

CEG 3555: Systèmes Numériques II
(Automne 2023)

Prof. Rami Abielmona

Solutions Possibles pour Devoir #3: *Concepts de Circuits Séquentiels*

06 Novembre, 2023

Question I

Cette question se concerne avec la réduction de table d'état en utilisant la *méthode de division d'équivalence à classe*. La partie c est question 6.18 de votre manuel.

Partie a

Les cloisons pour cette table d'état sont:

$$\Pi_1 = (ABCDEFGH)$$

$$\Pi_2 = (ACDE)(BFG)(H)$$

$$\Pi_3 = (ACDE)(BG)(F)(H)$$

$$\Pi_4 = (ACE)(D)(BG)(F)(H)$$

$$\Pi_5 = (ACE)(D)(BG)(F)(H)$$

Alors, on peut réduire notre table originale de 8 à 5 états. En éliminant les rangées correspondantes aux états C, E, et G, et en remplaçant dans les colonnes, les états C et E avec A, et G avec B, on obtient la table réduite qui est démontrée en table 1.

Partie b

Les cloisons pour cette table d'état sont:

| Présent état $y_2y_1y_0$ | Prochain état | | Rendement | |
|-----------------------------|---------------|-------------|-------------|-------------|
| | $x = 0$ | $x = 1$ | $Y_2Y_1Y_0$ | $Y_2Y_1Y_0$ |
| | $Y_2Y_1Y_0$ | $Y_2Y_1Y_0$ | | |
| A | B | A | 0 | 0 |
| B | A | D | 1 | 0 |
| D | F | A | 0 | 0 |
| F | F | H | 1 | 0 |
| H | D | F | 1 | 1 |

Table 1: Table réduite de question Ia

$$\Pi_1 = (ABCDEFGH)$$

$$\Pi_2 = (ACEFH)(BDG)$$

$$\Pi_3 = (AEF)(CH)(BG)(D)$$

$$\Pi_4 = (AEF)(CH)(BG)(D)$$

Alors, on peut réduire notre table originale de 8 à 4 états. En éliminant les rangées correspondantes aux états E, F, H et G, et en remplaçant dans les colonnes, les états E et F avec A, H avec C, et G avec B, on obtient la table réduite qui est démontrée en table 2.

| Présent état y_1y_0 | Prochain état | | Rendement |
|--------------------------|---------------|----------|-----------|
| | $x = 0$ | $x = 1$ | |
| | Y_1Y_0 | Y_1Y_0 | |
| A | B | C | 1 |
| B | D | A | 0 |
| C | A | A | 1 |
| D | A | C | 0 |

Table 2: Table réduite de question Ib

Partie c

Les cloisons pour cette table d'état sont:

$$\Pi_1 = (ABCDEFG)$$

$$\Pi_2 = (ADG)(BCEF)$$

$$\Pi_3 = (AG)(D)(B)(CE)(F)$$

$$\Pi_4 = (A)(G)(D)(B)(CE)(F)$$

$$\Pi_5 = (A)(G)(D)(B)(CE)(F)$$

Alors, on peut réduire notre table originale de 7 à 6 états. En éliminant la rangée correspondante à l'état E, et en remplaçant dans les colonnes, l'état E avec C, on obtient la table réduite qui est démontrée en table 3.

| Présent état $y_2y_1y_0$ | Prochain état | | Rendement | |
|-----------------------------|---------------|-------------|-----------|---------|
| | $x = 0$ | $x = 1$ | $x = 0$ | $x = 1$ |
| | $Y_2Y_1Y_0$ | $Y_2Y_1Y_0$ | | |
| A | B | C | 0 | 0 |
| B | D | - | 0 | 1 |
| C | F | C | 0 | 1 |
| D | B | G | 0 | 0 |
| F | C | D | 0 | 1 |
| G | F | - | 0 | 0 |

Table 3: Table réduite de question Ic

Question II

Cette question se concerne avec la conception en utilisant la méthode de FSM.
Les parties a et b sont questions 6.37 et 6.38 de votre manuel.

Partie a

Pour s'assurer que le dispositif 3 obtiendra le bus, le FSM doit être modifié comme suit (figure 1).

