

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。

2. 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。

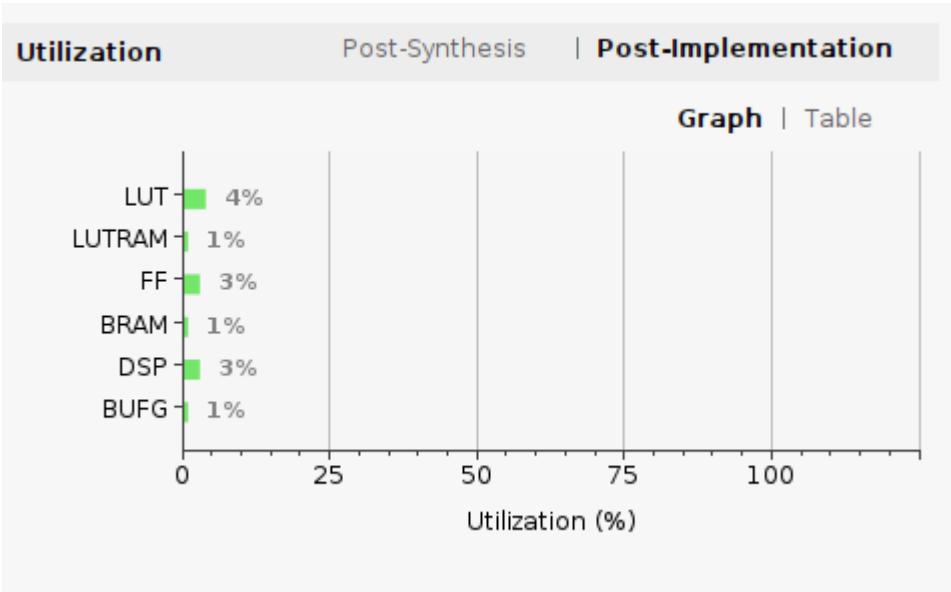
● **MAXI**

Performance:

Performance Estimates

Modules & Loops	Avg II	Max II	Min II	Avg Latency	Max Latency	Min Latency
▼ fir_n11_maxi				724	724	724
▶ fir_n11_maxi Pipeline XFER LOOP				711	711	711

Utilization:



Interface:

HW Interfaces

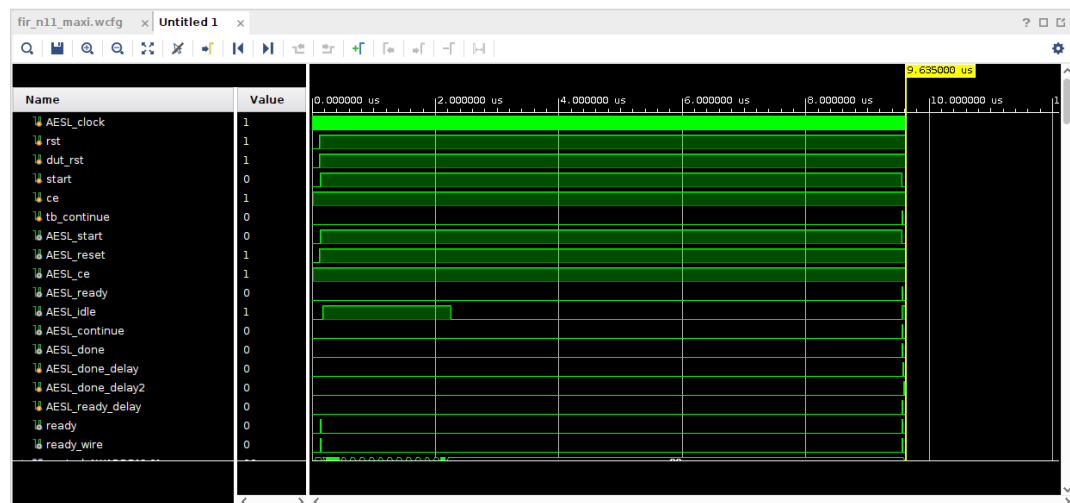
▼ M_AXI

Interface	Data Width (SW->HW)	Address Width	Latency	Offset	Register	Max Widen Bitwidth	Max Read Burst Length	Max Write Burst Length	Num Read Outstanding	Num Write Outstanding
m_axi_gmem	32->32	64	0	slave	0	0	16	16	16	16

▼ S_AXILITE Interfaces

Interface	Data Width	Address Width	Offset	Register
s_axi_control	32	7	16	0

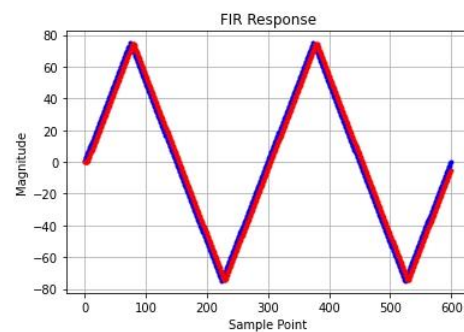
Co-simulation transcript/waveform:



Jupyter Notebook execution results:

```
70 plt.show() # In Jupyter, press Tab + Shift keys to show plot then redo run
71
72
73 print("=====")
74 print("Exit process")
75
76
```

```
Entry: /usr/local/share/pynq-venv/lib/python3.8/site-packages/ipykernel_launcher.py
System argument(s): 3
Start of "/usr/local/share/pynq-venv/lib/python3.8/site-packages/ipykernel_launcher.py"
Kernel execution time: 0.00026345252990722656 s
```




```
=====
Exit process
```

