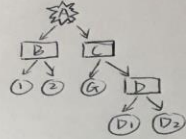


1. Advantage: statecharts 适合于复杂的程序 可以建立模型层次结构 并且具有并行性。
其状态转换可由 condition 来支配, 与 actions 相关联
statecharts 操作可以对变量执行计算, 生成新事件

2. Disadvantage: statecharts 在大型系统中大小变大。statecharts 只有有限的再利用潜力;
广泛使用 actions 将系统状态信息的一部分以状态本身转移到变量。

3. Trees

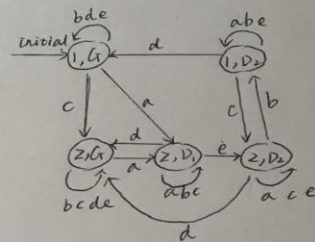
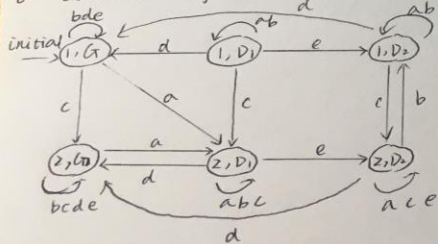


$$\begin{aligned}
 4. \quad Z_A &= Z_B \times Z_C = (Z_1 \cup Z_2) \times (Z_A \times Z_D) \\
 &= (Z_1 \cup Z_2) \times (Z_A \cup (Z_{D1} \cup Z_{D2})) \\
 &= (Z_1, Z_A) \cup (Z_1, Z_{D1}) \cup (Z_1, Z_{D2}) \cup (Z_2, Z_A) \cup (Z_2, Z_{D1}) \cup (Z_2, Z_{D2})
 \end{aligned}$$

5.

Event	State B	State C	State A
initial	1	G	1, G
a	2	D1	2, D1
b	2	D1	2, D1
e	2	D2	2, D2
b	1	D2	1, D2
d	1	G	1, G
b	1	G	1, G

b. Conversion of statechart to a FSM

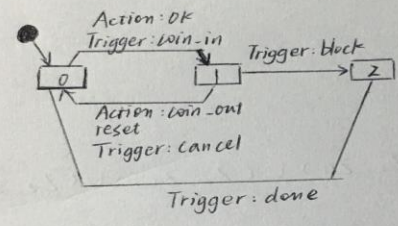


7.

1. $A1.0 \xrightarrow{\text{coin_in}/OK} A1.1$ $A2.A \xrightarrow{OK} A2.B$
 $A2.B \xrightarrow{\text{req_tea}/\text{start_tea}} A2.D$ $A2.D \xrightarrow{\text{drink_ready}/\text{done}} A2.A$
 $A1.1 \xrightarrow{\text{done}} A1.0$

2. $A1.0 \xrightarrow{\text{coin_in}/OK} A1.1$ $A2.A \xrightarrow{OK} A2.B$
 $A2.B \xrightarrow{\text{req_tea}/\text{start_tea}} A2.D$ $A1.1 \xrightarrow{\text{cancel}/\text{coin_out}, \text{reset}} A1.0$
 $A2.D \xrightarrow{\text{drink_ready}/\text{done}} A2.A$

3. 修改 $A1$:



$A2$ 修改 $B \rightarrow C$ 和 $B \rightarrow D$

