Run II XFT Linker Design Specification

Richard Hughes, Brian Winer, Phil Koehn,
Mark Johnson, John Hoftiezer, John Gerstenslager
Ohio State University
October 24, 1997, Version 1.1
Sepember 7, 1998, Version 2.0

1 Overview

The Linker is the second stage of the eXtremely Fast Tracker (XFT). It has the responsibility of linking pixels *between* the axial superlayers of the COT. The pixels themselves are found by the Finder, the first stage of the XFT. The Linker is separated into 288 ϕ slices of $1.25°$. Each slice has a dedicated chip with the task of finding tracks within the slice. The results of the Linker are passed to the XTRP for extrapolation to other portions of the detector and used in the Level 1 trigger decision. The chips will be commercially available programmable logic devices (PLD's). These provide the benefit of being relatively inexpensive and readily available. The programmability adds a tremendous amount of flexibility to the system. This will allow a change of the Level 1 tracking algorithm during the run if necessary and increase our ability to respond to unforseen problems.

This document is the design specification for the Linker portion of the XFT. The Linker system will consist of 24 identical 9U VME cards which reside in three Linker crates (8 cards in each crate). The Linker crates will be located below the Finder crates in the three XFT racks in the first floor counting room. Figure 1 shows the Finder and Linker crates in a single XFT rack. Also, shown are the interconnects between the Finder and Linker.

2 Finder System Summary

The Finder boards are described in detail in [5, 6]. The main points relevant for the Linker system are the following:

1. There are two types of Finder boards: 24 Axial Layer 1-3 boards and 24 Axial Layer 2-4 boards.

2. 1-3 boards contain 2 Finder chips for SL1 and 4 Finder chips for SL3.

3. 2-4 boards contain 3 Finder chips for SL2 and 5 Finder chips for SL4.

4. Each Finder chip processes 4 cells, and the Finder chip output is 12 bits of pixel data/cell every 33 nsec.

5. The Finder board output is sent on 2 identical cables to two different Linker boards. The cables carry data from 6 (or 8) Finder chips, arranged in pairs. Each pair of Finder chip data is driven by a single 28-bit National Channel Link. Each channel link drives 28 bits of data: 24 bits of pixel data from two Finders, and 4 bits of error/diagnostic data.

3 Linker Board

A block diagram of the Linker Card is shown in Figure 2. The parts of the Linker board are:

1. Channel Link receivers: There are 11 channel link receivers per board, numbered 0-10, starting at the top edge of the board. These devices receive the 28 bits of data from the Finder boards, on 5 pairs of differential LVDS lines (4 data, 1 clock). The data arrives at $7 \times 33Mhz$. The receivers demultiplex the incoming data and output 28 single-ended bits of data at 33nsec clock speed. The channel link also outputs a 33 nsec clock in time with the data. Each channel link contains data from 2 Finder chips, except for channel link #11, which contains data from 1 Finder chip.

2. Input Formatter chips: There are 6 Input Formatter chips per board, numbered from 0-5, starting at the top edge of the board. These latch the data from two channel links (except for Input Fomratter #6, which has 1 channel link input) using the channel link 33nsec clocks, then synchronize the data using the on-board 33nsec clock. The output is distributed to the Linker chips. The input chips also contain the pipeline and L2 buffers for the input pixel data.

3. Linker chips: The Linker chips look at the outut of the Input Formatter chips, and search for the best track. There are 12 Linker chips per board, each covering 1.25 degrees.

4. Output Formatter chips: There are 2 of these chips per board. Each chip takes 14 bits of data from 6 Linker chips every 132 nsec, and formats the data into four 26 bit words. The four words are sent over the J3 backplane to the output drivers (located on a transition card in a neighboring slot), 1 word every 33 nsec.

5. Drivers to XTRP The Output Formatter chips send data to LVDS drivers (National DS90C031). The output of the LVDS drivers is sent through the J3 backplane to a passive transition card with 2 60 conductor cables connected to the XTRP.

6. Control chip: VME interface, plus response to Halt-Recover-Run sequence. Also performs loading of Altera devices.

7. Clock circuitry: Generates on-board 33nsec, 66nsec clock, and 132 nsec clocks. The on-board 132nsec clock is phase shifted from the backplane CDF_CLK(132 nsec external clock) using a silicon delay line.

8. Ram for Loading: Stores bit pattern to be loaded onto Altera chips.

9. JTAG interface: The JTAG controller allows a boundary scan of all Linker chips, both Formatter chips, and the VME interface chip. The access to the JTAG circuitry is via a connector on the front panel.

10. BitBlaster ports: Allows downloading of Altera chips using serial port of a PC. Two separate ports: one for the VME chip, the other for serial loading of all other Altera chips.

Each of these are described in detail in the following sections.

4 Control

The control logic consists of 3 main pieces: the VME slave interface, the response to trigger and diagnostic commands, and the loading of the programmable logic devices. In addition, there is the need for the Linker to be synchronized with the rest of the trigger system. Each of these are described below.

4.1 VME Slave Interface

The Linker board will implement a modified version of a VMEbus slave interface. Only 32 bit aligned data transfers will be supported; these may either be single word transfers or block transfers. Only extended (32-bit) addressing modes will be supported. We have implemented and tested a version of this interface using an Altera 10K50 device (note that the the interface uses less than 10% of the resources of this device).

The readout of the Level 2 buffers is accomplished by performing block reads of memory mapped address space. The memory map is detailed in Appendix A.

4.2 Response to Control Signals

The Linker is designed to continually process data at input, provide outputs, respond to L1/L2 accepts, etc. The exception to this is at power-up, or after an error, when the HALT-RECOVER-RUN control sequence is issued.

The response to the HALT-RECOVER-RUN sequence follows guidelines described in reference [7].

- HALT
  - The clocking of the Level 1 pipeline will halt (the enable for the clock will be removed).
- RECOVER
  - Reset will be held high.
  - Counters for the Level 1 pipeline are reset, in both the Input Buffer chip and the Formatter chip.
  - This means the following stages of each Linker chip (each described in 7.4) are reset:
    * RPIX output
    * ROAD_FIND output
    * OR output
    * BEST_TRACK output
  - Bunch synchronization counter is reset.
- RUN
  - The Reset condition will be held high until we observe the B0 marker from the Finder board. Note that this will occur a number of clock ticks after the actual B0 marker is set on the backplane.

The above response will be implemented using a state machine.

4.3 Loading of Programmable Chips

The Linker board will contain the following Altera devices:

- 6 Input Buffer chips implemented using 10K30 chips
- 12 Linker chips implemented using 10K50 chips
- 1 VME Interface + control implemented using a 10K30 chip
- 2 Formatter chips implemented using 10K30 chips

Each of these chips needs to be configured at system power-up. They can also be reconfigured in circuit by resetting the device and loading new data. Reconfiguration takes typically 200ms per device.

We plan to load the devices in the following way. The VME/control chip will be a fixed program and will be loaded using a socketed configuration EPROM (an EPC1 from Altera). If there are errors or changes needed to the VME/control , the

EPROM can be removed and reprogrammed. It is also possible to program the VME chip directly from the serial port of a PC via an Altera BitBlaster connection.

The remaining 20 chips (the 6 Input buffer chips, the 12 Linker chips and the 2 formatter chips) will be loaded using a passive serial scheme. The 12 Linker chips will be implemented using 10K50 devices, which need 609k bits each for programming, while the Input and Output Formatter chips will be implemented using 10K30 devices, which each need 368k bits for programming. The total number of bits required are 10.4MB, and will be stored using six 4-Mbit Flashram (organized as 524,288 Kbytes of 8 bits each), using AMD's AM29F040B chips. Logic in the VME/control chip will toggle the control lines and serialize the data from the Flash RAMs for distribution to the Altera chips.

The procedure for loading the Altera chips on a Linker board will be:

- at power up the VME/control is loaded from the EPROM
- the FLASH rams will be loaded from VME
- the configuration command is given from VME and the remaining Altera chips are loaded.
- if there is a power failure, the chips will reload automatically from EPROM, and then FLASH ram.

An alternate serial programming path is available via the VME interface, through a block transfer. This mode is useful in the test stand, and when modifying and testing new design for the Linker/Input/Output chips. A fully loaded board takes 2 minutes to load using VME block transfers. An final alternate serial programming path is available via a BitBlaster connection (separate from the VME BitBlaster connection).

4.4 Synchronization

There will be an 8-bit bunch counter on each Linker board, following the specification set in CDF Note 3145[9]. This counter will be used to time stamp the data relative to the Bunch Zero signal. The counter output will be latched and stored in the pipeline along with the Linker input and output signal described above. An offset is applied to this counter which indicates the number of crossings used by systems prior to data receipt at the Linker Board (this includes the Finder Board). This offset will be loadable via VME.

5 Input Cables and Channel Link Receivers

We will use National's Channel Link Driver/Receiver pair for the Finder to Linker communication. The plan is to use 1 28 bit channel link per two Finders. The 28 bits include:

- 12 bits of pixel data for Finder 1. These data come in four 33 nsec blocks.
- 12 bits of pixel data for Finder 2. These data come in four 33 nsec blocks.
- 1 B0 marker bit. This is the B0 marker the Finder receives from the mezzanine card, and so marks the B0 crossing as observed at the mezzanine card. This bit is high for 132 nsec.
- 1 start bit. This bit goes high following the start of valid data after the Halt-Recover-Run sequence.
- 1 word-0 marker bit. This bit is high for the 1st 33 nsec of the 132 nsec on the Finder board.
- 1 bit indicating an error condition on the originating Finder board. This bit is high for 132 nsec when there is an error.

The data from a channel link receiver are accompanied by a 33 nsec clock which will be used to register the data at the Input Formatter chips.

Each Linker board receives 2 cables from 1-3 Finder boards, containing data from 3 channel links each, and 2 cables from 2-4 inder boards, containing data from 4 channel links each. There are a total of 14 channel links worth of data input to the Linker board, but data from only 11 channel links is needed for each Linker board. The specification for the cabling from Finder to Linker, including the pin assignments for the channel links, and the signal assigments for the cables, as well as which channel links are needed/board, are described in [1] and [2].

6 Input Formatter Chip

The output of the 11 channel link recievers are brought to 6 Input Formatter chips. Five of the Input Formatters recieve data from 2 channel links each, while the 6th Input Formatter chip receives data from only 1 channel link. The data from a single channel link is registered using a 33nsec clock from that channel link. The data is then aligned to the on-board 33 nsec clock, and output to the Linker chips. The timing diagram for the Input Formatter is displayed in 5.

The first five Input Formatter chips receive 25 bits of data from each of two channel links.. The 25 bits of input are all 24 bits of pixel data, and the word-0 marker bit. The 6th Input Formatter chip receives 13 bits of data from the 11th channel link. The 13 bits of input data are 12 bits of pixel data (from the 1 Finder used by this channel link) and the word-0 marker bit.

