22c:181 Formal Methods in Software Engineering

Part II
Reactive Systems and the Lustre language

Christoph Sticksel
christoph-sticksel@uiowa.edu

Verification by Model Checking

- Given:
  - *model:* a system implementation
  - *properties:* requirements in temporal logic

- Is the property *invariant* for all executions of the system?

- Answer: *yes*, or *no* with a *counterexample* trace

- Finite-state and infinite-state systems

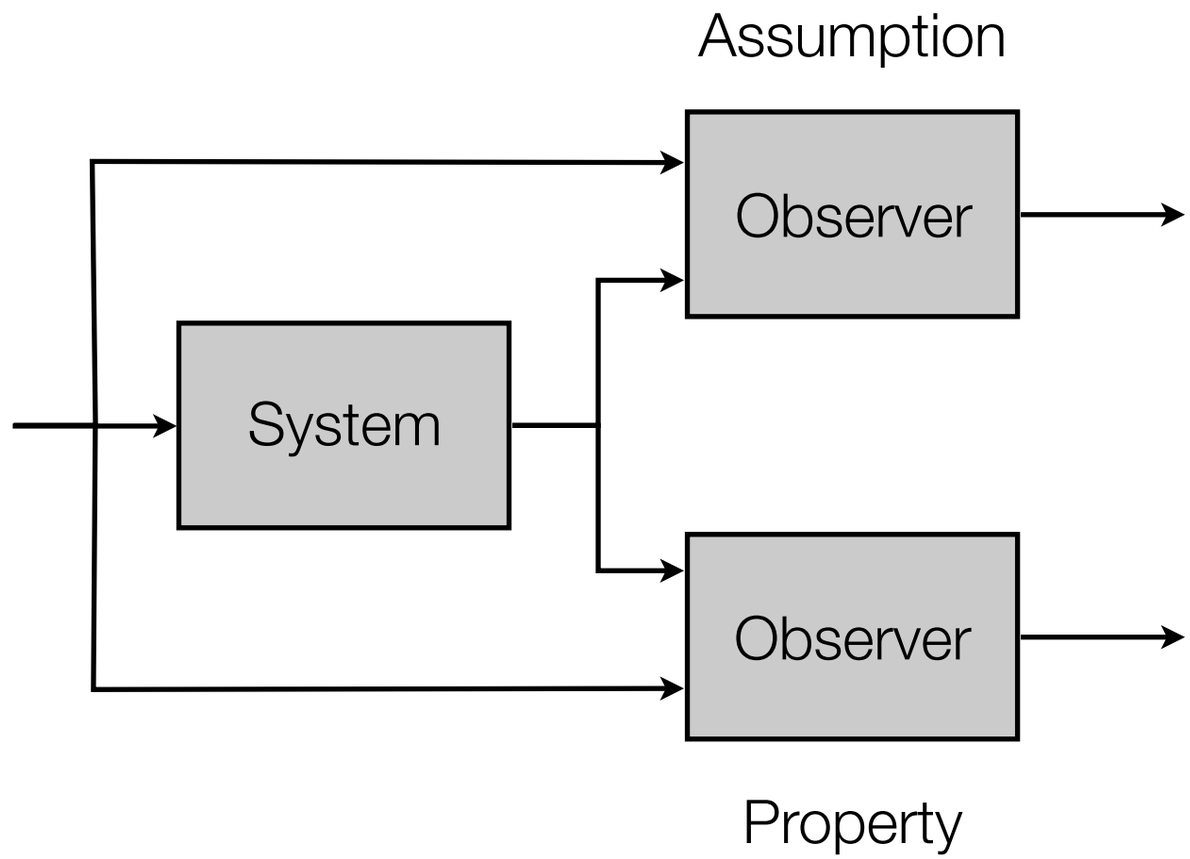
Synchronous Observers

- Validation Program:
  - System under consideration
  - Observer for *assumption*
  - Observer for *property*

- Pre- and postconditions

  “If the assumptions are realistic, then the output is correct.”

- Observers are executable



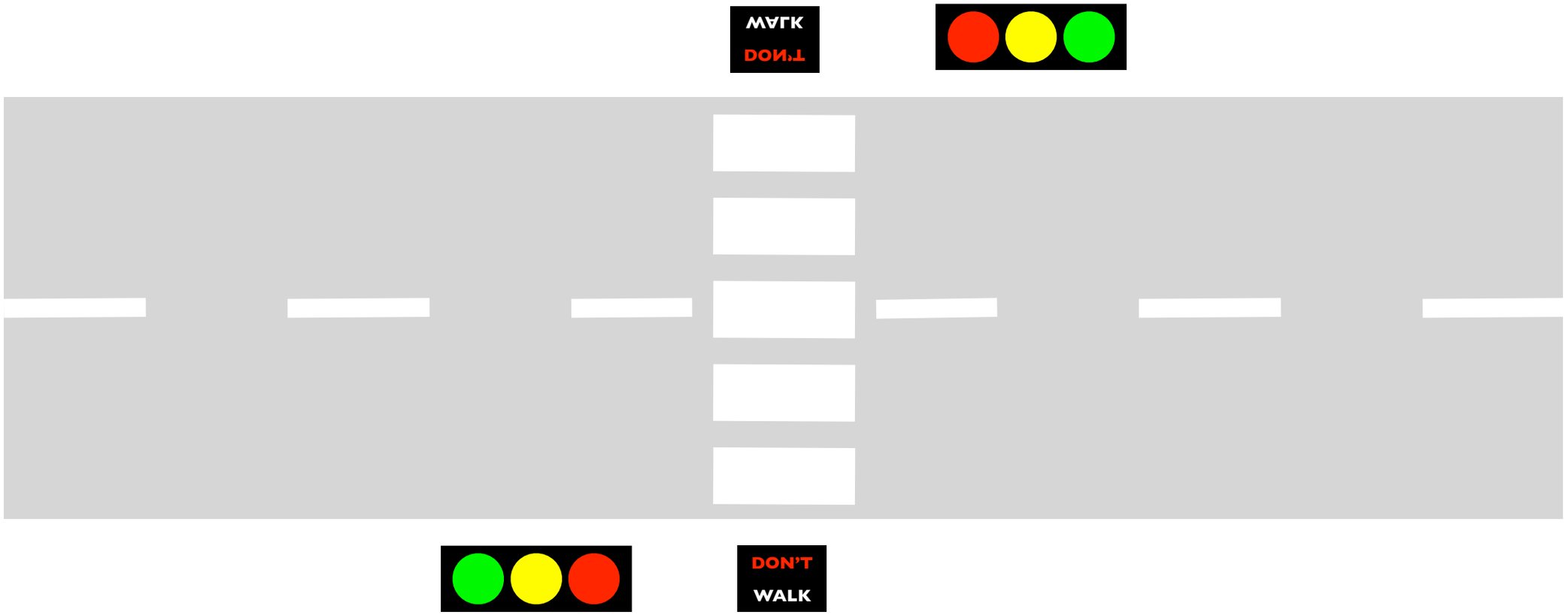
Switch Example

```
node Switch
  (Set, Reset, Init : bool)
returns (X : bool);
let
  X = if Set then true else
      if Reset then false else
          (Init -> pre(X));
tel
```

```
node Switch2
  (Set, Reset, Init : bool)
returns (X : bool);
let
  X = if Reset then false else
      if Set then true else
          (Init -> pre(X));
tel
```

- Do both programs *satisfy the requirements*?
- Are they *equivalent*?

Traffic Lights Example



- What are the requirements?

- Does the implementation satisfy the requirements?