

Oct 23, 2002

GAFE
GLAST ACD Front End Electronics ASIC
Test Report
Version: 2 (GAFE2)

Oct 23, 2002
(earlier rev dated: 10.10.02)

Gamma Ray Large Area Space Telescope (GLAST)
Laboratory for High Energy Astrophysics, Code 660
Goddard Space Flight Center
NASA, Greenbelt, MD 20771

Author: Satpal Singh
(Satpal@lheapop.gsfc.nasa.gov)

Contents for Digital section Provided by: Dave Sheppard

Tests Assisted by:
Bob Baker
Larry Olsen
Deneen Ferro

Scope:

The scope of this document is to describe the test results of the GLAST ACD ASIC version 2, referred to as GAFE2 for the Front End Electronics. This chip was fabricated in Ag/HP 0.5 um CMOS process, wire bonded into plastic chip carriers. The test results and problems are documented. Since all the functions of GAFE2 couldn't be tested, referral to GAFE1 Test report might be useful as GAFE1 and GAFE2 have identical analog modules, except that GAFE2 has DACs and associated buffers and logic on chip.

Table of Contents

<u>Introduction:</u>	4
<u>GAFE description</u>	4
<u>Test Configuration</u>	4
<u>Functional Tests</u>	6
<u>Problems</u>	29

1. Introduction:

This Document describes the test the configuration, methods and results of functional testing of the GLAST ACD Front-end Electronics ASIC (GAFE2). Since all the functions on GAFE2 couldn't be tested, one may refer to the Test Report on the earlier version, Proto 1_1, (GAFE1).

2. GAFE2 description

The GAFE ASIC is the front-end analog interface to the PMTs providing signal conditioning of the ionizing event signals for subsequent digitization and processing. The GAFE ASIC provides amplification, shaping, discrimination and timing functions for signal processing by the ACD. The GAFE ASIC is described in more detail in another document that describes the ASIC requirements and architecture.

The GAFE2 architecture can be divided into two sections, the analog and the digital. The analog section of GAFE2 is identical to that in GAFE1 except that it contains DACs and its associated buffers. The DACs are used to provide the threshold references for the comparators and to set the bias points.

The digital logic is used primarily to program the DAC registers and it communicates to GARC via a serial link.

3. Test Configuration

The chips were mounted in 52 pin plastic carriers. For purposes of testing, a printed circuit test board was made with a clamshell to accommodate the 52 pin plastic packaged chips. The use of the clamshell facilitated easy replacement of the chips to test multiple chips.

The GAFE test board also included the voltage reference to provide 2.5 v needed by GAFE, the ADC circuit that digitizes the output of Track and Hold of GAFE, and the connectors to hook up to the GARC simulator or GARC test board. In order to program the GAFE registers for the DACs, it is necessary that the GAFE test board be hooked up to either the GARC simulator test board or the GARC test board and the GSE board, both of which are interfaced to a PC for testing and user interface.

Fig.1 shows the GAFE Test board, only one half of the board has been populated to test only one GAFE2 chip. In Fig.2 is shown the picture of GAFE2 Test board hooked to a GARC Test board which in turn is hooked to a GARC simulator board, which then connects to a PC via a serial link. For testing purposes, the GAFE Test board could well be connected to a GARC simulator board instead of the GARC test board. When connected to the GARC simulator board, the GSE board is not required as the GARC simulator board connects directly to the PC.

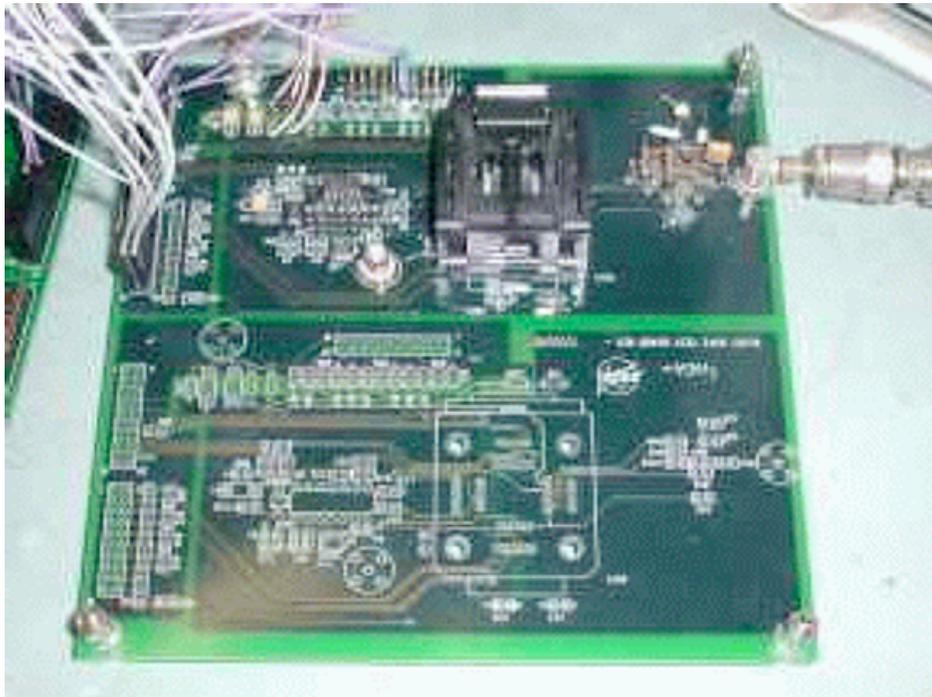


Fig.1: GAFE2 Test Board, only one half board populated for testing of one GAFE chip



Fig.2: GAFE2 Test setup

The tests were conducted using standard lab test equipment like "Tail Pulsers", Square wave generator, power supplies, digital oscilloscope, etc. The method and the test results are described below.

4.0 Functional Tests

The various tests performed are described next.

4.1 Power

The test board was powered by 3.3v dc. Vcc (analog supply), and the Vdd (digital supply) are derived from the same 3.3v supply but isolated from each other by RC filtering of 100 ohms and 10uF tantalum paralleled with 0.1 ceramic capacitors. The current draw by both the analog and digital portions of the ASIC produced a voltage drop across the 100 ohm resistor. The currents measured were:

Analog: 0.23 ma (Power = 3.3v * 0.23 ma = 2.5mw)

Digital: 0.51ma (Power = 3.3 v * 0.51 ma = 1.683 mw)

4.2 Shaping Amp

A quick check was made by applying a tail pulse at the input and also by injecting charge through a capacitor. The input resistance to the low energy channel (or the High gain channel) was 12 K, and that to the high energy channel (or the low gain channel) was 1.2 Meg.

Shaping amp gain and timing data was measured using the HP oscilloscope with 10x probes, ac couple with 1meg impedance.

The Peaking time was measured to be about 6 us compared to the designed value of 3us. This was attributed to the fabricated Nwell resistances being twice the value.

The linearity check of the shaping amp however revealed a non linearity which was not observed in the earlier generation GAFE1. It was seen that when the shaping amp output was around 0.75 V, the output suddenly started jumping towards the maximum value and saturate. This is shown in the scope plots shown below which show several accumulated traces. For these tests, the charge was injected through a 33pf cap, the resistance to salo was 12k, to sahi was 1.2Meg, and to discin was 120k.

Next the 12k and 1.2 Meg resistances were swapped, so that 12K was connected to sahi, and 1.2 Meg to salo. Now the high energy channel shaping out exhibited the same behavior seen on the low energy channel. This was expected as the two channels are identical on the chip except that the low energy channel has a diode across the feedback of its input.

It was also seen that when the low energy channel went to saturation, the high energy channel also saturated. This was later simulated due to cross coupling between two channels due to non ideal buffers in the DAC.

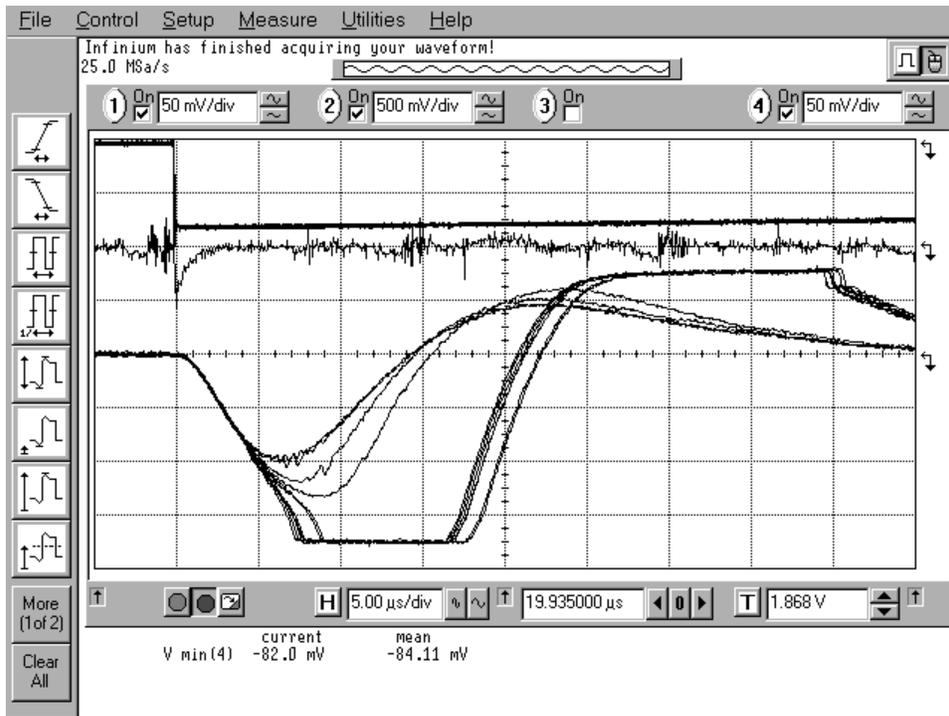


Fig. 1: Top trace is the input pulse, middle trace is the waveform at the node common to 33pf, 12k and 1.2 Meg. The bottom trace is the SA output for low energy channel

