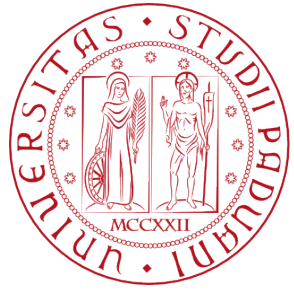
Formal Methods for Cyber-Physical Systems

Lab 3: GR(1) Synthesis



Università degli Studi di Padova

Davide Bresolin
Last updated on: January 17, 2023

Generalized Reactivity(1) Synthesis

- Introduced in 2012 by Bloem et al.
- What are we willing to trade?
  - ... the full expressivity of LTL!
- What do we get?
  - A reduction in complexity from doubly exponential to singly exponential!

GR(1) Synthesis vs LTL Synthesis

Full LTL Synthesis

1. Specification
2. Deterministic Automaton
3. Game
4. Strategy / Component

Doubly exponential complexity

GR(1) Synthesis

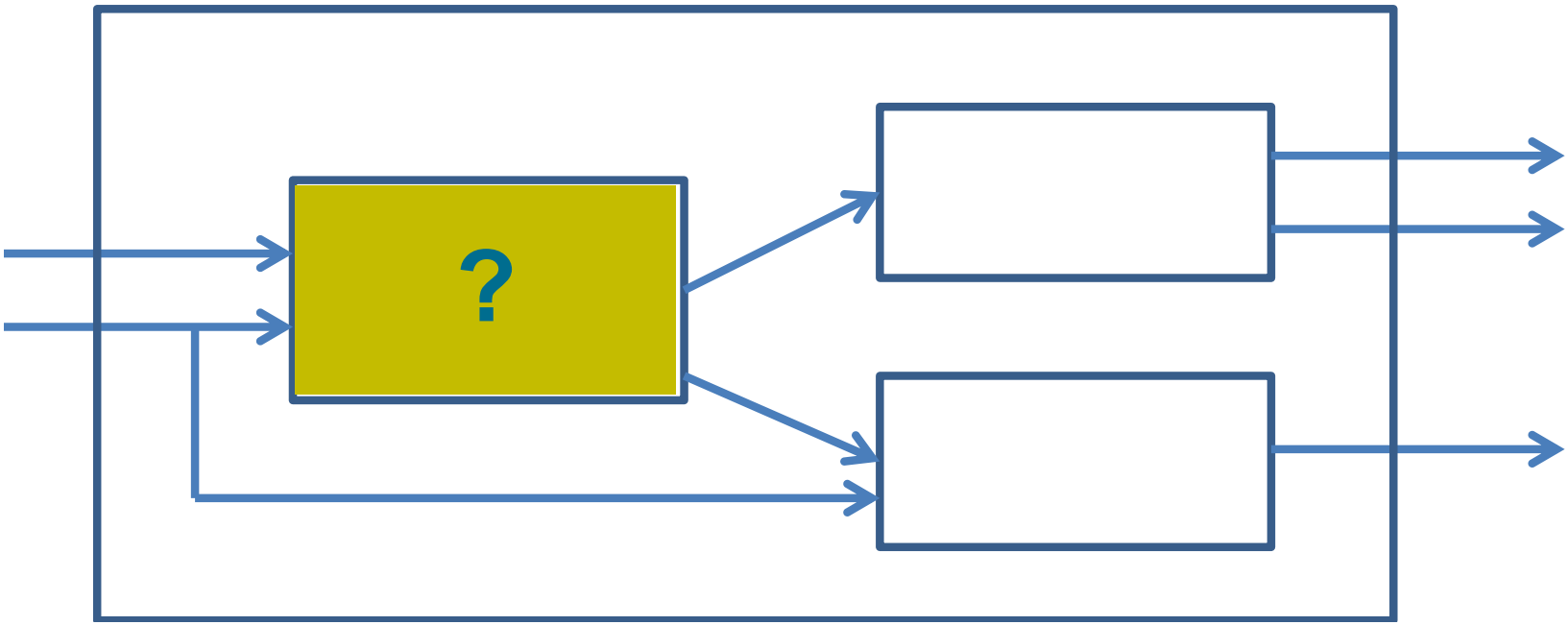
1. Specifcation
2. Direct translation to the Game, exponential blow-up
3. Strategy / Component

Singly exponential complexity

GR(1) – What should be supported?

