

Operační systémy reálného času s Win32 API

Ladislav Reisner

Časy, kdy se do „holého“ hardware počítače nahrál řídicí program, který vše řešil sám a nepoužíval operační systém, jsou až na výjimky minulostí. Snaha po zobecnění potřeb, hledání a nacházení společných rysů a následná standardizace přinesla nástup a nárůst používání operačních systémů. To nejdůležitější, co operační systém programátorům nabízí, jsou aplikační funkce (API). Ty jsou jako nedílná součást operačního systému připraveny v knihovnách a definovány v hlavičkových souborech.

Úvod

Řídicí průmyslové počítače na bázi architektury PC (IPC) dnes vynikají velkým výpočetním výkonem, vysokou spolehlivostí a stále nižší spotřebou energie. Lze je využít jako řídicí jednotky, které se dají dále propojovat do sítě počítačů s klientskými vizualizačními a parametrizačními programy. K těmto příznivým vlastnostem přistupuje i značná pestrost nabídky na trhu hardware a nízká pořizovací cena.

Na druhé straně však s rostoucím výkonem hardwarové platformy získává stále důležitější roli použitá softwarová platforma. Používání operačních systémů umožnilo výrazně usnadnit vývoj, zvýšit spolehlivost aplikací a podstatně zkrátit dobu od zadání po uvedení výrobku na trh. Přirozeným důsledkem této skutečnosti je existence různých operačních systémů s více či méně podobnými rysy, které mohou být více nebo méně vhodné, drahé nebo levné, profesionálně podporované nebo bezplatné.

Tento článek si klade za cíl vyzdvihnout a ukázat široké technické veřejnosti způsoby použití operačních systémů s nejznámějším programátorským rozhraním Win32 API v průmyslových a ostatních, časově kritických aplikacích.

Operační systémy s funkcemi Win32 API

Programátorské rozhraní Win32 API používá velká většina programátorů z celého světa, a to i v řídicích aplikacích. V důsledku toho je zajištěna velká přenositelnost a univerzálnost napsaných zdrojových textů programů, a tím i ochrana investic vložených do vývoje.

Protože existuje více operačních systémů reálného času s Win32 API, nastává nutnost optimální volby. Podstatné faktory při volbě představují:

- typ použitého procesoru CPU,
- rozsáhlost a požadovaná komplexnost řešení,
- počet opakovaného nasazení vyvinutého produktu.

Jestliže je projekt založen na procesoru z řady IBM x86 nebo Pentium, je možné použít operační systémy Windows nebo PharLap ETS. Jestliže projekt používá jiný procesor, jako třeba ARM, MIPS, PPC nebo SuperH, je možné použít jen OS Windows CE. Operační systémy Windows jsou víceúlohové a víceprocesní, zatímco PharLap ETS je víceúlohový, ale není to víceprocesní systém. OS Windows 2000/XP podporují rozhraní pro zpracování ně-

kolika dokumentů souběžně (MDI – Multiple Document Interface), naopak PharLap ETS a Windows CE poskytují pouze rozhraní pro jednoduchý dokument (SDI – Single Document Interface).

Jestliže se chystáte realizovat projekt menšího zařízení a máte zájem toto zařízení vícenásobně opakovat, pak je rozhodně nejlepší použít PharLap ETS. Naopak, pokud zamýšlíte realizovat kusovou výrobu nebo potřebujete řešit rozsáhlejší funkce, jako například použití DCOM nebo OPC (OLE for Process Control), pak je lépe použít OS Windows 2000/XP.

Operační systémy pevného reálného času

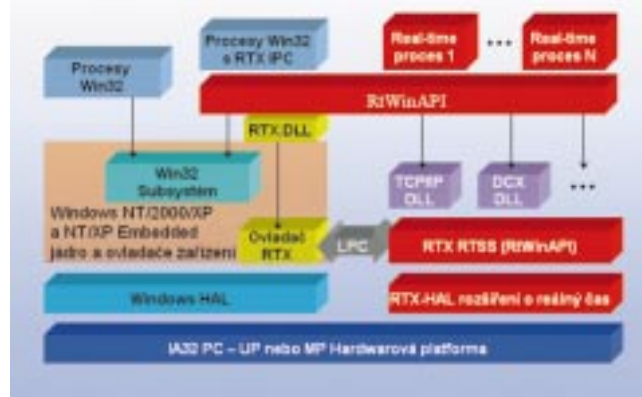
Operační systém pevného reálného času (hard real time) je obecně takový systém, jehož správný provoz závisí nejen na výsledcích, které z něj vzejdou, ale i na tom, kdy je získáme. Je důležité podotknout, že aktivita v pevném reálném čase neznamená nutně rychle - spíše ukazuje jak deterministické jsou časové charakteristiky reakcí systému. Jinými slovy, důležitým výkonovým hlediskem není průměrný čas reakce, ale čas reakce v tom nejhorsím případě, přitom požadavek na stanovení času reakce je absolutní, bez výjimek.

