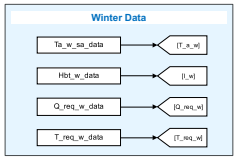
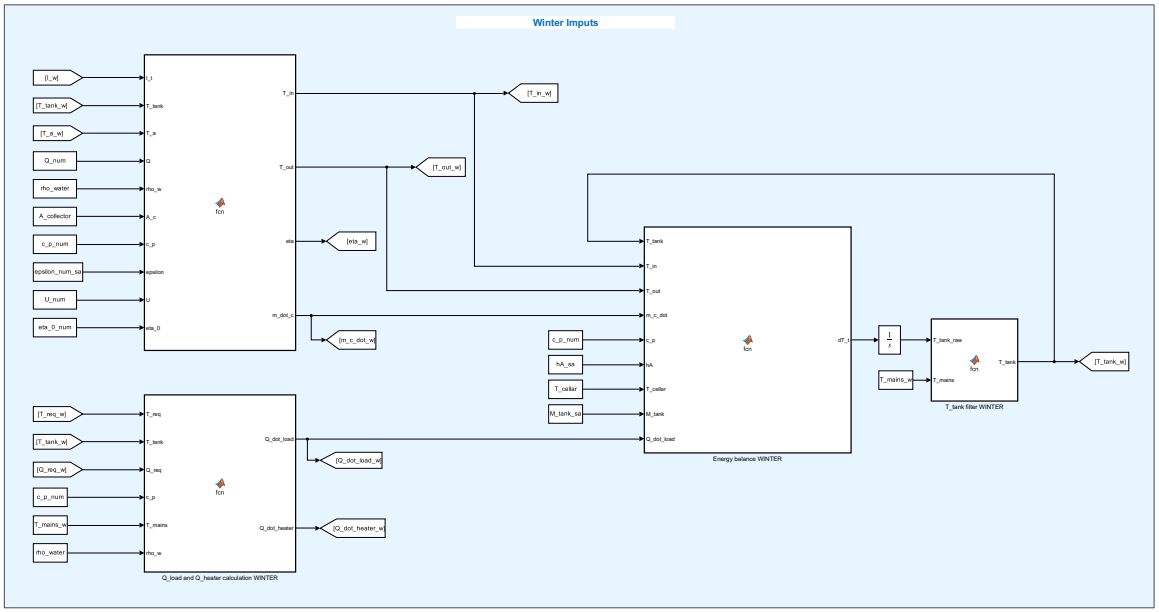
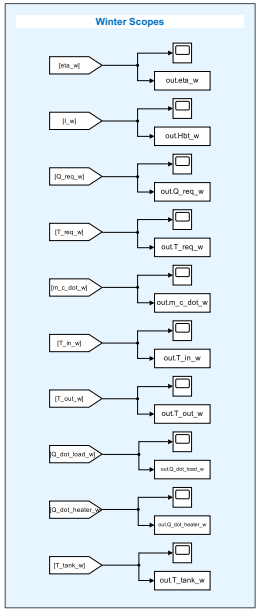
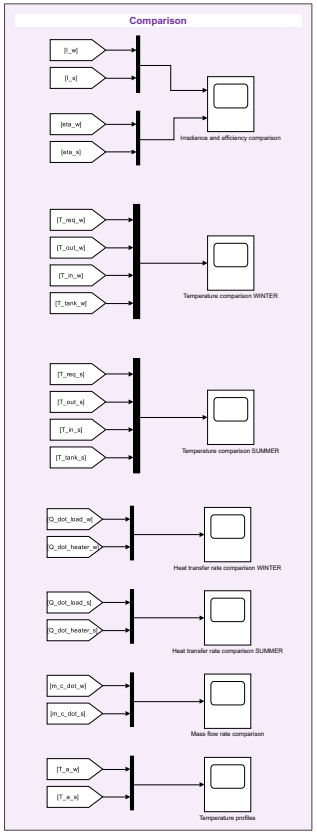
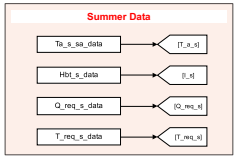
Sensitivity Analisys

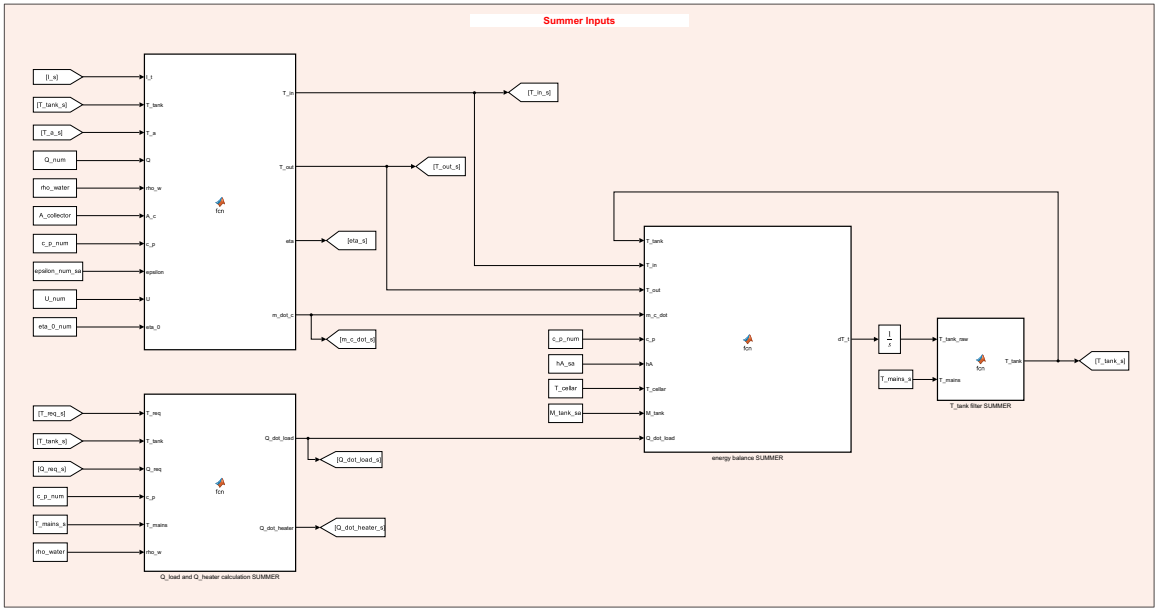


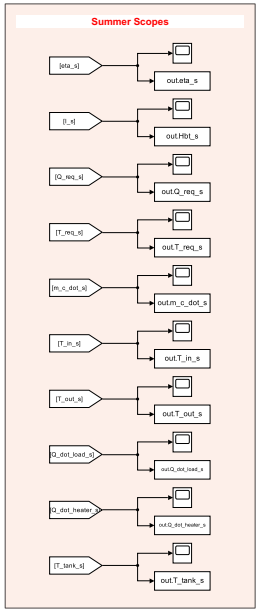












```
function dT_t = fcn(T_tank,T_in,T_out,m_c_dot,c_p,hA,T_cellar,M_tank,Q_dot_load)
dT_t = (m_c_dot*c_p*(T_out-T_in)-Q_dot_load-hA*(T_tank-T_cellar))/(M_tank*c_p);
```

```
function [T_in,T_out,eta,m_dot_c] = fcn(I_t,T_tank,T_a,Q,rho_w,A_c,c_p,epsilon,U,eta_0)

if I_t > 0
    T_in = (A_c*T_a*U+A_c*I_t*eta_0-A_c*T_a*U*epsilon-A_c*I_t*epsilon*eta_0+Q*T_tank*c_p*epsilon*rho_w)/(A_c*U-A_c*U*epsilon+Q*c_p
    T_out = (A_c*T_a*U+A_c*I_t*eta_0-A_c*T_tank*U*epsilon+Q*T_tank*c_p*epsilon*rho_w)/(A_c*U-A_c*U*epsilon+Q*c_p*epsilon*rho_w);
    eta = (Q*c_p*epsilon*rho_w*(T_a*U-T_tank*U+I_t*eta_0))/(I_t*(A_c*U-A_c*U*epsilon+Q*c_p*epsilon*rho_w));
    m_dot_c = rho_w*Q;

    if eta <= 0
        m_dot_c = 0;
        T_in = T_tank;
        T_out = T_tank;
        eta = 0;
    end
else
    m_dot_c = 0;
    T_in = T_tank;
    T_out = T_tank;
    eta = 0;
end

if eta >= 1
    eta = 1;
end
if T_in < T_tank
    T_in = T_tank;
end