Partie b

L'entité *questionIIb* est codée en VHDL comme ci-dessous:

```
LIBRARY ieee;
USE ieee.std_logic_1164.all;

ENTITY questionIIb IS
PORT (i_clock : IN STD_LOGIC;
      i_resetb : IN STD_LOGIC;
      i_r : IN STD_LOGIC_VECTOR(1 to 3);
      o_g : OUT STD_LOGIC_VECTOR(1 to 3));
END questionIIb;

ARCHITECTURE Behavior OF questionIIb IS TYPE

stateType IS (Idle, gnt1, gnt2, gnt3);
SIGNAL int_state: stateType;
```

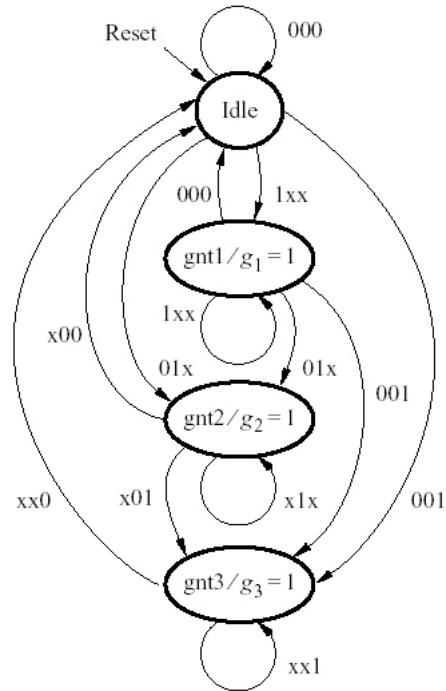


Figure 1: FSM de l'arbitre sans famine

```

BEGIN
PROCESS (i_resetb, i_clock)
BEGIN
    IF (i_resetb = '0') THEN
        int_state <= Idle;
    ELSIF (i_clock'EVENT AND i_clock = '1') THEN
        CASE int_state IS
            WHEN Idle =>
                IF i_r(1) = '1' THEN
                    int_state <= gnt1;
                ELSIF i_r(2) = '1' THEN
                    int_state <= gnt2;
                ELSIF i_r(3) = '1' THEN
                    int_state <= gnt3;
                ELSE int_state <= Idle;
                END IF;
            WHEN gnt1 =>
                IF i_r(1) = '1' THEN

```

```

        int_state <= gnt1;
ELSIF i_r(2) = '1' THEN
        int_state <= gnt2;
ELSIF i_r(3) = '1' THEN
        int_state <= gnt3;
ELSE int_state<= Idle;
END IF;
WHEN gnt2 =>
    IF i_r(2) = '1' THEN
        int_state <= gnt2;
ELSIF i_r(3) = '1' THEN
        int_state <= gnt3;
ELSE int_state<= Idle;
END IF;
WHEN gnt3 =>
    IF i_r(3) = '1' THEN
        int_state <= gnt3;
ELSE int_state <= Idle;
END IF;
END CASE;
END IF;
END PROCESS;

g(1) <= '1' WHEN int_state = gnt1 ELSE
    '0';
g(2) <= '1' WHEN int_state = gnt2 ELSE
    '0';
g(3) <= '1' WHEN int_state = gnt3 ELSE
    '0';
END Behavior;

```

Partie c

Commençons par construire un diagramme d'état des spécifications données. Le diagramme contient quatre états: *A* pour désigner qu'on a pas reçu notre premier '0', *B* pour désigner qu'on la reçu, *C* pour désigner qu'on a pas reçu un deuxième '0', et *D* pour désigner qu'on la reçu. Alors, le diagramme d'état peut être vu dans la figure 2. Constatez qu'on a choisi une machine de Moore parcqu'on peut simplifier nos expressions d'excitation.

On peut convertir notre diagramme d'état en table d'état. La table est montrée dans la table 4.

Essayons de réduire notre table d'état en utilisant la méthode de division d'équivalence à classe. Nos cloisons sont:

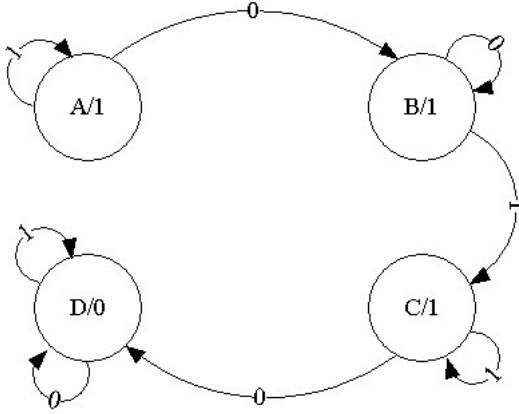


Figure 2: Diagramme d'état

| Présent état | Prochain état | | Rendement |
|-----------------|---------------|---------|-----------|
| | $x = 0$ | $x = 1$ | |
| A | B | A | 1 |
| B | B | C | 1 |
| C | D | C | 1 |
| D | D | D | 0 |

Table 4: Table d'état pour le détecteur de blocs de '0'

$$\begin{aligned}\Pi_1 &= (ABCD) \\ \Pi_2 &= (ABC)(D) \\ \Pi_3 &= (AB)(C)(D) \\ \Pi_4 &= (A)(B)(C)(D)\end{aligned}$$

On réalise que notre table est déjà réduite à quatre états. En suivant nos étapes de la méthode de FSM, et en choisissant un encodage séquentiel et l'attribution suivante: $A \rightarrow 00$, $B \rightarrow 01$, $C \rightarrow 10$ et $D \rightarrow 11$, on peut obtenir notre table de transition (table 5).

