

MASARYKOVA UNIVERZITA
FAKULTA INFORMATIKY



Řídicí systém vysokoteplotní tavné pece

Bakalářská práce

Petr Raška

Brno, 2011

Prohlášení

Prohlašuji, že tato práce je mým původním autorským dílem, které jsem vypracoval samostatně. Všechny zdroje, prameny a literaturu, které jsem při vypracování používal nebo z nich čerpal, v práci řádně cituji s uvedením úplného odkazu na příslušný zdroj.

.....

Petr Raška

Poděkování

Rád bych poděkoval vedoucímu bakalářské práce prof. Ing. Václavu Přenosilovi, CSc. za odborné vedení při práci, zprostředkování hardwaru a psychickou podporu v těžkých situacích vývoje aplikace. Dále Ing. Zbyňku Burešovi, Ph.D., kterému vděčím za pomoc při problémech, na které jsem narazil a za poskytnutí kontaktů na pracovníky slévárny.

Shrnutí

Tato bakalářská práce se zabývá analýzou stavu vysokoteplotní tavné pece pro vytavování vosku a průzkumem současných způsobů regulace. Dále jsem se v práci věnoval návrhu a sestavení řídicího systému za použití programovatelného logického automatu s důrazem na funkčnost a možnost pozdějšího rozšíření systému pro řízení více pecí. Pro realizaci řídicího systému byl použit automat PP45 firmy B&R spolu s vývojovým nástrojem Automation Studio. Program je psaný v programovacím jazyce C s využitím funkcí, které vývojový nástroj poskytuje formou knihoven.

Klíčová slova

Řídicí systém, regulace teploty, PLC, tavná pec, PID regulace.

Obsah

1	Úvod.....	2
2	Analýza stávajícího stavu.....	3
2.1	Metoda přesného lití.....	3
2.2	PID regulace.....	4
2.3	Požadavky	6
3	Návrh řídicího systému	8
3.1	PLC	8
3.2	Systém X20	10
3.3	Limitní jednotka	13
4	Realizace	14
4.1	Použitý hardware.....	15
4.2	Programování	19
4.3	Vyhodnocení výsledků.....	30
5	Závěr	31
	Použité informační zdroje	32
	Seznam příloh	33

1 Úvod

Pec se nalézá ve slévárenské dílně firmy MSR Engines. Tato firma se specializuje především na vývoj a výrobu nízko objemových závodních i klasických spalovacích motorů. Nabízí řešení výroby odlitků metodu přesného lití hliníku¹.

Cílem mé práce bylo analyzovat stav vysokoteplotní tavné pece pro vytavování vosku a pro tuto pec poté navrhnout a sestavit vhodný řídicí systém tak, aby vyhovoval aktuálním technologickým požadavkům provozu a požadavkům pracovníků slévárny. Mým úkolem dále bylo seznámit se s možnostmi PLC² a vývojového prostředí, navrhnout vhodné ovládací a senzorické prvky pece a k těmto prvkům navrhnout vhodné připojení. To všechno s důrazem na možnost dalšího rozšíření. Realizovanou část systému jsem otestoval nejprve pomocí programového simulátoru pece a nakonec i na testovacím zařízení, kde jsem si ověřil její vlastnosti.

První část práce je teoretická a je věnována současným metodám regulace teploty, způsobu výroby přesných forem pro metodu přesného lití, PID³ regulačnímu algoritmu, analýze požadavků na funkci a bezpečnosti nového systému. Další část je zaměřená na návrh nového řídicího systému s popisem vlastností zvoleného PLC a jeho periférií. Na návrh navazuje část popisující implementaci, která je rozdělená na tři části:

- 1) Použité komponenty – jejich vlastnosti a zapojení
- 2) Vývoj programu – použité nástroje, struktura programu a knihovny
- 3) Výsledky – zhodnocení vlastností vytvořeného systému

Na závěr jsem uvedl možnosti dalšího rozšíření systému.

¹ Více o firmě na stránkách <<http://www.msrengines.eu>>

² Programmable Logic Controller – programovatelný logický automat

³ Proporcionálně integračně derivační metoda řízení

2 Analýza stávajícího stavu

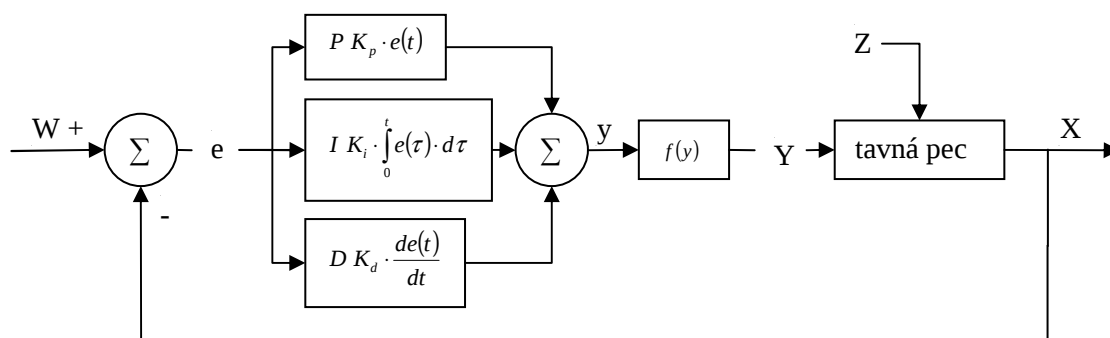
Stávající tavné pece ve slévárně jsou řízeny pomocí kompaktních teplotních průběhových regulátorů, které k regulaci teploty používají P, PI, PD, nebo PID regulační algoritmy. V kapitole 2.2 PID regulace se budu věnovat jednotlivým algoritmům a nastavení jejich parametrů. Teplotní průběhové regulátory poskytují obvykle možnost předem naprogramovat průběh teploty v 8 až 16 časových úsecích. Vyznačují se nízkou cenou, vysokou spolehlivostí a po dobrém nastavení parametrů, potřebných pro regulační algoritmy, také vysokou přesností měření tedy i samotné regulace. Cena regulátoru, který by splňoval minimální potřeby většiny technologických procedur výroby forem pro metodu přesného lití, se pohybuje již od 10 000,- Kč. Nastavení parametrů regulačních algoritmů není snadné, a proto je vhodné pořídit důmyslnější regulátor s automatickou kalibrací. I tak se za všech podmínek nemůžeme spolehnout pouze na automatickou kalibraci, protože podmínky v okolí pece se mohou i při průchodu programu měnit. Tavná pec, pro kterou jsem měl za úkol vytvořit řídicí systém, je vybavena ručně ovládanou stropní klapkou. Při uzavření nebo otevření klapky v průběhu spuštěné procedury se do značné míry změní závislost teploty uvnitř pece na ohřevu. Při použití levného řešení systému pomocí kompaktního průběhového regulátoru se obvykle automatická kalibrace parametrů PID regulace musí spustit před prvním programem regulace. Výrobní procesy často vyžadují otevírání klapky v průběhu programu. Přesnost řízení je přímo úměrná přesnosti nastavení parametrů. Změna parametrů v průběhu regulace je obvykle možná jen ručním zadáním, nebo u některých dražších modelů regulátoru složitějším způsobem automatického ladění parametrů v průběhu regulace. Všeobecně v nabídce kompaktních regulátorů teploty platí, že pro většinu nadstandardních rozšíření, je nutné investovat do složitějšího modelu, nebo přidat k systému další komponenty. Informace, které lze získat o stavu pece v průběhu výrobní procedury jsou velice omezené. Obvykle může obsluha zjistit pouze aktuální teplotu v peci a u dražších modelů i informace o pozici v programu. Neposkytují možnost zobrazovat časové průběhy teploty, nebo zaznamenávat různé události.

2.1 Metoda přesného lití

Metoda přesného lití metodou výtavných vosků se používá pro výrobu odlitků přesného tvaru z různých druhů slitin a oceli. Výsledné odlitky se vyznačují vysokou kvalitou povrchu a přesností tvaru. Výroba je technologicky náročná, ale lze touto metodou vyrobit i odlitky takových tvarů, které by byly jinými metodami jen těžko uskutečnitelné. Proces výroby začíná vytvořením tzv. matečního modelu podle technických výkresů. Ten se obvykle vyrábí z oceli obráběním, nebo jinou technologií. Pomocí matečního modelu se z vhodné slitiny vyrobí přesná mateční forma. V této formě poté odlijeme potřebné množství odlitků z voskové směsi, která se rozehřátá do tekutého skupenství pod tlakem vstříkne do formy. Voskové odlitky se poté opakovaně střídavě namáčí ve speciální keramické hmotě a vysoušejí. Kolem voskových odlitků se vytvoří keramický obal, který se nazývá skořepina. Nyní je nutné odstranit ze skořepiny vosk. Voskové odlitky se ve vysokoteplotní tavné peci vytaví nebo vypálí z keramické skořepiny,

která se dále ještě žihá a pročišťuje. V této fázi lze skořepinu použít k lití konečného výrobku. Po vychladnutí přesného odlitku se skořepina rozbije, z výsledného odlitku se odstraní vtokové kanálky a pozůstatky skořepiny z povrchu. Vznikne přesný odlitek, který se posílá dále na defektoskopii. [3]

2.2 PID regulace



Obrázek 1 Blokové schéma začlenění PID regulátoru do regulované soustavy
inspirováno schématem na webu [7]

PID regulátory jsou nejčastěji používané regulátory teploty pro aplikace, kde je důležitá přesnost a rychlost reakce na změny požadované teploty W a vnějších vlivů Z . Rozdíl mezi požadovanou teplotou a naměřenou teplotou X vstupuje do jednotlivých částí regulátoru jako regulační (chybová) odchylka „ e “. Pro správnou funkci celého systému je klíčové správné nastavení konstanty zesílení proporcionální složky K_p , a také integrační K_i a derivační K_d konstanty. [1]

Proportionální část regulátoru funguje jako lineární zesilovač. Kde konstanta K_p odpovídá zesílení. Změna regulační odchylky se neprodleně projeví na výstupu. Pokud bychom používali pouze P složku regulátoru, nebude z důsledku tepelných ztrát nikdy teplota na výstupu doladěna na požadovanou hodnotu. V určitém bodě blízko cílové teploty dojde k situaci, kdy tepelné ztráty budou větší, než tepelný zisk z topných spirál při malé střídě pulzně šířkové modulační akční veličiny Y . Výsledná teplota bude tedy nižší než teplota požadovaná. P regulátor má sice rychlou reakci na změny požadované teploty W a změny vnějších vlivů, ale samotný pro regulaci teploty v tavné peci nestačí. Pro odstranění problému s ustálením teploty na požadované teplotě P regulátoru se používá integrační část. Spojením proporcionální a integrační složky vzniká takzvaný PI regulátor, který je již v některých řídicích systémech použitelný. Při dlouhodobě kladné regulační odchylce integrační složka postupně zvyšuje svůj výstup a v součtu s proporcionální složkou pomáhá k doladění výstupní teploty na požadovanou hodnotu. Nevýhodou přidání integrační složky je pomalá reakce regulátoru na všechny změny a při nastavení vysoké konstanty zesílení může regulátor zcela ztratit stabilitu a rozkmitat se. Volba nízkého zesílení integračního regulátoru sníží velikost překmitů a stabilitu, ale také zpomalí celý proces regulace a naopak. PI regulátory lze použít pro systémy, ve kterých nedochází ke skokovým změnám požadované hodnoty regulované veličiny. Pro zrychlení reakce na změny

požadované teploty se přidává derivační složka D, u které je akční veličina přímo úměrná derivaci regulační odchylky. To že systém dokáže rychleji reagovat na změny, redukuje negativní vlastnosti integrační složky snížením překmitů, jelikož integrační složka nabývá na síle pokaždé, když je skutečná teplota v peci po dlouhou dobu nižší, nebo naopak vyšší než požadovaná teplota. Nevýhodou derivační složky je fakt, že zesiluje šum. Je tedy vhodné, aby bylo měření teploty na výstupu co nejpresnější, odolné proti rušení a filtrované na výchyly zapříčiněné atmosférickými vlivy. Tím se redukuje negativní vlivy D složky a zachová se její pozitivní vliv na zrychlení reakce a zvýšení stability regulačního procesu. Sloučením akčních veličin jednotlivých částí vznikne výsledný PID regulátor. [1,2]

