MPC – SP Synchronization Procedure

Petersburg Nuclear Physics Institute / University of Florida
Revision 2.0
March 25, 2002

Introduction

The distance between on-chamber electronics (FEBs, TMBs, MPC) and the counting room (SP) is about 100 m. The Muon Port Card (MPC) uses optical links to transmit 32-bit data per link every 25 ns. Each MPC contains parallel to serial converters and optical transmitter modules, while each Sector Processor (SP) contains optical receiver modules and serial to parallel converters. Texas Instrument's TLK2501 1.6 to 2.5 Gbps Transceiver [1] serves as a serializer/deserializer circuit and Finisar's FTRJ-8519-1-2.5 2-Gigabit Short-Wavelength 2x5 Pin Small Form Factor (SFF) Transceiver [2] serves as an optical transmitter/receiver at both ends of the link, Figure 1. To transmit and receive 32-bit data @ 40 MHz, while the TLK2501 is basically a 16-bit device, the data are multiplexed at the transmitter end @ 80 MHz and demultiplexed at the receiver end. De-multiplexers are implemented in Front FPGAs. For correct operation the SP should lock its deserializers to the link frequency, and its demultiplexers to the first data frame, Table 1 [3]. Furthermore, this synchronization procedure should be performed for all SPs (in the Track Finder Crate) and hence for all MPCs (in all Peripheral Crates) simultaneously. This note describes both the TLK2501 Transcevers and the link synchronization procedure.

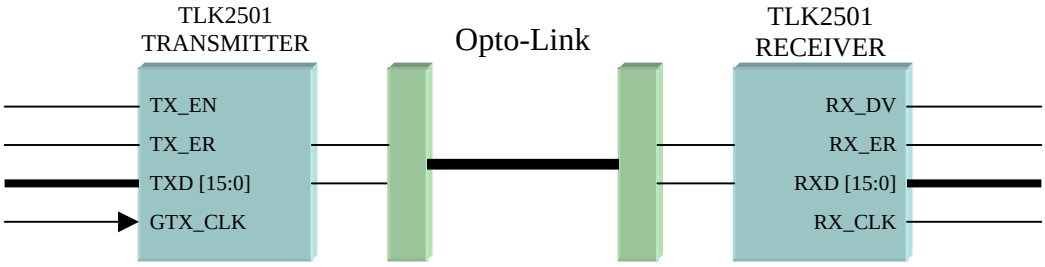

Figure 1 Transmit/Receive Diagram

Table 1 Transmit/Receive Data Format

Bits		15	14	13	12	11	08	07	06	00
Data	1	VP	Quality				CLCT Pattern #		Wire Group ID	
Frame	2	CSC ID				BC0	BX0	SE	L/R	CLCT Pattern ID

The TLK2501 Details

The transmitter latches 16-bit parallel data at a rate based on the supplied reference clock (GTX_CLK). The 16-bit parallel data (TXD [15-0]) is internally encoded into 20 bits using an 8-bit/10-bit (8B/10B) encoding format. The resulting 20-bit word is then

transmitted differentially at 20 times the reference clock rate. The receiver section performs the serial-to-parallel conversion on the input data, synchronizing the resulting 20-bit wide parallel data to the extracted reference clock (RX_CLK). It then decodes the 20-bit wide data using 8-bit/10-bit decoding format resulting in 16 bits of parallel data at the receive data terminals (RXD [15-0]). The outcome is an effective data payload of 1.28 Gbps to 2.0 Gbps (16-bit data times the GTX_CLK frequency).

The encoding is dependant upon two additional input signals, the TX_EN and TX_ER. When the TX_EN is HIGH and the TX_ER is LOW then the data bits TXD [15:0] are encoded and transmitted normally. When the TX_EN is LOW, and TX_ER is HIGH, then the encoder generates a carrier extend consisting of two K23.7 (F7F7) codes. Each valid Transmission Character of the 8B/10B Transmission Code is given a name using the following convention: cxx.y [4], where c is used to show whether the Transmission Character is a Data Character (c is set to D) or an Extended Character (c is set to K). If the TX_EN and the TX_ER are both HIGH, then the encoder generates a K30.7 (FEFE) code.