K požadavku na časový determinismus přistupují ještě další požadavky, které systémy reálného času typicky poskytují:

- preemptivní plánovač s velkým počtem nastavitelných priorit vláken,
- prediktivní mechanismus synchronizace vláken,
- systém dědičné priority,
- rychlé hodiny a časovače,

Rozšíření Windows na systém reálného času

Windows NT/2000/XP nejsou v základním provedení operačními systémy reálného času. Byly vytvořeny jako operační systémy pro obecné účely, vhodné jak pro interaktivní systém na počítačích typu desktop, tak i jako serverové systémy v síti. Nedostatky Windows 2000/XP spočívají zejména v následujících příčinách:



Spolupráce OS Windows a RTX

- příliš málo priorit vláken,
- neprůhledná a nedeterministická rozhodnutí plánovače,
- vznik inverze priorit, zvláště při zpracování přerušení.

Ani Windows XP Embedded, které jsou v podstatě Windows XP s utilitou pro nakonfigurování podle použitého hardware, nejsou operačním systémem reálného času. Teprve po nainstalování doplňku RTX (Real-Time eXtension), od společnosti VenturCom se Windows 2000/XP začnou chovat jako opravdový systém reálného času. Rozšíření RTX přidává plánování úloh v reálném čase a zkracuje rozlišitelnou jednotku času z 5 ms přibližně na 20 μ s. Stručně řečeno, doplněk RTX je implementován jako kolekce knihoven, subsystém reálného času (RTSS). Knihovny poskytují přístup k subsystému prostřednictvím funkcí RTX API (Real-Time API), které tak tyto objekty zpřístupňují.

Za pozornost stojí, že RTX API lze vyvolat jak ze standardního prostředí Win32, tak i v rámci RTSS. Při používání RTX API z Win32 však není k dispozici determinismus poskytovaný v rámci RTSS. Vše, co je nutné ke konverzi programu používající Win32 do programu používající RTSS – je znovu slinkovat aplikaci s jinou kolekcí knihoven.

Přidáním RTX k Windows 2000/XP se okna začnou chovat jako opravdový systém reálného času. Nezávislý plánovač vláken RTSS v RTX může běžet v popředí i při obsluze vlákna přerušení Windows a může být používán jak na jednoprocessorovém, tak na symetrickém víceprocessorovém stroji, a to beze změny jádra Windows a ovladačů zařízení. Aplikace pak také nemusí při startu či restartu počítače čekat, než se spustí operační systém (to znamená, že může být spuštěna za méně než 3 s od startu počítače). K RTX patří i podpůrné nástroje a doplňky – TimeView, Platform Evaluator a RT TCP/IP.

- TimeView je určeno pro časovou analýzu vzniklých událostí. V přirozeném pořadí s časovými značkami zaznamenává volitelné systémové a aplikační události. Ukazuje sled spuštění a časování událostí, přerušování, přepínání kontextů vláken a vzájemné interakce aplikací.
- Platform Evaluator umožňuje dokumentovat a charakterizovat schopnosti reálného času použité hardwarové platformy. S jeho pomocí lze určit optimální konfigurace hardwaru pro dosažení vlastností reálného času.
- RT TCP/IP umožňuje použít vysoce výkonné síťové protokoly reálného času do aplikace v prostředí Windows + RTX. Umožňuje pokračování běhu síťových aplikací i při obsluze systémové výjimky OS Windows (Blue Screen).

