**Real Time: CTS**

Let $T$ be a totally ordered monoid, to which we refer as the time domain. The most frequently used timed domains are $\mathbb{R}^+$, the non-negative reals, and $\mathbb{N}$, the naturals.

A **clocked transition system (CTS)** $\Phi = (V, \Theta, \rho, \Pi)$ consists of:

- $V$ – A finite set of state variables. The set $V = D \cup C$ is partitioned into $D = \{u_1, \ldots, u_n\}$ the set of discrete variables and $C = \{t_1, \ldots, t_k\}$ the set of clocks. Clocks always have the type $T$. The discrete variables can be of any type. Optionally, $C$ may include a special clock $T \in C$, called the master clock.

- $\Theta$ – An initial condition. A satisfiable assertion characterizing the initial states. In case $T \in C$, it is required that $\Theta \rightarrow T = 0$, i.e., $T = 0$ at all initial states.

- $\rho$ – A transition relation. An assertion $\rho(V, V')$, referring to both unprimed (current) and primed (next) versions of the state variables.

  In case $T \in C$, it is required that $\rho \rightarrow T' = T$, i.e., the master clock is modified by no transition.

- $\Pi$ – The time-progress condition. An assertion over $V$. The assertion is used to specify a global restriction over the progress of time.

### Runs and Computations

Let $\Phi = (V, \Theta, \rho, \Pi)$ be a CTS. A **run** of $\Phi$ is a finite or infinite sequence of states $\sigma : s_0, s_1, \ldots$ satisfying:

- **Initiality:** $s_0 \models \Theta$

- **Consecution:** For each $j \in [0,|\sigma|)$ $s_{j+1}$ is a $\rho_j$-successor of $s_j$.

A state is called **($\Phi$-)accessible** if it appears in a run of $\Phi$.

A **computation** of $\Phi$ is an infinite run satisfying:

- **Time Divergence:** The sequence $s_0[T], s_1[T], \ldots$ grows beyond any bound. That is, as $i$ increases, the value of $T$ at $s_i$ increases beyond any bound.

A CTS $\Phi$ is called **non-zeno** if every finite run of $\Phi$ can be extended into a computation.

**The Extended Transition Relation**

Let $\Phi : (V, \Theta, \rho, \Pi)$ be a CTS. We define the **extended transition relation** $\rho_T$ associated with $\Phi$ as

$$\rho_T = \rho \lor \rho_{\text{tick}},$$

where $\rho_{\text{tick}}$ is given by:

$$\rho_{\text{tick}} : \exists \Delta. \Omega(\Delta) \land D' = D \land C' = C + \Delta,$$

and $\Omega(\Delta)$ is given by

$$\Omega(\Delta) : \Delta > 0 \land \forall t \in [0, \Delta), \Pi(D, C + t)$$

Let $D = \{u_1, \ldots, u_m\}$ be the set of discrete variables of $\Phi$ and $C = \{c_1, \ldots, c_k\}$ be the set of its clocks. Then, the expression $C' = C + \Delta$ is an abbreviation for

$$c_1' = c_1 + \Delta \land \cdots \land c_k' = c_k + \Delta,$$

and $\Pi(D, C + t)$ is an abbreviation for

$$\Pi(u_1, \ldots, u_m, c_1 + t, \ldots, c_k + t)$$

### A Frequently Occurring Case

In many cases, the time-progress condition $\Pi$ has the following special form

$$\Pi : \bigwedge_{i \in I} (p_i \rightarrow t_i < E_i),$$

where $I$ is some finite index set and, for each $i \in I$, the assertion $p_i$ and the $T$-valued expression $E_i$ do not depend on the clocks, and $t_i \in C$ is some clock. This is, for example, the form of the time-progress condition for any CTS representing a real-time program. For such cases, the time-increment limiting formula $\Omega(\Delta)$ can be significantly simplified and assumes the following form:

$$\Omega(\Delta) : \Delta > 0 \land \bigwedge_{i \in I} (p_i \rightarrow t_i + \Delta \leq E_i)$$

Note, in particular, that this simpler form does not use quantifications over $t$. 