The 6th Input Formatter chip also receives the 3 remaining bits (B0 marker, start bit, and error bit) for each of the 11 channel links. This data is registered with the on-board 132 nsec clock. The 6th Input Formatter chip is responsible for all of the error checking on the board. The following error checks are performed:

1. Start bit is the same on each channel link (11 total)
2. word 0 present at correct time on each channel link (11 total)
3. error present on any Finder board
4. B0 marker at correct time relative to backplane B0 marker, on each channel link (11 total)

The 6th Input Formatter chip stores the following error summary data in the Level 1 pipeline/Level 2 buffers:

1. latch if any error present (1 bit)
2. latch which cable the error came from, as well as which of 4 types of error on cable (unless start bit wrong across cables) (16 bits)
3. total count of errors (7 bits)

Each Input Formatter chip also stores the input pixel data in a Level 1 pipeline, and each chip contains 4 Level 2 buffers. The Level 2 buffers can be read out upon request via the VME interface. The VME address map is detailed in the Appendix.

We have implemented both types of Input Formatter chips using Altera 10K30 devices. The type for Input Formatters 1-5 uses 88% of available logic cell and 6/6 available memory blocks. The type for Input Formatter 6 uses xx% of available logic cell and 6/6 available memory blocks. We have fully simulated the functionality of the chips, including the L1 pipeline and L2 buffering. The timing diagram for the Input Formatter chip is displayed in Figure 5. The pin assignment for Input Formatter chips 1..5 are listed in Tables 4, and 5. The pin assignment for Input Formatter chip 6 is listed in Tables 6, and 7.

7 Linker Chip Design

Each Linker chip will be responsible for finding tracks within a 1.25° slice of the azimuthal angle phi. Each chip will locate the subset of tracks that pass through superlayer 6 (the third axial superlayer) within the slice.

7.1 Linker Chip Inputs

7.2 Input Mapping

Each of the first 5 Input Formatters processes two channel links, or 4 Finders worth of pixel information. The 6th Input Formatter processes 1/2 channel link or 1 Finder worth of pixel information. The pixel information from these 21 Finders is routed from the Input Formatter chips to the 12 Linker chips. Each Linker chip requires either 8, 9, or 10 Finder chips worth of pixel information, and many pixels are shared among several Linker chips. References [4] and [3] lists the Finders for each Linker board, as well as which subset of Finders feed each Linker chip on each board. Note that all boards are identical, but that the inputs for each Linker chip vary for each chip. (This implies that the Altera programming for each Linker chip is different, for the input sections only.)

7.3 Linker Chip Outputs

The Linker outputs information on the best track found every 132nsec. The output is 8 bits of P_t, 3 bits of ϕ, and 3 bits of code. The eight P_t bits are 3 bits of mini-P_t and 5 bits of section P_t. Only 7 bits are required to uniquely specify a P_t, but for space reasons this conversion is done in the Output Formatter chip.

7.4 Linker Chip Block Diagram

Figure 3 shows a block diagram of the functionality of the Linker chip. Each of the major blocks is implemented as a single routine written in AHDL (Altera's High Definition Language). The functionality of each block is described in broad terms below.

7.4.1 RPIX

Each finder sends the information from the 4 COT cells it processes to the Linker in 4 12-bit words, 1 word every 33nsec. These words then have to be time-demultiplexed so the linker can look at all cells in parallel, for all 4 superlayers. As mentioned earlier, the inputs to the 12 Linker chips are slightly different, and so this routine is different for each Linker chip. There is a two-fold symmetry in the Linker, which allows the number of stored roads to be cut in half. Effectively, each Linker is then arranged as two sub-linkers, which we call A and B. To take advantage of this, RPIX must latch the pixels which are used as input to the two sub-linkers. The following tasks are then performed:

- Latches the input data on the rising edge of 33ns clock
- Reformats the pixels into sublinkers A and B. Each sublinker covers one half of the 1.25° slice.
- Outputs the pixels for linker A on rising edge of 66ns clock
- Outputs the pixels for linker B on the falling edge of 66ns clock.

7.4.2 ROAD_FIND(4 and 3 layer)

The pixels for Linker A and B are passed on to the road finding logic (Altera routine RD48SL4.TDF and RDSL3.TDF). As mentioned above, by taking advantage of a two-fold symmetry in the Linker, only half the roads need to be implemented in the logic. Only 1229 4-layer Roads and 204 3-layer Roads are implemented in the chip logic. Each road is stored as the "AND" of 4 (or 3) pixels, one from each layer. Four-layer and three-layer roads are found in parallel. The four-layer road finding outputs a 96 bit P_t word every 66 ns. Each bit in this word corresponds to a valid track found with a given P_t bin. The three-layer road finding outputs a 32 bit Pt word every 66 ns. The number of output bits is smaller because the P_t resolution is worse using only 3 layers rather than 4. The P_T values of the bins for 4-layer tracks are listed in Table 1.

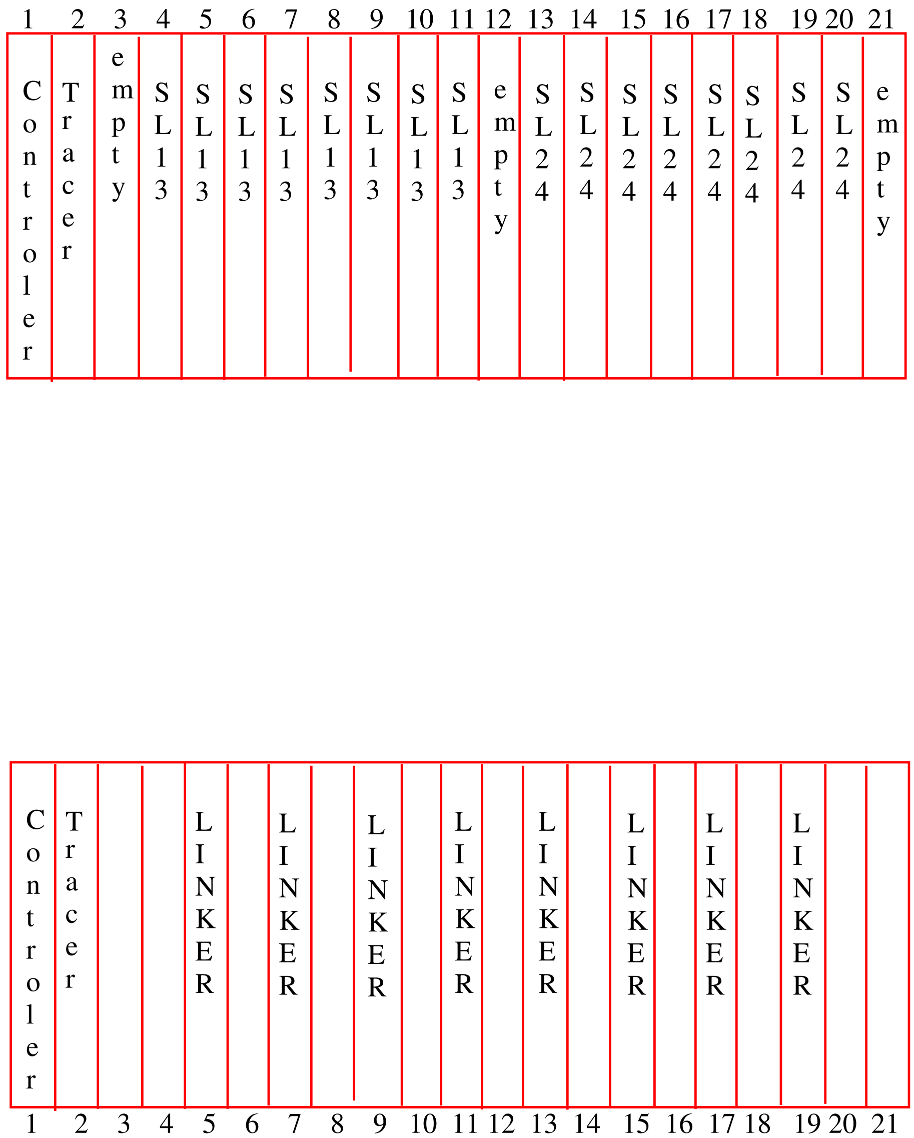

Figure 1: Finder and Linker Crate.

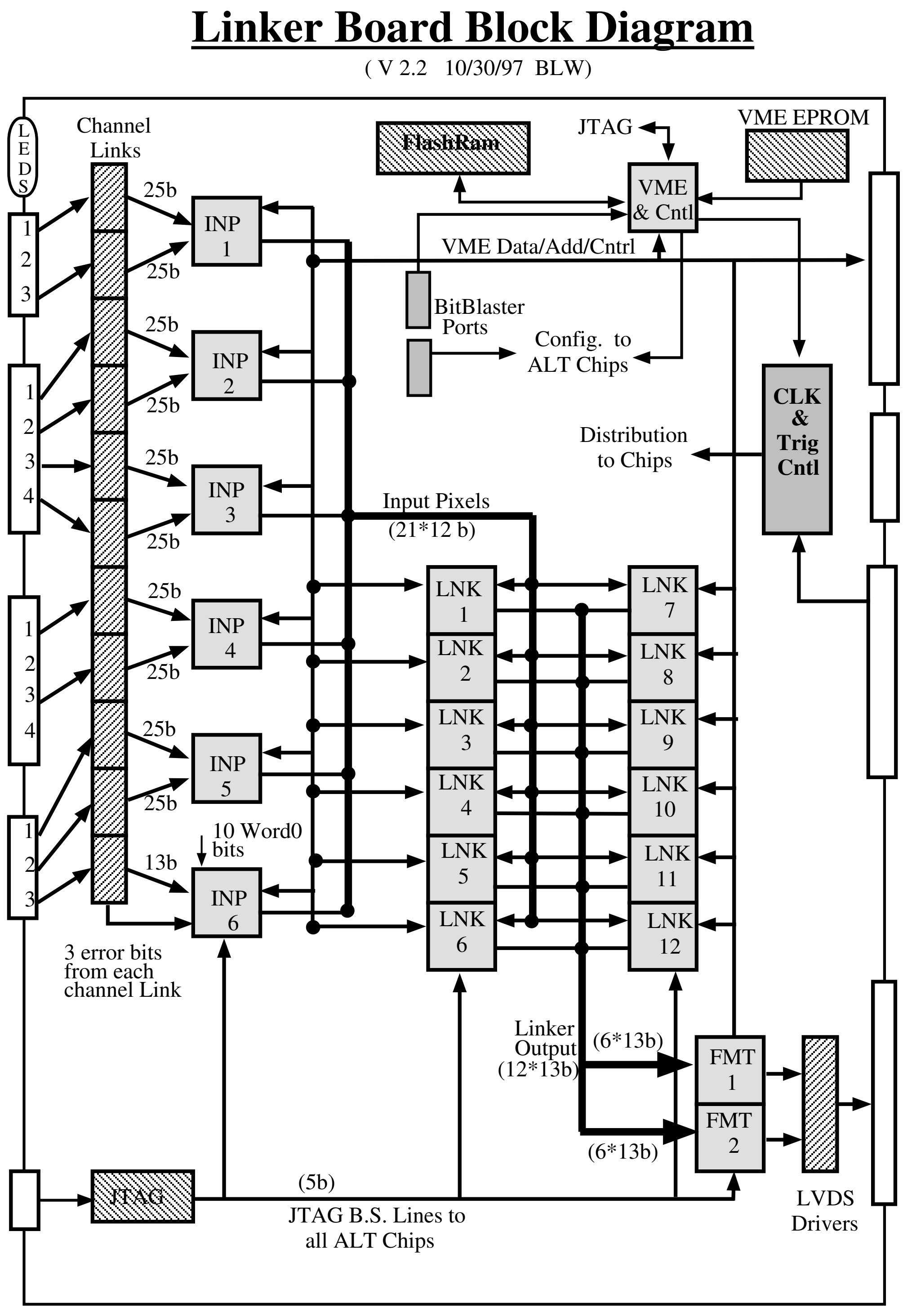

Figure 2: Linker Board Block Diagram.

