

# Návrh časově kritických systémů II: úlohy reálného času

Z úvodního článku [4] ze série článků na téma návrhu časově kritických systémů (systémy reálného času, *Real-Time Systems, systémy RT*) víme, že jsou-li požadované vlastnosti systému odvoditelné z jeho specifikace, tj. že mezi požadovanými vlastnostmi a specifikací neexistuje rozpor, je možné bez jakýchkoliv obav přistoupit k další etapě vývoje – vývoje cyklu systému RT. Touto etapou je zpravidla převod formální specifikace systému na tzv. množinu úloh RT.

## Úloha RT

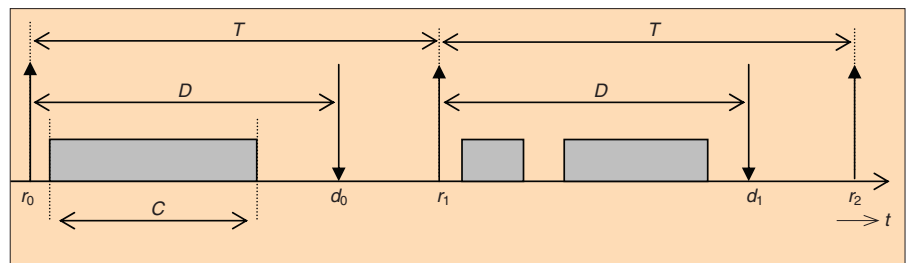
Každou z úloh RT (dále jen „úloh“) si lze představit jako výpočetní jednotku provádějící specifickou funkci – např. čekání na vstupní podnět, vzorkování či transformaci veličin, odměřování času, interakci s okolím nebo generování odezv. Jinými slovy – úloha je zodpovědná za správné provedení konkrétní dílčí části algoritmu řízení v reálném čase. Podle toho, jak je možné charakterizovat intervaly mezi příchody požadavků na zahájení úloh, lze každou z úloh zařadit do jedné z těchto kategorií:

- *periodická*: požadavky přicházejí s periodou  $T$ ,
- *aperiodická*: intervaly mezi příchody požadavků nelze nijak ohraničit,
- *sporadická*: je známa nejkratší možná doba mezi příchody požadavků.

## Model úlohy RT

Převodem specifikace systému na množinu úloh RT se sice přiblížíme cílové realizaci systému RT, avšak nesmíme zapomenout, že reprezentace systému na úrovni úloh stále abstrahuje jak od prostředků cílového operačního systému reálného času

(*Real-Time Operating System – RTOS*), tak od vlastností cílové platformy. Nicméně, některé (z pohledu systému RT) významné



Obr. 1. Ilustrace k plánu a základním parametrům úloh

parametry zanedbat nelze, a naopak je nutné je zahrnout do tzv. *modelu úlohy RT*. K základním takovým parametrům patří např.  $r$  – absolutní čas příchodu požadavku na provedení úlohy (čas vyvolání úlohy),  $T$  – perioda mezi příchody požadavků na provedení úlohy (pouze pro periodické úlohy),  $C$  – nejdelší z možných dob běhu úlohy,  $D$  – relativní časová mez odezvy úlohy,  $d$  – absolutní časová mez odezvy úlohy či  $P$  – priorita úlohy.

Pro úplnost dodejme, že úloha RT je obvykle označována písmenem řecké abecedy  $\tau$ ,

množina úloh RT písmenem  $\Gamma$ . Parametry jsou úloze obvykle přiřazeny formou uspořádané  $n$ -tice uvedené v závorce u symbolu  $\tau$ , tj. např.  $\tau_1(r, C, D, T) = \tau_1(0, 2, 100, 35)$  pro periodickou úlohu či  $\tau_4(r, C, D) = \tau_4(4, 5, 7)$  pro neperiodickou úlohu.

Vedle již uvedených základních statických (v čase neměnných) parametrů úloh je možné zpřesnit model úloh zahrnutím dalších statických či dynamických (v čase proměnných)

parametrů. Jistě platí, že čím větší počet parametrů model úloh zahrnuje, tím více se blíží chování navrhovaného systému RT v reálných podmínkách. Naproti tomu je však nutné si uvědomit, že s počtem parametrů rostou také požadavky kladené nejen na konstrukci plánovače, ale i na analýzu vlastností systému RT. Přehled často používaných parametrů úloh je v tab. 1.

