Mise en œuvre d'une liaison série V24
Sur la carte ZYBO

1) Problématique

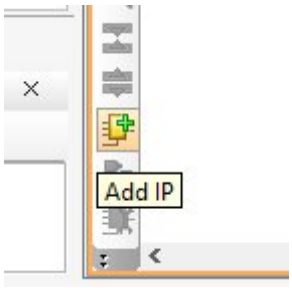
Pour des applications connectées la voie royale reste l'ETHERNET et les technologies IP. Dans certains cas mode dégradé ou simple connexion par exemple, une liaison série asynchrone V24 suffit largement. Pour recevoir ou transmettre en mode V24 une UART est nécessaire. Cette séance nous permettra de configurer dans la partie PS du ZYNQ une UART. On ne se servira pas de la logique programmable, le fichier bitstream nous sera d'aucune utilité

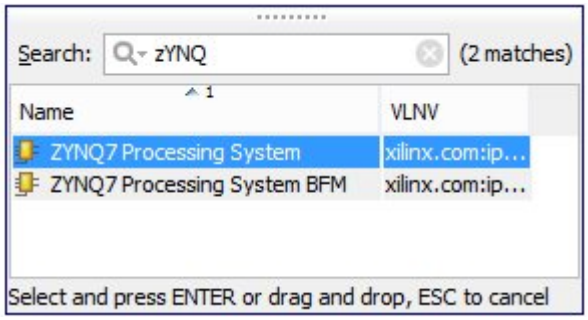
2) Création d'un projet sous Vivado

Cette étape reste identique au tutorial précédent et n'apporte aucune nouveauté. Le fichier de contrainte peut être intégré lors de la création du projet ou un petit peu plus tard

