

# Rapport du Travail 9

Alexandre Dewilde

March 26, 2021

Note : le fichier du circuit s'appelle circuit.asc dans le zip

## 1 Cahier des charges

Une FSM qui indique à un jardinier ce qu'il doit faire

Il y a 2 entrées différentes la saison et la météo s'il y a du soleil ou si il pleut.

- La premiere action c'est ne rien faire, cette action se déroule en hiver ou alors lorsque rien n'as été planter/remuer cette année. on attends que le printemps arrive et qu'il pleuvent pour pouvoir remuer la terre.
- La seconde action est l'action de remuer la terre, celui ce produit au printemps, seulement lors de jour de pluie, tant que les jours sont des jours de pluie on continue de remuer la terre.
- La troisieme action est l'action de planter, celle-ci ne se fait qu'au printemps lorsqu'il fait beau et que la terre a été remué, si l'été arrive avant d'avoir planté on retourne à la premiere action et on attends l'année d'après.
- La quatrieme action est l'action d'arrosage, tant que l'on est pas à l'automne ou en hiver (si on arrive en hiver il faut arreter le cycle et retourner à l'action de ne rien faire), il faut arroser les plantes qui ont été planté aux préalables.
- La cinquieme action est l'action de récolter, cela se passe à l'automne, apres cela on revient à la premiere action d'attente.

## 2 Inputs et outputs

### 2.1 Inputs

$I_2$  représente la météo :  $0 \rightarrow$  pluie et  $1 \rightarrow$  soleil

$I_0$  et  $I_1$  représente la saison

00  $\rightarrow$  Hiver

01  $\rightarrow$  Printemps

10  $\rightarrow$  Ete

11  $\rightarrow$  Automne

## 2.2 Outputs

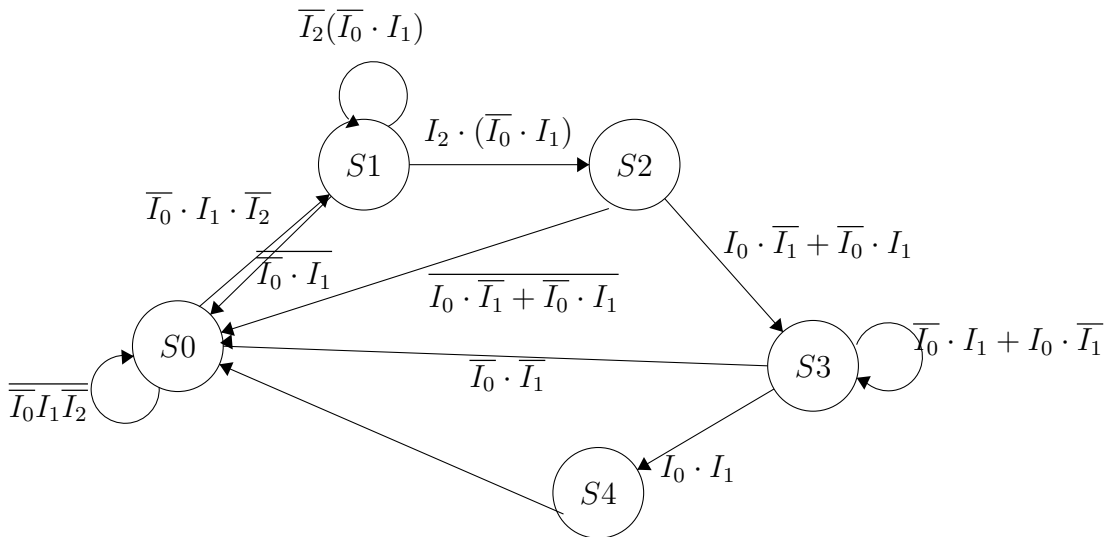
$Y_0$ ,  $Y_1$  et  $Y_3$  représente l'action que le jardinier doit réaliser

$Y_0$	$Y_1$	$Y_2$	output
0	0	0	Ne rien faire
0	0	1	Remuer la terre
0	1	1	Planter
0	1	0	Arroser
1	0	0	Cueillir

## 3 Choix entre FSM de Moore et Mealy

Pour réaliser, la FSM je vais prendre la FSM de moore, celle ci permetra de réaliser de manière simple le cahier des charges.

