Composition Operators for Event-B

CO4EB Rodin plugin

Idir AIT-SADOUNE[1] and Yamine AIT-AMEUR[2]

idir.aitsadoune@supelec.fr[1], yamine@enseeiht.fr[2]

Rodin User and Developer Workshop.
Toulouse, France. June 2 - 3 2014

Plan

Plan

The events composition/order

MACHINE M_0
VARIABLES
var_i
INVARIANTS
$i1:\ I(var_i)$
EVENTS
Initialisation
begin
$a1:\ init(var_i)$
end
Event $Evt_0 \widehat{=}$
when
$g1:\ G_0(var_i)$
then
$a1:\ S_0(var_i)$
end
Event $Evt_1 \widehat{=}$
when
$g1:\ G_1(var_i)$
then
$a1:\ S_1(var_i)$
end
Event $Evt_2 \widehat{=}$
when
$g1:\ G_2(var_i)$
then
$a1:\ S_2(var_i)$
end
END

The events order

- is defined by the evaluation of the $G_i(var_j)$ expressions at each state
- is not explicit in the model

The events composition/order

MACHINE M_0
VARIABLES
var_i
INVARIANTS
i1 : $I(var_i)$
EVENTS
Initialisation
begin
a1 : $init(var_i)$
end
Event $Evt_0 \widehat{=}$
when
g1 : $G_0(var_i)$
then
a1 : $S_0(var_i)$
end
Event $Evt_1 \widehat{=}$
when
g1 : $G_1(var_i)$
then
a1 : $S_1(var_i)$
end
Event $Evt_2 \widehat{=}$
when
g1 : $G_2(var_i)$
then
a1 : $S_2(var_i)$
end
END

The events order

- is defined by the evaluation of the $G_i(var_j)$ expressions at each state
- is not explicit in the model

The events composition/order

MACHINE M_0
VARIABLES
var_i
INVARIANTS
i1 : $I(var_i)$
EVENTS
Initialisation
begin
a1 : $init(var_i)$
end
Event $Evt_0 \widehat{=}$
when
g1 : $G_0(var_i)$
then
a1 : $S_0(var_i)$
end
Event $Evt_1 \widehat{=}$
when
g1 : $G_1(var_i)$
then
a1 : $S_1(var_i)$
end
Event $Evt_2 \widehat{=}$
when
g1 : $G_2(var_i)$
then
a1 : $S_2(var_i)$
end
END

The events order

- is defined by the evaluation of the $G_i(var_j)$ expressions at each state
- is not explicit in the model

The events composition/order

MACHINE M_0
VARIABLES
var_i
INVARIANTS
i1 : $I(var_i)$
EVENTS
Initialisation
begin
a1 : $init(var_i)$
end
Event $Evt_0 \widehat{=}$
when
g1 : $G_0(var_i)$
then
a1 : $S_0(var_i)$
end

Event $Evt_1 \widehat{=}$
when
g1 : $G_1(var_i)$
then
a1 : $S_1(var_i)$
end
Event $Evt_2 \widehat{=}$
when
g1 : $G_2(var_i)$
then
a1 : $S_2(var_i)$
end
END

The events order

- is defined by the evaluation of the $G_i(var_j)$ expressions at each state
- is not explicit in the model

The events composition/order

MACHINE M_0
VARIABLES
var_i
INVARIANTS
i1 : $I(var_i)$
EVENTS
Initialisation
begin
a1 : $init(var_i)$
end
Event $Evt_0 \widehat{=}$
when
g1 : $G_0(var_i)$
then
a1 : $S_0(var_i)$
end

Event $Evt_1 \widehat{=}$
when
g1 : $G_1(var_i)$
then
a1 : $S_1(var_i)$
end
Event $Evt_2 \widehat{=}$
when
g1 : $G_2(var_i)$
then
a1 : $S_2(var_i)$
end
END

The events order

- is defined by the evaluation of the $G_i(var_j)$ expressions at each state
- is not explicit in the model

The events composition/order

MACHINE M_0
VARIABLES
var_i
INVARIANTS
i1 : $I(var_i)$
EVENTS
Initialisation
begin
a1 : $init(var_i)$
end
Event $Evt_0 \widehat{=}$
when
g1 : $G_0(var_i)$
then
a1 : $S_0(var_i)$
end

Event $Evt_1 \widehat{=}$
when
g1 : $G_1(var_i)$
then
a1 : $S_1(var_i)$
end
Event $Evt_2 \widehat{=}$
when
g1 : $G_2(var_i)$
then
a1 : $S_2(var_i)$
end
END

The events order

- is defined by the evaluation of the $G_i(var_j)$ expressions at each state
- is not explicit in the model

Composition operators

Operators definition (BNF rules)

$A ::= A\ ;\ A \mid$ // sequence
$A\ []\ A \mid$ // choice
$A \parallel A \mid$ // independent order
$A^N \mid$ // loop
...
a // atomic

Example

$A_0 ::= A_1 \parallel A_2$
$A_1 ::= A_{11}\ ;\ A_{12}$
$A_0 ::= (\ A_{11}\ ;\ A_{12}\ ) \parallel A_2$

Composition operators

Operators definition (BNF rules)