- Computation model
  - Mealy machine: finite state machine with inputs and outputs
- Specification model
  A specification consists of assumptions and guarantees each of which are either
  - initialization properties
  - basic safety properties
  - basic liveness properties

Assumption and guarantees



Specification shape

$$\left(\bigwedge \text{Assumptions}\right) \rightarrow \left(\bigwedge \text{Guarantees}\right)$$

Overall specification shape

Specification shape

$$\left(\varphi_i^a \wedge \varphi_s^a \wedge \varphi_\ell^a\right) \rightarrow \left(\varphi_i^g \wedge \varphi_s^g \wedge \varphi_\ell^g\right)$$

Specification shape

$$\left(\underbrace{\varphi_i^a}_{\substack{\text{initialization} \\ \text{assumptions}}} \wedge \underbrace{\varphi_s^a}_{\substack{\text{safety} \\ \text{assumptions}}} \wedge \underbrace{\varphi_\ell^a}_{\substack{\text{liveness} \\ \text{assumptions}}} \right) \rightarrow \left(\varphi_i^g \wedge \varphi_s^g \wedge \varphi_\ell^g \right)$$

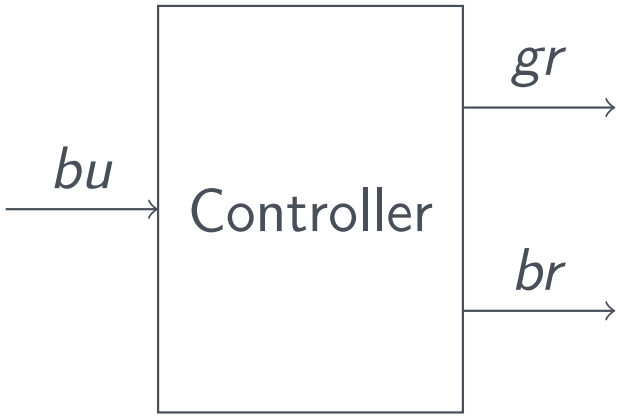
Overall specification shape

Specification shape

$$(\varphi_i^a \wedge \varphi_s^a \wedge \varphi_\ell^a) \rightarrow \left(\underbrace{\varphi_i^g}_{\text{initialization guarantees}} \wedge \underbrace{\varphi_s^g}_{\text{safety guarantees}} \wedge \underbrace{\varphi_\ell^g}_{\text{liveness guarantees}} \right)$$

Coffee machine example

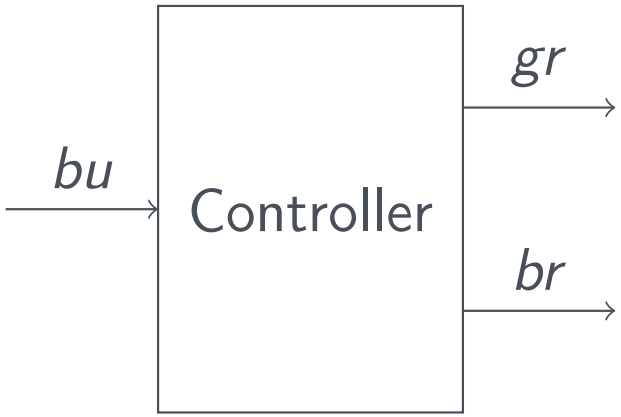
Controller shape



$X_u = \{bu\}$ (button) $\qquad X_c = \{gr, br\}$ (grind, brew)

Initialization assumptions

Controller shape



$I = \{bu\}$ (button) $\quad O = \{gr, br\}$ (grind, brew)

Initialization assumptions

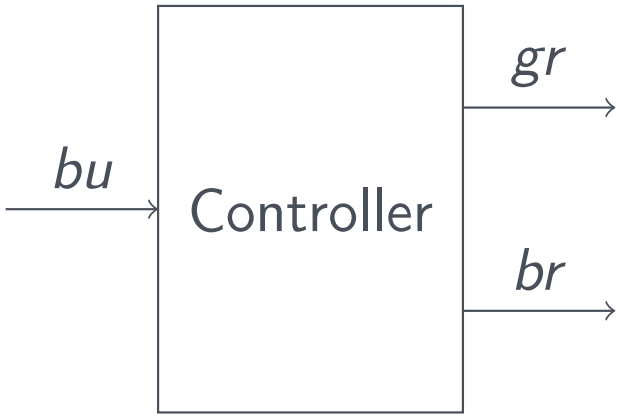
Properties without temporal operators over only I