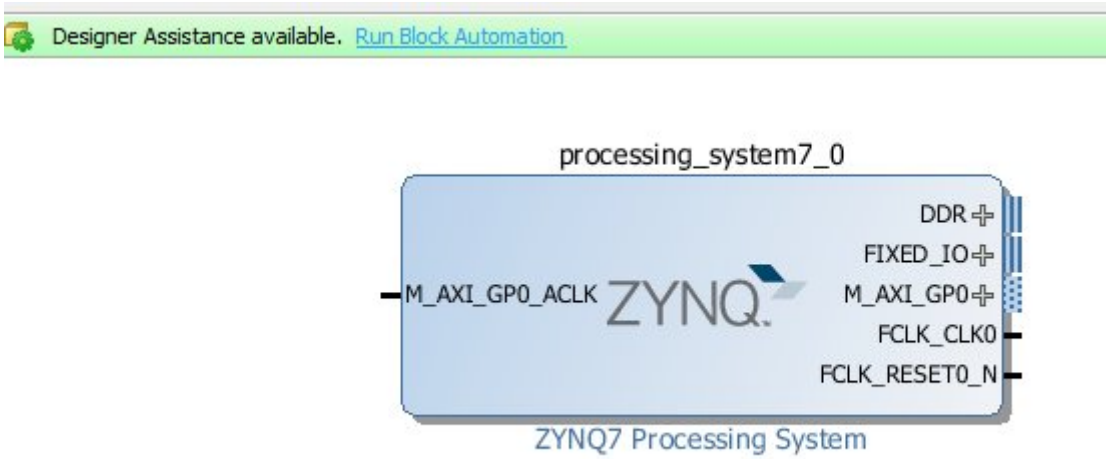
3) Elaboration du fichier bloc design

Comme d'habitude on intégrera le PS par l'intermédiaire du ZYNQ7

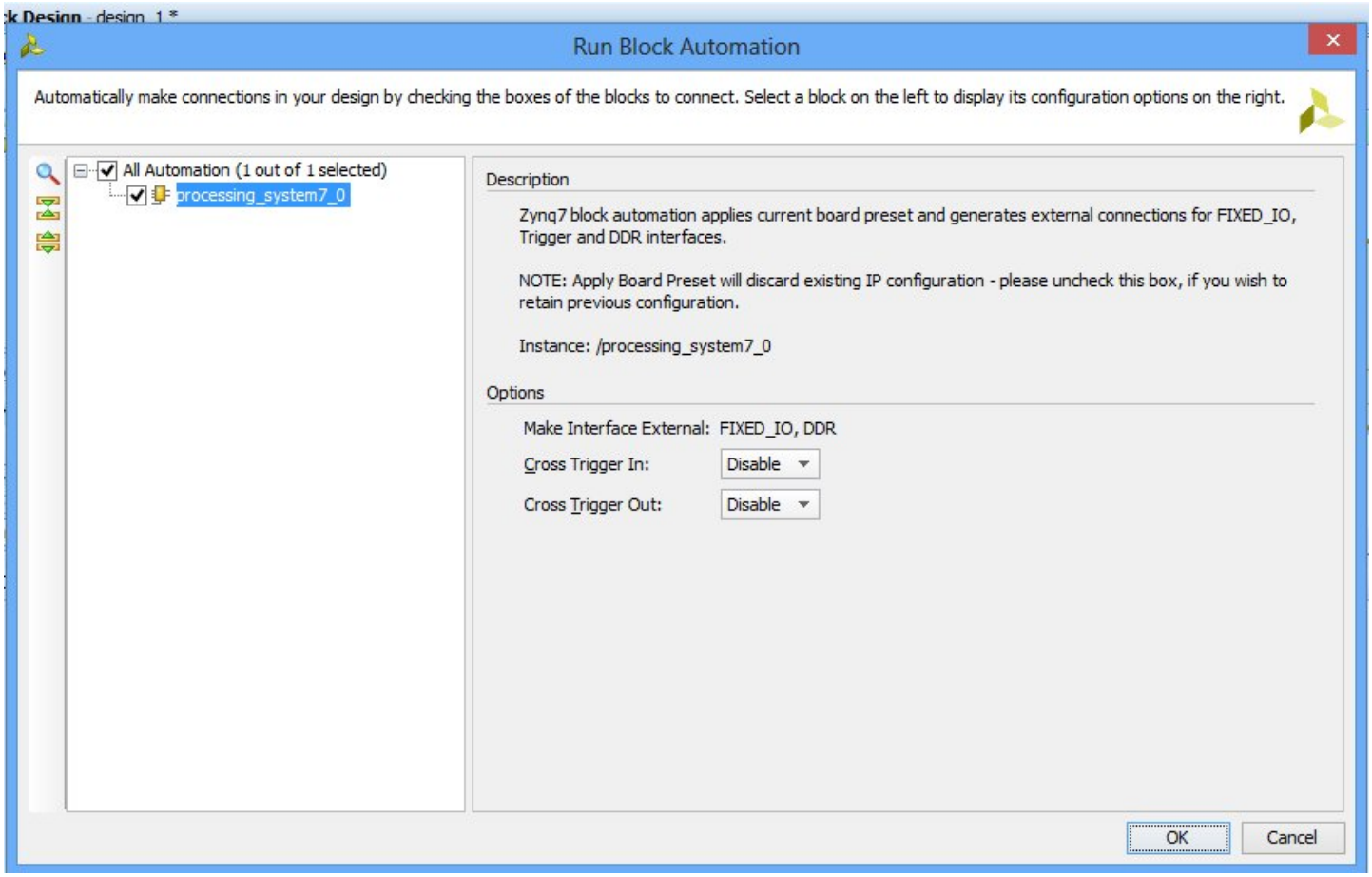




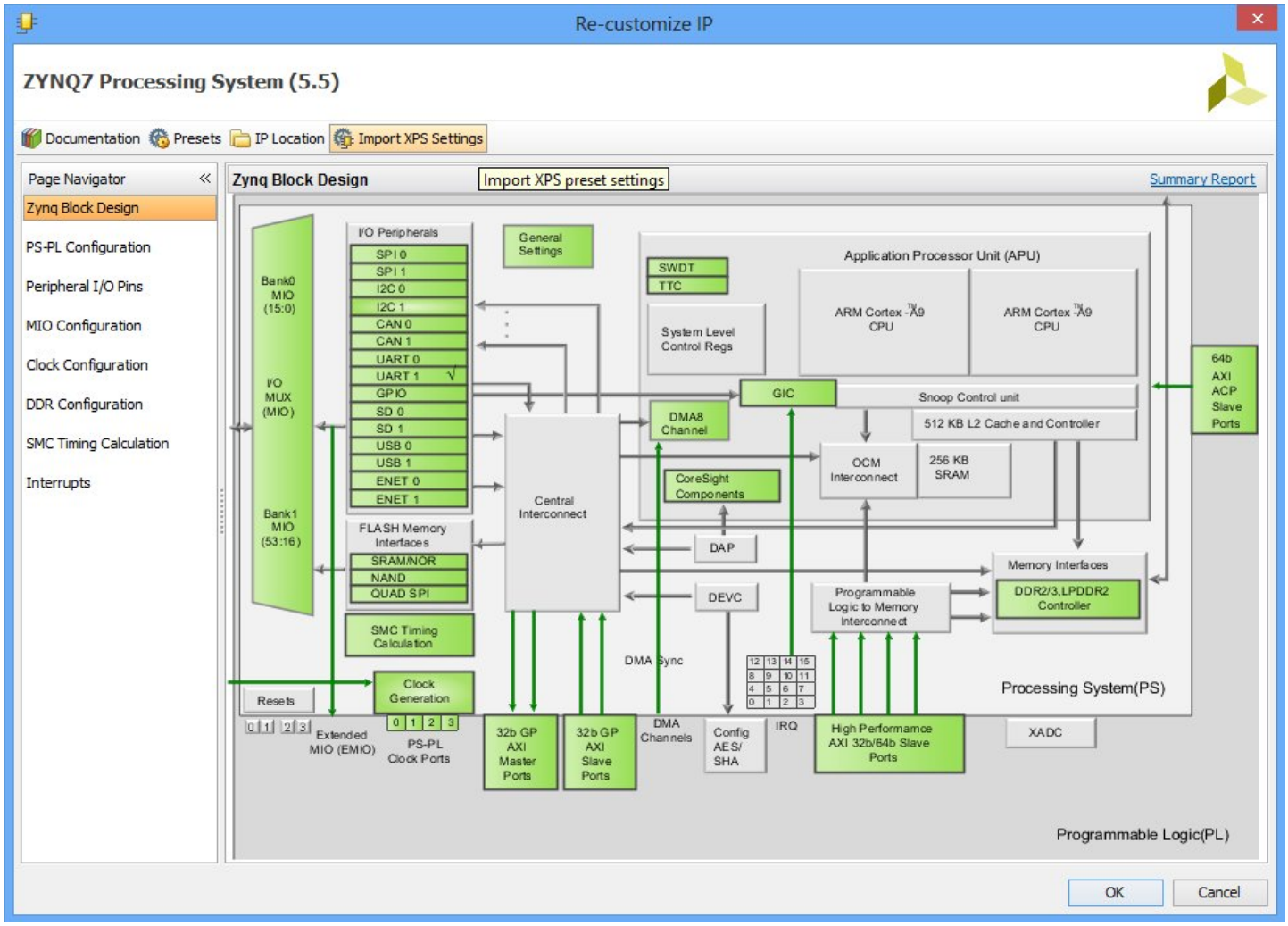
Automatisez la connexion du système de traitement avec la mémoire DDR3, cliquez sur Run Block automation



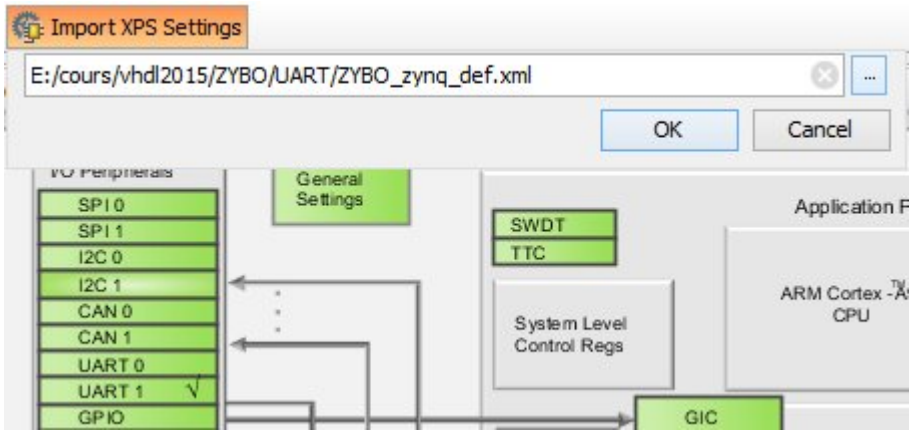
Cliquez sur yes



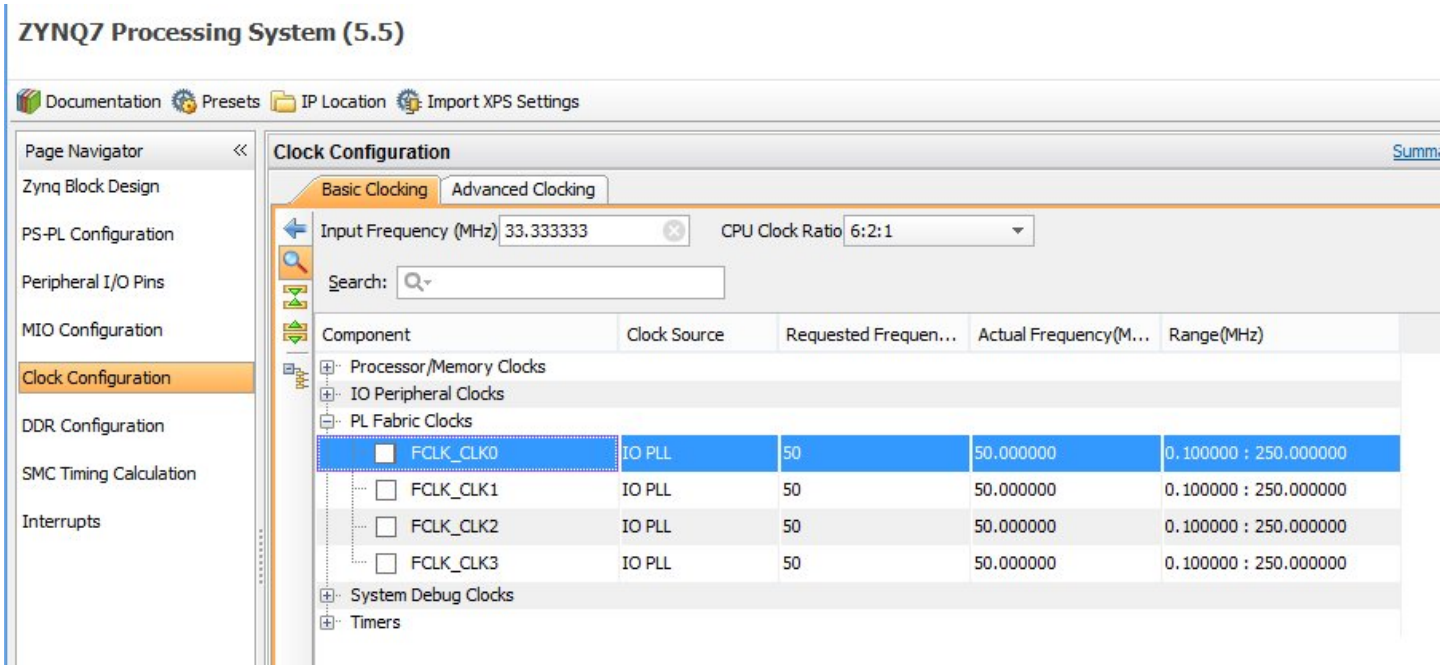
Double cliquez sur le PS ZYNQ7 dans le bloc design et puis cliquez sur import XPS preset settings



