KLM: TARGETX

Automated Testing with User-Interface
Bronson Edralin
05/08/15

Outline

I. KLM: TARGETX Testing Flow

II. User Interface for Testing TARGETX

III. Automate the Tests

IV. Pedestals

V. OPTIMIZE_BIAS

VI. TIMING_RESOLUTION_TEST

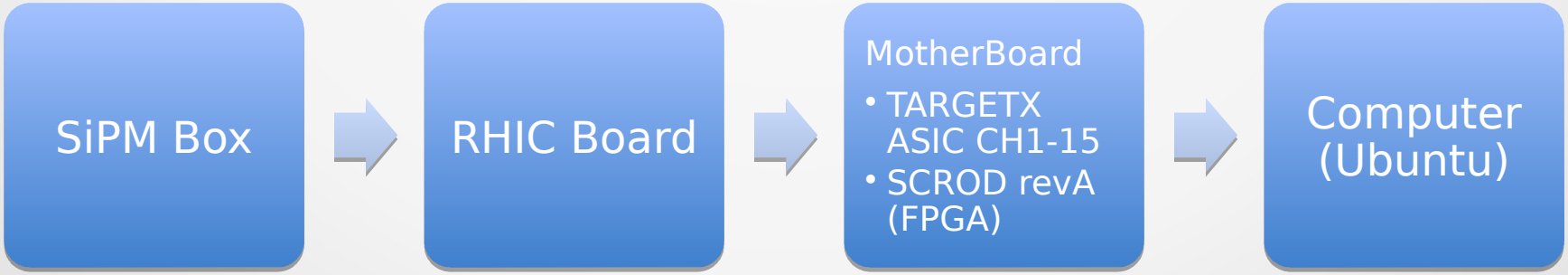
VII. PRODUCTION_TEST

VIII. Lessons Learned

KLM: TARGETX Testing Flow

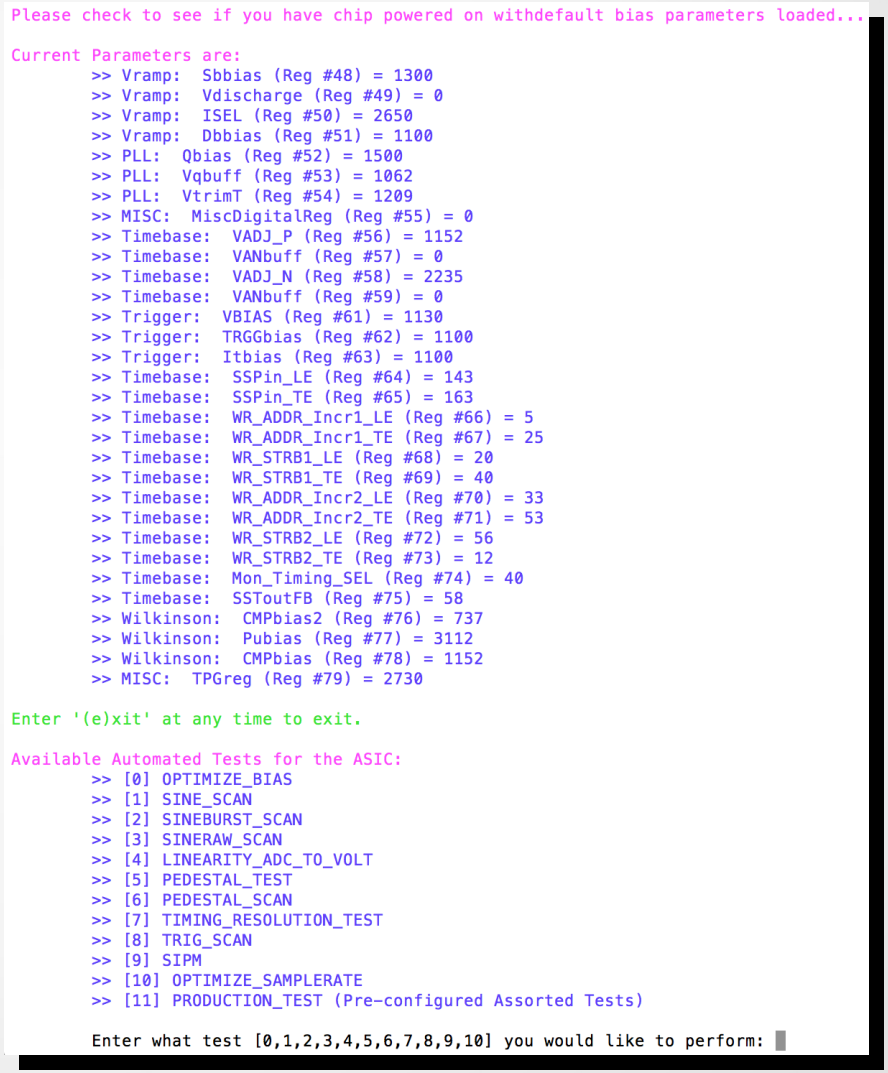
Optimizing, Evaluating and Testing TARGETX ASIC