$A ::= A\ ;\ A \mid$ // sequence
$A\ []\ A \mid$ // choice
$A \parallel A \mid$ // independent order
$A^N \mid$ // loop
...
a // atomic

Example

$A_0 ::= A_1 \parallel A_2$
$A_1 ::= A_{11}\ ;\ A_{12}$
$A_0 ::= (\ A_{11}\ ;\ A_{12}\ ) \parallel A_2$

Composition operators

Operators definition (BNF rules)

$A ::= A\ ;\ A \mid$ // sequence
$A\ []\ A \mid$ // choice
$A \parallel A \mid$ // independent order
$A^N \mid$ // loop
...
a // atomic

Example

$A_0 ::= A_1 \parallel A_2$

$A_1 ::= A_{11}\ ;\ A_{12}$

$A_0 ::= (\ A_{11}\ ;\ A_{12}\ ) \parallel A_2$

Composition operators

Operators definition (BNF rules)

$A ::= A\ ;\ A \mid$ // sequence
$A\ []\ A \mid$ // choice
$A \parallel A \mid$ // independent order
$A^N \mid$ // loop
...
a // atomic

Example

$A_0 ::= A_1 \parallel A_2$

$A_1 ::= A_{11}\ ;\ A_{12}$

$A_0 ::= (\ A_{11}\ ;\ A_{12}\ ) \parallel A_2$

Plan

The proposed approach

Composition operators formalisation

The left hand side part of a BNF rule is modelled by an abstract machine, and the right hand side part by a refinement of this abstract machine.

- A_i action is formalised by Evt_i event
- $A_0 ::= A_1$ op A_2 action is formalised by
  - M_0 Event-B Machine with Evt_0 event
  - M_1 Event-B refinement with Evt_0, Evt_1 and Evt_2 events
  - using an explicit decreasing variant for defining the control flow.

The proposed approach

Composition operators formalisation

The left hand side part of a BNF rule is modelled by an abstract machine, and the right hand side part by a refinement of this abstract machine.

- A_i action is formalised by Evt_i event
- $A_0 ::= A_1$ op A_2 action is formalised by
  - M_0 Event-B Machine with Evt_0 event
  - M_1 Event-B refinement with Evt_0, Evt_1 and Evt_2 events
  - using an explicit decreasing variant for defining the control flow.

The proposed approach

Composition operators formalisation

The left hand side part of a BNF rule is modelled by an abstract machine, and the right hand side part by a refinement of this abstract machine.

- A_i action is formalised by Evt_i event
- $A_0 ::= A_1$ op A_2 action is formalised by
  - M_0 Event-B Machine with Evt_0 event
  - M_1 Event-B refinement with Evt_0, Evt_1 and Evt_2 events
  - using an explicit decreasing variant for defining the control flow.

The sequence operator : $A_0 ::= A_1 ; A_2$

MACHINE M_1
VARIABLES
var_1 $varSeq$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varSeq \in \{0, 1, 2\}$
VARIANT
$varSeq$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varSeq := 2$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varSeq = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varSeq = 2$
then
act1 : $A_1(var_1)$
act2 :
$varSeq := varSeq - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varSeq = 1$
then
act1 : $A_2(var_1)$
act2 :
$varSeq := varSeq - 1$
end

The sequence operator : $A_0 ::= A_1 ; A_2$

MACHINE M_1
VARIABLES
var_1 $varSeq$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varSeq \in \{0, 1, 2\}$
VARIANT
$varSeq$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varSeq := 2$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varSeq = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varSeq = 2$
then
act1 : $A_1(var_1)$
act2 : $varSeq := varSeq - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varSeq = 1$
then
act1 : $A_2(var_1)$
act2 : $varSeq := varSeq - 1$
end

The sequence operator : $A_0 ::= A_1 ; A_2$

MACHINE M_1
VARIABLES
var_1 $varSeq$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varSeq \in \{0, 1, 2\}$
VARIANT
$varSeq$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varSeq := 2$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varSeq = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varSeq = 2$
then
act1 : $A_1(var_1)$
act2 :
$varSeq := varSeq - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varSeq = 1$
then
act1 : $A_2(var_1)$
act2 :
$varSeq := varSeq - 1$
end

The sequence operator : $A_0 ::= A_1 ; A_2$

MACHINE M_1
VARIABLES
var_1 $varSeq$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varSeq \in \{0, 1, 2\}$
VARIANT
$varSeq$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varSeq := 2$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varSeq = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varSeq = 2$
then
act1 : $A_1(var_1)$
act2 :
$varSeq := varSeq - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varSeq = 1$
then
act1 : $A_2(var_1)$
act2 :
$varSeq := varSeq - 1$
end

The sequence operator : $A_0 ::= A_1 ; A_2$

MACHINE M_1
VARIABLES
var_1 $varSeq$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varSeq \in \{0, 1, 2\}$
VARIANT
$varSeq$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varSeq := 2$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varSeq = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varSeq = 2$
then
act1 : $A_1(var_1)$
act2 :
$varSeq := varSeq - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varSeq = 1$
then
act1 : $A_2(var_1)$
act2 :
$varSeq := varSeq - 1$
end

