CDF/DOC/TRIGGER/PUBLIC/XXXX
January 11, 1999
DRAFT of Version 1.0

Hardware Design and Specifications of the SVT Hit Finder

Bill Ashmanskas, Jeff Berryhill,
Mircea Bogdan, Ray Culbertson
University of Chicago

1 Introduction

The SVT [1] Hit Finder board pre-processes a copy of the SVX-II raw data, providing a list of cluster positions to be used in finding and fitting SVT tracks. Several earlier notes [2, 3, 4, 5, 6] have described the Hit Finder design. Here, we document some details of the board-level design, in preparation for producing and testing a prototype board.

2 Data Flow

2.1 Big Picture

SVX-II [7] is a silicon strip detector providing charged-particle position measurements at five radii between 2.5 cm and 10.6 cm from the Tevatron beam line. Each of the five layers of silicon detectors is double-sided: one side (axial) measures $r \cdot \phi(r)$, while the other side (stereo) measures $z(r)$ in layers 0,1,3 and $r \cdot \phi(r) \pm z(r)\tan(1.2°)$ in layers 2,4. (The baseline Hit Finder design ignores the stereo data.) SVX-II is divided into six readout barrels in z and into twelve wedges in ϕ (Figure 1).

The detectors measure pulse height (ionization deposited) vs. strip number, with an axial strip pitch of $(60, 62, 60, 60, 65)$ μm respectively for layer $(0, 1, 2, 3, 4)$. A charged particle typically deposits ionization in two or three adjacent strips in each layer. Signals from 128 adjacent axial strips are digitized by one SVX3 [8] chip. The SVX readout is sparse, but if a given strip is above readout threshold, the neighboring strip on each side will also be read out.

(file=svxgeom.ps)

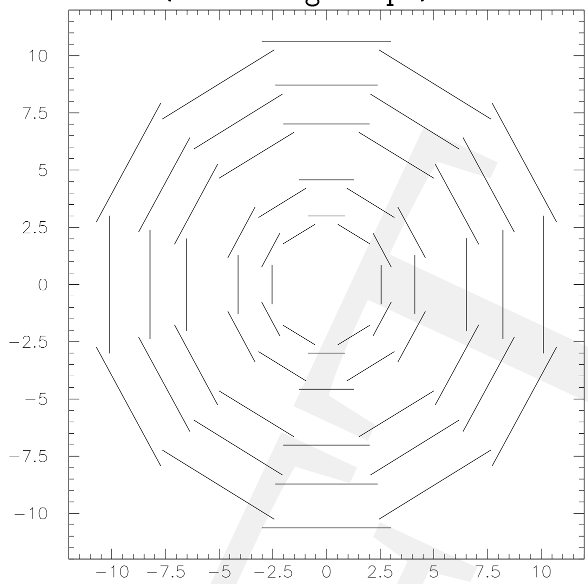

Figure 1: SVX-II geometry in x-y plane (dimensions in cm).

One SVX readout stream comprises the data for one layer of one wedge of one barrel. There are $6 \times 12 \times 5 = 360$ total streams. The number of axial chips per stream is $(2, 3, 5, 6, 7)$ respectively for layer $(0, 1, 2, 3, 4)$. The five streams from one wedge of one barrel are transmitted together on a pair of ~ 1 Gb/sec (G-link) fiber-optic serial links. The G-link signals are optically split in the counting room, so that the VME Readout Buffer [9] boards and the Hit Finder boards see identical copies of the SVX data.

Each Hit Finder processes data from two pairs of G-links (ten streams). (Three Hit Finders cover one complete ϕ sector of SVX-II.) An auxilliary card (VTM [10]) converts four G-links' serial data into four sets of 20-bit words, clocked out at 53 MHz. Each G-link's 20 data bits, accompanied by a clock signal and "data valid," "frame error," and "lost lock" control signals, are passed from the VTM to the Hit Finder via the VME J3 backplane.

The first stage of the Hit Finder (alignment) needs to separate four sets of 20-bit words corresponding to four G-links, each with its own clock, into ten sets of 8-bit words corresponding to ten SVX readout streams. The main complication is that each fiber carries 2.5 streams, so the data from each pair of fibers needs to be aligned into five complete streams. A second task performed by the alignment stage is to parallelize each stream from a 53 MHz sequence of 8-bit words into a 26.5 MHz sequence of 16-bit words. This conversion simplifies board design significantly, by halving the signal frequency, and simplifies data processing, since an SVX data stream

(Table 5) logically consists of byte pairs (e.g. strip number + pulse height).

The next stage of the Hit Finder (clustering) parses each data stream, recognizing event header words, chip ID words, and ordinary data words. The first main task of the clustering engine is to simplify the data format, by eliminating chip ID words and (in exchange) widening the strip number from seven bits to ten bits. The second task is to replace the sequence of (strip number + pulse height) entries with a sequence of cluster positions, obtained by computing the pulse-height-weighted average of consecutive strip positions. (Recall that a charged particle typically leaves a signal in several adjacent strips.) This average is quantized in sixteenths of the strip pitch (about 4 μm), making a cluster position fourteen bits wide. The clustering engine for each stream writes 14-bit clusters (plus a bit to flag a “long” cluster, a bit to flag an end-of-event word, and two spare bits) to a FIFO. An opportunity to write a word to the FIFO occurs once per 26.5 MHz clock cycle, but because of the clustering, actual write operations are sparse. At the end of each event, each clustering engine writes a word containing error flags and the bunch-crossing number.

Next, the merging stage, clocked by an on-board 30 MHz oscillator, combines all ten streams' data into a single data path, prepending a stream number to each cluster position. A final output stage maps stream numbers into barrel and layer bits and monitors error conditions (such as the merging of data from two different Tevatron bunch crossings). Errors are reported by setting bits in an end-of-event word and optionally by asserting error signals on the VME J2 backplane. The output processor sends 23-bit words (Table 10), clocked out with a “data strobe” signal, to a front-panel connector via a set of LVDS drivers. The downstream board (a Merger [11] board) returns a “hold” signal to indicate a nearly full input FIFO. A second front-panel connector, reserved for expansion, shares the 23 data lines with the first connector but has its own “data strobe” and “hold” lines.

Figure 2 sketches the Hit Finder data flow. Figure 4 shows the top-level Hit Finder schematic. Figure 5 shows the physical board layout.

2.2 P3

The Hit Finder receives SVX-II raw data from a VTM auxilliary card at 53 MHz, via the VME J3 backplane. The P3 connectors used on the Hit Finder are 110-pin ERNI 64785 (P5) and 125-pin ERNI 64179 (P6). Tables 1 and 2 show the P3 pin assignments connecting the VTM to the Hit Finder. The bus from P3 to Data Alignment on the schematic, GRT_Data(99:0), is described in Table 3. Because of the high speed, all traces from P3 to the DAD chips are less than four inches long.

(file=arch.ps)

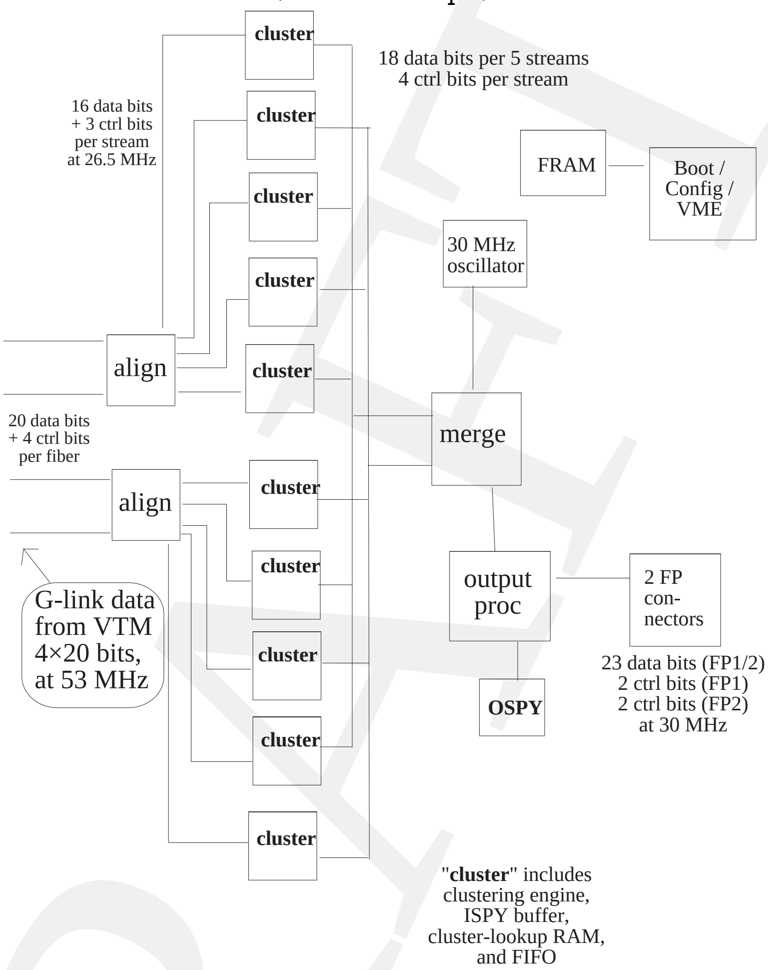

Figure 2: Data-flow cartoon.

