CDF/DOC/TRIGGER/CDFR/4578
Version 5.2
May 5, 1998

Specification of the XTRP, SVT, and Level 2 Interfaces

S. Belforte, J.Berryhill, R. Culbertson, M. Dell'Orso,
H. Frisch, P. Giannetti, E. Meschi, T.Nakaya, G. Punzi,
L. Ristori, M.Shochet, F. Spinella, P. Wilson and A. Zanetti

1 Introduction

This note specifies three interfaces:

1. XTRP to SVT
2. XTRP to Level 2
3. SVT to Level 2

The interface from XTRP to Level 2 is defined to be identical to the XTRP to SVT interface. We provide here one description for both of them.

If there is a Level 1 Accept for a given crossing, the XTRP delivers the list of tracks found by XFT to SVT and L2 as soon as possible. The SVT sends its track list to L2 as soon as it is available after the Level 1 Accept.

Connector type, pin assignment, electrical and timing specifications are common to these three interfaces and to all data communication within SVT.

2 Notation

Signal names will be written using capital letters. A signal whose name ends with an underline (e.g. HOLD_) is always *active-low*, while a signal whose name does not end with underline (e.g. HOLD) is always *active-high*. An *active-high* signal is in the *asserted* or *true* state when it is at the higher voltage value and is *deasserted* or *false* when at the lower voltage value. An *active-low* signal is in the *asserted* or *true* state when it is at the lower voltage value and is *deasserted* or *false* when at the higher voltage value.

Special attention must be paid in order not to confuse the underline symbol with logic negation. As an example consider the case in which two signals HOLD and HOLD_ are both present: when HOLD is *true* (high) then also HOLD_ is *true* (low). When describing signals, in order to minimize possible ambiguities we will try to avoid the use of *true* and *false* and always refer to *high* and *low* states.

3 The SVT Communication Protocol

A uniform communication protocol is used for all data transfer throughout the SVT system. Data flow through unidirectional links connecting one source to one destination. The protocol is a simple pipeline transfer driven by an asynchronous data strobe (DS_). To maximize speed no handshake is implemented on a word by word basis. A HOLD signal is used instead as a loose handshake to prevent loss of data when the destination is busy. The DS_ is used as a *write enable* into the receiving board (see Section 3.2) and data are sampled at every positive going DS_ edge. If no data is available to transmit, DS_ is high. The DS_ is driven asynchronously by the source. Correct DS_ timing must be guaranteed by the source (more details on the data transfer timing are given in Section 3.2).

Input data are pushed into a FIFO buffer. The FIFO provides an *Almost Full* signal that is used to drive the HOLD line high. The FIFO is popped by whatever processor sits in the destination device. If the destination processor does not keep up with the incoming data rate, the FIFO becomes *Almost Full* and the HOLD signal is set to the high state. The source responds to the HOLD signal going high by suspending the data flow. Using *Almost Full* instead of Full gives the source plenty of time to stop (the equivalent of 127 DS_ cycles in the current implementation). The source is not required to wait for an acknowledge from the destination device before sending the following data word, allowing the maximum data transfer rate compatible with the cable bandwidth even when transit times are long.

Signals are sent over flat cable as differential LVDS.

The maximum DS_ frequency is 30 MHz.

3.1 Cables and Connectors

All signals are LVDS, the transmitters are National DS90C031, the receivers are DS90C032. Communication between boards is through front panel connectors, 52 contacts each. Connector type is KEL 8822E-052-171H (receptacle) and 8831E-052-170LH (plug). Two 26 wire flat cables are used for the connection. The two cables of the harness should be taped against each other in at least three places to keep the ground wires close to all signals. This reduces radiation and minimizes inductance. The pitch of the wire in the flat cable is 1.27 mm and the conductor is 28 gage stranded. AMP part numb. 57040-3 (100 ft roll) or 971111-2 (500 ft roll).

