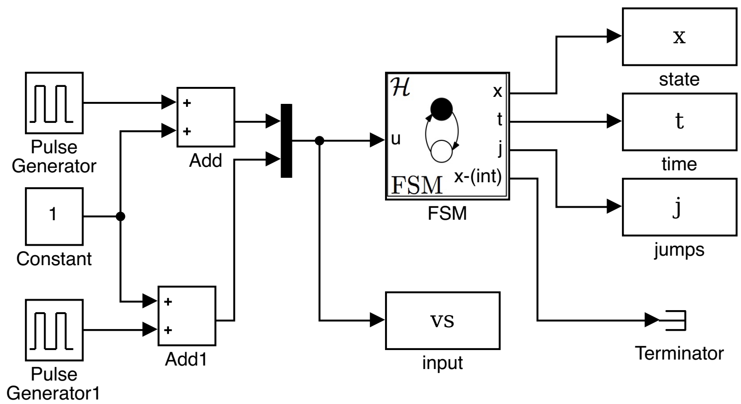

Figure 1: Simulation of a Finite State Machine (FSM)

5 Examples

The examples below illustrate the use of the Simulink implementation above.

Example 5.8 (Implementation of a FSM)

Transitions of the state of a FSM are triggered by changes of its input v. The system can be modeled as a cascade of two systems, in which an external signal drives the FSM. A finite state machine (FSM) or deterministic finite automaton (DFA) is a system with inputs, states, and outputs taking values from discrete sets that are updated at discrete transitions (or jumps) triggered by its inputs. Then, given a FSM and an initial state $q_0 \in Q$, a transition to a state $q_1 = \delta(q_0, v)$ is performed when an input $v \in \Sigma$ is applied to it. After the transition, the output of the FSM is updated to $\kappa(q_1)$. This mechanism can be captured by the difference equation

$$q^+ \;=\; \delta(q,v) \qquad \zeta \;=\; \kappa(q) \qquad (q,v) \in Q \times \Sigma \tag{1}$$

This model captures the dynamics the model of the cyber components in the instructions document with

$$x_c = q, \qquad \Upsilon = Q, \qquad \mathcal{V} = \Sigma, \qquad G_c = \delta, \qquad D_c = \Upsilon \times \mathcal{V}$$

Note that there is no notion of time associated with the FSM model above. An example of this model is presented in Example 5.8. Assuming that the input feeds the input u of the FSM, the state of the FSM is updated according to the transition function evaluated at the current input, that is, q is updated according to

$$q^+ = \delta(q,u)$$

The above model can be summarized as follows:

$$f(q,u) := \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \qquad C := \left\{ (q,u) \in \mathbb{R}^2 \times \{1,2\} \;\middle|\; \delta(q,u) = q \right\}, \tag{2}$$

$$g(q,u) := \delta(q,u) = \begin{bmatrix} 3-u_1 \\ 3-u_2 \end{bmatrix}, \qquad D := \left\{ (q,u) \in \mathbb{R}^2 \times \{1,2\} \;\middle|\; \delta(q,u) \neq q \right\}, \tag{3}$$

$$y := h(q) = q \tag{4}$$

where the input and the state are given by $u = (u_1, u_2) \in \{1,2\}$, and $q = (q_1, q_2) \in \mathbb{R}^2$, respectively.

For the hybrid system FSM in Figure 1 we have the following Matlab embedded functions that describe the sets C and D and functions f and g. Figure 2 depicts the corresponding inputs and state of the FSM.