	A	B	D	E
01	L0_D00	ground	ground	L2_D00
02	L0_D01	L0_D02	L2_D02	L2_D01
03	ground	L0_D03	L2_D03	ground
04	L0_D05	L0_D04	L2_D04	L2_D05
05	L0_D06	ground	ground	L2_D06
06	L0_D07	L0_D08	L2_D08	L2_D07
07	ground	L0_D09	L2_D09	ground
08	L0_D11	L0_D10	L2_D10	L2_D11
09	L0_D12	ground	ground	L2_D12
10	L0_D13	L0_D14	L2_D14	L2_D13
11	ground	L0_D15	L2_D15	ground
12	L0_D16	(L0_D16)	L2_D16	(L2_D16)
13	L0_D17	ground	ground	L2_D17
14	(L0_D17)	L0_D18	L2_D18	(L2_D17)
15	ground	(L0_D18)	(L2_D18)	ground
16	(L0_D19)	L0_D19	L2_D19	(L2_D19)
17	(L0_CAV*)	ground	ground	(L2_CAV*)
18	L0_DAV*	L0_LNKRDY*	L2_LNKRDY*	L2_DAV*
19	ground	(L0_LNKRDY*)	(L2_LNKRDY*)	ground
20	L0_STRBOUT	ground	ground	L2_STRBOUT
21	L0_SIGDETECT	L0_ERROR	(L2_ERROR)	L2_SIGDETECT
22	ground	(L0_ERROR)	L2_ERROR	ground
23	ground	ground	ground	ground
24	ground	ground	ground	ground

Table 1: P3 pin assignments (continued in Table 2). L0_D00..07 are B0L0 bits 0..7. L0_D08..15 are B0L1 bits 0..7. L0_D16..19 are B0L4 bits 0..3. L2_D00..07 are B1L0 bits 0..7. L2_D08..15 are B1L1 bits 0..7. L2_D16..19 are B1L4 bits 0..3. Pins whose names are in parentheses are not connected to the Hit Finder. STRBOUT is 53 MHz clock. DAV* is active-low "data valid" signal, used by Hit Finder to find event boundaries. ERROR indicates single-frame transmission error, flagged as an error by clustering engine. LNKRDY* is sent to MOP chip as active-high "lost lock" signal. SIGDETECT is connected to DAD chip but is unused.

	A	B	D	E
25	ground	ground	ground	ground
26	(L1_CAV*)	ground	ground	(L3_CAV*)
27	L1_DAV*	(L1_LNKRDY*)	(L3_LNKRDY*)	L3_DAV*
28	ground	L1_LNKRDY*	L3_LNKRDY*	ground
29	L1_STRBOUT	ground	ground	L3_STRBOUT
30	L1_SIGDETECT	L1_ERROR	L3_ERROR	L3_SIGDETECT
31	ground	(L1_ERROR)	(L3_ERROR)	ground
32	L1_D00	ground	ground	L3_D00
33	L1_D01	L1_D02	L3_D02	L3_D01
34	ground	L1_D03	L3_D03	ground
35	L1_D05	L1_D04	L3_D04	L3_D05
36	L1_D06	ground	ground	L3_D06
37	L1_D07	L1_D08	L3_D08	L3_D07
38	ground	L1_D09	L3_D09	ground
39	L1_D11	L1_D10	L3_D10	L3_D11
40	L1_D12	ground	ground	L3_D12
41	L1_D13	L1_D14	L3_D14	L3_D13
42	ground	L1_D15	L3_D15	ground
43	L1_D16	(L1_D16)	L3_D16	(L3_D16)
44	L1_D17	ground	ground	(L3_D17)
45	(L1_D17)	L1_D18	L3_D18	L3_D17
46	ground	(L1_D18)	(L3_D18)	ground
47	L1_D19	(L1_D19)	L3_D19	(L3_D19)

Table 2: P3 pin assignments (continued from Table 1). L1_D00..07 are B0L2 bits 0..7. L1_D08..15 are B0L3 bits 0..7. L1_D16..19 are B0L4 bits 4..7. L3_D00..07 are B1L2 bits 0..7. L3_D08..15 are B1L3 bits 0..7. L3_D16..19 are B1L4 bits 4..7. Pins C1–C35, C41–C44, and C47 are not connected. (Pin C47 is MODID, used by VRB to determine what type of auxilliary card is connected to it.) Pins C36–C40 are grounded. Pin C45 is bidirectional VTM_Serial I/O signal, connected to the Hit Finder Boot chip but currently unused. Pin C46 is VTM_Reset* signal, connected ot the Boot chip but currently never asserted.

	L0_···		**L1_···**		**L2_···**		**L3_···**
0	D00	25	D00	50	D00	75	D00
1	D01	26	D01	51	D01	76	D01
2	D02	27	D02	52	D02	77	D02
3	D03	28	D03	53	D03	78	D03
4	D04	29	D04	54	D04	79	D04
5	D05	30	D05	55	D05	80	D05
6	D06	31	D06	56	D06	81	D06
7	D07	32	D07	57	D07	82	D07
8	D08	33	D08	58	D08	83	D08
9	D09	34	D09	59	D09	84	D09
10	D10	35	D10	60	D10	85	D10
11	D11	36	D11	61	D11	86	D11
12	D12	37	D12	62	D12	87	D12
13	D13	38	D13	63	D13	88	D13
14	D14	39	D14	64	D14	89	D14
15	D15	40	D15	65	D15	90	D15
16	D16	41	D16	66	D16	91	D16
17	D17	42	D17	67	D17	92	D17
18	D18	43	D18	68	D18	93	D18
19	D19	44	D19	69	D19	94	D19
20	LNKRDY*	45	LNKRDY*	70	LNKRDY*	95	LNKRDY*
21	ERROR	46	ERROR	71	ERROR	96	ERROR
22	DAV*	47	DAV*	72	DAV*	97	DAV*
23	STRBOUT	48	STRBOUT	73	STRBOUT	98	STRBOUT
24	SIGDTCT	49	SIGDTCT	74	SIGDTCT	99	SIGDTCT

Table 3: GRT_Data bus pin assignments.

2.3 Alignment Device

The Hit Finder contains two Data Alignment Device (DAD) chips, one for each pair of G-links. Each DAD is an Altera Flex 10K30AQC240-1 chip, with 52% of its logic cells in use. It aligns the data from two G-links and parallelizes five streams of 53 MHz 8-bit words into five streams of 26.5 MHz 16-bit words. The data to be aligned for each G-link begin with the first word for which DAV* is asserted.[1] The DAD allows the two G-links to be out of phase by up to 2.5 clock cycles. (We expect the two G-links to be out of phase by less than one clock cycle. Is there an engineering reference for this?)

Between events, neither DAV* is asserted, and the DAD waits for valid data. As valid data words arrive for each G-link, they are written into a shallow FIFO implemented within the Altera chip. Once the later G-link's FIFO contains a few words, aligned data can be clocked out. Each G-link's DAV* is deasserted when that link's transmission for the current event is complete. When both DAV* signals are no longer asserted, the DAD returns to the wait state.

A Reset signal, needed to prepare the DAD for data taking, is driven by HF_Init (on the HF_Ctrl bus), which is generated by the Boot chip in response to SVT_Init*.

DAV* and ERROR signals are supplied per G-link at 53 MHz. Corresponding signals must be sent out per clustering engine at 26.5 MHz. For the un-split streams from a G-link, a 26.5 MHz word's ERROR bit is the OR of the corresponding two 53 MHz words' error bits; for the split stream, a four-way OR is needed. Similarly, "valid" outputs are computed with a two-way or four-way AND. Each DAD divides down its 53 MHz input clock, using a separate output pin for the 26.5 MHz clock sent to each clustering engine. The positive clock edge is centered between data transitions. Table 4 describes the HM_Data(189:0) bus, on which data travel from Data Alignment to Hit Processing.

LNKRDY* signals, sent from P3 on the GRT_Data bus, enter the Data Alignment box on the schematic and are passed on to the Hit_Merging box as Lost_Lock(3:0). Physically, these four lines are connected directly from P3 to the MOP chip. Since they are effectively slow-control signals, their length is not important.

The DAD has been successfully simulated at input speeds up to 61 MHz (15% above nominal). Our simulations have assumed that the positive STRBOUT clock edge is centered between data transitions.

The DAD contains no VME-addressable registers.

[1] Oops! Bug! Up to now we have assumed DAV* to be active-high. We need to invert DAV* in the DAD.

	stream									
signal	0	6	2	8	4	5	1	7	3	9
d00	0	19	38	57	76	95	114	133	152	171
d01	1	20	39	58	77	96	115	134	153	172
d02	2	21	40	59	78	97	116	135	154	173
d03	3	22	41	60	79	98	117	136	155	174
d04	4	23	42	61	80	99	118	137	156	175
d05	5	24	43	62	81	100	119	138	157	176
d06	6	25	44	63	82	101	120	139	158	177
d07	7	26	45	64	83	102	121	140	159	178
d08	8	27	46	65	84	103	122	141	160	179
d09	9	28	47	66	85	104	123	142	161	180
d10	10	29	48	67	86	105	124	143	162	181
d11	11	30	49	68	87	106	125	144	163	182
d12	12	31	50	69	88	107	126	145	164	183
d13	13	32	51	70	89	108	127	146	165	184
d14	14	33	52	71	90	109	128	147	166	185
d15	15	34	53	72	91	110	129	148	167	186
clk	16	35	54	73	92	111	130	149	168	187
dav	17	36	55	74	93	112	131	150	169	188
err	18	37	56	75	94	113	132	151	170	189

Table 4: HM_Data bus pin assignments. The strange ordering of stream numbers is because streams (1,3) have been exchanged with streams (6,8) in their connections from Hit Clustering to Hit Merging. This exchange is expected to reduce occupancy correlations among streams sharing a common data path to the Merger. Streams are numbered from the Merger's perspective, not P3's perspective.

2.4 Clustering Engine

A clustering engine (also called a Hit Squad on the schematic) consists of a HitMan (Altera Flex 10K50VBC356-2 chip, 39% logic used), an Input SPY buffer (Motorola MCM69F618CTQ9 64K×18 synchronous SRAM), a Cluster RAM (Samsung KM681002AJ-12 128K×8 asynchronous SRAM), and a FIFO (Cypress CY7C4245-10ASC 4K×18 synchronous). Figure 6 shows one clustering engine, and Figure 7 shows the complete Hit Processing sheet of the schematic.

Each clustering engine transforms one stream's raw SVX data (Table 5) into a sequence of cluster positions (Table 6), which are written to the FIFO. The Input SPY buffer (ISPY) stores a copy of the last 64K input words seen by the clustering engine, for debugging and validation. The ISPY can also be used as a data source, for initially testing the Hit Finder without a VTM.(2) In this test mode, bits 16 and 17 are the ERROR and "valid" inputs, respectively.