Table 1 shows the pin assignment for the source connector. Table 2 shows the pin assignment for the destination connector. Pin naming follows KEL data sheet: pins with the same letter (A or B) belong to the same flat cable. Please note that pin assignment is different for the source and destination connectors, namely A and B rows are interchanged. This allows standard termination of the cables corresponding to cable assembly code AA in KEL data sheet.

The pin assignment has been chosen to satisfy the following requirements:

1. Wires in the same differential pair are assigned to adjacent wires on the same cable (A or B).

2. Inverting inputs in the receiver chips are attached to odd numbered pins in the connector and non-inverting inputs are assigned to even numbered pins. This will hopefully

pin	use	pin	use
A1	D00_	B1	D01_
A2	D00	B2	D01
A3	D02_	B3	D03_
A4	D02	B4	D03
A5	D04_	B5	D05_
A6	D04	B6	D05
A7	D06_	B7	D07_
A8	D06	B8	D07
A9	D08_	B9	D09_
A10	D08	B10	D09
A11	D10_	B11	D11_
A12	D10	B12	D11
A13	D12_	B13	D13_
A14	D12	B14	D13
A15	D14_	B15	D15_
A16	D14	B16	D15
A17	D16_	B17	D17_
A18	D16	B18	D17
A19	D18_	B19	D19_
A20	D18	B20	D19
A21	D20_	B21	EP_
A22	D20	B22	EP
A23	EE_	B23	DS
A24	EE	B24	DS_
A25	HOLD_	B25	GND
A26	HOLD	B26	GND

Table 1: Pinout for the SOURCE front panel connectors.

make board layout easier and more regular.

3. If a cable is disconnected, the corresponding DS_ signal must stay high (deasserted) and the HOLD signal must stay high (asserted). This ensures that data transmission will stop in case a cable is accidentally pulled ultimately resulting in an error condition. To achieve this we rely on the feature of LVDS receivers that ensures the output stays high if the input is disconnected.

4. Two contacts (B25 and B26) are reserved for ground return (GND).

A possible implementation that satisfies this requirement is suggested in the Appendix.

