電物所311651052 吳挺宇

1. Differences between MAXI and Stream interface

數據傳輸模型：

MAXI接口用於內存映射的數據傳輸，支持隨機訪問內存位置，並使用讀寫協議。

Stream接口設計用於連續數據流處理，數據以連續、單向流動的方式進行。

數據流動：

MAXI接口支持讀和寫操作，有獨立的讀和寫通道，可以由主模塊和從模塊發起數據傳輸。

Stream接口通常是單向的，可以是輸入流或輸出流，數據以連續、順序方式推送或拉取，不涉及單個內存位置的地址訪問。

用途：

MAXI接口常用於內存訪問和內存映射外設的接口，適用於需要隨機訪問內存位置的任務。

Stream接口適用於數據密集型處理，如視頻和音頻處理、圖像過濾和信號處理，以連續流方式處理數據。

突發與連續數據：

MAXI接口支持突發數據傳輸，可以一次性讀取或寫入多個數據元素。

Stream接口數據連續流動，一次傳輸一個元素，不支持突發數據傳輸。

協議複雜性：

MAXI接口通常遵循複雜的AXI協議，包括突發數據、寫入 strobe 和地址解碼等功能。

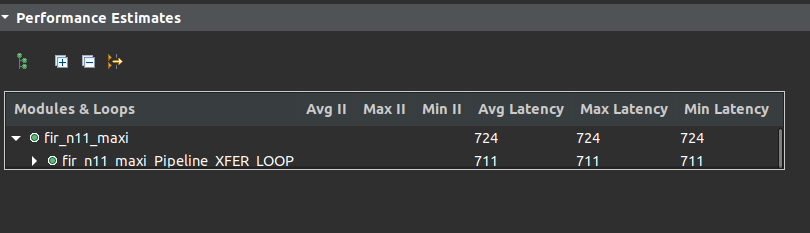
Stream接口更簡單，不涉及複雜的地址和控制信號，因此更容易實現。

總之，MAXI接口和Stream接口在用途、數據流動模式、數據處理方式和協議複雜性等方面存在顯著不同，選擇取決於設計應用的具體需求。

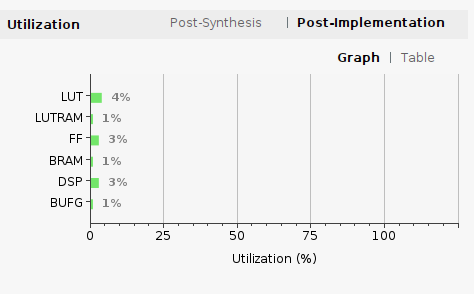
1. Differences between csim and cosim
   1. Csim:在text,kernal都是用c語言時，用來驗證演算法是否正確。
   2. Co-sim:在text是c語言，kernal是verilog時驗證，差異在於co-sim會涉及到數據傳輸的問題。

* MAXI

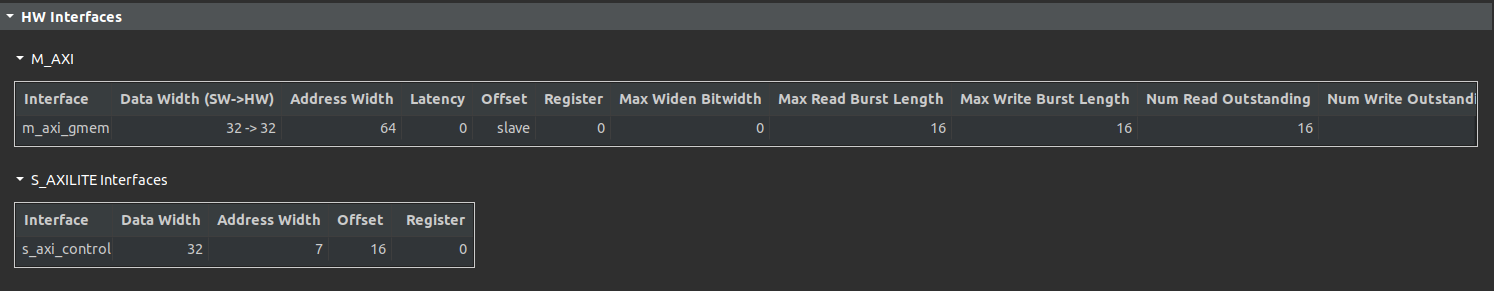
Performance:



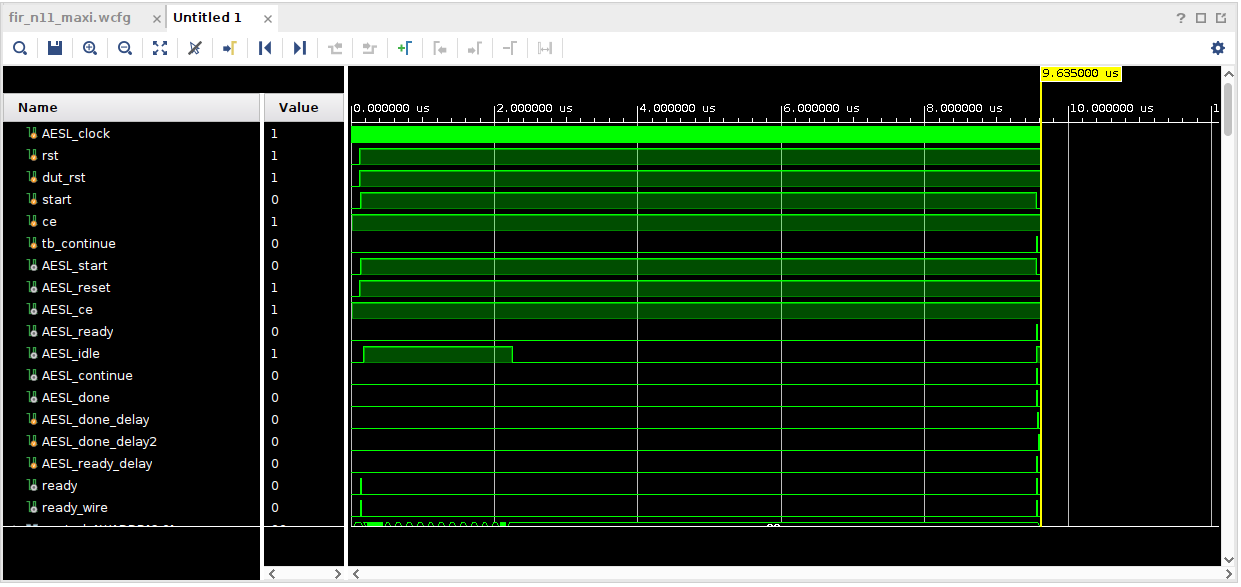
Utilization:



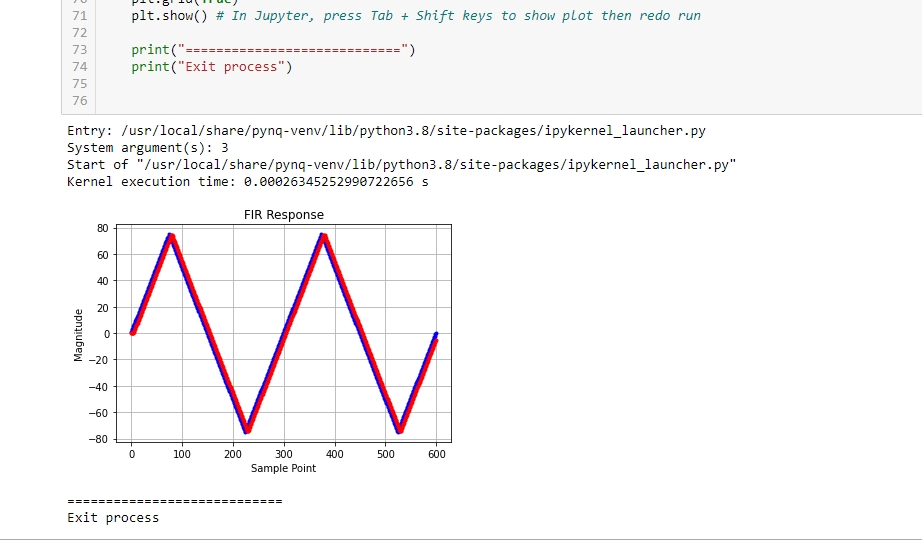
Interface:



Co-simulation transcript/waveform:

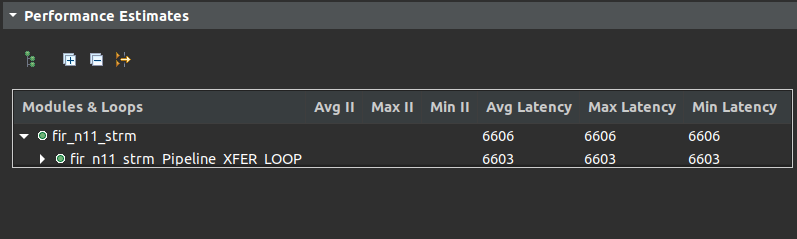


Jupyter Notebook execution results:

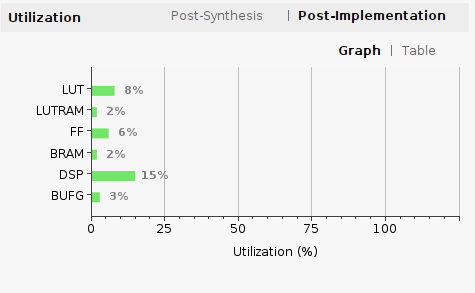


* Stream

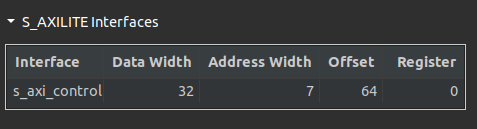
Performance:



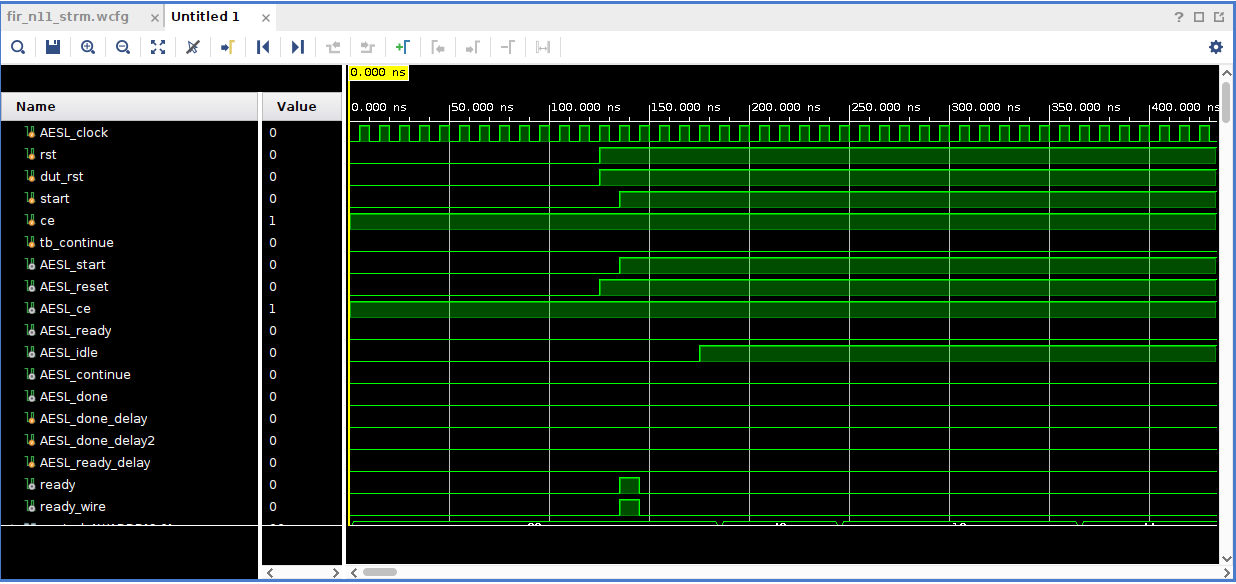
Utilization:



Interface:



Co-simulation transcript/waveform:



Jupyter Notebook execution results:

