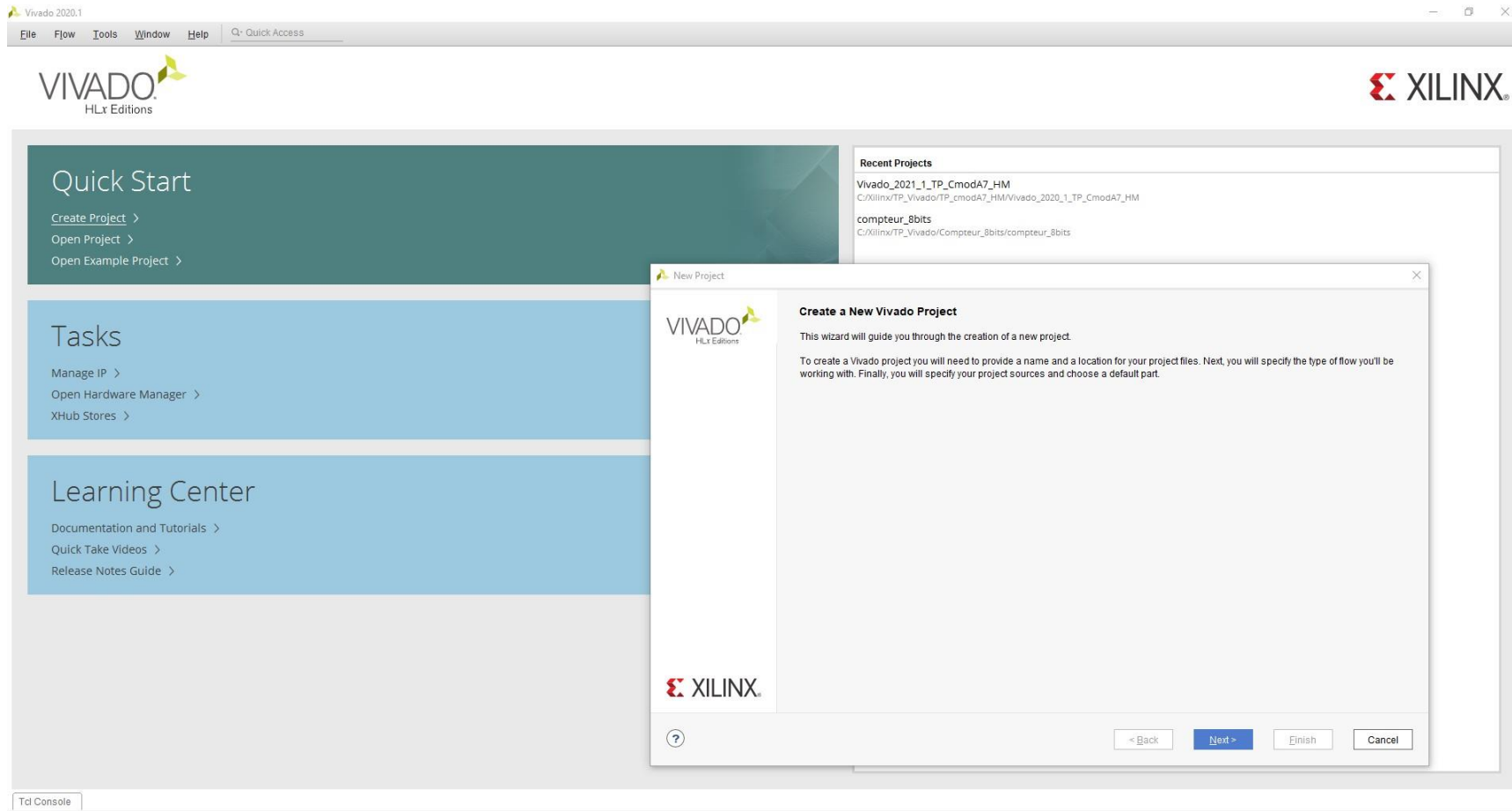


Démonstration avec le logiciel VIVADO 2020.1

Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

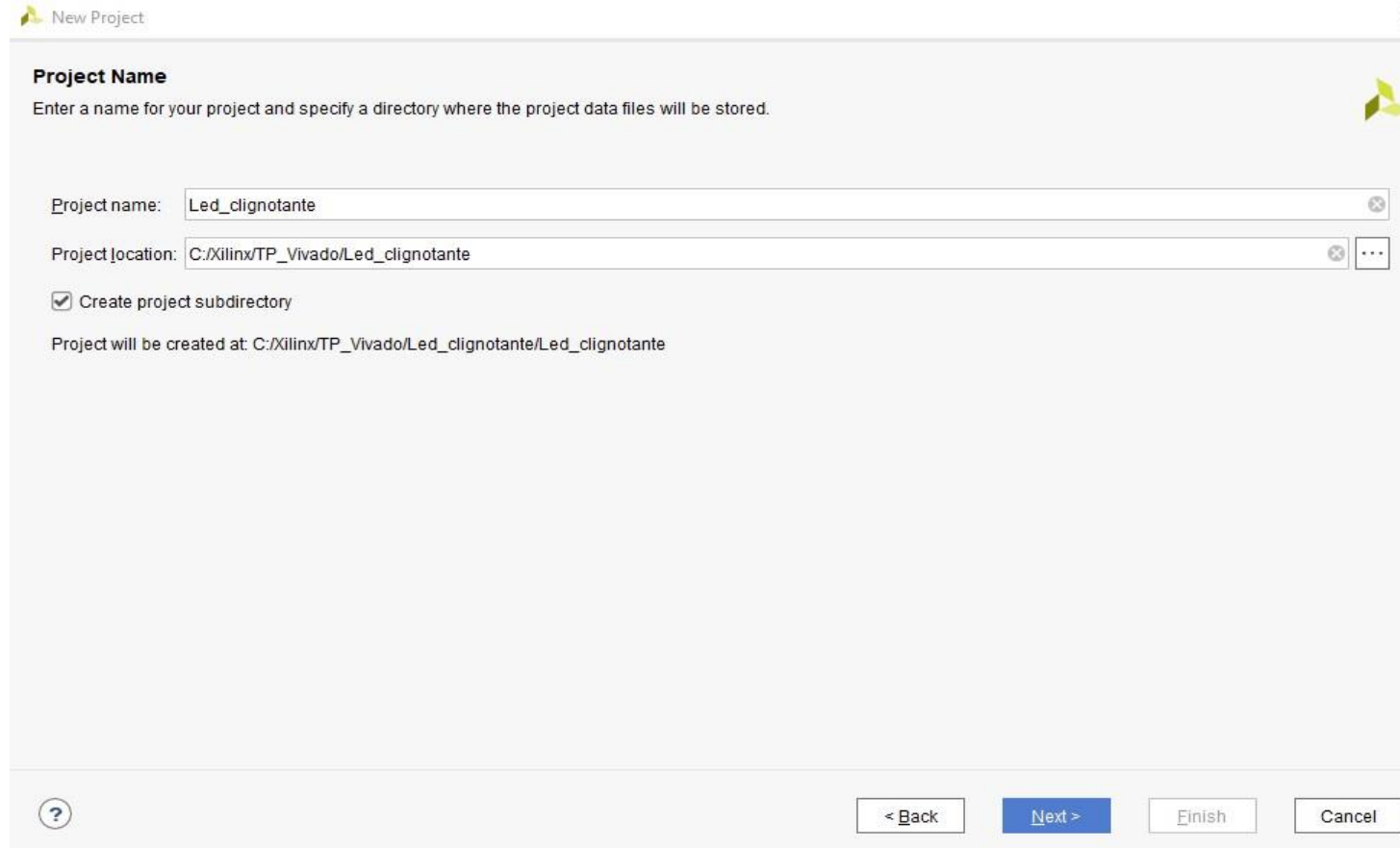


Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante



New Project

Project Name
Enter a name for your project and specify a directory where the project data files will be stored.

Project name:

Project location:

☒ Create project subdirectory

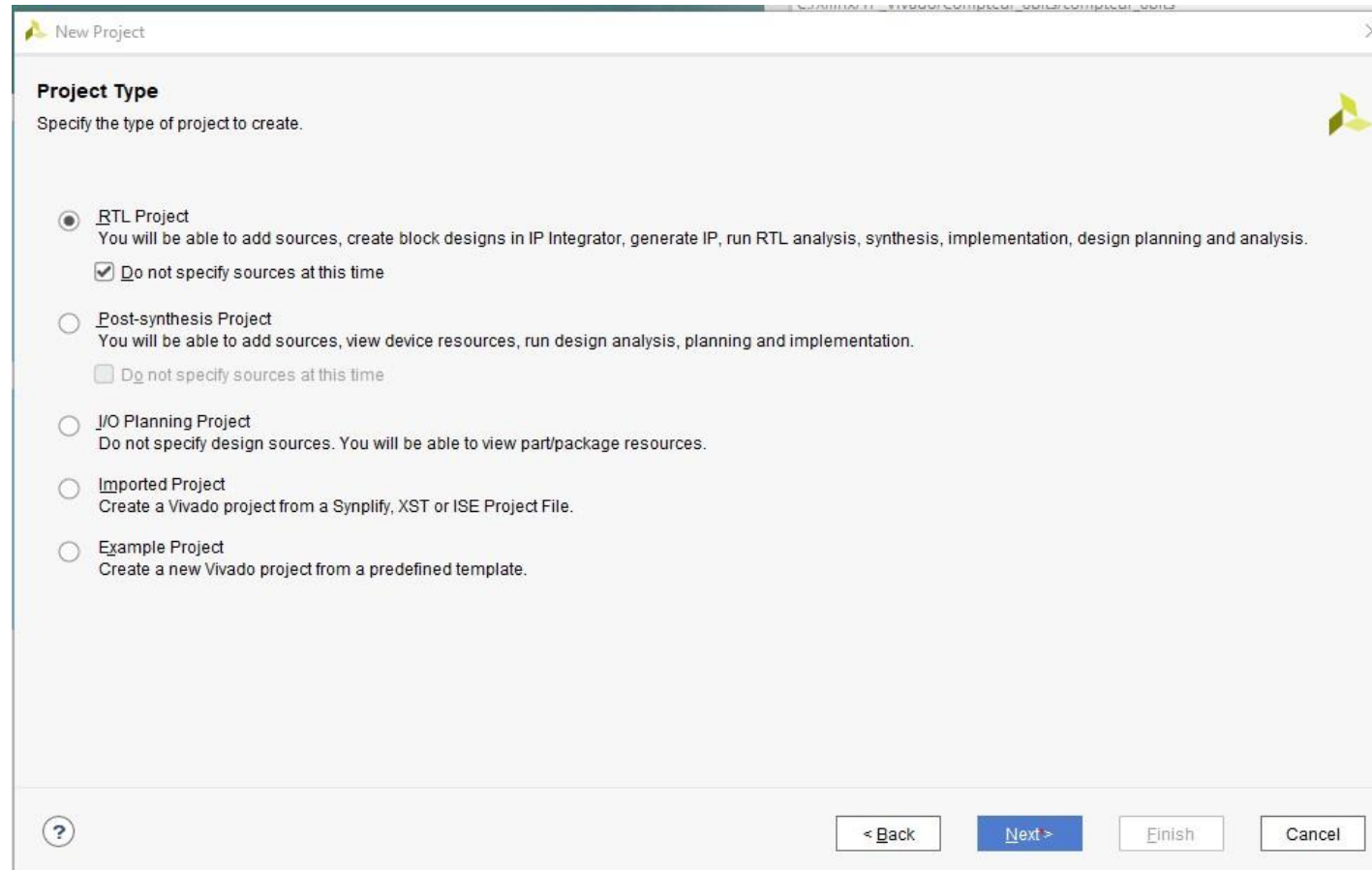
Project will be created at: C:/Xilinx/TP_Vivado/Led_clignotante/Led_clignotante

Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

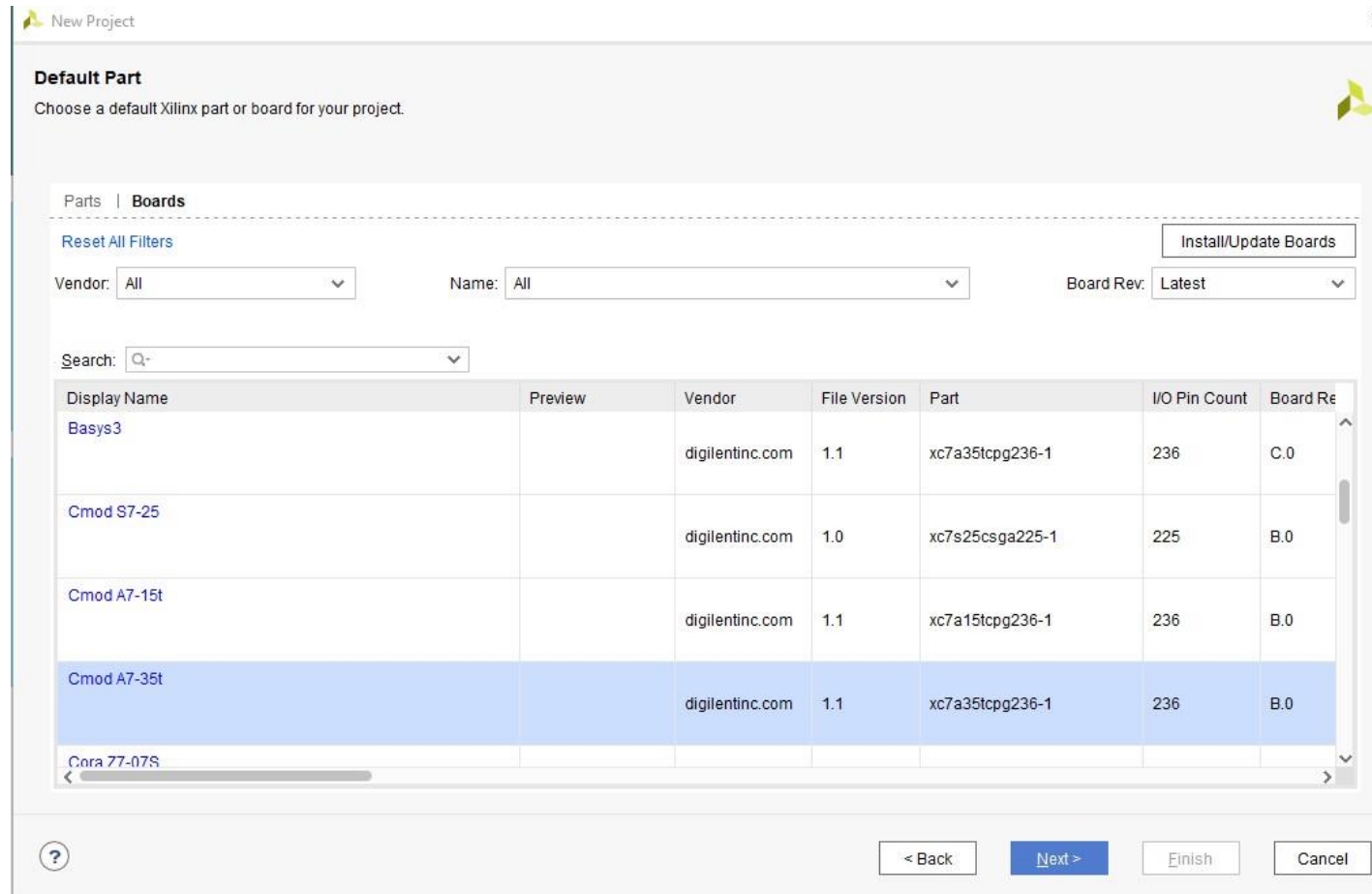


Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante



Si les boards ne sont pas présentes sous Vivado : <https://github.com/Digilent/vivado-boards>
Copier /coller le ou les dossiers dont on a besoin dans C:\Xilinx\Vivado\2020.1\data\boards
Ici c'est la board Cmod-a7-35T.



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

Fichiers sources

Informations sur le projet crée

exécution du projet

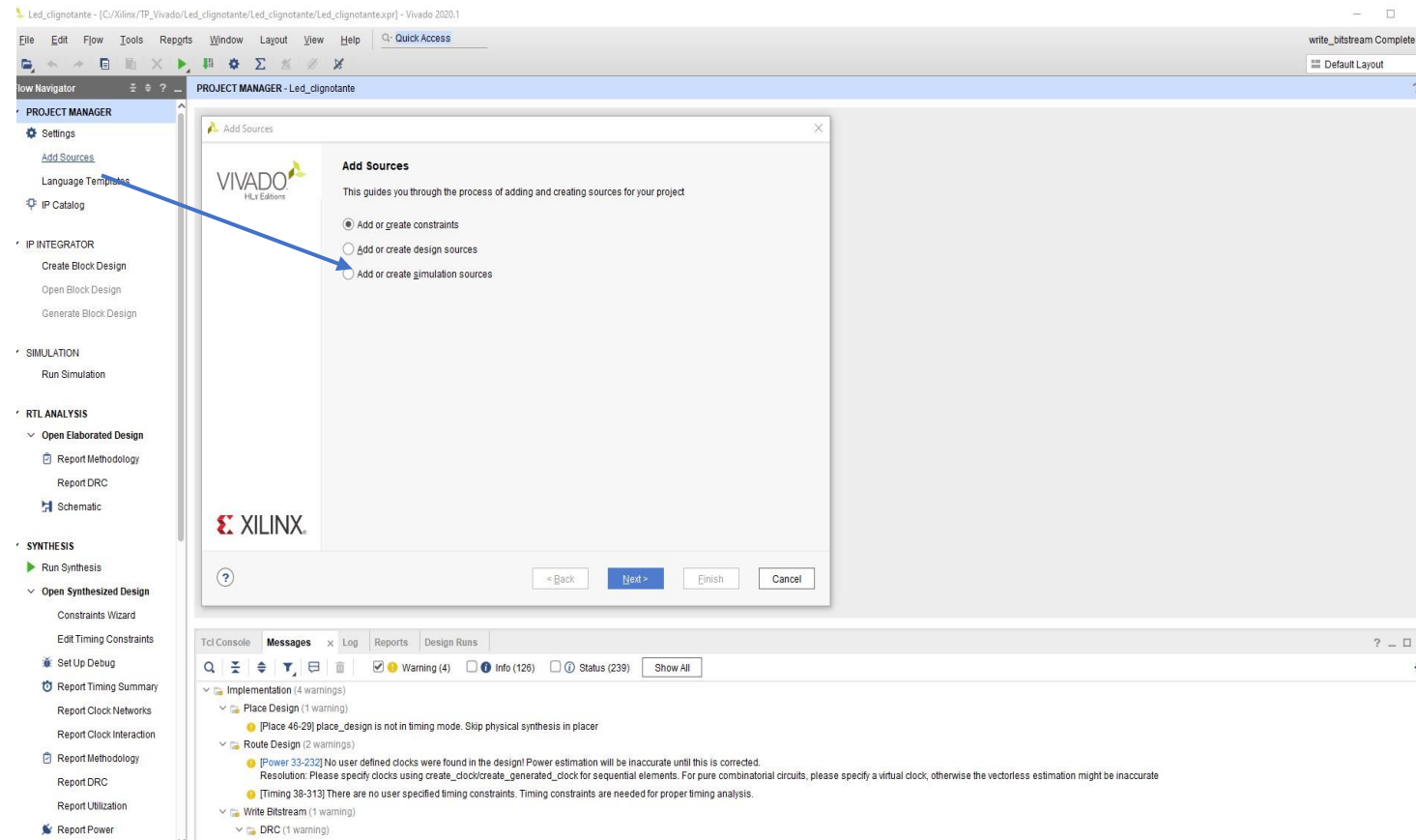
Name	Constraints	Status	WNS	TNS	WHS	THS	TP	Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy
synth_1	constrs_1	Not started															Vivado Synthesis Defaults (Vivado Synthesis 2020)	Vivado Synthesis Default Reports (Vivado Synthesis 2020)
impl_1	constrs_1	Not started															Vivado Implementation Defaults (Vivado Implementation 2020)	Vivado Implementation Default Reports (Vivado Implementation 2020)

Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

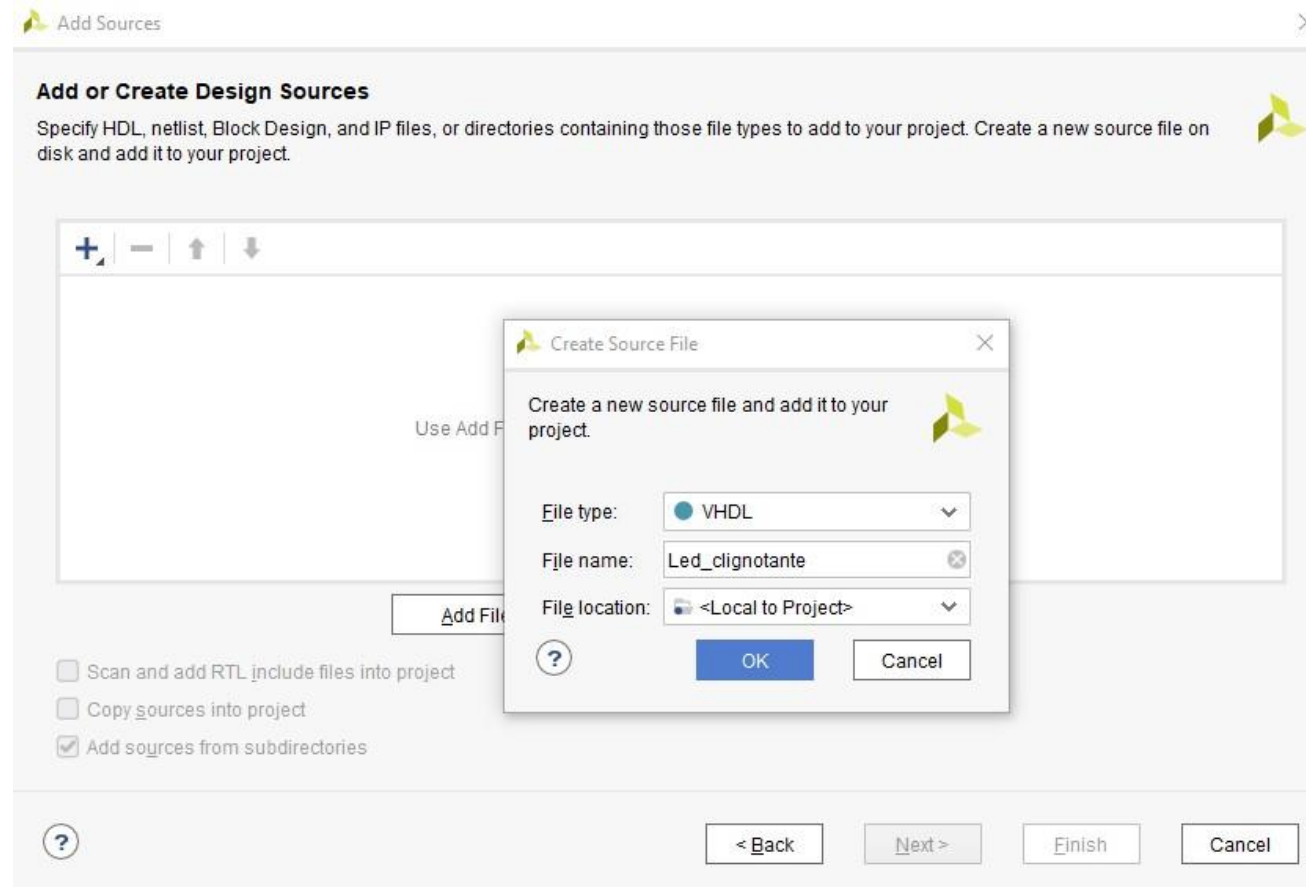


Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante



Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

Define Module

Define a module and specify I/O Ports to add to your source file.
For each port specified:
MSB and LSB values will be ignored unless its Bus column is checked.
Ports with blank names will not be written.

Module Definition

Entity name:

Architecture name:

I/O Port Definitions

+ - ↑ ↓

Port Name	Direction	Bus	MSB	LSB
clk	in	<input type="checkbox"/>	0	0
BTN0	in	<input type="checkbox"/>	0	0
LED0	out	<input type="checkbox"/>	0	0
	in	<input type="checkbox"/>	0	0

? OK Cancel

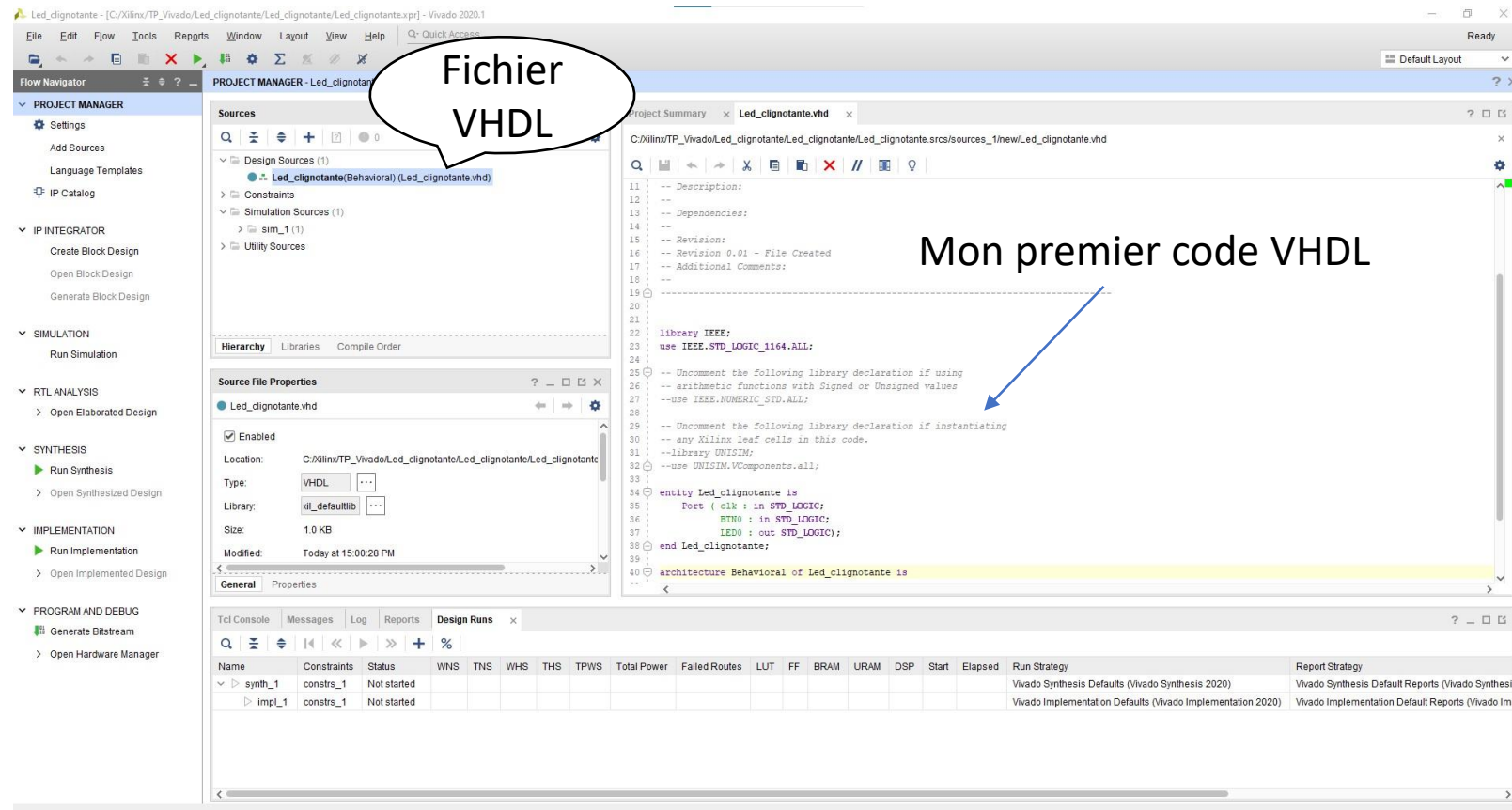
Définir les
entrées -
sorties

Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante



Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

entité

```
1 library IEEE; -- Déclaration des bibliothèques utilisées
2 use IEEE.STD_LOGIC_1164.ALL; -- inclus signaux améliorés du type std_ulogic--
3 --ce code VHDL est un code de test pouru la CmodA7 35t--
4 entity Led_clignotante is
5     Port ( clk : in STD_LOGIC; --12MHz
6           BTNO : in STD_LOGIC;
7           LED1 : out STD_LOGIC;
8           LED2 : out STD_LOGIC);
9 end Led_clignotante;
```

