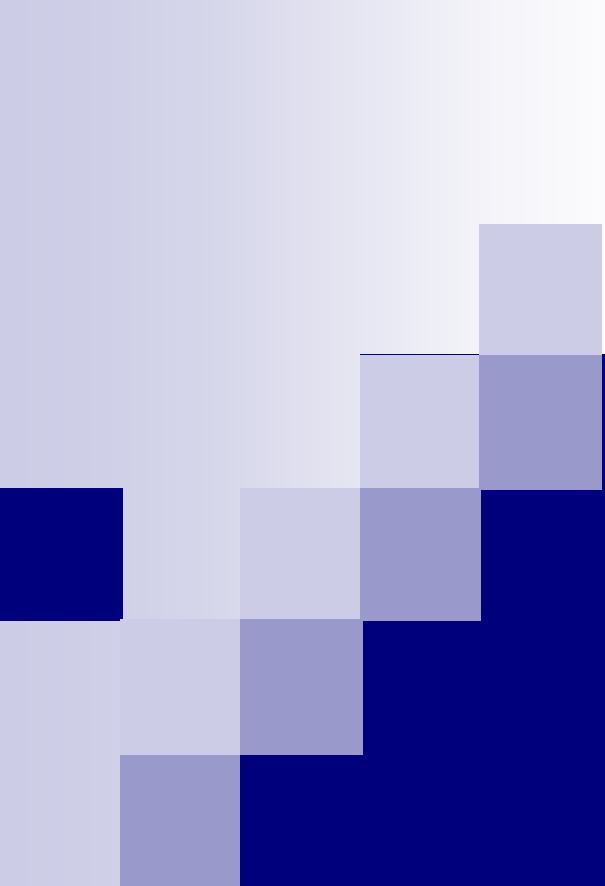




Model Checking

pro Timed Automata

Jiří Vyskočil

2011

Časově kritické systémy

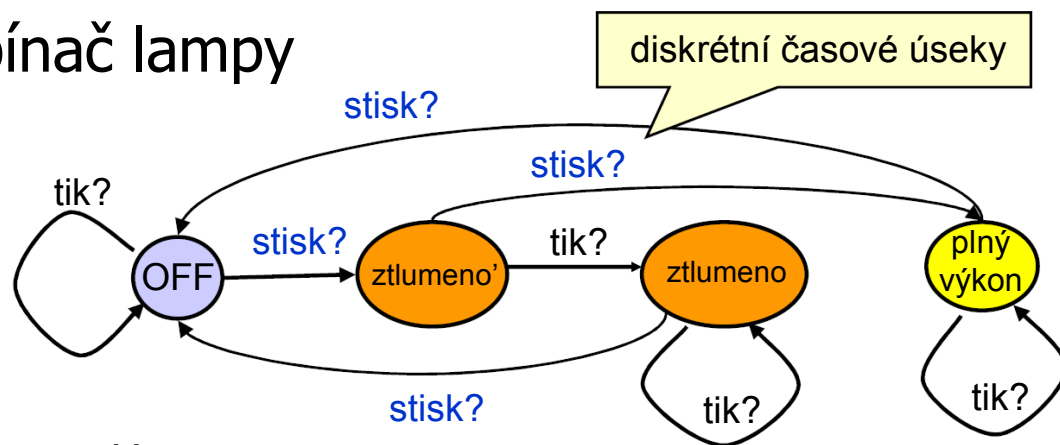
- korektnost fungování vestavěným a distribuovaným systémům závisí na:
 - správném výsledku výpočtu
 - správném načasování prováděných operací (odezva na události, ...)
- časově kritické systémy:
 - železniční přejezdy (řízení hradel, závor, ...)
 - řízení letového provozu
 - konzumní elektronika (Philips Audio Protocol, Bi-phase mark protocol, Bang & Olufsen Collision Detection Protocol, ...)
 - řízení výrobních procesů
 - ...

Diskrétní versus reálný čas

■ diskrétní model času

- časové úseky jsou namapovány na přirozená čísla
- události mohou nastat pouze v celočíselných časech
- čas se inkrementuje po násobcích nejmenšího časového úseku (označme ho tik)

■ Příklad: inteligentní vypínač lampy



- Po rychlém dvojitém stisknutí se lampa rozsvítí.
- Po jednoduchém stisknutí se ztlumí.
- Pokud je lampa rozsvícena nebo ztlumena, po jednoduchém stisknutí se vypne.

Diskrétní versus reálný čas

■ diskrétní model času

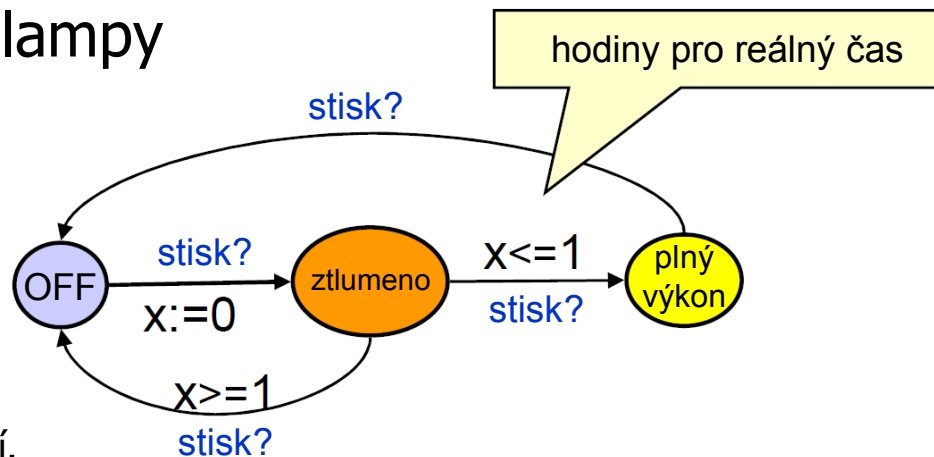
- + Událost „tik“ lze snadno modelovat pomocí modelů s konečným počtem stavů
 - + lze použít standardní model checkery a temporální logiku pro verifikaci
 - časové prodlevy jsou modelovány explicitně
 - synchronizace mezi komponentami je buď striktní nebo žádná
- => Vhodné pro modelování synchronních systémů, které se mění po pevně daných časových krocích.
- => Nevhodné pro velkou část asynchronních distribuovaných systémů. (systémy s nepřesnými lokálními hodinami, atd.)

Diskrétní versus reálný čas

■ reálný (spojitý) model času

- časové úseky jsou namapovány na kladná reálná čísla
- události mohou nastat kdykoliv v (reálném) čase
- čas se nemůže inkrementovat po násobcích nejmenšího časového úseku, protože takový úsek není.
- časovače a prodlevy se modelují pomocí proměnných reálného času (hodin)

■ Příklad: inteligentní vypínač lampy



- Po rychlém dvojitém stisknutí se lampa rozsvítí.
- Po jednoduchém stisknutí se ztlumí.
- Pokud je lampa rozsvícena nebo ztlumena, po jednoduchém stisknutí se vypne.