Flow map

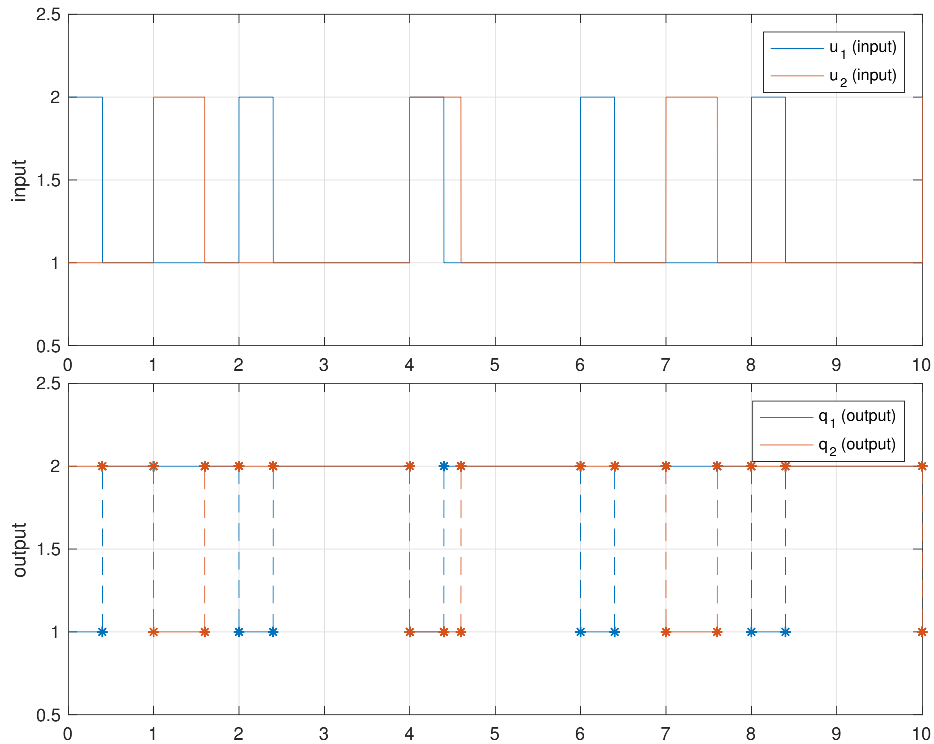

Figure 2: Finite state machine states and inputs.

```
function xdot = f(x, u)
%--------------------------------------------------------------------------
% Matlab M-file Project: HyEQ Toolbox @  Hybrid Systems Laboratory (HSL),
% https://hybrid.soe.ucsc.edu/software
% http://hybridsimulator.wordpress.com/
%--------------------------------------------------------------------------
% Project: Simulation of a hybrid system Finite state machine (FSM)
% Description: Flow map
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
%   See also HYEQSOLVER, PLOTARC, PLOTARC3, PLOTFLOWS, PLOTHARC,
%   PLOTHARCCOLOR, PLOTHARCCOLOR3D, PLOTHYBRIDARC, PLOTJUMPS.
%   Copyright @ Hybrid Systems Laboratory (HSL),
%   Revision: 0.0.0.3 Date: 05/20/2015 3:42:00
%--------------------------------------------------------------------------
% flow map: xdot=f(x,u);

n = length(x); % size of the q state
qdot = zeros(n,1); % dynamics of the q state
xdot = [qdot];
```

Flow set

```
function v  = C(x, u)
%--------------------------------------------------------------------------
% Matlab M-file Project: HyEQ Toolbox @  Hybrid Systems Laboratory (HSL),
% https://hybrid.soe.ucsc.edu/software
```

```
% http://hybridsimulator.wordpress.com/
%--------------------------------------------------------------------------
% Project: Simulation of a hybrid system Finite state machine (FSM)
% Description: Flow set
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
%   See also HYEQSOLVER, PLOTARC, PLOTARC3, PLOTFLOWS, PLOTHARC,
%   PLOTHARCCOLOR, PLOTHARCCOLOR3D, PLOTHYBRIDARC, PLOTJUMPS.
%   Copyright @ Hybrid Systems Laboratory (HSL),
%   Revision: 0.0.0.3 Date: 05/20/2015 3:42:00
%
% Check on flow conditions
% E.g.,
% if (x(1) >= u(1))  % flow condition
%     v = 1;  % report flow
% else
%     v = 0;   % do not report flow
% end

xtemp = zeros(2,1);
xtemp = x;
x = xtemp;

% n = length(x); % size of the q state
qplus = zeros(2,1); % dynamics of the q state
% e.g.,
qplus(1) = 3-u(1);
qplus(2) = 3-u(2);
if x(1)==qplus(1) && x(2)==qplus(2)
    v = 1; % report flow
else
    v = 0; % do not report flow
end

```

Jump map

```
function xplus = g(x, u)
%--------------------------------------------------------------------------
% Matlab M-file Project: HyEQ Toolbox @  Hybrid Systems Laboratory (HSL),
% https://hybrid.soe.ucsc.edu/software
% http://hybridsimulator.wordpress.com/
%--------------------------------------------------------------------------
% Project: Simulation of a hybrid system Finite state machine (FSM)
% Description: Jump map
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
%   See also HYEQSOLVER, PLOTARC, PLOTARC3, PLOTFLOWS, PLOTHARC,
%   PLOTHARCCOLOR, PLOTHARCCOLOR3D, PLOTHYBRIDARC, PLOTJUMPS.
%   Copyright @ Hybrid Systems Laboratory (HSL),
%   Revision: 0.0.0.3 Date: 05/20/2015 3:42:00
%--------------------------------------------------------------------------
% jump map: xplus = g(x,u);

% n = length(x); % size of the q state
```

```
qplus = zeros(2,1); % dynamics of the q state
% e.g.,
qplus(1) = 3-u(1);
qplus(2) = 3-u(2);

xplus = [qplus];

```

Jump set

```
function v  = D(x, u)
%--------------------------------------------------------------------------
% Matlab M-file Project: HyEQ Toolbox @  Hybrid Systems Laboratory (HSL),
% https://hybrid.soe.ucsc.edu/software
% http://hybridsimulator.wordpress.com/
%--------------------------------------------------------------------------
% Project: Simulation of a hybrid system Finite state machine (FSM)
% Description: Jump set
%--------------------------------------------------------------------------
%--------------------------------------------------------------------------
%   See also HYEQSOLVER, PLOTARC, PLOTARC3, PLOTFLOWS, PLOTHARC,
%   PLOTHARCCOLOR, PLOTHARCCOLOR3D, PLOTHYBRIDARC, PLOTJUMPS.
%   Copyright @ Hybrid Systems Laboratory (HSL),
%   Revision: 0.0.0.3 Date: 05/20/2015 3:42:00
%
% Check on jump conditions
% % E.g.,
% if (x(1) <= u(1)) && (x(2) <= 0)  % jump condition
%      v = 1;  % report jump
% else
%      v = 0;   % do not report jump
% end
xtemp = zeros(2,1);
xtemp = x;
x = xtemp;

% n = length(x); % size of the q state
qplus = zeros(2,1); % dynamics of the q state
% e.g.,
qplus(1) = 3-u(1);
qplus(2) = 3-u(2);
if x(1)==qplus(1) && x(2)==qplus(2)
    v = 0; % do not report jump
else
    v = 1; % report jump
end
```

□