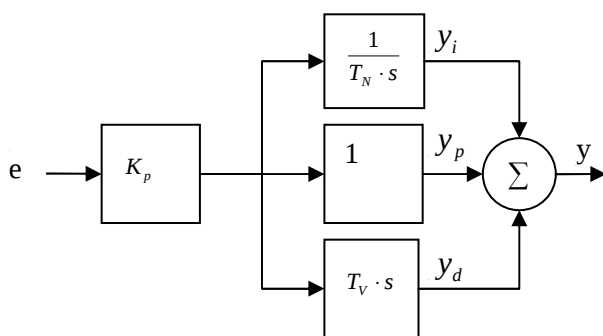
Rovnice 1 Vztah výstupní veličiny PID regulátoru v časovém kroku t

$$X(t) = K_p \cdot e(t) + K_i \cdot \int_0^t e(\tau) \cdot d\tau + K_d \frac{de(t)}{dt} + X_0$$

Rovnice 2 Přenosová funkce ideálního PID regulátoru po převedení pomocí Laplaceovy transformace

$$R(s) = K_p + \frac{K_i}{s} + K_d \cdot s = K_p \left(1 + \frac{1}{T_N \cdot s} + s \cdot T_v \right)$$

Ideální PID regulátor si poté můžeme znázornit v novém blokovém schématu:



Obrázek 2 Blokové schéma ideálního PID regulátoru

V praxi se nejčastěji používá vyjádření pomocí parametrů K_p , T_N a T_v . Konstanta K_p vyjadřuje lineární zesílení regulační odchylky. T_N je časová konstanta vyjádřená v sekundách, která definuje časový úsek, po kterém se integrační část regulátoru při konstantní regulační odchylce projeví změnou akční veličiny o stejnou hodnotu, jako produkuje proporcionální část regulátoru. Čím menší T_N použijeme, tím vyšší vliv bude mít integrační složka regulátoru. Naopak je tomu u derivační časové konstanty T_v , která definuje čas v sekundách, po kterém bude proporcionální část regulátoru generovat stejnou hodnotu akční veličiny jako derivační složka, poté co se začne

regulační odchylka začínající v nule konstantně zvyšovat. Jelikož není možné realizovat ideální derivační složku regulátoru, používá se v praxi filtrovaná derivační složka. Zavedeme si tedy další parametr T_F , který vyjadřuje filtrační periodu. Aby PID regulátor pracoval správně, je potřeba zajistit, aby $T_V > T_F \gg T_{CPU}$, přičemž T_{CPU} je perioda řízení. Většina kompaktních regulátorů toto zajišťuje tak, že filtrační časovou konstantu nastavuje automaticky na násobek filtrační periody, například $T_F = 1,4 \cdot T_{CPU}$ a buď nedovolí uživateli snížit derivační periodu na nižší hodnotu, nebo v případě nevhodně zvoleného T_V buď vyřadí derivační složku z regulačního procesu, nebo opět nastaví T_V na nějakou implicitní hodnotu. [1,8]

Rovnice 3 Přenosová funkce reálného PID regulátoru [8]

$$R(s) = K_p \left(1 + \frac{1}{T_N \cdot s} + \frac{s \cdot T_V}{1 + s \cdot T_F} \right)$$

2.3 Požadavky

Řídicí systém vysokoteplotní tavné pece, který jsem měl za úkol navrhnout, by měl splňovat vlastnosti běžně dostupných kompaktních průběhových PID regulátorů, kterými jsou již řízeny ostatní pece ve slévárně. Je nutné, aby systém nabízel možnost rozšíření o funkce, které zjednoduší každodenní rutiny. Tyto funkce budou poskytovat přehledné ovládání a položí tak základní kamen pro budoucí rozšiřování. V budoucnu by ve slévárně měla proběhnout další vlna automatizování výroby a mělo by se počítat s možností rozšířit řídicí systém pro řízení dalších pecí, které již jsou v provozu, ale jejich řízení není dost přizpůsobivé pro neustále se zdokonalující technologický proces výroby. Nový řídicí systém by tedy měl být dostatečně rozšiřitelný a výkonný, aby zvládal regulovat několik pecí rozmístěných po výrobní hale.

Ovládání by bylo vhodné zjednodušit a lépe graficky znázornit. Zaměstnancům by mělo být po krátkém pohledu na ovládací panel jasné, v jakém stavu jsou pece, nebo kolik času zbývá do dokončení výrobní procedury, aby se mohli věnovat přípravě na další práci s produkty a polotovary, aniž by museli příliš často provádět kontrolu.

Některé výrobní procedury se opakují, proto by měla existovat možnost tyto procedury ukládat do paměti řídicího systému a snadno opakovaně používat. Spouštění každodenních procedur je vhodné automatizovat a dát zaměstnancům možnost nastavit odklad pro spuštění naprogramované procedury, aby se například ráno po zahájení směny nemuselo čekat několik hodin, než se pec připraví. Někdy je naopak potřeba regulovat teplotu v peci ručně. Ovládání tedy musí být maximálně flexibilní.

V metodě přesného lití je důležitá vysoká přesnost řízení teploty, proto musí být regulace teploty na úrovni jednoúčelově vyrobených kompaktních regulátorů včetně automatické kalibrace parametrů pece.

Pokud by se během provozu objevily problémy s kvalitou výstupních produktů v důsledku špatné regulace teploty, ke které by docházelo kvůli špatnému nastavení parametrů PID regulace, bylo by nutné použít jiný mnohem dražší kompaktní průběhový regulátor. Takovým problémům lze předcházet realizací systému pomocí PLC (programovatelného logického automatu).

2.3.1 Automatizace/funkce

Použitím programovatelného logického automatu se vyhneme složitému klíčování pro budoucí rozšíření systému. Při použití na průmyslové podmínky připraveného PLC, lze splnit již zmíněné požadavky na přesnou regulaci včetně automatické kalibrace parametrů PID regulace. Dále toto řešení nabízí možnost optimálně naprogramovat uživatelské rozhraní a práci s daty na velkokapacitních datových zařízeních. Některé PLC poskytují možnost napojení na již existující systémy a jejich komunikační rozhraní. Bylo by možné vybrat takový automat, se kterým lze komunikovat přímo s počítači v kanceláři zaměstnanců vedle výrobní haly.

Možností rozšíření řídicího systému založeného na PLC jsou doslova obrovské. Je například možné naprogramovat řízení čtyř pecí s velkým výkonem, přičemž energetické vedení a pojistky v elektrickém rozvaděči by nezvládaly sepnutí ani tří pecí najednou. Programově můžeme zařídit, aby v případě, že nebudeme chtít využívat všechny pece na maximální výkon, bylo možné střídavě připojovat pece k síti tak, aby součet příkonů právě připojených pecí nepřekročil maximální možný výkon odebíraný ze sítě. Přičemž by se zachovala přesnost řízení a zvýšila se rychlost výroby ve slévárně. Systém řešený pomocí PLC by mohl být rozšířen i o ukládání provozních logů a záznamů kontrolních a chybových událostí.

2.3.2 Bezpečnost a ochrana

Kromě řízení teploty v peci musí být zajištěna i ochrana zaměstnanců před úrazem přerušením dodávky elektrického proudu do topných spirál při otevření pece. Jelikož na topných spirálách nemůže být izolace a zaměstnanci s materiálem manipulují pomocí vodivých tyčí, jsou v sázce životy zaměstnanců, a tudíž musí být přerušení zajištěno za jakékoli situace. Není vhodné spoléhat na odpojení na základě signálu z PLC. Elektronické obvody určené do průmyslu obecně nejsou natolik spolehlivé, aby se daly pro něco takového použít. PLC se může z důvodu špatného naprogramování nebo nepředvídané chyby zacyklit nebo selhat. Mechanismus přerušení napájení při otevření dveří musí být spolehlivý, a proto by měl být založený na spínacím prvku, například stykači, který v případě přerušení ovládacího okruhu vedeného bezpečnostním spínacím u dveří pece přeruší silové obvody topných spirál.

Další důležitou částí systému je ochrana topných spirál před přehřátím. Je potřeba zajistit měření teploty v blízkosti topných spirál a v případě překročení bezpečné meze odstavit elektrické napájení spirál dokud jejich teplota neklesne. K tomuto můžeme, podobně jako tomu je na již zprovozněných pecích, použít limitní jednotku.

3 Návrh řídicího systému

Z velkého množství firem zabývajících se automatizací jsme se nakonec dohodli na cenově dostupném a dobře škálovatelném řešení společnosti B&R.

Tato mezinárodní společnost, kterou roku 1983 založil Erwin Bernecker a Josef Rainer, je jak podle informací, které uvádějí na svých internetových stránkách průmyslovým lídrem v automatizaci průmyslových systémů. Firma nabízí technologická řešení pro textilní, balicí, dřevozpracující, elektrotechnický, plastikářský, kovozpracující a tiskový průmysl. Neustále přichází s inovačními nápady a díky dobrému citu k tržnímu vývoji se drží ve špičce. Působí ve více než 55 zemích po celém světě a má svou pobočku i v České republice. B&R neustále roste a upevňuje svou vedoucí pozici na mezinárodním trhu. [6]

Ze široké nabídky automatizačních systémů nabízených firmou B&R je dle mého názoru optimální řízení pomocí kombinace PLC v podobě Power panelu PP45 (úplné označení 4PP045.0571-062) a vstup výstupních modulů umístěných v blízkosti pece v řídicí skříni, které se k panelu připojí přes sériové rozhraní X2X. Power panel bude připojen k firemní počítačové síti, aby bylo možné po síti zálohovat programy tavení a provozní nebo chybové log soubory. Dále je potřeba zajistit bezpečné odpojení napájení topných spirál pece v případě selhání některého z elektronických obvodů, nebo například při odpojení řídicího systému od napájení. Toto je potřeba z důvodu ochrany zaměstnanců před úrazem elektrickým proudem, a také pro ochranu samotné pece před přehřátím topných spirál. Z těchto důvodů byly navržený ochranné obvody, které zajistí bezpečný chod pece v provozu slévárny a budou později zapojeny taktéž ve skříni řídicího systému v blízkosti pece.