Diskrétní versus reálný čas

■ spojitý model času

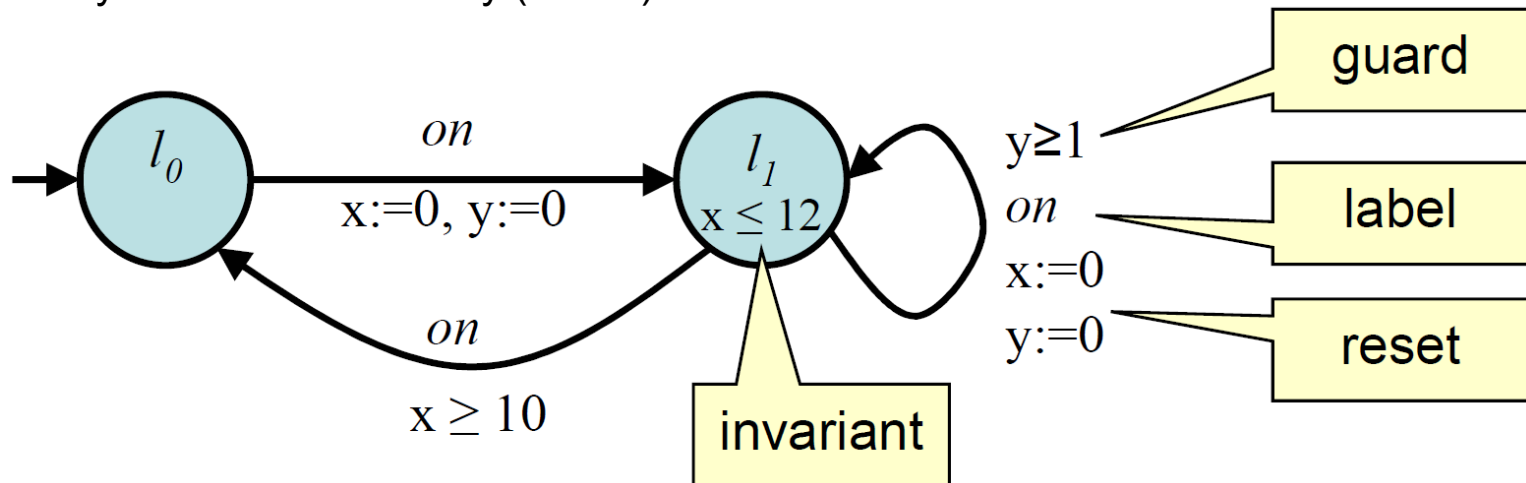
- + proměnné hodin umožňují přirozeně simulovat časovače a prodlevy
 - + prodleva (delay) je modelována implicitně
 - proměnné s reálnými hodnotami vedou k nekonečnému množství stavů
 - algoritmus pro model checking je složitější
- => Vhodné pro modelování systémů, které se synchronizují podle reálného času.
- => Zbytečně složité pro plně diskrétní systémy.

Pozor! Většina tzv. real-time systémů je typickým příkladem pro diskétní model času (operační systémy, simulátory, ...)

Timed Automata

■ Timed Automaton (časový automat)

- orientovaný graf s uzly — pozicemi (locations) a hranami (edges)
- okamžité přechody po hranách (instantaneous transitions) a časová prodleva v pozici (hrany, které vycházejí i vcházejí z/do stejné pozice)
- hodiny reálného času (real-valued clocks), které se všechny zvyšují stejnou rychlostí
- omezující podmínky — stráže (guards) na hranách
- reset hodin pro měření času mezi přechody
- invarianty v uzlech k zaručení dalšího postupu
- synchronizační značky (labels)



Timed Automata - formálně

Definice:

- typ hodiny (clocks)

typ *clock* je definován nad kladnými reálnými čísly $\mathbb{R}_{\geq 0}$

- povolené tvary podmínek (constraints)

Nechť $\mathbb{C}$ je množina proměnných typu *clock* a necht' $\Psi(\mathbb{C})$ je definováno následující gramatikou:

$$\varphi := \varphi \wedge \varphi \mid \neg \varphi \mid x \leq n \mid x < n \mid x - y \leq m \mid x - y < m$$

kde $x, y \in \mathbb{C}$, $n \in \mathbb{N}$, $m \in \mathbb{Z}$,

- ohodnocení hodin

ohodnocení hodin ν je definováno funkcí $\mathbb{C} \rightarrow \mathbb{R}_{\geq 0}^{|\mathbb{C}|}$;

pro $d \in \mathbb{R}_{\geq 0}$ je ohodnocení $\nu + d$ zobrazení hodin x na $\nu(x) + d$;

pro $r \subseteq \mathbb{C}$ a ohodnocení $\nu[r := 0]$ se x zobrazí na 0 pouze pokud $x \in r$ jinak se x zobrazí na $\nu(x)$.

Timed Automata - formálně

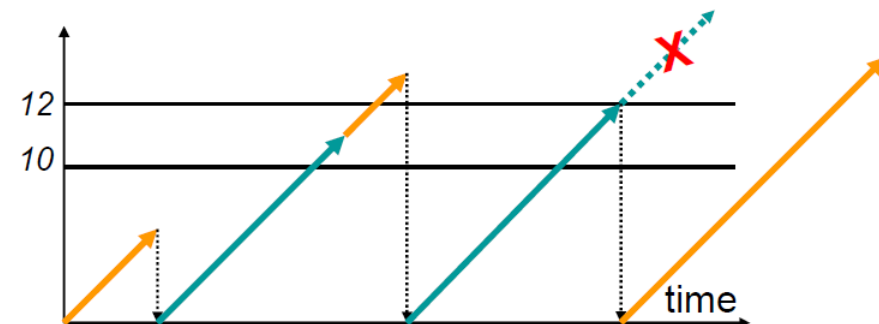
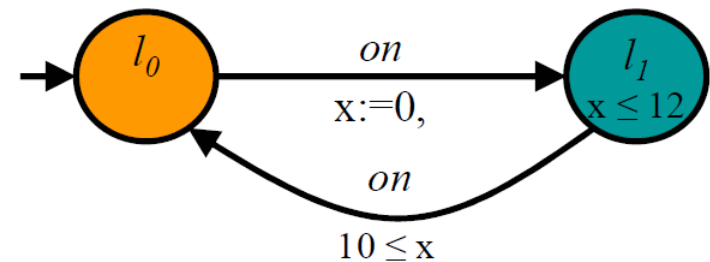
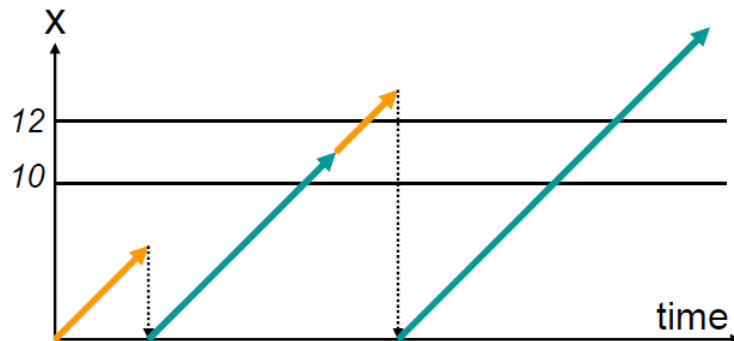
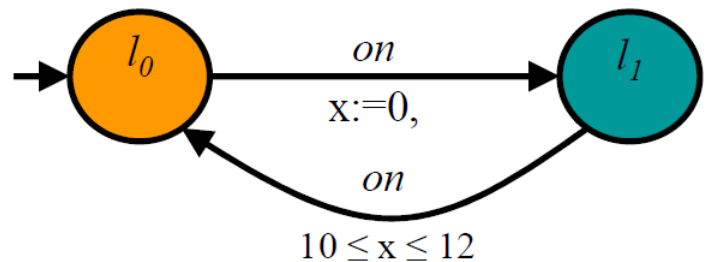
Nechť $\mathbb{C}$ je množina proměnných typu *clock*, potom **timed automaton** (časový automat) je uspořádaná pětice $(Loc, l_0, \Sigma, E, Inv)$,

kde

- Loc je konečná množina pozic (locations)
- l_0 je počáteční pozice
- Σ je množina značek (labels)
- $E \subseteq Loc \times \Psi(\mathbb{C}) \times \Sigma \times 2^{\mathbb{C}} \times Loc$ je množina hran (l, g, σ, r, l') kde
 - l je zdrojová pozice
 - g je stráž (guard) pro hodiny
 - σ je značka pro synchronizaci
 - r je množina resetů vybraných hodin z $\mathbb{C}$
 - l' je cílová pozice
- funkce $Inv: Loc \rightarrow \Psi(\mathbb{C})$ je zobrazení z pozic na invarianty