architecture

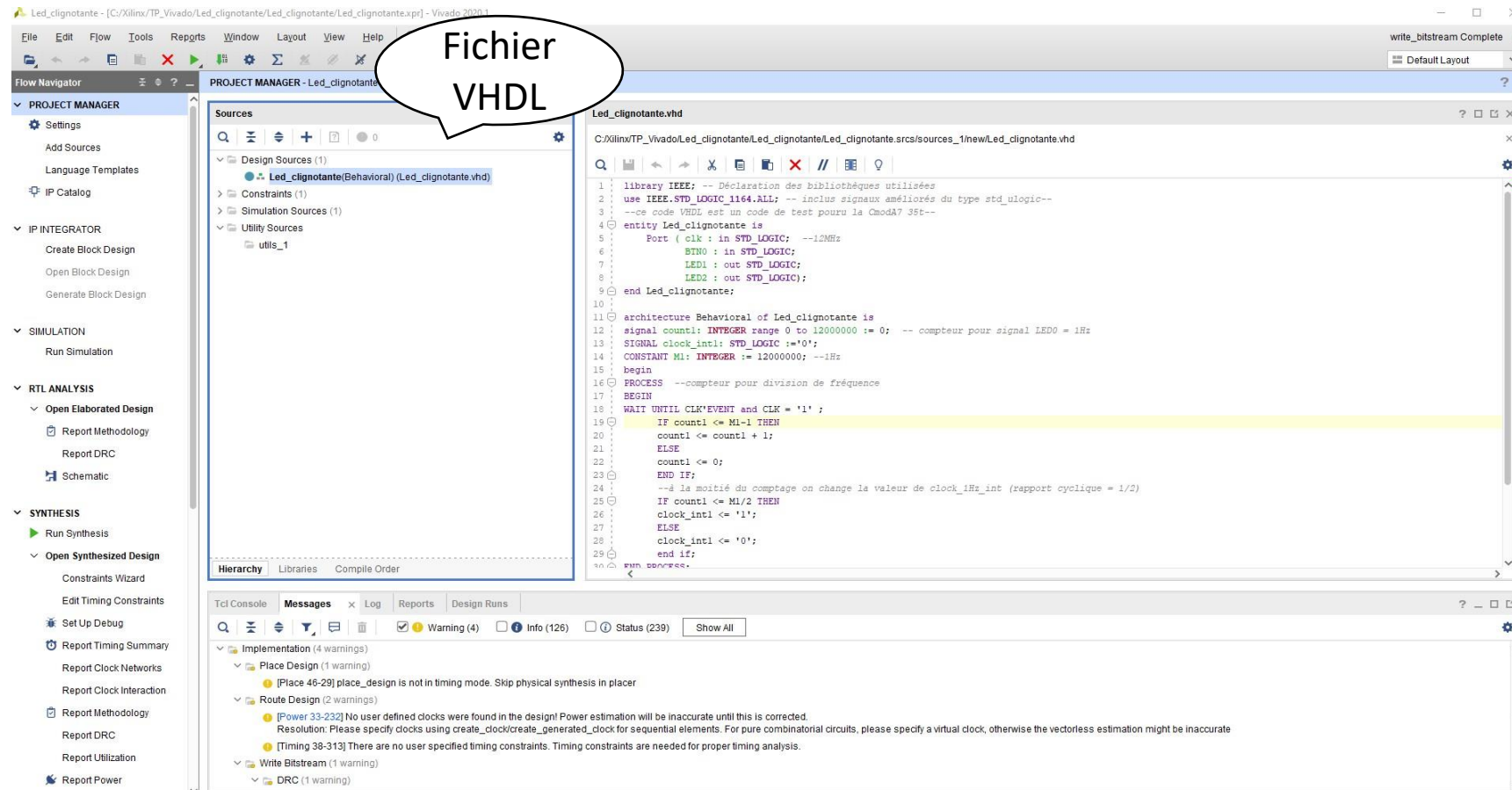
```
10
11 architecture Behavioral of Led_clignotante is
12     signal count1: INTEGER range 0 to 12000000 := 0; -- compteur pour signal LED0 = 1Hz
13     SIGNAL clock_int1: STD_LOGIC := '0';
14     CONSTANT M1: INTEGER := 12000000; --1Hz
15     begin
16     PROCESS --compteur pour division de fréquence
17     BEGIN
18     WAIT UNTIL CLK'EVENT and CLK = '1' ;
19     IF count1 <= M1-1 THEN
20     count1 <= count1 + 1;
21     ELSE
22     count1 <= 0;
23     END IF;
24     --à la moitié du comptage on change la valeur de clock_1Hz_int (rapport cyclique = 1/2)
25     IF count1 <= M1/2 THEN
26     clock_int1 <= '1';
27     ELSE
28     clock_int1 <= '0';
29     end if;
30     END PROCESS;
31     -- on affecte le signal issu du diviseur d'horloge dans la sortie LED0 la LED clignote à 1Hz
32     LED1 <= clock_int1;
33     -- la LED0 passa à 1 en focntion de BTNO
34     LED2<='1' when BTNO ='1' else '0';
35 end Behavioral;
```