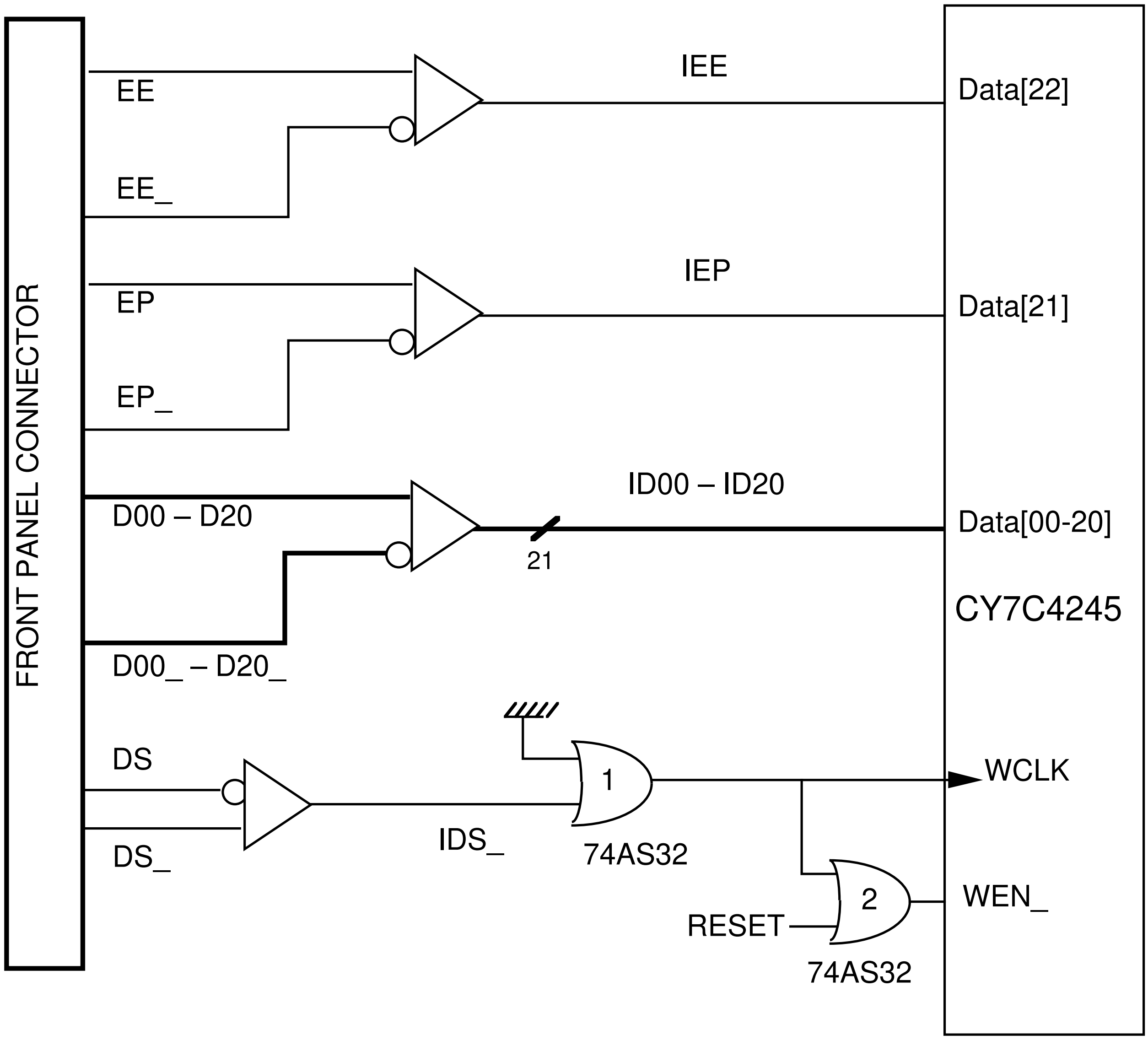

Figure 1: The DS_, D, EP and EE signals from the input connector to the CY7C4245-10 FIFO.

pin	use	pin	use
B1	D00_	A1	D01_
B2	D00	A2	D01
B3	D02_	A3	D03_
B4	D02	A4	D03
B5	D04_	A5	D05_
B6	D04	A6	D05
B7	D06_	A7	D07_
B8	D06	A8	D07
B9	D08_	A9	D09_
B10	D08	A10	D09
B11	D10_	A11	D11_
B12	D10	A12	D11
B13	D12_	A13	D13_
B14	D12	A14	D13
B15	D14_	A15	D15_
B16	D14	A16	D15
B17	D16_	A17	D17_
B18	D16	A18	D17
B19	D18_	A19	D19_
B20	D18	A20	D19
B21	D20_	A21	EP_
B22	D20	A22	EP
B23	EE_	A23	DS
B24	EE	A24	DS_
B25	HOLD_	A25	GND
B26	HOLD	A26	GND

Table 2: Pinout for the DESTINATION front panel connectors.

3.2 Timing specifications

The DS_ sent on the output cable is adjusted in time and width with respect to the data word being sent such that it can be successfully used as input to the logic shown in Figure 1 located into a downstream board.

The input FIFO is CY7C4245, 4Kx18 synchronous FIFO from CYPRESS. Two such chips are paralleled to obtain one 4kx36 FIFO with only the first 23 bits being actually used for D[0-20], EP, EE. Data are sampled on the rising edge of the strobe; Figure 1 shows how the DS_ (data strobe) is received from the front panel connector and used to write into the FIFO. The differential lines are sent to the LVDS receivers. Since the FIFO is very fast and sensitive to spikes the gate 74AS32 number 1 has been inserted to filter glitches shorter than few nanoseconds; it adds a 5.8 ns maximum delay on the IDS_ line compared to the data lines that go directly from the receivers to the FIFO. The IDS_ itself is used to produce the active-low write enable line (WEN_) to the FIFO. Since the FIFO is very fast and noise sensitive, Cypress recommends to enable it only for a short time before writing (see Application HandBook, "Understanding Clocked FIFOs", Figure 8, pag. 4-36). A short delay between the WCLK and WEN_ lines is due to the second gate 74AS32.