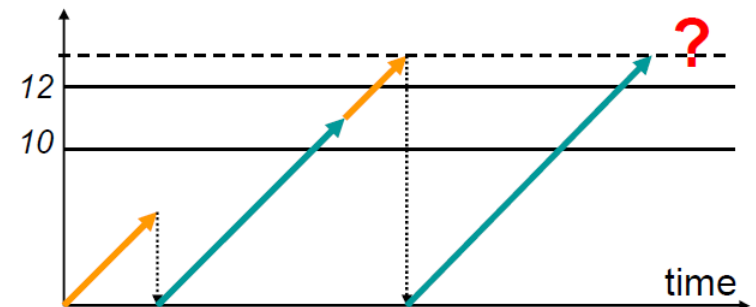
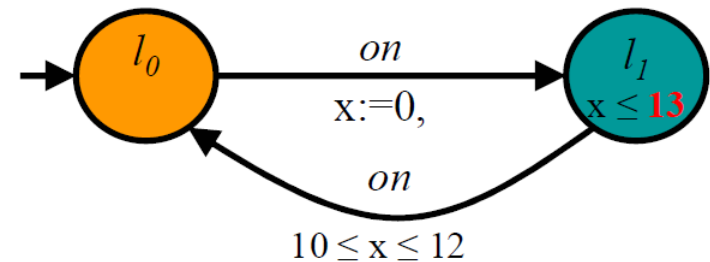
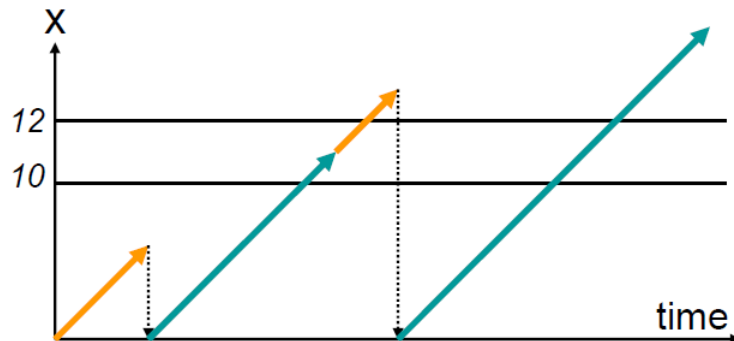
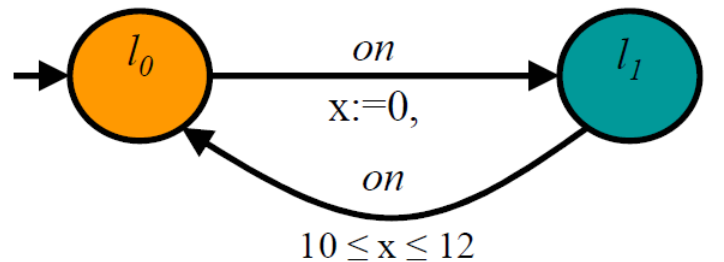
Stráže a invarianty

- *stráže* umožňují postup, *invarianty* vynucují postup



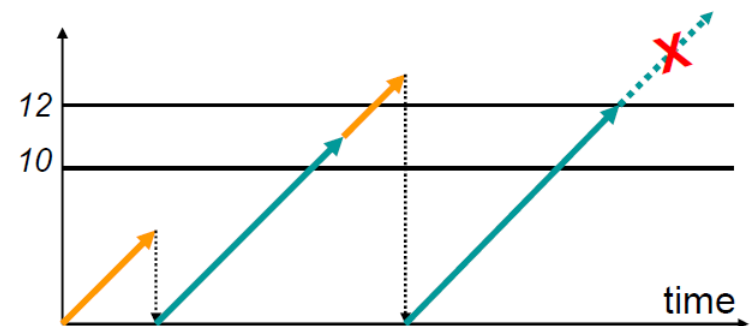
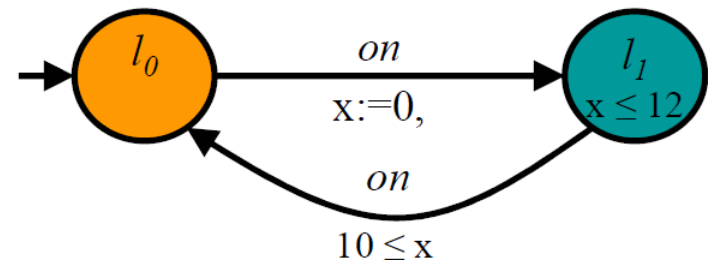
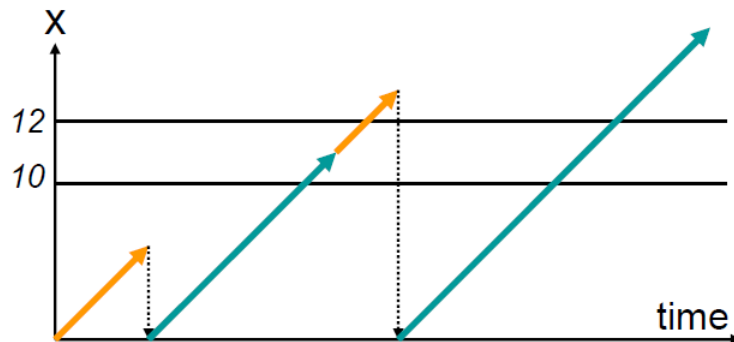
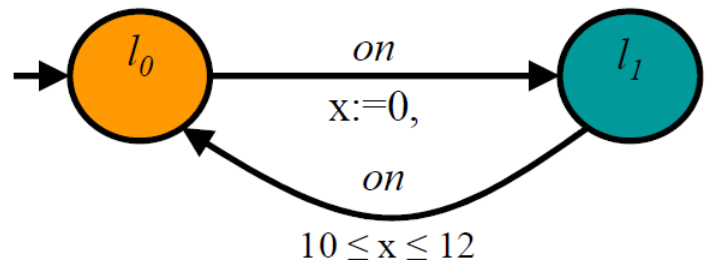
Stráže a invarianty

- *stráže* umožňují postup, *invarianty* vynucují postup
- *invarianty* mohou vést k deadlocku



Stráže a invarianty

- *stráže* umožňují postup, *invarianty* vynucují postup
- *invarianty* mohou vést k deadlocku
- Podle standardní definice *TA* jsou podmínky umožňující takový deadlock zakázány a je potřeba je upravit.



Timed Automata - sémantika

Operační sémantika pro *timed automaton* (časový automat)
(Loc, l_0, Σ, E, Inv) je definovaná pomocí časového přechodového systému
TTS (Timed Transition System) (S, s_0, R) kde

- **množina stavů** $S = \{ (l, v) \mid l \in Loc, v \models Inv(l) \}$
- **počáteční stav** $s_0 = (l_0, \mathbf{0})$
- **relace přechodu** $R \subseteq S \times (\Sigma \cup \mathbb{R}_{\geq 0}) \times S$ která obsahuje následující možnosti
 - **diskrétní přechody**
 $(l, v) \xrightarrow{\sigma} (l', v')$ pokud existuje $(l, g, \sigma, r, l') \in E$ takové, že
 $v \models g$ a $v[r:=0] = v'$
 - **časové přechody**
 $(l, v) \xrightarrow{d} (l, v+d)$ pro $d \in \mathbb{R}_{\geq 0}$ pokud pro všechny $0 \leq d' \leq d$ platí
 $v + d' \models Inv(l)$

Timed Automata – běh

- **běh** je konečná nebo nekonečná posloupnost přechodů

$$(l_0, v_0) \xrightarrow{a_0} (l_1, v_1) \xrightarrow{a_1} (l_2, v_2) \xrightarrow{a_2} (l_3, v_3) \xrightarrow{a_3} \dots$$

kde (l_0, v_0) je počáteční stav a $a_i \in \Sigma \cup \mathbb{R}_{\geq 0}$

- přechody mohou nastat v jakémkoliv bodě v reálném čase

$$(l_0, v_0) \xrightarrow{\sqrt{2}} (l_1, v_1) \xrightarrow{\text{on}} (l_2, v_2) \xrightarrow{\pi} (l_3, v_3) \xrightarrow{42} \dots$$

- V jakémkoliv časovém bodě může dojít k několika přechodům zároveň

$$(l_0, v_0) \xrightarrow{\text{on}} (l_1, v_1) \xrightarrow{0} (l_2, v_2) \xrightarrow{0} (l_3, v_3) \xrightarrow{\text{on}} \dots$$

Timed Automata – dosažitelnost

■ dosažitelnost

stav (l, v) je *dosažitelný* pokud platí

$$(l_0, v_0) \left(\xrightarrow{d} \cup \xrightarrow{\sigma} \right)^* (l, v)$$

pozice l je *dosažitelná* pokud existuje v takové, že je stav (l, v) je dosažitelný.