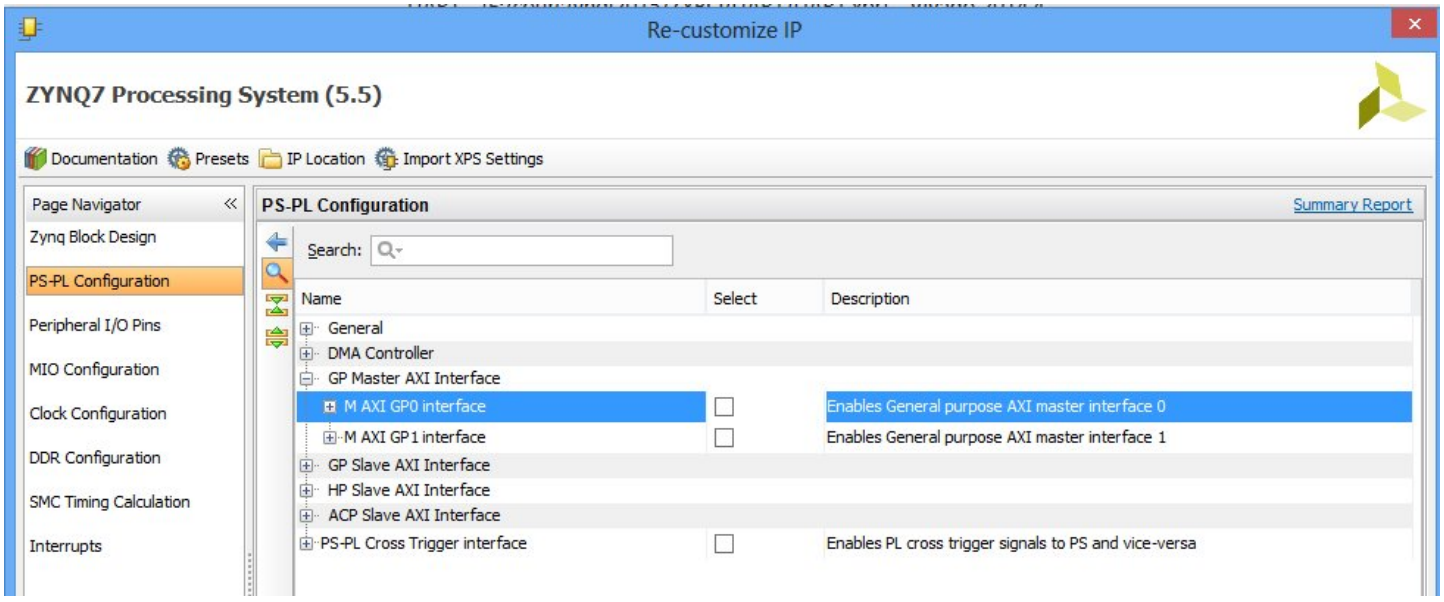
Sélectionnez le fichier ZYBO_ZYNQ_def.xml de la société Digilent



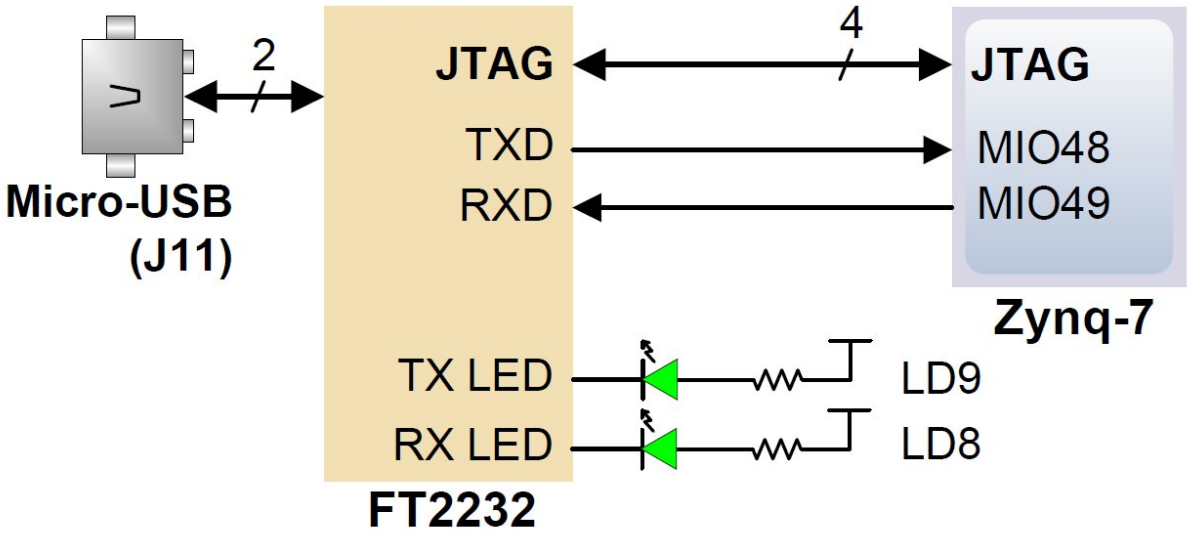
N'ayant pas de logique programmable on peut désélectionner FCLK_CLK0



Idem pour M_AXI_GPIO



La carte ZYBO inclut un pont USB_UART grâce au circuit FTDI FT2232HQ (connecteur J11) qui permet à des applications sur un PC de communiquer avec la carte en utilisant un port COM virtuel windows. Un pilote est disponible sur www.ftdichip.com afin de créer ce port virtuel et de convertir les paquets USB en paquet UART.



Le port est relié au PS par l'intermédiaire des broches MIO et est utilisé en combinaison avec le contrôleur UART1 du ZYNQ. Un translateur de niveau interface le FT2232 avec la banque MIO501.
Le fichier de définition de la carte ZYBO ZYBO_ZYNQ_def.xml permet de connecter les broches du MIO au contrôleur UART1 et utilise les paramètres de connexion par défaut suivant :

- 115200 bauds
- 1 bit de stop
- Aucune parité
- 8bits par caractère

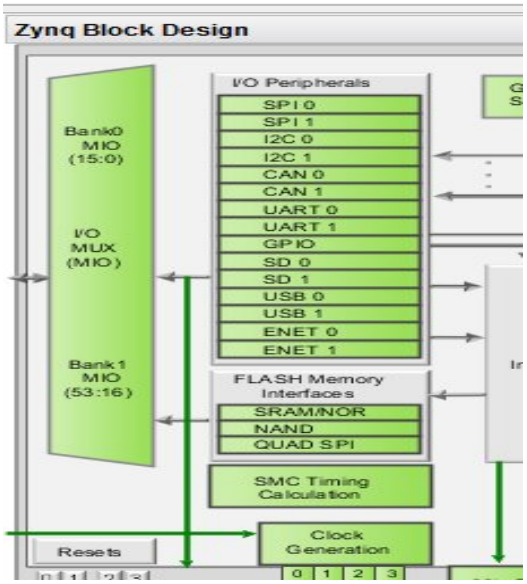
On peut observer dans le document de la carte ZYBO le branchement du module d'entrée sorties MIO par défaut et notamment l'UART lorsqu'on utilise le fichier de définition de la carte ZYBO ZYBO_ZYNQ_def.xml

