# Cyber Physical Systems - Discrete Models
# Exercise Sheet 9 Solution

Alper Ari
aa508@uni-freiburg.edu

Onur Sahin
os141@uni-freiburg.de

December 10, 2023

## Exercise 1: Invariants

### A

**Proposition:** Let $E \subseteq \left(2^{\mathrm{AP}}\right)^{\omega}$ be an LT property. $E$ is not an invariant if and only if there exists a trace $\sigma = A_0 A_1 \ldots \in \left(2^{\mathrm{AP}}\right)^{\omega}$ such that $\sigma \notin E$, but for every $i \in \mathbb{N}$, the set $A_i$ also occurs some other trace $\pi_i \in E$.

**Proof:** Assume that $E$ is not an invariant and there exists a trace $\sigma = A_0 A_1 \ldots \in \left(2^{\mathrm{AP}}\right)^{\omega}$ such that $\sigma \notin E$, but for every $i \in \mathbb{N}$, the set $A_i$ also occurs some other trace $\pi_i \in E$.

Let $\Phi$ be the invariant condition of $E$ then, by the definition of invariant we can conclude $\forall i \in \mathbb{N} \cdot \forall \sigma' \in \pi_i \cdot \sigma' \vDash \Phi$. Because $\sigma \notin E$, it means that $\exists i \in \mathbb{N} \cdot A_i \nvDash \Phi$. And there exists a corresponding $A_i \in \pi_i \land \pi_i \in E$. This means that $A_i \vDash \Phi$ which is a contradiction. So $E$ must be an invariant given the properties it exhibits ■

**B**

$E_1$

$$E_1 = \left\{A_0 A_1 ... \in \left(2^{\mathrm{AP}}\right)^\omega \mid \forall i \in \mathbb{N} \cdot \left(a \in A_i \rightarrow b \in A_{i+1}\right)\right\}$$

Let $w = A_0 A_1 ... = a^\omega$ and $p_1 = \mathrm{ab}^\omega$. Clearly $w \notin E_1$ but $p_1 \in E_1$. Also $\forall i \in \mathbb{N} \cdot A_i = a$ and $a \in p_1$, hence $E_1$ is not an invariant.

$E_2$

$$E_2 = \left\{A_0 A_1 ... \in \left(2^{\mathrm{AP}}\right)^\omega \mid \forall i, j \in \mathbb{N} \cdot A_i = A_j\right\}$$

$E = \{a^\omega, b^\omega\}$. Let $p_2 = A_0 A_1 ... = a(b^\omega)$, $p_2 \notin E$. Since $A_0 \in a^\omega$ and $\forall i > 0 \cdot A_j = b \wedge A_j \in b^\omega$, all sets either contained in $a^\omega$ or $b^\omega$. Therefore $E_2$ is not an invariant.

$E_3$

$$E_3 = \left\{A_0 A_1 ... \in \left(2^{\mathrm{AP}}\right)^\omega \mid |\{\forall i \in \mathbb{N} \mid a \in A_i\}| \geq 2\right\}$$

Let $\pi_3 = aa(b^\omega) \in E_3$ and $\sigma = aaa(b^\omega) \notin E_3$. Both $a$ and $b$ is in $\pi_3$ and they also in $\sigma$. Therefore, $E_3$ is not an invariant.

## Exercise 2: LT Properties

### $P_1$

**Part A**

$$P_1 = \left\{A_0 A_1 ... \in \left(2^{\mathrm{AP}}\right)^\omega \mid \forall i \in \mathbb{N} \cdot a \in A_i \vee b \in A_i\right\}$$

**Part B**

It's an invariant with invariant condition $\Phi = a \vee b$.

**Part C**

Since every invariant is a safety property, this is also a safety property. Set of bad prefixes can be denoted as
$\mathrm{BadPref} = \left\{A_0 A_1 ... A_n \in \left(2^{\mathrm{AP}}\right)^+ \mid \exists i \in 0..n \cdot \emptyset = A_i\right\}$

**Part D**

It's not a liveness property, because $P_1$ contains prefixes that can't be extended to satisfy the language. For example $\sigma = \{a\}\emptyset\{a\}$ can't be extended so that it would satisfy the langage.