Table 2 provides the transmit data control decoding. Since the data is transmitted in 20-bit serial words, K codes indicating carrier extend and transmit error propagation are transmitted as two 10-bit K-codes. The encoder inserts the IDLE character set when no payload data is available to be sent. IDLE consists of a K28.5 (BC) code and either a D5.6 (C5) or a D16.2 (50) character. IEEE802.3z defines the K28.5 character as a pattern consisting of 0011111010 (a negative number beginning disparity) with the 7 MSBs (0011111) referred to as the comma character. Since data is latched into the TLK2501 16 bits at a time, this in turn is converted into two 10-bit codes that are transmitted sequentially. This means IDLE consists of two 10-bit codes, 20 bits wide that are transmitted during a single GTX_CLK cycle.

Table 2. Transmit Data Controls

TX_EN	TX_ER	ENCODED 20-BIT DATA
Low	Low	Idle (< K28.5, D5.6>, < K28.5, D16.2>)
Low	High	Carrier extend (K23.7, K23.7)
High	Low	Normal data character (DX.Y)
High	High	Transmit error propagation (K30.7, K30.7)

When parallel data is clocked into a parallel to serial converter, the byte boundary that was associated with the parallel data is now lost in the serialization of the data. When the serial data is received and converted to parallel format again, a way is needed to recognize the byte boundary. Generally this is accomplished through the use of a synchronization pattern. This is generally a unique pattern of 1s and 0s that either cannot occur as part of valid data or is a pattern that repeats at defined intervals. 8-bit/10-bit encoding contains a character called the comma (b0011111 or b1100000), which is used by the comma detect circuit on the TLK2501 to align the received serial data back to its original byte boundary. The decoder detects the K28.5 comma, generating a synchronization signal aligning the data to their 10-bit boundaries for decoding. It then converts the data back into 8-bit data, removing the control words. The output from the two decoders is latched into the 16-bit register synchronized to the recovered parallel data

clock (RX_CLK) and output valid on the rising edge of the RX_CLK. Two output signals, RX_DV/LOS and RX_ER, are generated along with the decoded 16-bit data output on the RXD [15:0] terminals. The output status signals are asserted as shown in Table 3.

Table 3. Receive Status Signals

RECEIVED 20-BIT DATA	RX_DV/LOS	RX_ER
Idle (< K28.5, D5.6>, < K28.5, D16.2>)	Low	Low
Carrier extend (K23.7, K23.7)	Low	High
Normal data character (DX.Y)	High	Low
Receive error propagation (K30.7, K30.7)	High	High

When the TLK2501 decodes normal data and outputs the data on RXD [15:0], RX_DV/LOS is HIGH and RX_ER is LOW. When the TLK2501 decodes a K23.7 code (F7F7) indicating carrier extend, RX_DV/LOS is LOW and RX_ER is HIGH. If the decoded data is not a valid 8-bit/10-bit code, an error is reported by setting HIGH of both RX_DV/LOS and RX_ER. If the error was due to an error propagation code, the RXD bits outputs hex FEFE. If the error was due to an invalid pattern, the data output on RXD is undefined. When the TLK2501 decodes an IDLE code, both RX_DV/LOS and RX_ER are LOW and a K28.5 (BC) code followed by either a D5.6 (C5) or D16.2 (50) code are output on the RXD terminals.

The data transmission latency of the TLK2501 is defined as the delay from the initial 16-bit word load to the serial transmission of bit 0. The serial-to-parallel data receive latency is the time from when the first bit arrives at the receiver until it is output in the aligned parallel word with RXD0 received as first bit. The transmit/receive latency is fixed once the link is established. However, due to silicon process variations and implementation variables such as supply voltage and temperature, the exact delay varies slightly.
The minimum transmit latency is 34 bit times; the maximum is 38 bit times. The minimum receive latency is 76 bit times; the maximum is 107 bit times. The total latency variation might be as many as (38 - 34) + (107 - 76) = 35 bit times, or about 22 ns in our case.

The TLK2501 has a synchronization-state machine, which is responsible for handling link initialization and synchronization. Upon power up or reset, the state machine enters the acquisition (ACQ) state and searches for IDLE. Upon receiving three consecutive IDLEs or a carrier extend, the state machine enters the synchronization (SYNC) state. If, during the acquisition process, the state machine receives valid data or an error propagation code, it immediately transitions to the SYNC state. The SYNC state is the state for normal device transmission and reception. In fact, a state machine behavior is more complicated than described here. The only way to synchronize a transmitter / receiver pair is to force transmitter into IDLE mode for a while.