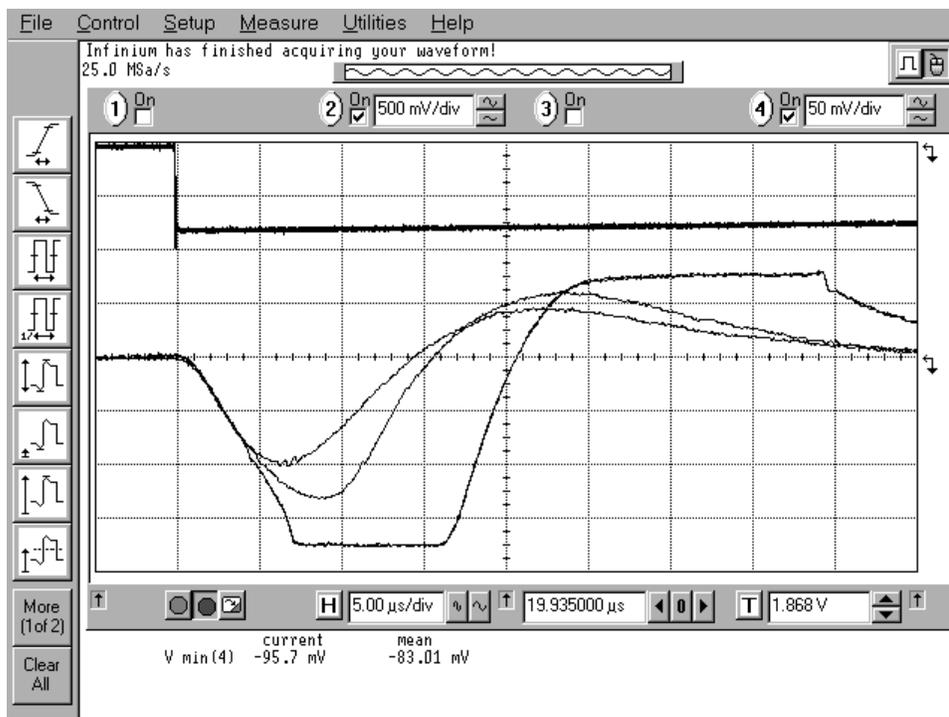


Fig. 2: Input pulse (top) and SA output (bottom)

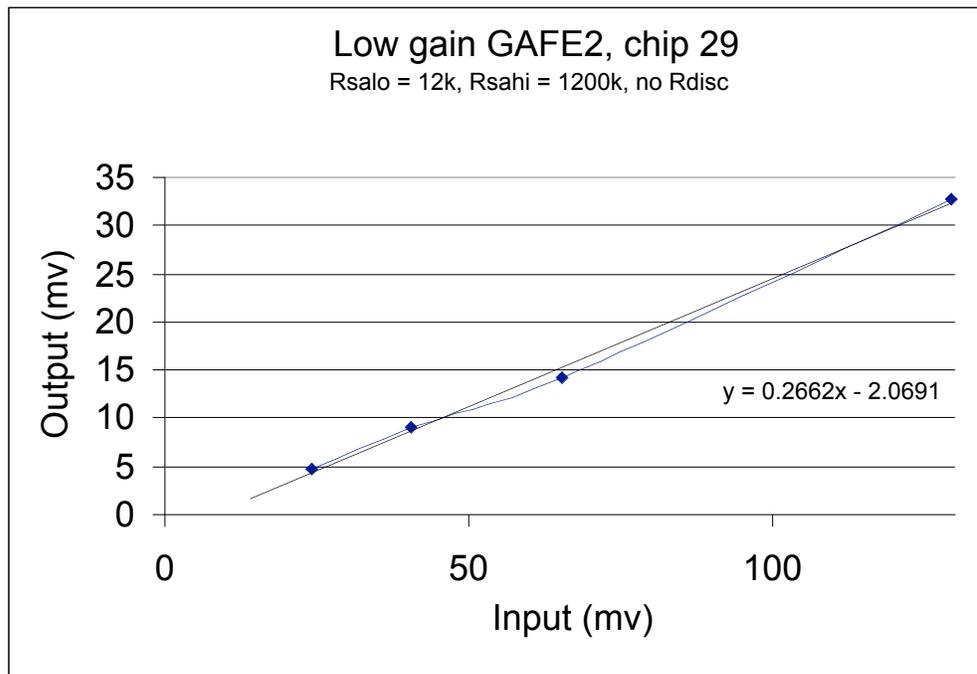
The cause of the above non linearity was traced to the high output impedance of the buffer providing the bias, VB_{SA}, to set the baseline of the shaping amps. Since the bias point V_{bsa} was not a zero impedance node but at a relatively high output impedance of the buffer amp, there resulted a ripple at this node causing the output to saturate. Also since the V_{bsa} node is common to both the channels, any ripple at this node will show up on both the channels and hence the low energy channel cross coupled to the high energy channel. The simulations with imperfect buffers also showed plots similar to those shown in Fig.1 and 2 above. On improving the buffers, the simulations indicated that disappearance of the above problem which was fixed in GAFE3 now in fab.

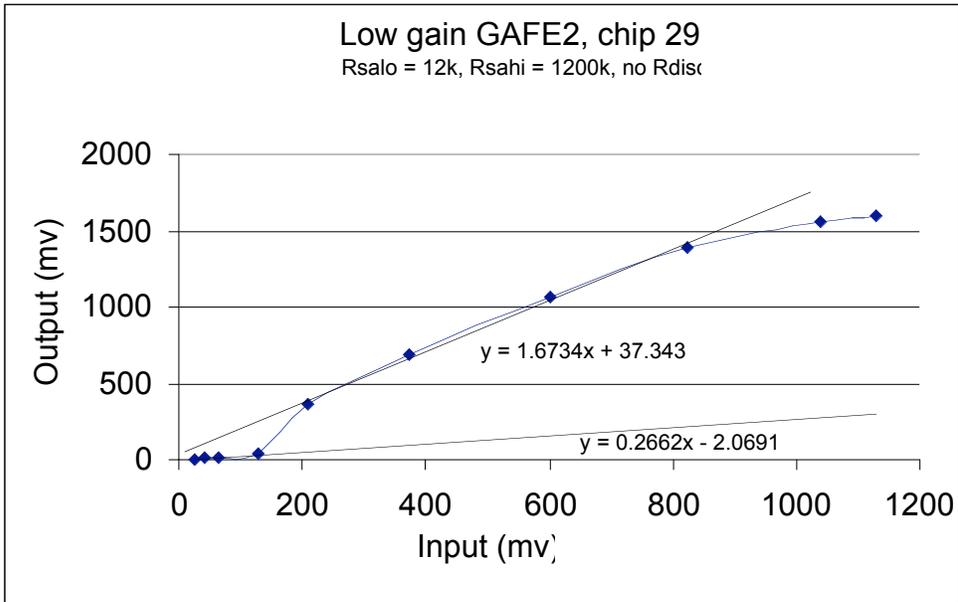
4.2.b PHA Linearity:

In order to characterize the chip, the PHA linearity was measured by injecting charge through a 33pf capacitor and using 12K to salo and 1.2 Meg to sahi inputs respectively.

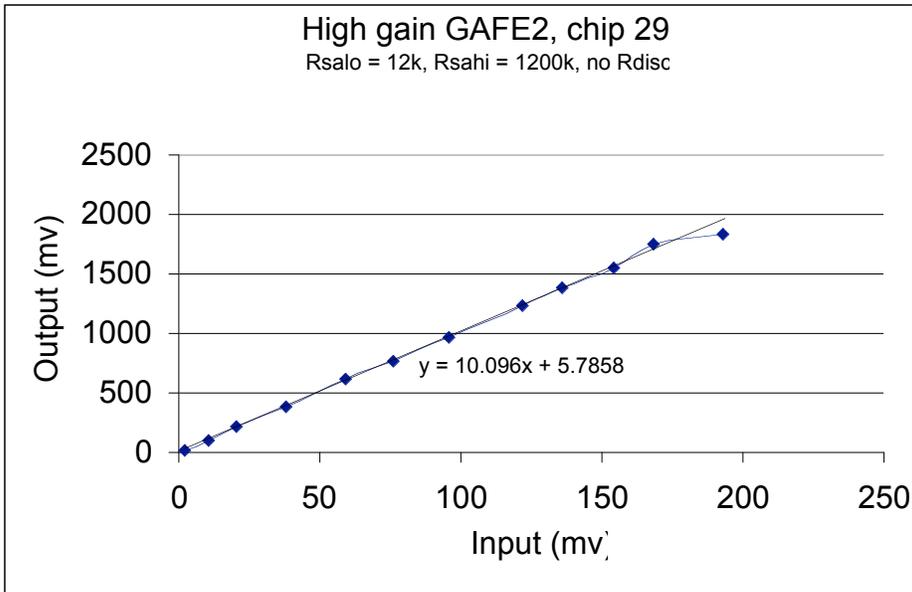
Input (mv) SA Out(mv)

24.2	4.8
40.4	9
65.4	14.2
129.5	32.8
210	365
373	683
601	1065
823.7	1396
1040	1554
1130	1596





Input - mv	SA Out : mv
1.91	19.6
10.4	108
20.7	217
37.8	387.5
59	611
76	770
95.6	973
121.7	1236
135.8	1381
154	1552
168.5	1758
193	1837