PharLap ETS – kompaktní operační systém s Win32 API

Ačkoliv se může zdát, že je jen málo rozdílů mezi víceúlohovými a víceprocesními systémy, je vhodné si oba pojmy trochu ozřejmit. Každý proces je složen z mnoha vláken a každé vlákno může mít vlastní prioritu pro plánování svého provádění (ve PharLap ETS jsou všechna uživatelská vlákna složkou jediného procesu jádra). Vlákna v procesu sdílejí stejný adresní prostor, globální proměnné, paměťovou hromadu (heap) a proměnné prostředí. Každé vlákno má vlastní zásobník registrů a priority. PharLap ETS je operační systém, který používá pouze vlákna k provádění více úloh v rámci jednoho procesu, a tudíž má nejmenší režii pro přepínání kontextů úloh. Proto je zde zapotřebí nejmenší množství registrů a dalších informací, jež musí být ukládány a obnovovány v průběhu přepínání mezi úlohami. Tato skutečnost dělá z PharLap ETS neobyčejně rychlý a současně hardwarově nenáročný operační systém pevného reálného času.

PharLap ETS je navržen pro „střed trhu“ – poskytuje síťové možnosti, podporuje internetový server i GUI (grafické rozhraní). V rámci systémů reálného času se vyznačuje nejlepším poměrem schopností reálného času k ceně systému. Pro mnohé platformy

může být operační systém PharLap ETS spuštěn i bez BIOSu, a to napsáním vlastního spouštěcího kódu.

Plánování běhu programových vláken (scheduling) je deterministické, tzn. že „poběží“ vždy povolené programové vlákno s nejvyšší prioritou a bude aktivní dokud buď není zablokováno jiným právě povoleným vláknem s vyšší prioritou, anebo, je-li povoleno časové sdílení, vypršel sdílený čas a je zde jiné povolené vlákno se stejnou prioritou. Plánovač běhu programových vláken brání vzniku chybového efektu prioritní inverze zajištěním běhu vlákna s nejvyšší prioritou tak, že se snaží uvolnit blokující vlákno dynamickým zvyšováním priorit.

Frekvence, s níž jádro ETS přepíná mezi vlákny se stejnou prioritou – časové sdílení procesoru (slicing), je řízena podle délky časového sdílení procesoru každým vláknem. Standardně je časové sdílení nastaveno na 10 ms. Omezení periody „tiků“ systémového časovače závisí na použitém hardware. Pro systémy kategorie PC/AT se tato perioda pohybuje mezi 1 až 55 ms. V jádře ETS jsou ovšem i funkce pro stanovení impulzů systémových hodin ve zlomcích milisekund.

Příklady konkrétního použití

Windows + RTX

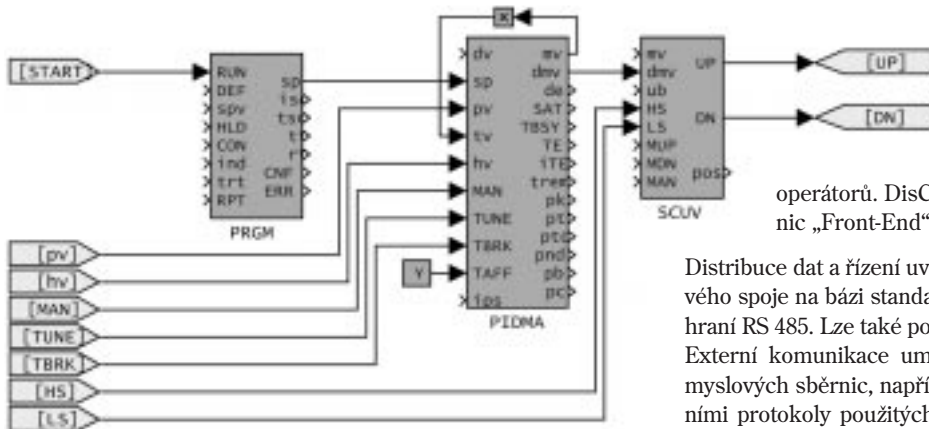
RTX zavádí do Windows subsystém reálného času RTSS a přináší schopnost provádět řídicí operace na průmyslové úrovni, v pevném reálném čase. Dovoluje nahradit funkce DSP (Digital Signal Processor) a jiných pomocných koprocesorů (tzv. „smart“ karet) pro funkčnost reálného času, čistě softwarovým řešením v rámci vyvíjeného programu.

