//Code generated from the grafcet metamodel instance "Eval Efficiency3 Modified"

//Date: 2018-09-10 21:30:53.157

#include "TimerOne.h"

//NB: The default value for every pin is 0. The user is supposed to modify it according to its need

//\*\*\*\* Declare INPUT pins mapped \*\*\*\* Total : 5

const byte pin\_h1 = 0;

const byte pin\_b = 0;

const byte pin\_door\_opened = 0;

const byte pin\_temperature\_in = 0;

const byte pin\_a = 0;

//\*\*\*\* Declare OUTPUT pins mapped \*\*\*\* Total : 10

const byte pin\_tmp = 0;

const byte pin\_C = 0;

const byte pin\_A = 0;

const byte pin\_F = 0;

const byte pin\_D = 0;

const byte pin\_G = 0;

const byte pin\_M = 0;

const byte pin\_N = 0;

const byte pin\_B = 0;

const byte pin\_E = 0;

//\*\*\*\* Declare DIGITAL INPUT pins states \*\*\*\* Total : 4

boolean h1 = false, h1\_Old = false ;

boolean b = false, b\_Old = false ;

boolean door\_opened = false, door\_opened\_Old = false ;

boolean a = false, a\_Old = false ;

//\*\*\*\* Declare states of DIGITAL OUTPUT \*\*\*\* Total : 8

boolean C = false, C\_Old = false ;

boolean A = false, A\_Old = false ;

boolean F = false, F\_Old = false ;

boolean D = false, D\_Old = false ;

boolean G = false, G\_Old = false ;

boolean M = false, M\_Old = false ;

boolean B = false, B\_Old = false ;

boolean E = false, E\_Old = false ;

//\*\*\*\* Declare states of ANALOG INPUT \*\*\*\* Total : 1

int temperature\_in = 0, temperature\_in\_Old = 0 ;

//\*\*\*\* Declare ANALOG OUTPUT pins states \*\*\*\* Total : 2

int tmp = 0, tmp\_Old = 0 ;

int N = 0, N\_Old = 0 ;

//\*\*\*\* Declare BOOLEAN Internal variables (With steps activity) \*\*\*\* Total : 10

boolean X6 = false, X6\_Old = false ;

boolean boo1 = false, boo1\_Old = false ;

boolean X4 = false, X4\_Old = false ;

boolean vv = false, vv\_Old = false ;

boolean X2 = false, X2\_Old = false ;

boolean X5 = false, X5\_Old = false ;

boolean X1 = false, X1\_Old = false ;

boolean boo2 = false, boo2\_Old = false ;

boolean X3 = false, X3\_Old = false ;

boolean X7 = false, X7\_Old = false ;

//\*\*\*\* Declare NUMERICAL Internal variables \*\*\*\* Total : 1

int local\_tmp = 0, local\_tmp\_Old = 0;

//Variables to evaluate firing transitions

//\*\*\*\* Declare Validated Transitions variables \*\*\*\*

boolean VT\_t3 = false;

boolean VT\_t1 = false;

boolean VT\_t4 = false;

boolean VT\_t5 = false;

boolean VT\_t2 = false;

boolean VT\_t7 = false;

boolean VT\_t6 = false;

//\*\*\*\* Declare Receptivities of Transitions variables \*\*\*\*

boolean R\_t3 = false;

boolean R\_t1 = false;

boolean R\_t4 = false;

boolean R\_t5 = false;

boolean R\_t2 = false;

boolean R\_t7 = false;

boolean R\_t6 = false;

//\*\*\*\* Declare Firing Transitions variables \*\*\*\*

boolean FT\_t3 = false;

boolean FT\_t1 = false;

boolean FT\_t4 = false;

boolean FT\_t5 = false;

boolean FT\_t2 = false;

boolean FT\_t7 = false;

boolean FT\_t6 = false;

//\*\*\*\* Declare STEPs timing variables for duration activity \*\*\*\*

unsigned long X1\_duration = 0, X1\_duration\_Old = 0;

unsigned long X2\_duration = 0, X2\_duration\_Old = 0;

unsigned long X3\_duration = 0, X3\_duration\_Old = 0;

unsigned long X4\_duration = 0, X4\_duration\_Old = 0;

unsigned long X5\_duration = 0, X5\_duration\_Old = 0;

unsigned long X6\_duration = 0, X6\_duration\_Old = 0;

unsigned long X7\_duration = 0, X7\_duration\_Old = 0;