■ aditivita času

dva časové přechody mohou být zkombinovány

$$(l_0, v_0) \xrightarrow{d_0} (l_1, v_1) \xrightarrow{d_1} (l_2, v_2) \quad \text{právě když} \quad (l_0, v_0) \xrightarrow{d_0+d_1} (l_2, v_2)$$

$$\text{pro } d_0, d_1 \in \mathbb{R}_{\geq 0}$$

Každý konečný běh lze přepsat jako posloupnost, ve které se pravidelně střídají diskrétní přechody a časové přechody.

Timed Automata – časové pojmy

- **časová akce** (timed action) je dvojice (σ, t) kde $\sigma \in \Sigma$ a $t \in \mathbb{R}_{\geq 0}$
- **časová stopa** (timed trace) pro TA A je konečná nebo nekonečná posloupnost $(\sigma_1, t_1), (\sigma_2, t_2), (\sigma_3, t_3), \dots$ pro $t_0 \leq t_1 \leq t_2 \leq \dots$ taková, že existuje běh

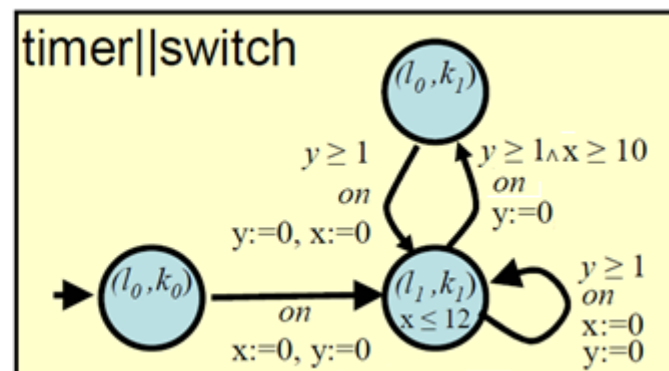
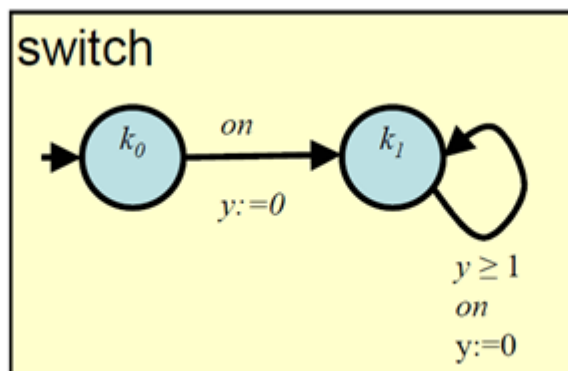
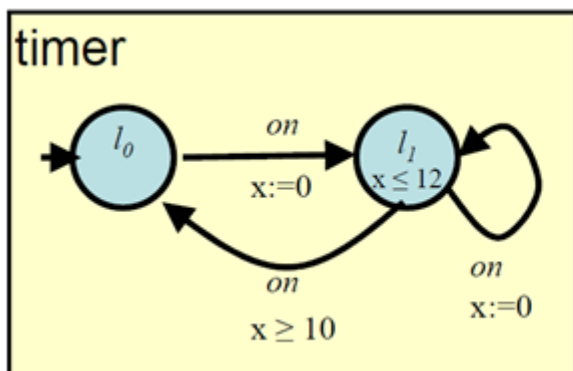
$$(l_0, v_0) \xrightarrow{d_1 \rightarrow \sigma_1} (l_1, v_1) \xrightarrow{d_2 \rightarrow \sigma_2} (l_2, v_2) \xrightarrow{d_3 \rightarrow \sigma_3} (l_3, v_3) \dots$$

kde $t_{i+1} = t_i + d_i$ pro $i > 0$ a $t_0 = 0$

Timed Automata – skládání

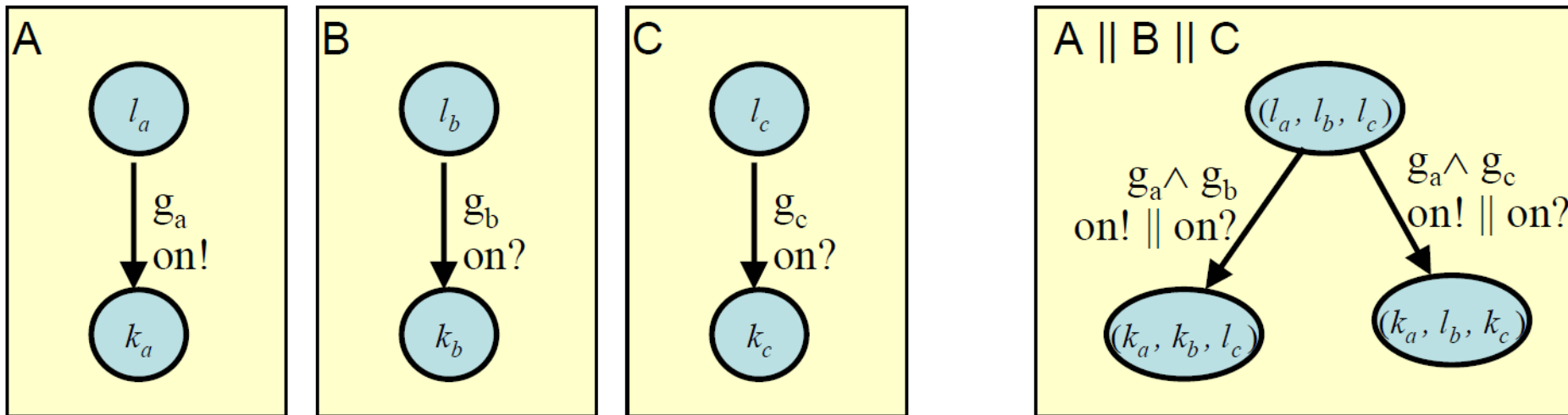
Skládání se společnými synchronizačními značkami

- množina pozic (l^1, l^2) je kartézský součin $l^1 \in Loc^1, l^2 \in Loc^2$
- invarianty v (l^1, l^2) jsou konjunkcí invariantů v l^1 a l^2
- hrany se musí synchronizovat na sdílených značkách
 - výsledná stráž je konjunkce stráží
 - výsledný reset hodin je sjednocení resetů
- hrany bez sdílených značek se mohou provádět bez synchronizace