Internally, the HitMan chip is divided into three stages, called Ready, Aim, and Fire. Ready is a state machine that parses the SVX raw data. Initially, the Ready machine waits in its Ready state until it sees an input word whose Valid bit is set. This first valid word is discarded, and the machine goes to the Header state, which latches the bunch-crossing number into bits 0–7 of the end-of-event-word register, initializes a few registers, and moves to the RunData state. The RunData state keeps track of the number of axial chips seen (initialized to the illegal value 15(3)), the last 10-bit chip+strip number seen (initialized to 0), and the error bits in the end-of-event-word register. If the input word is an axial chip ID number (high byte is 101xxxxx), its value is compared with the expected value, and the chip count is incremented (or set to 15 if the expected chip ID is not seen). If the input word is a stereo chip ID number (high byte is 100xxxxx), the machine goes to the EndData state. If the input word is an ordinary data word (high byte is 0xxxxxxx), an output word is computed from the 3-bit chip number, the 7-bit strip number, and the 8-bit pulse height. If the current chip+strip number is less than its predecessor, if the chip count is 15, or if the ERROR bit is set on the input word, then the output word is marked invalid, error bits are set in the end-of-event-word register, and the machine goes to the EndData state. The output Valid bit is only set when the machine is in the RunData state and is writing an ordinary data word in which no error is detected. The EndData state clocks out 21 words marked invalid (to flush the downstream pipeline), asserts an end-of-event flag for one clock cycle, and goes to the Wait state. The Wait state waits until it sees an input word whose Valid bit is clear, then goes to the Ready state. There is also a Disabled state which is entered and exited only via VME write.

(2) This test mode is currently broken, but will be made functional before we make the prototype.

(3) The illegal value used to be 7, but we made it 15 to make the Hit Finder compatible with the ISL, which has eight axial chips per stream.

Ready sends Aim a sequence of words that consist of a valid bit, a 3-bit chip number, a 7-bit strip number, and an 8-bit pulse height. Ready also outputs a 15-bit end-of-event word and an end-of-event flag used to multiplex data words and event-of-event words.

Aim is a five-stage pipeline used to subtract a pedestal value from each pulse height and to reject words whose pulse heights are below a threshold. Aim first replaces all negative pulse heights with zero.[4] It then looks up a 7-bit chip+strip-dependent pedestal and a 7-bit chip-dependent threshold in the Altera chip's internal SRAM. Aim subtracts the pedestal, zeroing negative results, and marks the word invalid if its pedestal-subtracted pulse height is below threshold.

Fire is a fourteen-stage pipeline used to recognize and cluster contiguous sequences of strip numbers and compute the charge-weighted average of strip positions in each cluster. Unlike Ready and Aim, which run continuously, the Ready pipeline uses a clock-enable that allows data to advance only if Aim outputs a valid word or Ready is in the EndData state. This selective clocking prevents chip ID words from splitting clusters that span chip boundaries, and allows HitMan's power consumption to benefit from the pulse-height cut. Two different clustering algorithms ("peaks" and "mixed") have been designed for the Hit Finder; currently only "mixed" is implemented. We describe what is implemented.

Fire latches in its input, increments a copy of the current chip+strip number, and evaluates whether whether the next strip is adjacent to the current strip. It then accumulates a running count of adjacent strips and a running sum of the pulse heights on adjacent strips. If the summed charge is above a programmable threshold, a good-cluster-charge bit is set. If the cluster length exceeds three strips, a long-cluster bit is set. An end-of-cluster bit flags the last strip in a cluster—if the next strip is non-adjacent or if the cluster length is a multiple of six strips. (Extremely long clusters are divided into six-strip clusters.) Bits 6–2 of the pulse height for three consecutive pipeline stages form a 15-bit CRAM address; one or two of the three 5-bit subaddresses may be forced to zero, depending on which adjacent pipeline stages actually contain adjacent chip+strip numbers. The CRAM is used as a lookup table for computing the charge-weighted average of one (trivial), two, or three adjacent strip positions; bits 0–5 of the CRAM data bus are used as a signed offset to the central strip position, quantized in sixteenths of the strip pitch. For a long cluster (4–6 strips), no charge-weighting is done; instead, the cluster length is right-shifted one bit and subtracted from the final strip position to compute the median. To write out clusters, one simply sets the Valid output bit only for the last strip in each cluster. For three-strip clusters, a special case is needed to overwrite the last strip's centroid with the middle strip's centroid, since the CRAM gives the desired average when the

[4] The pulse height is an 8-bit signed quantity; negative values have bit 7 set.

middle strip is the middle CRAM subaddress. Fire outputs a Valid bit and a 14-bit cluster centroid (10-bit chip+strip; 4-bit substrip).

By only asserting the FIFO's WEN* for valid Fire outputs, the HitMan writes clustered SVX data to its FIFO. WEN* is also asserted when Ready indicates that the end-of-event word should be written. A multiplexer switches the FIFO data lines between Fire's output and Ready's end-of-event word.

Each HitMan counts the number of clusters written for the current event. If the count exceeds a programmable limit, WEN* is suppressed for further clusters and an error signal is sent to Ready, causing it to flag a "truncated data" condition in the end-of-event word. This throttle has been implemented to prevent occasional high-occupancy events from clogging the SVT pipeline; instead, an error flag can inform Level 2 of an event that is too large to handle quickly. If the limit is set to zero (the default), then no limit is enforced.

The clustering engine has been successfully simulated at speeds up to about 35 MHz (30% above nominal), but the maximum speed has not been re-checked since a few changes (e.g. final choice of ISPY and CRAM components; addition of number-of-clusters cut; small fixes in Ready stage) have been made. The latency from HitMan input to HitMan output is about 0.95 nsec.

The HitMan allows VME read/write access to its ISPY and CRAM memories, its internal memories, and several internal registers (listed in Table 13). It also allows VME data to be written into the FIFOs, e.g. for testing the Merger.

2.5 Merger

The Merger, an Altera Max 7512AEQC208-7 chip (36% logic used), reads data from all ten FIFOs and writes them to a single data path. It is clocked by a 30 MHz oscillator, sent via the Boot chip. The Merger knows for which streams it has already seen an end-of-event word; among the remaining streams, it chooses one whose empty* flag is not asserted. Once all ten end-of-event flags have been set, all are cleared, and any non-empty FIFO can be read out. The FIFOs are grouped as two sets of five; each set shares a data path to the Merger. On even clock cycles, the Merger considers streams 0–4; on odd clock cycles, streams 5–9. A test mode can be enabled that forces the Merger to read all of stream 0 (until end-of-event is seen), then all of stream 1, and so on, up to stream 9. In this mode, the Merger necessarily outputs data at 15 MHz.

The Merger sees each FIFO's empty* flag, and sends read-clock and read-enable* signals to each FIFO individually. The FIFOs are clocked continuously (but not necessarily read-enabled), because the empty* flags are updated on positive read-clock edges. The FIFO read-enable* and output-enable* lines are tied together.

A stream label is prepended to each 18-bit data word (Table 6) read from

Byte $2n$	**Byte** $2n+1$
HDI ID *	HDI ID *
bunch crossing #	back-end state *
axial chip ID	status *
strip #	pulse height
strip #	pulse height
...	...
axial chip ID	status *
strip #	pulse height
strip #	pulse height
...	...
stereo chip ID	status *
strip # *	pulse height *
strip # *	pulse height *
... *	... *
stereo chip ID *	status *
strip # *	pulse height *
strip # *	pulse height *
... *	... *
EOR *	EOR *
... *	... *
EOR *	EOR *

Table 5: SVX-II raw data format. Bytes ignored by the Hit Finder are marked with an asterisk. Event header words can not be distinguished by their contents, so "data valid" transition must be observed. Axial chip IDs are 101xxxxx. Stereo chip IDs are 100xxxxx. Strip numbers are 0xxxxxxx. EOR bytes are 11xxxxxx. Pulse heights are 8-bit signed quantities; negative values have bit 7 set. All chip IDs occur, in order, even when a chip has no strips above readout threshold.

17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
not used		EE	LC	chip (0–6)			strip (0–127)							substrip (0–15)			
not used		EE	TD		ID					bunch ID							

Table 6: Clustering engine output data format for a normal word (above) and an end-of-event word (below). EE is set for end-of-event word. LC is long-cluster flag. TD indicates truncated data. ID indicates invalid data. Unlabeled end-of-event bits are spare error bits, ignored by the MOP, which the HitMan currently fills with diagnostic data to supplement the two defined error bits.

each FIFO. The stream number logically would fit into four bits, but to simplify the Merger's internal logic, six bits are used: a one-hot-bit encoding for stream modulo five (with no bit set when no data are present) and a sixth bit to distinguish streams 0–4 from streams 5–9.

Figure 8 shows the Hit Merging sheet of the schematic. Table 8 describes the HM_Afifoio(37:0) and HM_Bfifoio(37:0) buses that connect the FIFOs to the Merger. Table 7 summarizes the Merged_Data(25:0) bus that connects the Merger to the MOP.

0–17	See Table 6
18	Set if stream 0/5 read out
19	Set if stream 1/6 read out
20	Set if stream 2/7 read out
21	Set if stream 3/8 read out
22	Set if stream 4/9 read out
23	Clear: stream 0–4; set: stream 5–9
24	30 MHz clock
25	"Hold" (from FP, via MOP) to suspend output

Table 7: Merged_Data bus pin assignments.

The Merger provides VME read/write access to several internal registers (listed in Table 13 and read access to the FIFOs and their empty* flags.

The Altera compiler reports that the Merger can be run at 38 MHz (26% above nominal). We have successfully simulated the Merger at roughly 50% above nominal speed, but we have not re-checked this maximum speed since we migrated

0–17	See Table 6
18	0/5 empty*
19	0/5 full*
20	0/5 read-enable*
21	0/5 read clock
22	1/6 empty*
23	1/6 full*
24	1/6 read-enable*
25	1/6 read clock
26	2/7 empty*
27	2/7 full*
28	2/7 read-enable*
29	2/7 read clock
30	3/8 empty*
31	3/8 full*
32	3/8 read-enable*
33	3/8 read clock
34	4/9 empty*
35	4/9 full*
36	4/9 read-enable*
37	4/9 read clock

Table 8: HM_Afifoio, HM_Bfifoio bus pin assignments. Note that FIFO full* flags pass through Merger box on schematic but physically go straight to MOP.

from a 7256 to the larger 7512 device.

2.6 Output Processor

The Merger Output Processor (MOP) is an Altera Flex 10K50VBC356-2 chip (62% logic used). It accepts ten merged streams' cluster data from the Merger. An internal memory lookup translates stream numbers into layer and barrel numbers. Table 10 describes the MOP output. The MOP looks for each stream's end-of-event word, accumulates errors reported by the clustering engines ("invalid data," "truncated data"), and checks that all bunch-crossing IDs match ("lost sync"). The MOP also monitors the G-links' four LNKRDY* signals ("lost lock") and the FIFOs' ten full* signals ("FIFO overflow"). The MOP suppresses the HitMan end-of-event words as they pass through. Once it has seen the end of all streams, it writes its own end-of-event word, which contains error flags, the bunch ID, and an event parity bit.