The Demux Details

The demultiplexers are implemented in Front FPGAs. They are intended to restore the initial 32-bit @ 40MHz data structure from the received 16-bit @ 80MHz data stream.

The frame content is specified in Table 1. Note that there is no “frame bit” available, which could have helped to distinguish between frames. A procedure is required to achieve frame synchronization in the SP.

Synchronization Procedure

The MPC-SP link synchronization procedure is supposed to be performed:

- ✓ during cold start (power-up);
- ✓ after Hard Reset;
- ✓ when Loss of Synchronization is detected in any of SP input links due to:
  - mismatch between data BX0 bit and LSB of SP internal bunch crossing counter;
  - mismatch between data BC0 flag and zero state of SP internal bunch crossing counter;
  - mismatch between CCB BC0 command and data BC0 flag;
  - some other conditions to be specified later.

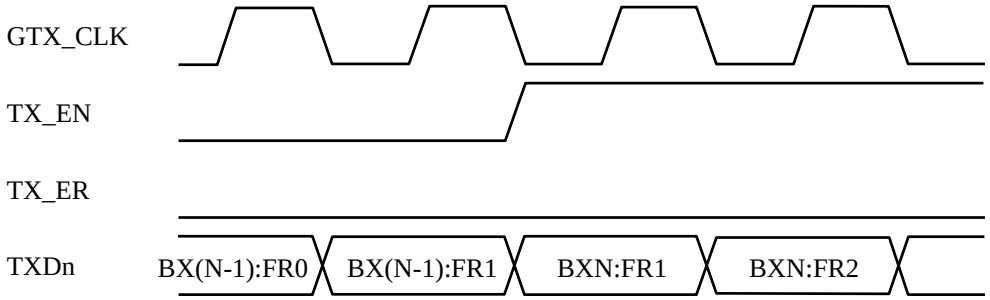

Figure 2. Transmit Side Resumes Normal Operation

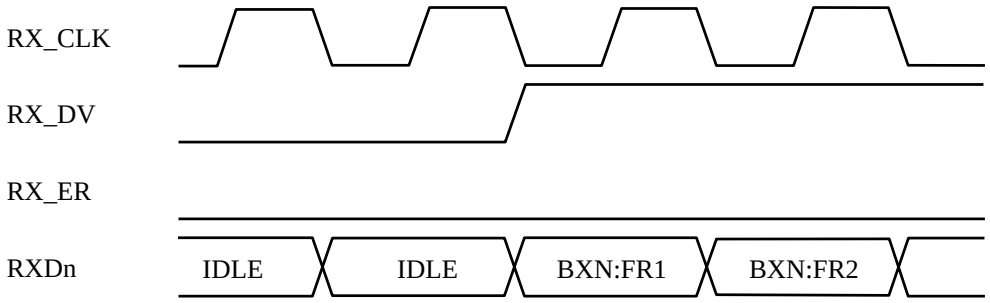

Figure 3. Receive Side Resumes Normal Operation

As we have learned already, the only way to synchronize or re-synchronize a TLK2501 receiver is to put a corresponding TLK2501 transmitter into IDLE mode for at least three consecutive periods of GTX_CLK. A practical number of periods to hold transmitter in an IDLE mode is determined by two factors:

1. The number should be even to correspond to a whole number of bunch crossings;
2. The IDLE interval should be larger than variation of BC0 arrival times at SP inputs for all MPC links.

IDLE interval of 3 to 5 bunch crossings seems to be sufficient to meet both requirements, and it should be the same for all MPCs.

There are also two prerequisite conditions for MPC-SP link synchronization procedure to complete successfully:

1. The MPC timing should be already aligned with the orbit timing. In other words, the TTC BC0 broadcast command should be aligned with the BC0 flag, marking bunch crossing zero data.
2. Switching of TX_EN and TX_ER control inputs should occur at 32-bit data boundaries, but not on 16-bit frame boundaries.