3.1 PLC

Výpočetní výkon Power panelu PP45 poskytuje dostatečně výkonný x86 procesor pracující s frekvencí 100MHz. Procesor ELAN SC520 100 MHz je kompatibilní s procesory firmy Intel typu i386. V porovnání s dnešními počítači, nebo novějšími modely Power panelů firmy B&R, se sice může zdát nevýkonný, ale protože PLC pracuje s jednoduchým operačním systémem specializovaným pro naše potřeby, je 100MHz naprosto dostačující a zvládne regulovat i více než jen jednu tavnou pec. [4]

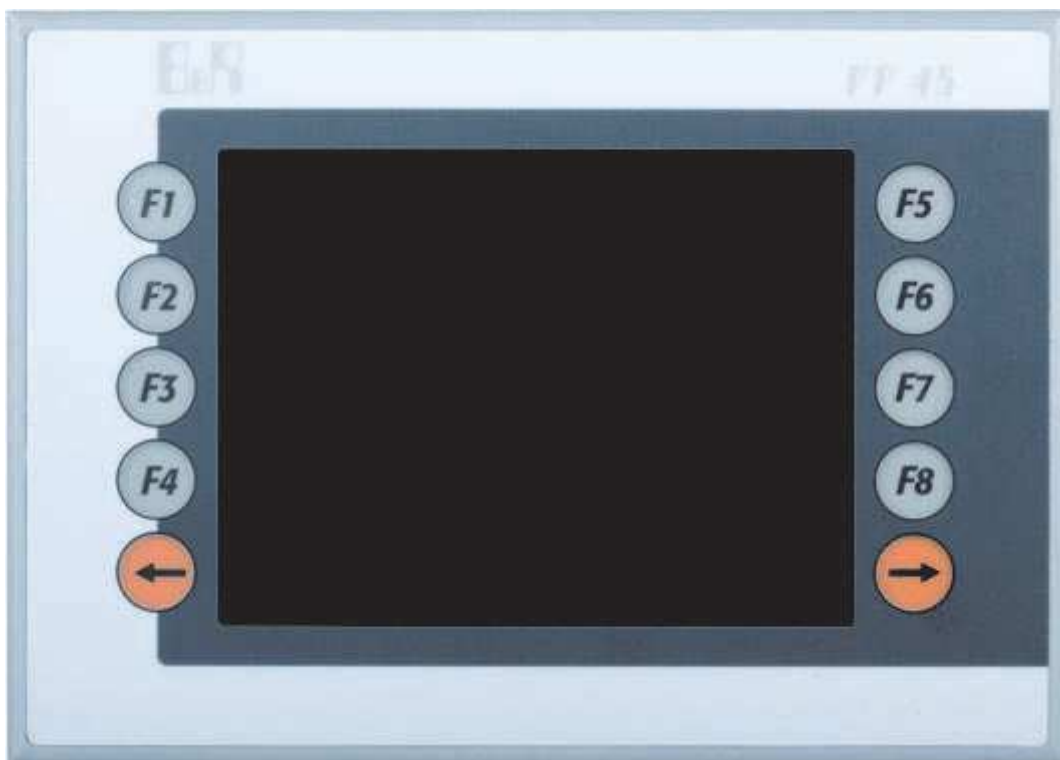
Operační paměť je ve dvou blocích. Jeden blok je paměť typu DRAM o velikosti 64MB, druhý blok typu SRAM zálohovaný baterií má kapacitu 32kB. SRAM se dá použít k uchování proměnných, jejichž obsah je důležité uchovávat i při restartu, nebo při výpadku proudu. Aplikační paměť PLC je řešena pomocí odnímatelné paměťové karty typu Compact Flash, která se objednává samostatně, její kapacita byla zvolena dle potřeb slévárny. Podle manuálu je panel kompatibilní s kartami o velikost 64 MB až 8 GB. Poskytuje tedy dostatečný prostor k uložení grafických objektů pro vizualizaci, nebo třeba pro ukládání provozních logů. Na kartě se nachází i operační systém, který ovšem zabírá jen několik megabytů. Po zapojení a spuštění PLC se

provádí zavádění (booting) systému následujícím způsobem. Nejprve se paměťové bloky operačního systému a uživatelských aplikací nakopírují do DRAM. Poté se vytvoří kopie dat z paměti SRAM do DRAM. Dále se pracuje jen s touto kopií, protože je práce s daty v DRAM mnohem rychlejší jelikož neprobíhá přes sběrnici, kterou sdílí i ostatní zařízení. Z tohoto důvodu používáme DRAM jako vyrovnávací paměť s tím, že synchronizace dat se provádí při vypínání systému nebo při výpadku napájení. Díky rychlé detekci poklesu vstupního napětí pod 15V při výpadku napájení, ještě může chvíli systém pracovat, přičemž je napájen nabitými kondenzátory na základní desce. Po příchodu signálu o poklesu napětí se ihned spustí NMI⁴ procedura, která obsah permanentních proměnných v DRAM překopíruje zpět do SRAM, kde zůstane jejich obsah zachován i po vybití kondenzátorů, jelikož je zálohován z baterie. Podrobněji se o využití jednotlivých pamětí budu věnovat později v realizační části práce v sekci 4.2.2.

PP45 je vybaven dotykovým QVGA⁵ barevným LCD displejem s úhlopříčkou 5,7 palců. Dotykový displej je vyroben rezistivní (odporovou) technologií, a proto se vyznačuje poměrně vysokou odolností a také je ho možno stisknout prakticky jakýmkoliv běžným předmětem. Displej umožňuje zobrazovat 256 barev, což je pro náš účel naprosto dostatečné. Umožnil mi barevně odlišit různé časové průběhy a stavy pece. Rozlišení displeje 320 × 240 obrazových bodů je ovšem poměrně omezující. Musel jsem tedy vynaložit větší úsilí grafickému návrhu ovládání. Návrháři panelu byli dostatečně prozíraví, aby nespolehali pouze na dotykový displej a věděli, že v takovém rozlišení se těžko zobrazují časové průběhy v případě, že by do obrazu musely navíc zasahovat i ovládací prvky. V okolí panelu tedy rozmístili 10 programovatelných tlačítek s výměnnými štítky pro popis funkce. Využití dotykového displeje umožňuje jednoduchou interakci obsluhy pece s řídicím systémem.

⁴ Non-maskable interrupt – žádost o přerušování činnosti procesoru, které nelze blokovat.

⁵ Quarter Video Graphics Array – označení pro displeje o rozlišení 320 × 240 obrazových bodů



**Obrázek 3 Čelní pohled na Power panel PP45,
obrázek pochází z manuálových stránek [4]**

PLC se připojuje k lokální počítačové síti zabudovaným RJ-45 konektorem. Zvládá běžné ethernetové připojení do 100 Mb/s. Po nahrání operačního systému na paměťovou kartu lze PLC programovat po síti. Po nastavení FTP (file transfer protocol) je možno jednoduše spravovat soubory uložené v aplikační paměti. To se mi výborně hodí, jelikož pak bude možné snadně spravovat na přístroji obsluhou navržené programy tavení a archivovat je například na počítači. Můžeme také monitorovat činnost PLC pomocí pokročilé systémové diagnostiky. Výstup ze systémové diagnostiky je dostupný přes http rozhraní. Přes webové rozhraní můžeme získat rychlý přehled o celkovém stavu PLC jako například teplota procesoru a jeho vytížení včetně časového průběhu využití, informace o softwarové a hardwarové konfiguraci, stavu baterie, záznamy chybových, varovných a informačních událostí a mnoho dalších informací. Záznamy událostí z webového rozhraní vystačí pro diagnostiku správné funkce jedné pece. Dozvíme se informace o pádu aplikace a ve většině případů odhalíme i příčinu problému. Plánovaná rozšíření systému budou už pravděpodobně vyžadovat více informací pro detekci příčiny chyby.

3.2 Systém X20

Kromě připojení k LAN je v standardní výbavě PP 45 sériové rozhraní X2X. B&R navrhlo a sestavilo velké množství vzdálených vstup/výstupních modulů, nebo dalších CPU (centrální výpočetní jednotka), které se k PLC připojují pomocí X2X na vzdálenost až 100m. Celý systém modulů (nazýváme X20) se sestavuje do jednoho nebo více oddělených bloků (ostrůvků), které

si může návrhář poskládat dle vlastních potřeb. Pomocí CPU v bloku X20 můžeme decentralizovat řízení a rozložit výpočetní operace. Pro plánované rozšíření systému pro řízení více pecí stojí za zmínku možnost realizovat komunikační rozhraní ostrůvku použitím speciálního komunikačního modulu, který nám umožní připojit se k centrálnímu PLC nebo počítačovému serveru běžnou počítačovou sítí. Je možno zvolit takovou síťovou topologii, která zajistí komunikaci i při ztrátě jedné ze síťových cest. Například zapojením několika ostrůvků do kruhové topologie sítě. Pro řízení jediné pece vystačí jeden ostrůvek, připojený X2X linkou. Tímto způsobem vznikne cenově dostupné řešení přizpůsobitelné pro řadu průmyslových řídicích aplikací, kde pro řízení postačují vstupy z měření analogových veličin časovou periodou nejméně 2 μ s. [5] Pro řízení tavných pecí, kde trvá i minuty, než se podstatně projeví změna na vstupu do změny teploty, je taková doba odezvy zcela postačující.



**Obrázek 4 Moduly X20,
obrázek pochází z manuálových stránek [5]**

Systém modulů X20 obsahuje celou řadu modulů. Od programovatelných procesorů vhodných pro decentralizaci řízení, přes komunikační a napájecí X2X moduly, dále digitální a analogové vstupy a výstupy pro řízení a nakonec o nic méně důležité jsou moduly pro podporu jiných existujících systémů. Mezi podporované komunikační rozhraní a systémy patří kromě standardního Ethernetu také CAN open, Profi BUS, Device Net, Ethernet POWERLINK a jiné.

Kromě X20 procesorů a některých napájecích modulů, které jsou do šířky rozměrnější a vyrábí se nerozebíratelné, jsou ostatní moduly, jejichž ukázkou můžeme vidět na obrázku Obrázek 4 Moduly X204, složeny ze tří stavebních prvků.

3.2.1 Sběrníkový modul (Bus module)

Jedná se o prvek, který zajišťuje jednoduché upevnění modulu k DIN liště. Slouží jako propojení napájení sběrnice, datová sběrnice, uzemnění a připojení k napájení vstup výstupních modulů do celého bloku nebo jen do jeho části. Svou funkcí nahrazuje rack, což je systém pro montáž elektronických zařízení do rámu, který můžeme vidět nejčastěji u počítačových serverů, hudebních sestav atd. Často jsou v něm jednotlivé prvky umístěny na výsuvných kolejničích. A zajišťuje jednoduchou montáž a přehledné propojení jednotlivých zařízení.

Existují dvě základní varianty sběrníkového modulu, první (X20BM11) rozvádí napájení vstup výstupních modulů v obou směrech, druhý (X20BM01) je z levé strany izolován a slouží k připojení napájecích modulů. Díky druhému typu je možno nakonfigurovat galvanicky oddělení skupiny bloků pro rozložení zátěže na více samostatných zdrojů. První varianta je k dispozici buď pro napájení 24V, nebo pro speciální účely s 240V úpravou.

Některé elektronické moduly vydávají větší teplo a proto se například okolo modulů pro měření teploty na termočláncích, jejichž přesnost závisí i na teplotě měřicího modulu, často vynechává jeden nebo více bloků volného prostoru pro lepší cirkulaci vzduchu. S nejbližšími moduly jsou tedy fyzicky spojeny jen pomocí sběrníkových komponent.

Pokud v době sestavování víme, že se bude zvyšovat počet potřebných vstupů nebo výstupů, můžeme si nechat uvnitř bloku modulů místo pro další elektronické moduly, které mohou být i za chodu přidávány. Dokonce si můžeme naprogramovat chování PLC tak, aby na změnu počtu vstupů dovedlo dynamicky reagovat.

3.2.2 Elektronický modul (Electronic module)

Elektronický modul se vsazuje do sběrníkového modulu. Vstupy a výstupy potřebné k jeho funkci jsou připojovány skrz terminálový modul. Všechny moduly mají zabudovanou vnitřní diagnostiku, jejíž výstup je viditelný na přehledně uspořádaných symbolech s LED podsvícením, které jsou při volbě horizontální montáže umístěny na čelní straně v horní části bloku modulů. V případě problému s napájením, X2X komunikací, nebo chováním řízeného systému můžeme na viditelné elektronické části modulu odhalit příčinu mnoha problémů, které v podmínkách průmyslového provozu mohou nastat.

B&R nabízí širokou škálu X20 modulů pro řízení pece vzdáleně z PP45. Hodil se mi především přijímač X2X linky X20BR9300, dále modul digitálního 24V vstupu DI4371, digitálního 24V výstupu DO4322 a modul pro měření teploty na termočláncích AT2402. BR9300 slouží nejen jako komunikační rozhraní mezi X2X linkou a blokem X20 modulů, spolu s posílením

signálu komunikační linky, ale má v sobě zabudovaný i rozvod napájení pro vstup výstupní moduly. Nepotřeboval jsem tedy žádné další zdroj pro napájení 24V výstupů. Čtyři jmenované typy modulů vystačí pro realizaci funkce průběhového regulátoru teploty se spoustou nadstandardních vlastností systému. O zapojení a vlastnostech použitých modulů se ještě zmíním na straně 15 v sekci použitý hardware.

