.0 Standard
Automatic Program Generator
© Copyright 1998-2003 HP InfoTech s.r.l.
http://www.hpinfotech.ro
e-mail:office@hpinfotech.ro
Project : Diplomka
Version : d31beta
Date : 16.3.2007
Author : Martin Krajca
Company :
Comments: 2.řád - Autokláv
Chip type : ATmega16
Program type : Application
Clock frequency : 14,000000 MHz
Memory model : Small
External SRAM size : 0
Data Stack size : 256
*****/
#include <mega16.h>
#include <math.h> //matematické funkce
#include <delay.h> //zpoždění
// Deklarace globálních proměnných
int i=0,kt=110,Tpoc=20,vysledek=0; //teplota
float Yt=0,T1t=15,T2t=5,x1=0,x2=0,x3=0,krokPt=0,krokVt=0,krokCH=0,tt=0;
//teplota
//kt=max teplota,krokT=teplota páry, krokVt=teplota vzduchu,
krokCH=teplota chlazení,Tpoc=počáteční teplota
float
Ytl=0,T1tl=15,T2tl=5,ktl=2.5,krokPtl=0,krokVtl=0,krokVV=0,x4=0,x5=0,x6=0,
ttl=0; //tlak
//ktl=max tlak, krokPtl=tlak páry, krokVtl=tlak vzduchu, krokVV=tlak
vypouštění
//deklarace funkcí
void teplota(void)
{ // vytvorim si vlastni funkci - rovnice 2.radu
tt=tt+krokPt-krokVt-krokCH-0.01;//udava casove kroky vypoctu
x1=T1t*(exp(-tt/T1t)); // prvni cast rovnice
x2=T2t*(exp(-tt/T2t)); // druha cast rovnice
x3=(1/(T1t-T2t)); //treti cast rovnice
Yt=Tpoc+(kt*(1-(x3*(x1-x2)))); //deleno na casti protoze neplati
priority znamének jako v matematice
vysledek=(Yt*1.9)/1; //deleni abychom dostali celociselný
vysledek kvuli DA prevodniku, nasobeni abychom vyuzili cely rozsah0-5V
}
void tlak(void)
{ // vytvorim si vlastni funkci - rovnice 2.radu
ttl=ttl+krokPtl+krokVtl-krokVV; //udava casove kroky
vypoctu
x4=T1tl*(exp(-ttl/T1tl)); // prvni cast rovnice
x5=T2tl*(exp(-ttl/T2tl)); // druha cast rovnice
x6=(1/(T1tl-T2tl)); //treti cast rovnice
Ytl=(ktl*(1-(x6*(x4-x5)))); //deleno na casti protoze neplati
priority znamének jako v matematice
}
void animace(void)
{ // vytvorim si vlastni funkci - rovnice 2.radu
```

```

        if(i==1)
        {
            PORTA.2=0;
            PORTA.3=1;
            PORTA.4=1;
            PORTA.5=1;
            PORTA.6=0;
        }
        else if(i==2)
        {
            PORTA.2=1;
            PORTA.3=0;
            PORTA.4=1;
            PORTA.5=0;
            PORTA.6=1;
        }
        else if(i==3)
        {
            PORTA.2=1;
            PORTA.3=1;
            PORTA.4=0;
            PORTA.5=1;
            PORTA.6=1;
            i=0;
        }
        else
        {
            PORTA.2=1;
            PORTA.3=1;
            PORTA.4=1;
            PORTA.5=1;
            PORTA.6=1;
        };
    }
}