4.3 Charge Splitting:

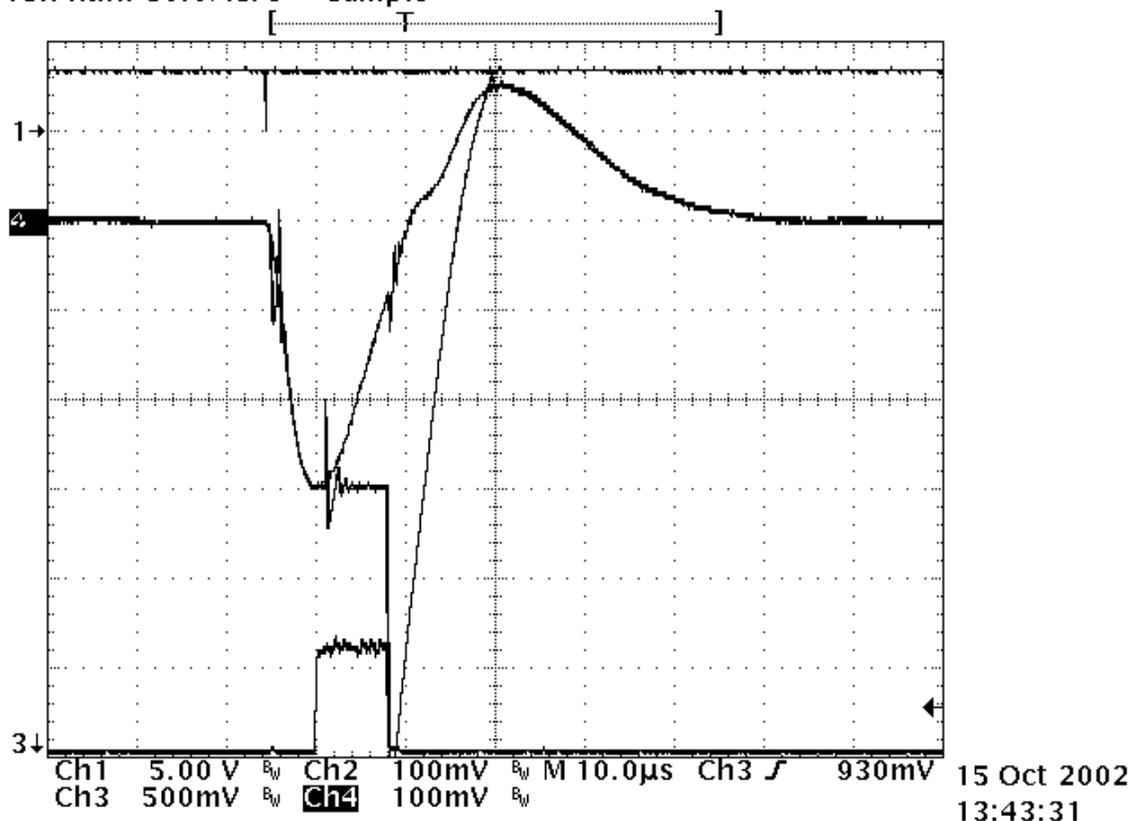
The charge splitting between two PHA channels was measured to be approximately 1:30 where as on GAFE1 it was approximately 1:90. This drop in charge ratio was attributed to the imperfect buffers that caused the non linearity and cross coupling between two channels.

To further investigate this, the 1.2Meg to sahi was connected to ground via a 0.1 uf capacitor while the 12k to salo input was left intact. Next the 12k was returned to ground via a 0.1uf cap and 1.2Meg connected to sahi input. The ratio of the two gains was measured to be approximately, 1:90. This matched with those of the simulations and the results of GAFE1. This also indicates that when the cross coupling is broken by returning one of the resistance to the ground via a capacitance, the charge splitting performs as expected. The remedy for this is to use low impedance buffers and also use separate buffers for the high and low energy channels so as to prevent any cross coupling between them.

4.4 Track and Hold Testing:

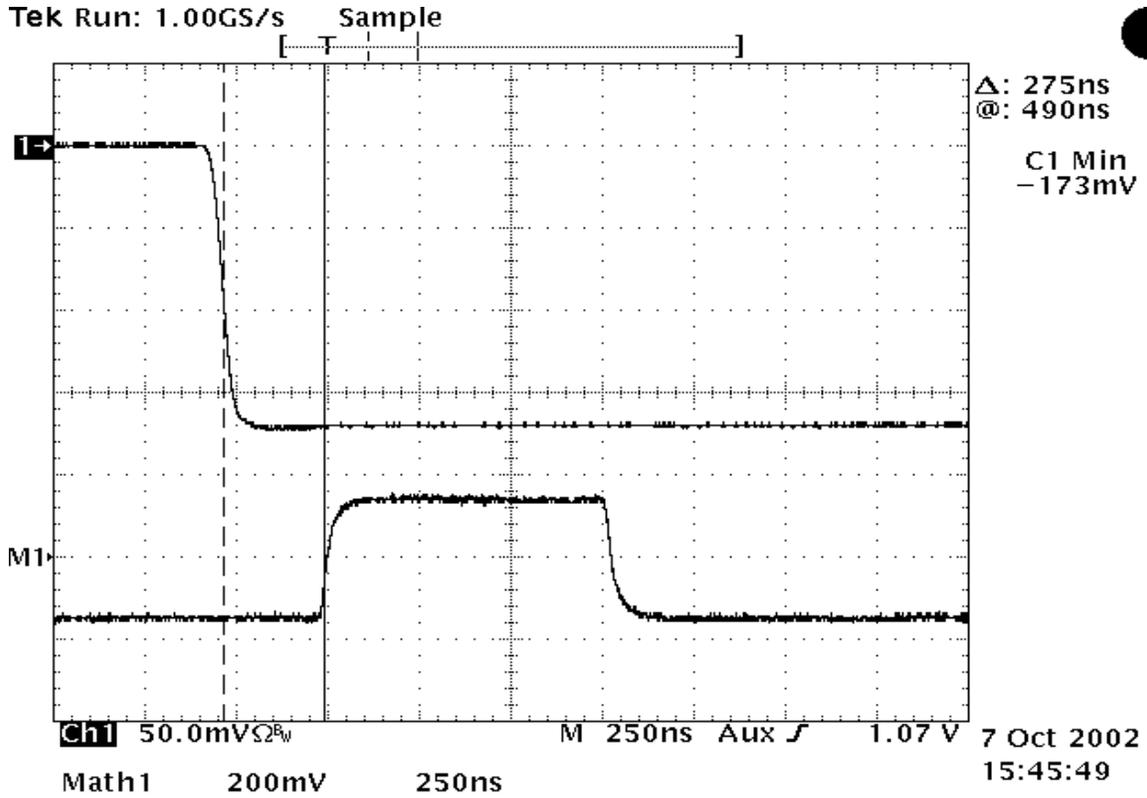
The track and hold output was found to be performing as expected

Tek Run: 50.0MS/s Sample



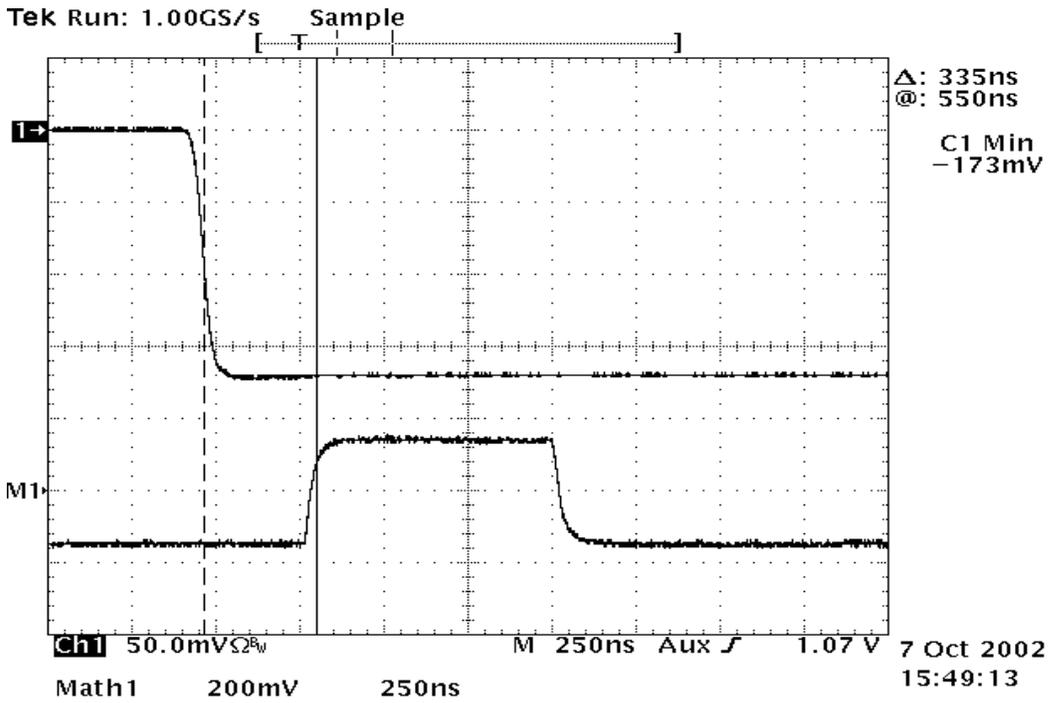
4.5 VETO Generation:

The veto discriminator was found to be working. The Veto delays were observed to be around 350 ns and varied with the value of resistance at the input, discin.