//\*\*\*\* Declare timing variables for duration activity of not step activity vars i.e. other Boolean Variables \*\*\*\*

unsigned long vv\_duration = 0, vv\_duration\_Old = 0;

unsigned long F\_duration = 0, F\_duration\_Old = 0;

unsigned long D\_duration = 0, D\_duration\_Old = 0;

unsigned long door\_opened\_duration = 0, door\_opened\_duration\_Old = 0;

unsigned long E\_duration = 0, E\_duration\_Old = 0;

unsigned long a\_duration = 0, a\_duration\_Old = 0;

unsigned long boo1\_duration = 0, boo1\_duration\_Old = 0;

unsigned long C\_duration = 0, C\_duration\_Old = 0;

unsigned long A\_duration = 0, A\_duration\_Old = 0;

unsigned long boo2\_duration = 0, boo2\_duration\_Old = 0;

unsigned long h1\_duration = 0, h1\_duration\_Old = 0;

unsigned long b\_duration = 0, b\_duration\_Old = 0;

unsigned long G\_duration = 0, G\_duration\_Old = 0;

unsigned long M\_duration = 0, M\_duration\_Old = 0;

unsigned long B\_duration = 0, B\_duration\_Old = 0;

//Number of timing Variables in Grafcet Expressions : 5

//[4200 ms/(X6 and (local\_tmp > 0))]: The following variable measures the duration since which <(X6 and (local\_tmp > 0))> has been ON, until its next Rising Edge (RE)

unsigned long X6\_AND\_local\_tmp\_GT\_0\_duration = 0; unsigned long X6\_AND\_local\_tmp\_GT\_0\_duration\_Old = 0;

//[not 25 m/X4]: The following variable measures the duration since which <X4> has been ON, until its next Rising Edge (RE)

//[not 15 s/(X5 or [8 s/(boo1 and (not boo2))])]: The following variable measures the duration since which <(X5 or [8 s/(boo1 and (not boo2))])> has been ON, until its next Rising Edge (RE)

unsigned long X5\_OR\_boo1\_AND\_NOT\_boo2\_D1\_8000\_duration = 0; unsigned long X5\_OR\_boo1\_AND\_NOT\_boo2\_D1\_8000\_duration\_Old = 0;

//[8 s/(boo1 and (not boo2))]: The following variable measures the duration since which <(boo1 and (not boo2))> has been ON, until its next Rising Edge (RE)

unsigned long boo1\_AND\_NOT\_boo2\_duration = 0; unsigned long boo1\_AND\_NOT\_boo2\_duration\_Old = 0;

//[25 s/(X1 or X2)/35s]: The following variable measures the duration since which <(X1 or X2)> has been ON, until its next Rising Edge (RE)

unsigned long X1\_OR\_X2\_duration = 0; unsigned long X1\_OR\_X2\_duration\_Old = 0;

//To manage timer or measuring duration of active steps

const unsigned int TIMER\_PERIOD = 100; //With 100 ms = 1/10 seconds

//Variable to detect if at least one transitions is fired

boolean transitions\_fired;

//Program Initialization

void setup(){

initializeTimer();

//INPUT PINs Configuration

pinConfigMode(pin\_door\_opened, INPUT);

pinConfigMode(pin\_temperature\_in, INPUT);

pinConfigMode(pin\_a, INPUT);

pinConfigMode(pin\_b, INPUT);

pinConfigMode(pin\_h1, INPUT);

//OUTPUT PINs Configuration

pinConfigMode(pin\_N, OUTPUT);

pinConfigMode(pin\_A, OUTPUT);

pinConfigMode(pin\_B, OUTPUT);

pinConfigMode(pin\_C, OUTPUT);

pinConfigMode(pin\_D, OUTPUT);

pinConfigMode(pin\_E, OUTPUT);

pinConfigMode(pin\_F, OUTPUT);

pinConfigMode(pin\_G, OUTPUT);

pinConfigMode(pin\_tmp, OUTPUT);

pinConfigMode(pin\_M, OUTPUT);