if T_out < T_tank
    T_out = T_tank;
end
if T_in > T_out
    T_in = T_tank;
    T_out = T_tank;
end
if T_in >= 100
    T_in = 100;
end
if T_out >= 100
    T_out = 100;
end
```

```
function [T_in,T_out,eta,m_dot_c] = fcn(I_t,T_tank,T_a,Q,rho_w,A_c,c_p,epsilon,U,eta_0)

if I_t > 0
    T_in = (A_c*T_a*U+A_c*I_t*eta_0-A_c*T_a*U*epsilon-A_c*I_t*epsilon*eta_0+Q*T_tank*c_p*epsilon*rho_w)/(A_c*U-A_c*U*epsilon+Q*c_p
    T_out = (A_c*T_a*U+A_c*I_t*eta_0-A_c*T_tank*U*epsilon+Q*T_tank*c_p*epsilon*rho_w)/(A_c*U-A_c*U*epsilon+Q*c_p*epsilon*rho_w);
    eta = (Q*c_p*epsilon*rho_w*(T_a*U-T_tank*U+I_t*eta_0))/(I_t*(A_c*U-A_c*U*epsilon+Q*c_p*epsilon*rho_w));
    m_dot_c = rho_w*Q;

    if eta <= 0
        m_dot_c = 0;
        T_in = T_tank;
        T_out = T_tank;
        eta = 0;
    end
else
    m_dot_c = 0;
    T_in = T_tank;
    T_out = T_tank;
    eta = 0;
end

if eta >= 1
    eta = 1;
end
if T_in < T_tank
    T_in = T_tank;
end
if T_out < T_tank
    T_out = T_tank;
end
if T_in > T_out
    T_in = T_tank;
    T_out = T_tank;
end
if T_in >= 100
    T_in = 100;
end
if T_out >= 100
    T_out = 100;
end
```

```
function [Q_dot_load,Q_dot_heater] = fcn(T_req,T_tank,Q_req,c_p,T_mains,rho_w)

m_req_dot = Q_req*rho_w;

if T_req <= T_tank
