Testing procedures and results for the ELEFANT ASIC

Michael Momayezi

Oct. 2, 1997

Introduction

The purpose of this document is to describe the setup and procedures used in testing the ELEFANT ASIC.

Hardware

The test-board

Besides the 5V analog and digital power supply voltages, the ASIC needs 5 reference voltages. Three are needed for the FADC resistor ladder and two for the analog receivers. 'fet_bias', which adjusts the reset time constant of the analog receivers, must track 'V_ref' which sets the receiver DC-level. While negligible power is demanded from the reference supplies, the voltages need to be variable and it should be possible to set them using switches. They have therefore been implemented as 6-bit DACs using LM317 voltage regulators, DIP switches and a few precision resistors.

The analog and TDC inputs of the ASIC require differential signals. In order to simplify interfacing to a pulse generator, single-ended to differential converters have been put on the board. In order to be able to measure cross talk, there are two independent channels each for the analog and the TDC inputs. Either of the two channels can be connected to any of the eight inputs of the ASIC via a switch yard and jumpers.

The converters are AD8015 transimpedance amplifiers with a rise time of 5ns and 100Ω output impedance. For the analog inputs this is too fast, and the great bandwidth allows too much noise. The signals are slowed down by placing 50pF capacitors across the differential pairs.

The TDC inputs should be driven with differential current drivers. However, having to use transistor arrays and discrete SMD components for the drivers results in a layout that needs quite some area. As a result the lines leading from the driver to the inputs are too long, and their capacitance deteriorates the signal rise time. For an evaluation of the true TDC performance, the pulser signals (1ns rise time, 50Ω output impedance) were lead directly

to the switch yard and from there to the TDC inputs, without intermediate electronics.

The ASIC is supplied with an external system clock, usually running at 60MHz. On the board, a 1ns rise time discriminator (AD9696KR) is used to restore the clock rise time. If it were necessary to delay the clock signal with respect to the command signals, there is a jumper option to use an external delay cable. The cable is driven by a CLC111 transimpedance 50Ω line driver, and the signal is received by a discriminator as above.

In a similar fashion, the sample clock output from the ASIC is fed into a AD9696KR and is then, via a delay cable, routed to the clock input of a logic analyser. This allows to adjust the timing between the data outputs and the sample clock at the logic analyser inputs.

Instrumentation

The following is a list of the instruments used for testing.

Hewlett Packard 16500C mainframe with a 16522 pattern generator card (100MHz) and a 16521 logic analyser card (100MHz state analyser). Fully programmable via GPIB. Pattern generator programs are loaded from the local disk or are generated by the host computer and down-loaded. Data from the logic analyser are transferred to the host computer for analysis.

Hewlett Packard 3245A dual channel universal source for remote control of reference voltages. Programmable via GPIB.

LeCroy 9210 dual pulser mainframes, 2pcs. One equipped with two 9211 modules to generate pulses for the analog and TDC inputs. One equipped with two 9212 modules for clock generation. Frequency, pulse height, shape and delay are programmable via GPIB.

American Reliance PPS-2322 power supplies (2 pcs) with output current reading. Programmable via GPIB.

Apple MacIntosh 7300/180 power mac equipped with National Instruments PCI-1200 I/O board and PCI-GPIB board implementing an IEEE-488.2 interface.

Software

The software has to support the testing in various ways.

- It must provide a convenient and reliable interface to program the test stand instrumentation.
- It must allow to display and analyse data online quickly and with little effort.
- It must allow inclusion of externally written programs to facilitate tasks outside its scope.
- It must provide a graphic push button interface to support production testing of a large number of ASICS by minimally trained personnel.

IGOR

The program package used is IGOR and is sold by Wavemetrics. While mainly a data presentation and analysis package, it does support the GPIB and PCI-1200 cards from National Instruments.

It has a graphical point and click interface, but is at the same time programmable in what amounts to a bare-bones no-frills version of the programming language C. As a boon, all point and click actions are echoed as commands in a window from where they can be grabbed and pasted into a macro. This feature tremendously cuts down the program development time as it is rarely necessary to consult the documentation.

Graphical presentation of the data is luxurious, with a wide variety of options available, and can be automatized like any other other task, including the printing.

A main program displays and receives the parameters necessary to run the tests. The interface consists of a window with boxes for user input. A push button panel has buttons to start tasks, for emergency power off, and the like. During production testing, this is the only manual interaction with the program that is necessary.

It is always possible to call any function or macro from the command line. So, one can run the main program and then manually call for more data analysis, change displays and the like. Thus, developing new tests or enhancing existing procedures can be done from within the main program.

IGOR extensions

While IGOR is fairly simple to program, there are tasks that for reasons of complexity or speed are better programmed in C. IGOR allows to bind in externally written programs and to exchange data with them via command line arguments. On start up, IGOR scans for the presence, but not the contents, of precompiled programs in its extension folder. As a result it is possible to work on an extension, recompile it and run it under IGOR without having to exit IGOR. For the C/C++ compiler we use Metrowerk's Code Warrior, which has worked fine.