//Inital steps activity variables initialization

X1\_Old = true; //X1 = true; //But not necessary

X4\_Old = true; //X4 = true; //But not necessary

}

//Program loop

void loop(){

//Reading states of Digital INPUT pins (Digital Input variables)

door\_opened = digitalPinRead(pin\_door\_opened);

a = digitalPinRead(pin\_a);

b = digitalPinRead(pin\_b);

h1 = digitalPinRead(pin\_h1);

//Reading states of ANALOG/Numeric INPUT pins (Analog Input variables)

temperature\_in = analogPinRead(pin\_temperature\_in);

transitions\_fired = false;

//Evaluate validated transitions (variables)

VT\_t1 = X6 ;

VT\_t2 = X1 ;

VT\_t3 = X2 ;

VT\_t4 = X3 ;

VT\_t5 = X4 ;

VT\_t6 = X5 ;

VT\_t7 = X7 ;

//Evaluate Receptivities of transitions

R\_t1 = (VT\_t1)? (X6\_AND\_local\_tmp\_GT\_0\_duration >= 4200/TIMER\_PERIOD): false ;

R\_t2 = (VT\_t2)? (X5 || (door\_opened\_Old == true && door\_opened == false)): false ;

R\_t3 = (VT\_t3)? X4: false ;

R\_t4 = (VT\_t4)? (X5 || ((boo1\_Old && boo2\_Old) == false && (boo1 && boo2) == true)): false ;

R\_t5 = (VT\_t5)? (X4\_duration <= 1500000/TIMER\_PERIOD): false ;

R\_t6 = (VT\_t6)? ((X5\_OR\_boo1\_AND\_NOT\_boo2\_D1\_8000\_duration <= 15000/TIMER\_PERIOD) && ((temperature\_in\_Old > (29 + (N\_Old \* 2))) == false && (temperature\_in > (29 + (N \* 2))) == true)): false ;

R\_t7 = (VT\_t7)? ((X1\_OR\_X2\_duration >= 25000/TIMER\_PERIOD) && (X1 || X2)) || ((X1\_OR\_X2\_duration <= 35000/TIMER\_PERIOD) && !(X1 || X2)): false ;

//Evaluate Clearing/firing transitions conditions

FT\_t1 = VT\_t1 && R\_t1;

FT\_t2 = VT\_t2 && R\_t2;

FT\_t3 = VT\_t3 && R\_t3;

FT\_t4 = VT\_t4 && R\_t4;

FT\_t5 = VT\_t5 && R\_t5;

FT\_t6 = VT\_t6 && R\_t6;

FT\_t7 = VT\_t7 && R\_t7;

//Calculation if there is any transition fired : 2nd alternative

transitions\_fired = FT\_t1 || FT\_t2 || FT\_t3 || FT\_t4 || FT\_t5 || FT\_t6 || FT\_t7 ;

//Evaluate steps activity variables

X1 = FT\_t1 || (X1\_Old && ! R\_t1);

X2 = FT\_t2 || (X2\_Old && ! R\_t2);

X3 = FT\_t3 || (X3\_Old && ! R\_t3);

X4 = FT\_t6 || (X4\_Old && ! R\_t6);

X5 = FT\_t5 || (X5\_Old && ! R\_t5);

X6 = FT\_t4 || (X6\_Old && ! R\_t4);

X7 = FT\_t7 || (X7\_Old && ! R\_t7);

//Reinitialize all the Digital Output variables

A = false;

B = false;

C = false;

D = false;

E = false;

F = false;

G = false;

M = false;

//Evaluate Digital OUTPUTs variables : 8

if(transitions\_fired == false){

//Evaluate Level Actions Associated to Step 1 : 1

if(X1){ if (((temperature\_in > 15) || ((boo1\_Old || boo2\_Old) == true && (boo1 || boo2) == false))) {A = true;}}