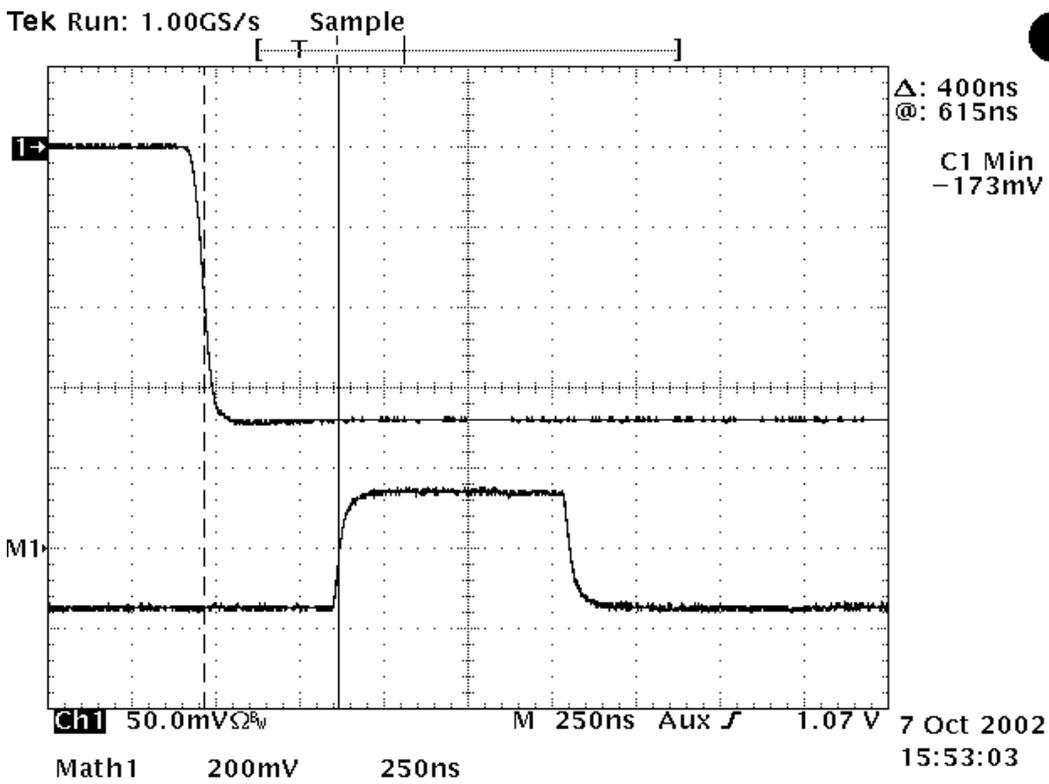
Rdisc = 3.3k ohms, Veto Delay = 250ns

Vin ~ 1mip, Threshold ~ 0.3mip, Chip #9 – GAFE2



Rdisc = 13k ohms, Veto Delay = 335ns

Vin ~ 1mip, Threshold ~ 0.3mip, Chip #9 – GAFE2



Rdisc = 21k ohms, Veto Delay = 400ns

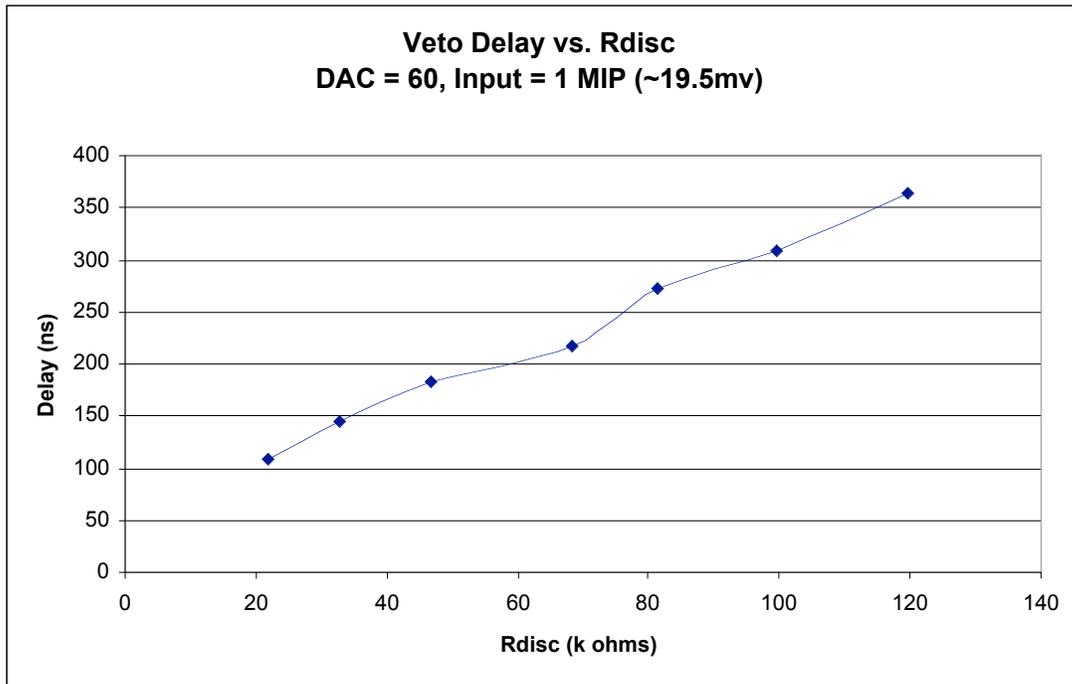
Vin ~ 1mip, Threshold ~ 0.3mip, Chip #9 – GAFE2

GAFE2 test board 1, Rsalo = 11.95k, Rsahi = 1357k
 Veto delays versus Rdisc

Rdisc (kohm)	DAC	~ Vin MIP	Vin (mv)	Tvdlay (ns)
119.7	62	0.28	5.4	800
119.7		0.49	9.6	455
119.7		1.00	19.4	236
119.7	60	0.70	13.6	855
119.7		1.00	19.4	490
119.7		1.99	38.6	225
119.7		5.04	97.7	109
119.7	50	0.00		
119.7		0.00		
119.7		2.84	55.1	563
119.7		5.04	97.7	290
119.7		0.00		
119.7		0.00		
119.7	40	5.05	97.9	625
99.5	62	0.25	4.86	927
99.5		0.49	9.6	436
99.5		1.01	19.5	218
99.5	60	0.62	12	945
99.5		1.01	19.5	364
99.5		0.00		
99.5	50	2.51	48.7	564
81.5	62	0.22	4.2	873
81.5		0.49	9.6	364
81.5		1.01	19.5	181
81.5	60	0.62	12	709
81.5		1.01	19.5	309
81.5		1.97	38.3	182
81.5	50	2.24	43.5	527
68.4	62	0.22	4.2	800
68.4		0.49	9.6	327
68.4		1.01	19.5	164
68.4	60	0.56	10.8	691
68.4		1.01	19.5	273
68.4		1.97	38.3	109
68.4	50	1.97	38.3	582
46.9	62	0.18	3.48	818
46.9		0.50	9.76	255
46.9		1.01	19.6	127
46.9	60	0.50	9.7	636
46.9		1.01	19.6	218
46.9		1.98	38.4	145
46.9	50	1.79	34.7	436
32.9	62	0.18	3.4	691
32.9		0.50	9.7	182
32.9		1.01	19.6	109
32.9	60	0.45	8.68	582
32.9		1.01	19.6	182

32.9		1.98	38.4	109
32.9	50	1.58	30.6	400
21.8	62	0.16	3.2	655
21.8		0.51	9.8	145
21.8		1.02	19.7	91
21.8	60	0.45	8.8	327
21.8		1.02	19.7	145
21.8		1.98	38.5	91
21.8	50	1.42	27.6	345
21.8		1.98	38.5	236
11.94	62	0.16	3.2	509
11.94		0.51	9.8	109
11.94		1.01	19.6	89
11.94	60	0.40	7.8	273
11.94		1.01	19.6	109
11.94		1.98	38.5	87
11.94	50	1.26	24.5	309
11.94		1.98	38.5	182

Rdisc	DAC	Input : MIPs	Input: mv	Delay: ns
119.7	60	1.00	19.4	490
99.5	60	1.01	19.5	364
81.5	60	1.01	19.5	309
68.4	60	1.01	19.5	273
46.9	60	1.01	19.6	218
32.9	60	1.01	19.6	182
21.8	60	1.02	19.7	145
11.94	60	1.01	19.6	109



4.6 Noise:

The noise at the SA output was measured using a digital oscilloscope, the mean value of the noise was measured to be 2.65 mV. Since the SA has an approximate voltage gain of 7, the noise referred to the input is $2.65/7 = 0.38$ mV. Since 20mV correspond to 1MIP at the input, the above voltage noise is equivalent to $0.38/20 = 0.02$ MIPs, where as the requirement is for 0.1 MIP.