## $P_2$

### Part A

$$P_2 = \left\{A_0 A_1 ... \in \left(2^{\mathrm{AP}}\right)^{\omega} \mid (|\{i \in \mathbb{N} \cdot a \in A_i\}| = 1) \lor (\forall i \in \mathbb{N} \cdot b \notin A_i)\right\}$$

### Part B

$P_2$ is not an invariant since there is no such $\Phi$ that we can check for individual states.

### Part C

$P_2$ is a safety property because once the condition is violated in a prefix, it can't be extended to satisfy it. It has the bad prefixes

$$\mathrm{BadPref} = \left\{A_0 A_1 ... A_n \in \left(2^{\mathrm{AP}}\right)^{+} \mid (|\{i \in 0..n \cdot a \in A_i\}| > 1) \land (\exists i \in 0..n \cdot b \in A_i)\right\}$$

### Part D

$P_2$ is not a liveness property because it contains prefixes that can't be added into language by appending some trace. For example: $\sigma = \{a\}\{a, b\}$.

## $P_3$

The wording "b will never hold in the next step" is ambiguous. It's not clear if b doesn't hold in the next (subsequent) step or for all later steps. I am assuming that it's only the next step.

### Part A

$$P_3 = \left\{A_0 A_1 ... \in \left(2^{\mathrm{AP}}\right)^{\omega} \mid \forall i \in \mathbb{N} \cdot (a \in A_i \to b \notin A_{i+1})\right\}$$

### Part B

$P_3$ is not an invariant because the constraint involves subsequent steps. Therefore, it's not possible to write a propositional logic formula $\Phi$ that would be evaluated for each step.

### Part C

$P_3$ is a safety property because it has bad prefixes. Set of bad prefixes are:

$$\mathrm{BadPref} = \left\{A_0 A_1 ... A_n \in \left(2^{\mathrm{AP}}\right)^{+} \mid \exists i \in 1..n \cdot a \in A_{i-1} \land b \in A_i\right\}$$

### Part D

$P_3$ is not a liveness property because there are bad prefixes for this language. Those bad prefixes can't be extended to satisfy the language, so the language doesn't satisfy of the condition of liveness properties having $\left(2^{\mathrm{AP}}\right)^{+}$ as the prefix set.

## $P_4$

### Part A

$$P_4 = \left\{A_0 A_1 ... \in \left(2^{\mathrm{AP}}\right)^{\omega} \mid \forall i \in \mathbb{N} \cdot \left(a \in A_i \rightarrow \left(\exists j \geq i \cdot b \in A_j\right)\right)\right\}$$

### Part B

$P_4$ is not an invariant because the language constraint involves multiple steps to check. Therefore, it's not possible to write a propositional logic formula $\Phi$ that would be evaluated for each step.

### Part C

$P_4$ is not a safety property, because for any prefix $\sigma \in \left(2^{\mathrm{AP}}\right)^{+}$ we can append $w = A_0 A_1 ... \in \left(2^{\mathrm{AP}}\right)^{\omega} \cdot \left(\forall i \in \mathbb{N} \cdot \left(a \in A_i \rightarrow b \in A_{i+1}\right)\right)$ which means $\sigma w \in P_4$. Hence $\mathrm{BadPref} = \emptyset$.

### Part D

$P_4$ is a liveness property because as explained in Part C we can extend any finite prefix $\sigma \in \left(2^{\mathrm{AP}}\right)^{+}$ with a trace $w \in \left(2^{\mathrm{AP}}\right)^{\omega}$ so that $\sigma w \in P_4$.

## $P_5$

### Part A

$$P_5 = \left\{A_0 A_1 ... \in \left(2^{\mathrm{AP}}\right)^{\omega} \mid \forall i \in \mathbb{N} \cdot \{a, b\} \neq A_i\right\}$$

### Part B

$P_5$ is an invariant with the invariant condition $\Phi = \neg(a \wedge b)$.