//Evaluate Level Actions Associated to Step 2 : 1

if(X2){ if (1) {B = true;}}

//Evaluate Level Actions Associated to Step 3 : 1

if(X3){ if (1) {C = true;}}

//Evaluate Level Actions Associated to Step 4 : 2

if(X4){ if (1) {D = true;}}

if(X4){ if (1) {E = true;}}

//Evaluate Level Actions Associated to Step 5 : 2

if(X5){ if (1) {F = true;}}

if(X5){ if (1) {G = true;}}

//Evaluate Level Actions Associated to Step 6 : 1

if(X6){ if (((boo1\_Old && boo2\_Old) == false && (boo1 && boo2) == true)) {M = true;}}

}

//Evaluate Analog/Stored OUTPUTs variables

//Evaluate Stored Actions Associated to Step 1

//Step 1: Action N On Activation

if(X1\_Old == false && X1 == true){

N = 0;

}

//Step 1: Action local\_tmp On Activation

if(X1\_Old == false && X1 == true){

local\_tmp = ((local\_tmp \* 10) - 25);

}

//Step 1: Action boo1 On Activation

if(X1\_Old == false && X1 == true){

boo1 = 1;

}

//Step 1: Action boo2 On Desactivation

if(X1\_Old == true && X1 == false){

boo2 = 0;

}

//Step 1: Action vv On Desactivation

if(X1\_Old == true && X1 == false){

vv = ((X1\_Old == false && X1 == true) || (! ((local\_tmp + 8) < 25)));

}

//Evaluate Stored Actions Associated to Step 5

//Step 5: Action N On Activation

if(X5\_Old == false && X5 == true){

N = (N + 1);

}

//Step 5: Action tmp On Activation

if(X5\_Old == false && X5 == true){

tmp = (tmp + 1);

}

//Evaluate Stored Actions Associated to Step 6

//Step 6: Action activerVanne On Activation

if(X6\_Old == false && X6 == true){

readVanneState(); //function for stored action

}

//Step 6: Action N On Activation

if(X6\_Old == false && X6 == true){

N = (N + 1);

}

//Updating LEVEL ACTIONS OR DIGITAL OUTPUTs

if(transitions\_fired == false){

//A stable situation is reached

if(A\_Old != A){

digitalPinWrite(pin\_A, A);

}

if(B\_Old != B){

digitalPinWrite(pin\_B, B);

}

if(C\_Old != C){

digitalPinWrite(pin\_C, C);

}

if(D\_Old != D){

digitalPinWrite(pin\_D, D);

}

if(E\_Old != E){

digitalPinWrite(pin\_E, E);

}

if(F\_Old != F){

digitalPinWrite(pin\_F, F);

}

if(G\_Old != G){

digitalPinWrite(pin\_G, G);

}

if(M\_Old != M){

digitalPinWrite(pin\_M, M);

}

}

//Updating Analog/Stored Actions/OUTPUTs

analogPinWrite(pin\_N, N);

analogPinWrite(pin\_tmp, tmp);

analogPinWrite(pin\_M, M);

// SAVE DIGITAL INPUT pins states/variables

h1\_Old = h1 ;

b\_Old = b ;

door\_opened\_Old = door\_opened ;

a\_Old = a ;

// SAVE DIGITAL OUTPUT pins states/variables

C\_Old = C ; A\_Old = A ; F\_Old = F ; D\_Old = D ; G\_Old = G ; M\_Old = M ;

B\_Old = B ; E\_Old = E ;

// SAVE ANALOG INPUT pins states/variables

temperature\_in\_Old = temperature\_in ;

// SAVE ANALOG OUTPUT pins states/variables

tmp\_Old = tmp ;

N\_Old = N ;

// UPDATE Old BOOLEAN Internal variables (With steps activity)

boo1\_Old = boo1; X6\_Old = X6; X4\_Old = X4; vv\_Old = vv;

X2\_Old = X2; X5\_Old = X5; X1\_Old = X1; boo2\_Old = boo2;

X3\_Old = X3; X7\_Old = X7;

// UPDATE Old NUMERICAL Internal variables

local\_tmp\_Old = local\_tmp;

}

int main(void)

{

setup();

for ( ; ; ) loop(); // repeat indefinitely the function loop()

return 0;