S příchodem dnešních výkonných a cenově dostupných procesorů s taktovací frekvencí přes 1 GHz se dají funkce zajišťované koprocesory jednoduše nahradit softwarovými úlohami v časově kritickém prostředí RTSS, které „běží“ současně se standardními aplikacemi. Toto řešení uspoří značné prostředky, například za ušetřené „smart“ karty tj. V/V karty s vlastním CPU. Ty lze nahradit běžnými kartami, které jsou řízeny v reálném čase přímo z cílové aplikace.

Jiným příkladem je realizace směrovače VoIP (Voice over IP – hlasové služby v prostředí internetu) s rozšířenými funkcemi. Tato implementace komunikačního zařízení, poskytuje současně několik funkcí. Řekněme, že systém musí pro dvě spojení LAN realizovat směrovač, který garantuje kvalitu služby (QoS) a běh aplikačního serveru. Windows XP poskytují tyto služby skoro na všech platformách, avšak realizace směrovače QoS vyžaduje zvýšenou pozornost. Windows XP sice dokáží zabezpečit softwarové směrování, ale nemohou zaručit dobu odezvy, speciálně v případě vysokého zatížení RAS (Remote Access Server) nebo zatížení od aplikačního serveru. Opět existují dvě možná řešení: jedno spočívá v přidání směrovací „smart“ karty s QoS, která provádí směrování „routování“ mezi dvěma spojeními LAN. Druhá možnost spočívá v naprogramování QoS a směrování v reálném čase s pomocí RTX přímo v hostitelském CPU. Pod Windows XP může běžet aplikace serveru a veškeré služby RAS a procesy VPN, zatímco pod RTSS, prioritně před všemi procesy Windows XP, jsou realizovány NIC-ovladače a samotné směrování. Výhodou tohoto přístupu je nižší cena, jelikož není potřeba žádný dodatečný speciální hardware, ale stačí nám obyčejné dvě „hloupé“ síťové karty.

PharLap ETS

Jak bylo výše uvedeno, PharLap ETS používá vlákna pouze k provádění více úloh v rámci jednoho procesu, takže se jedná o jednoprocessní operační systém. Tato vlastnost ho předurčuje pro použití v dedikovaných (jednúčelových) aplikacích, například pro zabudování (embedded) řídicí jednotky do strojů nebo je velmi vhodné použít PharLap ETS na realizaci pokročilých algoritmů řízení pro průmyslové regulátory.



Obr.2 Regulační smyčka se samonastavujícím se regulátorem a programátorem řídicí veličiny

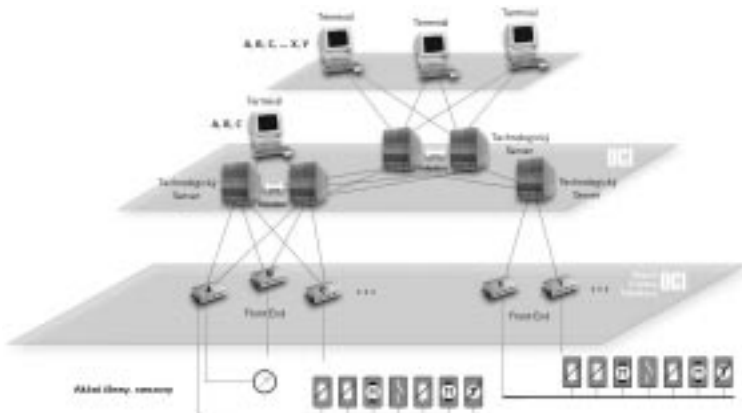
Pod operačním systémem PharLap ETS může být například provozován řídicí systém REX popsáný v článku „Nový řídicí systém REX podporuje návrh a simulaci v prostředí Matlab/Simulink“ v předchozím čísle tohoto časopisu. Následující obrázek naznačuje jednoduchost, s jakou je zde možné konfigurovat pokročilé regulační algoritmy. Uvedené funkční schéma realizuje regulační smyčku se samonastavujícím se regulátorem a programátorem řídicí veličiny.