### Part C

Since $P_5$ is an invariant, it's automatically a safety property. The set of bad prefixes are:

$$\mathrm{BadPref} = \left\{A_0 A_1 ... A_n \in \left(2^{\mathrm{AP}}\right)^{+} \mid \exists i \in 0..n \cdot \{a, b\} = A_i\right\}$$

### Part D

Since $P_5$ is a safety property, it can't be a liveness property. A counter example is prefix $\sigma = \{a, b\}$. Because for any $\forall w \in \left(2^{\mathrm{AP}}\right)^{\omega} \cdot \sigma w \notin P_5$.

## $P_6$

**Part A**

$$P_6 = \left\{A_0 A_1 \ldots \in \left(2^{\mathrm{AP}}\right)^\omega \mid \left(\overset{\infty}{\exists} i \in \mathbb{N} \cdot a \in A_i\right) \to \left(\overset{\infty}{\exists} i \in \mathbb{N} \cdot b \in A_i\right)\right\}$$

**Part B**

$P_6$ is not an invariant because condition requires checking multiple steps at the same time. Therefore there is no boolean proposition formula $\Phi$ to check for a single step.

**Part C**

$P_6$ is not a safety property, because for any bad prefix $\sigma \in \left(2^{\mathrm{AP}}\right)^+$ we can append $w = A_0 A_1 \ldots \in \left(2^{\mathrm{AP}}\right)^\omega \mid \overset{\infty}{\exists} i \in \mathbb{N} \cdot b \in A_i$ which means $\sigma w \in P_6$. Hence $\mathrm{BadPref} = \emptyset$.

**Part D**

$P_6$ is a liveness property because as explained in Part C we can extend any finite prefix $\sigma \in \left(2^{\mathrm{AP}}\right)^+$ with a trace $w \in \left(2^{\mathrm{AP}}\right)^\omega$ so that $\sigma w \in P_6$.

## $P_7$

**Part A**

$$P_7 = \left\{A_0 A_1 \ldots \in \left(2^{\mathrm{AP}}\right)^\omega \mid \exists i \in \mathbb{N} \cdot \forall j > i \cdot a \notin A_j\right\}$$

**Part B**

$P_7$ is not an invariant because the condition involves checking multiple steps at the same time. Therefore there is no boolean proposition formula $\Phi$ to check for a single step.

**Part C**

$P_7$ is not a safety property, because for any bad prefix $\sigma \in \left(2^{\mathrm{AP}}\right)^+$ we can append $w = A_0 A_1 \ldots \in \left(2^{\mathrm{AP}}\right)^\omega \mid \forall i \in \mathbb{N} \cdot a \notin A_i$ which means $\sigma w \in P_7$. Hence $\mathrm{BadPref} = \emptyset$.

**Part D**

$P_6$ is a liveness property because as explained in Part C we can extend any finite prefix $\sigma \in \left(2^{\mathrm{AP}}\right)^+$ with a trace $w \in \left(2^{\mathrm{AP}}\right)^\omega$ so that $\sigma w \in P_7$.

## $P_8$

**Part A**

$$P_8 = \left\{A_0 A_1 \ldots \in \left(2^{\mathrm{AP}}\right)^{\omega} \mid \mathrm{true}\right\}$$

**Part B**

$P_8$ is an invariant with the invariant condition $\Phi = \mathrm{true}$.