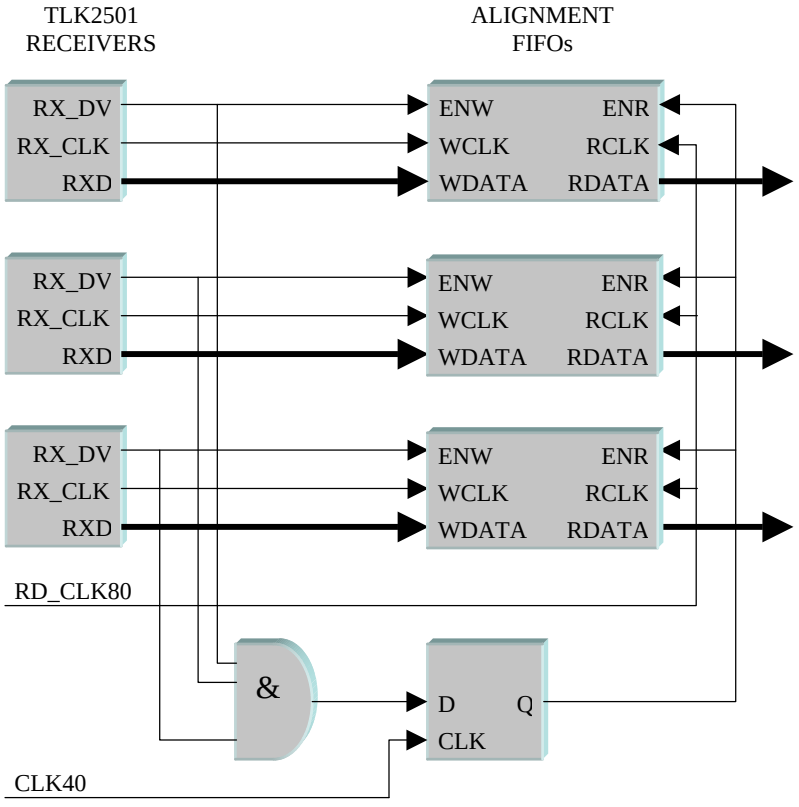

Figure 4. Simplified Alignment Schematics.

Under the above-mentioned conditions a TTC L1 Reset broadcast command restores link synchronization between MPC and SP, recovers the correct frame sequence in SP, and simultaneously aligns SP input data:

- Upon receiving of an L1 Reset broadcast command, the MPC responds by setting each TLK2501 transmitter into IDLE mode (TX_EN goes LOW; TX_ER has already been LOW for normal operation) for N bunch crossings (N=3…5);
- Each SP TLK2501 receiver switches into IDLE mode when the signal reaches it, that is after transmission latency, link propagation delay, and data reception latency;

- The SP Alignment FIFOs (see Figure 4) have been reset by the same L1 Reset broadcast command (reset not shown) and are waiting for valid data from each TLK2501 receiver.
- Each Alignment FIFO resumes writes after the corresponding TLK2501 receiver has returned to normal operation (started receiving data). All FIFOs have BXN data written as first two data words, see Figure 2 and Figure 3.
- After all receivers have started getting valid data (all RX_DVs became HIGH), the AND of RX_DVs is synchronized with the SP bunch crossing clock (CLK40) and enables the FIFO reads. All Alignment FIFOs have data belonging to the same bunch crossing (BXN) at their outputs. The SP is aligned to the latest link.

References

[1] TLK2501 1.6 to 2.5 GBPS Transceiver Datasheet available at:
http://red.pnpi.spb.ru/~uvarov/parts/TLK2501.pdf or
http://www.phys.ufl.edu/~uvarov/parts/TLK2501.pdf
[2] 2 Gigabit Short-Wavelength 2x5 Pin SFF Transceiver
FTRJ-8519-1-2.5 Product Specification available at:
http://red.pnpi.spb.ru/~uvarov/parts/FTRJ-8519-1-2dot5_Spec_Rev_D.pdf or
http://www.phys.ufl.edu/~uvarov/parts/FTRJ-8519-1-2dot5_Spec_Rev_D.pdf
[3] MPC-SP Data Format, Revision 2.0 available at:
http://red.pnpi.spb.ru/~uvarov/tf_crate/LU-MPC_SP_Data_Format_2d0.pdf or
http://www.phys.ufl.edu/~uvarov/tf_crate/LU-MPC_SP_Data_Format_2d0.pdf
[4] Fibre Channel Physical and Signaling Interface (FC-PH), Revision 4.3, page 63 (pdf-98), available at
http://www.nowhere.net/~raster/FC/fcph_43.pdf

Revision History

Date	Version	Comment
October 3, 2001	1.0	Initial Release
October 10, 2001	1.1	Reference [3] has been added
March 25, 2002	2.0	Acronyms BC1 and BC0 in Table 1 have been replaced with BX1, BX0. MPC-SP Format changed: BX1 replaced with BC0 – Bunch Crossing Zero timing mark. Synchronization procedure has been re-written