The parallel operator : $A_0 ::= A_1 \parallel A_2$

MACHINE M_1
VARIABLES
var_1 $varPar_1$ $varPar_2$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varPar_1, varPar_2 := 1, 1$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varPar_1 = 0$
grd3 : $varPar_2 = 0$
then
act : $A'(w)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varPar_1 = 1$
then
act1 : $A_1(var_1)$
act2 : $varPar_1 := varPar_1 - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varPar_2 = 1$
then
act1 : $A_2(var_1)$
act2 : $varPar_2 := varPar_2 - 1$
end

The parallel operator : $A_0 ::= A_1 \parallel A_2$

MACHINE M_1
VARIABLES
var_1 $varPar_1$ $varPar_2$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varPar_1, varPar_2 := 1, 1$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varPar_1 = 0$
grd3 : $varPar_2 = 0$
then
act : $A'(w)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varPar_1 = 1$
then
act1 : $A_1(var_1)$
act2 : $varPar_1 := varPar_1 - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varPar_2 = 1$
then
act1 : $A_2(var_1)$
act2 : $varPar_2 := varPar_2 - 1$
end

The parallel operator : $A_0 ::= A_1 \parallel A_2$

MACHINE M_1
VARIABLES
var_1 $varPar_1$ $varPar_2$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varPar_1, varPar_2 := 1, 1$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varPar_1 = 0$
grd3 : $varPar_2 = 0$
then
act : $A'(w)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varPar_1 = 1$
then
act1 : $A_1(var_1)$
act2 : $varPar_1 := varPar_1 - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varPar_2 = 1$
then
act1 : $A_2(var_1)$
act2 : $varPar_2 := varPar_2 - 1$
end

The parallel operator : $A_0 ::= A_1 \parallel A_2$

MACHINE M_1
VARIABLES
var_1 $varPar_1$ $varPar_2$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varPar_1, varPar_2 := 1, 1$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varPar_1 = 0$
grd3 : $varPar_2 = 0$
then
act : $A'(w)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varPar_1 = 1$
then
act1 : $A_1(var_1)$
act2 : $varPar_1 := varPar_1 - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varPar_2 = 1$
then
act1 : $A_2(var_1)$
act2 : $varPar_2 := varPar_2 - 1$
end

The choice operator : $A_0 ::= A_1 \; [] \; A_2$

MACHINE M_1
VARIABLES
var_1 $varCho$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varCho \in \{0, 1, 2\}$
VARIANT
$varCho$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varCho :\in \{1, 2\}$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varCho = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varCho = 1$
then
act1 : $A_1(var_1)$
act2 :
$varCho := varCho - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varCho = 2$
then
act1 : $A_2(var_1)$
act2 :
$varCho := varCho - 2$
end

The choice operator : $A_0 ::= A_1 \; [] \; A_2$

MACHINE M_1
VARIABLES
var_1 $varCho$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varCho \in \{0, 1, 2\}$
VARIANT
$varCho$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varCho :\in \{1, 2\}$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varCho = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varCho = 1$
then
act1 : $A_1(var_1)$
act2 :
$varCho := varCho - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varCho = 2$
then
act1 : $A_2(var_1)$
act2 :
$varCho := varCho - 2$
end

The choice operator : $A_0 ::= A_1 \; [] \; A_2$

MACHINE M_1
VARIABLES
var_1 $varCho$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varCho \in \{0, 1, 2\}$
VARIANT
$varCho$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varCho :\in \{1, 2\}$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varCho = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varCho = 1$
then
act1 : $A_1(var_1)$
act2 :
$varCho := varCho - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varCho = 2$
then
act1 : $A_2(var_1)$
act2 :
$varCho := varCho - 2$
end

The choice operator : $A_0 ::= A_1 \; [] \; A_2$

MACHINE M_1
VARIABLES
var_1 $varCho$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varCho \in \{0, 1, 2\}$
VARIANT
$varCho$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varCho :\in \{1, 2\}$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varCho = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varCho = 1$
then
act1 : $A_1(var_1)$
act2 :
$varCho := varCho - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varCho = 2$
then
act1 : $A_2(var_1)$
act2 :
$varCho := varCho - 2$
end

The choice operator : $A_0 ::= A_1 \; [] \; A_2$

MACHINE M_1
VARIABLES
var_1 $varCho$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varCho \in \{0, 1, 2\}$
VARIANT
$varCho$
EVENTS
Initialisation
begin
act1 : $Init(var_1)$
act2 : $varCho :\in \{1, 2\}$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varCho = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varCho = 1$
then
act1 : $A_1(var_1)$
act2 :
$varCho := varCho - 1$
end

Event $evt_2 \widehat{=}$
Status convergent
when
grd1 : $G_2(var_1)$
grd2 : $varCho = 2$
then
act1 : $A_2(var_1)$
act2 :
$varCho := varCho - 2$
end

The loop operator : $A_0 := A_1{}^n$

MACHINE M_1
VARIABLES
var_1 $varLoop$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varLoop \in \mathbb{N}$
VARIANT
$varLoop$
EVENTS

Initialisation
begin
act1 : $Init(var_1)$
act2 : $varLoop :\in \mathbb{N}1$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varLoop = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varLoop \neq 0$
then
act1 : $A_1(var_1)$
act2 :
$varLoop := varLoop - 1$
end