Timed and Hybrid Systems, NYU, Spring, 2007
Examples

System $\Phi_1$ can be presented by the following diagram:

\[ y := y + 1 \]

\[ \ell_0 : t < 2 \]

\[ \ell_1 \]

Textually, this can be presented as:

\[ V : \{\pi : \{0, 1\}, y, t, T\} \]
\[ \Theta : \pi = y = t = T = 0 \]
\[ \rho : \pi = 0 \land \text{pres}(t, T) \land \left\{ \begin{array}{l} \pi' = 0 \land y' = y + 1 \\ \pi' = 1 \land y' = y \end{array} \right. \]
\[ \Pi : \pi = 0 \rightarrow t < 2 \]

The $\text{tick}$ transition is given by

\[ \rho_{\text{tick}} : \exists \Delta > 0 \left\{ (t', T') = (t + \Delta, T + \Delta) \land \text{pres}(\pi, y) \right. \]

\[ \land (\pi = 0 \rightarrow t + \Delta \leq 2) \]

System $\Phi_2$

System $\Phi_2$ can be presented by the following diagram:

\[ (y, t) := (y + 1, 0) \]

\[ \ell_0 : t < \frac{1}{10^{99+\pi}} \]

\[ \ell_1 \]

Textually, this can be presented as:

\[ V : \{\pi : \{0, 1\}, y, t, T\} \]
\[ \Theta : \pi = y = t = T = 0 \]
\[ \rho : \pi = 0 \land T' = T \land \left\{ \begin{array}{l} \pi' = 0 \land (y', t') = (y + 1, 0) \\ \pi' = 1 \land (y', t') = (y, t) \end{array} \right. \]
\[ \Pi : \pi = 0 \rightarrow t < \frac{1}{10^{99+\pi}} \]

The $\text{tick}$ transition is given by

\[ \rho_{\text{tick}} : \exists \Delta > 0 \left\{ (t', T') = (t + \Delta, T + \Delta) \land \text{pres}(\pi, y) \right. \]

\[ \land (\pi = 0 \rightarrow t + \Delta \leq \frac{1}{10^{99+\pi}}) \]
**System $\Phi_3$**

System $\Phi_3$ below is not a non-zeno system.

\[
(y, t) := (y + 1, 0)
\]

Following is a finite run which cannot be extended to a computation:

\[
\begin{align*}
\langle \pi, 0, y: 0, t: 0, T: 0 \rangle & \xrightarrow{\ell_0} \langle \pi, 0, y: 0, t: 0, T: 0 \rangle \\
\langle \pi, 0, y: 1, t: 0, T: 0 \rangle & \xrightarrow{\ell_0} \langle \pi, 0, y: 1, t: 0, T: 0 \rangle \\
\langle \pi, 0, y: 2, t: 0, T: 0 \rangle & \xrightarrow{\ell_0} \langle \pi, 0, y: 2, t: 0, T: 0 \rangle \\
\langle \pi, 0, y: 3, t: 0, T: 0 \rangle & \xrightarrow{\ell_0} \langle \pi, 0, y: 3, t: 0, T: 0 \rangle \\
\langle \pi, 0, y: 4, t: 0, T: 0 \rangle & \xrightarrow{\ell_0} \langle \pi, 0, y: 4, t: 0, T: 0 \rangle \\
\langle \pi, 0, y: 5, t: 0, T: 0 \rangle & \xrightarrow{\ell_0} \langle \pi, 0, y: 5, t: 0, T: 0 \rangle \\
\langle \pi, 0, y: 6, t: 0, T: 0 \rangle & \xrightarrow{\text{tick}(10^{-7})} \langle \pi, 0, y: 6, t: 10^{-7}, T: 10^{-7} \rangle \\
\langle \pi, 0, y: 7, t: 0, T: 10^{-7} \rangle & \xrightarrow{\text{tick}(10^{-8})} \langle \pi, 0, y: 7, t: 10^{-8}, T: 1.1 \cdot 10^{-7} \rangle
\end{align*}
\]

**The Timed Version of SPL Programs**

Let $P$ be an SPL program. To obtain the real-time version of $P$, we associate with each $\ell$, a location of $P$, a pair of values $[L_\ell, U_\ell]$, called the lower and upper bounds of $\ell$. These values, satisfying $0 \leq L_\ell \leq U_\ell < \infty$, are intended to provide a lower and upper bound on the length of time execution can stay at location $\ell$ without taking a transition. We refer to a program with an assignment of time bounds as an SPL$_T$ program, and view it as a real-time program.