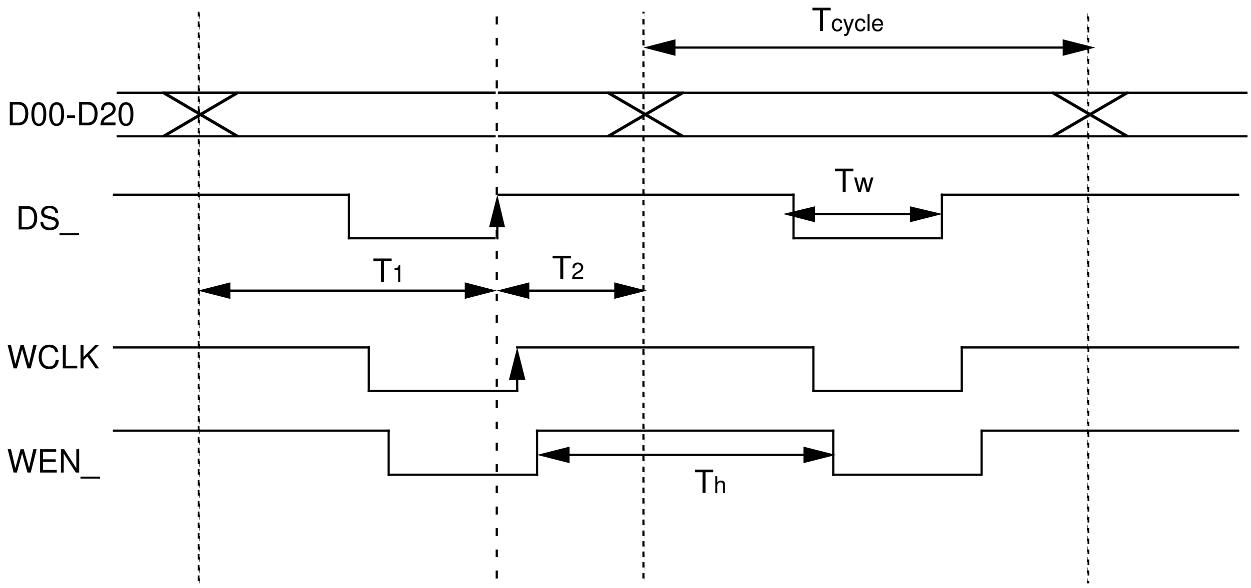

Figure 2: The DS_ signal timing relative to the D bus timing. Timing specs for EP and EE are the same as for the D bus.

Figure 2 shows the timing with which the DS_ signal must be generated by an upstream board to be compatible with the function of writing into the CY7C4245-10. The DS_ signal will enable the clock with the correct set-up time and will strobe data on the rising edge, at the end of the pulse. To understand the limits for T1, T2 and Tw in Figure 2, we need also to consider some of the FIFO switching characteristics shown in table 3 and the 74AS32 switching characteristics.

Description	Min.	Max.	Unit
Clock Cycle Frequency		100	MHz
Data Access time	2	8	ns
Clock Cycle time	10		ns
Clock HIGH time	4.5		ns
Clock LOW time	4.5		ns
Data Set-up time	3		ns
Data Hold time	0.5		ns
Enable Set-up time	3		ns
Enable Hold time	0.5		ns

Table 3: Relevant switching characteristics of the Fifo CY7C4245-10

The minimum clock low time is 4.5 ns, but the width of the signal needs to be at least Tw_min=8.8 ns since WEN_ has a 3 ns Set-up time and can be 5.8 ns delayed compared to the WCLK (gate 74AS32 number 2 in Figure 1). However the small delay of WEN_ (the minimum 74AS32 delay is 1 ns) with respect to WCLK guarantees the Enable Hold time (0.5 ns) is satisfied.