## 4 Diagramme d'état



S	$I_0 I_1 I_2$	S'
S0	1 X X	S0
S0	X 0 X	S0
S0	X X 1	S0
S0	0 1 0	S1
S1	0 1 0	S1
S1	0 1 1	S2
S1	1 X X	S0
S1	X 0 X	S0
S2	0 1 X	S3
S2	1 0 X	S3
S2	0 0 X	S0
S2	1 1 X	S0
S3	0 1 X	S3
S3	1 0 X	S3
S3	1 1 X	S4
S3	0 0 X	S0
S4	X X X	S0

## 5 Encodage des états

States	$S_1S_2S_3$
S0	0 0 0
S1	0 0 1
S2	0 1 1
S3	0 1 0
S4	1 0 0

## 6 Table de vérité pour l'état suivant et optimisation avec Karnaugh

States S			Inputs			Next state S'		
$S_0$	$S_1$	$S_2$	$I_0$	$I_1$	$I_2$	$S'_0$	$S'_1$	$S'_2$
0	0	0	1	X	X	0	0	0
0	0	0	X	0	X	0	0	0
0	0	0	X	X	1	0	0	0
0	0	0	0	1	0	0	0	1
0	0	1	0	1	0	0	0	1
0	0	1	0	1	1	0	1	1
0	0	1	1	X	X	0	0	0
0	0	1	X	0	X	0	0	0
0	1	1	0	1	X	0	1	0
0	1	1	1	0	X	0	1	0
0	1	1	0	0	X	0	0	0
0	1	1	1	1	X	0	0	0
0	1	0	0	1	X	0	1	0
0	1	0	1	0	X	0	1	0
0	1	0	1	1	X	1	0	0
0	1	0	0	0	X	0	0	0
1	0	0	X	X	X	0	0	0

## 6.1 Optimisation karnaugh

### 6.1.1 Pour $S_0$

$\frac{S_0 S_1 S_2}{I_0 I_1 I_2}$	000	001	011	010	100
000	0	0	0	0	0
001	0	0	0	0	0
010	0	0	0	0	0
011	0	0	0	0	0
100	0	0	0	0	0
101	0	0	0	0	0
110	0	0	0	1	0
111	0	0	0	1	0

$$S'_0 = I_0 \cdot I_1 \cdot \overline{S_0} \cdot S_1 \cdot \overline{S_2}$$

### 6.1.2 Pour $S_1$

$\frac{S_0 S_1 S_2}{I_0 I_1 I_2}$	000	001	011	010	100
000	0	0	0	0	0
001	0	0	0	0	0
010	0	0	1	1	0
011	0	1	1	1	0
100	0	0	1	1	0
101	0	0	1	1	0
110	0	0	0	0	0
111	0	0	0	0	0

$$S'_1 = \overline{S_0} \cdot S_1 \cdot \overline{I_0} \cdot I_1 + S_2 \cdot \overline{S_0} \cdot \overline{I_0} \cdot I_1 \cdot I_2 + \overline{S_0} \cdot S_1 \cdot I_0 \cdot \overline{I_1}$$

## 6.2 Pour $S_2$

$\frac{S_0 S_1 S_2}{I_0 I_1 I_2}$	000	001	011	010	100
000	0	0	0	1	0
001	0	0	1	1	0
010	0	0	0	0	0
011	0	0	0	0	0
100	0	0	0	0	0
101	0	0	0	0	0
110	0	0	0	0	0
111	0	0	0	0	0

$\frac{S_0 S_1 S_2}{I_0 I_1 I_2}$	000	001	011	010	100
000	0	0	0	1	0
001	0	0	1	1	0
010	0	0	0	0	0
011	0	0	0	0	0
100	0	0	0	0	0
101	0	0	0	0	0
110	0	0	0	0	0
111	0	0	0	0	0

$$S'_2 = \overline{I_0} \cdot I_1 \cdot \overline{I_2} \cdot \overline{S_0} \cdot \overline{S_1} + \overline{I_0} \cdot I_1 \cdot \overline{S_0} \cdot \overline{S_1} \cdot S_2$$