V3.0 (10/31/97) BLW

XFT Linker Chip Block Diagram

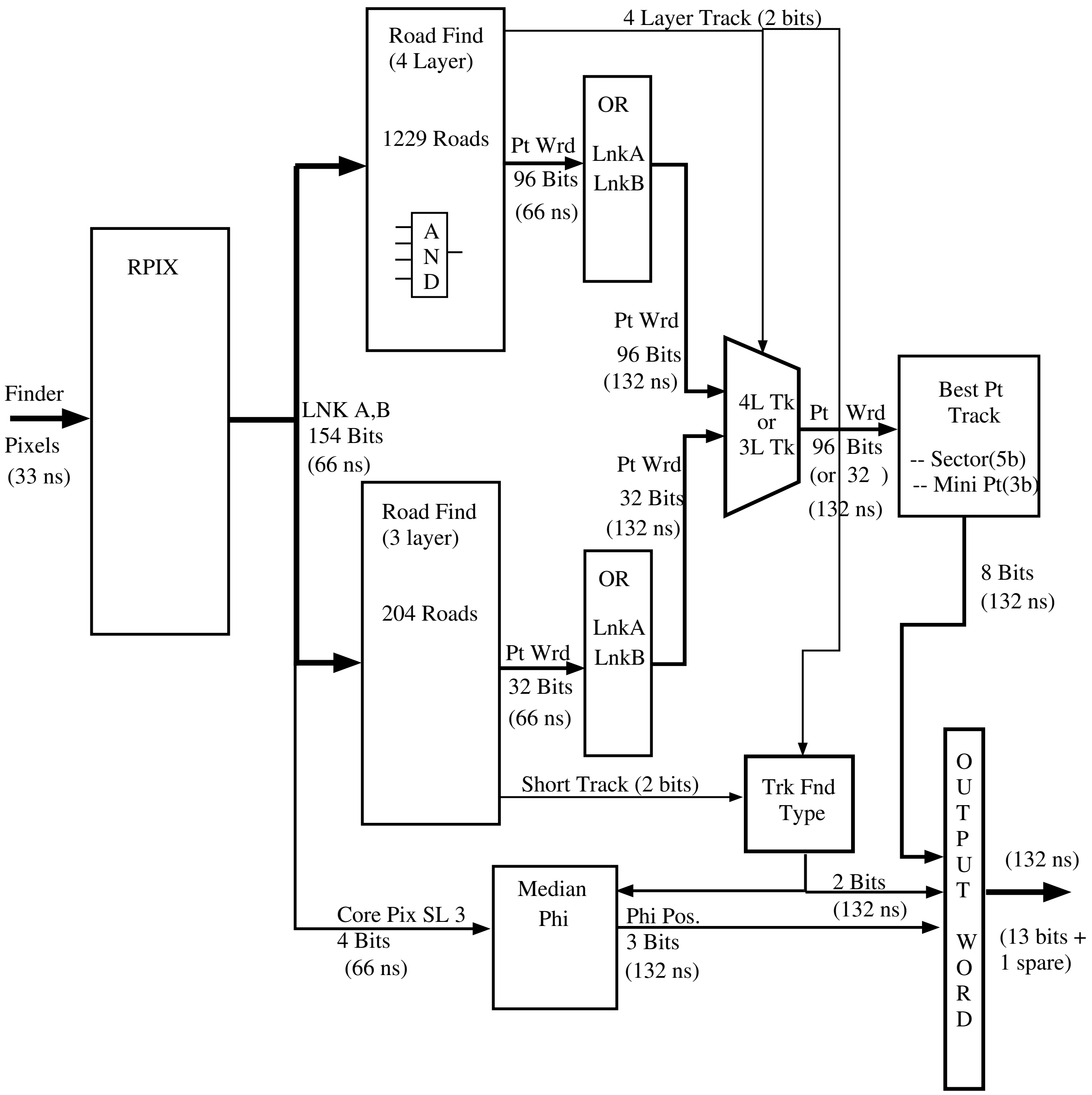

Figure 3: Linker chip block diagram.

7.4.3 OR (4 and 3 layer)

The P_t words for sublinker A and B are "OR'ed" together.

7.4.4 MUX

The inputs to this section are the 96 bit word from the 4 layer track finding and the 32 bit word from the 3 layer track finding. If a 4 layer track is found, then it is output from the multiplexor, otherwise the 3 layer track word is output.

7.4.5 BEST_PT

Either the 96 (4 layer) or 32 (3 layer) bit P_t word is presented to the "Best P_t Track" logic (Altera routine LPOUT48.TDF) which determines what P_t to report. Its output is a 8 bit word with Pt and charge information: 3 bits of mini-P_t and 5 bits of section P_t. This set of logic is fairly complicated. It must of course handle the simplest case, where 1 track has passed through a given linker chip. Since the Finder tends to report more than one pixel per layer even for isolated tracks, the Linker will tend to find multiple tracks (all neighboring in P_t bins). The BEST_PT logic is designed to report the median P_t bin of this cluster of P_t bins. In addition, the BEST_PT logic must handle the case where more than one track has passes through a given Linker chip. The bias at present when this occurs is to select the cluster of P_t bins which is higher in P_t and then report the median of this cluster.

7.4.6 MEDIAN_PHI

This section looks at the 8 pixels reported from the FINDER for superlayer 3 and determines a simple median of the pixels, outputing a 3 bit word. Remember that tracks are found in sub-linkers A and B. If no track is found in a given sub-linker, the pixels for superlayer 3 for that sub-linker are masked out. Beyond this, the correlation between the phi position and the reported P_t is lost (note that this is on average a 1 pixel shift).

7.5 Linker Chip Timing Diagram

Figure 4 shows the timing diagram for the Linker chips processes. From input to output, the total number of 132nsec clock ticks needed by the Linker chip is 4.5.

7.6 Linker Chip Implementation in PLD

As mentioned earlier, each of the 12 Linker chips on a board are slightly different, since the number of Finders needed per chip varies. All 12 versions of the Linker chip have been implemented using the Altera 10K50 device. They have all been fully simulated,

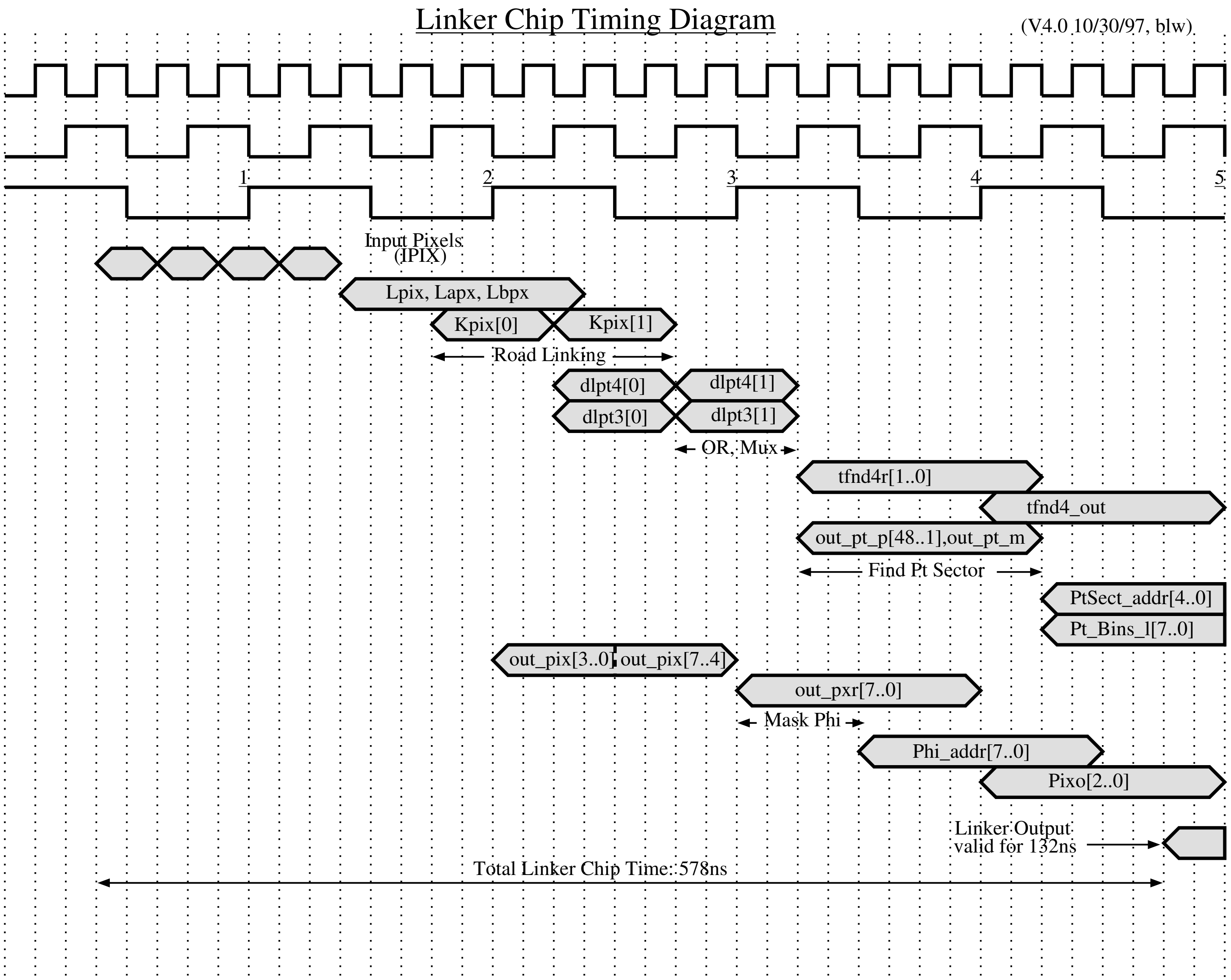

Figure 4: A simple timing diagram for the linker chip processing.

and using worst case timing can run as fast as 30nsec (they need to run at 33nsec). The devices usage is:

- 2391 Logic Elements used, 86% of available elements.
- 5 out of the available 10 EAB's are used, although the total memory usage is 3328 bits (16% of available memory).

The resource usage does not vary much across the 12 Linker types. The pin assignment for all 12 Linker chips is the same, and is listed in Tables 10, and 11.

8 The Formatter Chip

The role of the Formatter chip is to prepare the tracking data out of the XFT for input to the extrapolating unit (XTRP). First, the Formatter chip receives a new list of tracks from a set of Linker chips every 132ns. Next, duplicate tracks near the boundaries of the region covered by each Linker chip are removed. Finally, the tracking information is multiplexed into 4 data words that are sent on sequentially (every 33ns) to the XTRP.