**The CTS Corresponding to an SPL$_T$ Program**

Let $P$ be an SPL$_T$ program with $m$ processes, and let $D^u_P : (V^u, \Theta^u, \rho^u, J^u, C^u)$ be the FDS corresponding to $P$.

The CTS corresponding to $P$ is given by $\Phi_P : (V, \Theta, \rho, \Pi)$, where

- **System Variables**: $V = V^u \cup \{t_1, \ldots, t_m, T\}$.

  Thus, $V$ consists of the system variables of $D^u_P$, to which we add $m + 1$ clocks, one clock $t_i$ for each process $P_i$, $i = 1, \ldots, m$, plus the master clock $T$.

- **Initial Condition**: $\Theta : \Theta^u \land t_1 = \cdots = t_m = T = 0$.

- **Transition Relation**: For each location $\ell$ in process $P_i$, to which $\rho^u$ associated the disjunct $\rho^u_\ell$, $\rho$ contains the disjunct

  $\rho : \rho^u_\ell \land t_i \geq L_\ell \land t'_i = 0 \land \text{pres}(t_1, \ldots, t_{i-1}, t_{i+1}, \ldots, t_m, T)$

  Thus, the transition at $\ell$ can be taken only when $t_i$, the clock corresponding to process $P_i$, is not below $L_\ell$, the lower bound associated with $\ell$. When taken, the transition resets clock $t_i$ to 0 and preserves all other clocks.
**Example: A Timed Version of ANY-Y**

Reconsider program ANY-Y.

\[
\ell_0 : \quad \text{while } x = 0 \do
\begin{align*}
\ell_1 : \quad & x := y + 1 \\
\ell_2 : \quad & P_1 \quad - \\
\end{align*}
\\]
\[
\begin{align*}
m_0 : \quad & x := 1 \\
m_1 : \quad & P_2 \quad -
\end{align*}
\]

To make it an SPL\_T program, we uniformly associate each of its non-terminal locations with the time bounds \([3, 5]\), and the terminal locations \(\ell_2, m_1\), with the time bounds \([0, \infty]\).

**The Corresponding CTS**

**System Variables:** \(V = \{\pi_1, \pi_2, x, y, t_1, t_2, T\}\). In addition to the control variables \(\pi_1, \pi_2\) and data variables \(x\) and \(y\), the system variables also include clock \(t_1\), measuring delays in process \(P_1\), clock \(t_2\), measuring delays in process \(P_2\), and the master clock \(T\), measuring time from the beginning of the computation.

**Initial Condition:**

\(\Theta : \quad \pi_1 = 0 \land \pi_2 = 0 \land x = y = 0 \land t_1 = t_2 = T = 0.\)

**Transition relation:**

\(\rho = \rho_{\ell_0} \lor \rho_{\ell_1} \lor \rho_{m_0}\) where:

\[
\begin{align*}
\rho_{\ell_0} : \quad & \pi_1 = 0 \land \left(x = 0 \lor \pi'_1 = 1\right) \land t_1 \geq 3 \land t'_1 = 0 \\
& \land \pres(\pi_2, x, y, t_2, T) \\
\rho_{\ell_1} : \quad & \pi_1 = 1 \land \pi'_1 = 0 \land y' = y + 1 \land t_1 \geq 3 \land t'_1 = 0 \\
& \land \pres(\pi_2, x, t_2, T) \\
\rho_{m_0} : \quad & \pi_2 = 0 \land \pi'_2 = 1 \land x'' = 1 \land t_2 \geq 3 \land t'_2 = 0 \\
& \land \pres(\pi_1, y, t_1, T).
\end{align*}
\]

**Time-progress condition:**

\(\Pi : \quad (\text{at} \ell_{0, 1} \rightarrow t_1 < 5) \land (\text{at} \cdot m_0 \rightarrow t_2 < 5)\)
The \textit{tick} transition for this system is given by

\[ \rho_{\text{tick}} : \exists \Delta > 0 \left\{ \begin{array}{l}
(at_{\ell_{0,1}} \rightarrow t_1 + \Delta \leq 5) \\
\wedge (at_{m_0} \rightarrow t_2 + \Delta \leq 5) \\
\wedge \text{pres}(\pi_1, \pi_2, x, y) \\
\wedge (t_1', t_2', T') = (t_1 + \Delta, t_2 + \Delta, T + \Delta)
\end{array} \right\} \]