Example:

- $\neg bu$

Safety assumptions

Controller shape



$I = \{bu\}$ (button) $\quad O = \{gr, br\}$ (grind, brew)

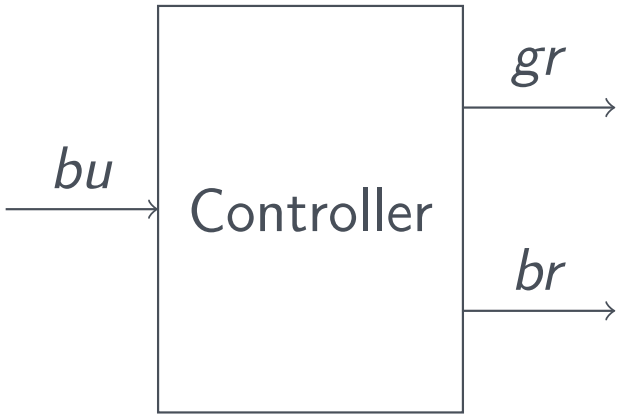
(Basic) Safety assumptions

Properties of the form $\square\psi$, where ψ is a boolean formula over I, O and $\{\bigcirc y \mid y \in I\}$. Examples:

- $\square(bu \rightarrow \neg \bigcirc bu)$
- $\square((gr \vee br) \rightarrow \neg \bigcirc bu)$

Liveness assumptions

Controller shape



$I = \{bu\}$ (button) $\quad O = \{gr, br\}$ (grind, brew)

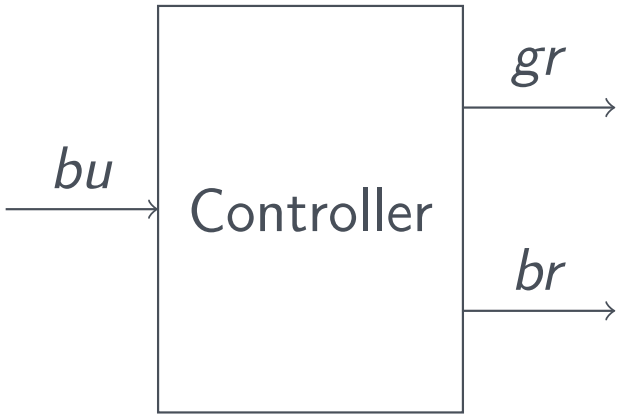
(Basic) Liveness assumptions

Properties of the form $\square\lozenge\psi$, where ψ is a boolean formula over I, O and $\{\bigcirc y \mid y \in I \cup O\}$. Examples:

- $\square\lozenge(bu)$
- $\square\lozenge(\neg br \wedge \neg gr \wedge \bigcirc bu)$

Initialization guarantees

Controller shape



$I = \{bu\}$ (button) $\quad O = \{gr, br\}$ (grind, brew)

Initialization guarantees

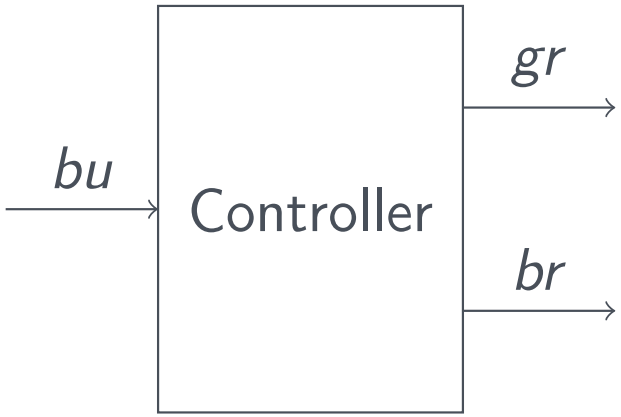
Properties without temporal operators over I and O