**Part C**

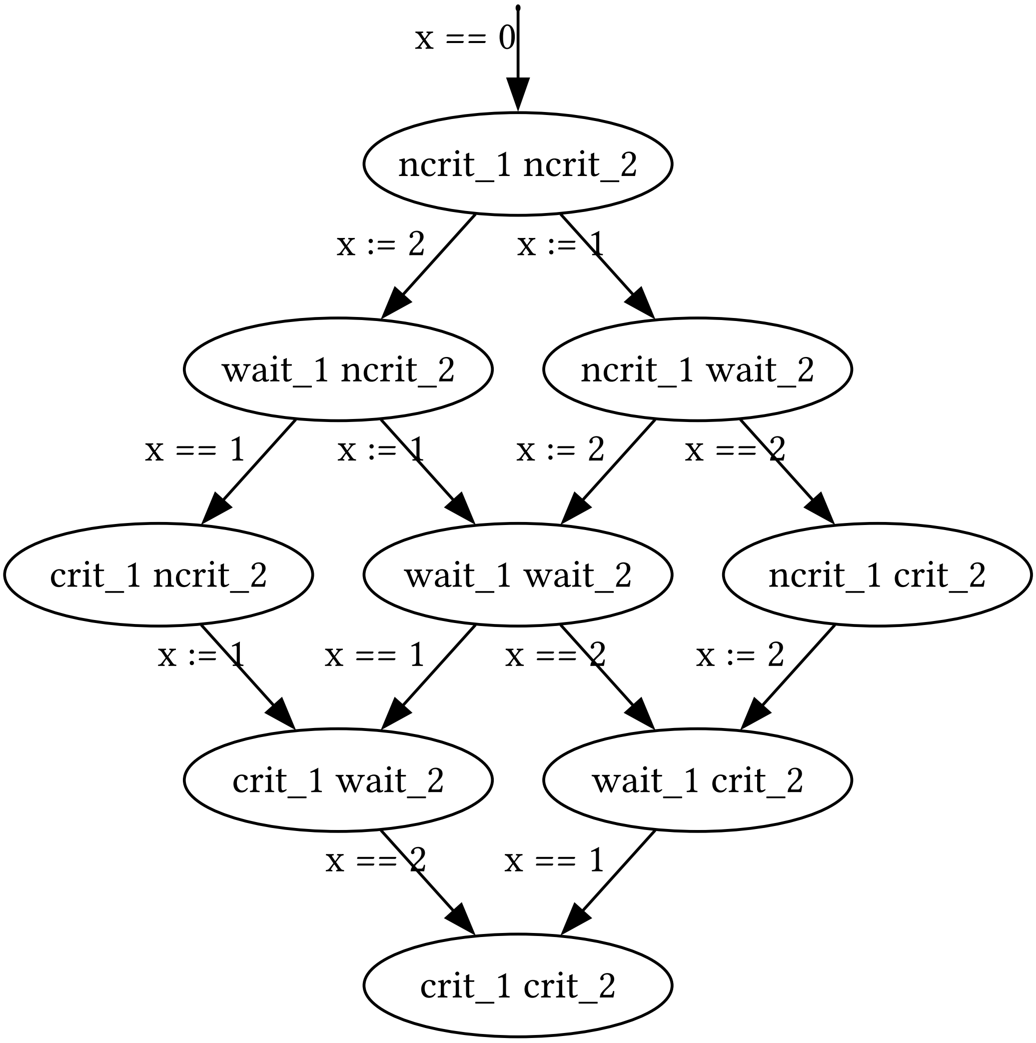
$P_8$ is a safety property, because even if it doesn't have any bad prefixes it doesn't have any traces that is not in the language either. So it doesn't need to have any bad prefixes.