Kromě již zmíněných modulů X20 se vyrábějí ještě moduly s analogovým vstupem nebo výstupem napětí a proudu, nebo také moduly různých digitálních čítačů. Pro rozšíření systému o další vzdálené bloky X20 modulů se může použít vysílač X2X linky v podobě modulu X20BT9100. Tento modul můžeme zapojit do nejbližšího bloku a navázat spojení s přijímačem BR9300 v dalším bloku.

3.2.3 Terminálový modul (Terminal module)

Jedná se o konstrukční prvek, vyráběný ve dvou verzích, a sice 6 a 12 pinové. Montáž vodičů pro vstupy, výstupy, komunikační rozhraní a napájení je velice snadná a rychlá, díky tzv. push-in technologii. V případě poruchy elektronického modulu odklopíme terminálový modul i s připojenými vodiči, nahradíme elektronický modul za nový, terminálový modul vrátíme zpět, otestujeme připojení vodičů a oprava je hotova. Nad všemi vstupy je prostor pro popisný štítek, který si můžeme vytisknout na speciální tiskárně prodávané firmou B&R pro tento účel. Vstupy jsou také vybaveny měřicím bodem pro diagnostiku a testování jehlou měřicího přístroje. Není tedy nutné odpojovat jednotlivé vodiče. Kombinací vnitřní diagnostiky elektronických modulů, přehledných popisků, topologicky a funkcionálně dobře navrženém systému a díky důmyslnému mechanickému provedení je údržba rychlá, a proto i poměrně levná.

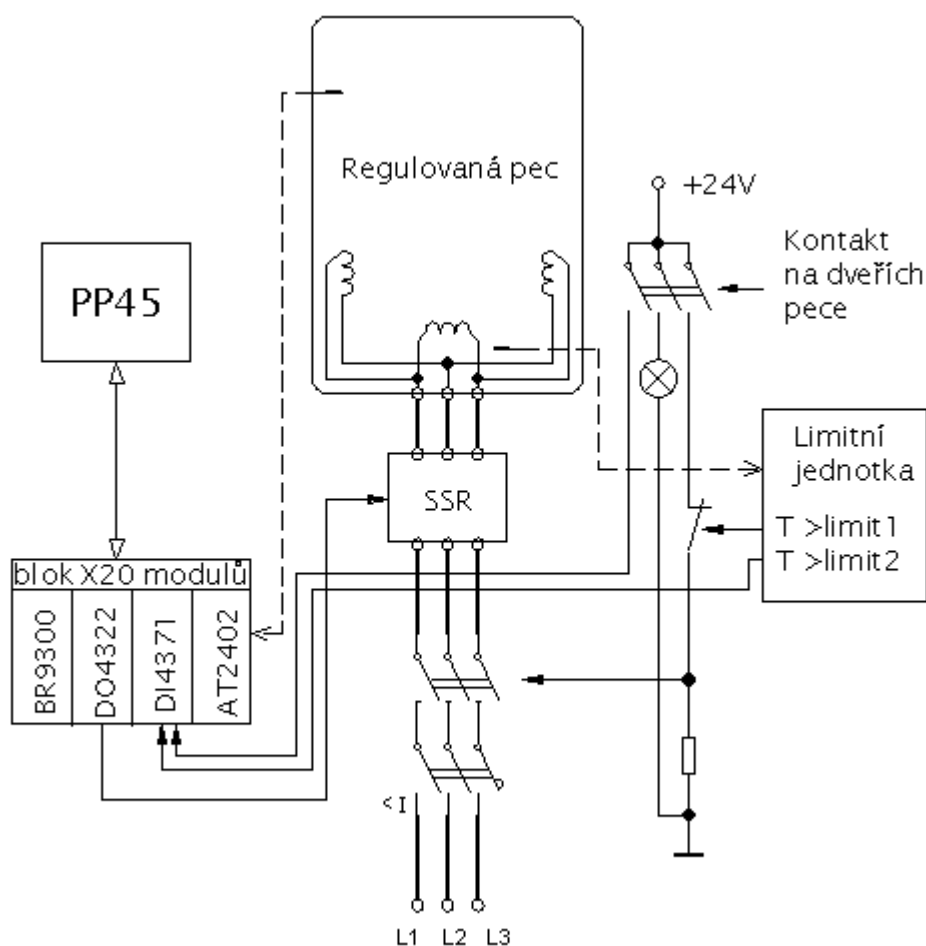
3.3 Limitní jednotka

Pro ochranu pece před sebezničením přehřátím je nutné měřit teplotu v blízkosti topných spirál. Toto zajistí limitní jednotka, která v případě, že teplota překročí maximální povolený limit, vyšle signál na spínací prvek silové části obvodu, který odpojí pec od zdroje napětí. Limitní jednotka je elektronické zařízení, které není nutné navrhovat, na trhu je dostupných mnoho hotových řešení, a proto se pouze zmíním o začlenění limitní jednotky do systému. Limitní jednotky obvykle umožňují nastavit dvě teplotní úrovně. Při překročení 1. úrovně se výstup připojený k digitálnímu vstupu PLC přepne do aktivní úrovně. Na to systém zareaguje a odpojí pec programově. Pokud ani toto opatření nepomůže z důvodu chyby řídicího systému nebo špatně navoleného programu a dojde k překročení 2. úrovně limitní jednotka aktivuje i 2. výstup, kterým přeruší napájení pece pomocí stykače nebo SSR⁶.

⁶ Solid State Relay – polovodičový spínací prvek, který je bezkontaktním nahrazením stykače.

4 Realizace

V průběhu práce se vyskytly určité interní problémy na straně zadavatele, firmy MSR Engines. Firma dostala novou zakázku, která je tak časově i finančně vytížila, že celá situace vyústila v nemožnost zapojit a otestovat navržený řídicí systém přímo v provozovně slévárny. Vymysleli jsme tedy náhradní řešení, že řídicí systém otestuji na softwarovém simulátoru pece. Pro otestování komunikačního protokolu a ověření vstupů a výstupů z PLC přes moduly X20 jsem si sestavil jednoduché testovací zařízení. Na testovacím zařízení jsem pracoval s nižšími teplotními rozsahy, a jelikož jsem neměl prostředky k vytvoření prostředí, kde bude změna teploty pomalá podobně jako u skutečné pece v provozu, nebyla výsledná přesnost regulace ideální, protože i krátké spuštění teplotního tělesa během jedné sekundy zvýšilo teplotu asi o 10 stupňů, to mi ale nebránilo v odzkoušení řídicího systému. Na softwarovém simulátoru se dala ověřit přesnost regulačního procesu a testovací zařízení sloužilo už jen k ověření správnosti napojení na skutečná čidla a ovládací prvky. Softwarovému simulátoru se budu věnovat ještě v kapitole 4.2.1.



Obrázek 5 Schéma zapojení řídicího systému

4.1 Použitý hardware

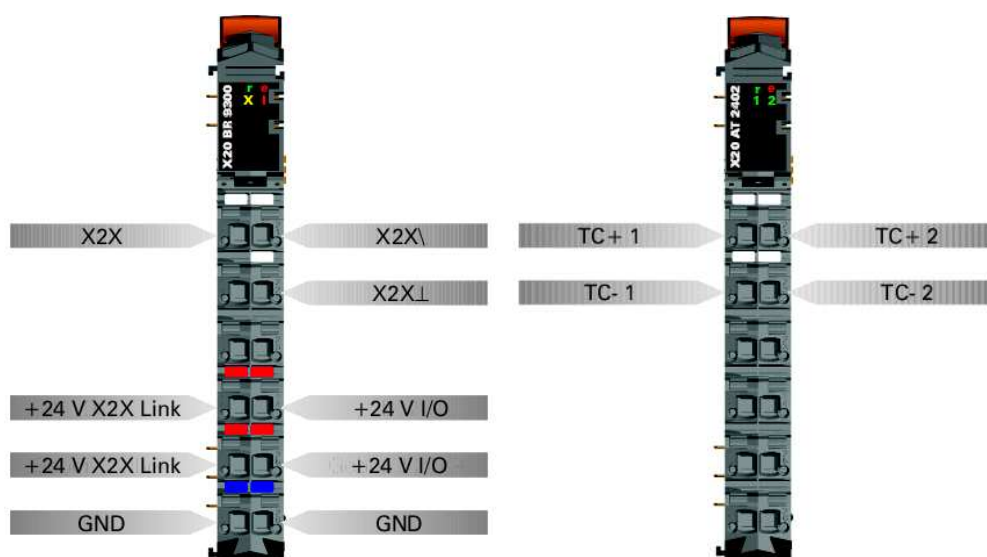
K měření teploty jsem použil modul X20AT2402. Tento modul je určen pro měření teploty pomocí termočlánků typu J, K, N nebo S.

Tabulka 1 Teplotní rozsahy podporovaných typů termočlánků [5]

typ termočlánku	možné rozsahy	maximální chyba měření
termočlánek J	-210°C to +1200°C	0,1 % z měřené hodnoty
termočlánek K	-270°C to +1372°C	0,11 % z měřené hodnoty
termočlánek N	-270°C to +1300°C	0,11 % z měřené hodnoty
termočlánek S	-50°C to +1768°C	0,17 % z měřené hodnoty

Použil jsem termočlánky typu K, které se v tomto odvětví průmyslu běžně používají. Termočlánky byly dodány konzultantem a dozvěděl jsem se o nich jen to, že jsou typu K a umí měřit teploty od 200°C přibližně až do teploty 700°C. Bohužel se mi nepodařilo zjistit výrobce ani konkrétní typ termočlánků. Omezený pracovní rozsah termočlánku není problém. Měření nižších teplot než 200°C je možné, i když jsou naměřené hodnoty nepřesné a k testování regulace nad 700°C nemáme v prostředí laboratoře potřebné vybavení ani prostory, kde by bylo možné bezpečně pracovat při tak vysokých teplotách. Při montáži do provozu si odpovědný zaměstnanec slévárny bude muset sám zvolit z velkého sortimentu termočlánků tak, aby pokryl rozsah teplot, ve kterých bude pec používána. Případná změna nastavení typu termočlánku například na typ S, je otázkou okamžiku. Testovací přípravek se skládal z jednofázového SSR, kterým jsem připojoval horkovzdušnou pistoli do elektrické sítě. Silová část systému lze snadno zapojit tak, aby byla dimenzovaná na proudy a celkový výkon, který je očekávaný v provozu. Řídicí systém použitý při testování bude tedy podobným způsobem aplikovatelný na tavnou pec v provozu.

Na rozdíl od testovacího zapojení bude potřeba použít třífázové SSR dimenzované na odpovídající proudy a zapojení třífázového spínacího stykače, který se sám mechanicky odpojí v případě, že limitní jednotka zaznamená překročení bezpečné teploty v blízkosti topných spirál a přestane napájet ovládací část stykače. Stejně tak i šířka průřezu napájecích vodičů topných spirál bude muset zvládat mnohem vyšší jmenovitý proud. Blok X20 modulů se uchycuje pomocí DIN lišty, takže jeho montáž nebude složitá.



Obrázek 6 Rozložení pinů modulů X20BR9300(vlevo) a X20AT2402, obrázek pochází z manuálových stránek [5]

Piny, se stejným označením jsou vnitřně propojeny. Nezáleží tedy na tom, který z pinů se využije.