The loop operator : $A_0 := A_1{}^n$

MACHINE M_1
VARIABLES
var_1 $varLoop$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varLoop \in \mathbb{N}$
VARIANT
$varLoop$
EVENTS

Initialisation
begin
act1 : $Init(var_1)$
act2 : $varLoop :\in \mathbb{N}1$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varLoop = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varLoop \neq 0$
then
act1 : $A_1(var_1)$
act2 :
$varLoop := varLoop - 1$
end

The loop operator : $A_0 := A_1{}^n$

MACHINE M_1
VARIABLES
var_1 $varLoop$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varLoop \in \mathbb{N}$
VARIANT
$varLoop$
EVENTS

Initialisation
begin
act1 : $Init(var_1)$
act2 : $varLoop :\in \mathbb{N}1$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varLoop = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varLoop \neq 0$
then
act1 : $A_1(var_1)$
act2 :
$varLoop := varLoop - 1$
end

The loop operator : $A_0 := A_1{}^n$

MACHINE M_1
VARIABLES
var_1 $varLoop$
INVARIANTS
inv1 : $I(var_1)$
inv2 : $varLoop \in \mathbb{N}$
VARIANT
$varLoop$
EVENTS

Initialisation
begin
act1 : $Init(var_1)$
act2 : $varLoop :\in \mathbb{N}1$
end

Event $evt_0 \widehat{=}$
when
grd1 : $G'(var_1)$
grd2 : $varLoop = 0$
then
act : $A'(var_1)$
end

Event $evt_1 \widehat{=}$
Status convergent
when
grd1 : $G_1(var_1)$
grd2 : $varLoop \neq 0$
then
act1 : $A_1(var_1)$
act2 :
$varLoop := varLoop - 1$
end

The proposed approach

example

$A_0 ::= A_1 \parallel A_2$

$A_1 ::= A_{11} ; A_{12}$

$A_0 ::= (A_{11} ; A_{12}) \parallel A_2$

Machine $M_0 \mathrel{\widehat{=}} A_0$

MACHINE M_0
VARIABLES
  var_i
INVARIANTS
  i : $I(var_i)$
EVENTS
  Initialisation
    begin
      a1 : $init(var_i)$
    end
  Event $Evt_0 \mathrel{\widehat{=}}$
    when
      g : $G_0(var_i)$
    then
      a : $S_0(var_i)$
    end
  END

The proposed approach

Machine $M_1 \widehat{=} A_1 \parallel A_2$,
variant = $varPar_1 + varPar_2$

MACHINE M_1
REFINES M_0
VARIABLES
var_j $varPar_1$ $varPar_2$
INVARIANTS
i1 : $J(var_j, var_i)$
i2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
a1 : $init(var_j)$
a2 : $varPar_1, varPar_2 := 1, 1$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G'_0(var_j)$
g2 : $varPar_1 = 0 \wedge varPar_2 = 0$
then
a : $S'_0(var_j)$
end
Event $Evt_1 \widehat{=}$
Status convergent
when
g1 : $G_1(var_j)$
g2 : $varPar_1 = 1$
then
a1 : $S_1(var_j)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
Status convergent
when
g1 : $G_2(var_j)$
g2 : $varPar_2 = 1$
then
a1 : $S_2(var_j)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_1 \widehat{=} A_1 \,||\, A_2$,
variant = $varPar_1 + varPar_2$

MACHINE M_1
REFINES M_0
VARIABLES
var_j $varPar_1$ $varPar_2$
INVARIANTS
i1 : $J(var_j, var_i)$
i2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
a1 : $init(var_j)$
a2 : $varPar_1, varPar_2 := 1, 1$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G'_0(var_j)$
g2 : $varPar_1 = 0 \wedge varPar_2 = 0$
then
a : $S'_0(var_j)$
end
Event $Evt_1 \widehat{=}$
Status convergent
when
g1 : $G_1(var_j)$
g2 : $varPar_1 = 1$
then
a1 : $S_1(var_j)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
Status convergent
when
g1 : $G_2(var_j)$
g2 : $varPar_2 = 1$
then
a1 : $S_2(var_j)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_1 \widehat{=} A_1 \,||\, A_2$,
variant = $varPar_1 + varPar_2$

MACHINE M_1
REFINES M_0
VARIABLES
var_j $varPar_1$ $varPar_2$
INVARIANTS
i1 : $J(var_j, var_i)$
i2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
a1 : $init(var_j)$
a2 : $varPar_1, varPar_2 := 1, 1$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G'_0(var_j)$
g2 : $varPar_1 = 0 \wedge varPar_2 = 0$
then
a : $S'_0(var_j)$
end

Event $Evt_1 \widehat{=}$
Status convergent
when
g1 : $G_1(var_j)$
g2 : $varPar_1 = 1$
then
a1 : $S_1(var_j)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
Status convergent
when
g1 : $G_2(var_j)$
g2 : $varPar_2 = 1$
then
a1 : $S_2(var_j)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_1 \widehat{=} A_1 \,||\, A_2$,
variant = $varPar_1 + varPar_2$

MACHINE M_1
REFINES M_0
VARIABLES
var_j $varPar_1$ $varPar_2$
INVARIANTS
i1 : $J(var_j, var_i)$
i2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
a1 : $init(var_j)$
a2 : $varPar_1, varPar_2 := 1, 1$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G'_0(var_j)$
g2 : $varPar_1 = 0 \wedge varPar_2 = 0$
then
a : $S'_0(var_j)$
end