8.1 Format Chip Input

Every Linker board contains 12 Linker and 2 Formatter chips. The 12 Linkers are divided into two sets of 6, and the output of each set of 6 is routed to a single Output Formatter chip. Because a Linker chip puts out track information as a 14 bit word every 132ns, each Formatter chip receives one 84 bit word (6×14 bits) every 132ns. The 14 bit Linker word (the format is described in detail above) includes 8 bits of P_T and 3 bits of ϕ information. Within the 84 bit word, the 6 Linker words are ordered in increasing ϕ with increasing Linker number. Since each Linker covers an angular region of $1.25°$ in ϕ, the total coverage for each Formatter chip is $7.5°$˙

8.2 Formatter chip L1 Pipeline/L2 Buffers

This section of logic stores the Linker Chip output data (a total of 84 bits for all 6 Linker chips/formatter chip). The Level 1 pipeline is implemented using a programmable-depth ring buffer (up to 128 deep). Two 32 bit words and one 20 bit word are written into the buffer each 132 nsec. The buffer feeds 4 Level 2 buffers.

8.3 P_T word compression

As mentioned in the Linker chip section above, each Linker presents the Formatter chip with an 8-bit P_T word. The 8 bits actually correspond to 96 separate P_T bins, so only 7 bits are needed to specify the P_T. This is a simple shift and add operation, but due to timing and space considerations this is easier to implement in the Formatter rather than in each Linker chip.

8.4 Duplicate Track Removal

Often, a track near the edge of the region covered by a Linker chip will also be found by the neighboring Linker. Keeping both tracks may cause problems with the trigger. This situation is resolved by keeping only one of the *duplicate* tracks. Duplicate track removal is performed by the Formatter chip across the Linkers input to that Formatter chip. The XTRP will remove duplicate tracks for Linkers between Formatter chips as well as at Linker board boundaries.

The duplicate track removal algorithm begins by checking the ϕ bits for each Linker word to determine the angular position of the track. The 3 ϕ bits in the 14 bit Linker word give 8 bin positions (0 - 7) spanning 1.25°. The ϕ bin arrangement of adjacent Linker chips is illustrated in Table 2. Duplicate tracks are identified by comparing phi bin (0-7) on Linker N to Linker N-1. If Linker N-1 has a phi position within 2 bins of Linker N, the two tracks are considered to be duplicates. To remove the duplicate, the P_T of the track of Linker N-1 is set to zero. For example, if Linker-6 has phi-bin 0 registered and Linker-5 has phi-bin 7 registered, the P_T for Linker 5 is set to zero.

8.5 Formatter Chip Output

After duplicate tracks are removed, the Linker information is reformatted before it is sent to the XTRP. The data out of the format chip will consist of 4, 30 bit words, each clocked out at 33ns. The format of the output words is mapped to the data sent over a 60 pin cable to the XTRP. The cable format is given in Table 3:

Pt#(7) is the 7 bits of Pt, Mk(2) is the word marker count (1 - 4), B0(1) is the beam zero marker, SP indicates spare bits (2 in the first 3 words and 27 in the last word), Ph# is the 3-bit phi word, S is the short track bit, I is the isolated track bit, X is an extra bit left over for future use. The order of the output words is important so that enough time is permitted for the XTRP to remove duplicate tracks at board boundaries.

8.6 Formatter Chip Implementation and Simulation

We have implemented the Formatter algorithm using the Altera 10K30RC240-4, which is a 30,000 gate device. The simulation was performed on the formatter chip consisted of sending the chip a 84-bit long word that contained the pt+phi words for 6 Linkers. The output of the 1st word occurs $\sim$ 80 ns after the 84 bit Linker word arrives, and so the Formatter adds 0.75 132nsec clock ticks to the Linker board timing.. The timing diagram shown in Figure 5 illustrates the performance of the formatter chip. The pin assignment for Output Formatter chip is listed in Tables 12, and 13.

9 Linker Board Output

The output of each formatter chip is sent to a National DS90C031 LVDS differential driver. Each driver sends the data on to the XTRP over a 60 conductor flat cable. Each signal of the 30 bit output is paired a separate ground conductor. The drivers will be located on the Linker board, and will send the information to the Linker transition card, which at present is a passive card with connectors only. The data format on the two cables is described in the Formatter chip section above.

Each Linker board sends (57 bits/formatter chip $\times$ 2 Formatters per 132 nsec) for a bandwidth of 8.64×10^8 bits/second

10 Linker Board Clock

The Linker chip expects 3 clocks on input: 132nsec, 66nsec, and 33 nsec. The Finder data is clocked in at 33nsec, and the linker data is clocked to the XTRP module on output (from the Formatter chips) at 33 nsec. We plan on generating these clocks either from hardware or software control.

In hardware mode, the clocks will be generated on each Linker board. First, a phase-shifted 132nsec clock will be generated from CDF_CLOCK, using the silicon delay chip DS1020 from Dallas Semiconductor. This chip allows delays of up to 256nsec, in 1nsec increments. This 132nsec clock will then be used as a reference clock input to the Cypress Programmable Skew Clock Buffer CY7B991 (aka RoboClock). Roboclock divides down the 132nsec clock to produce a 33nsec and 66nsec clock, each with zero phase difference with the reference.

In software mode, the clock source is VME. By writing to a control register, we send "33 nsec" steps are then sent from VME (this clocks a synchronous counter to generate 33, 66, 132nsec software clocks).

11 Linker Board Operating Modes

The Linker board will have a number of operating modes. These modes are determined both by the programs loaded in the various Altera chips, as well as by the setting of 6 read-write control lines in the VME/Control chip. The control lines are split into 3 groups of 2, with 2 lines (INP_CTL0,1) going to the 6 Input Formatter chips, 2 lines (LNK_CTL0,1) going to the 12 Linker chips, and 2 lines (OUT_CTL0,1) going to the 2 Output Formatter chips.

11.1 Normal Mode

In this mode, the Linker board is configured as it would be during normal data taking.

The program codes for the chips, as stored in the VME/Control chip register addresses 0xYY00024 and 0xYY000028 (see the memory map in appendix A for the bit defintions), should be:

1. Input Formatter Chip code: 1 (normal).
2. Linker Chip program code: 1 (normal).
3. Output Formatter Chip code: 1 (normal).

The control lines should have the following settings:

1. INP_CTL[1..0] = 0.
2. LNK_CTL[1..0] = 0.
3. OUT_CTL[1..0] = 0.

11.2 Diagnostic Mode

In this mode, the programs for each of the chips is the same as used during normal data taking. The function of the Input and/or Output Formatters is changed according to the control lines from VME.

In diagnostic mode, the memory used for the Level 1 pipeline in the Input Buffer chip is instead used as a buffer memory. This memory could either be read out from VME, to check the data path from the Finder (cables and channel links), or it could be loaded from VME for storing test vectors. The test vectors will have the same configuration as data from the Finders.

In a similar way, the memory used for the Level 1 pipeline in the Formatter chip can be used as buffer memory. The main use here will be to store the output of the Linker chips as test vectors are clocked in from the Input Buffer chip. This memory will then be read out from VME.

The program codes for the chips, as stored in the VME/Control chip register addresses 0xYY00024 and 0xYY000028 (see the memory map in appendix A for the bit defintions), should be:

1. Input Formatter Chip code: 1 (normal).
2. Linker Chip program code: 1 (normal).
3. Output Formatter Chip code: 1 (normal).

The control lines should have the following settings:

1. INP_CTL[1..0] = 0 (normal) or 1 (diagnostic memory use).
2. LNK_CTL[1..0] = 0.
3. OUT_CTL[1..0] = 0 (normal) or 1 (diagnostic memory use).

11.3 XTRP Cable-Test Mode

In this mode, the Linker board is configured to test the XTRP cables.

The program codes for the chips, as stored in the VME/Control chip register addresses 0xYY00024 and 0xYY000028 (see the memory map in appendix A for the bit defintions), should be:

1. Input Formatter Chip code: 1 (normal program).
2. Linker Chip program code: 1 (normal program).
3. Output Formatter Chip code: 3 (XTRP-cable test program).

The control lines should have the following settings:

1. INP_CTL[1..0] = 0.
2. LNK_CTL[1..0] = 0.
3. OUT_CTL[1..0] = 0.

11.4 Through-Test Mode

In this mode, the Linker board is configured to test the interconnections from the Input Formatter chips to the Linker chips. The test also verifies that the data from Input to Linker chips can be registered at 33nsec.

The program used for the Input Formatter chips is the normal one, while the Linker and Output Formatter chips use specialized prgrams. The Input chip is put into its diagnostic mode (by using the control lines), and data patterns are written into its memory via VME writes. These test patterns are then sent to the Linker chips. The Linker chip programs are designed to allow the user to select by VME any of the up to 10 Finder chip inputs, and route that data to the Linker chip output. In addition, the phase (meaning which of the 4 33 nsec cycles) of the Finder input to capture can be selected by VME. The Output chip has a specialized program to store the Linker input in memory, for later readout from VME.

The program codes for the chips, as stored in the VME/Control chip register addresses 0xYY00024 and 0xYY000028 (see the memory map in appendix A for the bit defintions), should be:

1. Input Formatter Chip code: 1 (normal program).
2. Linker Chip program code: 2 (through-test program).
3. Output Formatter Chip code: 2 (through-test program).

The control lines should have the following settings:

1. INP_CTL[1..0] = 0.
2. LNK_CTL[1..0] = 0.
3. OUT_CTL[1..0] = 0.

12 JTAG

JTAG-Boundary Scan is a serial protocol, used to test pins on chips which have implemented its interface. Each of the 21 altera devices implement the JTAG protocol. We can access the JTAG ports on the Altera chips via a serial connector on the front-panel. The connector attaches to a TI-XXXX Scan Path Linker. This device allows us to break the JTAG serial chain into 4 separate paths, one for the VME interface, one for the 6 Input Formatter chips, one for he 12 Linker chips, and one for the 2 output Formatter chips. The following tests are possible:

- Individual input pins can be set for all Altera devices (and any other JTAG-capable chips on the board).
- All I/O pins can then be read back for shorts to GND/VCC or across pins.
- By setting the clock pins, we can also clock test vectors through the device.

13 Linker Board Timing Diagram

Figure 5 shows the timing diagram for the Linker board processes. From first Finder cell input to the 1st word output to the XTRP, the total number of 132nsec clock ticks needed by the Linker board is 5.5.

