

```
//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);

    //Initial 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);
}
}

```