

TP04 – Pilotage de LED et mémoire

Partie 2

Cédrine Socquet

Rendu

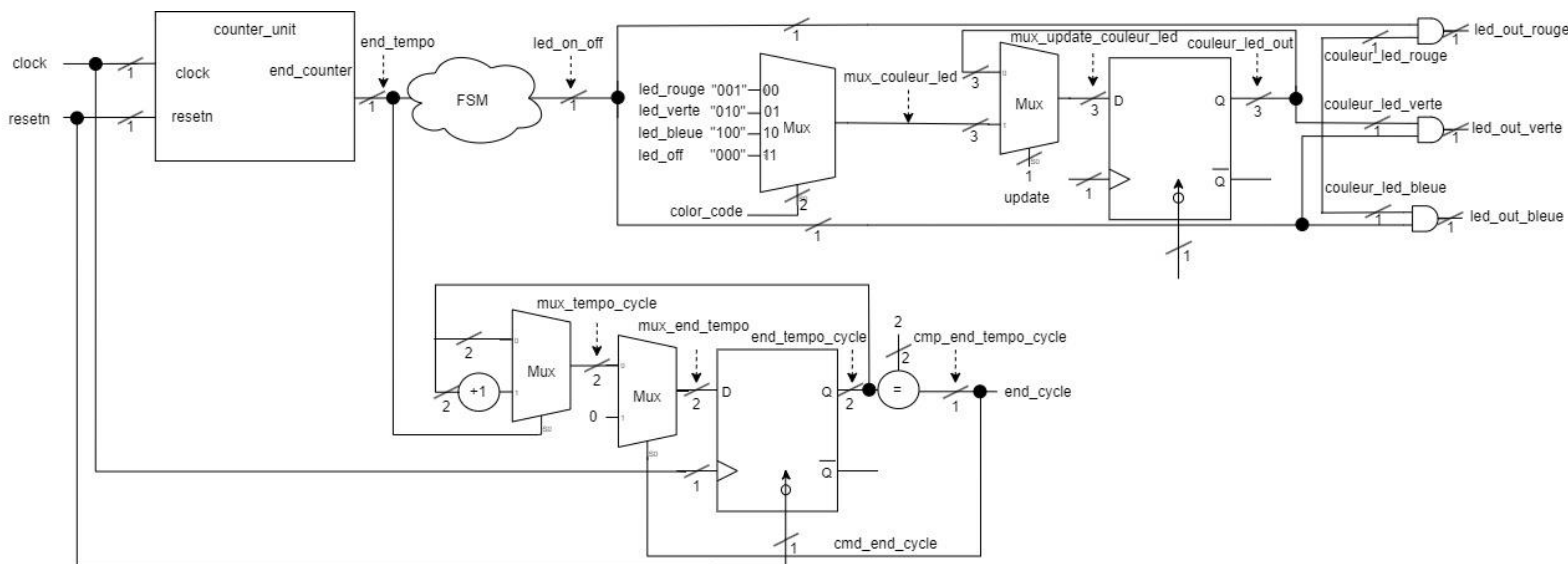
Votre rapport devra contenir :

- Vos schéma RTL
- Vos résultats de simulation avec vos chronogrammes commentés
- Vos résultats de synthèse (analyse de vos ressources utilisées)
- Vos résultats de STA (analyse du rapport de timing)
- Une démonstration de votre design

Vous fournirez également vos codes source commentés.

Réponses

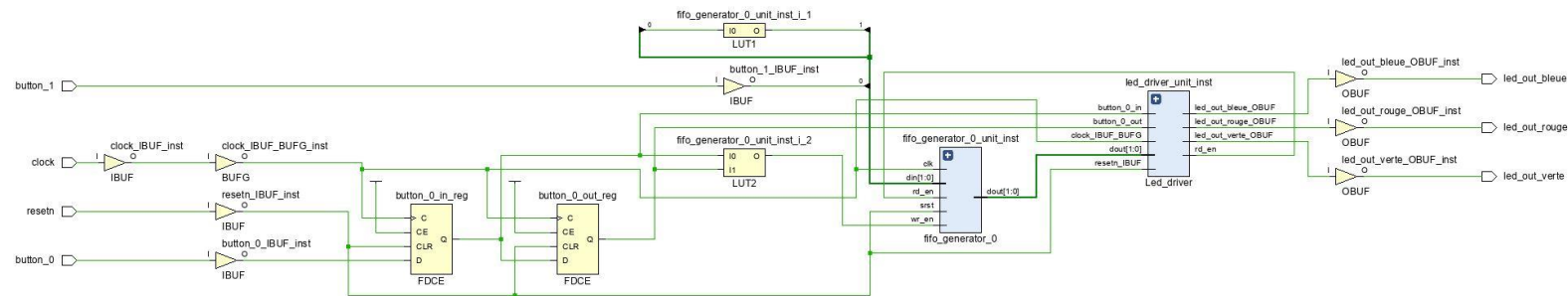
1. Voici mon schéma RTL avec la modification du module LED_driver avec une sortie end_cycle. Cette sortie vaut 1 à la fin d'un cycle allumé/éteint de la LED RGB.