**Part D**

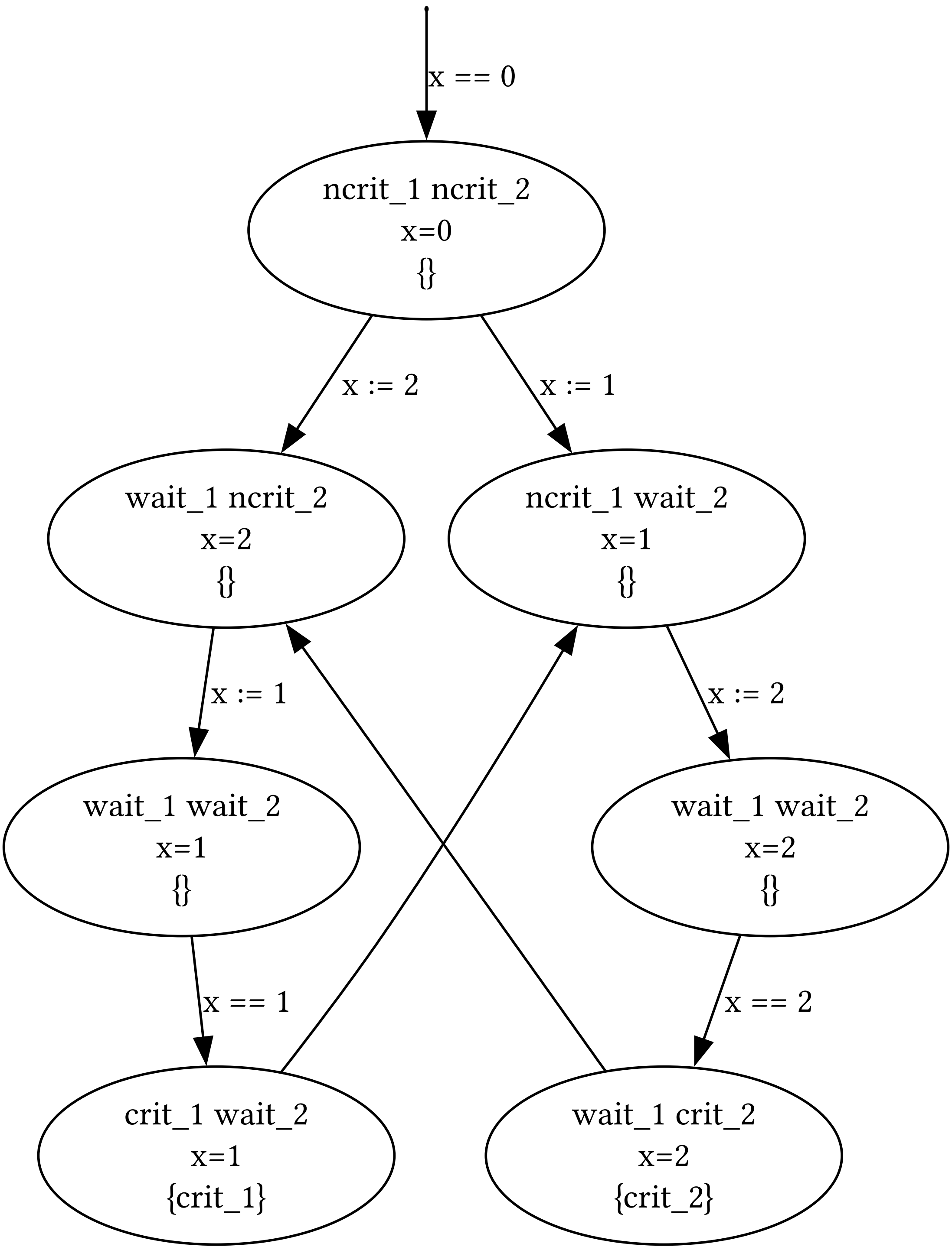
$P_8$ is a liveness property, because $\mathrm{pref}(P_8) = \left(2^{\mathrm{AP}}\right)^{+}$.

## Exercise 3: Mutual Exclusion

### Part A




### Part B




### Part C

Yes because in all states, invariatn $\Phi = \neg \text{crit}_1 \lor \neg \text{crit}_2$ is satisfied.

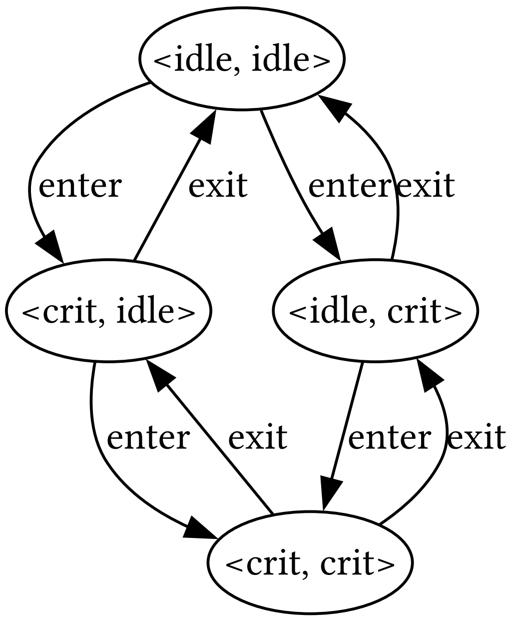
### Part D

Yes, because both mutual exclusion and fairness is satisfied in this TS. Fairness is satisfied because the system forces alternating sequences of critical sequence entrance for both programs.

