1. **Reserved lane marker characters**

In this mode, the transmitter will identify a reserved code on one LS lane called the “search character” and replace it with a new code called the “replace character.” The receiver can be told what character to look for in the high speed input (its search character) and then use that as a reference to know which lane the data came from. This can only work if the TLK is able to identify 10-bit word boundaries in the data. Here is how to configure this mode:

* Transmitter:
	+ Choose the lane to be “marked” by selecting it in TX\_LANE\_MARKER[2:0] (0x17 bits 14:12).
	+ Identify a character that will be present in that lane’s data and write it to the TX\_SEARCH\_MARKER register (0x8000).
		- If 8b/10b encoding is to be used, the LS 8 bits refer to the un-encoded (8-bit) representation of the data and bit 9 indicates whether it should represent a K-code (=1) or a D-code (=0).
		- If 8b/10b encoding is not used, the LS 10 bits of this register represent the 10-bit code to be matched.
	+ Choose a new character that will be used to replace the “search marker” character and write it to TX\_REPLACE\_MARKER (register 0x8001). This character should be one that it typically not going to be present in the data.
	+ Enable word replacement by setting TX\_LANE\_MARKER\_EN (0x1D bit 7) = 1.
* Receiver
	+ Program in a “search” character that matches the transmitter’s “replace” character (register 0x8002).
	+ Choose the character that will replace the marker character when it is output on the LS side (register 0x8003). If you make this equal to the transmitter’s “search” character, then the lane re-ordering scheme will be transparent to the end system (i.e., LS data input on one side will match the LS data output on the other side).
	+ Choose the lane which you expect to be marked in RX\_LANE\_MARKER[2:0] (0x18 bits 14:12).
	+ Enable marker detection by setting RX\_LANE\_MARKER\_EN (0x1D bit 11) = 1.
1. **Training Sequence**

For data that does not have a 10-bit alignment (thus requiring bit interleaving), it isn’t possible to use the above search/replace method. Since the TLK won’t be aware of word boundaries or encoding type in this case, it cannot make any modifications to the LS data itself. Since the LS data cannot be modified, the lane ordering should be determined at start-up. This is done by replacing the data of one of the lanes with a special alignment pattern that the TLK is designed to recognize. Here is how to configure this mode:

* Transmitter:
	+ Choose the lane that will carry the alignment pattern by selecting it in TX\_LANE\_MARKER[2:0] (0x17 bits 14:12).
	+ Enable the lane marker sequence by setting BIT\_LM\_EN (0x17 bit 11) = 1.
	+ After the receiver reaches alignment, turn off the pattern by setting BIT\_LM\_EN = 0.
		- This should happen quickly, so as long as the HS link is stable you can just do this step immediately after the previous one. Otherwise (i.e., if the receiver board is not ready yet), you can wait a longer time or wait for the receiver to indicate it is done through BIT\_LM\_DONE (0x0F bit 4).
* Receiver:
	+ Choose the lane that is carrying the alignment pattern by identifying it in RX\_LANE\_MARKER[2:0] (0x18 bits 14:12).
	+ Enable alignment pattern detection by setting BIT\_LM\_PATT\_DETECT\_EN (0x1D bit 11) = 1. (This is already enabled by default and can be left on during normal operation without an issue, since it is unlikely that the receiver would detect the alignment pattern by mistake.)
1. **Manual Lane Rotation**

This mode is something that can be used in bit-interleaved mode if customers have some other way of detecting if the lane ordering is correct and do not want it automatically controlled by the TLK. In this case the system would just tell the receiver to cycle through the different possible lane orderings until it determines the correct one has been reached. To signal the receiving TLK to try a new lane ordering, you would just need to toggle the RXCTRL\_0/1 pins. Alternatively, you could use the crosspoint switch function built into the device to manually reassign lanes.

1. **Reserved Lane**

This mode is actually the same as mode 2 (training sequence). If one lane is never used in the application, it can always send the training pattern. This can be useful, since it makes it easy for the receiver to recover if the high speed link is lost temporarily.

Settings for bit interleave mode:

1) Set bit interleave bits, write 0x4300 to 0x01 (or 0x0300 if you don't want link training)
2) Disable ch\_sync and hs/ls enc/dec. Write 0x002C to 0x1c and 0x1d
3) Set the BIT\_LM\_EN, write 0x0ABC to 0x17
4) Wait 1s, clear BIT\_LM\_EN, write 0x02BC to 0x17

Steps 3 & 4 are the bit interleave lane auto-rotation function.