Finalement, on arrive à nos équations de conception de notre variable de prochain état et de notre sortie de système. Notez que puisque cette machine est de type Moore, la sortie dépend sur la variable de présent état seulement. Vérifiez la figure 3 pour le circuit final, réalisé avec deux bascules de type D.

$$Y_1 = y_1 + y_0 \cdot x \quad (1)$$

| Présent état y_1y_0 | Prochain état | | Rendement z |
|--------------------------|---------------|----------|------------------|
| | $x = 0$ | $x = 1$ | |
| | Y_1Y_0 | Y_1Y_0 | |
| A \rightarrow 00 | 01 | 00 | 1 |
| B \rightarrow 01 | 01 | 10 | 1 |
| C \rightarrow 10 | 11 | 10 | 1 |
| D \rightarrow 11 | 11 | 11 | 0 |

Table 5: Table de transition de notre exemple

$$Y_0 = \bar{x} + y_1 \cdot y_0 \quad (2)$$

$$z = \overline{y_1 \cdot y_0} \quad (3)$$

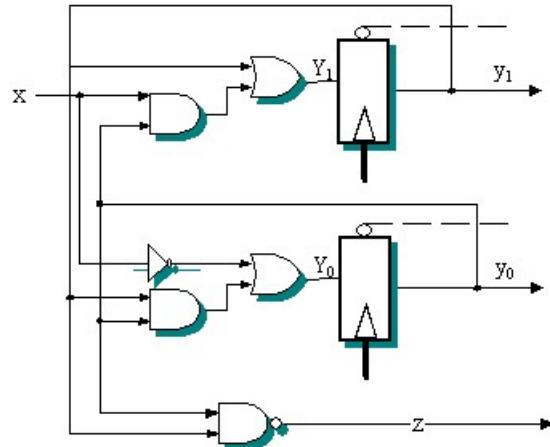


Figure 3: Circuit final

Question III

Cette question se concerne avec les tables de transition d'états.

Partie a

La table de transition pour un encodage séquentiel est démontrée en table 6

Pour la réalisation à bascules de type D, la table d'excitation est la même que table 6, alors que les équations de conception deviennent:

$$Y_2 = y_1 \cdot y_0 \cdot x$$

| Présent état $y_2y_1y_0$ | Prochain état | | Rendement | |
|-----------------------------|---------------|-------------|-----------|---------|
| | $x = 0$ | $x = 1$ | $x = 0$ | $x = 1$ |
| | $Y_2Y_1Y_0$ | $Y_2Y_1Y_0$ | | |
| A → 000 | 001 | 000 | 0 | 0 |
| B → 001 | 000 | 010 | 1 | 0 |
| C → 010 | 011 | 000 | 0 | 0 |
| D → 011 | 011 | 100 | 1 | 0 |
| E → 100 | 010 | 011 | 1 | 1 |

Table 6: Table de transition de question IIIa

| Présent état $y_2y_1y_0$ | Prochain état | | | | | | Rendement | |
|-----------------------------|---------------|----------|----------|----------|----------|----------|-----------|---------|
| | $x = 0$ | | | $x = 1$ | | | $x = 0$ | $x = 1$ |
| | S_2R_2 | S_1R_1 | S_0R_0 | S_2R_2 | S_1R_1 | S_0R_0 | | |
| 000 | 0d | 0d | 10 | 0d | 0d | 0d | 0 | 0 |
| 001 | 0d | 0d | 01 | 0d | 10 | 01 | 1 | 0 |
| 010 | 0d | d0 | 10 | 0d | 01 | 0d | 0 | 0 |
| 011 | 0d | d0 | d0 | 10 | 01 | 01 | 1 | 0 |
| 100 | 01 | 10 | 0d | 01 | 10 | 10 | 1 | 1 |