2. Voici ci-dessous mon schéma RTL avec la modification de la logique en entrée du module avec une FIFO. Cette FIFO prend en entrée le code couleur « vert » ou « bleu » suivant l'état du bouton_1 et est connectée en sortie à l'entrée color_code du module LED_driver. La donnée est écrite dans la FIFO lorsqu'il y a un front montant du bouton_0. La donnée de la FIFO est lue lorsque le signal end_cycle du module LED_driver vaut 1.

5. Voici le rapport de synthèse, les ressources utilisées correspondent bien au schéma RTL du tp_with_LED_driver.

En effet, on retrouve bien les entrées du schéma (clock, resetn, button_0 et button_1) et les sorties (led_out_bleue, led_out_verte et led_out_rouge) ainsi que les deux registres du button_0, le composant LED_driver et la FIFO avec ses deux Luts.



Report Cell Usage:

	Cell	Count
1	fifo_generator	1
2	BUFG	1
3	CARRY4	7
4	LUT1	2
5	LUT2	35
6	LUT3	2
7	LUT4	4
8	LUT5	3
9	LUT6	3
10	FDCE	36
11	FDPE	1
12	IBUF	4
13	OBUF	3

Detailed RTL Component Info :

----Adders :

2 Input 2 Bit Adders := 1

----Registers :

3 Bit Registers := 1

2 Bit Registers := 1

1 Bit Registers := 4

----Muxes :

4 Input 3 Bit Muxes := 1

2 Input 2 Bit Muxes := 2

2 Input 1 Bit Muxes := 3

4 Input 1 Bit Muxes := 1

Dans le Report Cell Usage, on retrouve bien les 4 entrées du schéma (clock, resetn, button_0 et button_1) ainsi que les 3 sorties (led_out_bleue, led_out_verte et led_out_rouge). On retrouve également les 36 registres FDCE, les 28 du compteur, les 2 des boutons, 3 pour les couleur_led_out et 2 pour end_tempo_cycle du Led_driver et le dernier pour le present_state de la FSM. On retrouve aussi 1 registre FDPE pour la led_on_off.

Dans le Detailed RTL Component Info, on retrouve les registres ainsi que leur nombre de bits mais aussi l'ensemble des multiplexeurs utilisés dans le schéma, ainsi qu'un adder pour la FIFO.

6. On voit dans le rapport de timing qu'il n'y a pas de violations (TNS et WHS = 0), le Worth Negative Slack (WNS) est de 4.965 ns et le chemin critique (Max Delay Paths) prend sa source ici :

« led_driver_unit_inst/counter_unit_inst/Q_reg[23]/C » et sa destination ici :

« led_driver_unit_inst/counter_unit_inst/Q_reg[25]/D »

| Design Timing Summary

| -----

WNS(ns)	TNS(ns)	TNS Failing Endpoints	TNS Total Endpoints	WHS(ns)	THS(ns)
-----	-----	-----	-----	-----	-----
4.965	0.000	0	144	0.081	0.000

Max Delay Paths

Slack (MET) : 4.965ns (required time - arrival time)

Source: led_driver_unit_inst/counter_unit_inst/Q_reg[23]/C
(rising edge-triggered cell FDCE clocked by sys_clk_pin {rise@0.000ns fall@5.000ns period=10.000ns})

Destination: led_driver_unit_inst/counter_unit_inst/Q_reg[25]/D
(rising edge-triggered cell FDCE clocked by sys_clk_pin {rise@0.000ns fall@5.000ns period=10.000ns})

7. Le bitstream a été généré et le comportement observé sur carte est celui attendu.