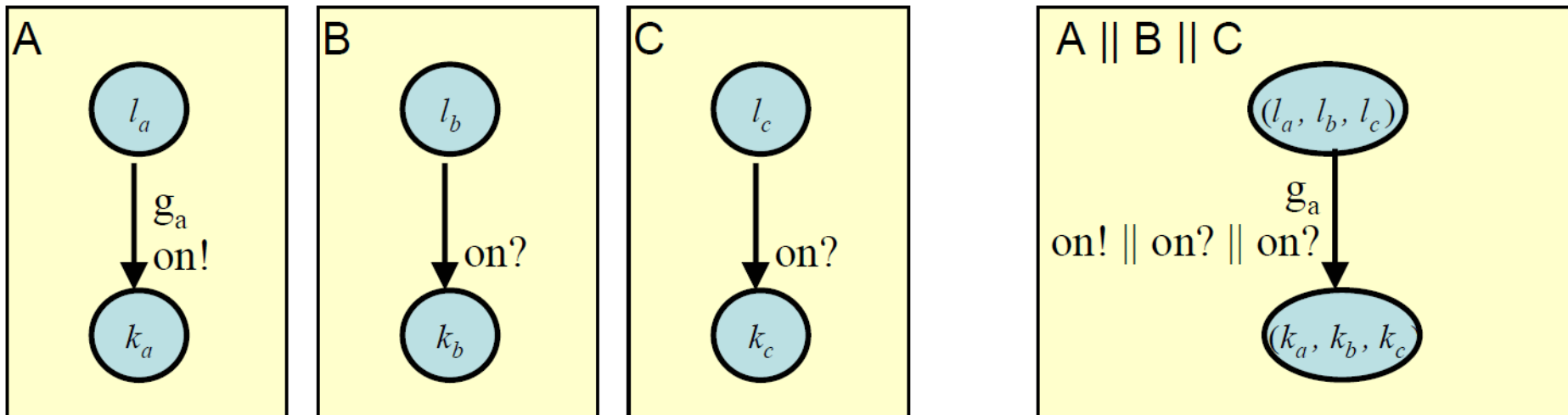
UPPAAL – synchronizace „handshake“

- V UPPAALu se synchronizace „handshake“ provádí mezi dvojicí hran se stejnojmennými značkami, kde vždy jedna končí „?“ a druhá „!“.
- obě stráže musí být splněny
- Pokud je více možností výběru, vybírá se nedeterministicky (v simulaci se uvažují všechny možnosti)



UPPAAL – synchronizace „broadcast“

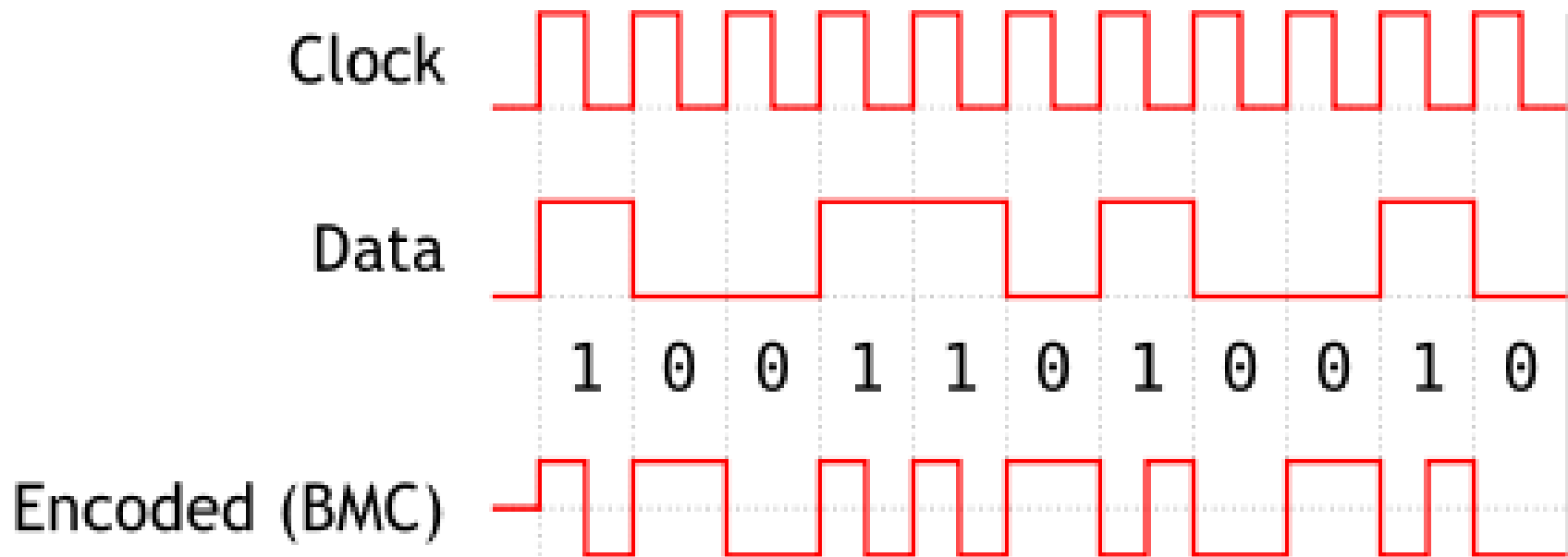
- V UPPAALu se synchronizace „broadcast“ provádí mezi několika hranami se stejnojmennými značkami, kde vždy právě jedna končí „!“ (vysílač) a ostatní končí „?“ (příjímače).
- stráž může být pouze u vysílače
- Pokud je více možností výběru vysílače, vybírá se nedeterministicky (v simulaci se uvažují všechny možnosti)



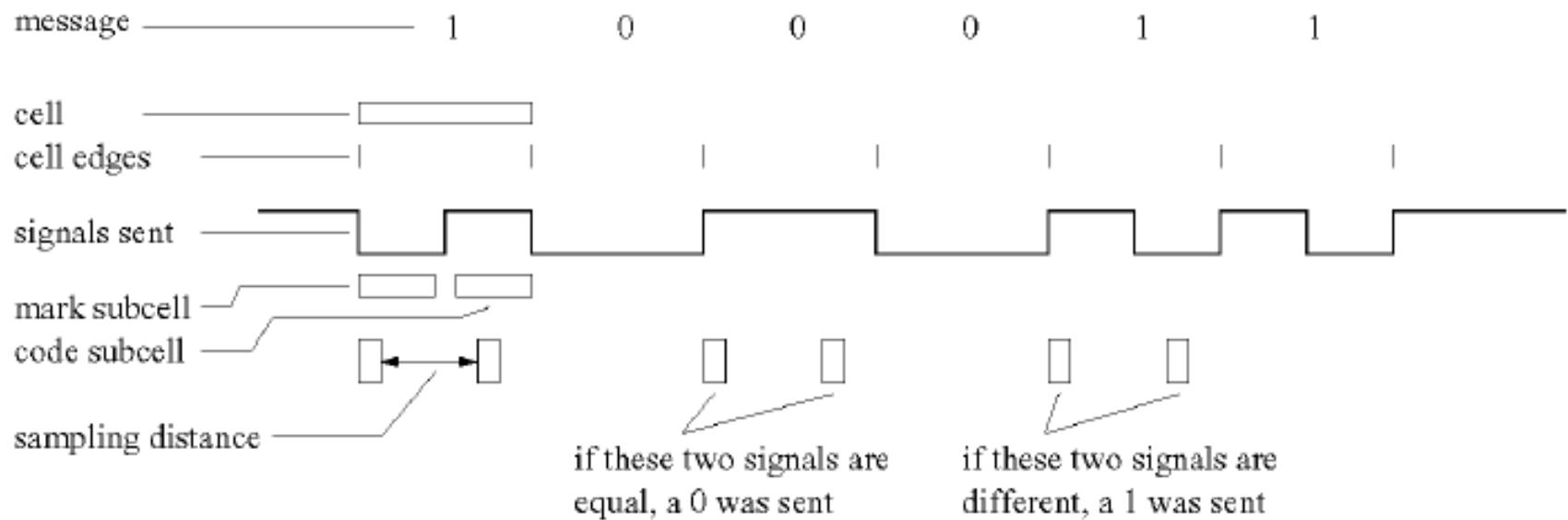
Příklad - Biphase Mark Protocol

- Kóduje datové bity a hodinový signál do jediného digitálního signálu.
- použití:
 - ☐ Intel 82530 Serial Communications Controller
 - ☐ Ethernet
 - ☐ Optical communications
 - ☐ Satellite telemetry applications
 - ☐ a další...

