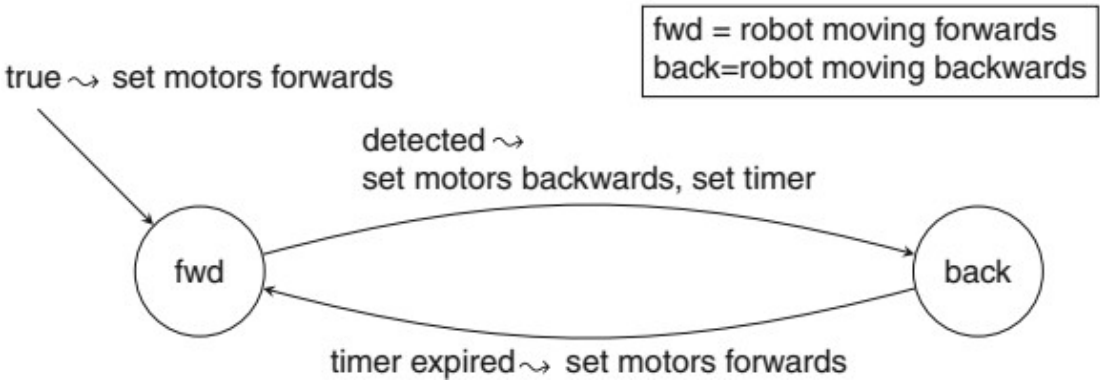
Intro To Robotics

FINITE STATE MACHINES

Reactive Behaviour w/ State

- **Specification (Persistent)**, for a Braitenberg vehicle, non-reactive
  - *Robot move forward until object detected.*
    Then move backward for one second and reverses to move forward again.

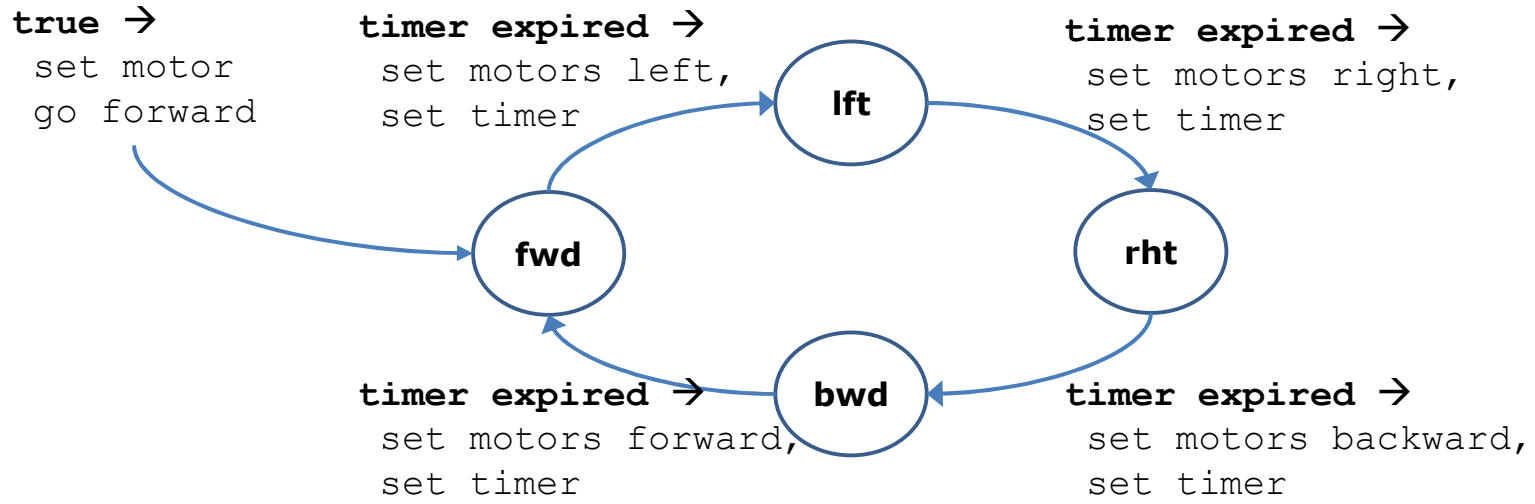


FSM for persistent Braitenberg vehicle

* System turned ON, motors set move forward (condition always **TRUE**, this unconditionally done).
* at **fwd state**: If object detected → transition to state **back**, move backward, timer set
* at **back state**: after one second → transition to state **fwd**, move forward
** If object detected → no action performed, since no transition labeled w/ this condition
***therefore not reactive, depends on current state of robot & event happening

Reactive Behaviour w/ State

- Activity 4.1: **Specification (Consistent)**, for a Braitenberg vehicle
  - *Robot cycle through four states.*
    Changing states every second: forward, turn left, turn right, backward

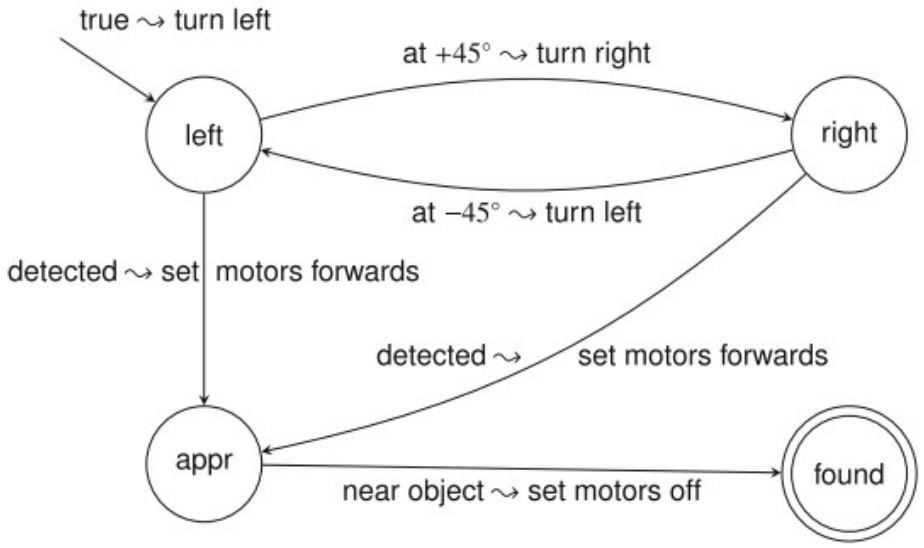


FSM for consistent Braitenberg vehicle

```
fwd = robot moving forward; bwd = robot moving backward;
lft = robot turning left; rht = robot turning right
```

Search & Approach

- **Specification (Search & Approach),**
  - *Robot search left & right (±45°).*
    If object detected, robot approaches object and stops when it's near the object.
- FSM
  - **(3) Final State** *(double circle)*
  - *Finite num of states & transitions*
  - *Behaviour can be finite or infinite*
- Current example behaviour
  - **Finite**: *robot stops when it finds an object & approaches it*
  - **Infinite**: *robot indefinitely continues search if object never found*



FSM for search & approach

left = robot turning left to search
right = robot turning right to search
appr = robot approaching object
found = robot found object

Final state

Assign#03: Finite State Machines

Give the Finite State Machine for the given problem:

- Activity 4.3: **Specification** (**Paranoid**(alternate direction))
  - *Object detected in front → move forward*
    Object detected at right → turn right
    Object detected at left → turn left
    Turning (even if no object detected) → alternate dir of turn every second
    No object detected & not turning → robot stops