```
CONTEXT
   System_Ctx
CONSTANTS
   S
   TIME
   sigma
AXIOMS
   axm1   :   S=RReal×RReal
   axm2   :   TIME=RRealPlus
   axm3   :   sigma∈ RRealPlus ∧ sigma ↦Rzero ∈gt
END
```

CONTEXT

Thoerems

AXIOMS

axm1 : ∀a,b,c,d·a↦b∈ leq ∧ c↦d ∈leq ⇒ plus(a↦c) ↦plus(b↦d) ∈ leq

axm2 : ∀a,b,c,d·Rzero↦a ∈ leq ∧ Rzero↦b ∈ leq ∧ Rzero↦c ∈ leq ∧ Rzero↦d ∈ leq ∧ a↦b∈ leq ∧ c↦d ∈leq ⇒ times (a↦c) ↦times(b↦d) ∈ leq

axm3 : ∀a,b,c· a↦b ∈ leq ∧ b↦c ∈ leq⇒ a ↦c ∈ leq

axm4 : ∀a,b· a∈ RReal ∧ b ∈ RReal ⇒ minus(times(a↦a) ↦times(b↦b))=times(plus(a↦b)↦minus(a↦b))

axm5 : ∀a· a∈ RReal ⇒ uminus(a)=minus(Rzero↦a)

axm6 : ∀a· a∈ RReal ⇒ a=plus(times(divide(Rone ↦ Rtwo) ↦a) ↦ times(divide(Rone ↦ Rtwo) ↦a))

axm7 : ∀a,b· a∈ RReal ∧ b∈ RReal ∧ times(a↦b)∈ RRealStar ⇒ inverse(times(a↦b))=times(inverse(a)↦inverse(b))

END

```
MACHINE
    System_M
SEES
    System_Ctx
    Thoerems
VARIABLES
    t
    plantV
INVARIANTS
    inv1   :   t ∈ TIME
    inv2   :   plantV ∈ Closed2Closed(Rzero, t) ⇸ S
EVENTS
    INITIALISATION   ≙
    STATUS
      ordinary
    BEGIN
      act1   :   t≔Rzero
      act2   :   plantV :∈ {Rzero} ⟶ S
    END

    Progress   ≙
    STATUS
      ordinary
    BEGIN
      act1   :   t :| t' ∈ TIME ∧ (t ↦ t' ∈ lt ∧ minus(t'↦t) ↦ sigma ∈ geq)
    END

    Plant   ≙
    STATUS
      ordinary
    ANY
      e
      plant1
    WHERE
      grd1   :   e ∈ DE(S)
      grd2   :   Solvable(Closed2Closed(Rzero, t)\dom(plantV),e)
      grd3   :   plant1 ∈ Closed2Closed(Rzero, t)\dom(plantV) ⟶ S ∧
                 AppendSolutionBAP(e,
                 Closed2Closed(Rzero, t)\dom(plantV),
                 Closed2Closed(Rzero, t)\dom(plantV), plant1)
    THEN
      act1   :   plantV≔plantV⩤plant1
    END

END
```