Biphase Mark Protocol



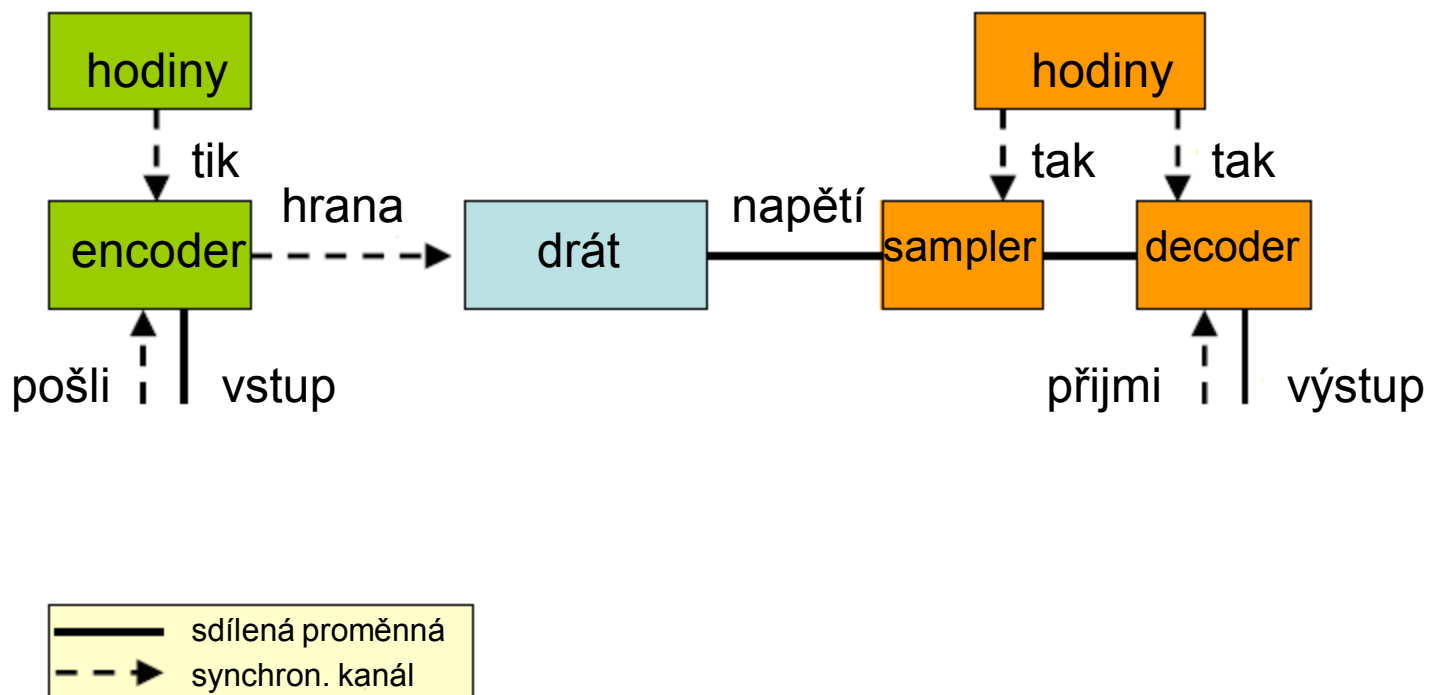
Biphase Mark Protocol



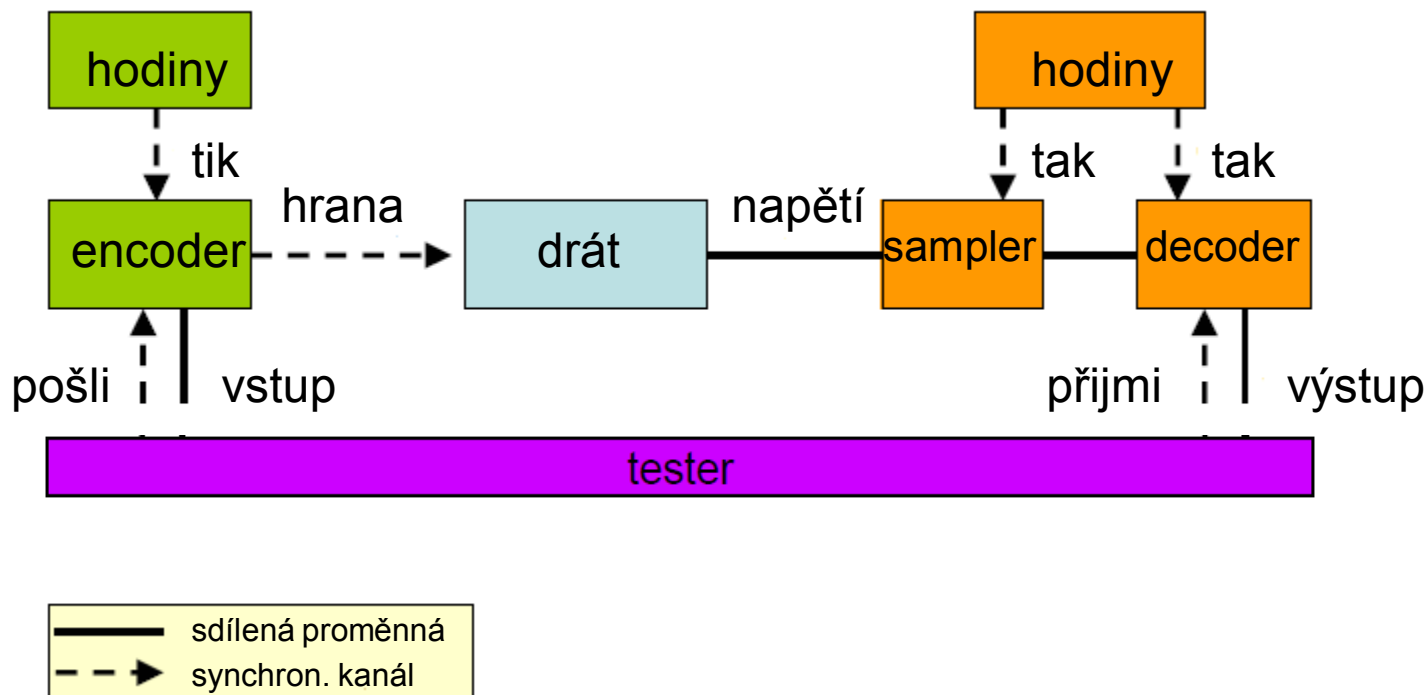
Biphase Mark Protocol

- Jaké předpoklady chceme zahrnout do modelu:
 - Vysílač i přijímač mají vlastní hodiny.
 - Obě hodiny nejsou zcela přesné ani stejné.
 - Při změně napětí může signálu trvat nějaký čas než se stabilizuje.
 - Vzorkování v době nestabilního signálu může produkovat libovolnou hodnotu.
 - Přijímač smí vzorkovat časově nedeterministicky v intervalu ohraničeném hodinovým cyklem.