}

void initializeTimer(){

unsigned int FT\_Steps = 1000/TIMER\_PERIOD; //FT\_Steps = frequency\_timer

Timer1.initialize(1000000/FT\_Steps);

Timer1.attachInterrupt(update\_timingVars\_callback);

}

void update\_timingVars\_callback(){

//function called after every period of the timer to update timing variables

//Updating durations of steps activity variables

//for the step 1

if(X1\_Old == true && X1 == false){

X1\_duration = 0;

}else if(X1 == true){

X1\_duration ++;

}

//for the step 2

if(X2\_Old == true && X2 == false){

X2\_duration = 0;

}else if(X2 == true){

X2\_duration ++;

}

//for the step 3

if(X3\_Old == true && X3 == false){

X3\_duration = 0;

}else if(X3 == true){

X3\_duration ++;

}

//for the step 4

if(X4\_Old == true && X4 == false){

X4\_duration = 0;

}else if(X4 == true){

X4\_duration ++;

}

//for the step 5

if(X5\_Old == true && X5 == false){

X5\_duration = 0;

}else if(X5 == true){

X5\_duration ++;

}

//for the step 6

if(X6\_Old == true && X6 == false){

X6\_duration = 0;

}else if(X6 == true){

X6\_duration ++;

}

//for the step 7

if(X7\_Old == true && X7 == false){

X7\_duration = 0;

}else if(X7 == true){

X7\_duration ++;

}

//Updating activity duration of other Boolean Variables of the Grafcet. Size = 15

if(vv\_Old == true && vv == false){

vv\_duration = 0;

}else if(vv == true){

vv\_duration ++;

}

if(F\_Old == true && F == false){

F\_duration = 0;

}else if(F == true){

F\_duration ++;

}

if(D\_Old == true && D == false){

D\_duration = 0;

}else if(D == true){

D\_duration ++;

}

if(door\_opened\_Old == true && door\_opened == false){

door\_opened\_duration = 0;

}else if(door\_opened == true){

door\_opened\_duration ++;

}

if(E\_Old == true && E == false){

E\_duration = 0;

}else if(E == true){

E\_duration ++;

}

if(a\_Old == true && a == false){

a\_duration = 0;

}else if(a == true){

a\_duration ++;

}

if(boo1\_Old == true && boo1 == false){

boo1\_duration = 0;

}else if(boo1 == true){

boo1\_duration ++;

}

if(C\_Old == true && C == false){

C\_duration = 0;

}else if(C == true){

C\_duration ++;

}

if(A\_Old == true && A == false){

A\_duration = 0;

}else if(A == true){

A\_duration ++;

}

if(boo2\_Old == true && boo2 == false){

boo2\_duration = 0;

}else if(boo2 == true){

boo2\_duration ++;

}

if(h1\_Old == true && h1 == false){

h1\_duration = 0;

}else if(h1 == true){

h1\_duration ++;

}

if(b\_Old == true && b == false){

b\_duration = 0;

}else if(b == true){

b\_duration ++;

}

if(G\_Old == true && G == false){

G\_duration = 0;

}else if(G == true){

G\_duration ++;

}

if(M\_Old == true && M == false){

M\_duration = 0;

}else if(M == true){

M\_duration ++;

}

if(B\_Old == true && B == false){

B\_duration = 0;

}else if(B == true){

B\_duration ++;

}

//Updating activity duration of other Boolean Variables of Grafcet Timing Expressions. Size = 5

boolean b\_old, b\_new;

//For [4200 ms/(X6 and (local\_tmp > 0))]

//if FE [4200 ms/(X6 and (local\_tmp > 0))]

if((X6\_Old && (local\_tmp\_Old > 0)) == true && ! (X6 && (local\_tmp > 0)) == false){

X6\_AND\_local\_tmp\_GT\_0\_D1\_4200\_duration = 0;

}

if((X6 && (local\_tmp > 0)) == true){

X6\_AND\_local\_tmp\_GT\_0\_D1\_4200\_duration ++;

}

//For [not 25 m/X4]

//For [not 15 s/(X5 or [8 s/(boo1 and (not boo2))])]