CONTEXT
EventTriggered_Ctx
EXTENDS
System_Ctx
SETS
EXEC
CONSTANTS
safe
evt_trig
ctrl
plant
prg
f_evol
f_evol_plantV
evade_value
AXIOMS
axm1 : safe ∈ (S × RReal) ⟶ BOOL
axm2 : evt_trig ∈ (S × RReal)×RReal ⟶ BOOL
axm3 : partition(EXEC, {ctrl},{plant},{prg})
axm4 : f_evol ∈ RReal ⟶ S
axm5 : f_evol_plantV ∈ (RReal ⟶ (TIME × S ⟶ (RReal×RReal)))
axm6 : ∀ ctrlV · ctrlV ∈ RReal ⇒ (f_evol_plantV(ctrlV) =
(λ t ↦ plantV · t ∈ TIME ∧ plantV ∈ S | f_evol(ctrlV)))
axm7 : evade_value⊆RReal ∧ evade_value≠∅
END

```
MACHINE
    EventTriggered_M
REFINES
    System_M
SEES
    EventTriggered_Ctx
VARIABLES
    t
    plantV
    ctrlV
    exec
INVARIANTS
    inv1   :   ctrlV ∈ RReal
    inv2   :   exec ∈ EXEC
    inv3   :   exec≠plant ⟹ dom(plantV)=Closed2Closed(Rzero, t)
    inv4   :   exec=plant ⟹ t∉ dom(plantV)
EVENTS
    INITIALISATION   ≙
      extended
    STATUS
      ordinary
    BEGIN
      act1   :   t≔Rzero
      act2   :   plantV :∈ {Rzero} ⟶ S
      act3   :   ctrlV :∈ RReal
      act4   :   exec ≔ ctrl
    END

    Progress   ≙
    STATUS
      ordinary
    REFINES
      Progress
    ANY
      t1
    WHERE
      grd1   :   exec=prg
      grd2   :   t1 ∈ TIME ∧ (t ↦ t1 ∈ lt ∧ minus(t1↦t) ↦ sigma ∈ geq)
      grd3   :   ctrlV∉ evade_value⟹evt_trig(plantV(t)↦minus(t1↦t)↦ctrlV) = TRUE
    THEN
      act1   :   t≔t1
      act2   :   exec ≔ plant
    END

    Plant   ≙
    STATUS
      ordinary
    REFINES
      Plant
    ANY
      plant1
    WHERE
      grd1   :   exec=plant
      grd2   :   plant1 ∈ Closed2Closed(Rzero, t)\dom(plantV) ⟶ S
      grd3   :   ode(f_evol_plantV(ctrlV),plant1(t),t) ∈ DE(S)
      grd4   :   Solvable(Closed2Closed(Rzero, t)\dom(plantV),
                          ode(f_evol_plantV(ctrlV),plant1(t),t))
      grd5   :   AppendSolutionBAP(ode(f_evol_plantV(ctrlV),plant1(t),t),
                 Closed2Closed(Rzero, t)\dom(plantV),
                 Closed2Closed(Rzero, t)\dom(plantV), plant1)
    WITH
      e   :   e = ode(f_evol_plantV(ctrlV),plant1(t),t)
    THEN
      act1   :   plantV≔plantV⩤plant1
```

```
    act2   :   exec≔ctrl
  END

  Ctrl_normal   ≜
  STATUS
    ordinary
  ANY
    nrml_value
  WHERE
    grd1   :   exec = ctrl
    grd2   :   nrml_value∈RReal
    grd3   :   nrml_value∉ evade_value ⟹safe(plantV(t)↦nrml_value) = TRUE
  THEN
    act1   :   ctrlV ≔nrml_value
    act2   :   exec ≔ prg
  END

  Ctrl_evade   ≜
  STATUS
    ordinary
  ANY
    evade_val
  WHERE
    grd1   :   exec = ctrl
    grd2   :   evade_val∈evade_value
  THEN
    act1   :   ctrlV≔ evade_val
    act2   :   exec ≔ prg
  END

END
```

```
CONTEXT
    TimeTriggered_Ctx
EXTENDS
    EventTriggered_Ctx
CONSTANTS
    epsilon
    safeEpsilon
AXIOMS
    axm1   :   epsilon ∈ TIME ∧ Rzero↦epsilon ∈leq ∧ sigma↦epsilon ∈leq
    axm2   :   safeEpsilon ∈ (S × RReal) ⟶ BOOL
END
```

```
MACHINE
    TimeTriggered_M
REFINES
    EventTriggered_M
SEES
    TimeTriggered_Ctx
VARIABLES
    t
    plantV
    ctrlV
    exec
EVENTS
    INITIALISATION   ≙
      extended
    STATUS
      ordinary
    BEGIN
      act1   :   t≔Rzero
      act2   :   plantV :∈ {Rzero} ⟶ S
      act3   :   ctrlV :∈ RReal
      act4   :   exec ≔ ctrl
    END

    Progress_time   ≙
    STATUS
      ordinary
    REFINES
      Progress
    ANY
      t1
    WHERE
      grd1   :   exec=prg
      grd2   :   t1 ∈ TIME ∧ t ↦ t1 ∈ lt ∧ minus(t1↦t) ↦ sigma ∈ geq ∧
                 minus(t1↦t) ↦ epsilon ∈ leq
      grd3   :   ctrlV∉ evade_value⟹evt_trig(plantV(t)↦minus(t1↦t)↦ctrlV) = TRUE
    THEN
      act1   :   t≔t1
      act2   :   exec ≔ plant
    END

    Plant   ≙
      extended
    STATUS
      ordinary
    REFINES
      Plant
    ANY
      plant1
    WHERE
      grd1   :   exec=plant
      grd2   :   plant1 ∈ Closed2Closed(Rzero, t)\dom(plantV) ⟶ S
      grd3   :   ode(f_evol_plantV(ctrlV),plant1(t),t) ∈ DE(S)
      grd4   :   Solvable(Closed2Closed(Rzero, t)\dom(plantV),
                          ode(f_evol_plantV(ctrlV),plant1(t),t))
      grd5   :   AppendSolutionBAP(ode(f_evol_plantV(ctrlV),plant1(t),t),
                 Closed2Closed(Rzero, t)\dom(plantV),
                 Closed2Closed(Rzero, t)\dom(plantV), plant1)
    THEN
      act1   :   plantV≔plantV⩤plant1
      act2   :   exec≔ctrl
    END

    Ctrl_normal_time   ≙
      extended
```

```
  STATUS
    ordinary
  REFINES
    Ctrl_normal
  ANY
    nrml_value
  WHERE
    grd1   :   exec = ctrl
    grd2   :   nrml_value∈RReal
    grd3   :   nrml_value∉ evade_value ⟹safe(plantV(t)↦nrml_value) = TRUE
    grd4   :   nrml_value∉ evade_value ⟹safeEpsilon(plantV(t) ↦ nrml_value) = TRUE
  THEN
    act1   :   ctrlV ≔nrml_value
    act2   :   exec ≔ prg
  END

  Ctrl_evade   ≙
    extended
  STATUS
    ordinary
  REFINES
    Ctrl_evade
  ANY
    evade_val
  WHERE
    grd1   :   exec = ctrl
    grd2   :   evade_val∈evade_value
  THEN
    act1   :   ctrlV≔ evade_val
    act2   :   exec ≔ prg
  END

END
```