The MOP communicates with the Front Panel via bus MOPped_Hits(28:0). The Front Panel has two I/O connectors (one reserved for future expansion). There are 23 data lines, common to both connectors. Each connector receives a its own Data Strobe signal (so that the MOP can be programmed to send certain words to only one FP connector). Each FP connector sends the MOP its own Hold signal (for flow control); the MOP sends the logical OR of the two Hold signals to the Merger.(5) Finally, the MOP sends two active-low signals to the Front Panel, one per FP connector, which are asserted for the duration of any clock cycle in which the MOP writes valid data. The Data Strobes could not be used for this purpose because the Front Panel's 74HCT123 monostables require a minimum input pulse width of 25 nsec. Table 9 describes the MOPped_Hits bus.

0–22	See Table 10
23	Data Strobe 1
24	Hold 1
25	Data Strobe 2
26	Hold 2
27	DS LED 1
28	DS LED 2

Table 9: MOPped_Hits bus pin assignments.

The SVT Data Strobe signal is a strobe, not a free-running clock: It runs only when there is a valid word to send. Because gate delays can be difficult to predict

(5) A VME-programmable Hold override may be a useful addition to the MOP.

within an Altera chip, the MOP does not gate the clock internally. Instead, it sends out a clock and a Valid signal. The Valid signal is generated by an inverted clock so that it is one-half clock cycle ahead of the corresponding data word; it is then delayed 10 nsec so that it switches about 6 nsec before the positive clock edge. The NAND of the delayed Valid signal and the clock is computed by a fast 74AS00 gate ($1 < \delta t < 5$ nsec) and used as the Data Strobe. Since this is a bit confusing, it is illustrated in Figure 3.

(file=strobe.ps)

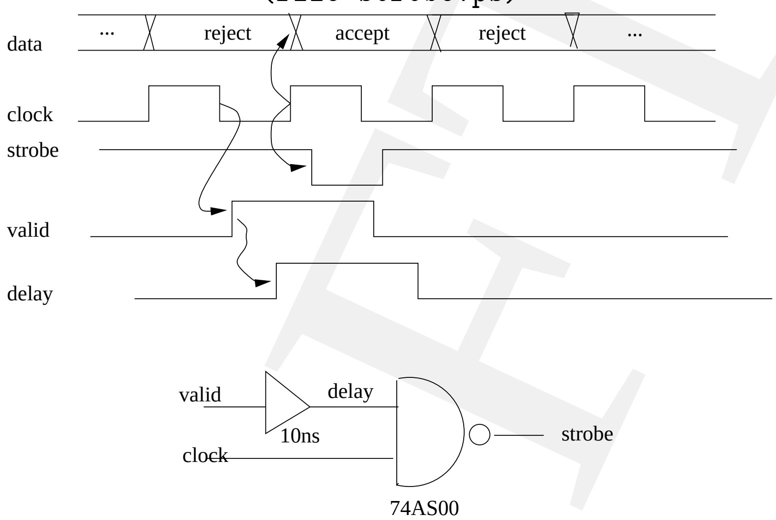

Figure 3: Data strobe generation. The MOP outputs a clock and a Valid signal. External logic generates Data Strobe (called DS_ in the SVT protocol document [13]). According to the protocol, the quiescent state must be high, and the data must be stable for > 6 nsec before the positive DS_ edge and > 10 nsec after the positive DS_ edge. By approximately centering the positive edge within the data, we meet these specifications.

Discussion of error registers goes here.

The MOP requires an Output SPY buffer (OSPY) that is at least 23 bits wide, which we implement with two Motorola MCM69F618CTQ9 64K×18 synchronous SRAMs. While 36 data bits are 50% more than we need, using the same component for ISPY and OSPY reduces the number of distinct components we must stock. The OSPY stores a copy of the last 64K output words written by the MOP, for debugging and validation. The OSPY can also be used as a source of data to be sent directly to the Front Panel. This test mode would be useful mainly to check front-panel connections or to provide test data to another SVT board. We have not yet checked this test mode in the simulation.

The MOP provides VME read/write access to several internal registers (listed in Table 13), to its internal barrel/layer memory, and to 64K×32 bits of OSPY memory.

The MOP has been successfully simulated at speeds well above 30 MHz. However, only fairly simple error conditions, such as "invalid data" reported by the HitMan, have been simulated. We still need to verify that the MOP correctly reports and recovers from serious errors such as G-link lost lock.

24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
AF	DS	EE	EP	layer (1–5)			barrel (1–6)			LC	chip (0–6)			strip (0–127)							substrip (0–15)			
AF	DS	EE	EP	error flags (12 bits)												PA	bunch crossing # (8 bits)							

Table 10: Hit Finder output data format for a normal word (above) and an end-of-event word (below). AF is "almost full," which we also call "hold." EE is "end of event" flag. EP is "end packet," which we always assert because HF outputs only single-word packets. PA is a parity bit for downstream error detection. DS is the active-low strobe signal described in Figure 3.

2.7 Front Panel

The Front Panel includes two 52-pin KEL 8822E-052-171H standard SVT protocol connectors, eight National DS90C031TM quad LVDS drivers (four per connector), and two National DS90C032TM quad LVDS receivers (one per connector). The connector pin assignments are shown in Table 11. The traces from the LVDS drivers to the front-panel connectors have been routed as differential pairs and kept below about two inches in length (the longest is 2.04 inches).

The Front Panel contains three connectors for Altera bit-blaster configuration and one Altera EPC1 configuration PROM. One connector goes to the Merger (bus Merger_Config(3:0)), for JTAG-based configuration. A second connector is multiplexed with the EPC1 (as in Crate Sum [12]), to configure the Boot chip (bus Boot_Config(4:0)). The third connector goes to ordinary I/O pins on the Boot chip (bus Bkup_Config(4:0)); the Boot chip multiplexes these pins with its own chip-configuration logic, so that we have the option to use a bit blaster as a work-around if the Boot chip fails to configure other Flex 10K chips correctly.

The Hit Finder expresses its internal state through twelve front-panel LEDs, described in Table 12.

Figure 9 shows the complete Front Panel sheet of the schematic. Figure 10 shows the connections from the MOPped_Hits bus to LVDS drivers/receivers and the front-panel connectors.

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	ground
A26	HOLD	B26	ground

Table 11: SVT protocol connector pin assignments (for connector that is signal source).

Name	Color	Position	SVT std	'123	'00
Power	red	tiptop, left	Y	N	N
VME Ack	yellow	tiptop, right	Y	Y	N
Boot Mode	red	neartop, left	N	N	N
Load Mode	yellow	neartop, middle	N	N	N
Test Mode	green	neartop, right	N	N	N
Strobe 1	green	above FP1, left	Y	Y	Y
Hold 1	red	above FP1, right	Y	Y	Y
Strobe 2	green	above FP2, left	Y	Y	Y
Hold 2	red	above FP2, right	Y	Y	Y
CDF Error	red	bottom, left	Y	Y	Y
SVT Error	yellow	bottom, middle	Y	Y	Y
Run Mode	green	bottom, right	Y	N	N

Table 12: Hit Finder front-panel LEDs. We add three LEDs that are not part of the SVT standard, to indicate whether the Hit Finder is in Boot, Load, or Test mode. The Ack, Boot, Load, Test, Run, CDF Error, and SVT Error LEDs are controlled by the Boot chip. The Strobe LEDs are controlled by the MOP. The Hold and Power LEDs' signals are generated local on the Front Panel schematic. '123 indicates a one-shot to make the LED illuminate for long enough to be visible. '00 indicates a NAND (active-low OR) gate to illuminate the LED continuously if the corresponding signal is asserted continuously.

3 VME I/O and Configuration

3.1 VME Data Path

3.2 VME Address Space

Chip	Device	Stream	Subaddr	What
Boot	0x00	any	0x0000	Brain register
Boot	0x00	any	0x0001	FRAM boot base address
Boot	0x00	any	0x0002	Clocks VME test clock
Boot	0x00	any	0x0003	Strobe HF_Init signal
Boot	0x0a	A(20:17)	A(16:1)	FRAM location A
HitMan	0x01	0–9/15	A(11:0)	pedestal RAM location A
HitMan	0x01	0–9/15	0x100N	chip ID N (0–7)
HitMan	0x01	0–9/15	0x101N	strip threshold N (0–7)
HitMan	0x01	0–9/15	0x102N	cluster charge cut N (0–7)
HitMan	0x01	0–9	0x1102	ISPY counter (read)
HitMan	0x01	0–9/15	0x1103	ISPY counter reset (write)
HitMan	0x01	0–9/15	0x1104	max # clusters
HitMan	0x01	0–9/15	0x1105	# − 1 expected chip IDs
HitMan	0x01	0–9/15	0x1106	set chip IDs to defaults (write)
HitMan	0x02	0–9/15	A(15:0)	ISPY location A
HitMan	0x03	0–9/15	A(15:0)	CRAM location A
HitMan	0x04	0–9/15	any	FIFO (write)
Merger	0x04	0–9	any	FIFO (read)
Merger	0x07	any	0x0000	stream disable mask
Merger	0x07	any	0x8000	FIFO empty* flags

Table 13: VME address space (part 1).

Chip	Device	Stream	Subaddr	What
MOP	0x05	0	0x000N	barrel/layer map N (0-9)
MOP	0x05	0	0x1000	stream disable mask
MOP	0x05	0	0x1010	OSPY counter
MOP	0x05	0	0x1020	EE word error mask (15:0)
MOP	0x05	0	0x1021	EE word error clear (15:0)
MOP	0x05	0	0x1022	CDF error mask (15:0)
MOP	0x05	0	0x1023	SVT error mask (15:0)
MOP	0x05	0	0x1024	error register (15:0)
MOP	0x05	0	0x1030	EE word error mask (31:16)
MOP	0x05	0	0x1031	EE word error clear (31:16)
MOP	0x05	0	0x1032	CDF error mask (31:16)
MOP	0x05	0	0x1033	SVT error mask (31:16)
MOP	0x05	0	0x1034	error register (31:16)
MOP	0x05	0	0x1040	EE word error mask (43:32)
MOP	0x05	0	0x1041	EE word error clear (43:32)
MOP	0x05	0	0x1042	CDF error mask (43:32)
MOP	0x05	0	0x1043	SVT error mask (43:32)
MOP	0x05	0	0x1044	error register (43:32)
MOP	0x05	0	0x1035	reset EE word error flags
MOP	0x05	0	0x1036	reset error register
MOP	0x06	0	A(15:0)	OSPY location A

Table 14: VME address space (part 2).