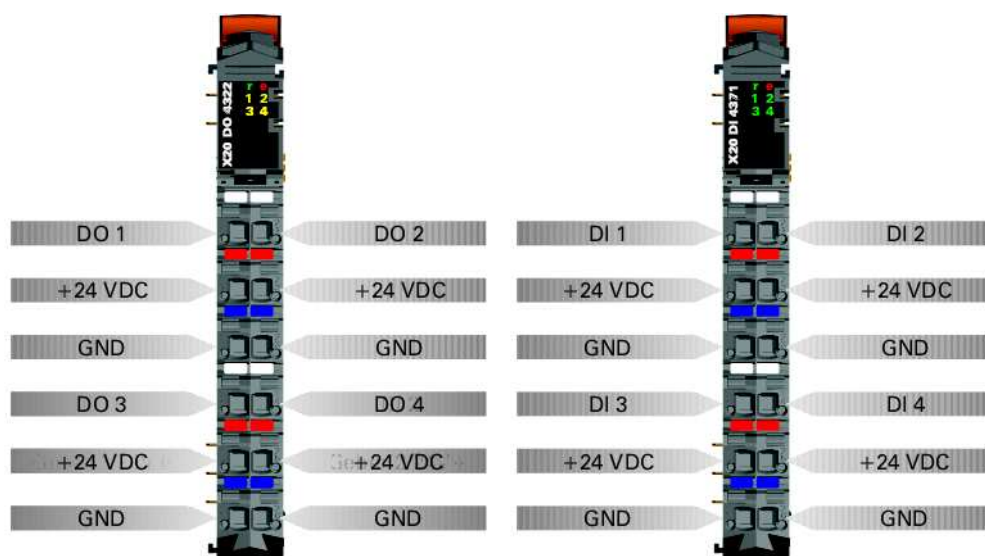
Modul X20BR9300 slouží jako komunikační rozhraní mezi PP 45 a blokem vstup výstupních modulů a zároveň jako prvek rozvádějící napájení do ostatních modulů v bloku. Je nezbytné jej zapojovat do sběrnice modulu X20BM01. Pokud bychom jej zapojili do X20BM11 nebude nám správně fungovat komunikace ani rozvod napájení ostatních modulů. Tuto skutečnost zmiňuji především proto, že jsem se sám dopustil omylu, jehož odhalení mi trvalo několik dní pročítání katalogových listů, manuálů výrobce a práce s osciloskopem. Nakonec jsem na příčinu nefunkční X2X komunikace přišel až ve spolupráci s technikem firmy B&R na místní pobočce.

Na obrázku 6 vpravo vidíme rozložení jednotlivých pinů. Bílou barvou označené piny se připojují k lince X2X. První pin označený jako X2X je datový kanál, druhý (X2X\) slouží podobně jako tomu je u standardizované asynchronní linky RS-485 jako invertovaný datový kanál. Pin umístěný na čtvrté pozici (X2X_) slouží k rozvodu elektrické země. Důvodem, proč se na rozdíl od RS-485, kde pro polo-duplexní přenos postačují dva vodiče, v sériové komunikaci X2X rozvádí i elektrická zem je ten, že elektronické obvody zajišťující datovou komunikaci a vstup-výstupní obvody jsou galvanicky odděleny. Díky tomu je PLC nebo jiný vysílač X2X, pokud použijeme i oddělené zdroje napájení X2X linky a vstup-výstupních modulů, chráněn před poškozením, které by mohlo vzniknout v případě poruchy jednoho z bloků X20. B&R nabízí i síťové kabely. Specifikacím X2X linky ale odpovídají i běžné kroucené dvojlinky používané v síť Ethernet. Pro testovací účely jsem použil nestíněný kabel UTP, pro konečné zapojení ve slévárně doporučuji použít stíněné kroucené dvojlinky, aby se co nejvíce omezilo rušení přenosu, které by mohly zapříčinit chybnou funkci celého systému. Mezi zdroje rušení přenosu

patří atmosférické vlivy a zdroje elektromagnetického rušení, které v prostředí slévárny vzniká v okolí transformátorů, při spínání silových obvodů, nebo například při rozpojování indukční zátěže, kterou jsou mimo jiné i topné spirály regulované pece. Datové kabely prodávané firmou B&R mají jeden kroucený pár pro datovou linku (X2X) společně s elektrickou zemí (X2X $\perp$) a druhý pár pro invertovanou datovou linku (X2X $\setminus$), která je kroucená společně s vodičem spojeným ke stínění. Volba spárování kabelů může mít své opodstatnění, a proto jsem zvolil jednotlivé páry obdobným způsobem byť s použitím běžného síťového kabelu.

Protože jsem při testování měl k dispozici pouze jeden zdroj 24V, použil jsem společné napájení X2X linky a vstup výstupních obvodů přivedením kladného pólu zdroje na pin +24V I/O a rozvedením napětí dále na X2X linku pomocí propojky. Záporný pól zdroje jsem přivedl do vstupu pro elektrickou zem GND (Elektrická zem nemusí být vždy uzemněná. Často se pracuje s elektrickou zemí, na které bychom při měření napětí proti zemi neměřili 0V).

X20AT2402 je modul pro měření teploty na jednom nebo dvou termočláncích. Zapojení senzoru je snadné, kladný pól termočláncu připojíme k pinu TC+1, záporný pól se připojí k pinu TC-1. Dále je vhodné v bloku X20 modulů vynechat jednu pozici po obou stranách AT2402. Na pozici vedle dáme tedy pouze sběrnicové části X20 modulů X20BM11. Zajistí se tak lepší chlazení a především se zamezí nevhodným změnám teploty měřicího modulu z okolních modulů, které by mohly při zatížení generovat teplo a snížit tím přesnost měření teploty. Pro zvýšení přesnosti a zlepšení odolnosti proti rušení můžeme ještě uzemnit pin TC-1 přivedením propojky ze zemnicího vstupu GND modulu BR9300.



Obrázek 7 Rozložení pinů modulů X20DO4322(vlevo) a X20DI4371,
obrázek pochází z manuálových stránek [5]

B&R nabízí moduly s dvěma, čtyřmi, šesti, osmi nebo dvanácti digitálními výstupy v jednom slotu. K regulaci pece připojováním a odpojováním topných spirál ke zdroji skrz SSR

pomocí impulzní šířkové modulace stačí jeden digitální výstup 24V, zvolil bych tedy modul s dvěma výstupy X20DO2322, ale jelikož jsem měl k dispozici již zakoupené čtyř nebo osmi výstupové moduly, vybral jsem si modul se čtyřmi výstupy X20DO4322, který se zapojuje stejným způsobem jako X20DO2322. Pro testovací účely jsem použil jednofázové SSR KSD215AC8. Jedná se o model pro spínání střídavé proudu do 15A při jmenovitém napětí až do 250V. SSR je určeno k montáži na chladič a spíná v nule. Což znamená, že při detekci náběžné nebo sestupné hrany na svém vstupu sepne, nebo vypne v momentu, kdy se střídavé napětí na výstupu protíná s nulou (s maximální odchylkou 35V). Taková konstrukce výrazně prodlužuje životnost všech druhů spínacích prvků.

Ke kladnému pólu ovládací části SSR jsem přivedl výstup z *DO 1* modulu X20DO2322, k zápornému pólu se přivádí libovolný výstup X20DO2322 označený jako *GND*. Mezi výstupem *DO 1* a *GND* je podle logické hodnoty buď napětí 0V pro logickou 0, nebo 24V pro logickou 1. Nezáleží, který výstup *GND* využijeme, protože jsou všechny uvnitř elektronického obvodu spojeny v jeden. Ve skutečnosti nebude výstupní napětí přesně 24V, ale o něco méně v důsledku úbytků napětí na spínacích tranzistorech. To ale nevádí, protože je SSR konstruováno tak, že spíná již při 4V a vydrží vstupní napětí až 30V. V návrhu bylo nutné brát v úvahu zatížení výstupních X20 modulů. Vnitřní odpor ovládacího obvodu SSR je přibližně 600Ω, takže příkon potřebný k jeho správné funkci je 0,96W. Moduly X20DO4322 jsou navrženy, pro maximální zatížení 12W v rozsahu teploty okolí od 0°C do 55°C při horizontální instalaci na DIN lištu bez potřeby aktivního chlazení. Jelikož bude modul umístěn v řídicí skříni poblíž pece, může se teplota ve skříni především v letním období blížit horní hranici únosných pracovních podmínek X20 modulů. Je proto vhodné zatěžovat výstupy co možná nejméně a nepoužívat spínací prvky s vyšším odebíraným výkonem. Dále bude potřeba zajistit uchlazení na požadovanou maximální teplotu 55°C, k čemuž by mělo stačit zajištění dobré cirkulace vzduchu kolem modulů. Při vysokých teplotách se může z některých produktů odpařovat voda a vznikat tak vysoká relativní vlhkost vzduchu. Zajištění dobré cirkulace vzduchu uvnitř řídicí skříně by mělo také zabránit kondenzaci vodní páry. Relativní vlhkost vzduchu by neměla překračovat 95%.

Pro digitální vstupy jsem použil modul X20DI4371. Vstupní napětí se měří na děliči napětí proti zemi. Při zapojení vstupu na prázdko tedy detekujeme logickou nulu. Digitální vstup bude v logické jedničce, pokud k měřicímu pinu přivedeme spínacím prvkem nebo senzorem kladný pól napájecího napětí 24V.

Jedním z potřebných digitálních vstupů do systému je detekce otevřených dveří pece. Pro tyto účely lze použít jednu ze tří variant. První variantou je použít nějaký druh dvoupólového vypínače, který vydrží vysoké teploty při otevření rozehřáté pece. Jeden pól pro připojení ovládací části stykače, sloužící k bezpečnostnímu odpojení napájení topných spirál. Druhým pólem spínače by spojoval jako vstup pro snadnou detekci otevření dveří pro PLC, která se zapíše do logu událostí a indikuje otevření pece na kontrolním panelu. Můžeme použít oddělený zdroj napájení pro stykač, nebo SSR. Jako druhá varianta se nabízí použít pouze jednopólový vypínač na dveřích pece, kterým by procházel proud napájecí limitní jednotku. Limitní jednotka s ní-

kým příkonem by mohla být napájena přímo z modulu digitálního vstupu. Musíme opět počítat s tím, že v součtu nesmí příkon všech senzorů překročit 12W. Jinak by mohlo dojít ke zničení modulu digitálního vstupu. Třetí variantu, kterou jsem se chystal z počátku použít, bylo použít jednopólový vypínač na dveřích pece pro digitální vstup s využitím druhé sondy pro měření teploty v blízkosti topných spirál. Poté by se odpojení pece při otevření pece i při dosažení limitní teploty spirál řešilo programově pomocí PLC. Toto řešení jsme zavrhlí z důvodů nízké bezpečnosti. Pokud by došlo k nějakému zacyklení, nebo bych v programu udělal sémantickou chybu, mohlo by výsledné neodpojení napájení spirály zničit, nebo ještě hůře, mohlo by to ohrozit životy zaměstnanců slévárny. Použití externích obvodů je bezpečnější varianta.

4.2 Programování

K programování svých produktů firma B&R vyvinula vývojový software pojmenovaný B&R Automation Studio (AS). Jedná se o vývojové prostředí podporující pouze operační systémy Microsoft Windows XP, Windows Vista a Windows 7. Prostředí poskytuje všechny nástroje potřebné pro vývoj včetně integrovaného návrhu grafického uživatelského rozhraní. Umožňuje vytvoření základní konfigurace operačního systému s nastavením komunikačního rozhraní pro zvolený PLC. Následné nahrání operačního systému na flash kartu power panelu. Poté vzdálené programování, ladění a údržbu řídicích procesů. Jednoduché napojení těchto procesů přes proměnné k vizuálnímu návrhu, nebo k digitálním a analogovým vstupům a výstupům z řídicích soustav, připojených k PLC například linkou X2X. Díky dobré nápovědě, několika přibaleným vzorovým projektům, tréninkovým materiálům a zkušenosti s návrhem aplikací v prostředích jako například Borland developer studio jsem se v průběhu několika dní naučil efektivně využívat jeho možnosti.

Při otevření nového projektu se spustí průvodce vytvořením projektu. Po volbě umístění a názvu projektu je potřeba zvolit používaný model PLC. Power Panel PP45 vybraný pro řídicí systém nese označení 4PP045.0571-062. Po dokončení průvodce nám AS nabídne vytvoření konfiguračního souboru s příponou „dis“, tento soubor slouží k přiřazení naprogramovaných virtuálních kláves k událostem na tlačítkách okolo displeje. AS celý projekt rozdělí do 3 pohledů ve formě záložek v levém panelu vývojového prostředí:

- 1) *Logical View* – V tomto pohledu můžeme spravovat zdrojové kódy programů v různých programovacích jazycích, knihovny a grafické návrhy aplikace. Prvky v *logical view* jsou hardwarově nezávislé a přenositelné do jiných projektů a hardwarových sestav.
- 2) *Configuration View* – Pohled zahrnuje několik souborů s nastavením mapování vstupů a výstupů z PLC. Slouží především ke správě hardwarově závislých konfigurací.
- 3) *Physical View* – Pohled umístěný pod třetí záložkou se užívá ke správě hardwarových modulů a jejich zobrazení ve stromové struktuře.