The minimum clock high time the fifo accept is 4.5 ns, but in our case the clock high time Th will be longer since the period is roughly Tcycle=32 ns and Th=Tcycle-Tw.

The minimum value for T1 is given by the Data Set-up time (3ns), while the minimum value for T2 is given by the Data Hold time (0.5 ns) added to the 74AS32 maximum delay (5.8 ns). We add to these numbers three nanoseconds to allow for difference in propagation times between different receivers. In conclusion we recommend the minimum values for T1 and T2:

$$\text{T1\_min} = 6 \text{ ns}; \quad \text{T2\_min} = 10 \text{ ns};$$

As an example for a Tcycle=33 ns, the DS_ rising edge can have a delay T1 between 6 and 23 ns.

4 Data Format

4.1 Packets and Events

On each cable there are 21 *Data* bits, *Data Strobe* (DS_), *HOLD*, *End Packet* (*EP*), *End Event* (*EE*), for a total of 25 signals plus two ground wires (52 wires).

On the cable, data bits are bits 0-20, 21 is EP, 22 is EE, 23 is DS_ and 24 is HOLD, see Table 4.

bit	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
use	HD	DS_	EE	EP	data																				

Table 4: The SVT word. Data field definition is stream dependent.

Data are sent as packets of words. In the simplest case a packet may consist of a single word, otherwise, if more than 21 bits are needed, a packet may consist of two or more words. The EP bit marks the last word of each packet — the *End Packet* word. In one-word packets the word has EP=1.

The EE bit is used to mark the end of the data stream for the current event. The complete sequence of words in a data stream is called an *Event*. The last word of each Event — the *End Event* word — has a special format. It has EE=1 and EP=1. The data field in the *End Event* word is used for Event Tag (8 bits), Parity (PA) (1 bit), Error Flags (10 bits) and Level 2 Buffer number (L2B). The definition of the bits in the End Event word is shown in Table 5.

The EE word contains an 8-bit Event Tag (the bunch crossing number). The Event Tags in different input streams in the same board must be compared to check that they actually belong to the same event. If tags are found to be different, then we know that the system is out of sync and output data is probably meaningless. This error condition is flagged by the error bit *Lost Sync* (see below) in the output stream. The Event Tag must be included

bit	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
use	HD	DS_	EE	EP	L2B		error flags										PA	Event Tag							

Table 5: Data field in the End Event word.

in each EE word in the output stream. If the Event Tags of the input streams are different, than the tag received in one of the input streams is included.

The 1-bit parity code is checked on the fly for every event in order to promptly detect the occurrence of data corruption. The parity code is computed by taking the bitwise XOR of all data words in the event (21 data bits + 1 End Packet bit per word) excluding the *End Event* word. Each SVT board must compute the parity code of incoming data in all input streams and compare it with the code stored in each *End Event* word. Each SVT board must also compute the parity code for the output stream and include it in the *End Event* word.

The implementation of the parity bit is not mandatory for the XTRP-SVT and XTRP-L2 interfaces.

Errors detected within SVT are grouped into a number of classes. One bit is assigned to each error class in the *End Event* word. These error bits can be set by any board and will be propagated downstream to the track list output by SVT so that they will eventually be detected and/or recorded. Error flags [16-18] are yet unassigned but reserved for future use.

1. Parity Error (Bit 9): the parity code in the EE word in one (at least) of the input streams does not check.

2. Lost Sync (Bit 10): the Event Tags from different input streams do not match.

3. Fifo Overflow (Bit 11): in normal operation the mechanism of the communication protocol prevents the input Fifo's from becoming full. If the "Fifo Full" signal is detected by the input control logic, it is a symptom that something is going wrong in the data transfer and that part of the information has probably been lost.