void main(void)
{
    // Deklarace lokálních proměnných
    // Input/Output Ports initialization
    // Port A initialization
    // Func0=Out Func1=Out Func2=Out Func3=Out Func4=Out Func5=Out Func6=Out
    Func7=Out
    // State0=0 State1=0 State2=0 State3=0 State4=0 State5=0 State6=0
    State7=0
    PORTA=0x00;
    DDRA=0xFF;
    // Port B initialization
    // Func0=Out Func1=Out Func2=Out Func3=Out Func4=Out Func5=Out Func6=Out
    Func7=Out
    // State0=0 State1=0 State2=0 State3=0 State4=0 State5=0 State6=0
    State7=0
    PORTB=0x00;
    DDRB=0xFF;
    // Port C initialization
    // Func0=Out Func1=Out Func2=Out Func3=Out Func4=Out Func5=Out Func6=Out
    Func7=Out
    // State0=0 State1=0 State2=0 State3=0 State4=0 State5=0 State6=0
    State7=0
    PORTC=0x00;
    DDRC=0xFF;
    // Port D initialization
    // Func0=In Func1=In Func2=In Func3=In Func4=Out Func5=Out Func6=Out
    Func7=Out
    // State0=T State1=T State2=T State3=T State4=0 State5=0 State6=0
    State7=0
    PORTD=0x00;
}

```

```

DDRD=0xF0;
// Timer/Counter 0 initialization
// Clock source: System Clock
// Clock value: Timer 0 Stopped
// Mode: Normal top=FFh
// OC0 output: Disconnected
TCCR0=0x00;
TCNT0=0x00;
OCR0=0x00;
// Timer/Counter 1 initialization
// Clock source: System Clock
// Clock value: Timer 1 Stopped
// Mode: Normal top=FFFFh
// OC1A output: Discon.
// OC1B output: Discon.
// Noise Canceler: Off
// Input Capture on Falling Edge
TCCR1A=0x00;
TCCR1B=0x00;
TCNT1H=0x00;
TCNT1L=0x00;
OCR1AH=0x00;
OCR1AL=0x00;
OCR1BH=0x00;
OCR1BL=0x00;
// Timer/Counter 2 initialization
// Clock source: System Clock
// Clock value: Timer 2 Stopped
// Mode: Normal top=FFh
// OC2 output: Disconnected
ASSR=0x00;
TCCR2=0x00;
TCNT2=0x00;
OCR2=0x00;
// External Interrupt(s) initialization
// INT0: Off
// INT1: Off
// INT2: Off
MCUCR=0x00;
MCUCSR=0x00;
// Timer(s)/Counter(s) Interrupt(s) initialization
TIMSK=0x00;
// Analog Comparator initialization
// Analog Comparator: Off
// Analog Comparator Input Capture by Timer/Counter 1: Off
// Analog Comparator Output: Off
ACSR=0x80;
SFIOR=0x00;
while (1)
{
    if(PIND.0==0) //zjistuje zda li má tlakovat autokláv vzduchem
    {
        //pokud ano potom
        PORTA.0=0; //rozsvit diodu VZDUCH
        krokVt=0.01; //nastav casovy krok (pomalu chladne)
        krokVtl=0.1; //nastav casovy krok (rychle tlakuje)
    }
    else //pokud se netlakuje vzduchem potom
    {
        PORTA.0=1; //zhasni diodu VZDUCH
        krokVt=0; //nastav casovy krok (0)
        krokVtl=0; //nastav casovy krok (0)
    };
    if(PIND.1==0) //zjistí zda li se napouští pára
    {
        //pokud ano potom
        PORTA.7=0; //rozsvit diodu PÁRA a
    }
}

```