## Exercise 4: Mutual Exclusion without Request

## Part A

$\mathrm{TS}_1 \ ||| \ \mathrm{TS}_2$




### Transitions

$$\langle \text{idle}, \text{idle}\rangle \xrightarrow{\text{enter}} \langle \text{crit}, \text{idle}\rangle : \frac{\text{idle} \xrightarrow{\text{enter}}_1 \text{crit}}{\langle \text{idle}, \text{idle}\rangle \xrightarrow{\text{enter}} \langle \text{crit}, \text{idle}\rangle} : \text{SOS}_1$$

$$\langle \text{idle}, \text{idle}\rangle \xrightarrow{\text{enter}} \langle \text{idle}, \text{crit}\rangle : \frac{\text{idle} \xrightarrow{\text{enter}}_2 \text{crit}}{\langle \text{idle}, \text{idle}\rangle \xrightarrow{\text{enter}} \langle \text{idle}, \text{crit}\rangle} : \text{SOS}_2$$

$$\langle \text{crit}, \text{idle}\rangle \xrightarrow{\text{enter}} \langle \text{idle}, \text{idle}\rangle : \frac{\text{crit} \xrightarrow{\text{exit}}_1 \text{idle}}{\langle \text{crit}, \text{idle}\rangle \xrightarrow{\text{exit}} \langle \text{idle}, \text{idle}\rangle} : \text{SOS}_1$$

$$\langle \text{idle}, \text{crit}\rangle \xrightarrow{\text{enter}} \langle \text{idle}, \text{idle}\rangle : \frac{\text{crit} \xrightarrow{\text{exit}}_2 \text{idle}}{\langle \text{idle}, \text{crit}\rangle \xrightarrow{\text{exit}} \langle \text{crit}, \text{idle}\rangle} : \text{SOS}_2$$

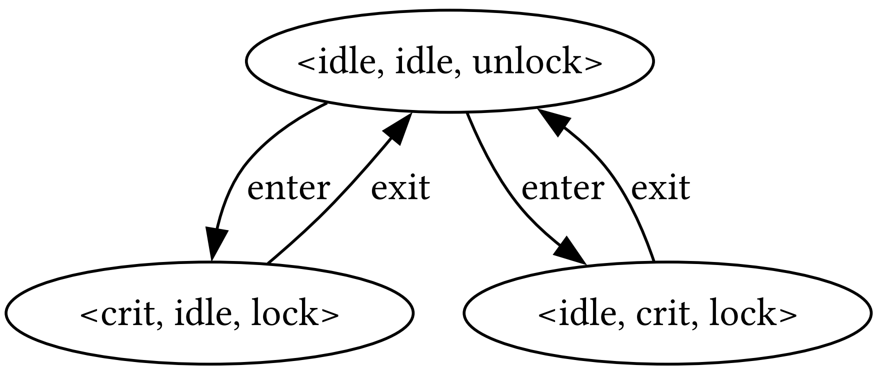
$$\langle \text{crit}, \text{idle}\rangle \xrightarrow{\text{enter}} \langle \text{crit}, \text{crit}\rangle : \frac{\text{idle} \xrightarrow{\text{enter}}_2 \text{crit}}{\langle \text{crit}, \text{idle}\rangle \xrightarrow{\text{enter}} \langle \text{crit}, \text{crit}\rangle} : \text{SOS}_2$$