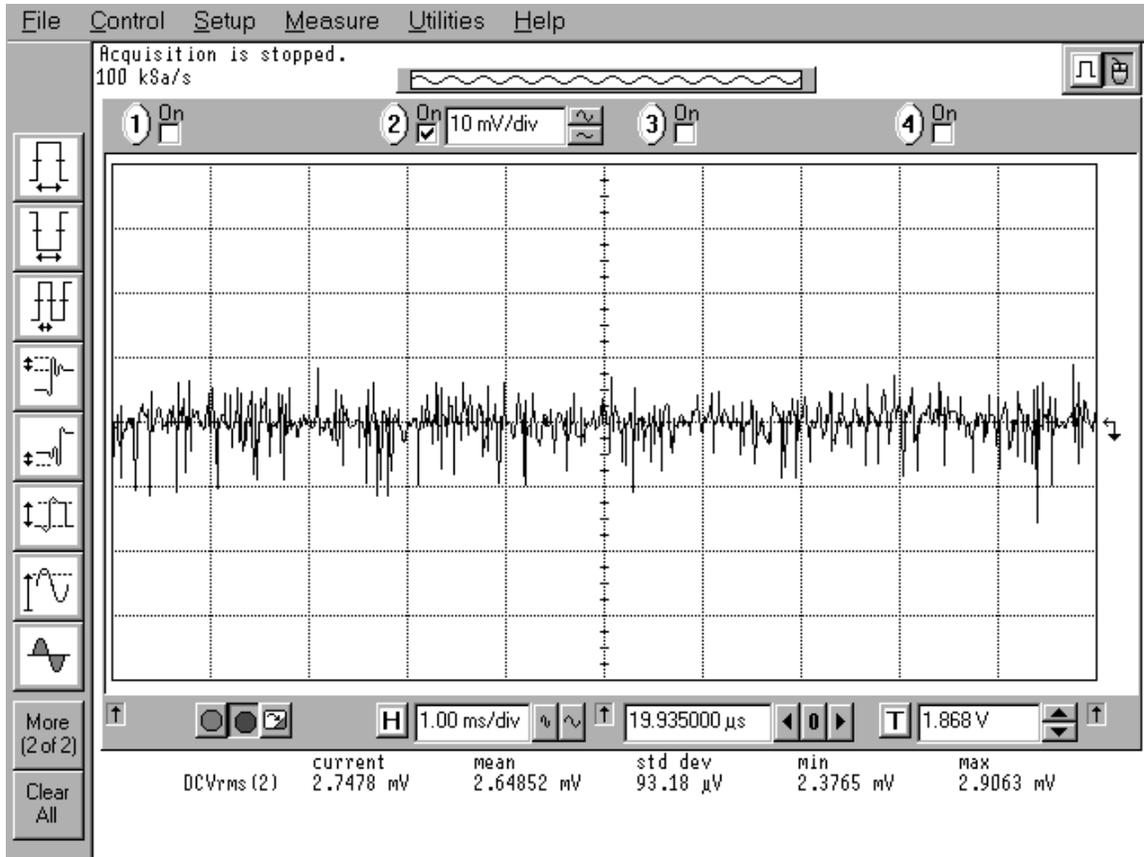
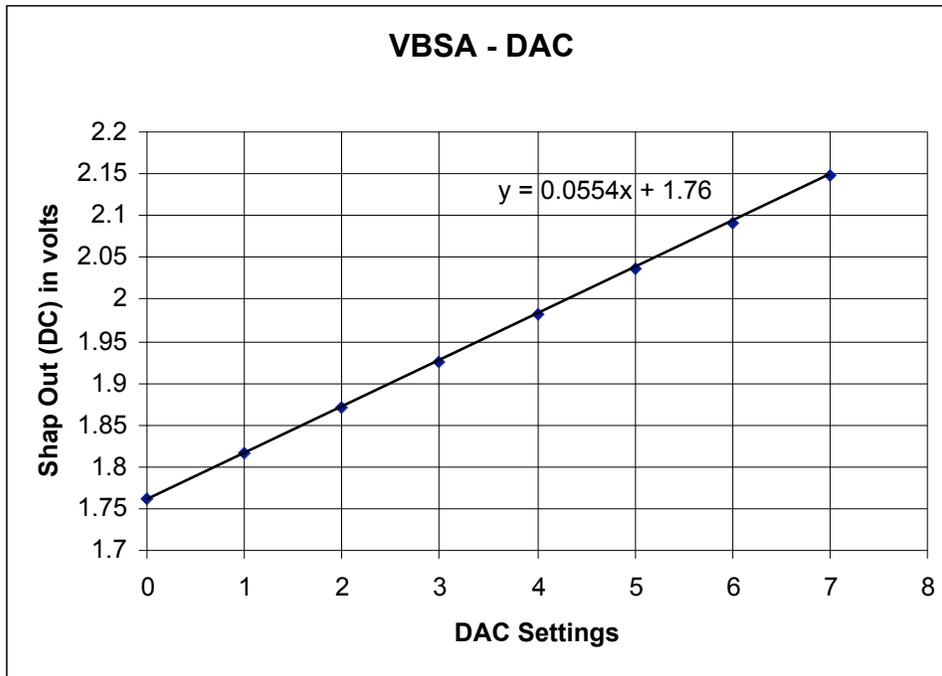


Fig: Noise measurement

4.7 VBSA DAC:

This is a 3 bit DAC that sets the base line of the shaping amps. Below are the measurements of DAC setting versus the DC value observed at the shaping amp output.

bias dac	sa baseline
0	1.761
1	1.816
2	1.87
3	1.925
4	1.981
5	2.037
6	2.092
7	2.149



4.8 **LLD/HLD DAC Operation:**

The DACs seemed to be operating correctly. However it was found that LLD and HLD DAC were swapped. It was also observed that as the LLD DAC was changed, the Peaking time changed. This was attributed to the faulty amp/buffer in the HLD DAC output which resulted in shift in VDC bias and hence the performance of the circuit.

4.9 **VETO Discriminator Offset and sensitivity:**

The Veto offsets and the low level signal sensitivity were measured for about 30 GAFE2 chips. The offsets were computed by measuring "VDC" at the ASIC side of 12k resistor going to salo input. A variable DC voltage source, "Vin" was applied at the common node of 33pf, 12k, 1.2M and 12 K to discin. The difference between the two is shown as the dc offset shown under the column, "Offset" in the table below. For these measurements, the VETO DAC was set to 63, that is for the smallest signal detection.

In addition, the sensitivity was measured by injecting charge through a 3.7pf cap to the common node.

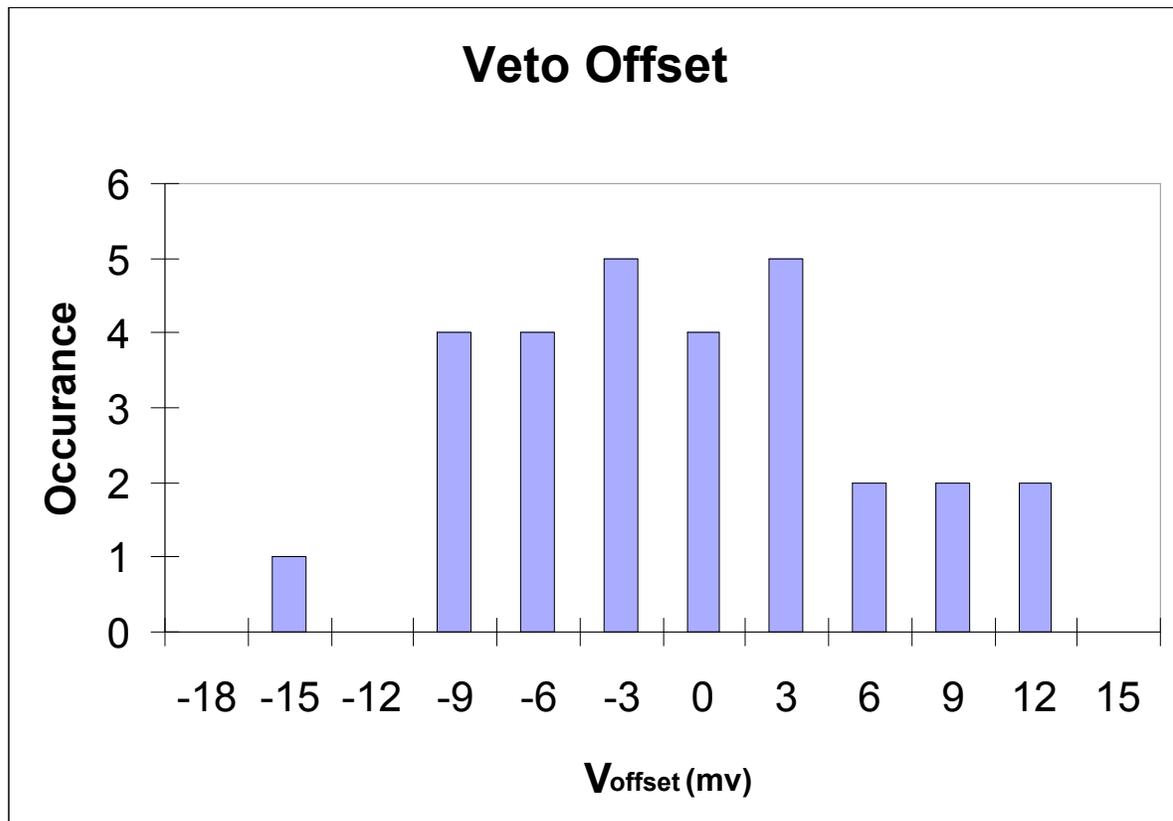
For those chips that had the VETO disc off at DAC=63, the signal was raised till the Veto disc fired. The DAC was then changed to DAC=62 and the signal raised further to fire the veto again, the difference between the two is shown in the column for DAC sensitivity. The measurements under Vmin and Vmin+1 are the signal amplitude applied