Code VHDL complété



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

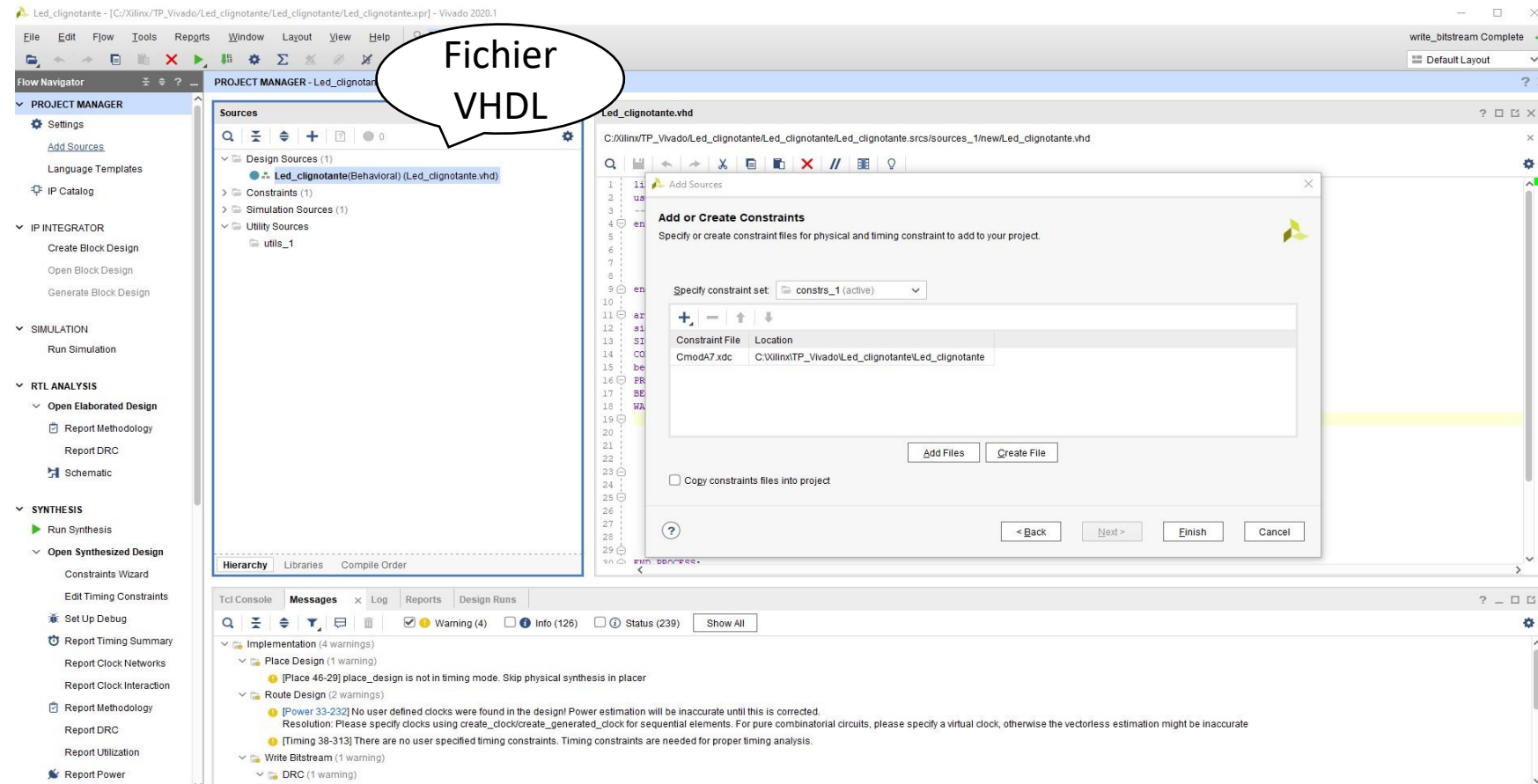


Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante



Présentation de Vivado 2020.1 : démonstration

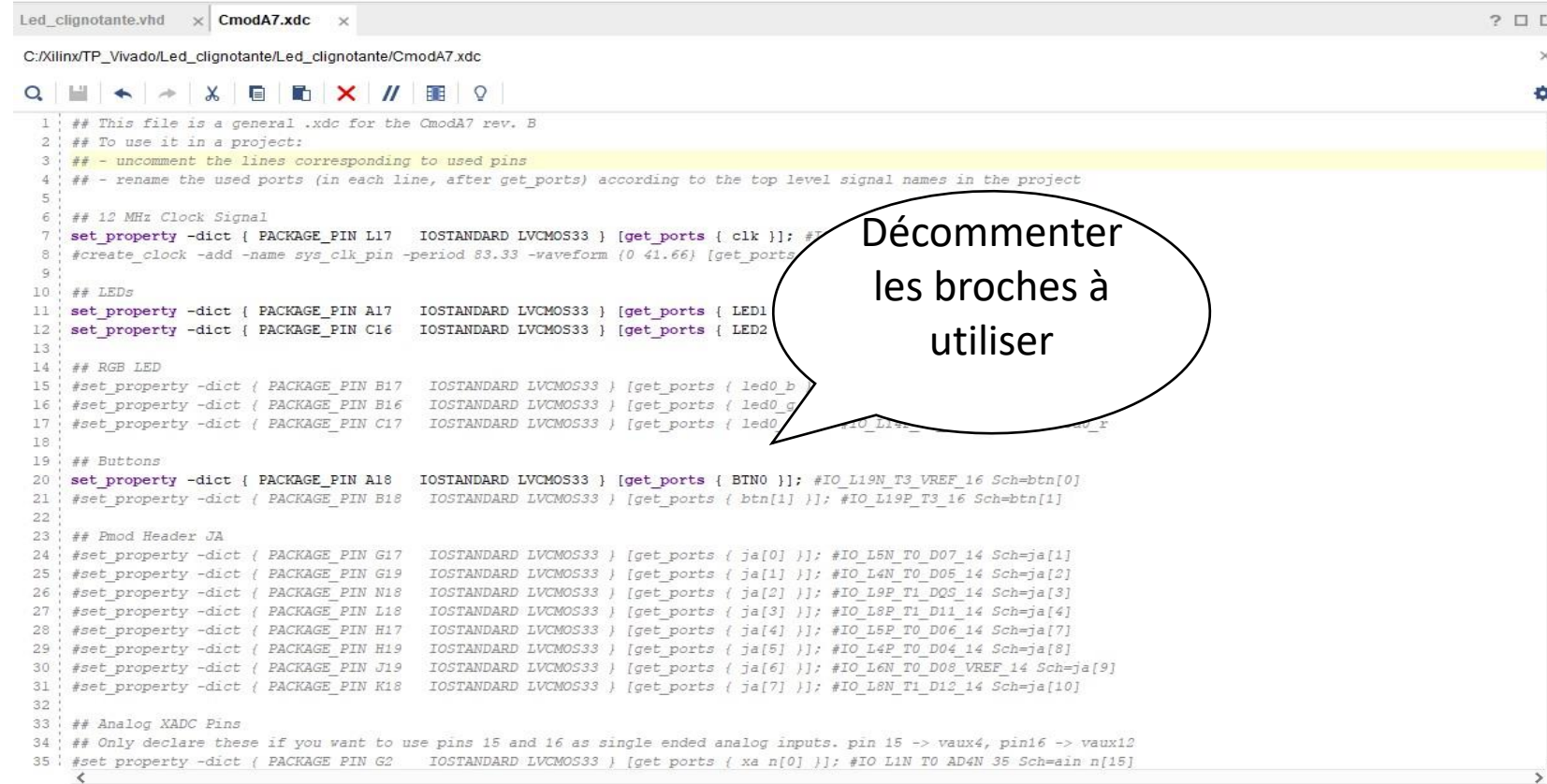
Ajout du fichier de contrainte



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

Récupérer le fichier de contrainte de la carte sur : <https://github.com/Digilent/digilent-xdc/>



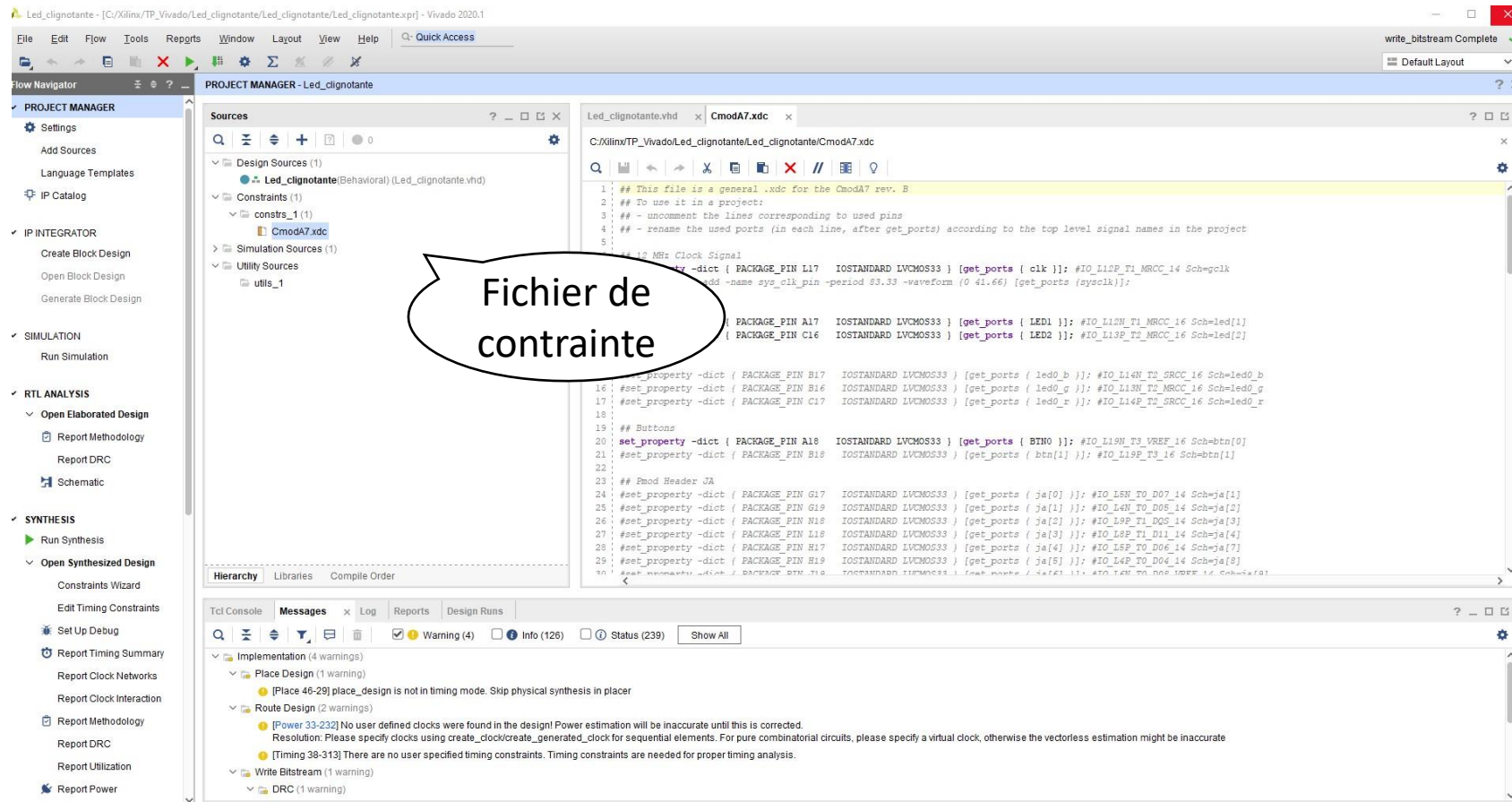
```
1 ## This file is a general .xdc for the CmodA7 rev. B
2 ## To use it in a project:
3 ## - uncomment the lines corresponding to used pins
4 ## - rename the used ports (in each line, after get_ports) according to the top level signal names in the project
5
6 ## 12 MHz Clock Signal
7 set_property -dict { PACKAGE_PIN L17 IOSTANDARD LVCMOS33 } [get_ports { clk }]; #
8 #create_clock -add -name sys_clk_pin -period 83.33 -waveform {0 41.66} [get_ports {
9
10 ## LEDs
11 set_property -dict { PACKAGE_PIN A17 IOSTANDARD LVCMOS33 } [get_ports { LED1
12 set_property -dict { PACKAGE_PIN C16 IOSTANDARD LVCMOS33 } [get_ports { LED2
13
14 ## RGB LED
15 #set_property -dict { PACKAGE_PIN B17 IOSTANDARD LVCMOS33 } [get_ports { led0_b
16 #set_property -dict { PACKAGE_PIN B16 IOSTANDARD LVCMOS33 } [get_ports { led0_g
17 #set_property -dict { PACKAGE_PIN C17 IOSTANDARD LVCMOS33 } [get_ports { led0_r
18
19 ## Buttons
20 set_property -dict { PACKAGE_PIN A18 IOSTANDARD LVCMOS33 } [get_ports { BTN0 }]; #IO_L19N_T3_VREF_16 Sch=btn[0]
21 #set_property -dict { PACKAGE_PIN B18 IOSTANDARD LVCMOS33 } [get_ports { btn[1] }]; #IO_L19P_T3_16 Sch=btn[1]
22
23 ## Pmod Header JA
24 #set_property -dict { PACKAGE_PIN G17 IOSTANDARD LVCMOS33 } [get_ports { ja[0] }]; #IO_L5N_T0_D07_14 Sch=ja[1]
25 #set_property -dict { PACKAGE_PIN G19 IOSTANDARD LVCMOS33 } [get_ports { ja[1] }]; #IO_L4N_T0_D05_14 Sch=ja[2]
26 #set_property -dict { PACKAGE_PIN N18 IOSTANDARD LVCMOS33 } [get_ports { ja[2] }]; #IO_L9P_T1_DQ5_14 Sch=ja[3]
27 #set_property -dict { PACKAGE_PIN L18 IOSTANDARD LVCMOS33 } [get_ports { ja[3] }]; #IO_L8P_T1_D11_14 Sch=ja[4]
28 #set_property -dict { PACKAGE_PIN H17 IOSTANDARD LVCMOS33 } [get_ports { ja[4] }]; #IO_L5P_T0_D06_14 Sch=ja[7]
29 #set_property -dict { PACKAGE_PIN H19 IOSTANDARD LVCMOS33 } [get_ports { ja[5] }]; #IO_L4P_T0_D04_14 Sch=ja[8]
30 #set_property -dict { PACKAGE_PIN J19 IOSTANDARD LVCMOS33 } [get_ports { ja[6] }]; #IO_L6N_T0_D08_VREF_14 Sch=ja[9]
31 #set_property -dict { PACKAGE_PIN K18 IOSTANDARD LVCMOS33 } [get_ports { ja[7] }]; #IO_L8N_T1_D12_14 Sch=ja[10]
32
33 ## Analog XADC Pins
34 ## Only declare these if you want to use pins 15 and 16 as single ended analog inputs. pin 15 -> vaux4, pin16 -> vaux12
35 #set_property -dict { PACKAGE_PIN G2 IOSTANDARD LVCMOS33 } [get_ports { xa n[0] }]; #IO_L1N_T0_AD4N_35 Sch=ain n[15]
```

Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante



Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

The screenshot shows the Vivado IDE interface. The 'Add Sources' dialog is open, guiding the user through adding sources. The 'Add or create simulation sources' option is selected. The background shows the 'Sources' panel with the project 'Led_clignotante' and the 'Design Runs' table.

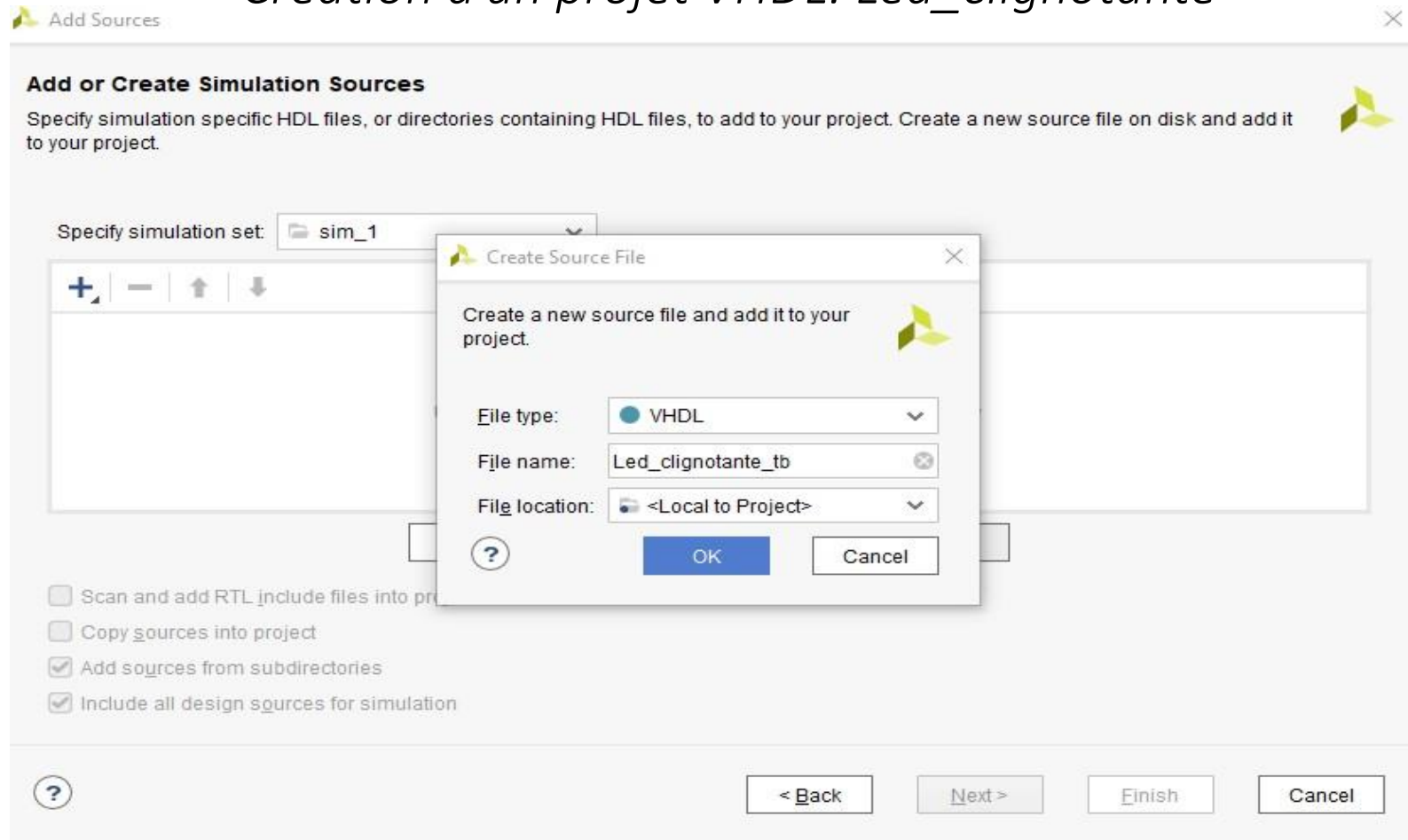
Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy
✓ synth_1	constrs_1	synth_design Complete!								12	25	0.0	0	0	3/8/22, 4:26 PM	00:00:41	Vivado Synthesis Defaults (Vivado Synthesis 2020)	Vivado Synthesis De
✓ impl_1	constrs_1	write_bitstream Complete!	NA	NA	NA	NA	NA	1.861	0	12	25	0.0	0	0	3/8/22, 4:27 PM	00:01:28	Vivado Implementation Defaults (Vivado Implementation 2020)	Vivado Implementati