Aby bylo možné jakkoli konfigurovat funkci kláves, je nutné přidat do projektu objekt spravující vizualizaci, který se stará kromě grafického výstupu i o vstupy z dotykového displeje

a okolních kláves. PP45 patří mezi PLC s procesory čtvrté generace, pro které se využívá vizuální objekt VC4. Objekt můžeme přidat do projektu v *logical view*. Dále je potřeba nastavit konfigurační soubor „Visual.vcm“, který je spojujícím článkem mezi grafickým výstupem a vstupy z tlačítek a dotykového displeje. V *configuration view* je potřeba pomocí grafického nástroje přiřadit vytvořený VC4 objekt ke konfiguračnímu souboru s příponou „dis“, který se vytvořil po dokončení průvodce vytvořením projektu. V této fázi je již možné v *logical view* pomocí VC4 objektu otevřít nástroj pro grafický návrh aplikace.

Pro usnadnění vývoje jsem si nakonfiguroval Ethernetové TCP/IP rozhraní, přes které jsem poté PP 45 programoval. Je možné vytvořit celý projekt včetně řídicích procesů a vizualizace a až poté jej nahrát přímo na CompactFlash kartu, která tvoří aplikační paměť PLC. Možnost programovat PLC po síti je však mnohem pohodlnější a rychlejší. Ušetříme čas strávený vyjímáním karty a dobu nutnou na tzv. studený restart. Obsah permanentních programovatelných proměnných v paměti typu SRAM, který je v případě běžného odpojení od napájení zálohovaný baterií, je při studeném restartu ztracen. Do nabídky konfigurace Ethernetu se lze dostat z *physical view*.

Pokud je možnost nakonfigurovat si statickou alokaci DHCP serveru a připojit si PLC k síti tak, aby mu DHCP server přidělil podle MAC adresy pokaždé stejnou IP adresu, pak doporučuji nechat konfiguraci sítě na implicitním nastavení, tedy získání IP adresy z DHCP serveru. Pokud by při restartu PLC obdrželo jinou IP adresu, bylo by nutné vyhledávat nově přidělenou IP adresu v síti a změnit nastavení síťového připojení v AS.

Celý postup vytvoření projektu, kdy je nutné mnohokrát přecházet mezi jednotlivými pohledy, pro vytvoření objektu, a následné konfigurování mi připadá velice zdoluhavé a nepřehledné. Při použití PLC s dotykovým displejem jej nebude využívat jen malé procento aplikací, a proto by bylo vhodné již během průchodu průvodcem vytvoření projektu přidat vývojáři možnost zavedení prázdného vizuálního objektu s funkční konfigurací již při volbě PP45. Velice by se tím usnadnily začátky práce ve vývojovém prostředí, které často končí zdoluhavým procházením mnoha nabídek a oken s nastavením.

AS poskytuje možnost programovat PLC v jednom z osmi programovacích jazyků s možností kombinovat některé jazyky v rámci jednoho projektu. [5]

- 1) ANSI C
- 2) Structured text (ST) – Textový programovací jazyk používající strukturované angličtiny, svou syntaxí podobný jazyku pascal. Je definovaný standardem IEC 61131-3.
- 3) Instruction list (IL) – Textový programovací jazyk se sekvenčním zápisem jednoduchých instrukcí. Co se týká míry abstrakce, jedná se o jazyk nižší úrovně než jazyk C. (IEC 61131-3)

- 4) Sequential function chart (SFC) – Z větší části graficky orientovaný programovací jazyk s prvky textového jazyka (IEC 61131-3). Používaný k popisu časových posloupností akcí uvnitř programu.
- 5) Ladder diagram (LD) – Grafický orientovaný programovací jazyk (IEC 61131-3). Program se podobá elektronickému schématu, kde jednotlivé součástky zastupují jednoduché funkce.
- 6) Function block diagram (FBD) – Graficky orientovaný programovací jazyk (IEC 61131-3). Využívající předdefinované funkční bloky, které se pouze graficky spojují. Lze si vytvořit vlastní knihovnu, ve které si definujeme funkčními bloky. Více o využití knihoven s funkčními bloky uvádím v kapitole 4.2.1.
- 7) Continuous function chart (CFC) – Graficky orientovaný programovací jazyk, který oproti SFC umožňuje nepřetržité vykonávání programu ve smyčkách pomocí zpětných vazeb.
- 8) Automation basic (AB) – Testový programovací jazyk vyvinutý speciálně pro B&R. Umožňuje využívat všechny dostupné programovatelné

Ze standardu IEC 61131-3 vychází i označení datových typů a několik standardních funkcí.

Pro vývoj řídicího systému pece jsem si zvolil jazyk ANSI C, protože s prací v něm už mám zkušenosti a dobře se v něm pracuje s ukazateli. Některé části systému by bylo možné navrhnout i v grafických programovacích jazycích, kde by byl program i snadno čitelný. Složitější části by se pak mohly napsat v AB, nebo pomocí jazyka ST. Tyto dva zmíněné jazyky spolu s ANSI C umožňují využívat všech možností vývojového nástroje. Jazyk ANSI C jsem si zvolil také proto, že je všeobecně známý a moje zdrojové kódy by měly sloužit i jako základní kámen, nebo alespoň jako zdroj inspirace a ukázka použití funkčních bloků pro další rozšiřování systému jinými programátory.

Možnost použít pro návrh aplikací jiný programovací jazyk než C je vítaná především pro odborníky v daném průmyslovém odvětví, kteří se vyznají například ve strojírenství a PID regulátorech, ale nemají mnoho zkušeností s textovými programovacími jazyky.

Každý program, který v projektu vytvoříme, musíme v softwarové konfiguraci přiřadit k jedné ze tříd cyklických programů. U cyklické třídy můžeme měnit periodu vykonávání a časovou toleranci, během které musí být všechny procesy v třídě proběhnout do konce. Pokud nestihnou v daném čase ukončit svou činnost, systém ukončí běh programu a provede takzvaný studený restart, při kterém smaže obsah trvalých proměnných a po dalším zavedení se podle konfigurace systému pokusí buď spustit běžným způsobem, nebo naběhne do diagnostického režimu. Ze kterého musí být restartován obsluhou. Také je potřeba počítat s tím, že pokud nastavíme procesu příliš krátkou periodu vykonávání, přičemž jejich vykonání bude trvat půl této periody, vyčerpáme tímto procesem 50% výkonu procesoru a pro ostatní programy nebude zbývat

mnoho operačního času. Cyklické třídy jsou seřazeny podle priority. Procesy s vyšší prioritou jsou upřednostňovány před procesy s nižší prioritou. Proces regulující teplotu v peci je důležitější než ostatní, a proto by měl mít vyšší prioritu než proces starající se o vizualizaci a výpočty dat pro zobrazení grafu průchodu programem.

Napojení na vstupy a výstupy z hardwarových modulů je velice jednoduché, stačí v konfiguraci příslušného modulu v *physical view* přiřadit ke vstupu či výstupu programovatelnou proměnnou stejného typu. Podle zařazení aplikačních procesů, které s proměnnou pracují, do cyklické třídy o určité periodě vykonávání AS automaticky přiřadí předávce dat obslužný proces, který zařídí předávání parametrů z nebo do datového výstupu. Zní to sice složitě, ale pro programátora to znamená pouze to, že když zapíše hodnotu do proměnné napojené na výstup X20 modulu, tak po provedení celého cyklu programu může počítat s tím, že příslušný modul nastaví na svůj výstup stejnou hodnotu.

Jelikož, jak jsem se již zmínil, je plánováno rozšíření PLC na řízení několika pecí zároveň, je zapotřebí navrhovat software univerzálně pro více druhů pecí a vyhnout se používání velkého množství globálních proměnných fixovaných pouze k jedné peci. Veškeré konfigurace, kalibrační nastavení, programy řízení a jiné uchovávané informace pro konkrétní pec musí být snadně zpracovatelná navrženou metodou struktury, nebo třídy.

V dnešní době už B&R pravidelně pořádá školení, které v týdenním kurzu seznámí vývojáře s možnostmi Automation studia a na několika příkladech jim pod vedením zkušených lektorů dovolí osvojit si metody návrhu řízení automatizačních systémů, návrhu vizualizace, nebo řídicích systémů strojů využívajících různé pohyblivé části. Tyto kurzy jsou pro studenty vysokých škol zdarma.

S většinou možných problémů spojených s programováním pomůže nápověda AS, nebo technická podpora firmy B&R. Zaměstnanci firmy mají aktivní přístup k řešení problému svých zákazníků.

4.2.1 Funkční bloky

AS v sobě zahrnuje velké množství knihoven, ty obsahují řadu funkcí a pokročilých objektů tzv. funkčních bloků. V řídicím systému jsem kromě knihoven, které do projektu přidal průvodce vytvořením projektu, přidal knihovnu *asTime*, umožňující snadný přístup k systémovému času spolu s jeho naformátováním pro grafický výstup. Dále velice užitečná knihovna *asString* pro práci s textovými řetězci. Pro práci se soubory a adresáři jsem využíval knihovnu *FileIO*, jejíž popis a příklad použití funkčních bloků uvádím v kapitole 4.2.3. Jako poslední avšak ne méně významná je knihovna *LoopConR*.

Jedná se o knihovnu obsahující regulační funkce pracující s argumenty vyjádřenými jako číslo s plovoucí řádovou čárkou. Funkční bloky *LCRTempTune* a *LCRTempPID* jsem využil pro automatické ladění PID parametrů pomocí skokové metody a pro následnou regulaci teploty.

ty. Výstupem regulátoru je vždy číslo uvádějící v zadaném rozsahu procentuální výkon, kterým by pec měla topit. Jelikož se pro tavnou pec s vysokým výkonem je těžko realizovatelné řízení spojitým signálem jinak než změnou střídý⁷ pulsně šířkové modulace. Musel jsem pro výstup z regulátoru převést pomocí funkčního bloku LCRPWM, který je taktéž součástí knihovny *LoopConR*. Do funkčního bloku vstupuje perioda modulace a střída. Tato vrstva řízení bude muset být pro budoucí rozšíření pro řízení více pecí se střídavým spouštěním nahrazena jinou funkcí, která by měla zpracovávat výsledky z regulátorů pro jednotlivé pece a podle nich přerozdělovat příkon dostupný z energetické sítě do pecí podle procentuální potřeby a nastavených priorit takovým způsobem, aby nevyrazila jističe elektrické rozvodny v hale.

Knihovna *LoopConR* dále obsahuje funkční blok LCRSimModExt, který slouží jako simulační model vytlačujícího lisu se zahříváním materiálu. Využil jsem simulaci ohřevu, který se počítá jako součet proporcionálního zesílení s derivacemi prvního a druhého řádu. Po nastavení správných parametrů jsem získal systém, který reagoval podobně jako reálná pec.

4.2.2 Paměť a programové proměnné

Jak jsem se již zmínil v kapitole 3.1 PLC, operační paměť je složena ze dvou typů paměti, a sice DRAM a SRAM. Primárně jsem ve své aplikaci upřednostňoval používání paměti DRAM, je mnohem větší a také rychlejší. Jedna z nevýhod programování aplikací pomocí PP45 je, že nedovoluje alokovat paměťový prostor pro proměnné dynamicky za běhu programu. Musíme tedy znát velikost používaných dat již v době kompilace programu. Často tak dochází k plýtvání prostorem, je ale pravda, že se tímto ušetří výpočetní prostředky, které by se jinak musely systémem vynaložit ke správě paměti.