BMP – schéma modelu



BMP – schéma modelu

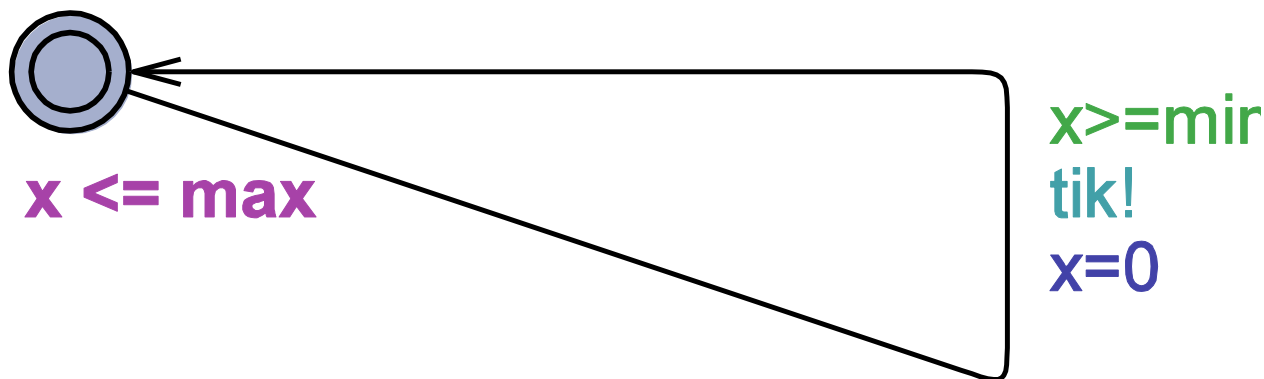


BMP – konstanty modelu

■ Konstanty modelu, které vychází z počátečních předpokladů:

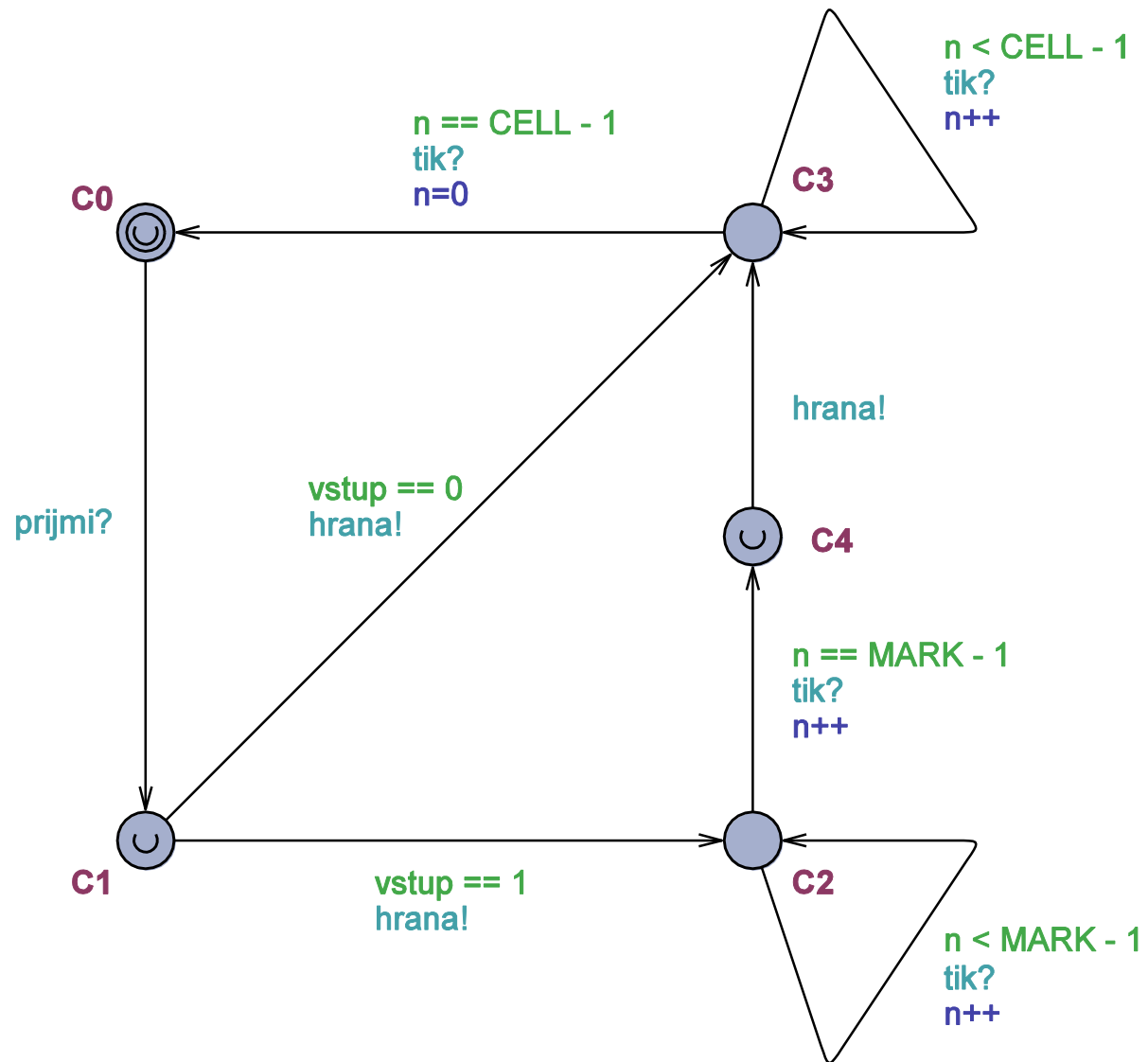
- | | |
|---------------------|--|
| □ CELL = 32 | -- délka buňky v hodinových cyklech |
| □ MARK = 16 | -- délka podbuňky v hodinových cyklech |
| □ DELKA_VZORKU = 23 | -- délka jednoho vzorkování v hodinových cyklech |
| □ MIN = 81 | -- minimální délka hodinového cyklu hodin |
| □ MAX = 100 | -- maximální délka hodinového cyklu hodin |
| □ DELKA_HRANY < 81 | -- max. délka náběhu (nestabilní) hrany |

BMP – digitální hodiny

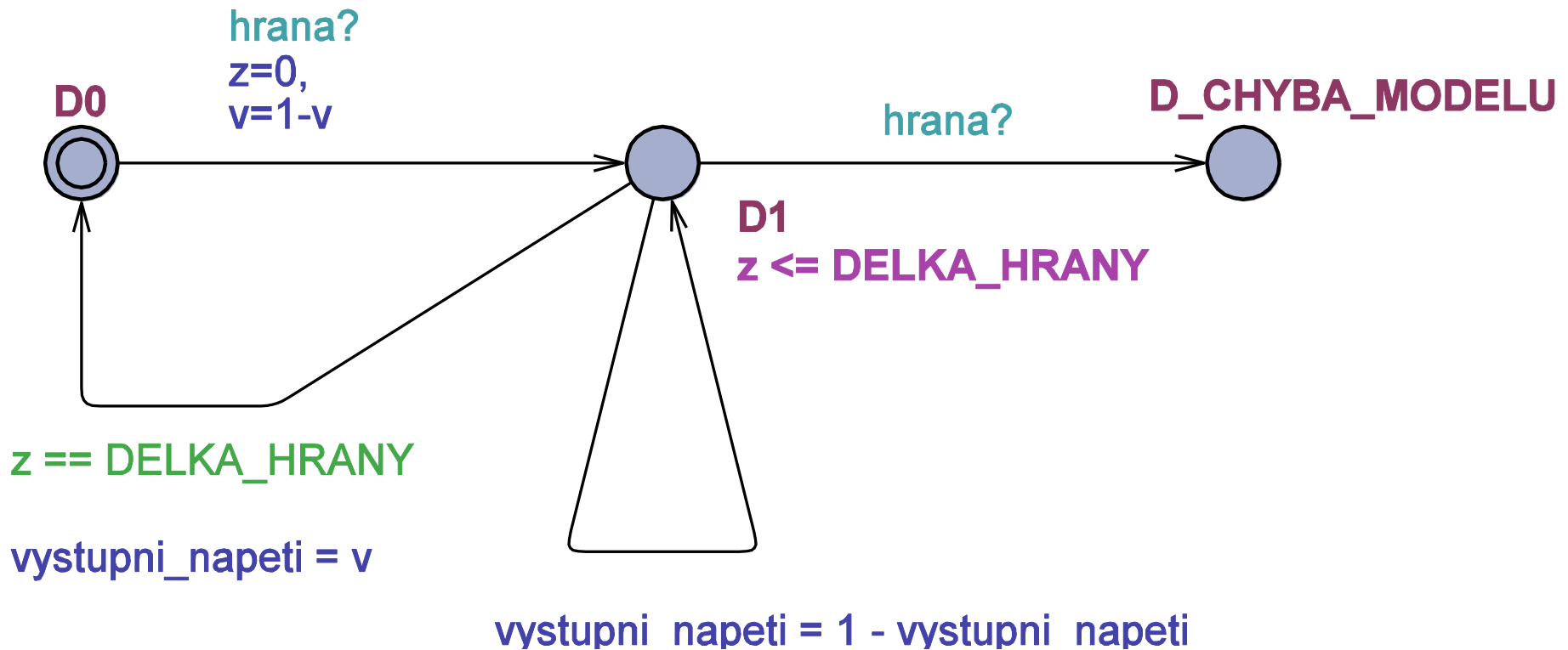


- Hodinové signály jsou vysílány broadcastovým kanálem tik.
- Přesnost hodin je dána nastavením nedeterministického rozmezí min a max.