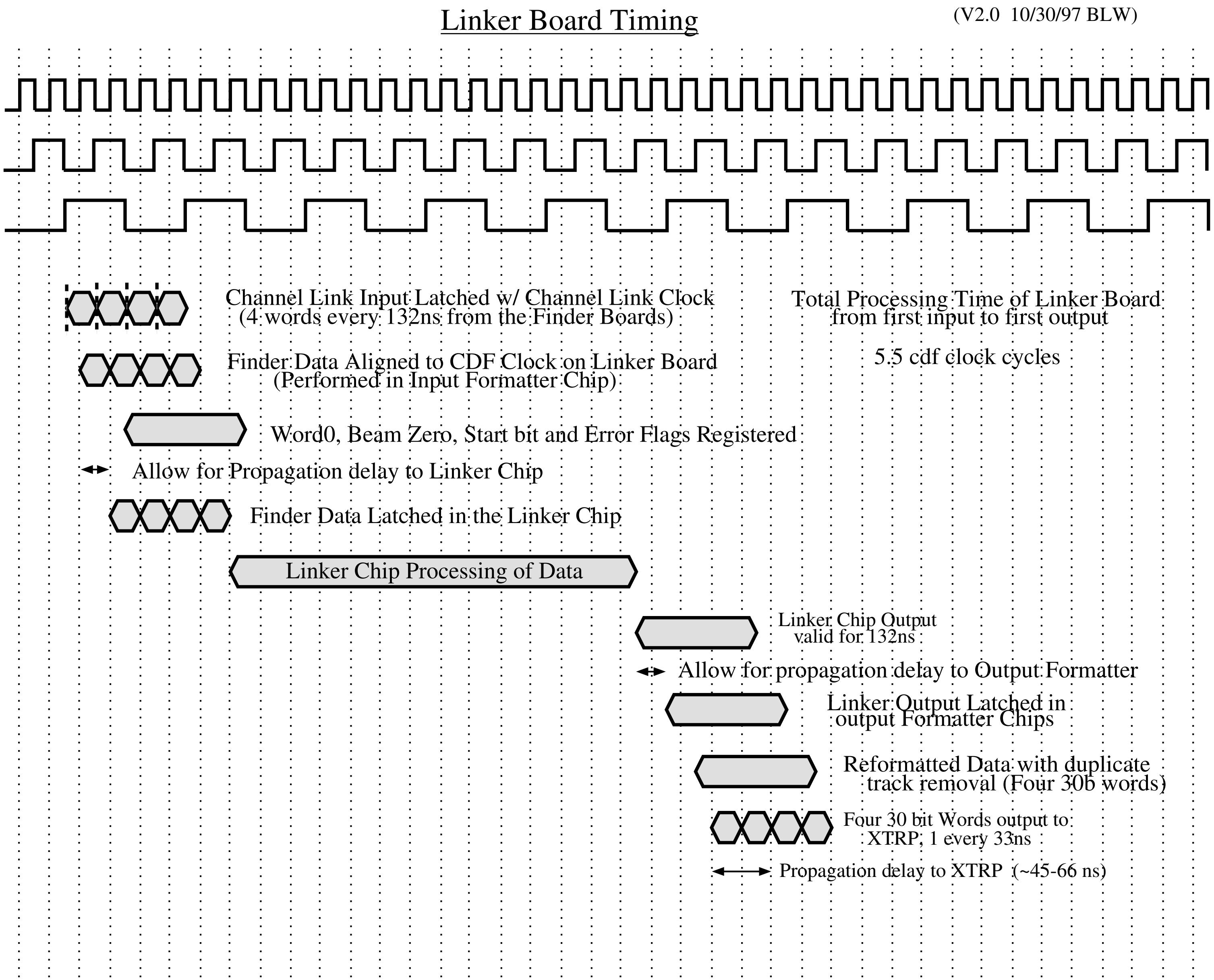

Figure 5: A simple timing diagram for the linker board processing.

Bin Number	P_T	Bin Number	P_T
47	Infinite	48	-Infinite
46	64.00	49	-64.00
45	38.40	50	-38.40
44	27.43	51	-27.43
43	21.33	52	-21.33
42	17.45	53	-17.45
41	14.77	54	-14.77
40	12.80	55	-12.80
39	11.29	56	-11.29
38	10.11	57	-10.11
37	9.14	58	-9.14
36	8.35	59	-8.35
35	7.68	60	-7.68
34	7.11	61	-7.11
33	6.62	62	-6.62
32	6.19	63	-6.19
31	5.82	64	-5.82
30	5.49	65	-5.49
29	5.19	66	-5.19
28	4.92	67	-4.92
27	4.68	68	-4.68
26	4.47	69	-4.47
25	4.27	70	-4.27
24	4.09	71	-4.09
23	3.92	72	-3.92
22	3.76	73	-3.76
21	3.62	74	-3.62
20	3.49	75	-3.49
19	3.37	76	-3.37
18	3.25	77	-3.25
17	3.15	78	-3.15
16	3.05	79	-3.05
15	2.91	80	-2.91
14	2.74	81	-2.74
13	2.59	82	-2.59
12	2.46	83	-2.46
11	2.34	84	-2.34
10	2.23	85	-2.23
9	2.13	86	-2.13
8	2.04	87	-2.04
7	1.96	88	-1.96
6	1.88	89	-1.88
5	1.81	90	-1.81
4	1.75	91	-1.75
3	1.68	92	-1.68
2	1.63	93	-1.63
1	1.57	94	-1.57
0	1.52	95	-1.52

Table 1: P_T at bin center vs bin number.

	Linker(N+1)	Linker(N)	Linker(N-1)
ϕ bin position	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0	7 6 5 4 3 2 1 0

Table 2: Arrangement of phi bins for adjacent Linker chips. Note: the direction of increasing ϕ is the direction of increasing Linker number.

pin(+)	pin(-)	word0	word1	word2	word3
1	2	GND	GND	GND	GND
3	4	mk0	mk0	mk0	mk0
5	6	mk1	mk1	mk1	mk1
7	8	ph0_0	ph4_0	ph2_0	X0
9	10	ph0_1	ph4_1	ph2_1	X1
11	12	ph0_2	ph4_2	ph2_2	X2
13	14	S0	S4	S2	X3
15	16	I0	I4	I2	X4
17	18	Pt0_0	Pt4_0	Pt2_0	X5
19	20	Pt0_1	Pt4_1	Pt2_1	sp
21	22	GND	GND	GND	GND
23	24	Pt0_2	Pt4_2	Pt2_2	b0
25	26	Pt0_3	Pt4_3	Pt2_3	err
27	28	Pt0_4	Pt4_4	Pt2_4	sp
29	30	Pt0_5	Pt4_5	Pt2_5	sp
31	32	Pt0_6	Pt4_6	Pt2_6	sp
33	34	ph1_0	ph5_0	ph3_0	sp
35	36	ph1_1	ph5_1	ph3_1	sp
37	38	ph1_2	ph5_2	ph3_2	sp
39	40	GND	GND	GND	GND
41	42	S1	S5	S3	sp
43	44	I1	I5	I3	bcnt0
45	46	Pt1_0	Pt5_0	Pt3_0	bcnt1
47	48	Pt1_1	Pt5_1	Pt3_1	bcnt2
49	50	Pt1_2	Pt5_2	Pt3_2	bcnt3
51	52	Pt1_3	Pt5_3	Pt3_3	bcnt4
53	54	Pt1_4	Pt5_4	Pt3_4	bcnt5
55	56	Pt1_5	Pt5_5	Pt3_5	bcnt6
57	58	Pt1_6	Pt5_6	Pt3_6	bcnt7
59	60	GND	GND	GND	GND

Table 3: Pin assignment and word mapping for the XFT Linker to XTRP cable.

A APPENDIX: Memory Map

The memory map follows the guidelines described in CDF Note 2388[10] and CDF Note 3145[9].

```
Memory Map:

General:
  YY00 0000 Diagnostic register (32 bits) (R/W)

  YY00 0004 Control (R/W)
Bit  Function
 31  Reset
 30  Software clock enable
 29  Software clock reset
 28  Software clock toggle
 27  Enable Error flag to CDF_ERR
  8  Program from flash
  5  input control line 1
  4  input control line 0


  YY00 0008 Pipeline Length (R/W)
  This equals 42 minus the number of stages before the Linker.
Bit     Function
24-31   Programmable pipeline depth

  YY00 000C Pipeline offset (R/W)
  This equals the number of pipeline stages in the Linker.

  YY00 0010 User diagnostic register(R/W)
Bit     Function
 26     diagnostic counter reset
 25     generate N 33 nsec counts
 24     generate 33 nsec counts in infinite loop
 0-7    Maximum count number of 33nsec counts to generate


  YY00 0014 Flash Load write enable(W)
Bit     Function
                7-0     Data to be written
       10-8     FLASH Ram write code
```

```
  YY00 0018 Flash Load read enable(W)
Writing to this address generates a read.

  YY00 001C Flash Load RAM enable(R/W)
Bit     Function
                    0     RAM enable for FLASH-ram 1
                    1     RAM enable for FLASH-ram 2
                    2     RAM enable for FLASH-ram 3
                    3     RAM enable for FLASH-ram 4
                    4     RAM enable for FLASH-ram 5
                    5     RAM enable for FLASH-ram 6

  YY00 0020 (R/W)
        Bit    Function
13-22  Board Number
23-31  Board Type

  YY00 0024 (R/W)
        Bit    Function
0-11   VME chip program code
       12-23   Linker chip program code

  YY00 0028 (R/W)
        Bit    Function
 0-11  Input chip program code
12-23  Output chip program code



Flash Ram
  YY60 0000 Flash RAM write register
Bit    Function
                 0-7   Data
                8-26   Address to write to

  YY61 0000 Flash RAM read register (read only)
Bit    Function
                 0-7   Data


Memories (R/W)
```

```
  YY50 0XXX Input Formatter 1 (only 9 address bits, 0-23 data bits)
  YY51 0XXX Input Formatter 2 (only 9 address bits, 0-23 data bits)
  YY52 0XXX Input Formatter 3 (only 9 address bits, 0-23 data bits)
  YY53 0XXX Input Formatter 4 (only 9 address bits, 0-23 data bits)
  YY54 0XXX Input Formatter 5 (only 9 address bits, 0-23 data bits)
  YY55 0XXX Input Formatter 6 (only 9 address bits, 0-23 data bits)
  YY56 0XXX Output Formatter 1 (only 9 address bits, 0-23 data bits)
  YY57 0XXX Output Formatter 2 (only 9 address bits, 0-23 data bits)


L2 buffers (Read only)
  L2 buffers start at address YYZ0 0000, where
Z=8 for buffer 0
Z=9 for buffer 1
Z=A for buffer 2
Z=B for buffer 3
  YYZ0 0000 L2 Buffer Header word
Bit    Function
0-7    Bunch ID: 8 bit counter from Bunch Zero
8-12   Geographical address
13-22  Board Number
23-31  Board Type
  YYZ0 0004 Program codes 1
        Bit    Function
0-11   VME chip program code
       12-23   Linker chip program code
  YYZ0 0008 Program codes 2
        Bit    Function
0-11   Input chip program code
       12-23   Output chip program code

  YYZ0 000C Linker Board Error register (bits 0-23)
Bit    Function
                0      Error present on Linker board
                31-24  Error count

  YYZ0 0010 Linker 0 and Linker 1 Output (bits 0-23)
Bit    Function
0-6    Pt Linker 0
7-9    Phi Linker 0
10     Isolation bit Linker 0
11     Short track bit Linker 0
```

```
12-18  Pt Linker 1
19-21  Phi Linker 1
22     Isolation bit Linker 1
23     Short track bit Linker 1
  YYZ0 0014 Linker 2 and Linker 3 Output (bits 0-23)
Bit    Function
0-6    Pt Linker 2
7-9    Phi Linker 2
10     Isolation bit Linker 2
11     Short track bit Linker 2
12-18  Pt Linker 3
19-21  Phi Linker 3
22     Isolation bit Linker 3
23     Short track bit Linker 3
  YYZ0 0018 Linker 4 and Linker 5 Output (bits 0-23)
Bit    Function
0-6    Pt Linker 4
7-9    Phi Linker 4
10     Isolation bit Linker 4
11     Short track bit Linker 4
12-18  Pt Linker 5
19-21  Phi Linker 5
22     Isolation bit Linker 5
23     Short track bit Linker 5
  YYZ0 001C Duplicate track info, Linkers 0-5
Bit    Function
0      Duplicate track flag Linker 0
1      Duplicate track flag Linker 1
2      Duplicate track flag Linker 2
3      Duplicate track flag Linker 3
4      Duplicate track flag Linker 4
21     Linker Board error flag
23     beam zero marker
  YYZ0 0020 Linker 6 and Linker 7 Output (bits 0-23)
Bit    Function
0-6    Pt Linker 6
7-9    Phi Linker 6
10     Isolation bit Linker 6
11     Short track bit Linker 6
12-18  Pt Linker 7
19-21  Phi Linker 7
22     Isolation bit Linker 7
```

```
23      Short track bit Linker 7
  YYZ0 0024 Linker 8 and Linker 9 Output (bits 0-23)
Bit     Function
0-6     Pt Linker 8
7-9     Phi Linker 8
10      Isolation bit Linker 8
11      Short track bit Linker 8
12-18   Pt Linker 9
19-21   Phi Linker 9
22      Isolation bit Linker 9
23      Short track bit Linker 9
  YYZ0 0028 Linker 10 and Linker 11 Output (bits 0-23)
Bit     Function
0-6     Pt Linker 10
7-9     Phi Linker 10
10      Isolation bit Linker 10
11      Short track bit Linker 10
12-18   Pt Linker 11
19-21   Phi Linker 11
22      Isolation bit Linker 11
23      Short track bit Linker 11
  YYZ0 002C Error checking and Duplicate track info, Linkers 6-11
Bit     Function
0       Duplicate track flag Linker 0
1       Duplicate track flag Linker 1
2       Duplicate track flag Linker 2
3       Duplicate track flag Linker 3
4       Duplicate track flag Linker 4
23      beam zero marker

  YYZ0 0030 Input Formatter 6 pixel info (bits 0-23)
Bit     Function
                0-11   Finder 0, cell 0 pixels
  YYZ0 0034 Input Formatter 6 pixel info (bits 0-23)
Bit     Function
                0-11   Finder 0, cell 1 pixels
  YYZ0 0038 Input Formatter 6 pixel info (bits 0-23)
Bit     Function
                0-11   Finder 0, cell 2 pixels
  YYZ0 003C Input Formatter 6 pixel info (bits 0-23)
Bit     Function
                0-11   Finder 0, cell 3 pixels
```

```
  YYZ0 0040 Input Formatter 6 Error summary info (bits 0-23)
Bit    Function
0      Error present on board
1      Error on cable 1 from Start bit
2      Error on cable 1 from word 0 bit
3      Error on cable 1 from Finder Error Bit
4      Error on cable 1 from Beam0 bit
5      Error on cable 1 from Start bit
6      Error on cable 1 from word 0 bit
7      Error on cable 1 from Finder Error Bit
8      Error on cable 1 from Beam0 bit
9      Error on cable 1 from Start bit
10     Error on cable 1 from word 0 bit
11     Error on cable 1 from Finder Error Bit
12     Error on cable 1 from Beam0 bit
13     Error on cable 1 from Start bit
14     Error on cable 1 from word 0 bit
15     Error on cable 1 from Finder Error Bit
16     Error on cable 1 from Beam0 bit
                  17-23  Total Error count

  YYZ0 0044 Input Formatter Extra word

  YYZ0 0048 Input Formatter Extra word

  YYZ0 004C Input Formatter Extra word



  YYZ0 0050 Input Formatter 5 pixel info (bits 0-23)
Bit    Function
                  0-11   Finder 0, cell 0 pixels
                  12-23  Finder 1, cell 0 pixels
  YYZ0 0054 Input Formatter 5 pixel info (bits 0-23)
Bit    Function
                  0-11   Finder 2, cell 0 pixels
                  12-23  Finder 3, cell 0 pixels
  YYZ0 0058 Input Formatter 5 pixel info (bits 0-23)
Bit    Function
                  0-11   Finder 0, cell 1 pixels
                  12-23  Finder 1, cell 1 pixels
```

```
  YYZ0 005C Input Formatter 5 pixel info (bits 0-23)
Bit    Function
                 0-11   Finder 2, cell 1 pixels
                 12-23  Finder 3, cell 1 pixels
  YYZ0 0060 Input Formatter 5 pixel info (bits 0-23)
Bit    Function
                 0-11   Finder 0, cell 2 pixels
                 12-23  Finder 1, cell 2 pixels
  YYZ0 0064 Input Formatter 5 pixel info (bits 0-23)
Bit    Function
                 0-11   Finder 2, cell 2 pixels
                 12-23  Finder 3, cell 2 pixels
  YYZ0 0068 Input Formatter 5 pixel info (bits 0-23)
Bit    Function
                 0-11   Finder 0, cell 3 pixels
                 12-23  Finder 1, cell 3 pixels
  YYZ0 006C Input Formatter 5 pixel info (bits 0-23)
Bit    Function
                 0-11   Finder 2, cell 3 pixels
                 12-23  Finder 3, cell 3 pixels


  YYZ0 0070 Input Formatter 4 pixel info (bits 0-23)
Bit    Function
                 0-11   Finder 0, cell 0 pixels
                 12-23  Finder 1, cell 0 pixels
  YYZ0 0074 Input Formatter 4 pixel info (bits 0-23)
Bit    Function
                 0-11   Finder 2, cell 0 pixels
                 12-23  Finder 3, cell 0 pixels
  YYZ0 0078 Input Formatter 4 pixel info (bits 0-23)
Bit    Function
                 0-11   Finder 0, cell 1 pixels
                 12-23  Finder 1, cell 1 pixels
  YYZ0 007C Input Formatter 4 pixel info (bits 0-23)
Bit    Function
                 0-11   Finder 2, cell 1 pixels
                 12-23  Finder 3, cell 1 pixels
  YYZ0 0080 Input Formatter 4 pixel info (bits 0-23)
Bit    Function
                 0-11   Finder 0, cell 2 pixels
                 12-23  Finder 1, cell 2 pixels
```

```
  YYZ0 0084 Input Formatter 4 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 2, cell 2 pixels
                12-23  Finder 3, cell 2 pixels
  YYZ0 0088 Input Formatter 4 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 0, cell 3 pixels
                12-23  Finder 1, cell 3 pixels
  YYZ0 008C Input Formatter 4 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 2, cell 3 pixels
                12-23  Finder 3, cell 3 pixels


  YYZ0 0090 Input Formatter 3 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 0, cell 0 pixels
                12-23  Finder 1, cell 0 pixels
  YYZ0 0094 Input Formatter 3 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 2, cell 0 pixels
                12-23  Finder 3, cell 0 pixels
  YYZ0 0098 Input Formatter 3 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 0, cell 1 pixels
                12-23  Finder 1, cell 1 pixels
  YYZ0 009C Input Formatter 3 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 2, cell 1 pixels
                12-23  Finder 3, cell 1 pixels
  YYZ0 00A0 Input Formatter 3 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 0, cell 2 pixels
                12-23  Finder 1, cell 2 pixels
  YYZ0 00A4 Input Formatter 3 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 2, cell 2 pixels
                12-23  Finder 3, cell 2 pixels
  YYZ0 00A8 Input Formatter 3 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 0, cell 3 pixels
                12-23  Finder 1, cell 3 pixels
```

```
  YYZ0 00AC Input Formatter 3 pixel info (bits 0-23)
Bit    Function
               0-11   Finder 2, cell 3 pixels
               12-23  Finder 3, cell 3 pixels


  YYZ0 00B0 Input Formatter 2 pixel info (bits 0-23)
Bit    Function
               0-11   Finder 0, cell 0 pixels
               12-23  Finder 1, cell 0 pixels
  YYZ0 00B4 Input Formatter 2 pixel info (bits 0-23)
Bit    Function
               0-11   Finder 2, cell 0 pixels
               12-23  Finder 3, cell 0 pixels
  YYZ0 00B8 Input Formatter 2 pixel info (bits 0-23)
Bit    Function
               0-11   Finder 0, cell 1 pixels
               12-23  Finder 1, cell 1 pixels
  YYZ0 00BC Input Formatter 2 pixel info (bits 0-23)
Bit    Function
               0-11   Finder 2, cell 1 pixels
               12-23  Finder 3, cell 1 pixels
  YYZ0 00C0 Input Formatter 2 pixel info (bits 0-23)
Bit    Function
               0-11   Finder 0, cell 2 pixels
               12-23  Finder 1, cell 2 pixels
  YYZ0 00C4 Input Formatter 2 pixel info (bits 0-23)
Bit    Function
               0-11   Finder 2, cell 2 pixels
               12-23  Finder 3, cell 2 pixels
  YYZ0 00C8 Input Formatter 2 pixel info (bits 0-23)
Bit    Function
               0-11   Finder 0, cell 3 pixels
               12-23  Finder 1, cell 3 pixels
  YYZ0 00CC Input Formatter 2 pixel info (bits 0-23)
Bit    Function
               0-11   Finder 2, cell 3 pixels
               12-23  Finder 3, cell 3 pixels


  YYZ0 00D0 Input Formatter 1 pixel info (bits 0-23)
Bit    Function
```

```
                0-11   Finder 0, cell 0 pixels
                12-23  Finder 1, cell 0 pixels
  YYZ0 00D4 Input Formatter 1 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 2, cell 0 pixels
                12-23  Finder 3, cell 0 pixels
  YYZ0 00D8 Input Formatter 1 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 0, cell 1 pixels
                12-23  Finder 1, cell 1 pixels
  YYZ0 00DC Input Formatter 1 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 2, cell 1 pixels
                12-23  Finder 3, cell 1 pixels
  YYZ0 00E0 Input Formatter 1 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 0, cell 2 pixels
                12-23  Finder 1, cell 2 pixels
  YYZ0 00E4 Input Formatter 1 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 2, cell 2 pixels
                12-23  Finder 3, cell 2 pixels
  YYZ0 00E8 Input Formatter 1 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 0, cell 3 pixels
                12-23  Finder 1, cell 3 pixels
  YYZ0 00EC Input Formatter 1 pixel info (bits 0-23)
Bit    Function
                0-11   Finder 2, cell 3 pixels
                12-23  Finder 3, cell 3 pixels
```