Ajout d'un fichier testbench pour la simulation fonctionnelle



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante



Présentation de Vivado 2020.1 : démonstration



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

Fichier pour la simulation

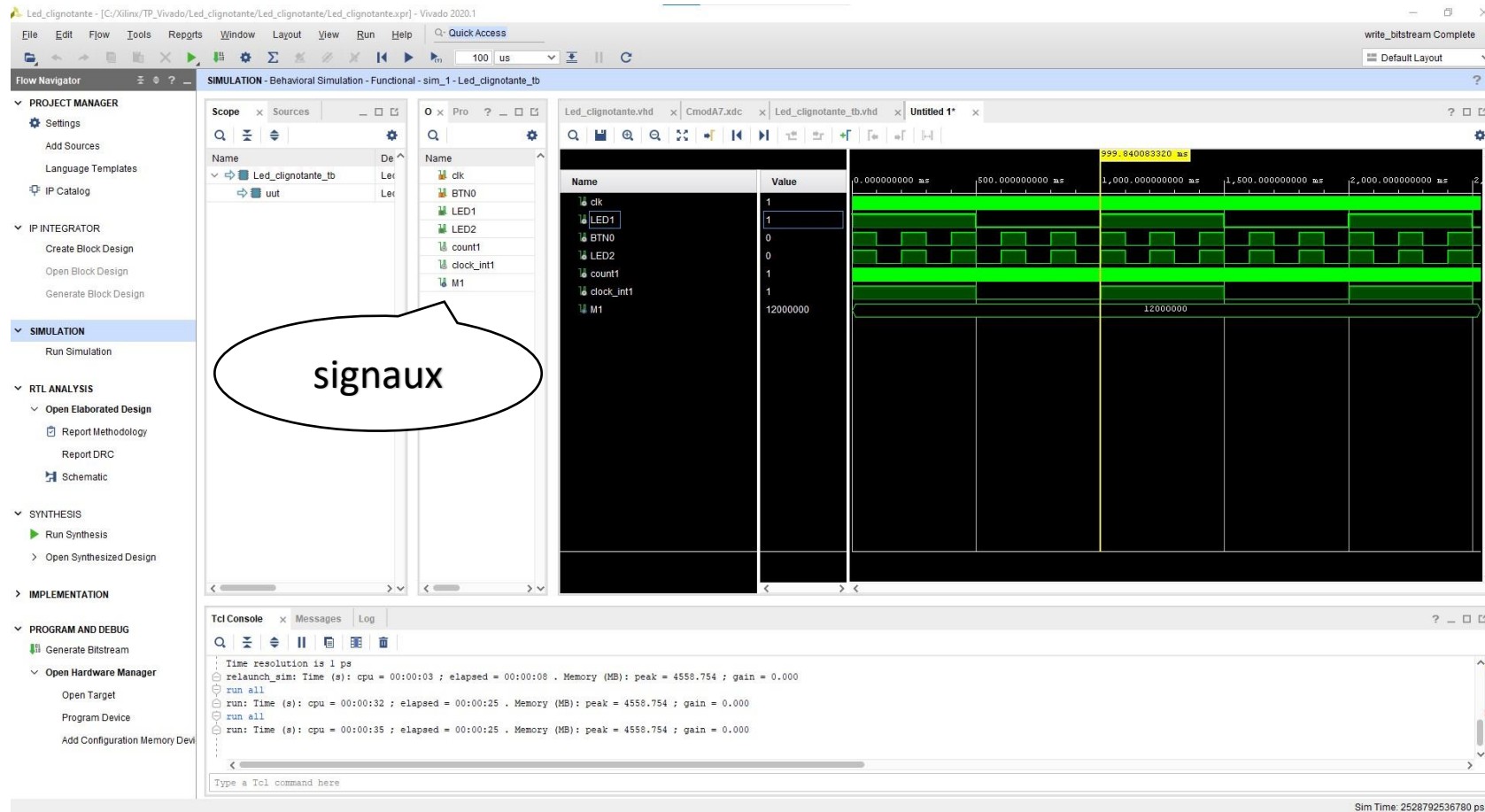
Édition du fichier simulation

Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy
synth_1	constrs_1	synth_design Complete!								12	25	0.0	0	0	3/8/22, 4:26 PM	00:00:41	Vivado Synthesis Defaults (Vivado Synthesis 2020)	Vivado Synthesis De
impl_1	constrs_1	write_bitstream Complete!	NA	NA	NA	NA	NA	1.861	0	12	25	0.0	0	0	3/8/22, 4:27 PM	00:01:28	Vivado Implementation Defaults (Vivado Implementation 2020)	Vivado Implementati

Il faut éditer le fichier et déclarer le composants Led_clignotante et générer des stimuli

Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante



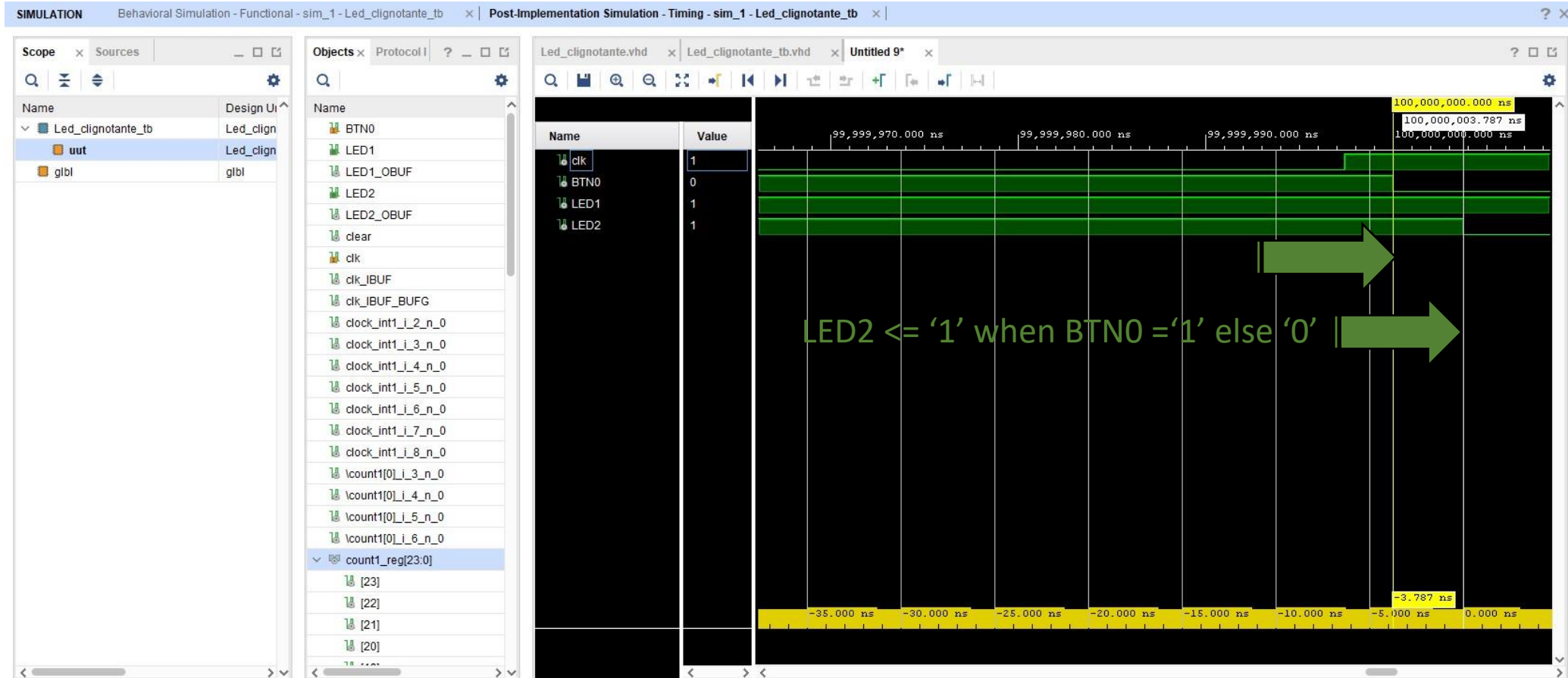
Simulation fonctionnelle du code VHDL Led_clignotante
Run Behavioral Simulation



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

Run Post-Implementation Timing Simulation



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

The screenshot displays the Vivado 2020.1 interface for a project named 'Led_clignotante'. The left-hand 'Flow Navigator' pane shows the project workflow, with 'RTL ANALYSIS' selected. A blue arrow points from the 'Schematic' option in this pane to the main schematic window. The 'SOURCES' pane lists the project files, including 'Led_clignotante(Behavioral) (Led_clignotante.vhd)'. The central 'SCHEMATIC' window shows a high-level circuit diagram with components like 'RTL_MUX', 'RTL_ADD', 'RTL_REG_SYNC', and 'RTL_REG'. A blue arrow points from the text 'Register Transfer Level (RTL)' to this schematic. The bottom 'DESIGN RUNS' pane shows the status of the synthesis and implementation steps, both marked as complete.

Register Transfer Level (RTL)

representations of a high-level circuit

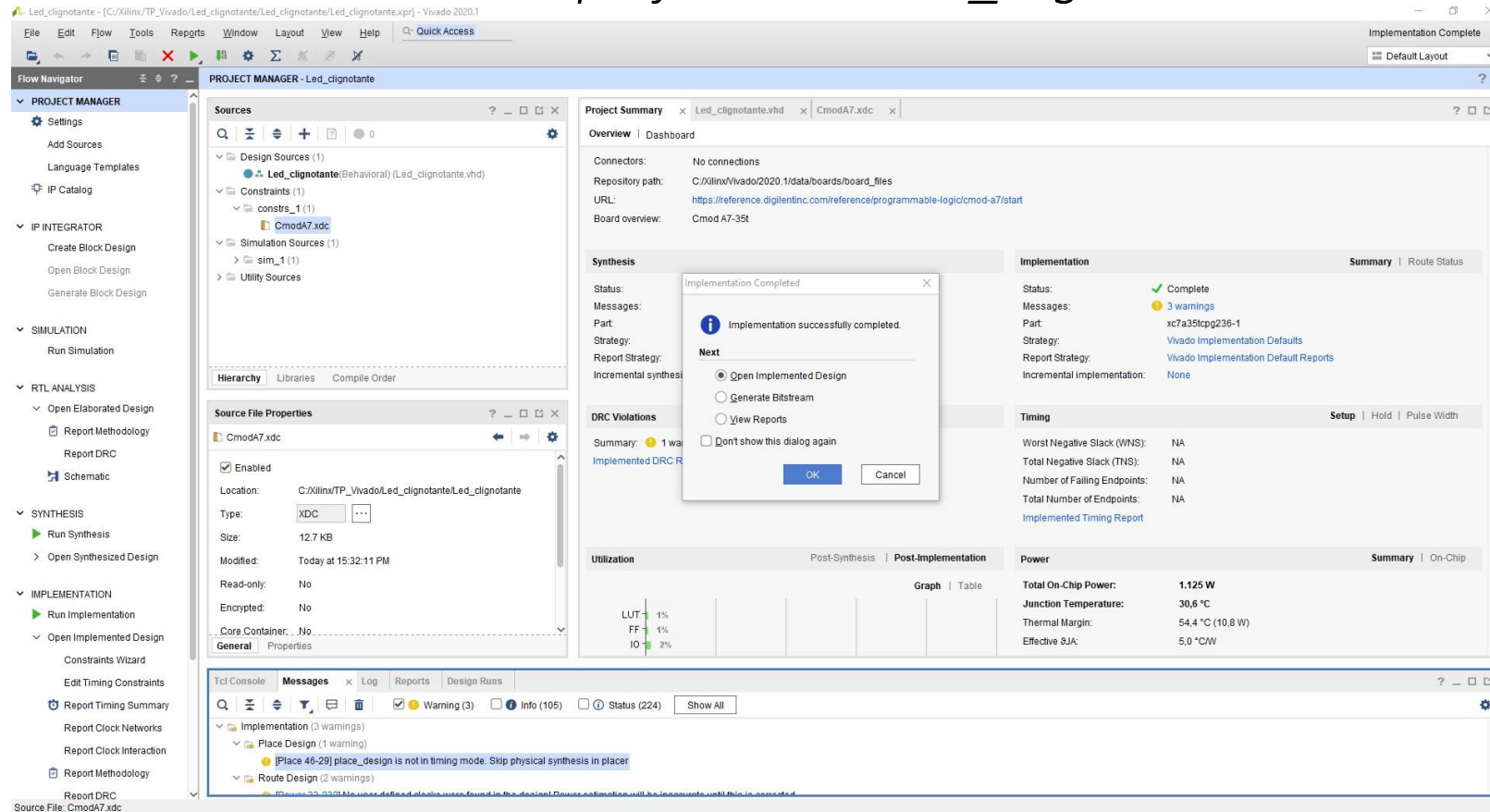
Name	Constraints	Status	WNS	TNS	WHS	THS	TPWS	Total Power	Failed Routes	LUT	FF	BRAM	URAM	DSP	Start	Elapsed	Run Strategy	Report Strategy
✓ synth_1	constrs_1	synth_design Complete!								12	25	0.0	0	0	3/8/22, 4:17 PM	00:00:37	Vivado Synthesis Defaults (Vivado Synthesis 2020)	Vivado Synthesis De
✓ impl_1	constrs_1	write_bitstream Complete!	NA	NA	NA	NA	NA	1.861	0	12	25	0.0	0	0	3/8/22, 4:17 PM	00:01:27	Vivado Implementation Defaults (Vivado Implementation 2020)	Vivado Implementati