BMP – encoder

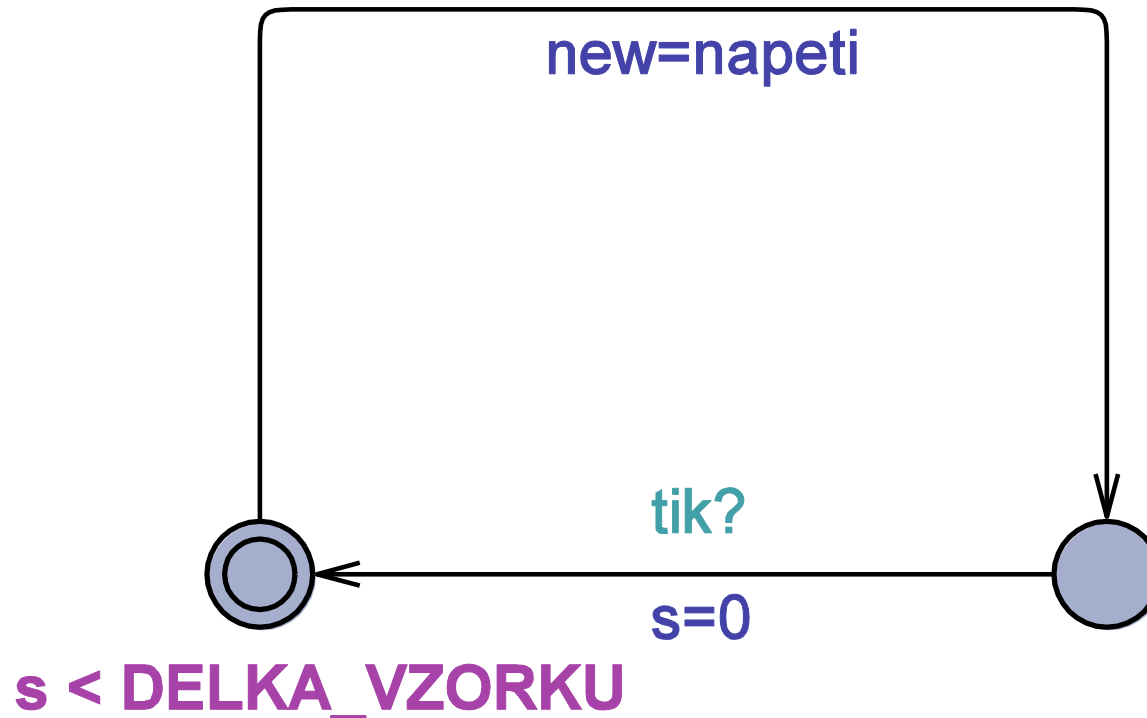


BMP – drát



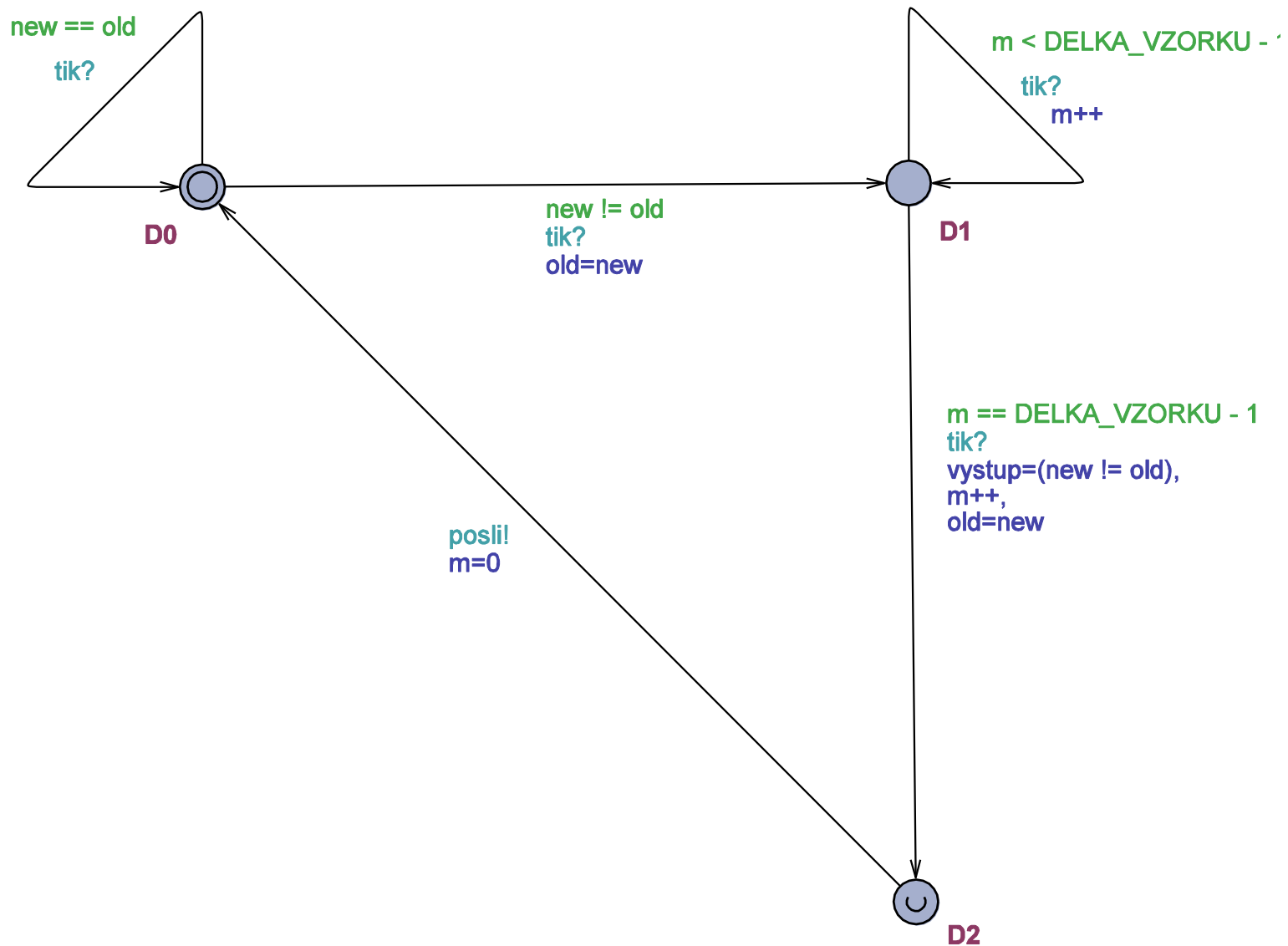
- Vstupní událost je **hrana?** (po této události dojde ke změně napětí).
- Napětí se stabilizuje až po uplynutí **DELKA_HRANY** (jinak se nedefinovaně mění).

BMP – sampler (vzorkovač)



- Vstupní proměnná je **napeti**.
- Synchronizace přes **tik?**.

BMP – decoder



BMP – tester