    Q_dot_load = m_req_dot*c_p*(T_req-T_mains);
    Q_dot_heater = 0;
else
    Q_dot_load = m_req_dot*c_p*(T_tank-T_mains);
    Q_dot_heater = m_req_dot*c_p*(T_req-T_tank);
end

if Q_dot_load < 0 || Q_dot_heater < 0
    Q_dot_load = 0;
    Q_dot_heater = 0;
end
```

```
function [Q_dot_load,Q_dot_heater] = fcn(T_req,T_tank,Q_req,c_p,T_mains,rho_w)

m_req_dot = Q_req*rho_w;

if T_req <= T_tank
    Q_dot_load = m_req_dot*c_p*(T_req-T_mains);
    Q_dot_heater = 0;
else
    Q_dot_load = m_req_dot*c_p*(T_tank-T_mains);
    Q_dot_heater = m_req_dot*c_p*(T_req-T_tank);
end

if Q_dot_load < 0 || Q_dot_heater < 0
    Q_dot_load = 0;
    Q_dot_heater = 0;
end
```

```
function T_tank = fcn(T_tank_raw,T_mains)

if T_tank_raw < T_mains
    T_tank = T_mains;
else
    T_tank = T_tank_raw;
end

if T_tank_raw < 0
    T_tank = 0;
end

if T_tank >= 100
    T_tank = 100;
end
```

```
function T_tank = fcn(T_tank_raw,T_mains)

if T_tank_raw < T_mains
    T_tank = T_mains;
else
    T_tank = T_tank_raw;
end

if T_tank < 0
    T_tank = 0;
end

if T_tank >= 100
    T_tank = 100;
end
```

```
function dT_t = fcn(T_tank,T_in,T_out,m_c_dot,c_p,hA,T_cellar,M_tank,Q_dot_load)
dT_t = (m_c_dot*c_p*(T_out-T_in)-Q_dot_load-hA*(T_tank-T_cellar))/(M_tank*c_p);
```