## 7 Table de vérité pour les sorties et optimisation avec Karnaugh

$S_0 S_1 S_2$	$Y_1 Y_2 Y_3$
0 0 0	0 0 0
0 0 1	0 0 1
0 1 0	0 1 0
0 1 1	0 1 1
1 0 0	1 0 0

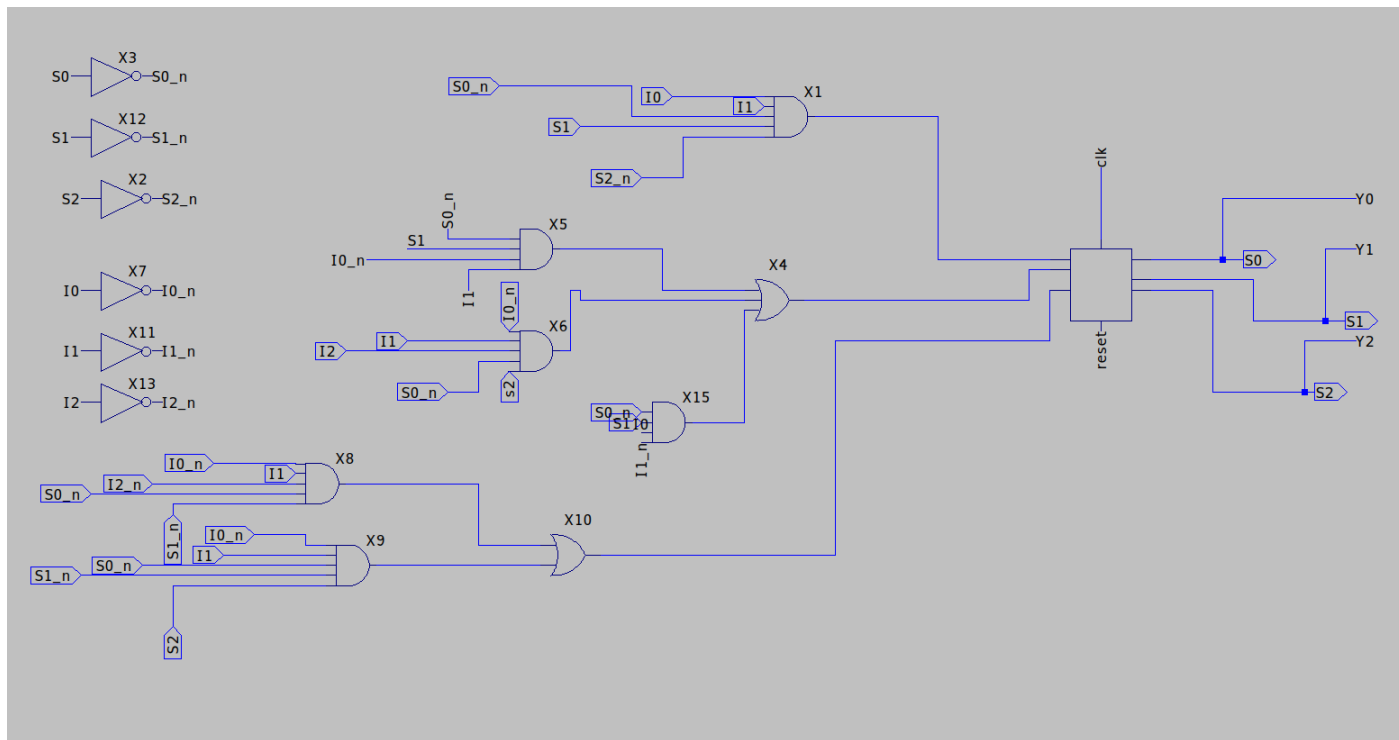
Ici le state égal l'output lui meme donc pas besoin d'optimiser.

$$Y = S$$

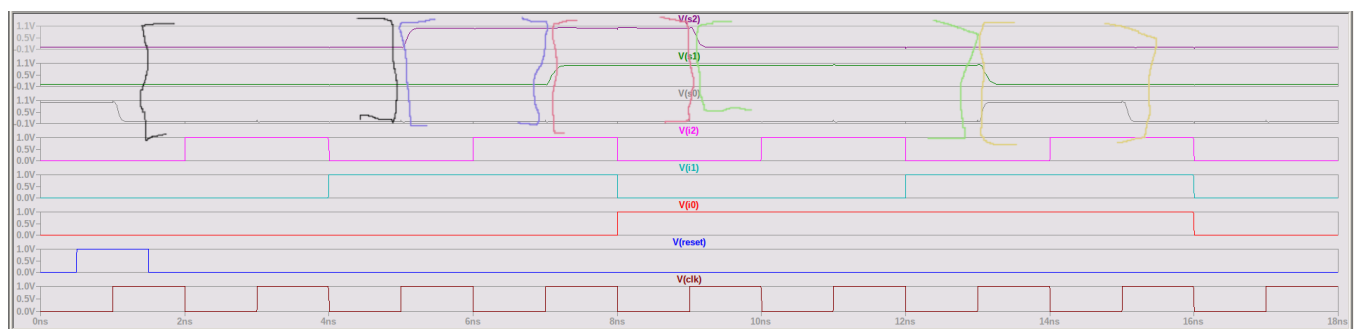
## 8 Implémentation en utilisant les portes logiques réalisées avec les transistors NMOS et PMOS en 50nm

Pour facilité la construction du circuit j'ai construit des portes AND à 4 et 5 entrées et un resetable D-FF à 4 entrées à l'aide de fichiers asy.

(Dans le dossier il s'agit du fichier circuit.asc)



## 9 Résultats de la simulation



Entre crochet noir c'est l'état :  $S_0$

Entre crochet Bleu l'état :  $S_1$

Entre crochet rouge l'état:  $S_2$

Entre crochet Vert l'état :  $S_3$

Entre crochet jaune l'état:  $S_4$

On constate le bon fonctionnement de la FSM, avec les inputs qui font changer d'état celle-ci.

## 10 Conclusion

En conclusion, tout s'est bien passé les résultats de la simulation donnent les résultats attendus.