

Exemple de code VHDL pour les machines à états

Dans cet exemple, on va parler des machines à états. Nous allons considérer la situation où il y a une machine qui vend les films pour 5\$. L'utilisateur a le droit de mettre des pièces de 1\$ ou de 2\$ et dès que le montant arrive à 5\$ ou plus, il y a un film qui sort. Il est aussi possible de l'utilisateur mette 6\$ et dans ce cas, la machine donne un film et remet aussi la monnaie.

Nous voulons faire une machine de Moore et donc, les sorties ne dépendent que de l'état. Une information de plus est que, si la personne mettait de l'argent pendant que le film sort, l'argent sera perdu.

Avec cette description, il devrait déjà être possible de déterminer quelques entrées et sorties. Puisque c'est une machine à états, il doit y avoir une horloge. On peut soit insérer 1\$, 2\$ ou rien, donc on va avoir une entrée pour chaque type de monnaie. À la sortie, on peut soit donner un film ou soit donner un film et de la monnaie.

On peut trouver plus de détails sur le code en regardant les acétates du cours.

```
LIBRARY IEEE;

USE IEEE.STD_LOGIC_1164.ALL;
USE IEEE.STD_LOGIC_UNSIGNED.ALL;

ENTITY machine IS
    PORT (
        clk           : IN  STD_LOGIC;
        un_dollar     : IN  STD_LOGIC;
        deux_dollars  : IN  STD_LOGIC;
        film          : OUT STD_LOGIC;
        monnaie       : OUT STD_LOGIC
    );
END machine;

ARCHITECTURE rtl OF machine IS

    TYPE etats IS (zero, un , deux, trois, quatre, cinq, six);

    SIGNAL etat_present  : etats;
    SIGNAL etat_prochain : etats;

BEGIN

    PROCESS (etat_present, un_dollar, deux_dollars)
    BEGIN
        CASE etat_present IS
            WHEN zero =>
```

```

    film <= '0';
    monnaie <= '0';
    IF un_dollar = '1' THEN
        etat_prochain <= un;
    ELSIF deux_dollars = '1' THEN
        etat_prochain <= deux;
    ELSE
        etat_prochain <= zero;
    END IF;
WHEN un =>
    film <= '0';
    monnaie <= '0';
    IF un_dollar = '1' THEN
        etat_prochain <= deux;
    ELSIF deux_dollars = '1' THEN
        etat_prochain <= trois;
    ELSE
        etat_prochain <= un;
    END IF;
WHEN deux =>
    film <= '0';
    monnaie <= '0';
    IF un_dollar = '1' THEN
        etat_prochain <= trois;
    ELSIF deux_dollars = '1' THEN
        etat_prochain <= quatre;
    ELSE
        etat_prochain <= deux;
    END IF;
WHEN trois =>
    film <= '0';
    monnaie <= '0';
    IF un_dollar = '1' THEN
        etat_prochain <= quatre;
    ELSIF deux_dollars = '1' THEN
        etat_prochain <= cinq;
    ELSE
        etat_prochain <= trois;
    END IF;
WHEN quatre =>
    film <= '0';
    monnaie <= '0';
    IF un_dollar = '1' THEN
        etat_prochain <= cinq;
    ELSIF deux_dollars = '1' THEN
        etat_prochain <= six;
    ELSE

```

```
        etat_prochain <= quatre;
    END IF;
    WHEN cinq =>
        film <= '1';
        monnaie <= '0';
        etat_prochain <= zero;
    WHEN six =>
        film <= '1';
        monnaie <= '1';
        etat_prochain <= zero;
    END CASE;
END PROCESS;

PROCESS (clk)
BEGIN
    IF clk'EVENT AND clk = '1' THEN
        etat_present <= etat_prochain;
    END IF;
END PROCESS;

END;
```