RTL analysis Schematic



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante



Run Implementation and generate Bitstream



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

The screenshot displays the Vivado IDE for a project named 'Led_clignotante'. The Project Manager on the left shows the project structure, with the 'IMPLEMENTATION' tab selected. The central Schematic view shows a complex circuit diagram with various logic blocks and interconnections. The Design Timing Summary at the bottom provides a summary of the implementation results.

Design Timing Summary

Setup	Hold	Pulse Width
Worst Negative Slack (WNS): NA	Worst Hold Slack (WHS): NA	Worst Pulse Width Slack (WPWS): NA
Total Negative Slack (TNS): NA	Total Hold Slack (THS): NA	Total Pulse Width Negative Slack (TPWS): NA
Number of Falling Endpoints: NA	Number of Falling Endpoints: NA	Number of Falling Endpoints: NA
Total Number of Endpoints: NA	Total Number of Endpoints: NA	Total Number of Endpoints: NA

There are no user specified timing constraints.

Implementation Schematic



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante

The screenshot displays the Vivado 2020.1 interface for a project named 'Led_clignotante'. The left sidebar shows the 'IMPLEMENTATION' tab selected. The central area shows the 'Package Pins' table, which lists the pins of the device and their configuration. A speech bubble points to the pin grid, indicating the connection of the FPGA to the periphery.

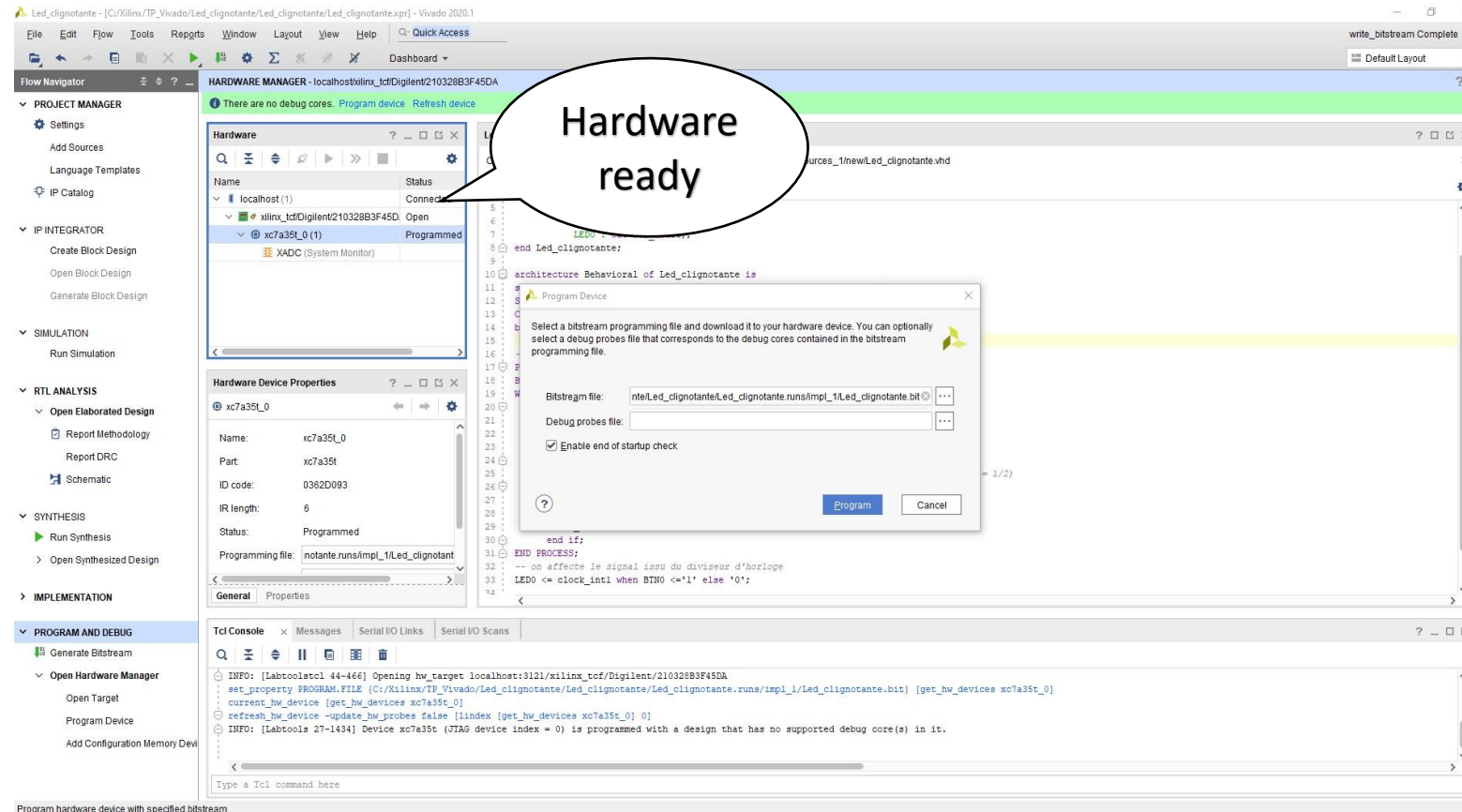
Name	Available	Prohibit	Ports	I/O Std	Dir	Vcco	Bank	Bank Type	Byte Group	Type	Diff Pair	Clock	Voltage	Config	XADC	Gigabit I/O	MCB	PCI	Min Trace Dly (ps)	Max Ti
A16	1	<input type="checkbox"/>					I/O Bank 16	HIGH_RANGE		Multi-R	L12P	MRCC							52.5	
A17	0	<input type="checkbox"/>	LED0	LVCNMOS33*	Outp	3.300	I/O Bank 16	HIGH_RANGE		Multi-R	L12N	MRCC							48.79	
C16	1	<input type="checkbox"/>					I/O Bank 16	HIGH_RANGE		Multi-R	L13P	MRCC							33.17	
B16	1	<input type="checkbox"/>					I/O Bank 16	HIGH_RANGE		Multi-R	L13N	MRCC							44.01	
C17	1	<input type="checkbox"/>					I/O Bank 16	HIGH_RANGE		Multi-R	L14P	SRCC							44.1	
B17	1	<input type="checkbox"/>					I/O Bank 16	HIGH_RANGE		Multi-R	L14N	SRCC							47.59	
B18	1	<input type="checkbox"/>					I/O Bank 16	HIGH_RANGE		User I/O	L19P								51.78	
A18	0	<input type="checkbox"/>	BTN0	LVCNMOS33*	Input	3.300	I/O Bank 16	HIGH_RANGE		Multi-R	L19N		VREF						58.48	

Open Implementation



Mise en œuvre de Vivado avec un exemple simple

Création d'un projet VHDL: Led_clignotante



Open Hardware Manager and program Device



Mise en œuvre de Vivado avec un exemple simple

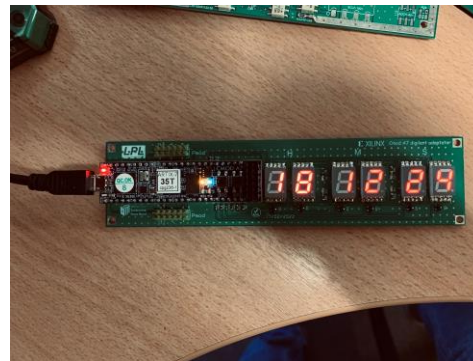
Création d'un projet VHDL: Led_clignotante

Vidéo démo VHDL simple

[https://github.com/fabzz60/Cmod-A7-Horloge-Demo-
/blob/main/%5Bfreemake.com%20LOGO%5D%20Led clignotante cm0dA7.avi](https://github.com/fabzz60/Cmod-A7-Horloge-Demo/blob/main/%5Bfreemake.com%20LOGO%5D%20Led%20clignotante%20cm0dA7.avi)

Démo Horloge numérique et chronomètre

[https://github.com/fabzz60/Cmod-A7-Horloge-Demo-](https://github.com/fabzz60/Cmod-A7-Horloge-Demo)
<https://github.com/fabzz60/Cmod-A7-Stopwatch-Demo-10s>



Fin de la démonstration

Merci de votre attention!