1.8V

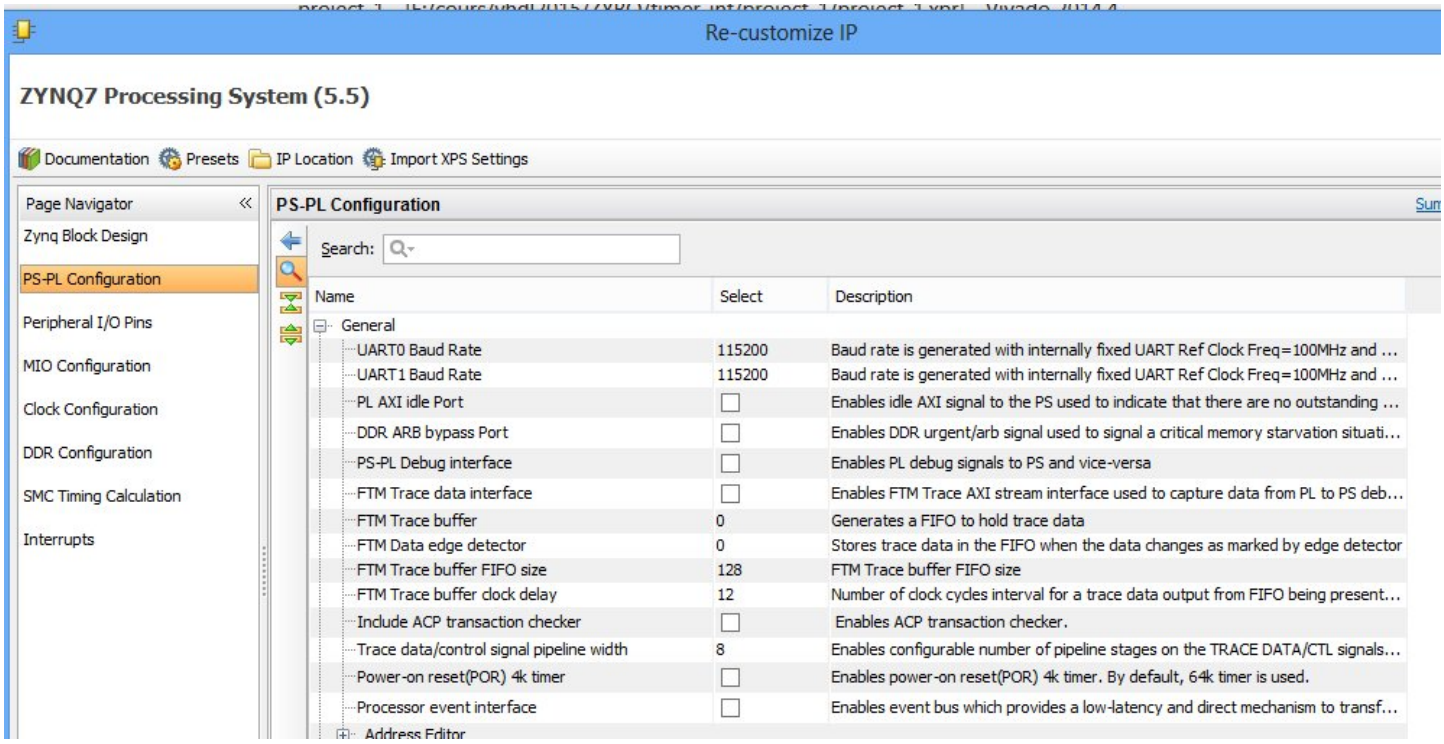
MIO 501 1.8V	Peripherals				
Pin	ENET 0	USB 0	SDIO 0	UART 1	GPIO
16	TXCK				
17	TXD0				
18	TXD1				
19	TXD2				
20	TXD3				
21	TXCTL				
22	RXCK				
23	RXD0				
24	RXD1				
25	RXD2				
26	RXD3				
27	RXCTL				
28		DATA4			
29		DIR			
30		STP			
31		NXT			
32		DATA0			
33		DATA1			
34		DATA2			
35		DATA3			
36		CLK			
37		DATA5			
38		DATA6			
39		DATA7			
40			CCLK		
41			CMD		
42			D0		
43			D1		
44			D2		
45			D3		
46		RESETN			
47			CD		
48				RXD	
49				TXD	
50					BTN4
51					BTN5
52	MDC				
53	MDIO				

Table 3. MIO Pinout. Pour le fichier de définition de la carte ZYBO (site digilent)

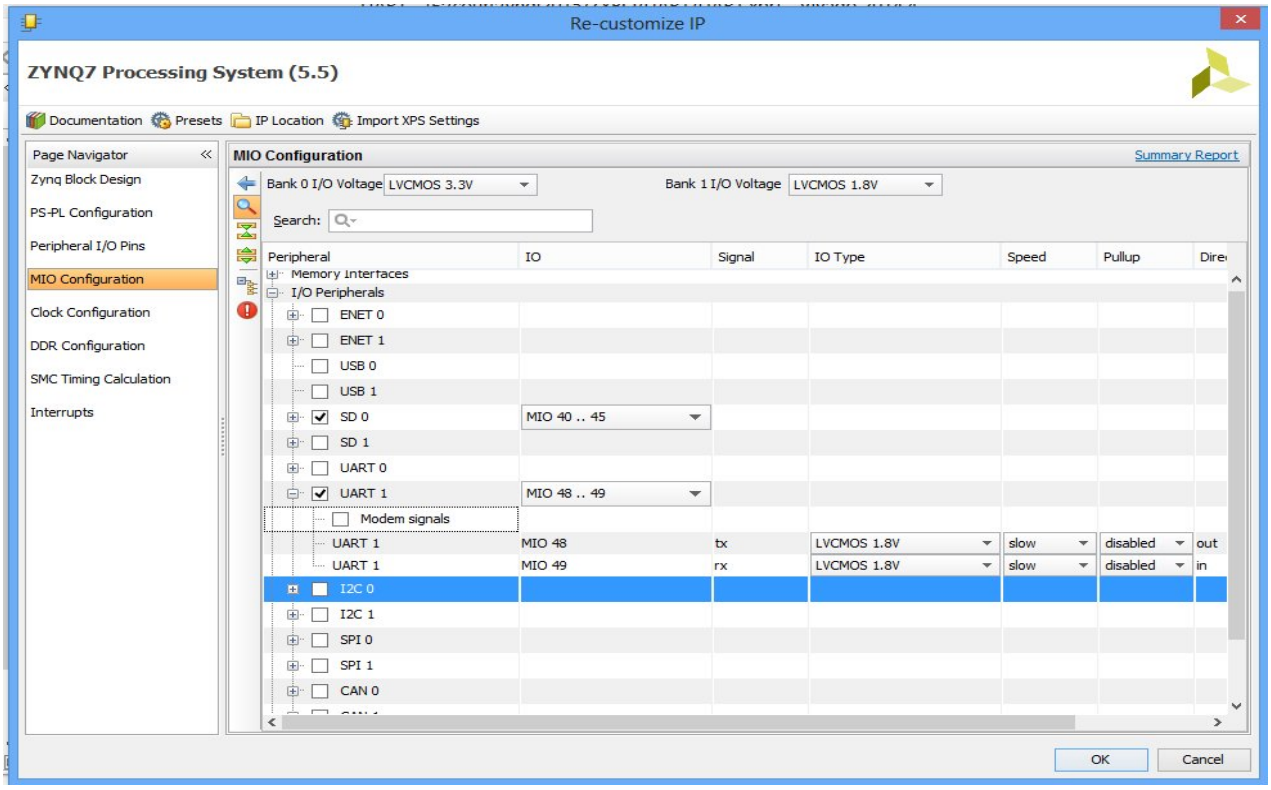
Vivado nous donne les deux banques d'entrée sorties du MIO