Chip #	Vdc (input open)	Vdc	Vin (DAC=63)	Offset(mv) (Vdc-Vin)	State @DAC=63	DACmin	Vmin	DACmin+1	Vmin @DACmin+1	Sensitivity (mv)	
1			1.7378	1.7373	0.5	off	63	35.2	62	62	26.8
2	1.7324		1.733	1.7367	-3.7	on	56	40.4	55	66	25.6
3			1.7458	1.7347	11.1	off	63	340	62	356	16
5			1.7294	1.7292	0.2	off	63	21.6	62	51.2	29.6
6			1.7453	1.7432	2.1	off	63	94.4	62	113.6	19.2
7			1.725	1.7239	1.1	off	63	54	62	79.2	25.2
8			1.7297	1.7378	-8.1	on	52	12.6	51	45.6	33
9			1.735	1.737	-2	on	59	15.8	58	45.2	29.4
10			1.7365	1.7389	-2.4	on	58	38.4	57	66	27.6
11			1.7409	1.736	4.9	off	63	428	62	468	40
12			1.7208	1.7263	-5.5	on	53	36.8	52	60	23.2
13			1.7396	1.7328	6.8	off	63	232	62	248	16
14			1.723	1.7246	-1.6	on	60	28.4	59	52.8	24.4
15			1.7291	1.7452	-16.1	on	34	16.6	33	44	27.4
16	1.7252		1.7269	1.7372	-10.3	on	46	14.2	45	39.6	25.4
17	1.7386		1.7397	1.7453	-5.6	on	53	27.2	52	52	24.8
18	1.7371		1.7397	1.7471	-7.4	on	50	41.2	49	74.4	33.2
19	1.7408		1.7428	1.7517	-8.9	on	48	27.2	47	50.4	23.2
20	1.7221		1.7294	1.7404	-11	on	44	84.8	43	148	63.2
21	1.7267		1.7282	1.735	-6.8	on	51	35.2	50	60.4	25.2
22			1.7481	1.7413	6.8	off	63	222	62	240	18
23	1.7352		1.7355	1.7374	-1.9	on	59	41.6	58	64.4	22.8
24	1.7433		1.7418	1.7325	9.3	off	63	290	62	306	16
25	1.7444		1.7438	1.74	3.8	off	63	140	62	156	16
26	1.7352		1.736	1.741	-5	on	54	36.4	53	59.2	22.8
27	1.7335		1.7332	1.7318	1.4	off	63	66	62	91.2	25.2
28	1.7352		1.7361	1.74	-3.9	on	56	30.6	55	53.6	23
29	1.7372		1.7382	1.748	-9.8	on	55	26.8	54	54	27.2
30	1.7247		1.7275	1.7392	-11.7	on	43	36	42	57.2	21.2

to the 3.7pf charge injection capacitor that fired the VETO discriminator. However, these values have relatively large error or spread in them as there is a relatively large spread of input signal over which the Veto just begins to fire to the point where it fires reliably. Initially, the Veto firings are very narrow as the signal is over the threshold for a short time only and for discriminator widths less than 100ns, the GARC may not recognize them as valid Veto pulses. Therefore one should look at the mean value of the sensitivity to get a feel of the sensitivity of the Veto discriminator, for the measurements, the mean was calculated as 25.88 mv.

For those chips that had Veto disc "ON" for DAC=63 and no signal input, the DAC was raised (lower number since the signal is inverted) till the DAC turned off. Then the signal was applied and raised till it turned on again. Next the DAC was raised (lower setting) by one step and the input signal raised again till veto fired.

The histogram of the DC offset is plotted below. It should be noted that since the VDC was measured at the salo input, the measured value of VDC is therefore different from the true value of VDC by the offset of the input opamp of the high gain PHA channel. Thus the measured VETO offset plotted below and shown in the table below, are actually the sum of the offsets of the input opamp of the PHA channel and the VETO discriminator channel offset.



4.10 VETO - DAC sensitivity:

The veto DAC sensitivity in GAFE2 was found to be 0.07 to 0.12MIP per DAC step compared to the design value of 0.05 MIP per step. This is probably due to the feedback resulting due to non ideal buffers setting the bias point, VDC.

The Veto discriminator sensitivity on GAFE1 was measured to be 0.065 MIP per 0.05MIP increase in threshold.

GAFE2 : Chip #14

GAFE2 Test board: R18/R19/R20/C10: 12K/1.2M/12K/15pF.

Charge injected through a 3.7 pF input injection cap, 173 mV/MIP,

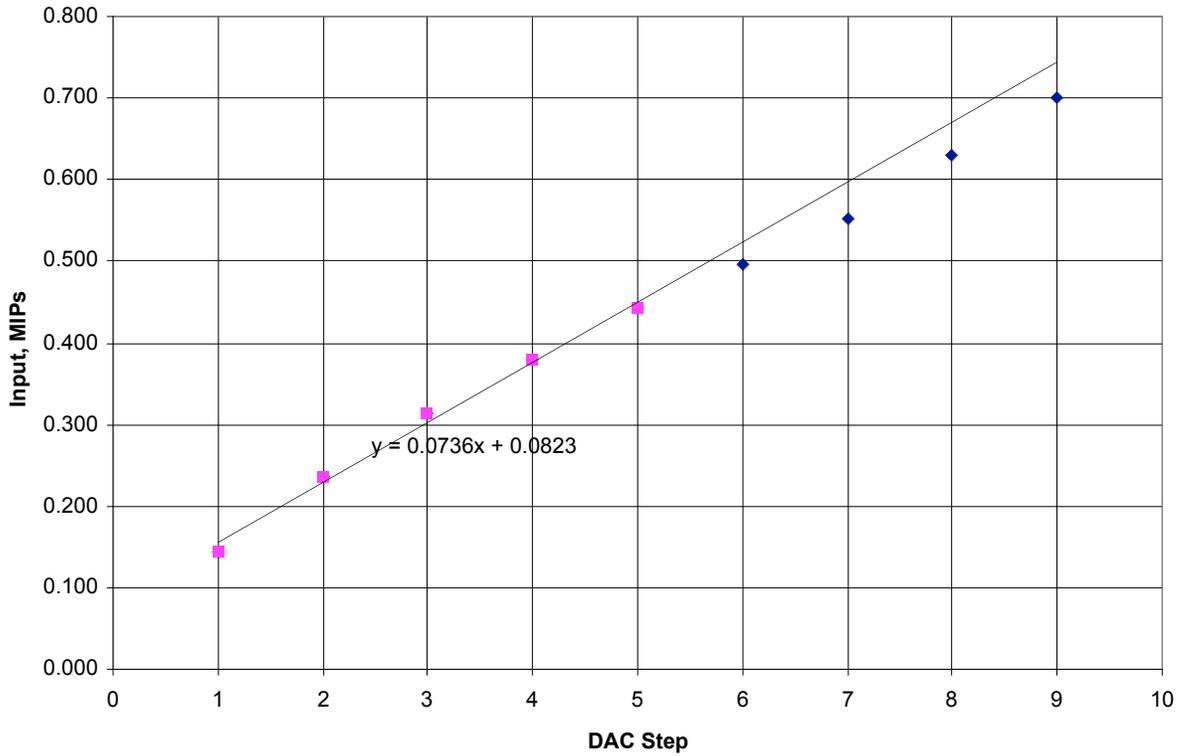
In the measurements below, the "input signal for mostly Vetos" means when Veto is firing for most of the time.

dac setting	input sig for mostly VETOS mV	DAC setting from baseline	input sig for mostly VETOS MIPs
60	25	1	0.145
59	41	2	0.237
58	54.2	3	0.313
57	65.5	4	0.379
56	76.4	5	0.442
55	85.8	6	0.496
54	95.4	7	0.551
53	109	8	0.630
52	121	9	0.699

Chip #14 has small discriminator offset (DAC=60).

Sensitivity is 0.074 MIPs/DAC step

GAFE2 Threshold, Mostly VETOS



GAFE2 :Chip #22

GAFE2 Test board: R18/R19/R20/C10: 12K/1.2M/12K/15pF.

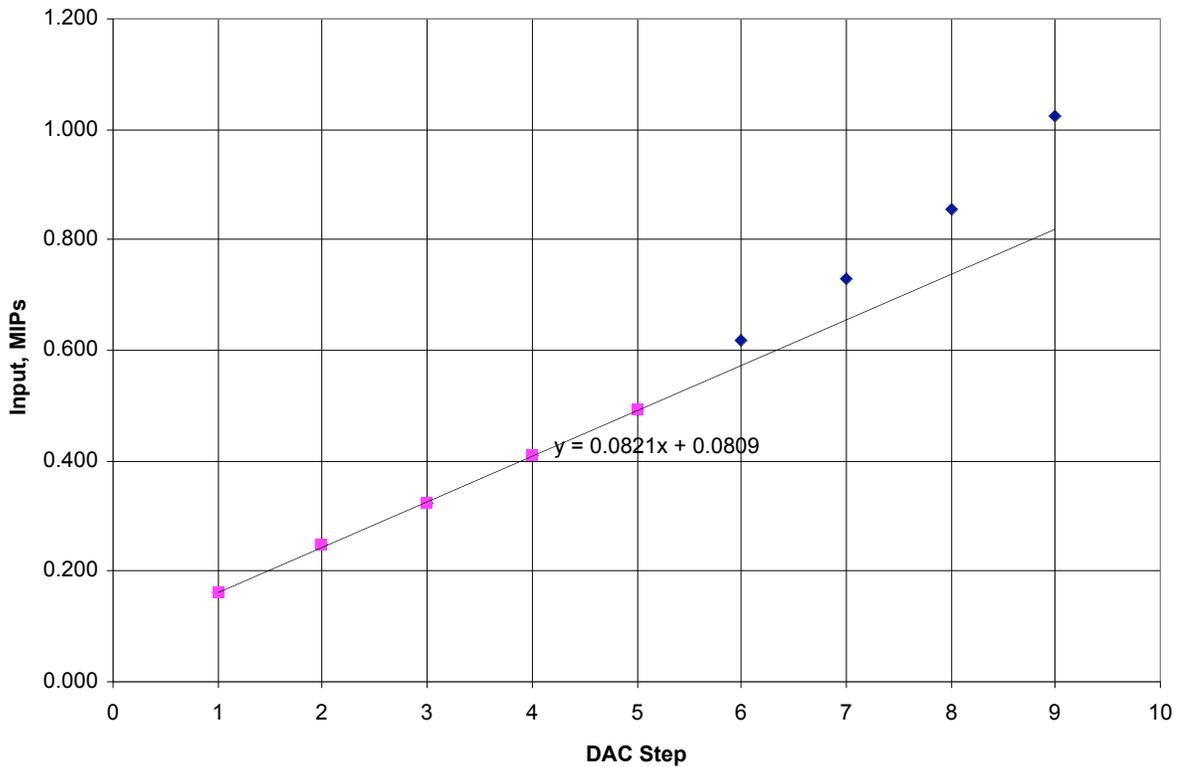
The 4.7pF plus 8pF scope probe replaced with a 15pF cap, no scope probe.

Used 3.7 pF input injection cap, 173 mV/MIP, and HP 8116A function generator.