```

krokPt=0.1; //nastav casovy krok (rychlé zahřívání)
krokPtl=0.03; //nastav casovy krok (pomalu tlakuje)
}
else //pokud se voda nenapouští
{
PORTA.7=1; //zhasni diodu PÁRA a
krokPt=0; //nastav casovy krok (0)
krokPtl=0; //nastav casovy krok (0)
};
if(PIND.2==0) //zjistí zda li je otevřený VYPOUŠTĚCÍ VENTIL
{
//pokud ano potom
PORTA.1=0; //rozsvit diodu VYPOUŠTĚCÍ VENTIL
krokVV=0.1; //nastav casovy krok (rychlé snižování tlaku)
}
else //pokud ne tak
{
PORTA.1=1; //zhasni diodu VYPOUŠTĚCÍ VENTIL
krokVV=0; //nastav casovy krok (0)
};
if(PIND.3==0) //zjistí zda li se CHLADÍ
{
//pokud ano potom
PORTB.0=0; //rozsvit diodu CHLAZENÍ
krokCH=0.1; //nastav casovy krok (rychlé chládnutí)
i++;
animace();
}
else //pokud ne tak
{
i=0;
animace();
PORTB.0=1; //zhasni diodu CHLAZENÍ
krokCH=0; //nastav casovy krok (0)
};
if(Ytl<1) //vystup tlaku + vizualizace
{
PORTB.5=1; //zhasni diodu Tlak 1 bar
PORTB.6=1; //zhasni diodu Tlak 1.5 baru
PORTB.7=1; //zhasni diodu Tlak 2 bary
PORTD.4=0; //0 na vstup PLC (Tlak 1 bar)
PORTD.5=0; //0 na vstup PLC (Tlak 1.5 baru)
PORTD.6=0; //0 na vstup PLC (Tlak 2 bary)
}
else if(Ytl>1&&Ytl<1.5)
{
PORTB.5=0; //rozsvit diodu Tlak 1 bar
PORTB.6=1; //zhasni diodu Tlak 1.5 baru
PORTB.7=1; //zhasni diodu Tlak 2 bary
PORTD.4=1; //1 na vstup PLC (Tlak 1 bar)
PORTD.5=0; //0 na vstup PLC (Tlak 1.5 baru)
PORTD.6=0; //0 na vstup PLC (Tlak 2 bary)
}
else if(Ytl>1.5&&Ytl<2)
{
PORTB.5=0; //rozsvit diodu Tlak 1 bar
PORTB.6=0; //rozsvit diodu Tlak 1.5 baru
PORTB.7=1; //zhasni diodu Tlak 2 bary
PORTD.4=1; //1 na vstup PLC (Tlak 1 bar)
PORTD.5=1; //1 na vstup PLC (Tlak 1.5 baru)
PORTD.6=0; //0 na vstup PLC (Tlak 2 bary)
}
else if(Ytl>2|Ytl==2)
{
PORTB.5=0; //rozsvit diodu Tlak 1 bar
PORTB.6=0; //rozsvit diodu Tlak 1.5 baru
PORTB.7=0; //rozsvit diodu Tlak 2 bary
}

```

```

    PORTD.4=1; //1 na vstup PLC (Tlak 1 bar)
    PORTD.5=1; //1 na vstup PLC (Tlak 1.5 baru)
    PORTD.6=1; //1 na vstup PLC (Tlak 2 bary)
};
if(Yt>128) //ERROR - vysoká teplota (nad 128°C)
{
    PORTB.4=0; //rozsviť diodu VYSOKÁ TEPLOTA
}
else //pokud ne
{
    PORTB.4=1; //zhasni diodu VYSOKÁ TEPLOTA
};
if(Ytl>2.2) //ERROR - vysoký tlak (nad 2.3 baru)
{
    PORTB.3=0; //rozsviť diodu VYSOKÝ TLAK
}
else //pokud ne
{
    PORTB.3=1; //zhasni diodu VYSOKÝ TLAK
};
if(Ytl>1&&Yt<75) //ERROR nízká teplota
{
    PORTB.2=0; //rozsviť diodu NÍZKÁ TEPLOTA
}
else if(Ytl>1.5&&Yt<95)
{
    PORTB.2=0; //rozsviť diodu NÍZKÁ TEPLOTA
}
else if(Ytl>2&&Yt<115)
{
    PORTB.2=0; //rozsviť diodu NÍZKÁ TEPLOTA
}
else
{
    PORTB.2=1; //zhasni diodu NÍZKÁ TEPLOTA
};
if(Yt>85&&Ytl<1) //ERROR nízký tlak
{
    PORTB.1=0; //rozsviť diodu NÍZKÝ TLAK
}
else if(Yt>105&&Ytl<1.5)
{
    PORTB.1=0; //rozsviť diodu NÍZKÝ TLAK
}
else if(Yt>125&&Ytl<2)
{
    PORTB.1=0; //rozsviť diodu NÍZKÝ TLAK
}
else
{
    PORTB.1=1; //zhasni diodu NÍZKÝ TLAK
};
if(tt<0) //nesmi byt zaporny čas
{
    tt=0; //nastav čas na nulu
}
else if(tt>100)
{
    tt=100; //nastav čas na sto
};
if(ttl<0) //nesmi byt zaporny čas
{
    ttl=0; //nastav čas na nulu
}
else if(ttl>100)

```