Example:

- $\neg gr \wedge \neg br$
- $\neg bu \rightarrow (\neg gr \wedge \neg br)$

Safety guarantees

Controller shape



$I = \{bu\}$ (button) $\quad O = \{gr, br\}$ (grind, brew)

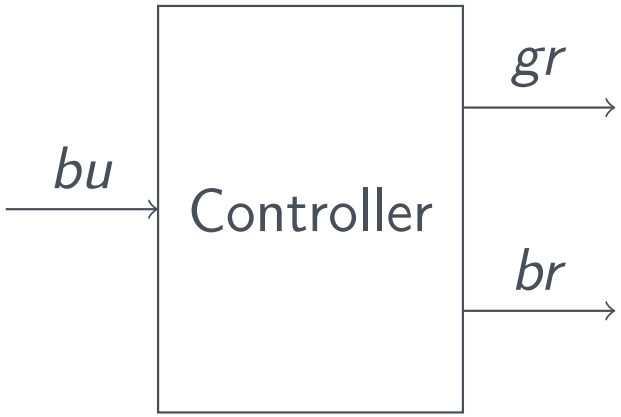
Safety guarantees

Properties of the form $\Box\psi$, where ψ is a boolean formula over I, O and $\{\bigcirc y \mid y \in I \cup O\}$. Examples:

- $\Box(gr \rightarrow \neg \bigcirc gr)$
- $\Box((gr \wedge \bigcirc bu) \rightarrow \bigcirc gr)$

Liveness guarantees

Controller shape



$I = \{bu\}$ (button) $\qquad O = \{gr, br\}$ (grind, brew)

Liveness guarantees

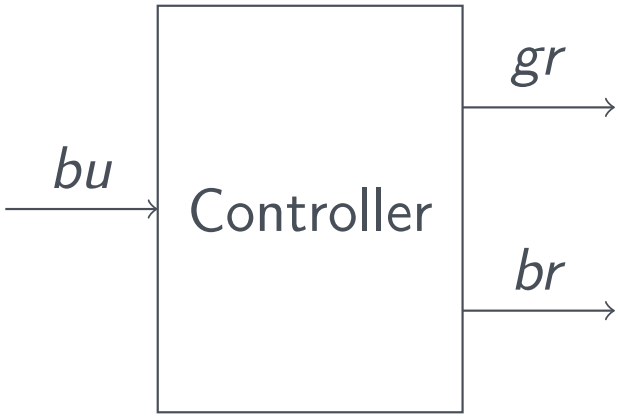
Properties of the form $\square\lozenge\psi$, where ψ is a boolean formula over I, O and $\{\bigcirc y \mid y \in I \cup O\}$. Examples:

- $\square\lozenge(gr \wedge \bigcirc br)$
- $\square\lozenge(bu \vee br)$

Execution semantics

Inputs and outputs

$I = \{bu\}$
$O = \{gr, br\}$



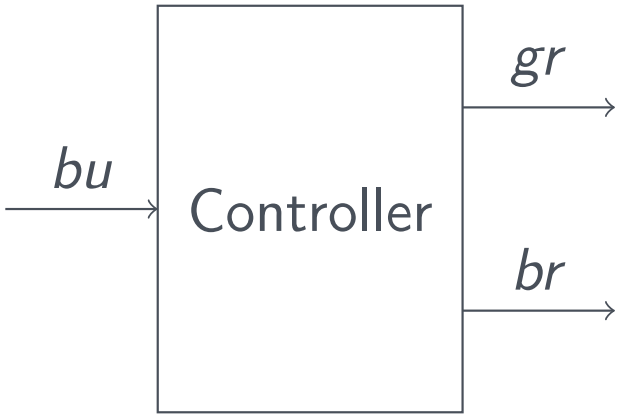
A trace of the system

$$\rho = \left(\right)$$

Execution semantics

Inputs and outputs

$I = \{bu\}$
$O = \{gr, br\}$



A trace of the system

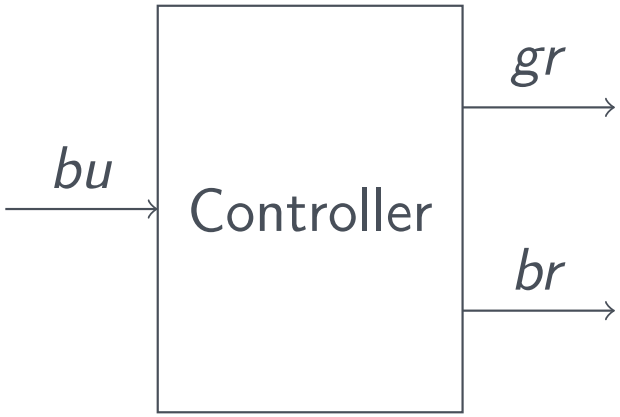
$$\rho = \begin{pmatrix} 0 \\ \\ \end{pmatrix}$$

Step 1

The environment selects values for I that satisfy the initialization assumptions

Inputs and outputs

$I = \{bu\}$
$O = \{gr, br\}$



A trace of the system

$$\rho = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$$

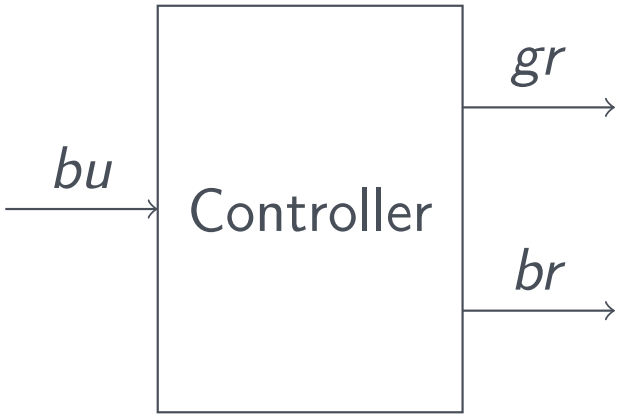
Step 2

The controller selects values for O that satisfy the initialization guarantees

Execution semantics

Inputs and outputs

$I = \{bu\}$
$O = \{gr, br\}$



A trace of the system

$$\rho = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ \end{pmatrix}$$

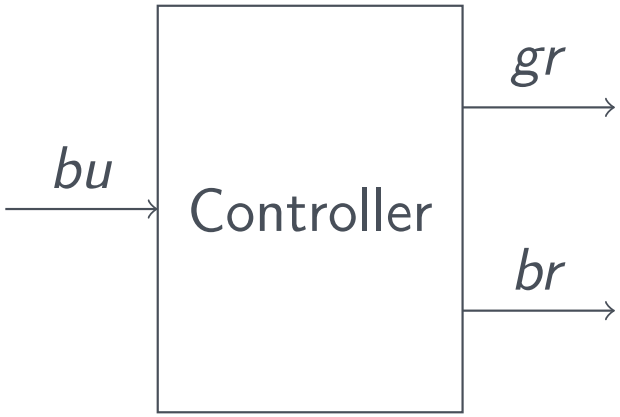
Step $2n + 1$

The environment selects values for I that the last element of ρ and the new values for I satisfy the safety assumptions

Execution semantics

Inputs and outputs

$I = \{bu\}$
$O = \{gr, br\}$



A trace of the system

$$\rho = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}$$

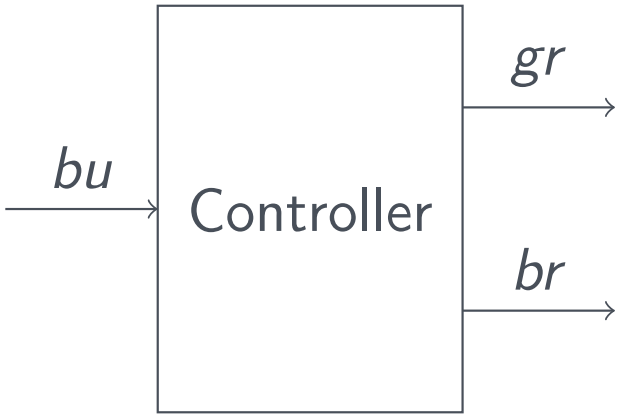
Step $2n + 2$

The controller selects values for O that the last element of ρ and the new values for I and O satisfy the safety guarantees