Uložení proměnných v paměti typu SRAM ve své aplikaci používám zejména k uložení statusů pro detekci neočekávaného odpojení napájení před řádným ukončením programu, nebo k uložení některých konfigurací, ke kterým musí být snadné čtení a zápis z grafického rozhraní. Velikost využití SRAM pro trvalé proměnné je limitována nastavením softwarových vlastností systému. Ke kterému se dostaneme z *configuration view* otevřením souboru *sysconf.br*. V záložce *memory* pod položkou *PV memory* bylo nutné zvýšit maximálního prostor pro trvalé proměnné, protože je implicitně po otevření projektu nastaven na 0 Bytu.

Abych mohl snadno ukládat a znovu načítat programy, navrhl jsem si datovou strukturu obsahující vše co je potřebné ke spojitě regulaci teploty v krocích uživatelem nastavené délky. Podobně jako tomu je u kompaktních průběhových regulátorů teploty. Uživatel si může nadefinovat až 10 kroků přičemž pro každý krok uživatel určí teplotu, na kterou se má teplota přizpůsobit a dobu za kterou má být krok dokončen. Datová struktura je definovaná v souboru *Global.typ*. V kapitole 4.2.4.2 jsem uvedl příklad použití řízení teploty programem.

⁷ střída = délka pulsu : délka periody

4.2.3 Uchování dat – ukládání programů řízení a logů

AS dovoluje s použitím knihovny *FileIO* jednoduše pracovat se soubory na CF kartě vložené v PLC, nebo s jiným datovým nosičem, jako například přes USB připojený Flash Disk. K práci se soubory se používá podobných funkcí jako při použití standardní knihovny *stdio.h*, najde se zde ale několik rozdílů. Jelikož se jednotlivé procesy cyklicky spouštějí a dovolují pouze rychlé synchronní akce, ale práce se soubory vyžaduje přenos po interní sběrnici PLC, čtení i zápis jsou řádově pomalejší, než zápis do dynamické paměti jsou funkční bloky nezbytné pro asynchronní práci se soubory pomocí synchronního volání z cyklických procesů. Knihovna poskytuje funkční bloky umožňující práci se soubory a adresáři na zařízení pro uložení dat. Pro otevření existujícího souboru se používá funkční blok *FileOpen*. Ten je nutné nejprve inicializovat strukturou definovanou funkčním blokem. Vstupy a výstupy z funkčního bloku vidíme po otevření souboru *FileIO.fun*, který nalezneme v záložce *logical view*. Nastavíme ukazatele na datové zařízení, ukazatele na řetězec s názvem souboru, který chceme otevřít a módu otevření souboru. K dispozici je mód pro čtení *FILE_R*, mód pro zápis *FILE_W* a mód pro čtení i zápis *FILE_RW*. Všechny tyto módy jsou binární, což dovoluje ukládat do souborů například obsah uživatelsky definované struktury. Této možnosti jsem využil pro ukládání programů regulace a dala by se v budoucnu využít na uložení konfigurací a parametrů tavných pecí. Prakticky je možné uložit si jakoukoli datovou strukturu do binárního souboru a poté ji opět načíst. Operace čtení a zápisu vyžadují jako mezi svými parametry, i zadání počtu čtených nebo zapisovaných Bytů a offset. Je tedy možné v jednom souboru ukládat skupiny proměnných různých typů.

```
/* deklarace proměnných potřebných proměnných */
_LOCAL FileOpen_typ FOpen;
_LOCAL unsigned long dwIdent;
_LOCAL unsigned char byState;
_LOCAL unsigned short wError;
_LOCAL unsigned short wStatus;

/* inicializace funkčního bloku pro otevření souboru */
FOpen.enable = 1;
FOpen.pDevice = (UDINT) "CF";
FOpen.pFile = (UDINT) "test.txt";
FOpen.mode = FILE_R;
byState = IN_PROGRESS;
```

Poté opakovaně při každém vykonání cyklické části programu voláme funkční blok s referencí na instanci inicializované struktury. Funkční blok vrací v proměnné *status* stav asynchronní operace. Stav 65535 znamená, že je operace nedokončená a musíme tedy funkční blok zavolat v příštím cyklu znovu. Pokud vše proběhne v pořádku, vrátí funkční blok *status = 0* a z proměnné *ident* se získá identifikátor souboru pro pozdější volání bloků *FileWrite*, *FileRead* a *FileClose*. Na rozdíl od funkce *fopen* knihovny *stdio.h* funkční blok nevytvoří neexistující

soubor, ani pokud požadujeme mód zápisu. Soubor však můžeme vytvořit pomocí funkčního bloku *FileCreate*. Každý soubor, který otevřeme nebo vytvoříme je vždy nutné na konci práce uzavřít funkčním blokem *FileClose*, aby byl uvolněn manipulátor souboru, kterých je omezené množství. Pokud by operačnímu systému došly manipulátory, mohlo by dojít k pádu.

```
/* cyklicky volaná část programu */
if (byState == IN_PROGRESS) {
    FileOpen(&FOpen); /* volání funkčního bloku */
    dwIdent = FOpen.ident;
    wStatus = FOpen.status;
    if (wStatus == 0) {
        byState = DONE;
    } else if (wStatus != 65535) {
        byState = ERROR;
        if (wStatus == 20799)
            wError = FileIoGetSysError();
    } else {
        byState = IN_PROGRESS;
    }
}
```

Práce se soubory lze využít i pro ukládání provozních logů. Jejich implementace sice nebyla součástí zadání, ale protože jsem chvíli zkoušel možnosti využití práce se soubory a udělal jsem si částečnou implementační analýzu, vložil jsem nakonec do projektu i ukázkou zdrojového kódu pro zápis do souboru, která by měla pomoci v dalším rozšíření systému o logy. Jelikož se soubory zapisují asynchronně a během zápisu může nastat situace, kdy bude potřeba zapsat záznam o uživatelské akci nebo jiné události do logu, musíme zajistit, aby řetězec, který zapisujeme, nebyl upravován jinými procesy. Tomu lze zabránit implementací semaforů, které ovšem mohou vést ke ztrátě některých informací, protože není možné, aby byly procesy řídící regulaci teploty pozastaveny na dobu potřebnou na dokončení zápisu. Procesy řízení jsou spouštěny mnohem častěji než proces zapisující do logů a přesnost řízení teploty je závislá na pravidelném spouštění regulačních algoritmů. Možností jak se vyhnout problémům se ztrátou logů událostí v průběhu zápisu je vytvořit textový řetězec dostatečné velikosti pro každý proces, který může vytvářet informace pro logy na místě přístupném servisnímu procesu, který se stará o zápisy logů na disk. Shromážděná provozní data by se neměla zapisovat do log souboru příliš často, interval mezi zápisy by měl být řádově v jednotkách až desítkách sekund, protože CF karta nemá neomezený počet zápisů a pokud bychom se snažili připsávat data příliš často, vydržela by karta jen několik dní. Na druhou stranu regulace teploty je dlouhodobý děj, u kterého stačí zaznamenávat teplotu přibližně jednou za minutu. Servisní proces starající se o zápis logů by měl mít nižší prioritu a časovou periodu než proces, který se stará o regulaci.

4.2.4 Uživatelské rozhraní

Displej poskytuje prostor o rozlišení 320×240 obrazových bodů, což není mnoho, oproti kompaktním PID regulátorům se segmentovými displeji je to ale obrovský rozdíl. Řízení jsem musel rozdělit na několik stránek, na kterých jsem se pokusil co nejefektivněji a nejintuitivněji rozmístit ovládací prvky. Pro snadnou orientaci a navigaci jsem na všechny stránky umístil záhlaví, které je navrženo jako univerzální vrstva. Obsahuje název stránky, na které se uživatel nachází a aktuální čas. Zanechal jsem ještě prostor pro stručnou informaci o stavu všech pecí, která může být řešená graficky formou obrázkových značek, nebo krátkého textu. Záhlaví má výšku 15 pixelů (obrazových bodů), takže pro zbylé komponenty na každé stránce zůstává prostor 320×225 bodů.

V nástroji návrhu vizuálních komponent, je k dispozici paleta komponent, které se jednoduše pomocí myši rozvrhnou na požadovanou stránku, nebo univerzální vrstvu. Ve stromové struktuře si pak můžeme nastavit vzhled, přiřadit datový vstup nebo výstup, přidělit komponentě události pro kliknutí a jiné pro danou komponentu specifické vlastnosti. AS nabízí pro grafický návrh řadu předdefinovaných objektů. Například komponentu *Numeric*, která může sloužit jako uživatelský vstup pro zadávání celých i desetinných čísel, a lze ji přiřadit po stisku zobrazení předdefinované virtuální numerické klávesnice. Vstupní hodnoty lze omezit na uživatelem zvolený rozsah, aby numerická klávesnice nedovolila potvrdit volbu čísla mimo dovolený rozsah. Po potvrzení volby nové hodnoty v numerické klávesnici se hodnota v následujícím průchodu obslužného procesu zapíše do programovatelné proměnné, ke které je komponenta *Numeric* přiřazena.

Numerická klávesnice je vícevrstvý objekt složený z obrázku podkladu překrytým další vrstvou obrázků, které se zobrazí při stisku virtuální klávesy. Stisk klávesy detekuje průhledná mapa tlačítek v nejvyšší vrstvě. Kromě numerické klávesnice jsou předdefinované ještě horizontální a vertikální navigační klávesnice nebo třeba alfanumerická klávesnice rozdělená na tři stánky obsahující čísla, všechny znaky anglické abecedy a většinu viditelných speciálních symbolů.

Po spuštění PP45 probíhá asi minutu zavádění operačního systému, poté se na displeji vždy objeví buď hlavní nabídka, nebo pokud došlo k výpadku proudu či restartu, pak se zobrazí stránka s obnovou přerušené činnosti. Hlavní stránka slouží především pro navigaci pro další činnosti a jako přehled funkce vnějších tlačítek v okolí displeje. Do hlavní nabídky se lze kdykoli vrátit pomocí tlačítka F1.

Pro snadnou orientaci v možnostech ovládání jsem používal pouze základní varianty tlačítek a pro komponenty s možností vkládání textu nebo čísel jsem zvolil jednotný vzhled s ohraničením, zatímco prvky využívané pouze jako výstup dat jsem volil bez ohraničení, přičemž taková komponenta se intuitivně jeví jako neaktivní. V případě že obsluha neví jaké navigační tlačítko použít například proto, aby se dostal z menu nastavení PID parametrů k nahrání a spuštění programu, má vždy možnost přejít do hlavního menu pomocí klávesy F1 a odtud pokračo-

vat. Na stránkách, na kterých je smysluplné zobrazovat aktuální teplotu v peci, jsem tuto informaci umístil do pravého dolního rohu displeje. V levém dolním rohu je zobrazena IP adresa přidělená DHCP serverem. Na stránkách s časovými průběhy se pak nachází jen nezbytně nutné ovládací prvky, aby mohla být co největší plocha využita trendem.



Obrázek 8 Hlavní nabídka uživatelského rozhraní

Komponenta zobrazující trendy nabízí zobrazení dvou typů dat. Prvním typem je trend pro sledování hodnoty proměnné v čase. Proměnná může být programově napojená například na teplotu z měřicího čidla. Druhý typ trendu, který jsem použil pro zobrazení časového průběhu uživatelem definovaného programu tavení, je uživatelský trend. Uživatelský trend zobrazuje pole proměnných číselného typu. Nadefinoval jsem si tedy v souboru *Global.typ* datovou strukturu *ProgView* pro uchování potřebných parametrů a dat pro uživatelský trend. Dále jsem si v programu *AddProg.c* nadefinoval proceduru *calc_view*, která spočítá lineární spojnicí mezi uživatelem definovanými kroky programu, které jsou uloženy ve struktuře pro uložení programu. Výsledek výpočtu uloží do struktury typu *ProgView* v takovém tvaru, který se v trendu zobrazí podobně jako v Graf 1 Průběh požadované regulace nacházejícího se v kapitole 4.2.4.2.

