```
1 public class HorizontalPneumaticCylinder extends Actor implements Runnable {
 2    protected Terminal<Boolean> airFlow=new Terminal<Boolean>(false);
 3  protected Terminal<Boolean> released=new Terminal<Boolean>(false);
 4  protected Terminal<Boolean> activated=new Terminal<Boolean>(false);
 5  private String targetName=null;
 6   private String directionName;
 7   private int direction=-1;
 8   private int leftSpeed=8;
 9   private int rightSpeed=8;
10   private int speed;
11   private int x0, xf;
12   private int extensionLength=55;
13
14   public HorizontalPneumaticCylinder(JSONObject src) {
15     super(src);
16     if (!src.isNull("leftSpeed"))
17       leftSpeed=src.getInt("leftSpeed");
18     if (!src.isNull("rightSpeed"))
19       rightSpeed=src.getInt("rightSpeed");
20     if (!src.isNull("targetName"))
21       targetName=src.getString("targetName");
22     if (!src.isNull("extensionLength"))
23       extensionLength=src.getInt("extensionLength");
24    if (!src.isNull("direction"))
```

```
25          directionName=src.getString("direction");
26      if (directionName.equals("left")) {
27          direction=-1;
28          x0=x-extensionLength; //93
29          xf=x;
30          this.x=x0;
31      } else {
32          direction=1;
33          x0=x;
34          xf=x+extensionLength; //55
35      }
36    }
37
38  void run() {
39      for (;; ) {
40          speed=0;
41          if (airFlow.read())
42              speed=leftSpeed*direction;
43          else
44              speed=rightSpeed*-direction;
45          x=x+speed;
46          if (x<x0)
47              x=x0;
48          if (x>xf)
```

```
49        x=xf;
50        activated.write(x==xf);
51        released.write(x==x0);
52        try {
53           java.lang.Thread.sleep(samplingTime);
54        }
55        catch (Exception e) {
56           println(e);
57        }
58      }
59    }
60
61  public Terminal getTerminal(String terminalName) {
62      if (terminalName.equals("airFlow"))
63         return airFlow;
64      else
65      if (terminalName.equals("released"))
66         return released;
67      else
68      if (terminalName.equals("activated"))
69         return activated;
70      return super.getTerminal(terminalName);
71    }
72 }
```