Projekty na bázi Win32 API

Protože PharLap ETS disponuje rozsáhlou a pečlivě vybranou podmožinou funkcí Win32 API, není příliš obtížné napsat takový zdrojový kód, který půjde překládat, linkovat a výsledný produkt spouštět jak na platformě ETS, tak na platformě Windows. Díky sadě funkcí RtWinAPI, kterou obsahují Windows CE, PharLap ETS i RTX jsou pak zdrojové texty časově kritických rutin mezi těmito platformami navzájem přenositelné.

Jako příklad aplikace, která využívá těchto vlastností, může být distribuovaný řídicí systém. Ten využívá specifické možnosti různých hardwarových a softwarových platform, a přitom je všude schopen pracovat s jádrem, které je napsáno jediným zdrojovým textem. Konkrétním příkladem realizace projektu na bázi Win32 API je distribuovaný (topologicky rozložený) řídicí systém. Takový systém je například DisCO, které pracuje na principu rozloženého zpracování dat a decentralizace řízení. Název systému vyjadřuje použití objektové technologie v distribuovaném prostředí: Distributed Computing Objects.

Po softwarové stránce DisCO tvoří balík spolupracujících programů pro libovolné počítače s procesory IBM x86, od 386 SX k Pentiu a potřebný vstupní/výstupní hardware. Výběr fyzického provedení je plně v rukou systémového aplikátora; co použije ze široké nabídky hardware pro aplikaci typu embedded, industry nebo desktop. Operační systém – jako platforma, na které funguje jednotné systémové jádro, je volitelná od Windows CE,



Obr.3 Schéma distribuovaného řídicího systému DisCO

Windows XP Embedded s nebo bez doplňku reálného času RTX nebo PharLap ETS pro stanice „Front-End“, Windows 2000/XP s nebo bez RTX pro „Technologické servery“ a Windows 2000/XP pro terminály operátorů. DisCO používá regulátory REX na úrovni stanic „Front-End“ nebo technologického serveru.

Distribuce dat a řízení uvnitř systému je podmíněna existencí síťového spoje na bázi standardu ethernetu nebo komunikačního rozhraní RS 485. Lze také použít prostředí sítě internet nebo intranet. Externí komunikace umožňuje připojovat zařízení pomocí průmyslových sběrnic, například sběrnice Profibus nebo komunikačními protokoly použitých zařízení. Export dat do jiných řídicích systémů se provádí komunikačním standardem OPC. Řídicí systém „DisCO“ je k dispozici jako software ve spustitelné binární formě. exe nebo jako dynamické knihovny. dll.

Závěr

RTX jako doplněk reálného času od americké firmy VenturCom prokazuje, že s vybranou škálou rozšíření je možné vylepšit systém Windows 2000/XP nebo embedded NT/XP tak, aby měl vlastnosti operačního systému pevného reálného času a zároveň s tím byl nadále používán jako univerzální platforma. Výsledný systém vyhovuje tlakům determinizmu, které jsou nutnou součástí světa reálného času a současně poskytuje prostředí, které je blízké širokému spektru uživatelů. Použití řešení umožňuje vystavět takové systémy, které mají nepřekonatelný poměr cena/výkon. Doplněk RTX je vhodný jak pro použití ve vestavěných (embedded) systémech, tak i (a to především) pro výkonné high-end aplikace.

Operační systém PharLap ETS je obecně považován za nejkompaktnější operační systém reálného času postavený na bázi Win32 API pro vestavěné aplikace. Jeho portace na 32-bitový hardware kompatibilní s IBM x86 a Pentium je bez problémů a celkové použití je snadné. Přestože se nejedná o produkt společnosti Microsoft, používá vývojové prostředí MS Visual C++, které spolupracuje s linkovacím programem LinkLoc ze softwarového „balíku“ ETS SDK a vytváří tak exekutivní mikrojádro, schopné spuštění dokonce i v paměti ROM.

Je až překvapující, že během deseti let se podařilo změnit pomalá, neefektivní a rozměrná PC na vysoce výkonné, efektivní embedded systémy pracující v reálném čase, které umožňují realizaci všech druhů časově kritických aplikací. Zároveň s hardwarem se vyvíjí i software, aby dokázal splnit všechny nové náročné požadavky. Krok směrem k univerzálním systémům a jejich dalším aplikacím je však stále v začátcích.

Podrobnější informace o zmiňovaných operačních systémech reálného času s Win32 API najdete na internetové adrese www.datapartner.cz.