DAC setting from baseline	input sig for mostly VETOS MIPs
1	0.162
2	0.249
3	0.324
4	0.410
5	0.491
6	0.618
7	0.728
8	0.855
9	1.023

Get 0.08 MIPs per DAC step.
 Discriminator offset shift of 3 steps
 (Possibly due to 10 Meg Ohm impedance of probe)

GAFE2 Threshold, Mostly VETOS

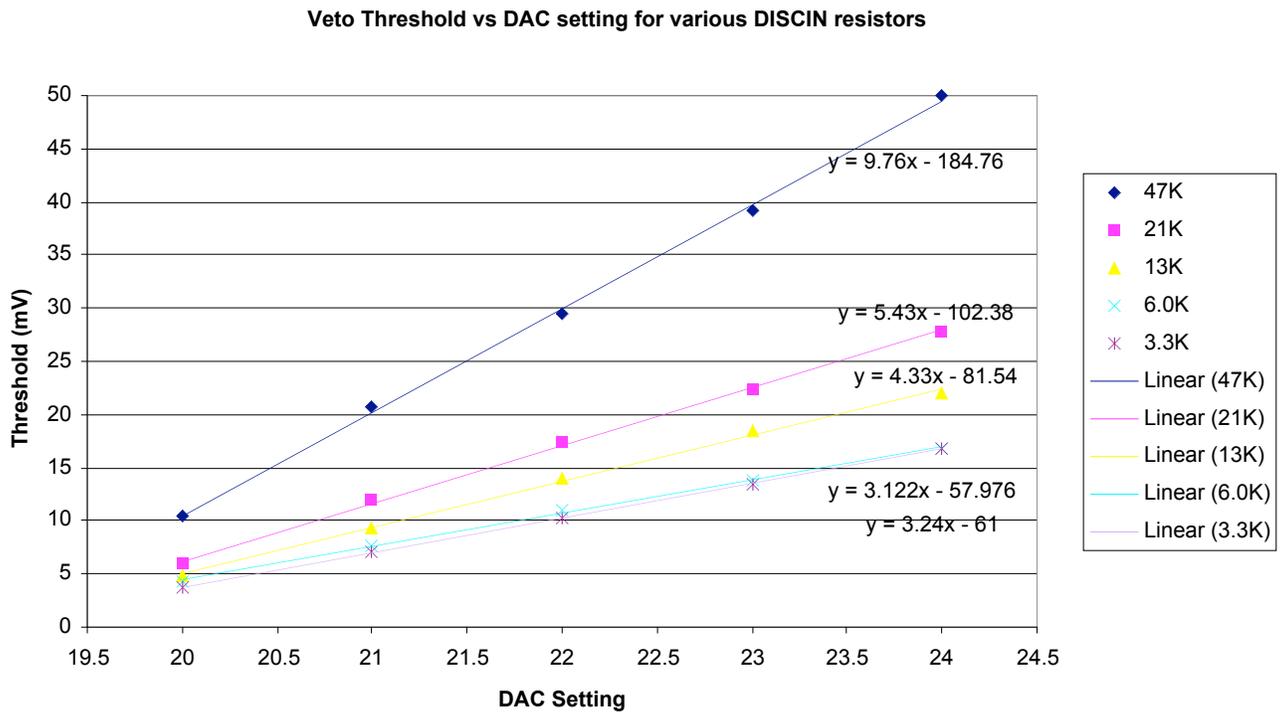


4.11 VETO Sensitivity to Resistor input at discin:

The variation of VETO sensitivity to the value of the resistor input to "discin" was also investigated. For this test a tail pulser was used with the leading edge time constant of 0.02 us and the decay time constant of 0.5us. The charge splitting resistors used for this test were, 12k to salo, and 1.2 Meg to sahi. In the measurements below, the DAC of value 1 means that it is 1 step below the base line, so the actual value programmed into the DAC is 63. Likewise, for DAC value of 2, the actual value programmed into the DAC would be 62, and so on.

In the table below, the measurements are for input signal level that reliably fires the discriminator. "Ideal" is based on DAC step size of 15.87 mV (calculated) and gain of 20 for Veto signal.

DAC	47K	21K	13K	6.0K	3.3K	1.07K	0.29K	Ideal
20	10.4	5.9	4.9	4.24	3.8	3.4	3.8	2.87
21	20.8	11.9	9.3	7.7	7.1	6.7	7.3	3.66
22	29.4	17.4	14	11	10.2	11.5	12.8	4.46
23	39.2	22.4	18.4	13.8	13.5	16.4	19.4	5.25
24	50	27.8	22	16.8	16.8	21.8	24.6	6.04



4.10 Mux and Channel Select

The multiplexers and channel select were found to be working as the low and high energy channels could be selected by applying the right digital level at the channel select pin.

DIGITAL SECTION **(Contents of this section provided by Dave Sheppard)**

Description of the GAFE ASIC Logic:

GAFE is the acronym for the Gamma-Ray Large Area Space Telescope (GLAST) Anti-Coincidence Detector (ACD) Front-End Application Specific Integrated Circuit (ASIC). The GAFE logic interfaces to the ACD Readout Controller ASIC (GARC), which is the

main logical interface for the ACD to the LAT instrument electronics. GARC provides the GAFE ASICs with commands and processes digital data returned from the logic core.

This document describes the testing of the GAFE ASIC logic. The logic is expected to meet the requirements of the ACD Level IV Electronics Requirements, LAT-SS-00352, and the ACD-LAT Interface Control Document, LAT-SS-00-363.

The design was done utilizing Verilog as the description language, Exemplar Leonardo Spectrum as the synthesis tool targeting the Tanner Agilent 0.5 μ m standard cell library, and Tanner L-Edit as the automated place and route layout tool. Core verification was performed via the Tanner LVS tool.

During the design of the GAFE logic, it was assumed that the Tanner tristate buffer signal OEN was active low. This proved to be incorrect. Three GAFE V2 ASICs were modified after fabrication to fix this error. Tests performed on the GAFE logic were run with the modified ASICs. This signal polarity error was corrected in the GAFE V3 core logic.

GAFE Logic Commands:

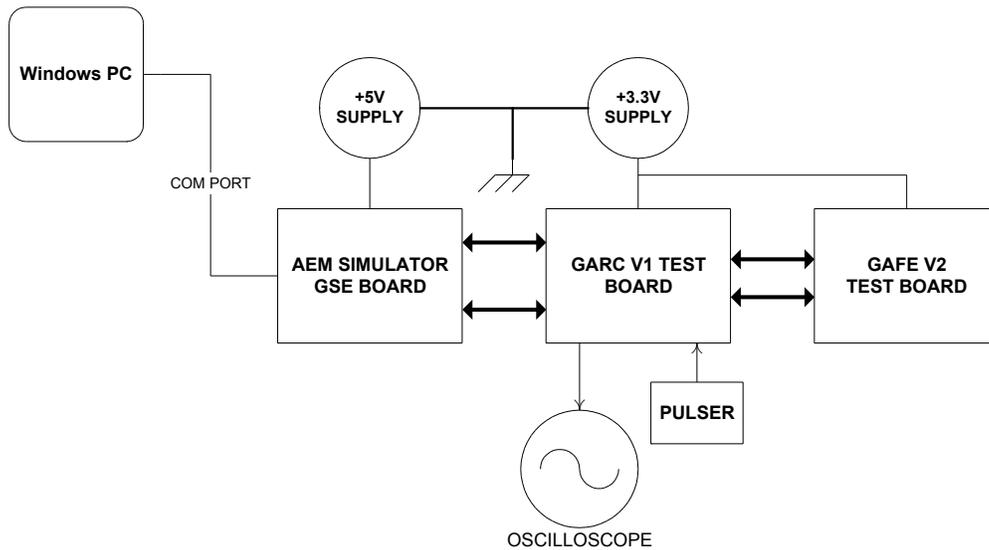
The following is a list of commands utilized by the GAFE logic.

GAFE Cmd No.	ACD Command Mnemonic	Rd/Wr Status	Select GARC=0 GAFE=1	Function Block	Register Number	No. of Data Bits	Command Function
1	GAFE_Mode_Wr	W	1	GAFE Addr	0	16	Writes the GAFE mode register for the ASIC addressed
2	GAFE_Mode_Rd	R	1	GAFE Addr	0	16	Reads the GAFE mode register contents for the ASIC addressed
3	GAFE_DAC1_Wr	W	1	GAFE Addr	1	6	Writes the DAC1 register in the GAFE addressed
4	GAFE_DAC1_Rd	R	1	GAFE Addr	1	6	Reads back the contents of the DAC1 register in the addressed GAFE
5	GAFE_DAC2_Wr	W	1	GAFE Addr	2	6	Writes the DAC2 register in the GAFE addressed
6	GAFE_DAC2_Rd	R	1	GAFE Addr	2	6	Reads back the contents of the DAC2 register in the addressed GAFE
7	GAFE_DAC3_Wr	W	1	GAFE Addr	3	6	Writes the DAC3 register in the GAFE addressed
8	GAFE_DAC3_Rd	R	1	GAFE Addr	3	6	Reads back the contents of the DAC3 register in the addressed GAFE
9	GAFE_DAC4_Wr	W	1	GAFE Addr	4	6	Writes the DAC4 register in the GAFE addressed
10	GAFE_DAC4_Rd	R	1	GAFE Addr	4	6	Reads back the contents of the DAC4 register in the addressed GAFE
11	GAFE_DAC5_Wr	W	1	GAFE Addr	5	6	Writes the DAC5 register in the GAFE addressed
12	GAFE_DAC5_Rd	R	1	GAFE Addr	5	6	Reads back the contents of the DAC5 register in the addressed GAFE

13	GAFE_Version	R	1	GAFE Addr	6	3	Reads back the GAFE ASIC version
14	GAFE_Write_Ctr	R	1	GAFE Addr	7	8	Reads back the contents of the GAFE write counter register
15	GAFE_Reject_Ctr	R	1	GAFE Addr	8	8	Reads back the contents of the GAFE command reject register
16	GAFE_Cmd_Ctr	R	1	GAFE Addr	9	8	Reads back the contents of the GAFE command counter
17	GAFE_Chip_Addr	R	1	GAFE Addr	10	5	Reads back the hardwired address of a GAFE ASIC

Testing the GAFE Digital Logic

The GAFE logic is based on a command-response protocol and requires a GARC or GARC simulator to access the logic functions. For each of the following commands, the proper GAFE logic response was tested. The test setup used is detailed in the diagram below. A GARC V1 ASIC mounted in the GARC test board was used as the GAFE interface.



