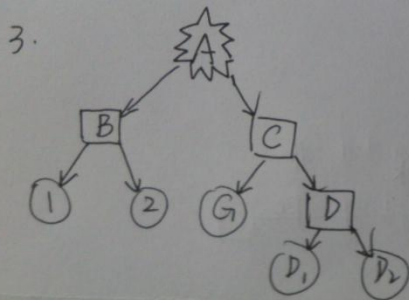


14353421 郑锐奇 M4.

1. ① statechart 适合复杂的程序, 可以建立模型层次结构并有并发性.
- ② 状态转换可以由 condition 监控.
- ③ 状态转换可以和 action 相关联.
- ④ statechart 操作可以对变量进行计算, 以及生成新事件.
2. ① 在大型系统中 statechart 会变大, 变得难以理解.
- ② 只有有限的再利用能力.
- ③ 过度使用 action 会将系统部分状态信息从状态本身转移到变量上.

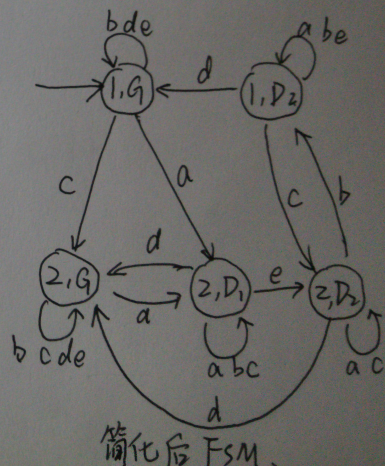
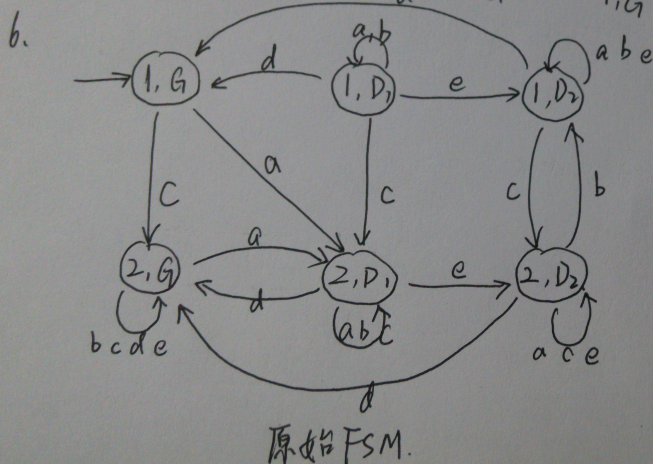


4.

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

5.

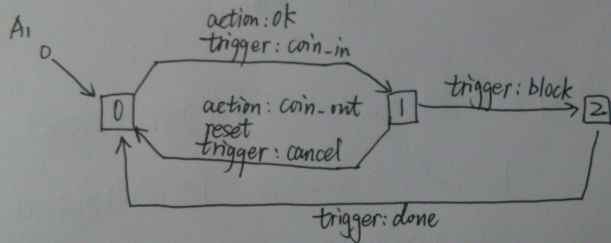
events	State B	State C	State A
Initial	1	G	1, G
a	2	D ₁	2, D ₁
b	2	D ₁	2, D ₁
e	2	D ₂	2, D ₂
b	1	D ₂	1, D ₂
d	1	G	1, G
b	1	G	1, G



7. ① $A_1.0 \xrightarrow{\text{coin-in/OK}} A_1.1$
 $A_2.A \xrightarrow{\text{OK/}} A_2.B$
 $A_2.B \xrightarrow{\text{req-tea/start-tea}} A_2.D$
 $A_2.D \xrightarrow{\text{drink-ready/done}} A_2.A$
 $A_1.1 \xrightarrow{\text{done/}} A_1.0$

② $A_1.0 \xrightarrow{\text{coin-in/OK}} A_1.1$
 $A_2.A \xrightarrow{\text{OK/}} A_2.B$
 $A_2.B \xrightarrow{\text{req-tea/start-tea}} A_2.D$
 $A_1.1 \xrightarrow{\text{cancel/coin-out, reset}} A_1.0$
 $A_2.D \xrightarrow{\text{drink-ready/done}} A_2.A$

③. A_1 修改为下图. A_2 加入 block.



A_2 : 在 $B \rightarrow C$, $B \rightarrow D$ 部分加 block