## Plán

Uspořádání provádění úloh v čase se nazývá plán (lze použít i označení rozvrh, v tomto a souvisejících článcích se však přidržíme označení „plán“ – pozn. aut.) a obvykle je zobrazováno v podobě časového diagramu, jak ilustruje obr. 1. Časy  $r$ , popř.  $d$ , bývají v diagramu zvýrazněny použitím symbolů  $\uparrow$ , popř.  $\downarrow$ . V souvislosti s plány uvedme alespoň tyto pojmy:

- plán je označován jako přípustný právě tehdy, když jsou v něm dodrženy časové meze všech úloh,
- množina úloh je plánovatelná právě tehdy, když pro ni lze nalézt přípustný plán,
- mechanismus generující plán je optimální na množině úloh právě tehdy, když je schopen pro libovolnou její podmnožinu nalézt přípustný plán,
- postup umožňující rozhodnout, zda množina úloh je, či není plánovatelná, se nazývá test plánovatelnosti,
- postup umožňující rozhodnout, zda přidáním nové úlohy, vzniklé na základě nové

Tab. 1. Základní statické a vybrané další parametry úloh RT (CPU – procesorová jednotka)

Parametr	Symbol	Vztah mezi parametry
priorita	$P$	
úloha	$\tau$	
množina úloh	$\Gamma$	
absolutní čas vyvolání úlohy	$r$	
nejdelší doba běhu úlohy	$C$	
relativní časová mez odezvy úlohy	$D$	
perioda úlohy	$T$	$C \leq T \leq D$
absolutní časová mez odezvy úlohy	$d$	$d = r + D$
relativní doba volnosti úlohy	$L$	$L = D - C$
činitel využití CPU úlohou	$u$	$u = C/T$
činitel zatížení CPU úlohou	$v$	$v = C/D$
kolísání dob volání úlohy	$J$	
doba blokování úlohy	$B$	
doba zbývající k dokončení úlohy v čase $t$	$C(t)$	
doba zbývající k dosažení časové meze odezvy úlohy v čase $t$	$D(t)$	$D(t) = d - t$
doba volnosti úlohy v čase $t$	$L(t)$	$L(t) = D(t) - C(t)$
činitel zatížení CPU úlohou v čase $t$	$v(t)$	$v(t) = C(t)/D(t)$

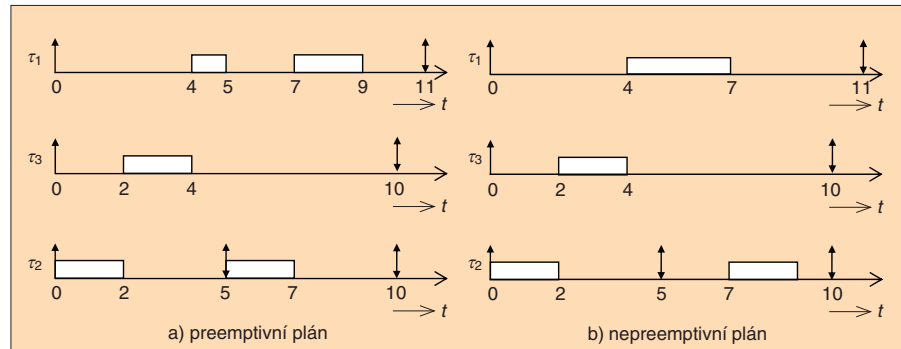
příchozího požadavku, do stávající plánovatelné množiny úloh zůstane tato množina plánovatelná, se nazývá test přijetí; např. při distribuovaném plánování může nepřijetí dané úlohy do plánu jednoho uzlu vést k přijetí úlohy do plánu jiného uzlu, tj. k migraci úlohy.

### Plánovač

Plán je generován tzv. *plánovačem*, který rozhoduje o tom, které z úloh bude v daném čase přidělen čas procesorové jednotky. Hlavní funkcí plánovače je volit pořadí provádění úloh tak, aby každá z úloh byla dokončena nejpozději do uplynutí její časové meze, tj. času odpovídajícího hodnotě parametru  $d$  dané úlohy. Téměř všechny moderní plánovače jsou *prioritní*; tj. o tom, které úloze bude přidělen čas procesoru, rozhodují na základě priorit přiřazených jednotlivým úlohám, přičemž v ideálním případě plánovač garantuje, že v daném časovém okamžiku vždy běží ta z úloh připravených k běhu, jejíž priorita je nejvyšší. Při vykonávání tohoto rozhodnutí je důležité, zda je plánovač *preemptivní* či *nepreemptivní* – zatímco preemptivní plánovač zajistí přepnutí na nejvyšší úlohu přerušením provádění momentálně běžící úlohy, nepreemptivní plánovač přepnutí odloží až do okamžiku jejího ukončení. Jelikož doba odezvy preemptivních systémů je kratší než u nepreemptivních systémů, jsou plánovače používané v RTOS téměř výhradně preemptivní. Avšak mnohé RTOS obsahují i prostředky umožňující dočasně využívat vlastnosti nepreemptivního plánovače, většinou při použití služeb typu zamknutí/odemknutí plánovače, a oba mechanismy tak kombinovat. K hlavním nedostatkům preemptivního plánovače patří režie spojené s přepínáním kontextu úloh a s ukládáním kontextu každé z úloh. Tato režie roste úměrně s počtem úloh a s počtem přepnutí úloh v rámci daného časového intervalu. Podstatnou část této režie tvoří složky *zpoždění přerušení*, *odezva přerušení* a *zotavení se z přerušení*, které, spolu s dobou provádění příslušných obsluh přerušení, hrají důležitou roli při určo-

vání nejdelší doby odezvy úloh. Navíc může být v rámci obsluhy přerušeni do systému zařazena úloha s významnější prioritou, než jakou měla úloha přerušena touto obsluhou. V takovém případě je doba odezvy přerušené úlohy ještě delší.