Table 7: Table d'excitation de question IIIa (bascules S-R)

$$Y_1 = y_2 + y_1 \cdot \bar{x} + \bar{y}_1 \cdot y_0 \cdot x$$

$$Y_0 = y_2 \cdot x + y_1 \cdot \bar{x} + \bar{y}_2 \cdot \bar{y}_0 \cdot \bar{x}$$

$$z = y_2 + y_0 \cdot \bar{x}$$

Pour la réalisation à bascules de type S-R, la table d'excitation est démontrée en table 7, alors que les équations de conception deviennent:

$$S_2 = y_1 \cdot y_0 \cdot x$$

$$R_2 = \bar{y}_0$$

$$S_1 = y_2 + \bar{y}_1 \cdot y_0 \cdot x$$

$$R_1 = y_1 \cdot x$$

$$S_0 = y_2 \cdot x + \bar{y}_2 \cdot \bar{y}_0 \cdot \bar{x}$$

$$R_0 = \bar{y}_1 \cdot y_0 + y_0 \cdot x$$

$$z = y_2 + y_0 \cdot \bar{x}$$

| | | y_1y_0 | | | |
|-------|---|----------|----|----|----|
| | | 00 | 01 | 10 | 11 |
| y_2 | 0 | A | B | - | - |
| 1 | C | D | E | - | - |

| Présent état $y_2y_1y_0$ | Prochain état | | Rendement | |
|-----------------------------|---------------|-------------|-----------|---------|
| | $x = 0$ | $x = 1$ | $x = 0$ | $x = 1$ |
| | $Y_2Y_1Y_0$ | $Y_2Y_1Y_0$ | | |
| A \rightarrow 000 | 001 | 000 | 0 | 0 |
| B \rightarrow 001 | 000 | 100 | 1 | 0 |
| C \rightarrow 100 | 101 | 000 | 0 | 0 |
| D \rightarrow 101 | 101 | 111 | 1 | 0 |
| E \rightarrow 111 | 100 | 101 | 1 | 1 |

Table 8: Carte d'attribution et table de transition de la question IIIb

Partie b

En appliquant les règles de contiguïtés, on obtient les conditions suivantes:

Règle I (C, D), (A, C)

Règle II (A, B), (A, C), (A, D), (D, E), (C, D)

Règle III (B, D, E)

Vérifiez la table 8 pour la carte d'attribution d'état pour cette table d'état, aussi que la table de transition.

Pour la réalisation à bascules de type D, les équations de conception sont:

$$Y_2 = y_2 \cdot \bar{x} + y_0 \cdot x$$

$$Y_1 = y_2 \cdot \bar{y}_1 \cdot y_0 \cdot x$$

$$Y_0 = \bar{y}_0 \cdot \bar{x} + y_1 \cdot x + y_2 \cdot \bar{y}_1 \cdot y_0$$

$$z = y_1 + y_0 \cdot \bar{x}$$

Finalement, pour la réalisation à bascules de type S-R, la table d'excitation est démontrée en table 9, et les équations de conception sont:

$$S_2 = y_0 \cdot x$$

$$R_2 = \bar{y}_0 \cdot x$$

$$S_1 = y_2 \cdot \bar{y}_1 \cdot y_0 \cdot x$$

$$R_1 = y_1$$

$$S_0 = \bar{y}_0 \cdot \bar{x}$$

$$R_0 = \bar{y}_2 \cdot y_0 + y_1 \cdot \bar{x}$$

$$z = y_2 + y_0 \cdot \bar{x}$$

| Présent état $y_2y_1y_0$ | Prochain état | | | | | | Rendement | |
|-----------------------------|---------------|----------|----------|----------|----------|----------|-----------|-------|
| | x = 0 | | | x = 1 | | | x = 0 | x = 1 |
| | S_2R_2 | S_1R_1 | S_0R_0 | S_2R_2 | S_1R_1 | S_0R_0 | | |
| 000 | 0d | 0d | 10 | 0d | 0d | 0d | 0 | 0 |
| 001 | 0d | 0d | 01 | 10 | 0d | 01 | 1 | 0 |
| 100 | d0 | 0d | 10 | 01 | 0d | 0d | 0 | 0 |
| 101 | d0 | 0d | d0 | d0 | 10 | d0 | 1 | 0 |
| 111 | d0 | 01 | 01 | d0 | 01 | d0 | 1 | 1 |

Table 9: Table d'excitation de question IIIb (bascules S-R)

Remerciements

Les réponses et figures liées aux questions du manuel sont tirées du manuel d'accompagnement de l'instructeur pour *Fundamentals of Digital Logic with VHDL Design* par Stephen Brown et Zvonko Vranesic.

Les réponses liées à la troisième question sont tirées du manuel d'accompagnement de l'instructeur pour *Digital Principles and Design* par Donald D. Givone.