Event $Evt_1 \widehat{=}$
Status convergent
when
g1 : $G_1(var_j)$
g2 : $varPar_1 = 1$
then
a1 : $S_1(var_j)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
Status convergent
when
g1 : $G_2(var_j)$
g2 : $varPar_2 = 1$
then
a1 : $S_2(var_j)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_1 \widehat{=} A_1 \,||\, A_2$,
variant = $varPar_1 + varPar_2$

MACHINE M_1
REFINES M_0
VARIABLES
var_j $varPar_1$ $varPar_2$
INVARIANTS
i1 : $J(var_j, var_i)$
i2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
a1 : $init(var_j)$
a2 : $varPar_1, varPar_2 := 1, 1$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G'_0(var_j)$
g2 : $varPar_1 = 0 \wedge varPar_2 = 0$
then
a : $S'_0(var_j)$
end

Event $Evt_1 \widehat{=}$
Status convergent
when
g1 : $G_1(var_j)$
g2 : $varPar_1 = 1$
then
a1 : $S_1(var_j)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
Status convergent
when
g1 : $G_2(var_j)$
g2 : $varPar_2 = 1$
then
a1 : $S_2(var_j)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_1 \widehat{=} A_1 \,||\, A_2$,
variant = $varPar_1 + varPar_2$

MACHINE M_1
REFINES M_0
VARIABLES
var_j $varPar_1$ $varPar_2$
INVARIANTS
i1 : $J(var_j, var_i)$
i2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
a1 : $init(var_j)$
a2 : $varPar_1, varPar_2 := 1, 1$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G'_0(var_j)$
g2 : $varPar_1 = 0 \wedge varPar_2 = 0$
then
a : $S'_0(var_j)$
end

Event $Evt_1 \widehat{=}$
Status convergent
when
g1 : $G_1(var_j)$
g2 : $varPar_1 = 1$
then
a1 : $S_1(var_j)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
Status convergent
when
g1 : $G_2(var_j)$
g2 : $varPar_2 = 1$
then
a1 : $S_2(var_j)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_1 \widehat{=} A_1 \,||\, A_2$,
variant = $varPar_1 + varPar_2$

MACHINE M_1
REFINES M_0
VARIABLES
var_j $varPar_1$ $varPar_2$
INVARIANTS
i1 : $J(var_j, var_i)$
i2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
a1 : $init(var_j)$
a2 : $varPar_1, varPar_2 := 1, 1$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G'_0(var_j)$
g2 : $varPar_1 = 0 \wedge varPar_2 = 0$
then
a : $S'_0(var_j)$
end

Event $Evt_1 \widehat{=}$
Status convergent
when
g1 : $G_1(var_j)$
g2 : $varPar_1 = 1$
then
a1 : $S_1(var_j)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
Status convergent
when
g1 : $G_2(var_j)$
g2 : $varPar_2 = 1$
then
a1 : $S_2(var_j)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_1 \widehat{=} A_1 \,||\, A_2$,
variant = $varPar_1 + varPar_2$

MACHINE M_1
REFINES M_0
VARIABLES
var_j $varPar_1$ $varPar_2$
INVARIANTS
i1 : $J(var_j, var_i)$
i2 : $varPar_1 \in \{0, 1\} \wedge varPar_2 \in \{0, 1\}$
VARIANT
$varPar_1 + varPar_2$
EVENTS
Initialisation
begin
a1 : $init(var_j)$
a2 : $varPar_1, varPar_2 := 1, 1$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G'_0(var_j)$
g2 : $varPar_1 = 0 \wedge varPar_2 = 0$
then
a : $S'_0(var_j)$
end

Event $Evt_1 \widehat{=}$
Status convergent
when
g1 : $G_1(var_j)$
g2 : $varPar_1 = 1$
then
a1 : $S_1(var_j)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
Status convergent
when
g1 : $G_2(var_j)$
g2 : $varPar_2 = 1$
then
a1 : $S_2(var_j)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_2 \widehat{=} (A_{11} ; A_{12}) \| A_2$

variant = $(varPar_1 + varPar_2) + varSeq$

MACHINE M_2
REFINES M_1
VARIABLES
var_k varSeq
INVARIANTS
i1 : $K(var_k, var_j, var_i)$
i2 : $varSeq \in \{0, 1, 2\}$
VARIANT
varSeq
EVENTS
Initialisation
begin
a1 : $init(var_k)$
a2 : $varSeq := 2$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G_0''(var_k)$
g2 : $varPar_1 = 0$
g3 : $varPar_2 = 0$
then
a : $S_0''(var_k)$
end

Event $Evt_{11} \widehat{=}$
Status convergent
when
g1 : $G_{11}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 2$
then
a1 : $S_{11}(var_k)$
a2 : $varSeq := varSeq - 1$
end
Event $Evt_{12} \widehat{=}$
Status convergent
when
g1 : $G_{12}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 1$
then
a1 : $S_{12}(var_k)$
a2 : $varSeq := varSeq - 1$
end

