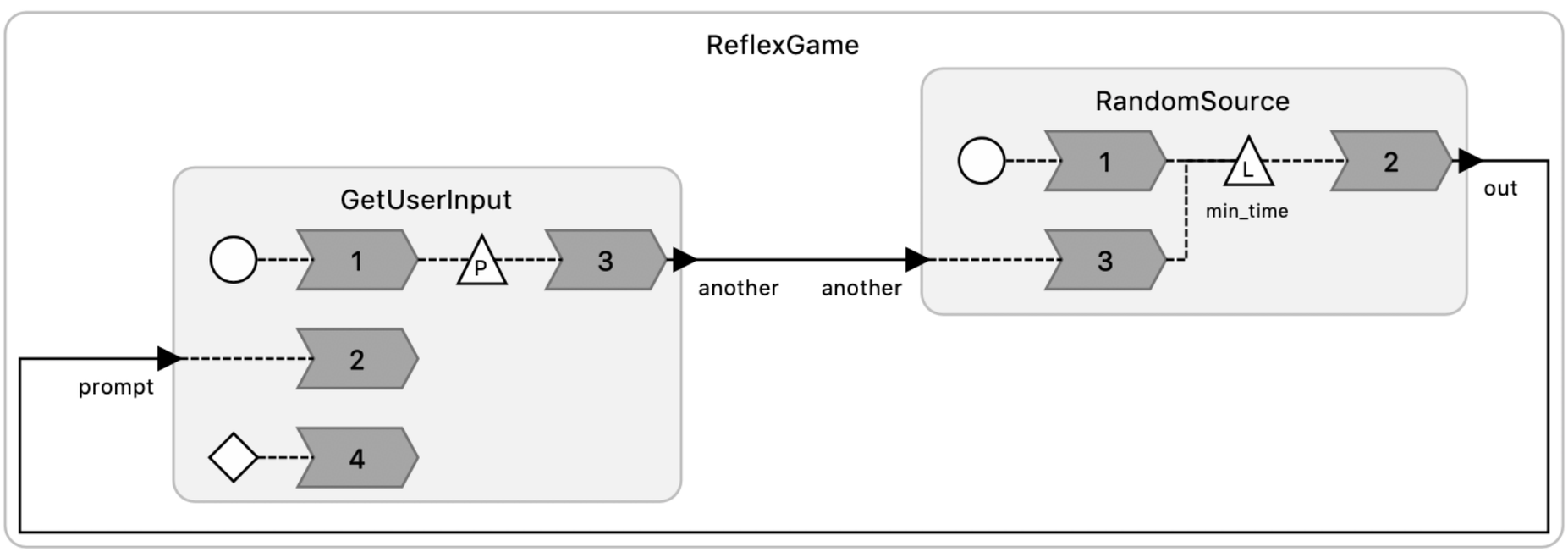
ReflexGame.lf

```
/** This example illustrates the use of logical and physical actions,
 *  asynchronous external inputs, the use of startup and shutdown
 *  reactions, and the use of actions with values.
 *  @author Edward A. Lee
 *  @author Marten Lohstroh
 */
target C {
    threads: 1,
    keepalive: true
};
// Produce a counting sequence at random times with a minimum
// and maximum time between outputs specified as parameters.
reactor RandomSource(min_time:time(2 sec), max_time:time(8 sec)) {
    preamble {=
        // Generate a random additional delay over the minimum.
        // Assume millisecond precision is enough.
        interval_t additional_time(interval_t min_time, interval_t max_time) {
            int interval_in_msec = (max_time - min_time) / MSEC(1);
            return (rand() % interval_in_msec) * MSEC(1);
        }
    =}
    input another:int;
    output out:int;
    logical action prompt(min_time);
    state count:int(0);
```

Tasks | Console | Error Log | Diagram



Actions

Hide all Details

Show all Details

Diagram Options

- [ ] All Reactors
- [x] Remember Collapsed/Expanded
- [x] Dependency Cycle Detection

Appearance

- [x] User Labels (@label in JavaD
- [ ] Expand/Collapse Hyperlinks