Pin Name	Pin Number	Note
clk33_in	91	On Board clocks
clk66_in	31	
clk132_in	211	
d0clk33	90	Channel Link 0 clock
d1clk33	212	Channel Link 1 clock
hard_clk_ena	40	On-board clock enable
soft_clk	15	Software clock toggle
inp_ctl0	138	From VME control
inp_ctl1	139	From VME control
inp_enable	17	Chip enable (from VME)
inp_reset	92	Chip reset (from VME)
input_oe	210	Enable for VME Data lines
pipeline_ena	43	Pipeline enable
cdf_l1a	133	L1 accept (from backplane)
l2wrtbuf0	55	L2 buffer0 (from backplane)
l2wrtbuf1	54	L2 buffer1 (from backplane)
l2bufadd0	208	L2 word address
l2bufadd1	209	
l2bufadd2	95	
l2rdbuf0	88	L2 read buffer 0
l2rdbuf1	238	L2 read buffer 1
input_addr0	45	Address lines to
input_addr1	41	internal memory
input_addr2	35	
input_addr3	34	
input_addr4	33	
input_addr5	28	
input_addr6	26	
input_addr7	25	
input_addr8	23	
chlnk_in0_0	61	Channel 0 link inputs
chlnk_in0_1	62	
chlnk_in0_2	63	
chlnk_in0_3	64	
chlnk_in0_4	65	
chlnk_in0_5	66	
chlnk_in0_6	67	
chlnk_in0_7	68	
chlnk_in0_8	70	
chlnk_in0_9	71	
chlnk_in0_10	72	
chlnk_in0_11	73	
chlnk_in0_12	74	
chlnk_in0_13	75	
chlnk_in0_14	76	
chlnk_in0_15	78	
chlnk_in0_16	79	
chlnk_in0_17	80	
chlnk_in0_18	81	
chlnk_in0_19	82	
chlnk_in0_20	83	
chlnk_in0_21	84	
chlnk_in0_22	86	
chlnk_in0_23	87	
chlnk_in1_0	99	Channel Link 1 Inputs
chlnk_in1_1	100	
chlnk_in1_2	101	
chlnk_in1_3	102	
chlnk_in1_4	103	
chlnk_in1_5	105	
chlnk_in1_6	106	
chlnk_in1_7	107	
chlnk_in1_8	108	
chlnk_in1_9	109	
chlnk_in1_10	110	
chlnk_in1_11	111	
chlnk_in1_12	113	
chlnk_in1_13	114	
chlnk_in1_14	115	
chlnk_in1_15	116	
chlnk_in1_16	117	
chlnk_in1_17	118	
chlnk_in1_18	119	
chlnk_in1_19	120	
chlnk_in1_20	126	
chlnk_in1_21	127	
chlnk_in1_22	128	
chlnk_in1_23	129	
wrd0_b0	49	Word 0 from Ch lnk 0
wrd0_b1	14	Word 0 from Ch lnk 1
lwrd0_out0	97	Registered Word 0 from Ch lnk 0
lwrd0_out1	98	Registered Word 0 from Ch lnk 1

Table 4: Input Formatter Chip Pin definition. Clock, control, memory address, and channel link input pins are shown.

Pin Name	Pin Number	Note
Lin_word0_0	237	Ch Lnk 1 Output to Linkers
Lin_word0_1	235	
Lin_word0_2	234	
Lin_word0_3	233	
Lin_word0_4	231	
Lin_word0_5	230	
Lin_word0_6	229	
Lin_word0_7	228	
Lin_word0_8	227	
Lin_word0_9	226	
Lin_word0_10	225	
Lin_word0_11	223	
Lin_word0_12	222	
Lin_word0_13	221	
Lin_word0_14	220	
Lin_word0_15	219	
Lin_word0_16	218	
Lin_word0_17	217	
Lin_word0_18	215	
Lin_word0_19	214	
Lin_word0_20	213	
Lin_word0_21	207	
Lin_word0_22	206	
Lin_word0_23	204	
Lin_word1_0	203	Ch Lnk 1 Output to Linkers
Lin_word1_1	202	
Lin_word1_2	201	
Lin_word1_3	200	
Lin_word1_4	199	
Lin_word1_5	198	
Lin_word1_6	196	
Lin_word1_7	195	
Lin_word1_8	194	
Lin_word1_9	193	
Lin_word1_10	192	
Lin_word1_11	191	
Lin_word1_12	190	
Lin_word1_13	188	
Lin_word1_14	187	
Lin_word1_15	186	
Lin_word1_16	185	
Lin_word1_17	184	
Lin_word1_18	183	
Lin_word1_19	182	
Lin_word1_20	181	
Lin_word1_21	175	
Lin_word1_22	174	
Lin_word1_23	173	
input_debug0	162	Output debug lines
input_debug1	18	
input_debug2	44	
input_debug3	9	
input_debug4	46	
input_debug5	137	
input_debug6	169	
input_debug7	20	
data_bus0	131	VME Bidir Data bus
data_bus1	30	
data_bus2	152	
data_bus3	29	
data_bus4	156	
data_bus5	38	
data_bus6	164	
data_bus7	39	
data_bus8	21	
data_bus9	36	
data_bus10	168	
data_bus11	19	
data_bus12	161	
data_bus13	163	
data_bus14	12	
data_bus15	6	
data_bus16	172	
data_bus17	171	
data_bus18	136	
data_bus19	142	
data_bus20	7	
data_bus21	24	
data_bus22	154	
data_bus23	134	

Table 5: Input Formatter Chip Pin definition. Output to Linker chips, debug output, and VME Databus pins are shown.

Pin Name	Pin Number	Note
clk33_in	91	
clk66_in	31	
clk132_in	211	
dclk33	90	
hard_clk_ena	40	
soft_clk	15	
inp_ctl0	138	
inp_ctl1	139	
inp_enable	17	
inp_reset	92	
input_oe	210	
pipeline_ena	43	
b0_delayed	212	
cdf_l1a	133	
l2wrtbuf0	55	
l2wrtbuf1	54	
l2bufadd0	208	
l2bufadd1	209	
l2bufadd2	95	
l2rdbuf0	88	
l2rdbuf1	238	
input_addr0	45	
input_addr1	41	
input_addr2	35	
input_addr3	34	
input_addr4	33	
input_addr5	28	
input_addr6	26	
input_addr7	25	
input_addr8	23	
chlnk_in0	61	Channel link inputs
chlnk_in1	62	
chlnk_in2	63	
chlnk_in3	64	
chlnk_in4	65	
chlnk_in5	66	
chlnk_in6	67	
chlnk_in7	68	
chlnk_in8	70	
chlnk_in9	71	
chlnk_in10	72	
chlnk_in11	73	
chlnk_in12	74	
chlnk_in13	75	
chlnk_in14	76	
chlnk_in15	78	
chlnk_in16	79	
chlnk_in17	80	
chlnk_in18	81	
chlnk_in19	82	
chlnk_in20	83	
chlnk_in21	84	
chlnk_in22	86	
chlnk_in23	87	
b0_mark0	99	B0 marker bits from 11 channel links
b0_mark1	100	
b0_mark2	101	
b0_mark3	102	
b0_mark4	103	
b0_mark5	105	
b0_mark6	106	
b0_mark7	107	
b0_mark8	108	
b0_mark9	109	
b0_mark10	110	
error_b0	111	
error_b1	113	
error_b2	114	
error_b3	115	
error_b4	116	
error_b5	117	
error_b6	118	
error_b7	119	
error_b8	120	
error_b9	126	
error_b10	127	

Table 6: Input Formatter Chip 6 Pin definition. Clock, control, memory address, and channel link input pins are shown.

Pin Name	Pin Number	Note
start_b0	173	
start_b1	174	
start_b2	175	
start_b3	181	
start_b4	182	
start_b5	183	
start_b6	184	
start_b7	185	
start_b8	186	
start_b9	187	
start_b10	188	
word0_inp0	202	
word0_inp1	201	
word0_inp2	200	
word0_inp3	199	
word0_inp4	198	
word0_inp5	196	
word0_inp6	195	
word0_inp7	194	
word0_inp8	193	
word0_inp9	192	
wrd0_b	49	
out_b0	240	
out_el	11	
Lin_word0	237	
Lin_word1	235	
Lin_word2	234	
Lin_word3	233	
Lin_word4	231	
Lin_word5	230	
Lin_word6	229	
Lin_word7	228	
Lin_word8	227	
Lin_word9	226	
Lin_word10	225	
Lin_word11	223	
Lin_word12	222	
Lin_word13	221	
Lin_word14	220	
Lin_word15	219	
Lin_word16	218	
Lin_word17	217	
Lin_word18	215	
Lin_word19	214	
Lin_word20	213	
Lin_word21	207	
Lin_word22	206	
Lin_word23	204	
data_bus0	131	
data_bus1	30	
data_bus2	152	
data_bus3	29	
data_bus4	156	
data_bus5	38	
data_bus6	164	
data_bus7	39	
data_bus8	21	
data_bus9	36	
data_bus10	168	
data_bus11	19	
data_bus12	161	
data_bus13	163	
data_bus14	12	
data_bus15	6	
data_bus16	172	
data_bus17	171	
data_bus18	136	
data_bus19	142	
data_bus20	7	
data_bus21	24	
data_bus22	154	
data_bus23	134	

Table 7: Input Formatter Chip 6 Pin definition. Output to Linker chips, and VME Databus pins are shown.

Pin Name	Pin Number	Note
b_vme_as	181	VME control lines
b_vme_ds0	223	
b_vme_ds1	220	
b_vme_iack	82	
b_vme_lword	221	
b_vme_sysreset	92	
b_vme_write	212	
vmedataoe	200	
vmerw	236	
weack	67	
cdfga0	118	Geographical address lines
cdfga1	119	
cdfga2	183	
cdfga3	182	
cdfga4	185	
vmeam0	21	Address modifier lines
vmeam1	83	
vmeam2	213	
vmeam3	88	
vmeam4	217	
vmeam5	86	
vmea0	39	VME address lines
vmea1	87	
vmea2	134	
vmea3	38	
vmea4	142	
vmea5	149	
vmea6	147	
vmea7	219	
vmea8	84	
vmea9	214	
vmea10	215	
vmea11	35	
vmea12	151	
vmea13	148	
vmea14	146	
vmea15	144	
vmea16	143	
vmea17	31	
vmea18	33	
vmea19	34	
vmea20	40	
vmea21	46	
vmea22	41	
vmea23	36	
vmea24	45	
vmea25	44	
vmea26	43	
vmea27	161	
vmea28	162	
vmea29	120	
vmea30	163	
vmea31	15	
clk33	211	On-board clocks
clk66	81	
clk132	91	
cdf_halt	53	CDF Backplane signals
cdf_recover	50	
cdf_run	49	
cdf_b0mark	54	
cdf_error	218	
read_clk	71	
write_clk	7	
input_addr0	199	Address lines to Flashram
input_addr1	175	
input_addr2	188	
input_addr3	240	
input_addr4	184	
input_addr5	64	
input_addr6	222	
input_addr7	202	
input_addr8	204	
input_addr9	238	
input_addr10	98	
input_addr11	235	
input_addr12	234	
input_addr13	237	
input_addr14	76	
input_addr15	78	
input_addr16	79	
input_addr17	233	
input_addr18	115	
out_el	62	Input error line from Input chip
delayed_b0	108	Delayed B0 output line
hard_clk_ena	159	Hardware clock enable
soft_clk	65	Software clock toggle

Table 8: VME/Control chip pins.