Event $Evt_1 \widehat{=}$
refines Evt_1
when
g1 : $G_1'(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 0$
then
a1 : $S_1'(var_k)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
refines Evt_2
when
g1 : $G_2'(var_k)$
g2 : $varPar_2 = 1$
then
a1 : $S_2'(var_k)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_2 \widehat{=} (A_{11} ; A_{12}) \| A_2$

variant = $(varPar_1 + varPar_2) + varSeq$

MACHINE M_2
REFINES M_1
VARIABLES
var_k varSeq
INVARIANTS
i1 : $K(var_k, var_j, var_i)$
i2 : $varSeq \in \{0, 1, 2\}$
VARIANT
varSeq
EVENTS
Initialisation
begin
a1 : $init(var_k)$
a2 : $varSeq := 2$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G_0''(var_k)$
g2 : $varPar_1 = 0$
g3 : $varPar_2 = 0$
then
a : $S_0''(var_k)$
end

Event $Evt_{11} \widehat{=}$
Status convergent
when
g1 : $G_{11}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 2$
then
a1 : $S_{11}(var_k)$
a2 : $varSeq := varSeq - 1$
end
Event $Evt_{12} \widehat{=}$
Status convergent
when
g1 : $G_{12}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 1$
then
a1 : $S_{12}(var_k)$
a2 : $varSeq := varSeq - 1$
end

Event $Evt_1 \widehat{=}$
refines Evt_1
when
g1 : $G_1'(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 0$
then
a1 : $S_1'(var_k)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
refines Evt_2
when
g1 : $G_2'(var_k)$
g2 : $varPar_2 = 1$
then
a1 : $S_2'(var_k)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_2 \widehat{=} (A_{11} ; A_{12}) \| A_2$

variant = $(varPar_1 + varPar_2) + varSeq$

MACHINE M_2
REFINES M_1
VARIABLES
var_k varSeq
INVARIANTS
i1 : $K(var_k, var_j, var_i)$
i2 : $varSeq \in \{0, 1, 2\}$
VARIANT
varSeq
EVENTS
Initialisation
begin
a1 : $init(var_k)$
a2 : $varSeq := 2$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G_0''(var_k)$
g2 : $varPar_1 = 0$
g3 : $varPar_2 = 0$
then
a : $S_0''(var_k)$
end

Event $Evt_{11} \widehat{=}$
Status convergent
when
g1 : $G_{11}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 2$
then
a1 : $S_{11}(var_k)$
a2 : $varSeq := varSeq - 1$
end
Event $Evt_{12} \widehat{=}$
Status convergent
when
g1 : $G_{12}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 1$
then
a1 : $S_{12}(var_k)$
a2 : $varSeq := varSeq - 1$
end

Event $Evt_1 \widehat{=}$
refines Evt_1
when
g1 : $G_1'(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 0$
then
a1 : $S_1'(var_k)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
refines Evt_2
when
g1 : $G_2'(var_k)$
g2 : $varPar_2 = 1$
then
a1 : $S_2'(var_k)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_2 \widehat{=} (A_{11} ; A_{12}) \| A_2$

variant = $(varPar_1 + varPar_2) + varSeq$

MACHINE M_2
REFINES M_1
VARIABLES
var_k varSeq
INVARIANTS
i1 : $K(var_k, var_j, var_i)$
i2 : $varSeq \in \{0, 1, 2\}$
VARIANT
varSeq
EVENTS
Initialisation
begin
a1 : $init(var_k)$
a2 : $varSeq := 2$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G_0''(var_k)$
g2 : $varPar_1 = 0$
g3 : $varPar_2 = 0$
then
a : $S_0''(var_k)$
end

Event $Evt_{11} \widehat{=}$
Status convergent
when
g1 : $G_{11}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 2$
then
a1 : $S_{11}(var_k)$
a2 : $varSeq := varSeq - 1$
end
Event $Evt_{12} \widehat{=}$
Status convergent
when
g1 : $G_{12}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 1$
then
a1 : $S_{12}(var_k)$
a2 : $varSeq := varSeq - 1$
end

Event $Evt_1 \widehat{=}$
refines Evt_1
when
g1 : $G_1'(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 0$
then
a1 : $S_1'(var_k)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
refines Evt_2
when
g1 : $G_2'(var_k)$
g2 : $varPar_2 = 1$
then
a1 : $S_2'(var_k)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_2 \widehat{=} (A_{11} ; A_{12}) \| A_2$

variant = $(varPar_1 + varPar_2) + varSeq$

MACHINE M_2
REFINES M_1
VARIABLES
var_k varSeq
INVARIANTS
i1 : $K(var_k, var_j, var_i)$
i2 : $varSeq \in \{0, 1, 2\}$
VARIANT
varSeq
EVENTS
Initialisation
begin
a1 : $init(var_k)$
a2 : $varSeq := 2$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G_0''(var_k)$
g2 : $varPar_1 = 0$
g3 : $varPar_2 = 0$
then
a : $S_0''(var_k)$
end

Event $Evt_{11} \widehat{=}$
Status convergent
when
g1 : $G_{11}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 2$
then
a1 : $S_{11}(var_k)$
a2 : $varSeq := varSeq - 1$
end
Event $Evt_{12} \widehat{=}$
Status convergent
when
g1 : $G_{12}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 1$
then
a1 : $S_{12}(var_k)$
a2 : $varSeq := varSeq - 1$
end