Pohyb v jednotlivých nabídkách je intuitivní. Obsluha se může plně soustředit na svěřené úkoly a není zatěžována přemýšlením nad tím, které tlačítko by měla zvolit, aby se dostala na požadovanou stránku. Po nasazení regulátoru do provozu se na základě ohlasu zaměstnanců slévárny bude uživatelské rozhraní ještě přizpůsobovat. V provozu se můžou ukázat některá grafické prvky nadbytečné, nebo rozložení ovládacích prvků nedokonalé.

4.2.4.1 Automatická kalibrace

Do stránky s automatickou kalibrací se dostaneme skrz stránku s nastavením. Máme zde možnost spustit automatickou kalibraci pomocí takzvané „step response“ metody. Tato metoda funguje tak, že spustí pec na 100% výkon, dokud se výsledná teplota nepřiblíží pracovnímu bodu, pro který parametry měříme. Během této procedury se zjišťuje strmost nárůstu teploty, rychlost detekce změny teploty o několik procent po startu kalibrace a setrvačnost zvyšování teploty po odpojení energie. Ze získaných informací funkční blok vypočítá parametry pro PID regulaci.

Na stránce s auto-kalibrací je možnost upravit si jednotlivé konstanty PID regulace ručně. Proto není problém, aby si zaměstnanci později v provozu v případě potřeby ručně upravili parametry regulace. Například pokud by se choval regulátor nestabilně a při změnách požadované teploty by se objevovaly překmity, mohla by obsluha zvýšit integrační periodu T_N a tím snížit podíl integrační konstanty na řízení. Před spuštěním auto-kalibrace se navíc nejprve musí zvolit pracovní bod, pro který se má celá kalibrace provádět. Pokud se kalibrace provede s pracovním bodem 600°C a poté budeme chtít teplotu v peci upravit na 1000°C, bude regulátoru trvat dlouho stabilizovat teplotu na požadované hladině s naměřenými parametry. V ideálním případě by mělo stačit zvýšit parametr K_p ve stejném poměru. I když při testování v softwarové simulaci pece toto platilo, teprve v provozu se ukáže, zda nebude potřeba měnit i nastavení časových konstant integrační a derivační složky, nebo zvyšovat proporcionální zesílení nelineárně.

Pro změnu dynamiky systému, tedy rychlosti reakce na změnu nastavené požadované teploty, má funkční blok PID regulátoru teploty knihovny LoopConR zvláštní parametr *dynGeneral*, kterou lze nastavit v rozsahu 0 až 1. Při vysokém nastavení dynamiky regulátor rychle reaguje na změny krátkodobou úpravou parametrů. Naopak při nastavení dynamiky na 0. Bude pokračovat v regulaci na základě nastavených parametrů bez pokusů o urychlení regulačního děje.

4.2.4.2 Příklad použití řízení teploty programem

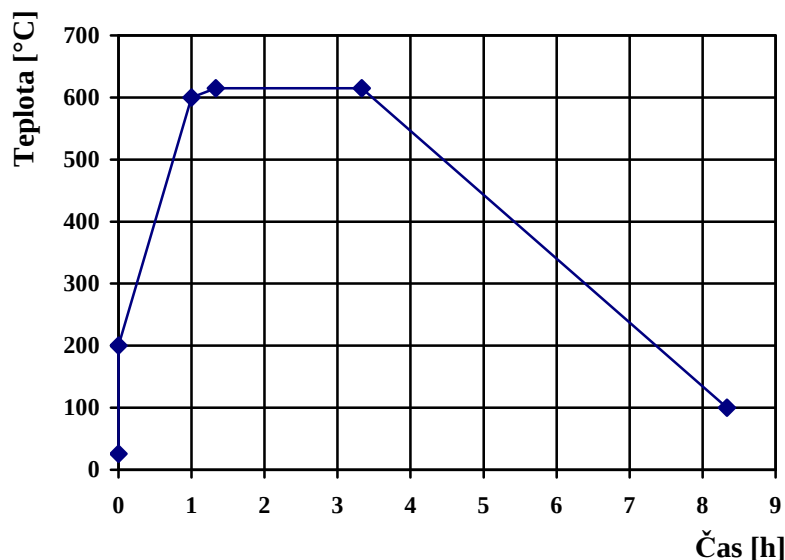
Na jednoduchém příkladu vytvoření programu řízení teploty se pokusím vysvětlit principy řízení pomocí uživatelsky nastavených kroků. Časy a teploty jsem si vymyslel, skutečné vlastnosti vosků a jiných materiálů jsou pro tuto ukázkou nepodstatné. Potřebujeme například ohřát pec na 615°C, přičemž od teploty 200°C nesmíme měnit teplotu příliš rychle. Teplotní překmity nad teplotu 615°C jsou nežádoucí. Po dosažení cílové teploty ji budeme 2 hodiny udržovat a nakonec necháme pec velice pomalu vychladnout.

Počáteční teplota v peci je 26°C. Aby se teplota v peci zvýšila maximální rychlostí na 200°C, přidáme první krok s časem 0 minut a teplotou 200 °C, potvrdíme přidání kroku. Jako další si přidáme krok, který zabrání případnému překmitu nad požadovanou teplotu pro případ, že PID parametry nebyly kalibrovány přesně na danou teplotu. Podle požadované strmosti růstu teploty zvolíme čas dalšího kroku na 1 hodinu a několik stupňů pod požadovanou hladinou na-

příklad na 600°C přidáme bod, ke kterému se bude teplota v peci lineárně zvyšovat, potvrdíme přidání 2. kroku. 3. krok zvolíme s nižší strmostí až na konečnou teplotu 615°C se zadáním času 20 minut. V následujícím kroku neměníme teplotu, jen zadáme čas, po který se má teplota udržovat a jako poslední přidáme sestupný krok. U vzestupných kroků se vyhodnocuje krok ukončený po překročení cílové teploty, čas kroku je jen pro určení strmosti. Podobně je tomu u sestupného kroku, ve kterém je krok považován za dokončený v okamžiku, kdy teplota klesne pod cílovou teplotu. Naproti tomu ve 4. kroku se nemění teplota, a proto systém vyhodnotí tento krok jako nerostoucí ani neklesající, udržuje tedy teplotu na požadované hladině a krok vyhodnotí jako dokončený po uplynutí času kroku, tedy 2 hodiny po ukončení 3. kroku.

Tabulka 2 Příklad programu regulace

	1. krok	2. krok	3. krok	4. krok	5. krok
Teplota [°C]	200	600	615	615	100
Zadaný čas kroku [hod:min]	0	1:00	0:20	2:00	5:00
Čas od startu [hod:min]	0	1:00	1:20	3:20	8:20



Graf 1 Průběh požadované regulace

4.3 Vyhodnocení výsledků

Všechny implementované funkce PLC jsem otestoval nejprve na softwarovém simulátoru pece a poté i na sestrojeném testovacím zařízení. Výsledky testu na simulátoru implementovaném pomocí funkčního bloku LCRSimModExt odpovídaly představám zadavatele. Při dalším testování systému zapojením vnějších modulů na zdroj tepla v podobě horkovzdušné pistole a měření teploty v blízkosti zahřívaného materiálu sice nebyly výsledky dokonalé, ale byly uspokojivé. Vzhledem k tomu, že zdroj tepla byl příliš výkonný a zahřívaný materiál příliš rychle přenášel teplo k čidlu, projevovalo se každé zapnutí a vypnutí horkovzdušné pistole záklitem teploty. Teplota se pohybovala v rozsahu několika °C okolo aktuálně požadované teploty. Tento problém by na skutečné peci neměl nastat, a proto považuji výsledky za přijatelné.

Sestavený řídicí systém umožňuje uživateli regulovat tavnou pec podle uvedených požadavků. Zvládá všechno, co nabízejí specializované PID regulátory. Tedy PID regulaci teploty s přesností, která bude stejně jako u kompaktních modulů záviset na typu zvoleného čidla, kvalitě jeho zapojení a správném nastavení parametrů regulace. Při měření teploty 1000°C modulem AT2402 je při použití článku typu S maximální chyba měření 1,7°C. Abychom získali celkovou maximální chybu měření, musíme k chybě měření modulu přičíst toleranci samotného čidla. Přesnost měření a regulace je tedy srovnatelná se specializovanými kompaktními PID regulátory.

K tomu přidává nový systém spoustu funkcí, které pro běžně prodávaná kompaktní řešení nejsou k dispozici. Mezi tyto funkce patří možnost sledovat teplotu v peci na časových trendech, podobně jako zobrazení vytvořeného programu tavení. Při krátkodobém výpadku proudu elektrické sítě se systém dokáže zotavit, pokud zjistí, že výpadek netrval příliš dlouho, naváže na předchozí činnost, pokračuje v regulaci teploty dle zvoleného programu a informuje o události uživatele.

5 Závěr

Při mé práci jsem se seznámil s potřebami firmy, technologií výroby odlitků metodou přesného lití, PID regulací teploty, vlastnostmi a možnostmi využití PLC PP45. Nejdůležitější poznatky jsem shrnout v této bakalářské práci. Navrhnul jsem řídicí systém s možností budoucího rozšíření. Řídicí systém jsem naprogramoval, sestavil a v laboratoři otestoval. Jakmile k tomu bude mít firma MSR Engines dostatek času, bude možné systém otestovat na reálné peci. Pravděpodobně bude aplikovatelný na pec v provozu bez velkých úprav. Nepodařilo se mi nasimulovat chování limitní jednotky, kterou jsem neměl k dispozici, ale PID regulátor je dostatečně přizpůsobivý, takže dokáže vykompenzovat dočasné přerušení ohřevu limitní jednotkou.

Systém nabízí mnoho možností budoucího rozšíření, například by mohl zvládat regulaci více pecí najednou, přičemž je možné přizpůsobit program i pro pece s řízením teploty s použitím ohřevu i chlazení. Diagnostické nástroje by se mohly rozšířit o záznamy teplotních průběhů a nástroj pro stahování logů do PC a jeho následnou archivaci nebo analýzu.

Použité informační zdroje

- [1] BALÁTEĚ, Jaroslav. *Automatické řízení* . 2. aktualizované vydání. Praha : BEN - technická literatura, 2004. 664 s. ISBN 978-80-7300-148-3.
- [2] SCHLEGEL, Miloš. *Průmyslové PID regulátory : Tutorial*. REX Controls [online]. 04.03.2009, č. 29, [cit. 2011-05-11]. Dostupný z WWW: <http://www.rexcontrols.cz/downloads/clanky/PIDTutor_CZ.pdf>.
- [3] KRACMAN, Ondřej. *Faktory ovlivňující přesnost odlitků u metody vytavitelného modelu*. [online]. [cit. 2011-05-12]. Dostupný z WWW: <http://www.fsid.cvut.cz/cz/u218/stc/Sbornik/S3/Kracman_Ondrej_12133.pdf>.
- [4] B&R. *Power Panel : User's Manual* [online]. version 2.10. [s.l.] : B&R, 2008 [cit. 2011-05-12]. Dostupné z WWW: <http://www.br-automation.com/cps/rde/xchg/br-productcatalogue/hs.xsl/services_114476_ENG_HTML.htm>.
- [5] B&R. *X20 System : User's Manual* [online]. version 2.10. [s.l.] : B&R, 2009 [cit. 2011-05-12]. Dostupné z WWW: <http://www.br-automation.com/cps/rde/xchg/br-productcatalogue/hs.xsl/services_133603_ENG_HTML.htm>.
- [6] B&R. *Perfection in automation* [online]. 26.4.2011 [cit. 2011-05-10]. Dostupný z WWW: <<http://www.br-automation.com>>.
- [7] Wikipedia : the free encyclopedia [online]. 2011, 30 April 2011 [cit. 2011-05-05]. PID controller. Dostupné z WWW: <http://en.wikipedia.org/wiki/PID_controller>.
- [8] B&R. *Automation Studio : Training Modules*. [CD-ROM]; 2006.

Seznam příloh

Příloha 1. CD s elektronickou verzí bakalářské práce a výsledným projektem.