```
    {
      ttl=100;    //nastav čas na sto
    };
    teplota();    //proved vypocet zahrivani - soustava 2.radu (vypocet)
    tlak();
      PORTC=vysledek; //do DA prevodniku se posle celociselny vysledek
    delay_ms(300);
  };
}
```


PŘÍLOHA P III: PROGRAM V IL PRO TECOMAT606

```
#def vzduch          %y0.7 ;deklarace výstupů
#def para           %y0.6 ;deklarace výstupů
#def vventil        %y0.5 ;deklarace výstupů
#def chlazení       %y0.4 ;deklarace výstupů
#def vstupAD        %XW2   ;deklarace výstupů
#def tlak1          %x0.4 ;deklarace vstupu
#def tlak15         %x0.5 ;deklarace vstupu
#def tlak2          %x0.6 ;deklarace vstupu
#reg float tepkonst;definice teplotní konstanty
#reg bit va, vb, vc, vd, ve
#reg bool Nulovani ;deklarace registrů
#reg uint Casovacl, Casovac2, Casovac3, Citac
P 63          ;nastavení počátečních hodnot promenných
RES va
RES vb
RES vc
RES vd
RES ve
E 63
;
P 0          ;hlavní program
;
zjistil:
ld va      ;nacti pomocnou1
JMC zahrvej1 ;pokud je nula potom skoc na zahrvej1
ld vb      ;nacti pomocnou2
JMC zahrvej2 ;pokud je nula potom skoc na zahrvej2
ld vc      ;nacti pomocnou3
JMC zahrvej3 ;pokud je nula potom skoc na zahrvej3
ld vd      ;nacti pomocnou4
JMC steril ;pokud je nula potom skoc na steril
ld ve      ;nacti pomocnou5
JMC chlad  ;pokud je nula potom skoc na chlazení
JMP konec ;jinak skoc na konec
zahrvej1:  ;1.faze zahrivani
    ld vstupAD ;nacti analogovy vstup
    GT $500     ;je li vetsi nez 80stupnu
    AND tlak1  ;a zaroven je li tlak vyssi jak 1bar
    CAD zahr1  ;potom skoc na zahr1
    ld vstupAD ;nacti analogovy vstup
    GT $508     ;je li vetsi nez 80stupnu
    ANC tlak1  ;a zaroven mensi nez 1bar
    CAD zahr2  ;potom skoc na zahr2
    ld vstupAD ;nacti analogovy vstup
    LT $500     ;je li mensi nez 80stupnu
    CAD zahr3  ;potom skoc na zahr3
    JMP konec
zahrvej2:  ;2.faze zahrivani
    ld vstupAD ;nacti analogovy vstup
    GT $640     ;je li vetsi nez 100stupnu
    AND tlak15 ;a zaroven je li tlak vyssi jak 1.5baru
    CAD zahr4  ;potom skoc na zahr4
    ld vstupAD ;nacti analogovy vstup
    GT $642     ;je li vetsi nez 100stupnu
    ANC tlak15 ;a zaroven mensi nez 1.5baru
    CAD zahr2  ;potom skoc na zahr2
    ld vstupAD ;nacti analogovy vstup
    LT $636     ;je li mensi nez 100stupnu
    CAD zahr3  ;potom skoc na zahr3
    JMP konec
zahrvej3:  ;3.faze zahrivani
    ld vstupAD ;nacti analogovy vstup
```