K názorné ilustraci rozdílu mezi preemptivním a nepreemptivním způsobem plánování úloh RT předpokládáme množinu  $\Gamma$



Obr. 2. Ilustrace k rozdílu mezi preemptivním a nepreemptivním plánováním

tvořenou podle obr. 2 úlohami  $\tau_1, \tau_2, \tau_3$  s parametry  $\tau_1(r_1, C_1, D_1, T_1) = \tau_1(0, 3, 11, 11)$ ,  $\tau_2(r_2, C_2, D_2, T_2) = \tau_2(0, 2, 5, 5)$  a  $\tau_3(r_3, C_3, D_3, T_3) = \tau_3(0, 2, 10, 10)$ . Pro jednoduchost předpokládáme, že nejvyšší priorita je přiřazena úloze  $\tau_2$ , střední úloze  $\tau_3$  a nejnižší úloze  $\tau_1$ . Pro úplnost poznamenejme, že plán systému s několika úlohami bývá zobrazován po tzv. hladinách (úrovních) priorit, tj. od nejvyšší prioritní úrovně (v našem případě spodní časový průběh pro  $\tau_2$ ) po nejnižší (v našem případě horní časový průběh pro  $\tau_1$ ).

I z uvedených jednoduchých plánů lze vyvodit tyto závěry:

- pořadí provádění úloh je závislé na prioritách přiřazených úlohám, tj. volba mechanismu přiřazování priorit určuje, kdy bude dané úloze přidělen čas procesoru,
- preemptivní systémy garantují kratší dobu odezvy prioritně významnějších úloh než systémy nepreemptivní,
- počet přepnutí mezi úlohami je větší u preemptivních systémů než u nepreemptivních.

### Závěr

Následující článek bude zaměřen na základní mechanismy přiřazování priorit v preemptivních systémech RT. Budou představovat typičtí zástupci statických a dynamických mechanismů, jejich přednosti a nedostatky a s nimi související testy plánovatelnosti množin úloh RT.

### Poděkování

Článek vznikl za podpory výzkumného záměru MSM0021630528 – Výzkum informačních technologií z hlediska bezpečnosti (agentura CEZ MŠMT) a projektu specifického výzkumu FIT-S-10-1 (VUT v Brně).

### Literatura:

[1] CHENG, A. M. K.: *Real-Time Systems: Scheduling, Analysis, and Verification*. Wiley, 2002, 552 s., ISBN 0-471-18406-3.  
 [2] COTTET, F. – DELACROIX, J. – KAISER, C. – MAMMERI, Z.: *Scheduling in Real-Time Systems*. John Wiley & Sons, 2002, 266 s., ISBN 0-470-84766-2.  
 [3] JOSEPH, M.: *Real-Time Systems Specification, Verification and Analysis*. Prentice Hall, 1996, 278 s., ISBN 0-13-455297-0.  
 [4] STRNADEL, J.: *Návrh časově kritických systémů I: specifikace a verifikace*. Automa, 2010, roč. 16, č. 10, s. 42–44, ISSN 1210-9592.

Ing. Josef Strnadel, Ph.D.,  
 Fakulta informačních technologií,  
 Vysoké učení technické v Brně  
 (strnadel@fit.vutbr.cz)

### ► Kamerový systém pro pražský okruh

Dne 20. září byl zahájen provoz na dlouho očekávané jihozápadní části silničního obchvatu okolo Prahy. Zatěžkávací zkoušce je nyní podrobován i kamerový systém, na jehož realizaci se podílely společnosti Schneider Electric (www.schneider-electric.cz) a Integoo (www.integoo.cz).

Kamerový systém sleduje a vyhodnocuje aktuální stav dopravy na dálničním okruhu, a výrazně tak přispívá k bezpečnosti při průjezdu dvou hlavních tunelů. Základem systému realizovaného firmou Integoo jsou otočné, na dálku říditelné kamery Pelco Esprit. Ty se již používají v mnoha dopravních systémech v České republice, ale i po celém světě. Mají možnost pohledu nad horizont v úhlu až 33°, integrovaný, dálkově ovládaný stěrač předního skla a robustní pro-

vedení, které zajistí funkčnost kamery i při větru o rychlosti 145 km/h. Čip CCD je velmi citlivý i při špatných světelných podmínkách. Kamera má 35násobný optický zoom a její součástí je optická stabilizace obrazu pro eliminaci otřesů způsobených projíždějícími vozidly.

Na silničním obchvatu kolem Prahy je použito 23 těchto kamer, další již byly instalovány v jiných dopravních stavbách v České republice i po celém světě. (ed)

krátké zprávy