TP1 CR

4. Adding new peripherals

Hardware modifications

On rajoute my_periph au projet vivado existant :

Nous le rajoutons a la liste de peripheriques :

```
43          CID_TIMER1,
44
45          CID_UART1,
46
47          CID_MY_PERIPH);
48
49      constant CID_MAX : integer := CID_ENUM'pos(CID_ENUM'right);
50
51  end package;
```

On initialise le composant ahblite_my_periph. L'initialisation se fait comme un composant classique vu dans les TPs precedents. Nous remarquons qu'il est branché directement aux bus de data et d'adresse commun aux autres périphériques.

```
101     component ahblite_defaultslave
102     port (
103         HRESETn     : in  std_logic;
104         HCLK        : in  std_logic;
105         HSEL        : in  std_logic;
106         HREADY      : in  std_logic;
107
108         -- AHB-Lite interface
109         AHBLITE_IN  : in  AHBLite_master_vector;
110         AHBLITE_OUT : out AHBLite_slave_vector);
111     end component;
112
113     component ahblite_my_periph
114     port (
115         HRESETn : in std_logic;
116         HCLK    : in std_logic;
117
118         HSEL    : in std_logic;
119         HREADY  : in std_logic;
120
121         --AHB-Lite interface
122         AHBLITE_IN : in AHBLite_master_vector;
123         AHBLITE_OUT : out AHBLite_slave_vector);
124     end component;
```

Dans ce process, lorsque nous avons l'adresse du périphérique voulu, nous stockons la valeur depuis le périphérique pour la stocker dans la variable sel. Rq, nous avons encore plein d'adresses de périphérique disponibles, ce qui veut dire que nous pourrons encore rajouter des périphériques facilement dans le futur.

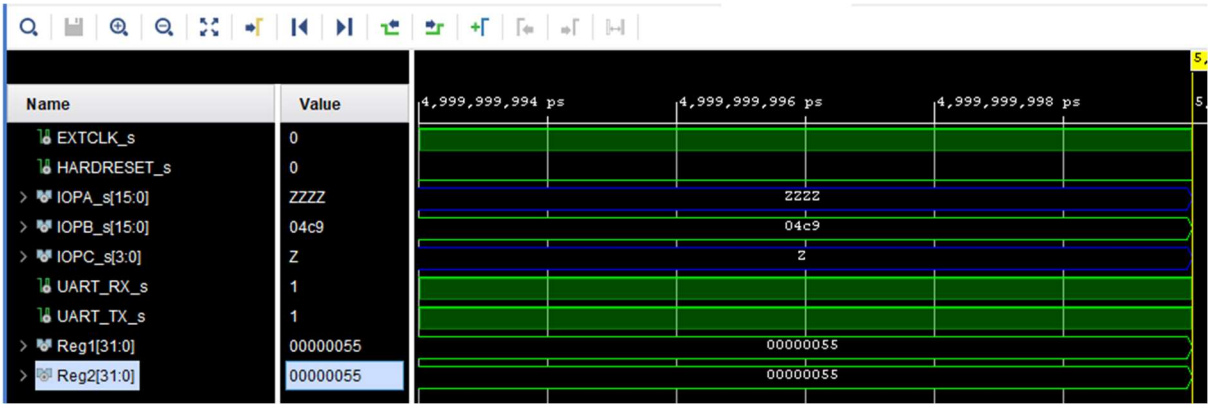
```
35      process (HADDR) begin
36          address <= HADDR;
37      end process;
38
39      process (address) begin
40          case address(31 downto 24) is
41              when x"11" =>
42                  case address(23 downto 20) is
43                      when x"0" =>
44                          case address(19 downto 10) is
45                              when x"00" & "00" => sel <= CID_GPIOA;
46                              when x"00" & "01" => sel <= CID_GPIOB;
47                              when x"00" & "10" => sel <= CID_GPIOC;
48
49                              when x"11" & "00" => sel <= CID_RSTCLK;
50
51                              when x"18" & "00" => sel <= CID_TIMER1;
52
53                              when x"20" & "00" => sel <= CID_UART1;
54
55                              when x"22" & "00" => sel <= CID_MY_PERIPH;
56
57                              when others => sel <= CID_DEFAULT;
58                          end case;
59                      when others => sel <= CID_DEFAULT;
60                  end case ;
61              when others => sel <= CID_DEFAULT;
62          end case;
63      end process;
```

Software modifications :

Dans la partie soft il n'y a quasiment rien a faire, appart faire la déclaration de la structure my-periph, inclure les includes, et utiliser le périphérique dans la fonction main de notre code.

Compile and update the COE file in the Vivado project, then check in U_MY_PERIPH the content of the registers with a simulation.

Nous voyons que les deux registres du nouveau périphérique que nous avons modifiés dans le main sont bien mis à 0x055 :

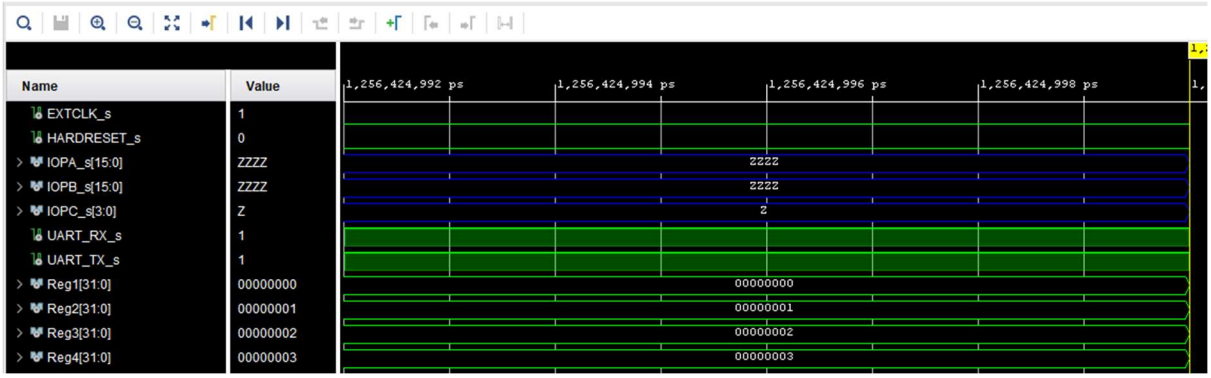


5. Add a 7-segment controller in the peripheral

Dans cette partie nous ne faisons que rajouter encore deux registres dans notre périphérique, tout en rajoutant un contrôleur pour les afficheurs 7 segments. Ce contrôleur sera câblé à l'intérieur de notre périphérique, et c'est lui qui, en utilisant les registres, affichera les datas stockées dans chacun des afficheurs.

La difficulté ici, est de bien faire remonter les signaux depuis l'intérieur du périphérique, jusqu'au icobs_light sans oublier de câbler les signaux à mi-chemin (sinon la simu fonctionnera, mais pas la vraie implémentation sur carte).

compile the code, update COE file, simulate (update the testbench with 7-segment display signals) and implement on board to check that it's working properly :



Et une fois implémenté sur carte :