```

GT $754          ;je li vetsi nez 120stupnu
AND tlak2       ;a zaroven je li tlak vyssi jak 2bary
CAD zahr5       ;potom skoc na zahr5
ld vstupAD     ;nacti analogovy vstup
GT $759         ;je li vetsi nez 120stupnu
ANC tlak2       ;a zaroven mensi nez 1bar
CAD zahr2       ;potom skoc na zahr2
ld vstupAD     ;nacti analogovy vstup
LT $754         ;je li mensi nez 120stupnu
CAD zahr3       ;potom skoc na zahr3
JMP konec

steril:
ld 1            ;nacteni hodin
ld 1
TON Casovac1.2 ; zapis do Casovac po sekundach
ld vstupAD     ;nacti analogovy vstup
GT $761         ;je li vetsi nez 120stupnu
CAD ster1       ;potom skoc na ster1

ld vstupAD     ;nacti analogovy vstup
LT $755         ;je li mensi nez 120stupnu
CAD ster2       ;potom skoc na zahr3
ld Casovac2
GT 9
CAD ster3
ld Casovac3
EQ 3
CAD ster4
ld Casovac1
EQ 50
CAD ster5
JMP konec

chlad:
ld 1
set chlazeni
ld 1            ;nacteni hodin
ld 1
TON Casovac1.2 ; zapis do Casovac po sekundach
ld Casovac1
LT 7
CAD chlaz1
ld Casovac1
GT 6
CAD chlaz2
ld Casovac1
EQ 11
CAD chlaz3
ld vstupAD     ;nacti analogovy vstup
LT $140        ;je li mensi nez 25 stupnu
CAD vynuluj    ;potom skoc na ster1
JMP konec

konec:
E 0
P 60
zahr1:
SET va
RET
zahr2:
ld 0
wr para
ld 1
wr vzduch
RET
zahr3:
ld 1

```

```

        wr para
        ld 0
        wr vzduch
        RET
zahr4:
        SET vb
        RET
zahr5:
        SET vc
        RET
ster1:
        ld 0
        wr para
        wr vzduch
        wr vventil
        RET
ster2:
        ld 1           ;nacteni hodin
        ld 1
        TON Casovac2.2   ; zapis do Casovac po sekundach
        ld 1
        wr para
        ld 0
        wr vzduch
        wr vventil
        RET
ster3:
        ld 1           ;nacteni hodin
        ld 1
        TON Casovac3.2   ; zapis do Casovac po sekundach
        ld 1
        wr vventil
        RET
ster4:
        ld 0
        wr para
        wr Casovac2
        wr Casovac3
        wr vventil
        RET
ster5:
        SET vd
        ld 0
        wr para
        wr vventil
        wr vzduch
        wr Casovac1
        wr Casovac2
        wr Casovac3
        RET
chlaz1:
        ld 1
        wr vventil
        RET
chlaz2:
        ld 0
        wr vventil
        RET
chlaz3:
        ld 0
        wr Casovac1
        RET
vynuluj:
        ld 0
        wr para

```