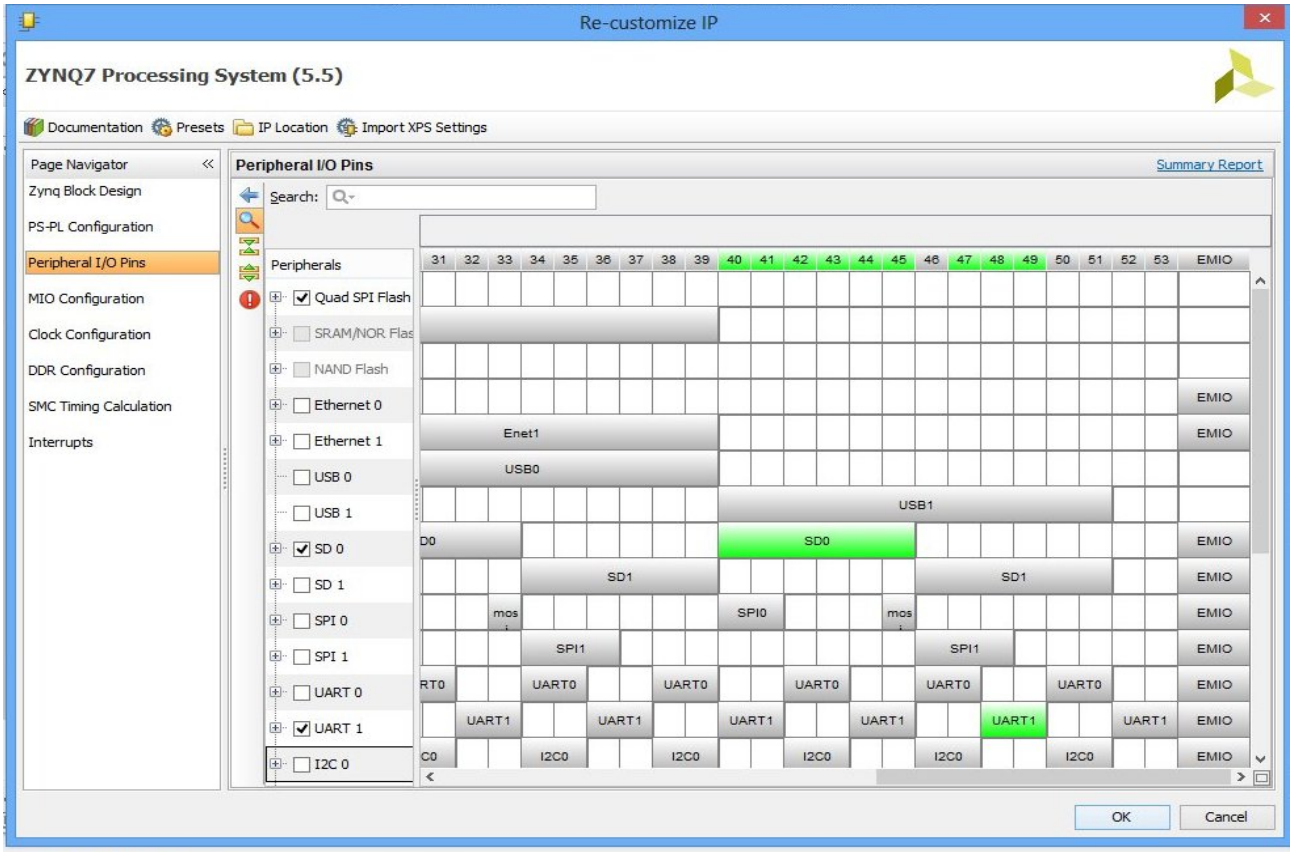


Le débit du protocole série est paramétrable sur la feuille suivante :