- Stream

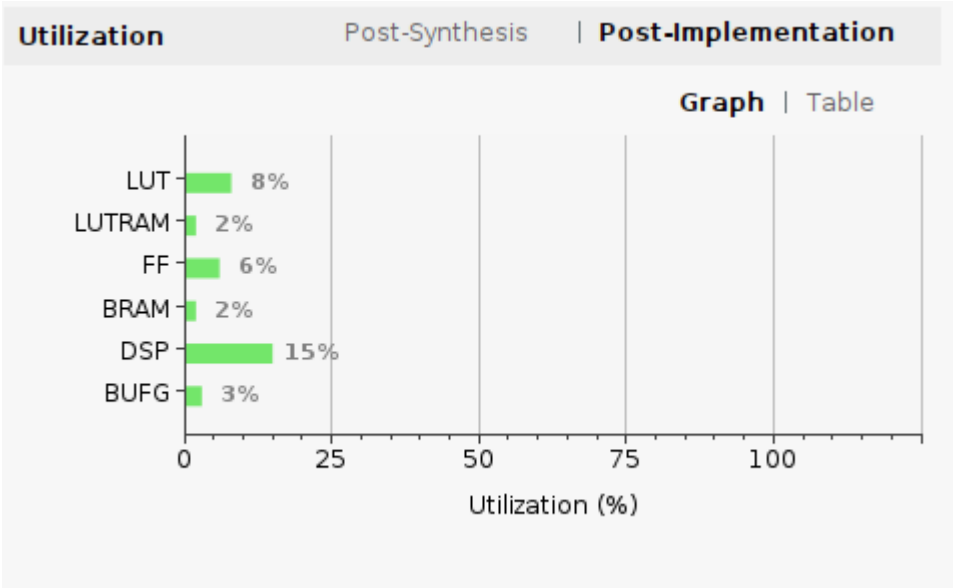
Performance:

▼ Performance Estimates



Modules & Loops	Avg II	Max II	Min II	Avg Latency	Max Latency	Min Latency
▼ ● fir_n11_strm				6606	6606	6606
▶ ● fir_n11_strm Pipeline XFER LOOP				6603	6603	6603

Utilization:

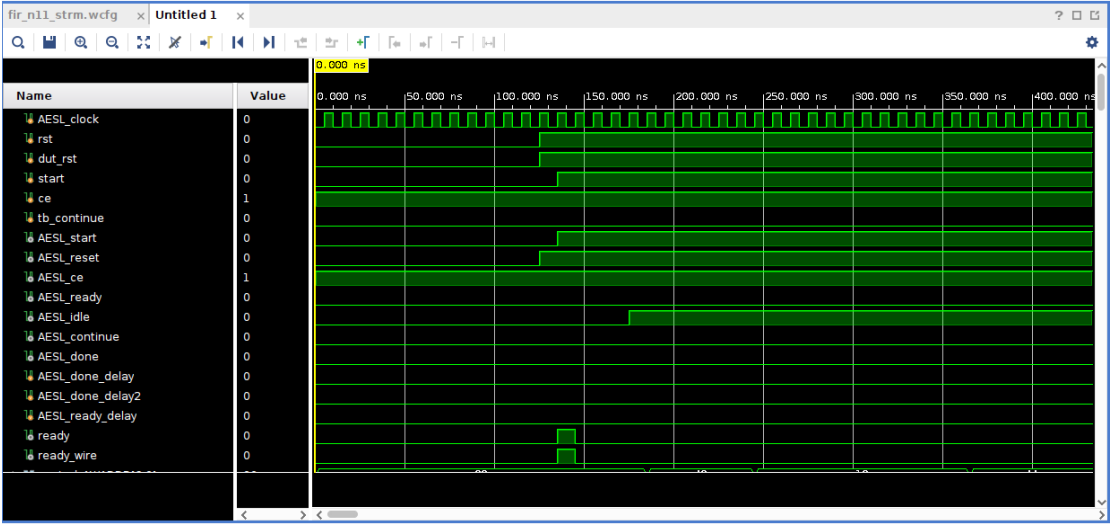


Interface:

▼ S_AXILITE Interfaces

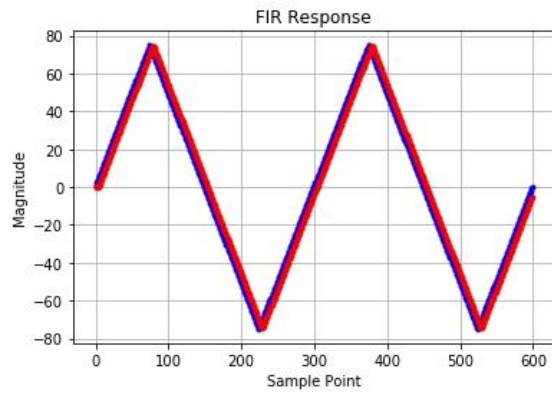
Interface	Data Width	Address Width	Offset	Register
s_axi_control	32	7	64	0

Co-simulation transcript/waveform:



Jupyter Notebook execution results:

```
Entry: /usr/local/share/pynq-venv/lib/python3.8/site-packages/ipykernel_launcher.py
System argument(s): 3
Start of "/usr/local/share/pynq-venv/lib/python3.8/site-packages/ipykernel_launcher.py"
Kernel execution time: 0.0016443729400634766 s
```



```
=====
Exit process
```