```
wr vventil  
wr vzduch  
wr chlazeni  
wr Casovac1  
wr Casovac2  
wr Casovac3  
ld 1  
wr vventil  
wr ve  
RET
```

E 60

PŘÍLOHA P IV: SEZNAM MNEMOKÓDU PRO PLC TECO600

Instrukce pro čtení a zápis dat

Mnemo kód	Doba výkonu [μs] (Délka kódu [B])						Význam instrukce
	Z			#	U		
	bool	byte usint sint	word uint int	word uint int	byte usint sint	word uint int	
LD	63 (3)	53 (3)	61 (3)	50 (3)	81 (3)	109 (3)	Čtení přímých dat
LDC	64 (3)	54 (3)	63 (3)	52 (3)	-	-	Čtení negovaných dat
WR	67 (3)	50 (3)	57 (3)	-	80 (3)	113 (3)	Zápis přímých dat
WRC	67 (3)	51 (3)	59 (3)	-	-	-	Zápis negovaných dat
PUT	77 (3)	60 (3)	66 (3)	-	-	-	Podmíněný zápis dat
	51	46	46	-	-	-	- podmínka splněna
							- podmínka nesplněna

Logické instrukce

Mnemo kód	Doba výkonu [μs] (Délka kódu [B])				Význam instrukce
	Z		#	A	
	bool	byte usint sint	word uint int	word uint int	
AND	66 (3)	53 (3)	49 (3)	50 (1)	AND s přímým operandem
ANC	67 (3)	54 (3)	-	-	AND s negovaným operandem
OR	66 (3)	49 (3)	49 (3)	50 (1)	OR s přímým operandem
ORC	67 (3)	50 (3)	-	-	OR s negovaným operandem
XOR	66 (3)	49 (3)	49 (3)	50 (1)	XOR s přímým operandem
XOC	67 (3)	50 (3)	-	-	XOR s negovaným operandem
NEG	-	-	-	42 (1)	Negace vrcholu zásobníku
SET	66 (3)	50 (3)	-	-	Podmíněné nastavení
RES	67 (3)	53 (3)	-	-	Podmíněné nulování
LET	79 (3)	55 (3)	-	-	Impulz od náběžné hrany
FLG	-	-	-	192 (1)	Logické funkce A0
STK	-	-	-	152 (1)	Sklopení úrovní zásobníku do A0
ROL n	-	-	-	81+10n (3)	Rotace čísla vlevo
SWP	-	-	-	39 (1)	Záměna horního a dolního bytu v A0

Čítače, posuvné registry, časovače, krokový řadič

Mnemo kód	Doba výkonu [μs] (Délka kódu [B])	Význam instrukce
CTU	186 (3)	Dopředný čítač
CTD	186 (3)	Zpětný čítač
CNT	219 (3)	Obousměrný čítač
SFL	190 (3)	Posuvný registr vlevo
SFR	190 (3)	Posuvný registr vpravo
TON	220 (3)	Časovač (zpožděný přítah)
TOF	221 (3)	Časovač (zpožděný odpad)
RTO	235 (3)	Integrovaný časovač, měřič času
IMP	222 (3)	Časovač - generátor impulzu zadané délky
STE	129 (3)	Krokový řadič (stepper) - změna stavu
	106	- stav nezměněn

Aritmetické instrukce

Mnemo kód	Doba výkonu [μs] (Délka kódu [B])						Význam instrukce
	Z		#		A		
	byte usint	word uint	byte usint	word uint	byte usint	word uint	
ADD	-	99 (3)	-	88 (3)	-	88 (1)	Sčítání s přenosem
SUB	-	101 (3)	-	90 (3)	-	90 (1)	Odčítání s přenosem
MUL	59 (3)	-	50 (3)	-	51 (1)	-	Násobení
DIV	74 (3)	-	65 (3)	-	66 (1)	-	Dělení
INR	-	-	-	-	-	79 (1)	Inkrementace (+ 1)
DCR	-	-	-	-	-	79 (1)	Dekrementace (- 1)
EQ	-	103 (3)	-	92 (3)	-	92 (1)	Porovnání (rovnost)
LT	-	103 (3)	-	92 (3)	-	92 (1)	Porovnání (menší než)
GT	-	104 (3)	-	93 (3)	-	93 (1)	Porovnání (větší než)
BIN	-	-	-	-	-	80 (1)	Převod čísla do binárního formátu
BCD	-	-	-	-	-	226 (1)	Převod čísla do BCD

Operace se zásobníky

Mnemo kód	Doba výkonu [μs] (Délka kódu [B])	Význam instrukce
POP n	42 + 7n (3)	Posun (rotace) zásobníku zpět o n úrovní
NXT	375 (3)	Aktivace následujícího zásobníku v řadě
PRV	375 (3)	Aktivace předcházejícího zásobníku v řadě
CHG	356 (3)	Aktivace zvoleného zásobníku bez zálohování S0 a S1
CHGS	373 (3)	Aktivace zvoleného zásobníku se zálohováním S0 a S1

Instrukce skoků a volání

Mnemo kód	Doba výkonu [μs] (Délka kódu [B])		Význam instrukce
	průchod	skok	
JMP	-	114 (3)	Nepodmíněný skok
JMD	42	120 (3)	Skok podmíněný nenulovostí výsledku
JMC	42	120 (3)	Skok podmíněný nulovostí výsledku
JMI	-	115 (1)	Skok na nepřímý cíl
CAL	-	138 (3)	Nepodmíněné volání podprogramu
CAD	42	144 (3)	Volání podprogramu podmíněné nenulovostí výsledku
CAC	42	144 (3)	Volání podprogramu podmíněné nulovostí výsledku
CAI	-	137 (1)	Volání podprogramu nepřímého cíle
RET	-	48 (1)	Nepodmíněný návrat z podprogramu
RED	40	54 (1)	Návrat z podprogramu podmíněný nenulovostí výsledku
REC	40	54 (1)	Návrat z podprogramu podmíněný nulovostí výsledku
L	36 (3)	-	Návěští n (cíl skoků a volání)

Organizační instrukce

Mnemo kód	Doba výkonu [μs] (Délka kódu [B])					Význam instrukce
	průchod	skok	P41 - P49	P50 - P57	P62 - P64	
P	151 (3)	-	76	73	143	Začátek procesu
E	-	54 (3)	229	77	54	Nepodmíněný konec procesu
ED	42 (1)	65	240	88	65	Konec procesu při nenulovém výsledku
EC	42 (1)	65	240	88	65	Konec procesu při nulovém výsledku
EOC	-	35 (1)	-	-	-	Konec cyklu
NOP	36 (3)	-	-	-	-	Prázdná operace
BP	-	189 (3)	-	-	-	Ladící bod

Tabulkové instrukce

Mnemo kód	Doba výkonu [μ s] (Délka kódu [B])						Význam instrukce
	Z			T			
	bool	byte usint sint	word uint int	bool	byte usint sint	word uint int	
LTB	136 (3)	118 (3)	131 (3)	258 (3)	225 (3)	250 (3)	Čtení položky z tabulky
WTB	151 (3)	126 (3)	140 (3)	-	-	-	Zápis položky do tabulky
LMS	-	-	-	-	-	248 (3)	Čtení položky sekvenčně
WMS	-	-	-	-	-	362 (3)	Zápis položky sekvenčně
FTB	-	68 (3) +49	75 (3) +55	-	182 (3) +39	203 (3) +46	Hledání položky v tabulce - čas. přírážka na 1 prohledávanou pol.
FTM	-	66 (3) +61	66 (3) +90	-	192 (3) +51	192 (3) +80	Hledání části položky v tabulce - čas. přírážka na 1 prohledávanou pol.
FTS	-	71 (3) +51	64 (3) +74	-	184 (3) +41	191 (3) +64	Zařazení položky podle tabulky - čas. přírážka na 1 prohledávanou pol.

Blokové operace

Mnemo kód	Doba výkonu [μ s] (Délka kódu [B])		Význam instrukce
	Z	T	
SRC	99 (3)	195 (3)	Specifikace zdroje dat pro přesun
MOV	188 (3) +17	392 (3) +28	Přesun bloku dat - časová přírážka na 1 položku bloku
FIL	69 (3) +48	-	Naplnění bloku konstantou - časová přírážka na 1 položku bloku

PŘÍLOHA P V: VÝVOJOVÝ DIAGRAM PRO PLC SAIA PCD2 V PROSTŘEDÍ GRAFTEC