Event $Evt_1 \widehat{=}$
refines Evt_1
when
g1 : $G_1'(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 0$
then
a1 : $S_1'(var_k)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
refines Evt_2
when
g1 : $G_2'(var_k)$
g2 : $varPar_2 = 1$
then
a1 : $S_2'(var_k)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_2 \widehat{=} (A_{11} ; A_{12}) \| A_2$

variant = $(varPar_1 + varPar_2) + varSeq$

MACHINE M_2
REFINES M_1
VARIABLES
var_k $varSeq$
INVARIANTS
i1 : $K(var_k, var_j, var_i)$
i2 : $varSeq \in \{0, 1, 2\}$
VARIANT
$varSeq$
EVENTS
Initialisation
begin
a1 : $init(var_k)$
a2 : $varSeq := 2$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G_0''(var_k)$
g2 : $varPar_1 = 0$
g3 : $varPar_2 = 0$
then
a : $S_0''(var_k)$
end

Event $Evt_{11} \widehat{=}$
Status convergent
when
g1 : $G_{11}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 2$
then
a1 : $S_{11}(var_k)$
a2 : $varSeq := varSeq - 1$
end
Event $Evt_{12} \widehat{=}$
Status convergent
when
g1 : $G_{12}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 1$
then
a1 : $S_{12}(var_k)$
a2 : $varSeq := varSeq - 1$
end

Event $Evt_1 \widehat{=}$
refines Evt_1
when
g1 : $G_1'(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 0$
then
a1 : $S_1'(var_k)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
refines Evt_2
when
g1 : $G_2'(var_k)$
g2 : $varPar_2 = 1$
then
a1 : $S_2'(var_k)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_2 \widehat{=} (A_{11} ; A_{12}) \| A_2$

variant = $(varPar_1 + varPar_2) + varSeq$

MACHINE M_2
REFINES M_1
VARIABLES
var_k $varSeq$
INVARIANTS
i1 : $K(var_k, var_j, var_i)$
i2 : $varSeq \in \{0, 1, 2\}$
VARIANT
$varSeq$
EVENTS
Initialisation
begin
a1 : $init(var_k)$
a2 : $varSeq := 2$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G_0''(var_k)$
g2 : $varPar_1 = 0$
g3 : $varPar_2 = 0$
then
a : $S_0''(var_k)$
end

Event $Evt_{11} \widehat{=}$
Status convergent
when
g1 : $G_{11}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 2$
then
a1 : $S_{11}(var_k)$
a2 : $varSeq := varSeq - 1$
end
Event $Evt_{12} \widehat{=}$
Status convergent
when
g1 : $G_{12}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 1$
then
a1 : $S_{12}(var_k)$
a2 : $varSeq := varSeq - 1$
end

Event $Evt_1 \widehat{=}$
refines Evt_1
when
g1 : $G_1'(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 0$
then
a1 : $S_1'(var_k)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
refines Evt_2
when
g1 : $G_2'(var_k)$
g2 : $varPar_2 = 1$
then
a1 : $S_2'(var_k)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_2 \widehat{=} (A_{11} ; A_{12}) \| A_2$

variant = $(varPar_1 + varPar_2) + varSeq$

MACHINE M_2
REFINES M_1
VARIABLES
var_k varSeq
INVARIANTS
i1 : $K(var_k, var_j, var_i)$
i2 : $varSeq \in \{0, 1, 2\}$
VARIANT
varSeq
EVENTS
Initialisation
begin
a1 : $init(var_k)$
a2 : $varSeq := 2$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G_0''(var_k)$
g2 : $varPar_1 = 0$
g3 : $varPar_2 = 0$
then
a : $S_0''(var_k)$
end

Event $Evt_{11} \widehat{=}$
Status convergent
when
g1 : $G_{11}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 2$
then
a1 : $S_{11}(var_k)$
a2 : $varSeq := varSeq - 1$
end
Event $Evt_{12} \widehat{=}$
Status convergent
when
g1 : $G_{12}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 1$
then
a1 : $S_{12}(var_k)$
a2 : $varSeq := varSeq - 1$
end

Event $Evt_1 \widehat{=}$
refines Evt_1
when
g1 : $G_1'(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 0$
then
a1 : $S_1'(var_k)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
refines Evt_2
when
g1 : $G_2'(var_k)$
g2 : $varPar_2 = 1$
then
a1 : $S_2'(var_k)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_2 \widehat{=} (A_{11} ; A_{12}) \| A_2$

variant = $(varPar_1 + varPar_2) + varSeq$

MACHINE M_2
REFINES M_1
VARIABLES
var_k varSeq
INVARIANTS
i1 : $K(var_k, var_j, var_i)$
i2 : $varSeq \in \{0, 1, 2\}$
VARIANT
varSeq
EVENTS
Initialisation
begin
a1 : $init(var_k)$
a2 : $varSeq := 2$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G_0''(var_k)$
g2 : $varPar_1 = 0$
g3 : $varPar_2 = 0$
then
a : $S_0''(var_k)$
end