4. Invalid data (Bit 12): a problem with input data is detected.

5. Internal Overflow (Bit 13): data are lost inside a board.

6. Truncated Output (Bit 14): a board truncates intentionally its output data stream because its length exceedes the set limit.

7. G-link Lost Lock (Bit 15): detected by Hit Finders.

The Level 2 Buffer number (L2B) is received from the XTRP and stored in two bits (19-20) (See table 6). These bits are propagated downstream with the same mechanism used to propagate error bits in the EE word. The L2B information will be absent in the data stream going from the Hit Finders to the Mergers (0 filled).

4.2 XTRP to SVT and XTRP to L2 Data Format

The two output streams from the XTRP will be identical. Since the XTRP uses one driver to push the data to both destinations simultaneously, data can be delivered only if both the destinations are ready to receive. Data transmission shall stop when one or both HOLD signals are asserted (high).

The track data will be pushed from XTRP to SVT and L2 as soon as it is available after the L1 accept. The XTRP will implement the HOLD protocol (See section 3).

Each data word describes a single track (see Table 6). The low order 12 bits contain the track ϕ. The XFT produces 9 bits of wedge number (0-287) and 3 bits of mini$-\phi$ (0-7). The XTRP will append the mini-ϕ to the wedge number making one continuous ϕ, 0-2303, which are the 12 bits sent to the SVT and L2. The next 7 bits are the track curvature ($\propto 1/p_t$). The 6 less significant bits are the magnitude of the curvature and the 7th bit is the track's charge. The sign bit will be set for negative charge tracks. The convention is that a curvature value of 1 corresponds to an infinite p_t and a value of 64 corresponds to a p_t of 1.5GeV with steps of equal curvature in between.

The next bit in the word (19) is an isolation bit which is set if less then n tracks were found in the 1.25° wedge, where n is programmable in the XFT linker. Bit 20 is set if the track does not have hits on on the outermost superlayer of the COT. This can be used to indicate a track with large η.

Each track will be a single word, therefore the EP bit will always be on. The last word of each event will have EP and EE on and will contain L2 buffer number (L2B) associated with this event, Error Flags (which are not defined at this time), the Parity bit if it will be provided by XTRP, and the Event Tag, as described in Section 3.

Events are sent in time order. If there are no tracks found by the XFT, an *End Event* word will still be sent.

bit	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0	9	8	7	6	5	4	3	2	1	0
use	HD	DS_	EE	EP	data																				
track				1	η	iso	sign	curvature						ϕ											
track				1	η	iso	sign	curvature						ϕ											
								⋮																	
track				1	η	iso	sign	curvature						ϕ											
end			1	1	L2B		error flags										PA	Event Tag							

Table 6: XTRP to SVT and XTRP to L2 data streams.

4.3 SVT to L2 Data Format

The track data will be pushed from SVT to the L2 receiver board as soon as it is available after the L1 accept. SVT will implement the HOLD protocol (see Section 3).

The order of the tracks is unrelated to the order they were transmitted by the XTRP. Seven data words are used to describe a single track (see Table 7). The first word contains 13 bits of ϕ and 5 bits of z. The second word contains 10 bits of signed impact parameter

(d) and 9 bits of curvature (c). In both fields, the most significant bit is the sign bit. The curvature sign bit is set for negative charge tracks. The third word contains the road ID and the phi sector. The words from fourth to sixth contain the chi square of the fit and the hit coordinates on the five silicon layers. The seventh word contains the Track Fitter status code and the 9-bit track number from XFT. The Track Fitter status code may be used for error flags associated with this track or it may be used for bits indicating this track passed a cut on P_t or impact parameter. The track number is simply the 9 most significant ϕ bits of the XFT track as it is delivered to the SVT, *i.e.* the *XFT wedge number*. Since the XFT can produce only one track per wedge, this uniquely identifies the track and allows L2 to match SVT tracks to XFT tracks.