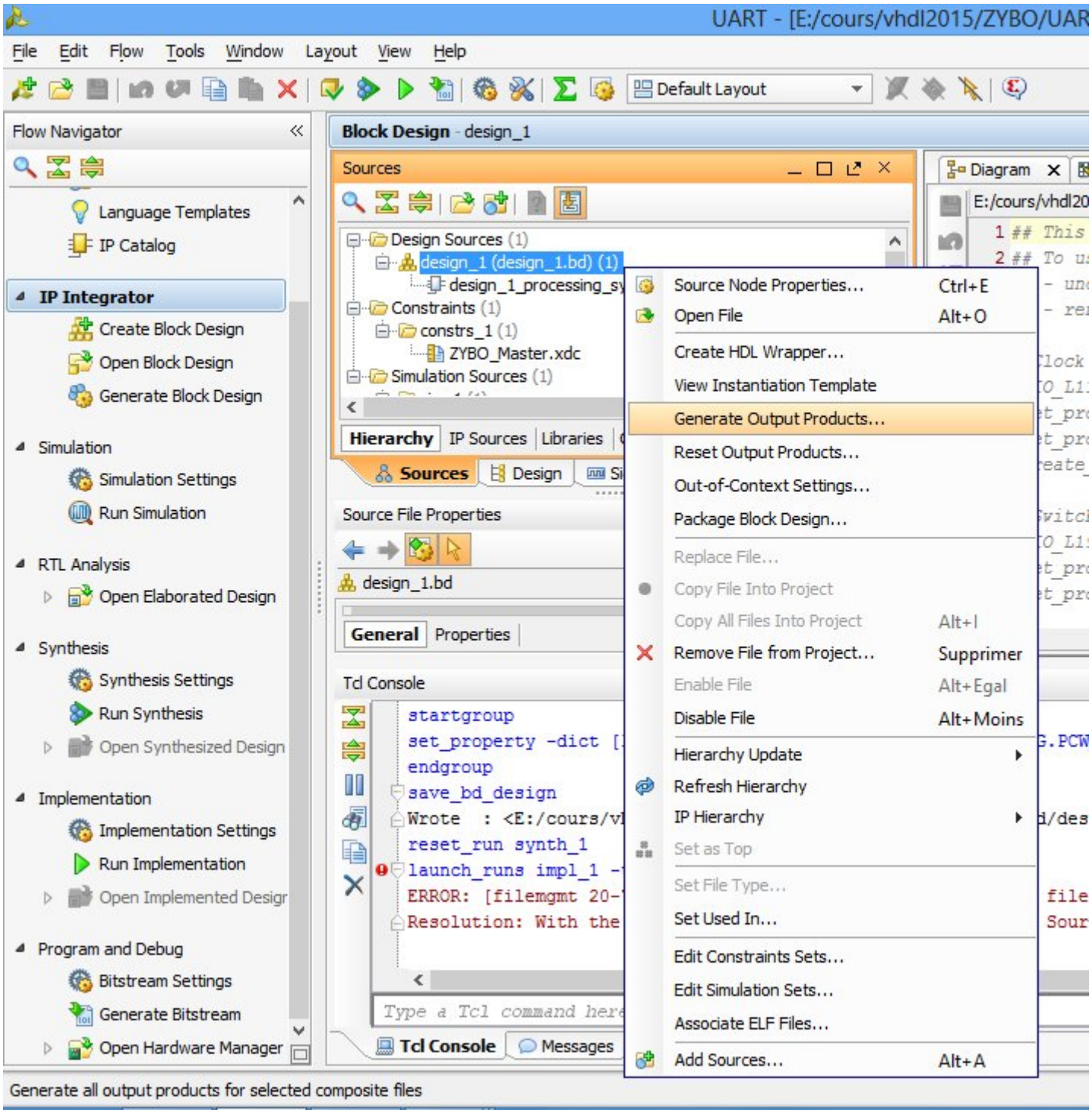


Ne gardez que la SDCARD et l'UART1 comme périphérique du PS

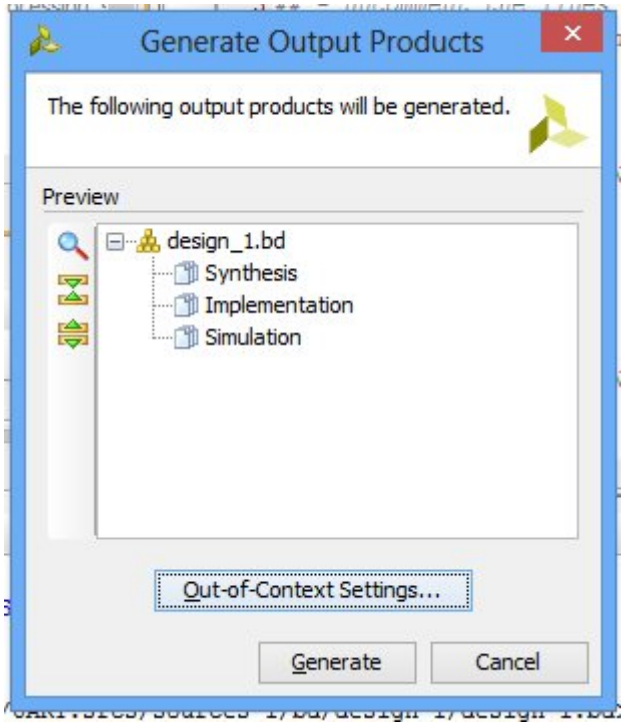




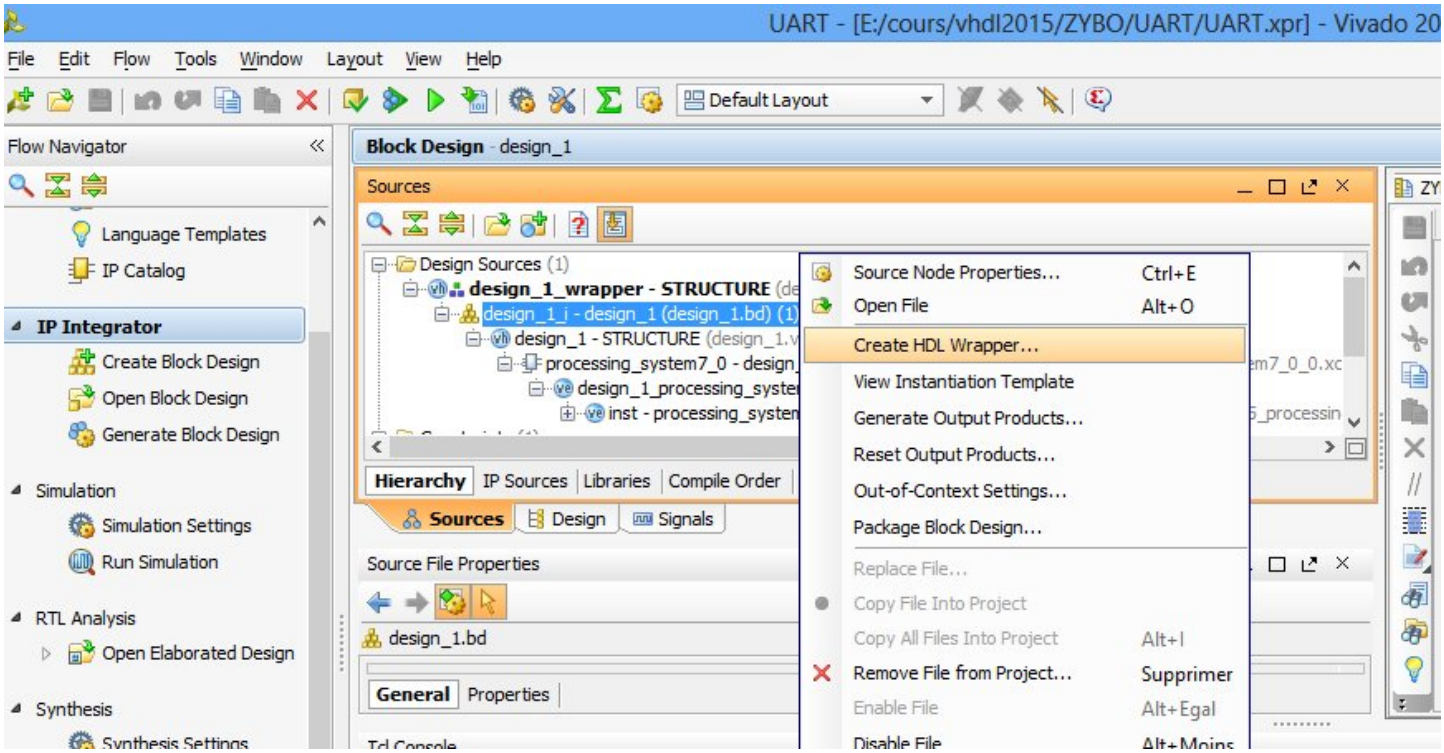
L'étape suivante consiste à générer les fichiers de configuration du PS pour le logiciel SDK