Execution semantics

Inputs and outputs

$I = \{bu\}$
$O = \{gr, br\}$



A trace of the system

$$\rho = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ \end{pmatrix}$$

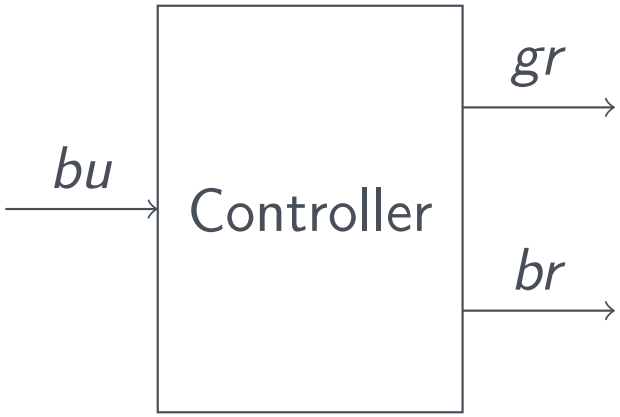
Step $2n + 1$

The environment selects values for I that the last element of ρ and the new values for I satisfy the safety assumptions

Execution semantics

Inputs and outputs

$I = \{bu\}$
$O = \{gr, br\}$



A trace of the system

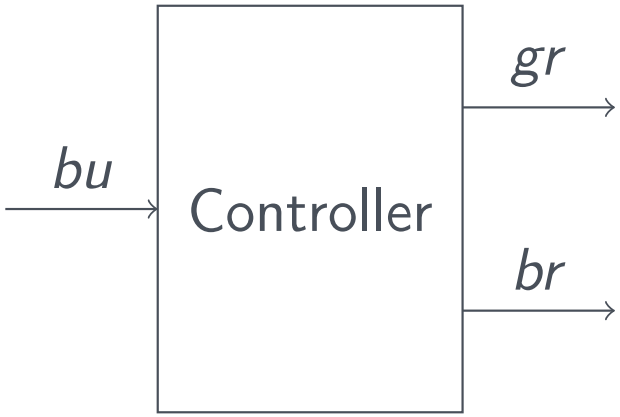
$$\rho = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}$$

Step $2n + 2$

The controller selects values for O that the last element of ρ and the new values for I and O satisfy the safety guarantees

Inputs and outputs

$I = \{bu\}$
$O = \{gr, br\}$



A trace of the system

$$\rho = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix} \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \ldots$$

And so on...

The process continues ad infinitum

Who wins the game?

Finitary winning

- If at some point one of the player violates the rules fo the game then the player doing so first loses the game

Infinitary winning

- If the game continues ad infinitum, then the controller wins if either:
  - the liveness assumptions are violated
  - or the liveness guarantees are satisfied

Let's explore the semantics by example

GR(1) synthesis tool used

Slugs – web-based version available at
`https://webslugs.ruediger-ehlers.de`

Specification

```
[INPUT]
bu

[OUTPUT]
br
gr

[ENV_INIT]

[SYS_INIT]
gr <-> bu
! br
```

Let's explore the semantics by example

Specification (cont'd)

```
[SYS_TRANS]
br' <-> gr
gr' -> bu'

[ENV_TRANS]
bu' -> !gr & !br
```

Let's explore the semantics by example

Specification (cont'd)

```
[SYS_TRANS]
br' <-> gr
gr' -> bu'

[ENV_TRANS]
bu' -> !gr & !br
```

Observation

The system can make coffee, but it does not have to do so

Let's fix the example

Added Liveness Guarantee

```
[SYS_LIVENESS]
br
```

Let's fix the example

Added Liveness Guarantee

```
[SYS_LIVENESS]
br
```

Observation

The system cannot enforce a button press, so it loses

Let's fix the example (2)

Added Liveness Assumption

```
[ENV_LIVENESS]
bu
```

Let's fix the example (2)

Added Liveness Assumption

```
[ENV_LIVENESS]
bu
```

Observation

Now everything works as expected

Credits

The concepts and portions of this presentation have been taken from:

- A Gentle Introduction to Reactive Synthesis by Rüdiger Ehlers, TU Clausthal
  `https://www.ruediger-ehlers.de/blog/introtoreactivesynthesis.html`