The seventh word has End Packet asserted because this is the last word for the track.

After all 7-word tracks are sent, the End Event word is sent. Events are sent in time order. If there are no tracks found by the SVT, an End Event word will still be sent.

Alternative output formats may be defined in the future. Bits [18-20] in the first word of each packet are reserved for this purpose and must always be 0 to comply with the format described here.

<table>
<tr><td>bit</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td><td>9</td><td>8</td><td>7</td><td>6</td><td>5</td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td>use</td><td>HD</td><td>DS_</td><td>EE</td><td>EP</td><td colspan="21">data</td></tr>
<tr><td>trk1 wd1</td><td></td><td></td><td></td><td></td><td colspan="3">0</td><td colspan="5">z</td><td colspan="13">ϕ</td></tr>
<tr><td>trk1 wd2</td><td></td><td></td><td></td><td></td><td colspan="2">spare</td><td>sign</td><td colspan="8">c</td><td>sign</td><td colspan="9">d</td></tr>
<tr><td>trk1 wd3</td><td></td><td></td><td></td><td></td><td colspan="4">ϕ sector</td><td colspan="17">Road ID</td></tr>
<tr><td>trk1 wd4</td><td></td><td></td><td></td><td></td><td colspan="7">spare</td><td colspan="5">chi square</td><td colspan="9">$x1$</td></tr>
<tr><td>trk1 wd5</td><td></td><td></td><td></td><td></td><td colspan="3">spare</td><td colspan="9">$x2$</td><td colspan="9">$x3$</td></tr>
<tr><td>trk1 wd6</td><td></td><td></td><td></td><td></td><td colspan="3">spare</td><td colspan="9">$x4$</td><td colspan="9">$x5$</td></tr>
<tr><td>trk1 wd7</td><td></td><td></td><td></td><td>1</td><td colspan="12">Track Fitter Status</td><td colspan="9">track number</td></tr>
<tr><td>trk2 wd1</td><td></td><td></td><td></td><td></td><td colspan="3">0</td><td colspan="5">z</td><td colspan="13">ϕ</td></tr>
<tr><td>trk2 wd2</td><td></td><td></td><td></td><td></td><td colspan="2">spare</td><td>sign</td><td colspan="8">c</td><td>sign</td><td colspan="9">d</td></tr>
<tr><td>trk2 wd3</td><td></td><td></td><td></td><td></td><td colspan="4">ϕ sector</td><td colspan="17">Road ID</td></tr>
<tr><td>trk2 wd4</td><td></td><td></td><td></td><td></td><td colspan="7">spare</td><td colspan="5">chi square</td><td colspan="9">$x1$</td></tr>
<tr><td>trk2 wd5</td><td></td><td></td><td></td><td></td><td colspan="3">spare</td><td colspan="9">$x2$</td><td colspan="9">$x3$</td></tr>
<tr><td>trk2 wd6</td><td></td><td></td><td></td><td></td><td colspan="3">spare</td><td colspan="9">$x4$</td><td colspan="9">$x5$</td></tr>
<tr><td>trk2 wd7</td><td></td><td></td><td></td><td>1</td><td colspan="12">Track Fitter Status</td><td colspan="9">track number</td></tr>
<tr><td></td><td colspan="25">⋮</td></tr>
<tr><td>end</td><td></td><td></td><td>1</td><td>1</td><td colspan="2">L2B</td><td colspan="10">error flags</td><td colspan="2">PA</td><td colspan="7">Event Tag</td></tr>
</table>

Table 7: SVT to L2 data stream.

5 Appendix

The following figure shows a possible implementation of the driver/receiver circuitry. Single ended TTL signals internal to the source and destination boards are prefixed with O (output) or I (input).

FRONT PANEL CONNECTOR PIN ASSIGNMENT

AND DRIVER/RECEIVER SUGGESTED IMPLEMENTATION