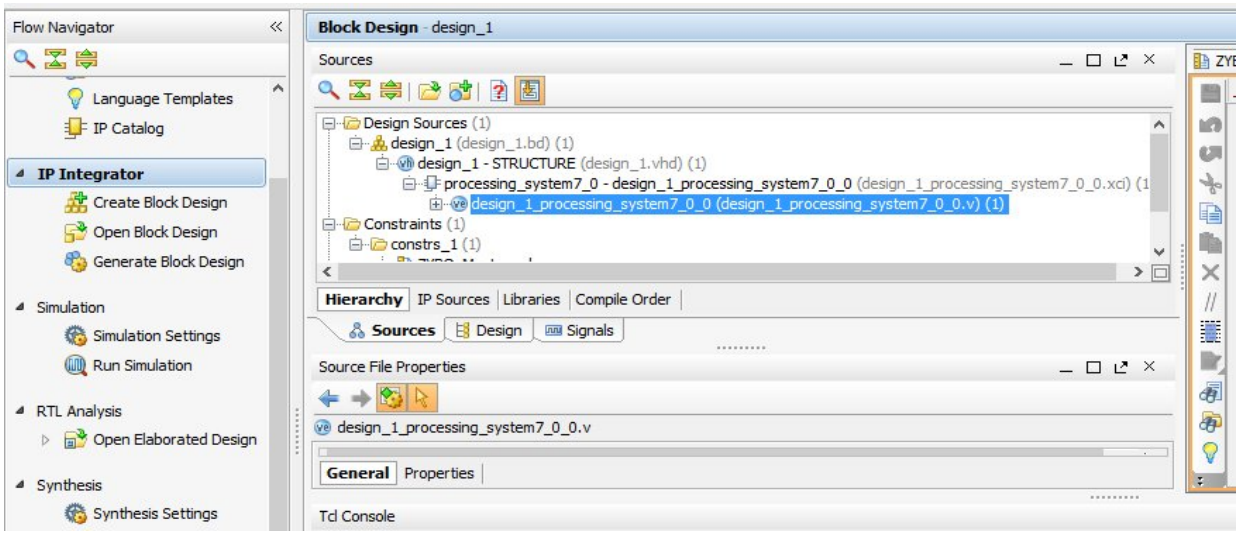


Cliquez sur generate

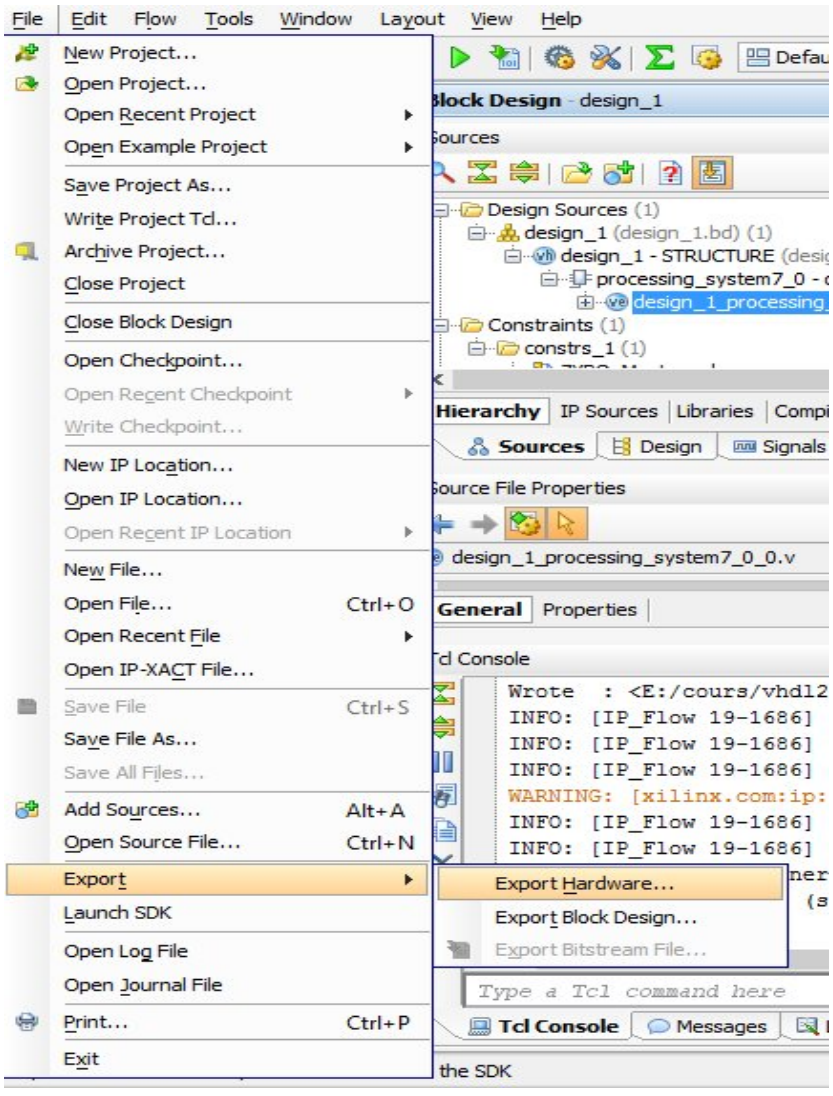


Créer un fichier top VHDL pour le bloc design

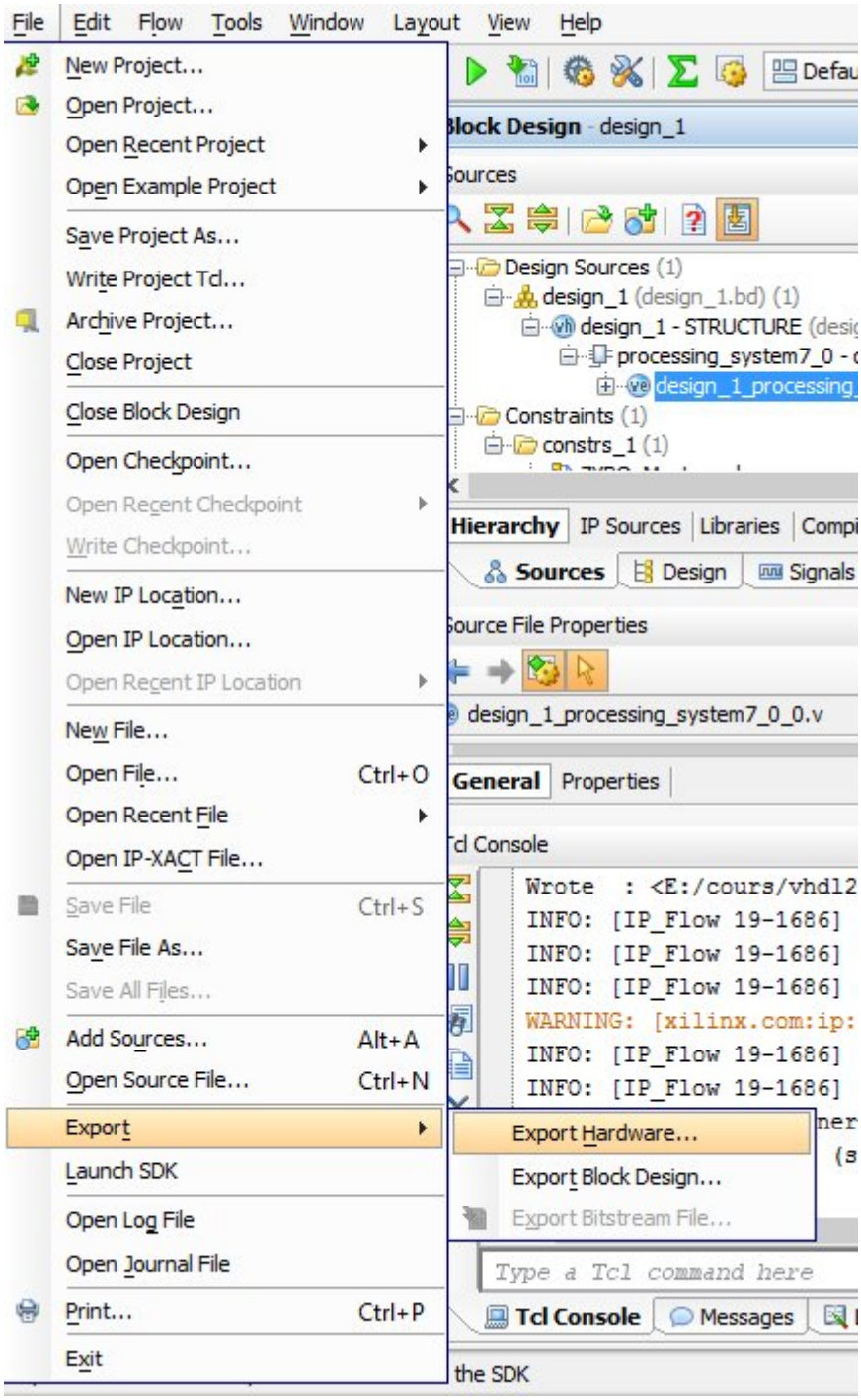




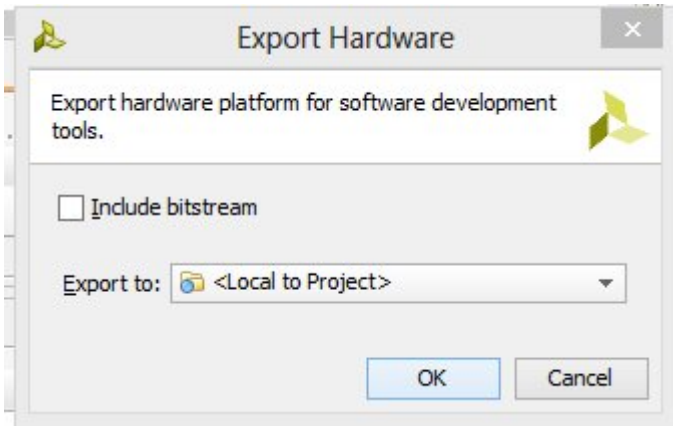
Exporter le matériel vers SDK



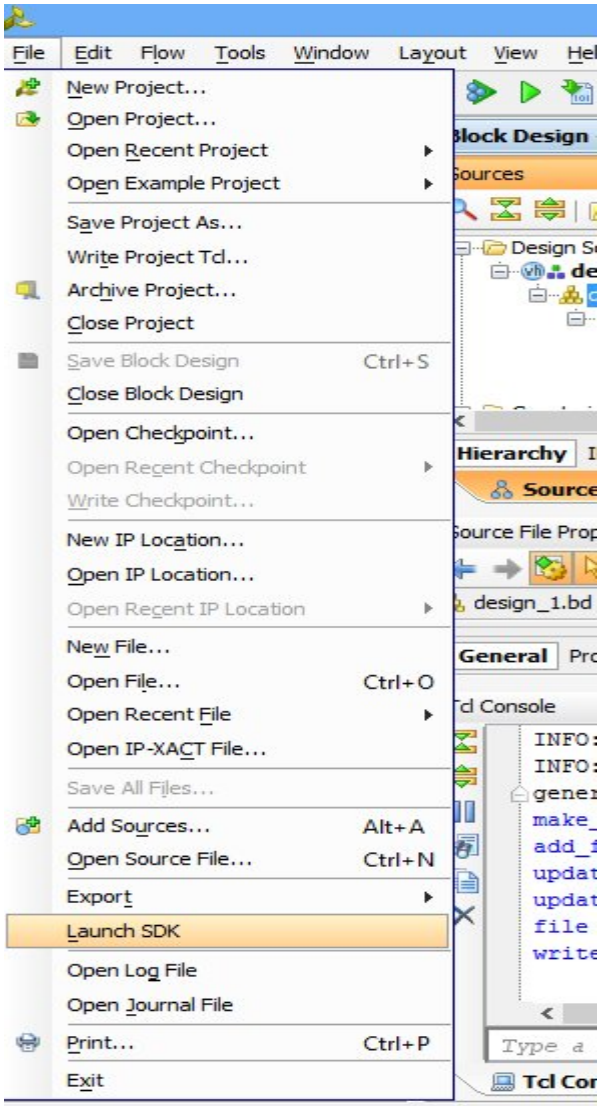
Vivado va exporter la spécification de la plateforme matérielle pour la conception dans SDK (system.xml). En plus du fichier system.xml, il y a quatre autres fichiers qui sont ps7_init.c, ps7_init.h, ps7_init.tcl, ps7_init.html
Les fichiers ps7_init.c, ps7_init.h contiennent le code d'initialisation pour le système de traitement ZYNQ et l'initialisation de la DDR, clocks, PLL et MIO
Exporter le matériel vers SDK



Cliquez sur ok sans inclure de fichier bitstream (aucune logique programmable)