3.3 Loading Memories from FRAM

3.4 Altera Configuration

3.5 P2 Slow-Control Signals

4 Power

5 Prototype Test Sequence

6 Logic Analyzer Connectors

7 Various and Sundry

Figure 13 is a parts-list spreadsheet. We need to check this against the CAD system's component catalog and see if we missed any small parts.

References

[1] S. Belforte *et al*, "Silicon Vertex Tracker Technical Design Report," CDF Note 3108 (April, 1995).

[2] J. Berryhill *et al*, "Interfaces to the SVT Hit Finder Board," (April, 1997). (See `http://hep.uchicago.edu/cdf/general_upg/svt/hitfinder.html` for text.)

[3] J. Berryhill *et al*, "Conceptual Design of the SVT Hit Finder Board," (April, 1997). (See above URL.)

[4] J. Berryhill *et al*, "Preliminary Component Specification of the SVT Hit Finder Board," (July, 1997). (See above URL.)

[5] R. Culbertson, "A Clustering Engine for the SVT Hit Finder Board," (September, 1996). (See above URL.)

[6] J. Berryhill *et al*, "Error Handling of the SVT Hit Finder Board," (February, 1998). (See above URL.)

[7] R. Blair *et al* (CDF II Collaboration), "The CDF II Detector Technical Design Report," FERMILAB-Pub-96/390-E (November, 1996).

[8] SVX3 reference goes here.

[9] VRB reference goes here.

[10] VTM reference goes here.

[11] Merger board reference goes here.

[12] Crate Sum reference goes here.

[13] CDF 4578 (SVT inter-board protocol) reference goes here.

(file=hfsheet01.ps)

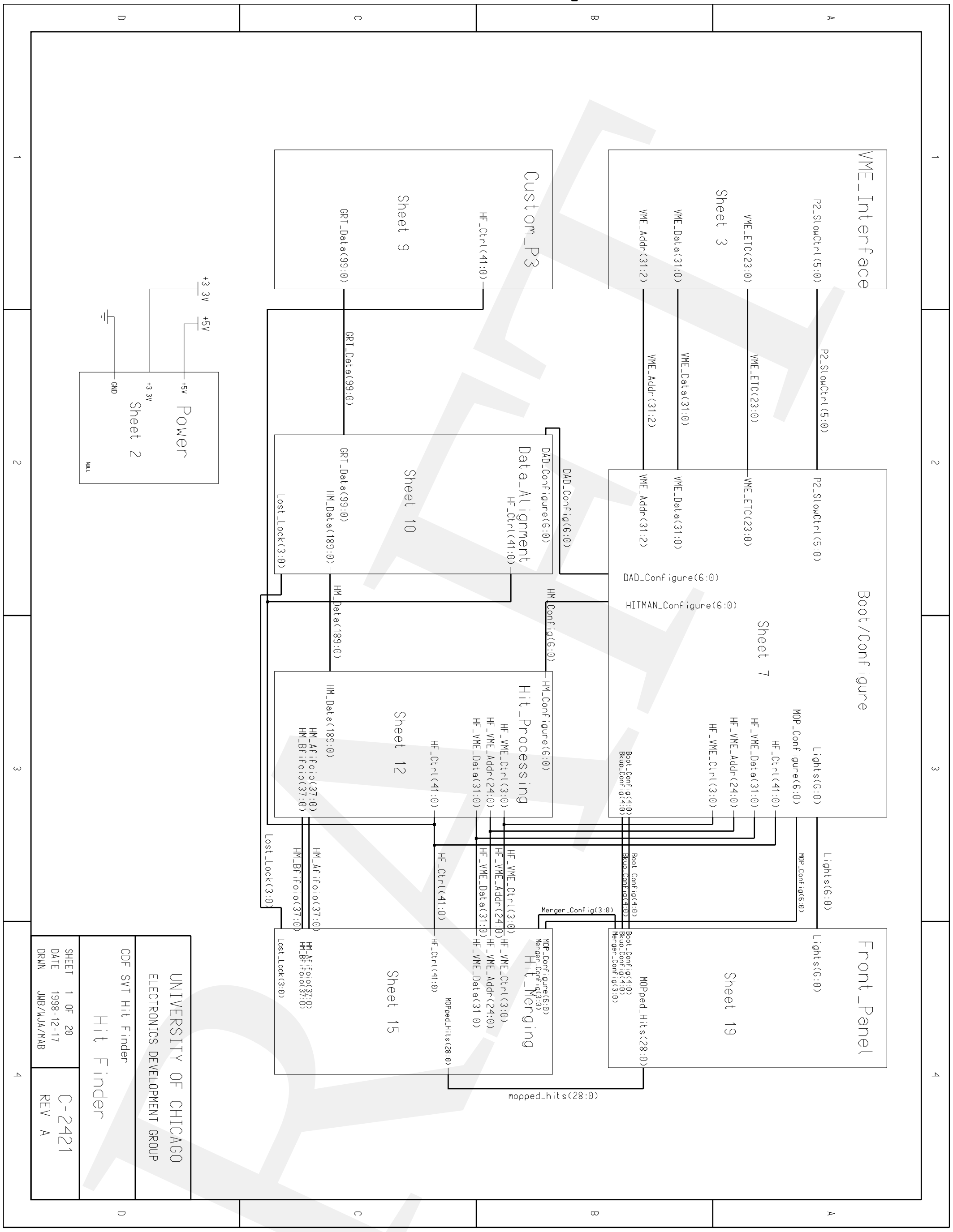

Figure 4: Hit Finder top-level schematic.

(file=layout.ps)

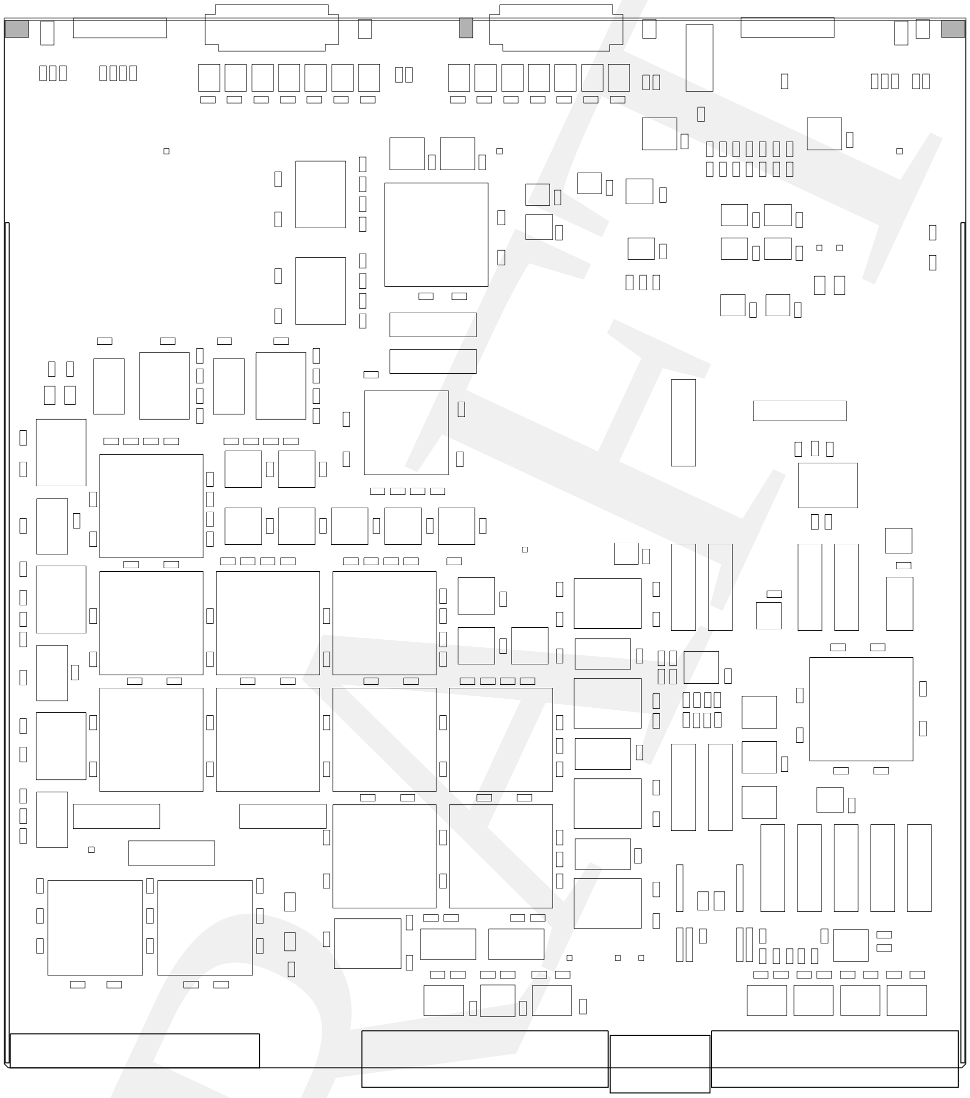

Figure 5: Hit Finder board layout.