Event $Evt_{11} \widehat{=}$
Status convergent
when
g1 : $G_{11}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 2$
then
a1 : $S_{11}(var_k)$
a2 : $varSeq := varSeq - 1$
end
Event $Evt_{12} \widehat{=}$
Status convergent
when
g1 : $G_{12}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 1$
then
a1 : $S_{12}(var_k)$
a2 : $varSeq := varSeq - 1$
end

Event $Evt_1 \widehat{=}$
refines Evt_1
when
g1 : $G_1'(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 0$
then
a1 : $S_1'(var_k)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
refines Evt_2
when
g1 : $G_2'(var_k)$
g2 : $varPar_2 = 1$
then
a1 : $S_2'(var_k)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_2 \widehat{=} (A_{11} ; A_{12}) \| A_2$

variant = $(varPar_1 + varPar_2) + varSeq$

MACHINE M_2
REFINES M_1
VARIABLES
var_k varSeq
INVARIANTS
i1 : $K(var_k, var_j, var_i)$
i2 : $varSeq \in \{0, 1, 2\}$
VARIANT
varSeq
EVENTS
Initialisation
begin
a1 : $init(var_k)$
a2 : $varSeq := 2$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G_0''(var_k)$
g2 : $varPar_1 = 0$
g3 : $varPar_2 = 0$
then
a : $S_0''(var_k)$
end

Event $Evt_{11} \widehat{=}$
Status convergent
when
g1 : $G_{11}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 2$
then
a1 : $S_{11}(var_k)$
a2 : $varSeq := varSeq - 1$
end
Event $Evt_{12} \widehat{=}$
Status convergent
when
g1 : $G_{12}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 1$
then
a1 : $S_{12}(var_k)$
a2 : $varSeq := varSeq - 1$
end

Event $Evt_1 \widehat{=}$
refines Evt_1
when
g1 : $G_1'(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 0$
then
a1 : $S_1'(var_k)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
refines Evt_2
when
g1 : $G_2'(var_k)$
g2 : $varPar_2 = 1$
then
a1 : $S_2'(var_k)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Machine $M_2 \widehat{=} (A_{11} ; A_{12}) \parallel A_2$

variant = $(varPar_1 + varPar_2) + varSeq$

MACHINE M_2
REFINES M_1
VARIABLES
var_k varSeq
INVARIANTS
i1 : $K(var_k, var_j, var_i)$
i2 : $varSeq \in \{0, 1, 2\}$
VARIANT
varSeq
EVENTS
Initialisation
begin
a1 : $init(var_k)$
a2 : $varSeq := 2$
end
Event $Evt_0 \widehat{=}$
refines Evt_0
when
g1 : $G_0''(var_k)$
g2 : $varPar_1 = 0$
g3 : $varPar_2 = 0$
then
a : $S_0''(var_k)$
end

Event $Evt_{11} \widehat{=}$
Status convergent
when
g1 : $G_{11}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 2$
then
a1 : $S_{11}(var_k)$
a2 : $varSeq := varSeq - 1$
end
Event $Evt_{12} \widehat{=}$
Status convergent
when
g1 : $G_{12}(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 1$
then
a1 : $S_{12}(var_k)$
a2 : $varSeq := varSeq - 1$
end

Event $Evt_1 \widehat{=}$
refines Evt_1
when
g1 : $G_1'(var_k)$
g2 : $varPar_1 = 1$
g3 : $varSeq = 0$
then
a1 : $S_1'(var_k)$
a2 : $varPar_1 := varPar_1 - 1$
end
Event $Evt_2 \widehat{=}$
refines Evt_2
when
g1 : $G_2'(var_k)$
g2 : $varPar_2 = 1$
then
a1 : $S_2'(var_k)$
a2 : $varPar_2 := varPar_2 - 1$
end
END

The proposed approach

Interleaving semantic of Event-B

- Machine M_1
  - $(A_1 ; A_2) \sqsubseteq (A_1 \parallel A_2)$
  - $(A_2 ; A_1) \sqsubseteq (A_1 \parallel A_2)$
- Machine M_2
  - $(A_{11} ; A_{12} ; A_2) \sqsubseteq ((A_{11} ; A_{12}) \parallel A_2)$
  - $(A_{11} ; A_2 ; A_{12}) \sqsubseteq ((A_{11} ; A_{12}) \parallel A_2)$
  - $(A_2 ; A_{11} ; A_{12}) \sqsubseteq ((A_{11} ; A_{12}) \parallel A_2)$

The proposed approach

Interleaving semantic of Event-B

- Machine M_1
  - $(A_1 ; A_2) \sqsubseteq (A_1 \parallel A_2)$
  - $(A_2 ; A_1) \sqsubseteq (A_1 \parallel A_2)$
- Machine M_2
  - $(A_{11} ; A_{12} ; A_2) \sqsubseteq ((A_{11} ; A_{12}) \parallel A_2)$
  - $(A_{11} ; A_2 ; A_{12}) \sqsubseteq ((A_{11} ; A_{12}) \parallel A_2)$
  - $(A_2 ; A_{11} ; A_{12}) \sqsubseteq ((A_{11} ; A_{12}) \parallel A_2)$

Plan

The CO4EB plugin/Conclusion

Update site

idir.aitsadoune.free.fr/tools/updatesite

Video demo

www.youtube.com/watch?v=oQwsu8CQFOw