$$\langle \text{idle}, \text{crit}\rangle \xrightarrow{\text{enter}} \langle \text{crit}, \text{crit}\rangle : \frac{\text{idle} \xrightarrow{\text{enter}}_1 \text{crit}}{\langle \text{idle}, \text{crit}\rangle \xrightarrow{\text{enter}} \langle \text{crit}, \text{crit}\rangle} : \text{SOS}_1$$

$$\langle \text{crit}, \text{crit}\rangle \xrightarrow{\text{exit}} \langle \text{idle}, \text{crit}\rangle : \frac{\text{crit} \xrightarrow{\text{exit}}_1 \text{idle}}{\langle \text{crit}, \text{crit}\rangle \xrightarrow{\text{exit}} \langle \text{idle}, \text{crit}\rangle} : \text{SOS}_1$$

$$\langle \text{crit}, \text{crit}\rangle \xrightarrow{\text{exit}} \langle \text{crit}, \text{idle}\rangle : \frac{\text{crit} \xrightarrow{\text{exit}}_2 \text{idle}}{\langle \text{crit}, \text{crit}\rangle \xrightarrow{\text{exit}} \langle \text{crit}, \text{idle}\rangle} : \text{SOS}_2$$

## Part B




### Transitions

All relations are formed via $\text{SOS}_3$, which is the rule:

$$\frac{s \xrightarrow[1]{\alpha} s' \land q \xrightarrow[2]{\alpha} q'}{\langle s, q\rangle \xrightarrow{\alpha} \langle s', q'\rangle}$$

$$\langle \text{idle}, \text{idle}, \text{unlock}\rangle \xrightarrow{\text{enter}} \langle \text{crit}, \text{idle}, \text{lock}\rangle : \frac{\langle \text{idle}, \text{idle}\rangle \xrightarrow{\text{enter}}_1 \langle \text{crit}, \text{idle}\rangle \land \text{unlock} \xrightarrow{\text{enter}}_2 \text{lock}}{\langle \text{idle}, \text{idle}, \text{unlock}\rangle \xrightarrow{\text{enter}} \langle \text{crit}, \text{idle}, \text{lock}\rangle} : \text{SOS}_3$$

$$\langle \text{idle}, \text{idle}, \text{unlock}\rangle \xrightarrow{\text{enter}} \langle \text{idle}, \text{crit}, \text{lock}\rangle : \frac{\langle \text{idle}, \text{idle}\rangle \xrightarrow{\text{enter}}_1 \langle \text{crit}, \text{idle}\rangle \land \text{unlock} \xrightarrow{\text{enter}}_2 \text{lock}}{\langle \text{idle}, \text{idle}, \text{unlock}\rangle \xrightarrow{\text{enter}} \langle \text{crit}, \text{idle}, \text{lock}\rangle} : \text{SOS}_3$$

$$\langle \text{idle}, \text{crit}, \text{lock}\rangle \xrightarrow{\text{exit}} \langle \text{idle}, \text{idle}, \text{unlock}\rangle : \frac{\langle \text{idle}, \text{crit}\rangle \xrightarrow{\text{exit}}_1 \langle \text{idle}, \text{idle}\rangle \land \text{lock} \xrightarrow{\text{exit}}_2 \text{unlock}}{\langle \text{idle}, \text{crit}, \text{lock}\rangle \xrightarrow{\text{exit}} \langle \text{idle}, \text{idle}, \text{unlock}\rangle} : \text{SOS}_3,$$

$$\langle \text{crit}, \text{idle}, \text{lock}\rangle \xrightarrow{\text{exit}} \langle \text{idle}, \text{idle}, \text{unlock}\rangle : \frac{\langle \text{crit}, \text{idle}\rangle \xrightarrow{\text{exit}}_1 \langle \text{idle}, \text{idle}\rangle \land \text{lock} \xrightarrow{\text{exit}}_2 \text{unlock}}{\langle \text{crit}, \text{idle}, \text{lock}\rangle \xrightarrow{\text{exit}} \langle \text{idle}, \text{idle}, \text{unlock}\rangle} : \text{SOS}_3,$$

# Exercise 5: Hardware Circuit

$$r' = x \land (\neg r)$$

$$y = x \lor r$$