1 Formalization

1.1 Syntax

```
nop                        (nop)
mem                        (memory access)
await e                    (event await -- int/ext)
emit e                     (event emit  -- int)
p ; q                      (sequence)
if mem then p else q       (conditional)
loop p                     (loop)
break                      (break loop)
p or q                     (parallel/or)
p and q                    (parallel/and)
```

$$mark\ (i, e)$$

$$mem(v)$$

$$p\ @\ loop\ p$$

1.2 A reaction chain

$$p = program \qquad ext = external\ event$$

$$p_0 = (await\ \$\ ;\ \ <PROG>) \qquad ext_0 = \$$$

$$p \xrightarrow[big]{(i,E)=(0,\{ext\})} \ [\ \ (p' \xrightarrow[small]{i} p'')* \xrightarrow[big]{(i,E)=nxtMarks(p)} \ \ p'''\ \ ]*$$

$$*^1 :\ until\ \ isBlocked(p'')\ \vee\ (p'' = nop)$$

$$*^2 :\ until\ \ nxtMarks(p'') = (0, \emptyset)$$

1.3 Big-step rules

(Await/0) $await\ e \xrightarrow{(i,E)} await\ e\ , \quad (e \notin E)$ **(Await/1)** $await\ e \xrightarrow{(i,E)} nop\ , \quad (e \in E)$

(Mark/emt) $mark\ (i, e \neq \$) \xrightarrow{(i,E)} mark\ (i, \$)$ **(Mark/cnt)** $mark\ (i, \$) \xrightarrow{(i,E)} nop$

(Seq) $$\frac{p \xrightarrow{(i,E)} p'}{(p\ ;\ q) \xrightarrow{(i,E)} (p'\ ;\ q)}$$

$$\textbf{(And)} \quad \frac{p \xrightarrow{(i,E)} p' \quad q \xrightarrow{(i,E)} q'}{(p\ and\ q) \xrightarrow{(i,E)} (p'\ and\ q')}$$

$$\textbf{(Or)} \quad \frac{p \xrightarrow{(i,E)} p' \quad q \xrightarrow{(i,E)} q'}{(p\ or\ q) \xrightarrow{(i,E)} (p'\ or\ q')}$$

1.4 Small-step rules

$$\textbf{(mem)} \quad mem(v) \xrightarrow{i} nop$$

$$\textbf{(emit)} \quad (emit\ e) \xrightarrow{i} mark\ (i+1, e)$$

$$\textbf{(seq/1)} \quad \frac{p \xrightarrow{i} p'}{(p\ ;\ q) \xrightarrow{i} (p'\ ;\ q)}$$

$$\textbf{(seq/2)} \quad (nop\ ;\ q) \xrightarrow{i} q \qquad \textbf{(seq/3)} \quad (break\ ;\ q) \xrightarrow{i} break$$

$$\textbf{(if/0)} \quad (if\ mem(0)\ then\, p\, else\, q) \xrightarrow{i} q \qquad \textbf{(if/1)} \quad (if\ mem(v)\ then\, p\, else\, q) \xrightarrow{i} p\ , \quad (v \neq 0)$$

$$\textbf{(loop)} \quad (loop\ p) \xrightarrow{i} (p\ @\ loop\ p)$$

$$\textbf{(@loop/1)} \quad \frac{p \xrightarrow{i} p'}{(p\ @\ loop\ q) \xrightarrow{i} (p'\ @\ loop\ q)}$$

$$\textbf{(@loop/2)} \quad (nop\ @\ loop\ p) \xrightarrow{i} loop\ p \qquad \textbf{(@loop/3)} \quad (break\ @\ loop\ p) \xrightarrow{i} nop$$

$$\textbf{(and/1)} \quad \frac{p \xrightarrow{i} p'}{(p\ and\ q) \xrightarrow{i} (p'\ and\ q)} \qquad \textbf{(and/2)} \quad \frac{q \xrightarrow{i} q'}{(p\ and\ q) \xrightarrow{i} (p\ and\ q')}$$

$$\textbf{(and/3)} \quad (nop\ and\ q) \xrightarrow{i} q \qquad \textbf{(and/4)} \quad (p\ and\ nop) \xrightarrow{i} p$$

$$\textbf{(and/5)} \quad \frac{q = break\ \vee\ isBlocked(q)}{(break\ and\ q) \xrightarrow{i} break} \qquad \textbf{(and/6)} \quad \frac{p = break\ \vee\ isBlocked(p)}{(p\ and\ break) \xrightarrow{i} break}$$

$$\textbf{(or/1)}\quad \frac{p \xrightarrow{i} p'}{(p\ or\ q) \xrightarrow{i} (p'\ or\ q)} \qquad \textbf{(or/2)}\quad \frac{q \xrightarrow{i} q'}{(p\ or\ q) \xrightarrow{i} (p\ or\ q')}$$

$$\textbf{(or/3)}\quad \frac{q = nop\ \vee\ isBlocked(q)}{(nop\ or\ q) \xrightarrow{i} nop} \qquad \textbf{(or/4)}\quad \frac{p = nop\ \vee\ isBlocked(p)}{(p\ or\ nop) \xrightarrow{i} nop}$$

$$\textbf{(or/5)}\quad \frac{q = nop\ \vee\ q = break\ \vee\ isBlocked(q)}{(break\ or\ q) \xrightarrow{i} break}$$

$$\textbf{(or/6)}\quad \frac{p = nop\ \vee\ p = break\ \vee\ isBlocked(q)}{(p\ or\ break) \xrightarrow{i} break}$$

1.5 Auxiliary functions

$$\begin{aligned} isBlocked(p) = {} & p = await\ e \quad \vee \\ & p = mark\ (i,e) \quad \vee \\ & p = (q\ ;\ r) \quad \wedge \quad isBlocked(q) \quad \vee \\ & p = (loop\ q) \quad \wedge \quad isBlocked(q) \quad \vee \\ & p = (q\ and\ r) \quad \wedge \quad (isBlocked(q) \wedge isBlocked(r)) \quad \vee \\ & p = (q\ or\ r) \quad \wedge \quad (isBlocked(q) \wedge isBlocked(r)) \end{aligned}$$

$$nxtMarks(p) = \begin{cases} (i,\{e\}), & p = mark\ (i,e) \\ nxtMarks(q), & p = (q\ ;\ r) \\ nxtMarks(q), & p = (q\ @\ loop\ r) \\ (i,E), & p = (q\ and/or\ r) \quad \wedge \quad (i,E) = nxtMarks(q) \quad \wedge \\ & \quad (j,F) = nxtMarks(r) \quad \wedge \quad i > j \\ (j,F), & p = (q\ and/or\ r) \quad \wedge \quad (i,E) = nxtMarks(q) \quad \wedge \\ & \quad (j,F) = nxtMarks(r) \quad \wedge \quad i < j \\ (i,E \cup F), & p = (q\ and/or\ r) \quad \wedge \quad (i,E) = nxtMarks(q) \quad \wedge \\ & \quad (j,F) = nxtMarks(r) \quad \wedge \quad i = j \\ (0,\emptyset), & otherwise(await, nop, mem, emit, if, break) \end{cases}$$