GAFE V2 ASIC FUNCTIONAL TEST SETUP

Initial Reset Test

This section will verify that GAFE registers were properly initialized during a reset. This test was successful. Using Bob Baker's AEM3.exe program as the command and readback software, the following register contents were readback directly after reset. The values are listed in the table below.

GAFE #2 Register	Readback (dec)	Readback (hex)	Matches ICD?
0	48	30	Yes
1	57	39	Yes
2	38	26	Yes
3	55	37	Yes

4	32	20	Yes
5	0	0	Yes
6	2	2	Yes
7	0	0	Yes
8	0	0	Yes
9	9	9	Yes
10	0	0	Yes

Register Read/Write Test

All the registers in the GAFE logic were tested for proper read and write response. Each register test was successful. The following register write and readback commands (taken from the as-run test) were successfully executed:

1. Send the GAFE_Mode_Write command to the GAFE being tested with a data field of 16'h00 (0). Send the GAFE_Mode_Read command. The data field in the return data stream should be 16'h00 (0).
2. Send the GAFE_Mode_Write command to the GAFE being tested with a data field of 16'hFF (255). Send the GAFE_Mode_Read command. The data field in the return data stream should be 16'hFF (255).
3. Send the GAFE_Mode_Write command to the GAFE being tested with a data field of 16'h30 (48). Send the GAFE_Mode_Read command. The data field in the return data stream should be 16'h30 (48).
4. Send the GAFE_DAC1_Write command to the GAFE being tested with a data field of 16'h00 (0). Send the GAFE_DAC1_Read command. The data field in the return data stream should be 16'h00 (0).
5. Send the GAFE_DAC1_Write command to the GAFE being tested with a data field of 16'h3F (63). Send the GAFE_DAC1_Read command. The data field in the return data stream should be 16'h3F (63).
6. Send the GAFE_DAC1_Write command to the GAFE being tested with a data field of 16'h39 (57). Send the GAFE_DAC1_Read command. The data field in the return data stream should be 16'h39 (57).
7. Send the GAFE_DAC2_Write command to the GAFE being tested with a data field of 16'h00 (0). Send the GAFE_DAC2_Read command. The data field in the return data stream should be 16'h00 (0).
8. Send the GAFE_DAC2_Write command to the GAFE being tested with a data field of 16'h3F (63). Send the GAFE_DAC2_Read command. The data field in the return data stream should be 16'h3F (63).
9. Send the GAFE_DAC2_Write command to the GAFE being tested with a data field of 16'h26 (38). Send the GAFE_DAC2_Read command. The data field in the return data stream should be 16'h26 (38).
10. Send the GAFE_DAC3_Write command to the GAFE being tested with a data field of 16'h00 (0). Send the GAFE_DAC3_Read command. The data field in the return data stream should be 16'h00 (0).

11. Send the GAFE_DAC3_Write command to the GAFE being tested with a data field of 16'h3F (63). Send the GAFE_DAC3_Read command. The data field in the return data stream should be 16'h3F (63).
12. Send the GAFE_DAC3_Write command to the GAFE being tested with a data field of 16'h37 (55). Send the GAFE_DAC3_Read command. The data field in the return data stream should be 16'h37 (55).
13. Send the GAFE_DAC4_Write command to the GAFE being tested with a data field of 16'h00 (0). Send the GAFE_DAC4_Read command. The data field in the return data stream should be 16'h00 (0).
14. Send the GAFE_DAC4_Write command to the GAFE being tested with a data field of 16'h3F (63). Send the GAFE_DAC4_Read command. The data field in the return data stream should be 16'h3F (63).
15. Send the GAFE_DAC4_Write command to the GAFE being tested with a data field of 16'h20 (32). Send the GAFE_DAC4_Read command. The data field in the return data stream should be 16'h20 (32).
16. Send the GAFE_DAC5_Write command to the GAFE being tested with a data field of 16'h00 (0). Send the GAFE_DAC5_Read command. The data field in the return data stream should be 16'h00 (0).
17. Send the GAFE_DAC5_Write command to the GAFE being tested with a data field of 16'h3F (63). Send the GAFE_DAC5_Read command. The data field in the return data stream should be 16'h3F (63).
18. Send the GAFE_DAC5_Write command to the GAFE being tested with a data field of 16'h00 (0). Send the GAFE_DAC5_Read command. The data field in the return data stream should be 16'h00 (0).

Testing the Addressing and GAFE Logic Read-Only Registers:

The GAFE version command correctly returns 2.

The GAFE write counter correct returns the number of write commands.

The GAFE address command correctly returns the hard-wired chip address.

In order to test register 9 (the command counter), we kept sending reads for readback register. The counter increments and rolls over at 63 → 0 as designed.

The chip was tested with address 0 and address 4, which functioned as designed. The ASIC was also tested for response to broadcast commands (i.e., chip address 'h1F). Broadcast writes function as designed.

Read commands to the broadcast address are intended to be ignored. The logic is not correct for this function – it responds with return data for reads to address 'h1F. This is a design error. The GAFE logic should be designed to reject readback commands to address 'h1F. This is not a valid command that should ever be sent by the AEM. The implication of sending this command to a set of GAFEs connected together would be that more than one ASIC would try to drive the RTND line simultaneously, which is an undesirable condition. This has been fixed in the logic core for GAFE V3.

GAFE Logic Clock-to-Data Delay:

On the GARC simulator board, we measured the delay from the positive edge of the GAFE clock out to the positive edge of the GAFE data received to be approximately 30 ns. This is satisfactory since the GAFE clock runs at a nominal 5 MHz.

GAFE Current Measurements:

The current at the nominal power supply voltage was measured via the GAFE test board for each of the ASICs tested.

ASIC	Analog 3.3V Current	Digital 3.3V Current	ASIC Power Dissipation
Modified GAFE #1	0.55 mA	1.13 mA	5.54 mW
Modified GAFE #2	0.56 mA	1.12 mA	5.54 mW
Modified GAFE #3	0.55 mA	1.11 mA	5.48 mW

Summary:

The GAFE core logic functions as designed with the exception of the following two items:

- (1) The polarity of the tristate buffer OEN on the bi-directional pad was incorrect and verified by this measurement. This has been updated for the next version of GAFE.
- (2) An error in the processing of readback commands at chip address to 0x1F was discovered. Although this is an invalid command, we have redone the logic core to eliminate this condition.

Problems:

1. Buffer related Problems:

It was determined that the buffer cell used in GAFE2 is inadequate. This buffer has been repeatedly used in a number of places in the ASIC, for example at the DAC outputs that drive the bias points, the multiplexed analog output and the track hold outputs of the two channels. The problems arising due to the use of this deficient buffer are:

1.1 VBSA Buffer:

The DAC buffer providing the VBSA bias is inadequate in that it does not have low enough impedance which results in feedback leading to non linearity

and saturation of signals, and also to cross coupling between low and high energy channels. This buffer needs to be strengthened, and independent buffers used for low and high gain channels to avoid cross coupling.

1.2 **DAC Buffer for VDC:**

This buffer also needs to be beefed up as this also contributes to non linearity in the discriminator channel. Also since this buffer is weak, the programming of LLD DAC (actually the HLD on the chip since the LLD and HLD DACs are swapped), leads to increase current, which the present VDC buffer can not handle which leads to shift in the VDC bias point and also the performance of the PHA channel.

The deficiency in above two buffers results in the following performance deficiencies:

- a. The analog channels are non linear
- b. There is cross coupling between low and high energy channels which leads to lower than expected ratio of charge splitting between the two channels.
- c. Shift in peaking time as the LLD DAC (actually HLD DAC since LLD and HLD DACs are swapped)
- d. Lowering of the gain in the discriminator and PHA channels.

1.3 **Track Hold Output Buffer to ADC:**

The present buffer is adequate to follow the shaping amp output and drive the 25pf ADC capacitance. However, to switch channels and be able to slew at less than 1us, the drive capability of this buffer needs to be increased. The present buffers in the hold module of each of the PHA channel will slew to 2v within a microsec, but to speed them up, doubling of the output transistors is recommended.

2. LLD and HLD DAC swapped:

The LLD and HLD DACs were found to be swapped.

3. Peaking Time:

The Peaking time is 6us instead of 3us, this is due to the twice the value of Nwell resistors fabricated. This has been replaced by right value Poly resistors in the next version.

4. VETO DAC / Discriminator Offsets:

The discriminator have positive and negative offsets. If the discriminator is on all the time because of negative offset, then the threshold may need to be set negative. Therefore the DAC should be altered to provide negative (with

respect to the baseline) settings also. In the existing design using a resistor chain, the VDC baseline should be tapped at a position few resistor links down the chain. In addition, since the VETO speed requirements are much more relaxed as a few hundred nano sec delays can be tolerated, the transistor sizes could be increased to reduce the offset.

5. Output enable for Return Data:

The polarity of the tristate buffer OEN on the bi-directional pad was incorrect and verified by this measurement. This has been updated for the next version of GAFE.

6. Error in Readback command:

An error in the processing of readback commands at chip address to 0x1F was discovered. Although this is an invalid command, we have redone the logic core to eliminate this condition.