(file=hfsheet13.ps)

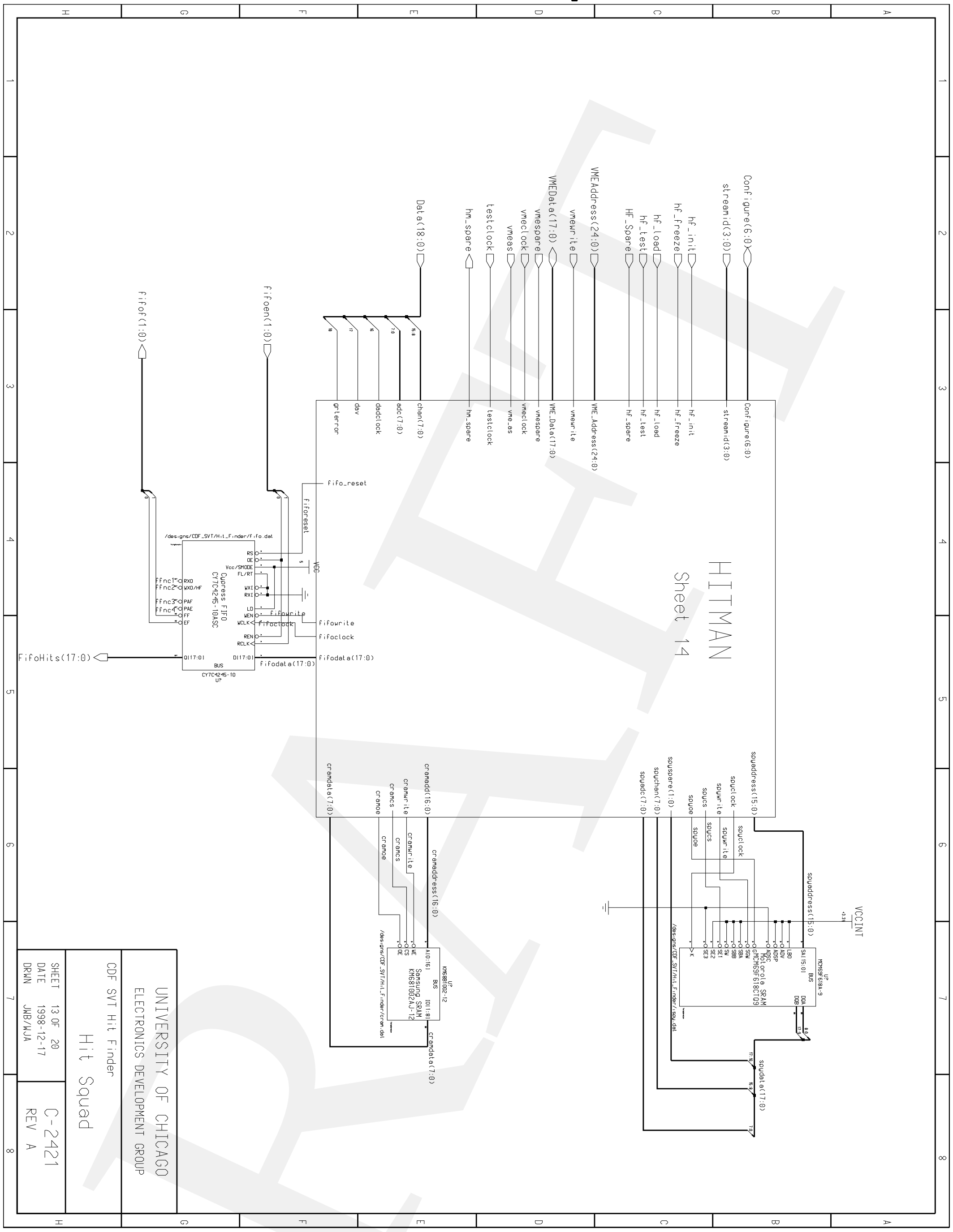

Figure 6: Schematic for one clustering engine.

(file=hfsheet12.ps)

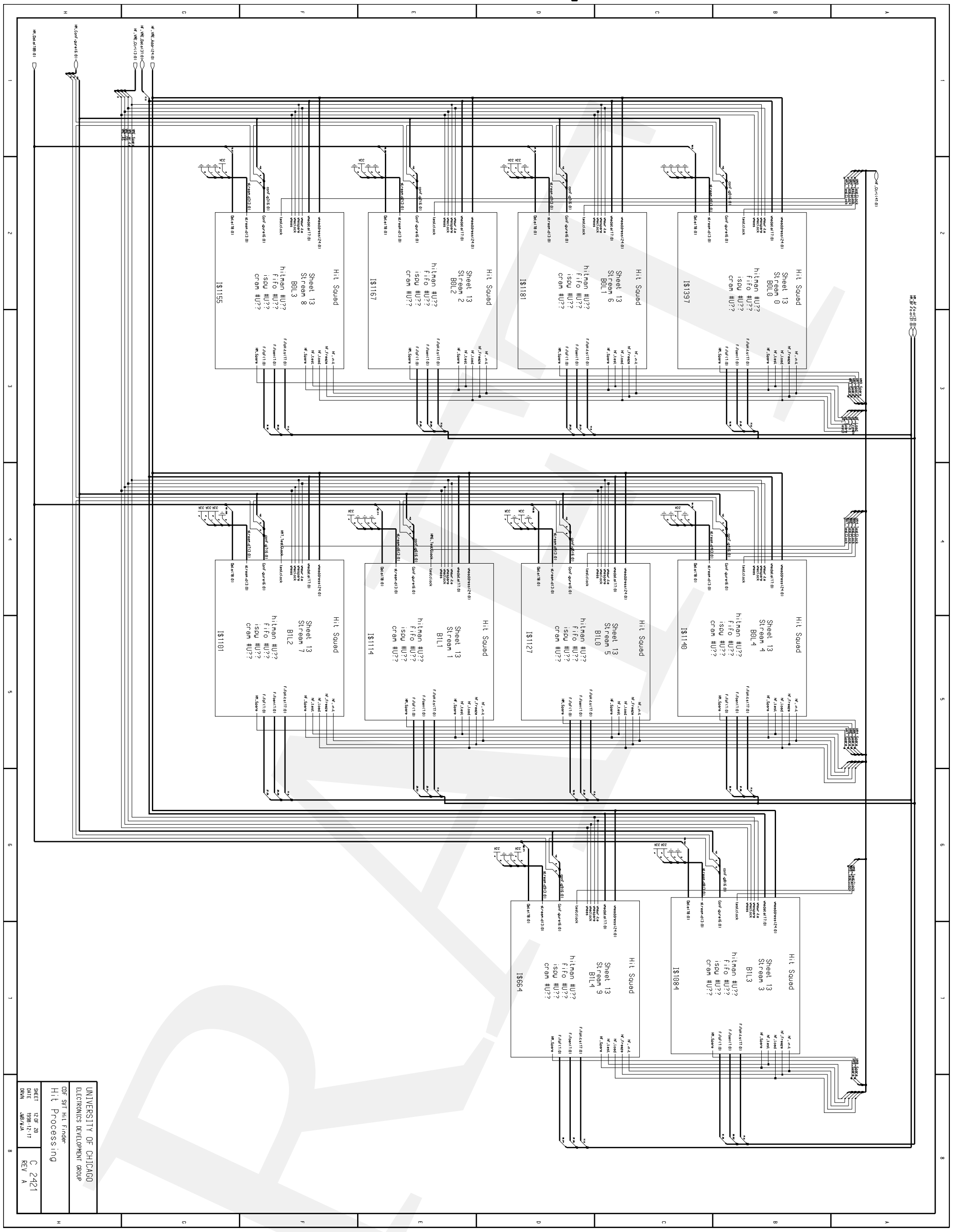

Figure 7: Hit Processing sheet of schematic.

(file=hfsheet15.ps)

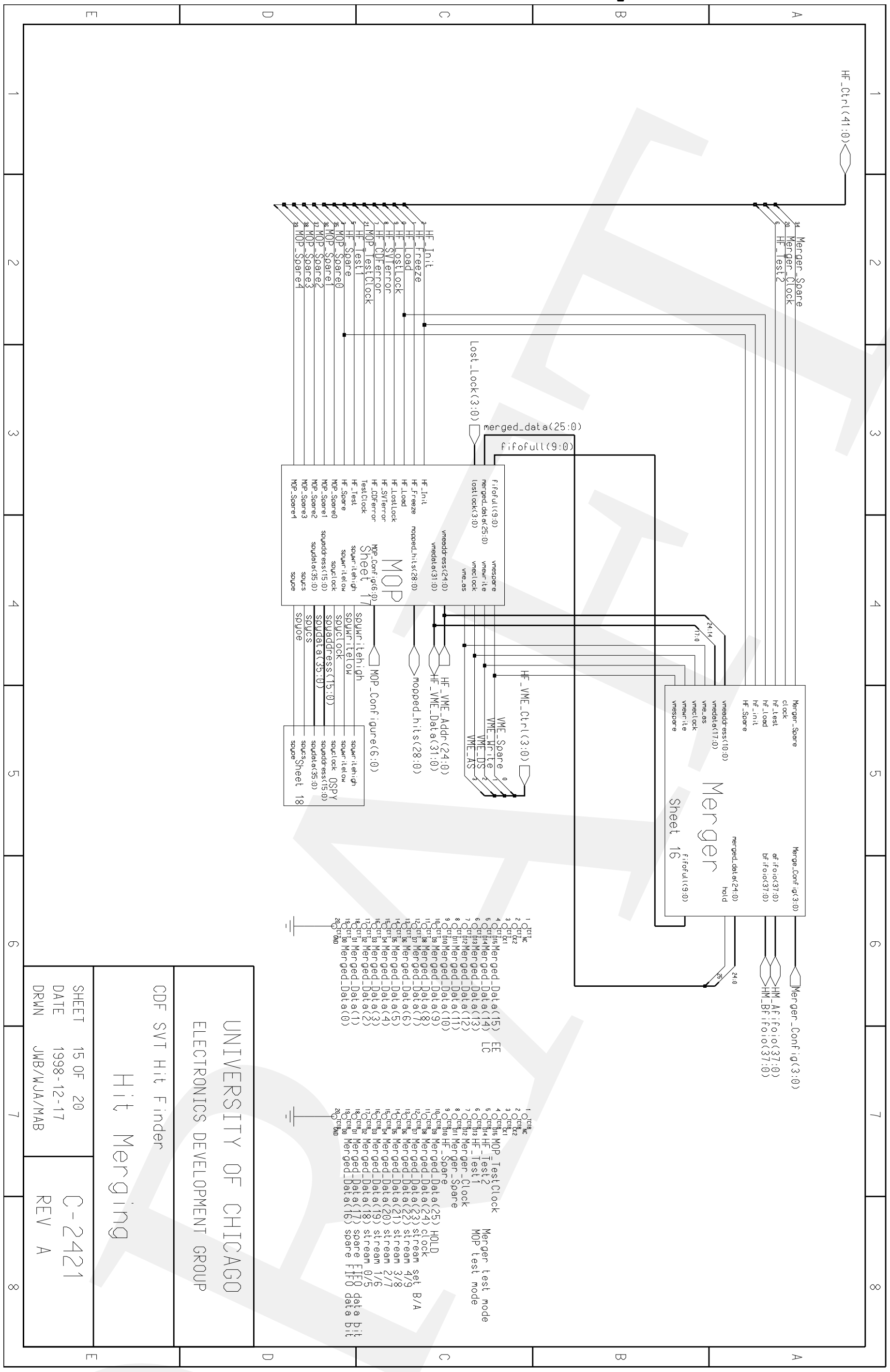

Figure 8: Hit Merging sheet of schematic.