Three IGOR extensions have been written in C:

The routine for unpacking the data from the logic analyser was originally written in IGOR but has been moved to C to speed it up.

The digital tests require that programs are developed for the pattern generator. Doing this manually can be fairly time consuming. To speed up this process, a simple command language was developed that allows the user to write the program in a mnemonic code. A procedure, written in C, then converts this human readable code into test vectors. It is a task similar to an assembler that converts mnemonic instructions into machine code.

In a digital test one wants to compare the ASIC response to its stimuli to an expected response. This expectation is created by a simulator that implements a behavioral model of the ASIC. The simulator is an adaption of a behavioral model provided by Michael Levi.

Results

As the digital sections in the preproduction version of the ELEFANT ASIC appear to be operating fine, the major test focus has been on the characterization of the FADC and TDC performance. For 84 ELEFANT ASICs from the preproduction run this has been done in some detail.

The findings are presented in a number of plots that may be found in the same directory as this document. The file README contains a concise description of the plots.

For the FADC and TDC characterization the settings common to all measurements were

- AVDD, DVDD = 5.0V

FADC					
offset		slope		bin width	
mean	rms	mean	rms	mean	rms
41 mV	1.7 mV	224 counts/V	2.5 counts/V	4.3 mV	1.4 mV

TDC					
offset		slope		bin width	
mean	rms	mean	rms	mean	rms
24.0 ns	0.17 ns	1.0158 counts/ns	0.00076 counts/ns	0.98 ns	0.10 ns

Table 1: FADC and TDC performance of more than 650 channels.

- sys_clock frequency = 64.00 MHz
- Gain setting = minimum gain (x11)
- $V_{\mathrm{top}} = 4.00\,\mathrm{V}$, $V_{\mathrm{bot}} = 1.00\,\mathrm{V}$, V_{mid} = notconnected
- About 1000 data points are used to cover the FADC and TDC range.

The results are summarized in table 1. See also the plots *fadc_slope.ps, fadc_zero.ps, tdc_slope.ps, tdc_zero.ps, dnl.ps.*

Crosstalk

Crosstalk between adjacent FADC channels has been investigated. To minimize coupling into the differential lines leading from the on-board amplifiers to the analog receivers on the chip, there are 50 pF capacitors placed across the two traces of each line.

For full-swing signals with a rise time (10% to 90%) of 20 ns the crosstalk to the closest neighbor is less or equal to one FADC count.

Supply voltage range

Operating the chip with the analog and digital supply tied together, the useful range extends from 4.7 V to 5.1 V. See plots *fadc_4V7.ps, fadc_5V1.ps, tdc_4V7.ps, tdc_5V1.ps.*

Below 4.6V the top most FADC-comparator is no longer functional if the top ladder reference voltage is kept at 4.0V. Above 5.2V, data from the memory are corrupted.

Frequency range

TDC and FADC-operation has been tested from 50MHz to 70MHz. While the TDC works flawlessly over this frequency range, the FADC shows a likelihood for erroneous codes that increases with the frequency. At 50MHz it works well, but at 60MHz occasional 2-count errors occur, which at 70MHz increase to 4-count errors. See plot *tdc_frequency.ps, fadc_frequency.ps.*

Irradiation

Two chips have been irradiated using a ^{60}Co γ-source. Within $16^{\mathrm{h}}45^{\mathrm{m}}$ they received a dose of 20krad human tissue equivalent. One of the chips was powered up and was supplied a system clock (64MHz). 'test_mode' and 'chip_select' were pulled high to keep most of the circuitry active. The other chip was not connected to power or any other circuitry.

After the irradiation the chips were tested again, and no difference to their pre-irradiation characteristics could be found.

Offset measurements

There is an offset that appears at the output of the analog receiver, which is caused by the injection of switching spikes into the analog receiver.

The plot *offset_resistor.ps* shows the effect of a resistor in between the analog receiver and the FADC. As one can see, the additional impedance does reduce the offset, but also makes the response nonlinear.

The resistance between the analog receiver and the FADC input was realized by cutting the connecting trace with an ion beam and depositing a new narrow trace of tungsten on the IC. No external components were used.

PLL-voltages

The plot *PLL_hist.ps* shows for 50 chips a histogram of the control voltages V_{ctrl} of the delay-locked loop when the circuit is in lock. At 60MHz the average value is 3.23V and the rms spread is 0.05V

Power consumption

The power consumption has been measured on a number of chips under nominal operating conditions: power supply voltage at 5V, 60MHz, in test mode , and with periodic L1_accepts and subsequent readout of 8 channels. The average supply current is 82mA and the rms spread is 2.6mA. See also plot *power_hs.ps*.

Note that current consumption is no useful indicator for 'bad' chips. Of all rejects only 2 draw excess current, namely 108mA and 165mA

Digital tests

Automatic digital testing has been implemented only recently, but a subset of the tests described in the next section has been done for 84 chips from the preproduction run.

Table 2 gives a summary of all chips that failed an analog or digital test. The FADC and TDC column show which channel failed. The 'buffer' test is there to verify the correct operation of all four buffers. The trigger and hit map tests seek to verify the correctness of response to analog and discriminator pulses.

Yield

According to the tests conducted so far, 13 out of 84 chips from the preproduction run would be rejects. Note that this number may increase slightly, as we have not yet tested these rigorously for correct operation of the δ-bit generation over the whole parameter space.

Production testing

Testing the ELEFANT production run will call for 2,000 chips to be tested. A time budget of 10min per chip translates into 42×8 hours of run time or three months real time. The apparatus can be upgraded to handle 5 chips at the same time, but the cost in time and money for the necessary preparations may well offset the gains.

From experience with the preproduction run, the yield is estiamted to be about 85%. Since 928 good parts are needed, 1100 to 1200 chips will have to

chip	FADC	TDC	buffer	trigger	hit map
57	all	all	all	OK	n/a
63	0	OK	OK	OK	OK
66	2	OK	2	OK	OK
77	OK	1	OK	1	1
78	1	OK	OK	OK	OK
96	OK	OK	OK	OK	all
103	OK	OK	OK	OK	1
105	OK	6	OK	OK	OK
110	3	OK	all(3)	OK	OK
119	0	0	all(0)	OK	OK
131	5	5	all(5)	OK	OK
132	OK	1	OK	1	1
134	all	all	all	OK	n/a

Table 2: Failed-chip summary for chip number 51 to 134.
First entry: number of affected chip. In subsequent columns OKmeans passed the test; a number indicates the channel that failed; in the 'buffer' column an entry $n(m)$ indicates that channel m in buffer n failed. Failure of the trigger or hit map test may be due to either tdc or fadc failure (in this table). n/a means not applicable

---

be tested quickly. Ten minutes (total) per chip then translate into 22 work days, or one month.

In the current setup, a detailed analog characterization of all channels comprises of 1000 data points spanning the ADC ranges and takes about 5min. The other 5min will be used for the digital tests. While analog evaluation can be done with fewer data points, the digital tests cannot be sped up.

Data from the analog tests are stored on disk for future reference, while digital test results are discarded if the chip passed the particular test in question and stored otherwise.

The actual testing has been automatized to the point where the operator needs to do no more than to insert the chip into the test socket and push a button.

Digital testing follows a simple routine. At first, a file containing a user-

supplied to-do-list is loaded. This list consists of names of command files, one for each test. Each command file contains mnemonic instructions which are converted into a pattern generator program and a test vector sequence. The program is downloaded to the pattern generator, and the test vector sequence is fed to an ELEFANT ASIC simulator. After running the pattern generator and reading and unpacking the data from the logic analyser they are compared with the expected results obtained by the simulator. Only if they match does the chip pass this particular digital test. This is repeated for all tests on the to-do-list.

With this approach, the number and nature of the digital tests can be changed without the need to change anything in the software. All the work to be done involves only short, human-readable files.

Analog tests are controlled by a number of parameters passed to the main program via a front panel. The parameters in question are the range and step size of input pulses to the ELEFANT ASIC. After data taking, the program computes for each channel the offset, slope, INL and DNL and compares these to limits set by values appearing in the front panel.

In a similar manner, the power supply voltage and the sys_clock frequency will also be controlled from the front panel.

Below follows the proposed test sequence. It has been designed with two aspects in mind. First it should test the chip as completely as can reasonably expected, and secondly it should recognize rejects early. The list begins with digital tests, and analog characterizations will not be performed until the chip has passed all digital tests.

1) Running in test mode, verify that all four buffers work. Correctness of the data in the buffers also ensures that the sys_clock counter, the 'synch' command and L1_accept work correctly. The four L1_accepts are timed such that one can see all 8 bits in the data stream toggling. During the first test, the power consumption is logged automatically and compared to preset limits.

2) In test mode verify that multiple readout of the same buffer works. Use simultaneous data acquisition and readout during the first read.

3) SRAM2 test in test mode. Verify that the hit map, buffer and tag word, and the sys_clock counter value are reported correctly. Time the L1_accepts such that one can see all counter bits toggle. Choose the

four tag words with a similar goal in mind. Note that in the current setup the hit map can only be checked for all channels in parallel, ie it will be either all 1's or all 0's.

4) Trigger test in normal mode. See the above note.

5) Correct δ-bit generation, as a function of threshold and FADC value. This is a multiple test. It is performed by sending a ramped pulse to the analog receivers and recording the FADC values. One then compares the δ-bits in the data stream with the expectation derived from the threshold value and the FADC values. In order to cover the whole range one has to run this test a couple of times with various ramps and thresholds.

6) Full characterization of FADC and TDC perfoprmance. All eight channels are treated at the same time. A run with 128 data points is short and allows a good measurement of the slope, offset, and linearity. If detailed information about the bin width variation (DNL) or the maximum code error (INL) is required, then 1000 data points are necessary.