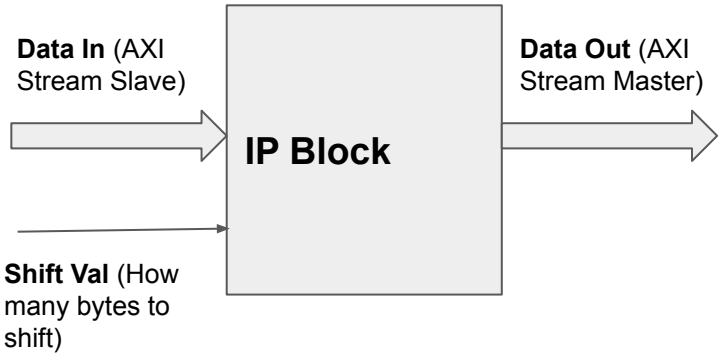
Task 1



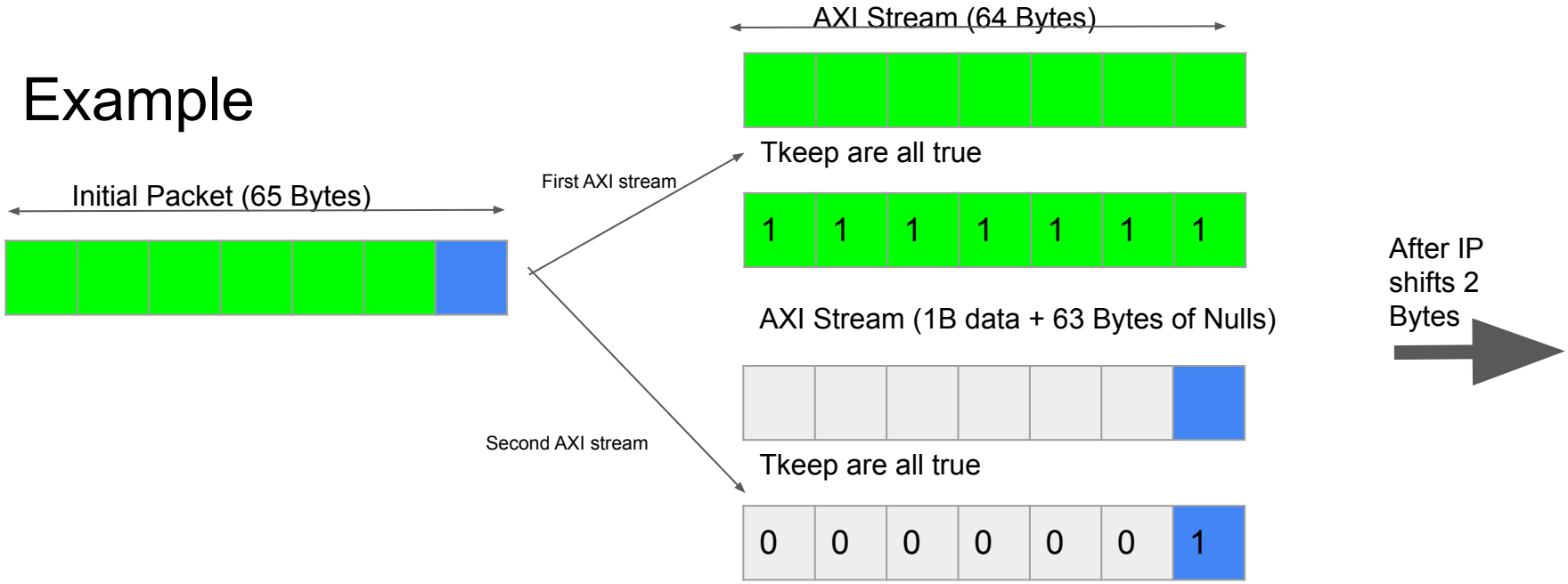
Overall IP definition

IP Block is a functional module performing shift functionality and should work in pipelined fashion. The inputs are a AXI Stream slave interface and also a signal defining how many bytes should be shifted from input stream interface before outputting it on the AXI Stream master interface. The frequency of the IP should be 250MHz(on Xilinx Ultrascale+). The AXI bus width should be configurable and can take values 256 bit or 512 bit.

Functionality Description

Let's assume we have a packet of 65 bytes and **Shift Val** = 2 Bytes(as a example). If the IP is configured with 512 bit width AXI interface it means testbench should divide this packet into 2 AXI streams data and feed to **IP Block** on the first clock with 64 Bytes of packet (tkeep = 64 bits and all are True) and on a second clock 1 Byte of packet data(the last byte) filling others with Nulls (tkeep in this case is equal to True for the first bit and other 63 bits are False). Please not that there is no gaps into a packet which means tkeep bits can be false only for the last **Data In** Bytes**. Shift Val** only indicates how many bytes should be shifted from the start of the packet. After removing the bytes from the packet all other bytes alongside with tkeep data should be shifted right. In this particular example of 65 Bytes of packet and 2 Bytes of **Shift Val** the output should be only one **Data Out** with 63 Bytes of packet data + one Byte of Null (this output should be issued after second **Data In**). In this case the tkeep should be 63 bits of Trues and one bit False.
Please note that IP should issue data only when 64 Bytes of data accumulated or the last packet appeared.

Example


Initial Packet (65 Bytes)
First AXI stream
Second AXI stream
AXI Stream (64 Bytes)
Tkeep are all true
1 1 1 1 1 1 1
AXI Stream (1B data + 63 Bytes of Nulls)
Tkeep are all true
0 0 0 0 0 0 1
After IP shifts 2 Bytes
AXI Stream (first 63B data + 1 Byte of Null)
Tkeep are all true
0 1 1 1 1 1 1
After IP shifts 2 Bytes