(file=hfsheet19.ps)

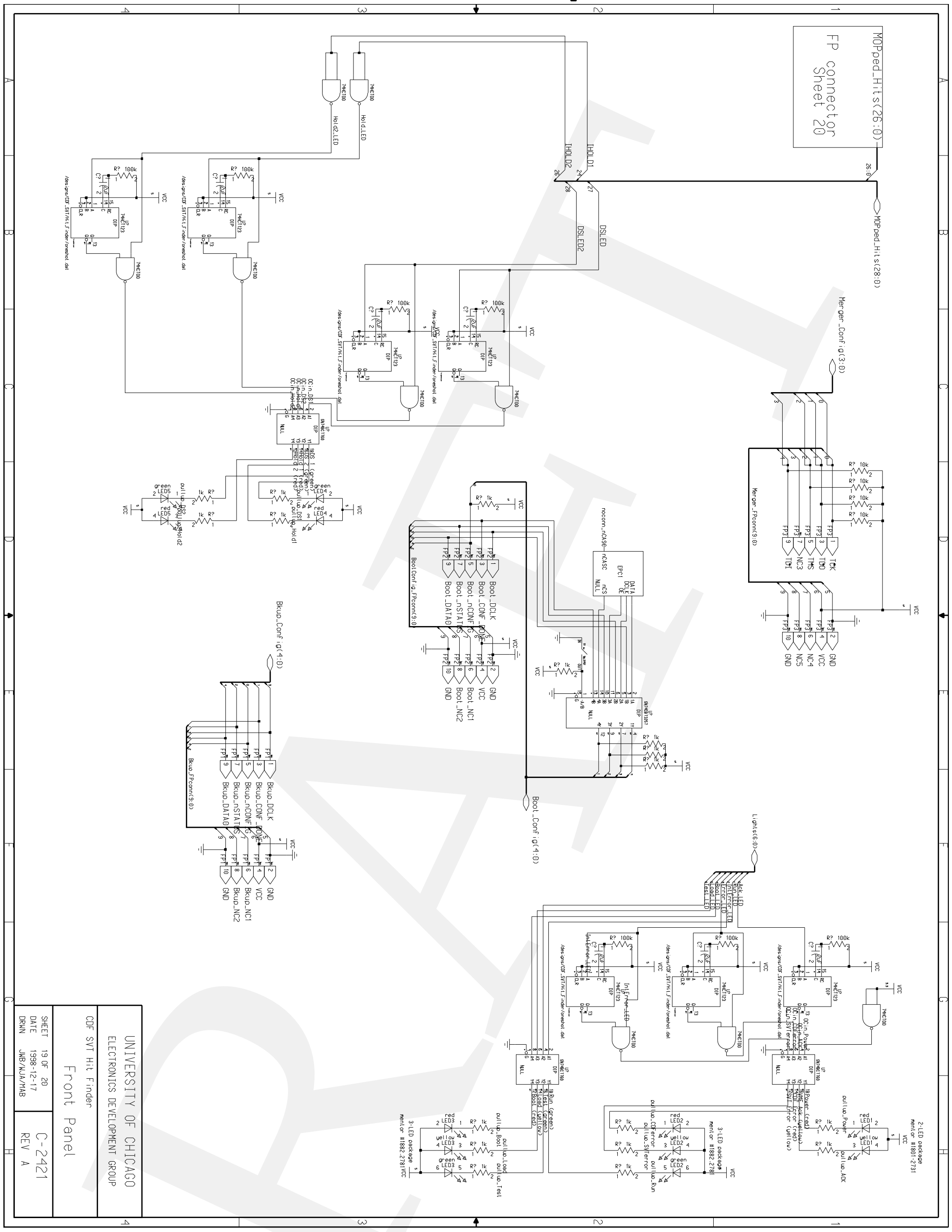

Figure 9: Hit Finder Front Panel schematic.

(file=hfsheet20.ps)

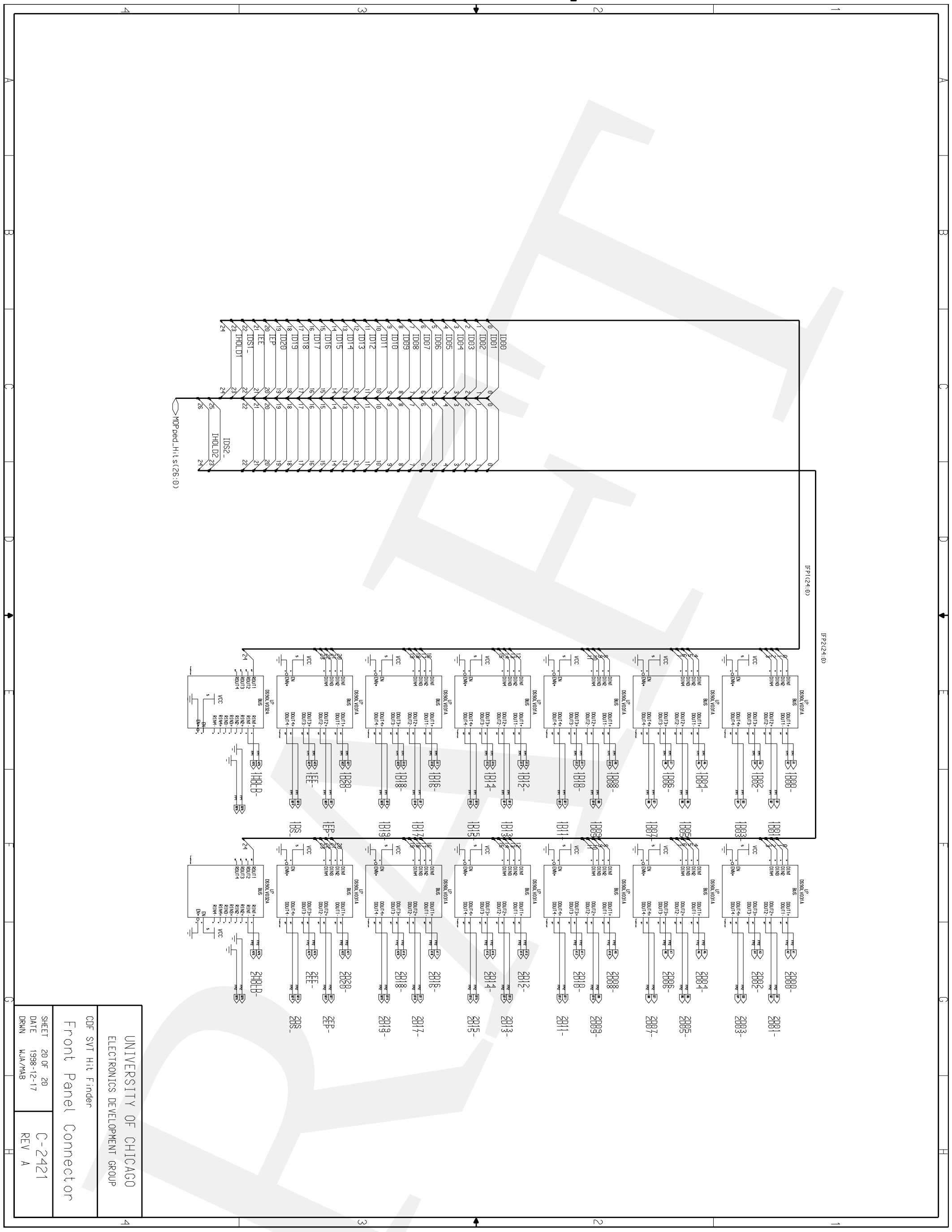

Figure 10: Hit Finder Front Panel Connectors schematic.

(file=hfsheet03.ps)

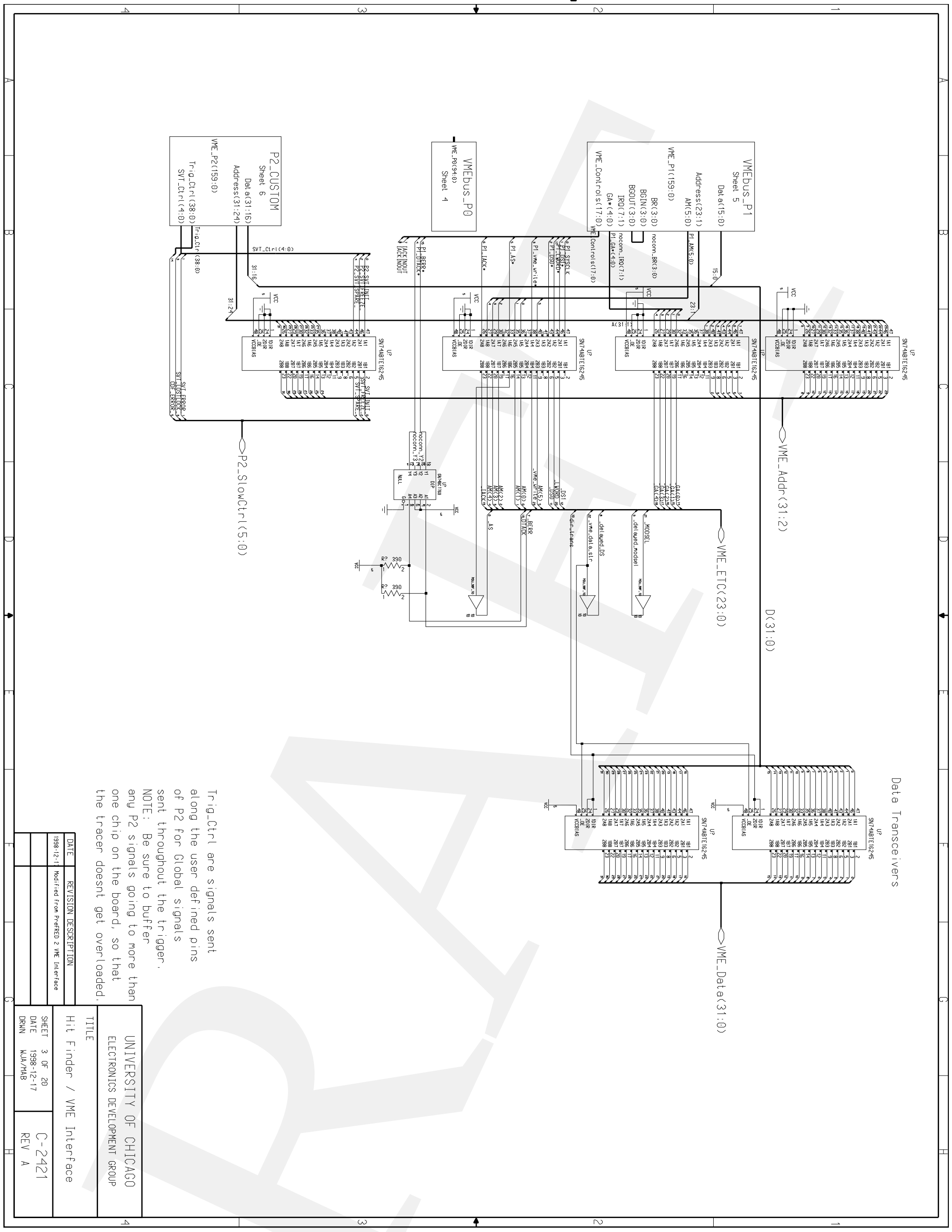

Figure 11: Hit Finder VME Interface schematic.