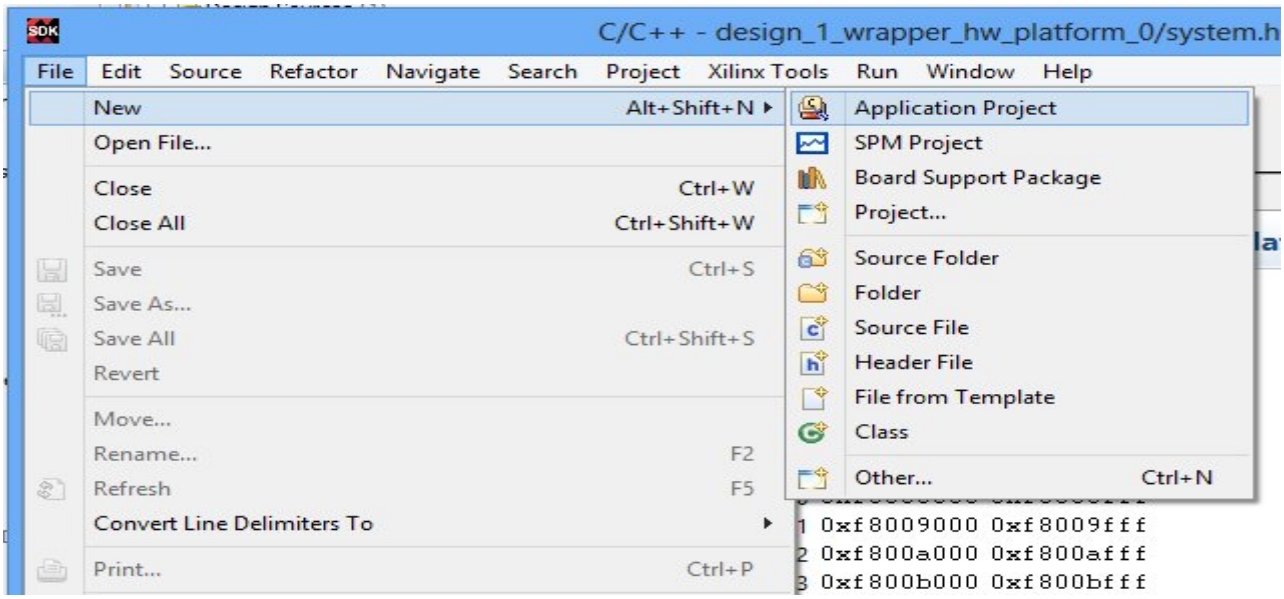


Lançons SDK

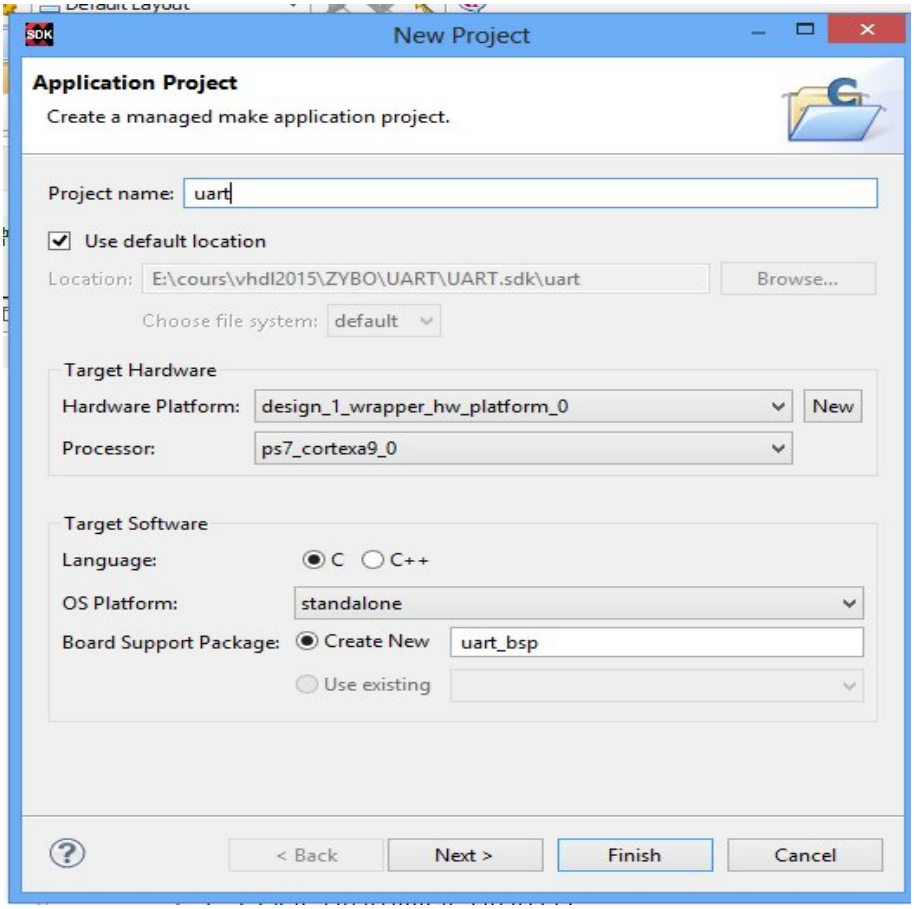


4) Développement logiciel de la partie traitement PS

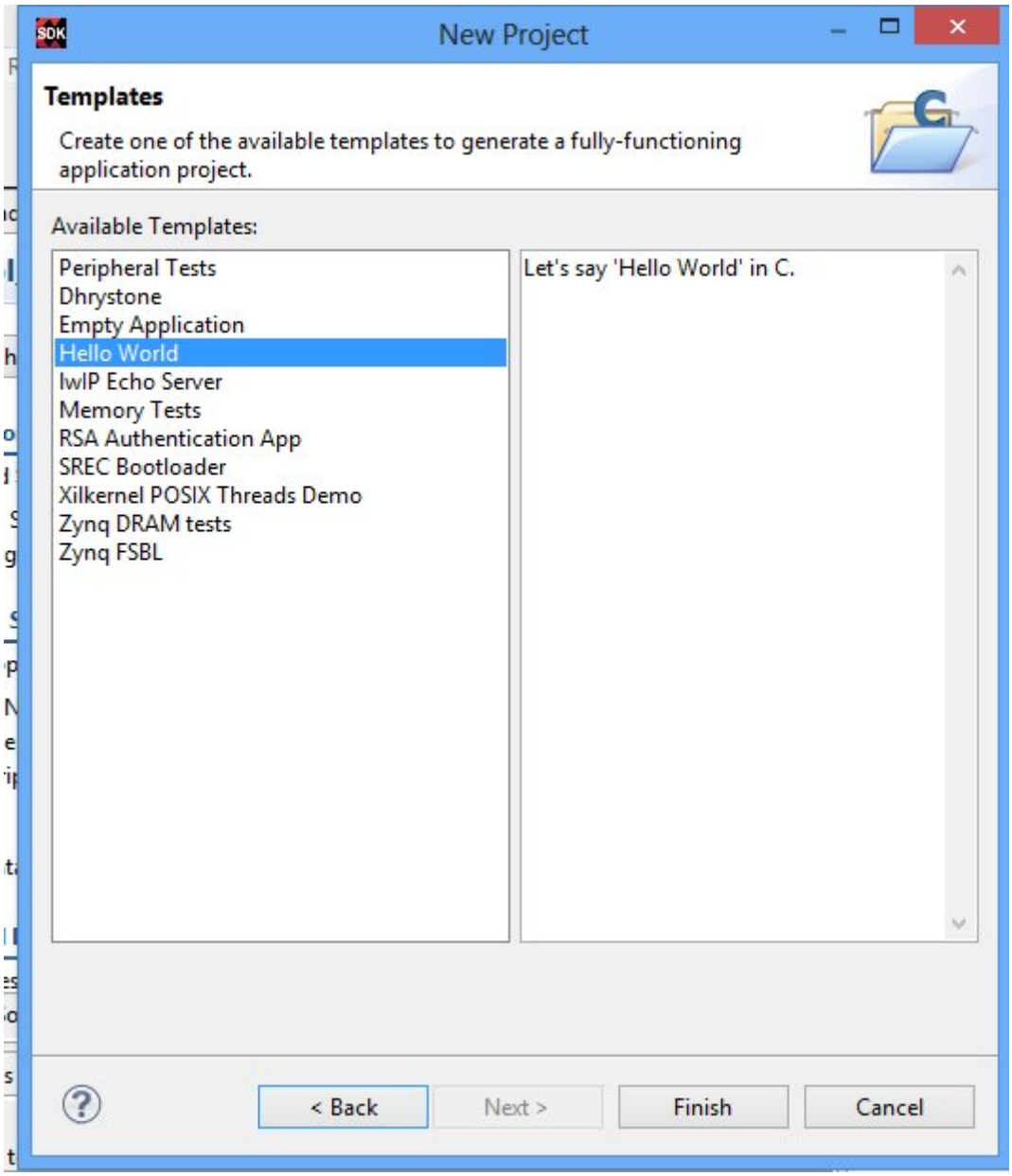
Créez une nouvelle application



Nommez votre projet UART



Choisir une application du type helloworld



Complétez votre code source afin qu'il réponde à l'application envisagée, débugez et testez votre programme.