Pin Name	Pin Number	Note
pipeline_ena	56	Pipeline enable output
inp_ctl0	102	Control lines to input formatters
inp_ctl1	198	
inp_enable	105	
inp_reset	129	
fmt_ctl0	196	Control lines to output formatters
fmt_ctl1	111	
fmt_enable	195	
fmt_reset	133	
lnk_ctl0	186	Control lines to linkers
lnk_ctl1	116	
lnk_enable	132	
lnk_reset	107	
format_oe1	24	VME data output enables
format_oe2	226	
input_oe1	227	VME data output enables
input_oe2	68	
input_oe3	169	
input_oe4	171	
input_oe5	6	
input_oe6	74	
cdf_l1a	80	L1 accept (from backplane)
l2wrtbuf0	48	L2 buffer0 (from backplane)
l2wrtbuf1	136	L2 buffer1 (from backplane)
l2bufadd0	103	L2 word address (output)
l2bufadd1	117	
l2bufadd2	203	
l2rdbuf0	97	L2 read buffer 0 (output)
l2rdbuf1	225	L2 read buffer 1 (output)
jtag_int0	110	Jtag stuff
jtag_int1	113	
jtag_int2	187	
jtag_int3	114	
emergency_pgm	55	Control lines for Altera Loading
nstatus	90	
nconfig	128	
npgm_ena	126	
pgm_clk	127	
pgm_data	12	
conf_done	210	
load_ram_num0	106	Enables for Flashrams
load_ram_num1	191	
load_ram_num2	194	
load_ram_num3	109	
load_ram_num4	193	
load_ram_num5	192	
vme_debug0	72	Debug lines for VME
vme_debug1	201	
vme_debug2	75	
vme_debug3	30	
vme_debug4	131	
vme_debug5	100	
vme_debug6	190	
vme_debug7	101	
data0	8	VME data lines
data1	138	
data2	139	
data3	141	
data4	9	
data5	14	
data6	174	
data7	164	
data8	51	
data9	157	
data10	172	
data11	154	
data12	173	
data13	158	
data14	137	
data15	156	
data16	152	
data17	11	
data18	153	
data19	166	
data20	168	
data21	17	
data22	19	
data23	167	
data24	20	
data25	26	
data26	25	
data27	23	
data28	18	
data29	29	
data30	28	
data31	13	

Table 9: VME/Control chip pins.

Pin Name	Pin Number	Note
finder1_0	61	Finder 1 Pixel Inputs
finder1_1	62	
finder1_2	63	
finder1_3	64	
finder1_4	65	
finder1_5	66	
finder1_6	67	
finder1_7	68	
finder1_8	70	
finder1_9	71	
finder1_10	72	
finder1_11	73	
finder2_0	240	Finder 2 Pixel Inputs
finder2_1	239	
finder2_2	238	
finder2_3	237	
finder2_4	236	
finder2_5	235	
finder2_6	234	
finder2_7	233	
finder2_8	231	
finder2_9	230	
finder2_10	229	
finder2_11	228	
finder3_0	97	Finder 3 Pixel Inputs
finder3_1	98	
finder3_2	99	
finder3_3	100	
finder3_4	101	
finder3_5	102	
finder3_6	103	
finder3_7	105	
finder3_8	106	
finder3_9	107	
finder3_10	108	
finder3_11	109	
finder4_0	204	Finder 4 Pixel Inputs
finder4_1	203	
finder4_2	202	
finder4_3	201	
finder4_4	200	
finder4_5	199	
finder4_6	198	
finder4_7	196	
finder4_8	195	
finder4_9	194	
finder4_10	193	
finder4_11	192	
finder5_0	75	Finder 5 Pixel Inputs
finder5_1	76	
finder5_2	78	
finder5_3	79	
finder5_4	80	
finder5_5	81	
finder5_6	82	
finder5_7	83	
finder5_8	84	
finder5_9	86	
finder5_10	87	
finder5_11	88	
finder6_0	126	Finder 6 Pixel Inputs
finder6_1	127	
finder6_2	128	
finder6_3	129	
finder6_4	131	
finder6_5	132	
finder6_6	133	
finder6_7	134	
finder6_8	136	
finder6_9	137	
finder6_10	138	
finder6_11	139	
finder7_0	226	Finder 7 Pixel Inputs
finder7_1	225	
finder7_2	223	
finder7_3	222	
finder7_4	221	
finder7_5	220	
finder7_6	219	
finder7_7	218	
finder7_8	217	
finder7_9	215	
finder7_10	214	
finder7_11	213	

Table 10: Linker chip pin definitions. Input from Input Fomratter chips are shown.

Pin Name	Pin Number	Note
finder8_0	11	Finder 8 pixel input
finder8_1	12	
finder8_2	13	
finder8_3	14	
finder8_4	15	
finder8_5	17	
finder8_6	18	
finder8_7	19	
finder8_8	20	
finder8_9	21	
finder8_10	23	
finder8_11	24	
finder9_0	43	Finder 9 pixel input
finder9_1	44	
finder9_2	45	
finder9_3	46	
finder9_4	48	
finder9_5	49	
finder9_6	50	
finder9_7	51	
finder9_8	53	
finder9_9	54	
finder9_10	55	
finder9_11	56	
finder10_0	162	Finder 10 pixel input
finder10_1	163	
finder10_2	164	
finder10_3	166	
finder10_4	167	
finder10_5	168	
finder10_6	169	
finder10_7	171	
finder10_8	172	
finder10_9	173	
finder10_10	174	
finder10_11	175	
iclk33	91	On Board clocks
iclk66	211	
iclk132	92	
hard_clk_ena	210	On-board clock enable
soft_clk	212	Software clock toggle
lnk_ctl0	94	From VME control
lnk_ctl1	95	From VME control
lnk_enable	90	Chip enable (from VME)
lnk_reset	25	Chip reset (from VME)
linker_debug0	146	Linker Debug Lines
linker_debug1	147	
linker_debug2	148	
linker_debug3	149	
linker_debug4	151	
linker_debug5	152	
linker_debug6	153	
linker_debug7	154	
lnkout0	26	Linker Output Lines
lnkout1	28	
lnkout2	29	
lnkout3	30	
lnkout4	31	
lnkout5	33	
lnkout6	34	
lnkout7	35	
lnkout8	36	
lnkout9	38	
lnkout10	39	
lnkout11	40	
lnkout12	41	

Table 11: Linker pins. Output to formatter chips, control, and debug pins are shown.

Pin Name	Pin Number	Note
cdf_l1a	11	trigger signals from backplane
l2wrtbuf0	229	
l2wrtbuf1	9	
l2bufadd0	172	trigger signals from VME chip
l2bufadd1	181	
l2rdbuf0	210	
l2rdbuf1	209	
delayed_b0	196	
clk33	91	on-board clocks
clk66	211	
clk132	7	
out_b0	207	signals from Input chip 6
out_el	139	
dsy_out_ef	200	
dsy_out_st	138	
fmt_ctl0	141	control lines from VME
fmt_ctl1	142	
fmt_enable	213	
fmt_reset	92	
format_oe	90	
pipeline_ena	8	
hard_clk_ena	6	clock control
softclk	169	
input_addr0	41	addresses for internal RAM
input_addr1	94	
input_addr2	95	
input_addr3	97	
input_addr4	98	
input_addr5	99	
input_addr6	100	
input_addr7	101	
input_addr8	71	
input_we	72	
read_clock	66	
write_clock	64	
link0_out0	73	Inputs from linker chip 0
link0_out1	111	
link0_out2	191	
link0_out3	23	
link0_out4	190	
link0_out5	113	
link0_out6	188	
link0_out7	110	
link0_out8	74	
link0_out9	227	
link0_out10	80	
link0_out11	75	
link0_out12	225	
link1_out0	78	Inputs from linker chip 1
link1_out1	171	
link1_out2	183	
link1_out3	118	
link1_out4	120	
link1_out5	56	
link1_out6	55	
link1_out7	54	
link1_out8	156	
link1_out9	157	
link1_out10	158	
link1_out11	53	
link1_out12	12	
link2_out0	51	Inputs from linker chip 2
link2_out1	50	
link2_out2	159	
link2_out3	30	
link2_out4	29	
link2_out5	28	
link2_out6	26	
link2_out7	49	
link2_out8	25	
link2_out9	24	
link2_out10	214	
link2_out11	88	
link2_out12	87	
link3_out0	221	Inputs from linker chip 3
link3_out1	175	
link3_out2	119	
link3_out3	182	
link3_out4	126	
link3_out5	127	
link3_out6	128	
link3_out7	129	
link3_out8	83	
link3_out9	215	
link3_out10	152	
link3_out11	131	
link3_out12	13	

Table 12: Output chip pins.

Pin Name	Pin Number	Note
link4_out0	132	Inputs from linker chip 4
link4_out1	133	
link4_out2	114	
link4_out3	153	
link4_out4	154	
link4_out5	192	
link4_out6	109	
link4_out7	187	
link4_out8	86	
link4_out9	217	
link4_out10	84	
link4_out11	219	
link4_out12	218	
link5_out0	82	Inputs from linker chip 5
link5_out1	174	
link5_out2	117	
link5_out3	185	
link5_out4	184	
link5_out5	116	
link5_out6	186	
link5_out7	115	
link5_out8	220	
link5_out9	81	
link5_out10	222	
link5_out11	223	
link5_out12	173	
format_debug0	228	Debug out lines
format_debug1	70	
format_debug2	162	
format_debug3	108	
format_debug4	206	
format_debug5	202	
format_debug6	107	
format_debug7	168	
lword_out0	151	Output to XTRP
lword_out1	149	
lword_out2	44	
lword_out3	164	
lword_out4	195	
lword_out5	198	
lword_out6	103	
lword_out7	102	
lword_out8	148	
lword_out9	40	
lword_out10	17	
lword_out11	166	
lword_out12	31	
lword_out13	240	
lword_out14	62	
lword_out15	239	
lword_out16	143	
lword_out17	106	
lword_out18	193	
lword_out19	208	
lword_out20	161	
lword_out21	163	
lword_out22	15	
lword_out23	79	
lword_out24	63	
lword_out25	65	
lword_out26	61	
lword_out27	236	
lword_out28	38	
lword_out29	39	
data_bus0	18	VME databus Lines
data_bus1	167	
data_bus2	20	
data_bus3	194	
data_bus4	134	
data_bus5	136	
data_bus6	43	
data_bus7	46	
data_bus8	45	
data_bus9	137	
data_bus10	48	
data_bus11	36	
data_bus12	76	
data_bus13	19	
data_bus14	33	
data_bus15	34	
data_bus16	35	
data_bus17	147	
data_bus18	14	
data_bus19	146	
data_bus20	199	
data_bus21	144	
data_bus22	238	
data_bus23	21	

Table 13: Output chip pins.

References

[1] *Finder Chip to Channel Link Data Base*, Finder to Linker Chip Map For All Linker BoardsR. Hughes. P. Koehn, B. Winer.

[2] *Finder to Linker Cabling*, R. Hughes. P. Koehn, B. Winer.

[3] *Finder to Linker Board Map*, R. Hughes. P. Koehn, B. Winer.

[4] *Finder to Linker Chip Map For All Linker Boards*, R. Hughes. P. Koehn, B. Winer.

[5] *Finder Chip Design Note*, Scott Holm.

[6] *Specification for XFT Finder Modules*, T. Shaw et al.

[7] *DAQ Manager's Notebook*, The CDF DAQ Group.

[8] *Trigger Supervisor Protocols for Run II* CDF Group, Yale University.

[9] *Specification of DAQ/Trigger Synchronization Checking Protocol and VME board Header Word, CDF Note 3145*, K. Ohl, J. Wahl, P. Wilson.

[10] *A Standard Front-end and Trigger VME Readout Crate for the CDF Upgrade, CDF Note 2388* T. Shaw and G. Sullivan.