(file=hfsheet07.ps)

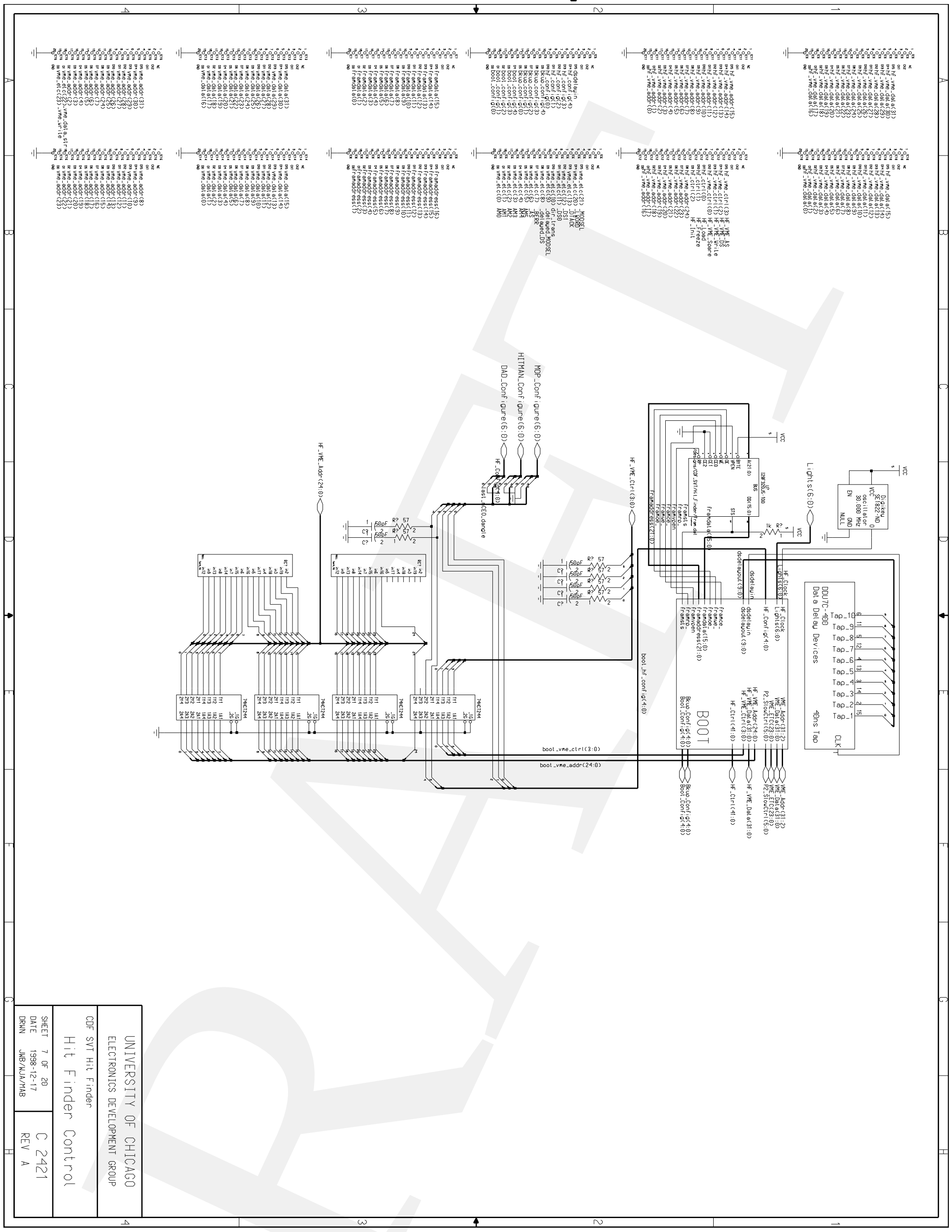

Figure 12: Hit Finder Boot/Configure schematic.

	SVT Hit Finder Parts List		Revised 1999-01-11 by WJA										
Purpose	**Vendor**	**Part Number**	**Description**	**Package**	**# to order**	**$/order**	**#/board**	**$/unit**	**$/board**	**# spare**	**Comment**	**When?**	**In?**
DAD	Altera	EPF10K30AQC240-1	FPGA	PQFP 240	5	530	2	106	212	1		1998-10-21	Y
Hitman/MOP/Boot	Altera	EPF10K50VBC356-2	FPGA	BGA 356	26	3406	12	131	1572	2		1998-10-21	Y
VME Interface	Altera	EPM7128SQC160-7	CPLD	PQFP 160	0	0	0	43	0		donated 3 to Track Fitter	1998-10-21	Y
Merger (original)	Altera	EPM7256AQC208-7	CPLD	PQFP 208	0	0	0	83	0		returned 3 to Wyle	1998-10-21	R
Merger (upgrade)	Altera	EPM7512AEQC208-7	CPLD	PQFP 208	3	474	1	158	158	1		1999-01-08	N
Boot (original)	Altera	EPM9560ABC356-10	CPLD	BGA 356	0	0	0	150	0	1	returned 3 to Wyle	1998-10-21	R
FIFO	Cypress	CY7C4245-10ASC	FIFO	A64 (10x10 TQFP 64)	22	880	10	40	400	2		1998-10-21	Y
VME DS/ack delay	Data Delay Devices	DDU7C-400	10 tap 400ns delay line	DIP 16	3	60	1	20	20			1998-10-21	Y
VME interf delay line	Data Delay Devices	MDU38F-10	10ns delay	DIP 8	5	100	1	20	20			1998-10-21	Y
MOP clock socket	Digikey	A463-ND	Socket for 4-pin oscillator	DIP 8	3	3	1	1	1		Aries (half-size)	1998-10-21	Y
MOP clock (slow)	Digikey	SE1243-ND	25.576 MHz Oscillator	DIP 8	1	3	0	3	0	1	Epson #SG-531P-25.576MC	1998-10-21	Y
MOP clock	Digikey	SE1822-ND	30.000 MHz Oscillator	DIP 8	3	9	1	3	3		Epson #SG-531PTJ-30.000MC	1998-10-21	Y
MOP clock (fast)	Digikey	SE1828-ND	36.000 MHz Oscillator	DIP 8	1	3	0	3	0	1	Epson #SG-531PTJ-36.000MC	1998-10-21	Y
VME receivers	IDT	74FCT541ASO	Octal buffer/receiver	SOIC 20	12	0	4					1998-10-21	Y
LVDS drivers	National	DS90C031TM	quad LVDS driver	M16A (SOP 16)	27	216	12	8	96			1998-10-21	Y
LVDS receiver	National	DS90C032TM	quad LVDS receiver	M16A (SOP 16)	5	40	2	8	16			1998-10-21	Y
clock/strobe gate	Philips	74HCT08	Quad AND gate	SOIC 14	5	0	1					1998-10-21	Y
VME transceivers	TI	SN74ABTE16245DL	16 bit transceiver	SOIC 48	8	0	2					1998-10-21	Y
VME o/c buffer	TI	SN74BCT760DW	Octal O/C buffer	SOIC 20	8	0	2					1998-10-21	Y
Flash RAM	Intel	DA28F320J5-100	StrataFlash FRAM	SSOP 56	3	60	1	20	20			1998-10-22	Y
ISPY/OSPY	Motorola	MCM69F618CTQ9	BurstRAM 64Kx18	TQFP 100	30	450	12	15	180			1998-11-05	Y
CRAM	Samsung	KM681002AJ-12	CMOS SRAM 128Kx8	32-SOJ-400	25	106.25	10	4.25	42.5			1998-11-05	Y
Spare delay lines	Data Delay Devices	MDU38F-10	10ns delay	DIP 8	5	100	1	20	20			1998-11-05	Y
PS config delay line	Data Delay Devices	MDU38F-50	50ns delay	DIP 8	5	100	0	20	0			1998-11-05	Y
LED stabilizer	Philips	74HCT123D	Monostable	SO16	20	0						1998-11-05	Y
Misc logic	Philips	74HCT00D	Quad 2-input NAND	SO14	20	0						1998-11-05	Y
Boot clock	Digikey	SE1217-ND	2.000 MHz Oscillator	DIP 8	3	9	0	3	0		Epson #SG-531P-2.000MC	1998-11-16	Y
Boot clock socket	Digikey	A463-ND	Socket for 4-pin oscillator	DIP 8	3	3	0	1	0		Aries (half-size)	1998-11-16	Y
P0 connector	ERNI	#64784	95-pin connector		2		1				from Fermilab		
P1/2 connectors	Harting	#0201 160 2101	160-pin connector		4		2				from Fermilab		
P5 connector	ERNI	#64785	110-pin connector		2		1				from Fermilab		
P6 connector	ERNI	#64179	125-pin connector		2		1				from Fermilab		
FP connectors	KEL	#8822E-052-171H	connector		4		2				from Trieste		
Logic probe connectors	3M	#2520-6002	20-pin header	.100" × .100" low-profile			17						
Fuses	Littelfuse		Fuse	Axial Lead									
decoupling capacitors?													
FP LEDs (bottom)													
FP LEDs (top)													
FP LEDs (top)													
pullup resistors?													
Per-board parts price	2760.5												
Total $ parts ordered	6552.25												

(file=parts.ps)

Figure 13: Parts list spreadsheet.