//if FE [not 15 s/(X5 or [8 s/(boo1 and (not boo2))])]

if((X5\_Old || (boo1\_AND\_NOT\_boo2\_duration\_Old >= 8000/TIMER\_PERIOD)) == true && ! (X5 || (boo1\_AND\_NOT\_boo2\_duration >= 8000/TIMER\_PERIOD)) == false){

X5\_OR\_boo1\_AND\_NOT\_boo2\_D1\_8000\_L\_15000\_duration = 0;

}

if((X5 || (boo1\_AND\_NOT\_boo2\_duration >= 8000/TIMER\_PERIOD)) == true){

X5\_OR\_boo1\_AND\_NOT\_boo2\_D1\_8000\_L\_15000\_duration ++;

}

//For [8 s/(boo1 and (not boo2))]

//if FE [8 s/(boo1 and (not boo2))]

if((boo1\_Old && (! boo2\_Old)) == true && ! (boo1 && (! boo2)) == false){

boo1\_AND\_NOT\_boo2\_D1\_8000\_duration = 0;

}

if((boo1 && (! boo2)) == true){

boo1\_AND\_NOT\_boo2\_D1\_8000\_duration ++;

}

//For [25 s/(X1 or X2)/35s]

b\_old = (X1\_Old || X2\_Old);

b\_new = (X1 || X2);

if((b\_old == false && b\_new == true) or (b\_old == true && b\_new == false)){

X1\_OR\_X2\_D2\_25000\_35000\_duration = 0;

}

else{

X1\_OR\_X2\_D2\_25000\_35000\_duration ++;

}

//Updating old timing Variables of any Grafcet Boolean variable : 22

X1\_duration\_Old = X1\_duration;

boo1\_duration\_Old = boo1\_duration;

boo2\_duration\_Old = boo2\_duration;

X2\_duration\_Old = X2\_duration;

door\_opened\_duration\_Old = door\_opened\_duration;

X3\_duration\_Old = X3\_duration;

X4\_duration\_Old = X4\_duration;

X5\_duration\_Old = X5\_duration;

X6\_duration\_Old = X6\_duration;

A\_duration\_Old = A\_duration;

B\_duration\_Old = B\_duration;

C\_duration\_Old = C\_duration;

D\_duration\_Old = D\_duration;

vv\_duration\_Old = vv\_duration;

E\_duration\_Old = E\_duration;

F\_duration\_Old = F\_duration;

G\_duration\_Old = G\_duration;

M\_duration\_Old = M\_duration;

a\_duration\_Old = a\_duration;

b\_duration\_Old = b\_duration;

h1\_duration\_Old = h1\_duration;

X7\_duration\_Old = X7\_duration;

//Updating old timing Variables of the Grafcet Expressions : 5

//fOR [4200 ms/(X6 and (local\_tmp > 0))]

X6\_AND\_local\_tmp\_GT\_0\_duration\_Old = X6\_AND\_local\_tmp\_GT\_0\_duration;

//fOR [not 25 m/X4]

//fOR [not 15 s/(X5 or [8 s/(boo1 and (not boo2))])]

X5\_OR\_boo1\_AND\_NOT\_boo2\_D1\_8000\_duration\_Old = X5\_OR\_boo1\_AND\_NOT\_boo2\_D1\_8000\_duration;

//fOR [8 s/(boo1 and (not boo2))]

boo1\_AND\_NOT\_boo2\_duration\_Old = boo1\_AND\_NOT\_boo2\_duration;

//fOR [25 s/(X1 or X2)/35s]

X1\_OR\_X2\_duration\_Old = X1\_OR\_X2\_duration;

}

//Expression of functions associated to stored actions

//Functions associated to stored actions of Step 6

void readVanneState(){

int state = digitalRead(0);

}

void pinConfigMode(int pin\_num, int mode){

pinMode(pin\_num, mode);

}

int digitalPinRead(int pin\_num){

return digitalRead(pin\_num);

}

long analogPinRead(int pin\_num){

return analogRead(pin\_num);

}

void digitalPinWrite(int pin\_num, int value){

digitalWrite(pin\_num, value);

}

void analogPinWrite(int pin\_num, int value){

analogWrite(pin\_num, value);

}