Verify health of RHIC Board and Hardware Trigger



User Interface for Testing TARGETX

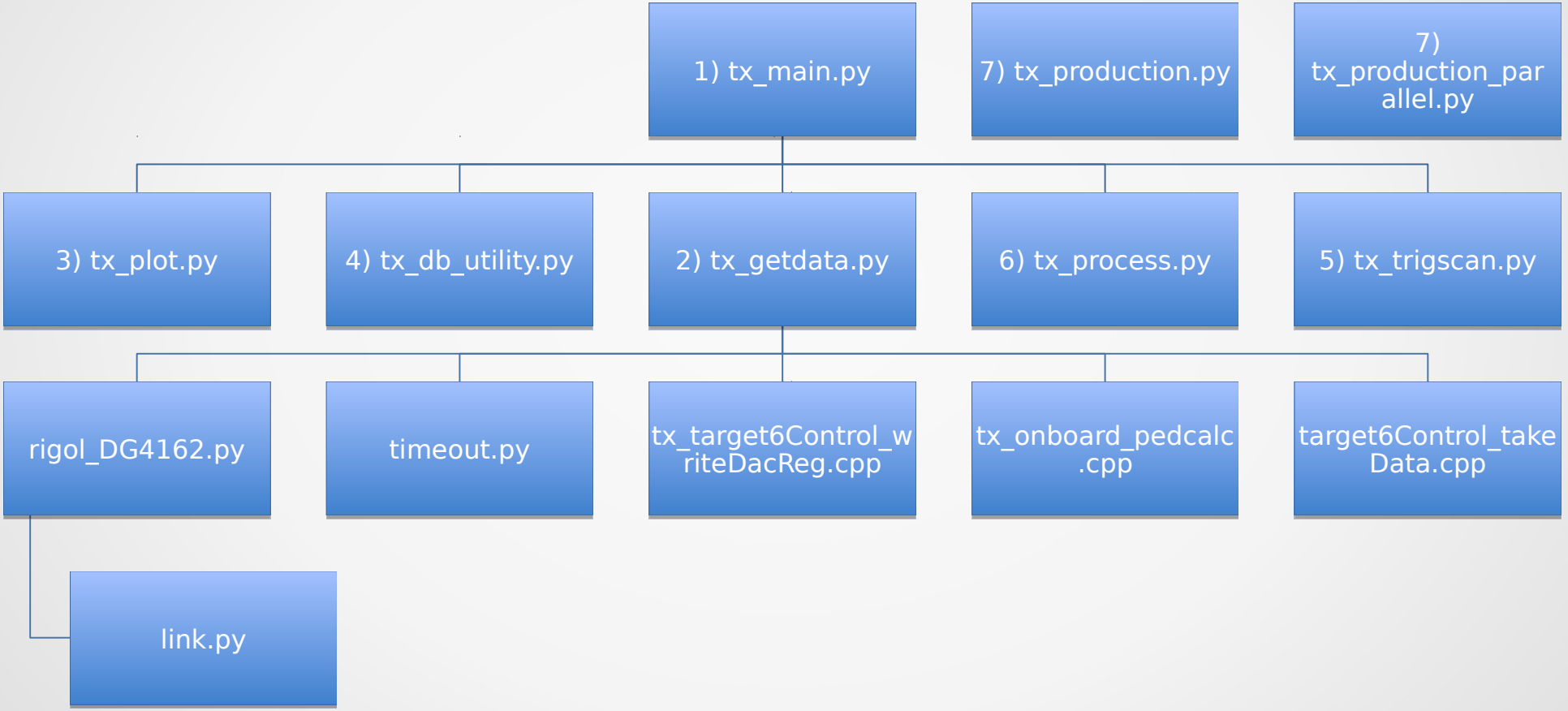
- Enter name of Tester
- Choose Test
- Enter Serial # of Motherboard, RHIC Board, and SCROD
- Choose ASIC number (0-9) and enter Serial # of the TARGETX
- Select Window range between 0-511 Windows (Note: **Stitches 4 Windows to readout a waveform**)
- Select Number of Events
- If optimizing register values, able to choose what register to optimize



Automate the Tests
1) Descriptions [1/3]

- **OPTIMIZE_BIAS**
  - Used to optimize timing register values
- **SINE_SCAN**
  - Verify clean waveform
- **SINEBURST_SCAN**
  - Scan sinusoid in burst mode
- **LINEARITY_TEST**
  - Extract transfer function of ADC Count vs Voltage
- **PEDESTAL_TEST**
  - Characterize input noise
- **TIMING_RESOLUTION_TEST**
  - Histogram of input period
- **TRIG_SCAN**
  - Verify trigger frequency
- **SIPM**
  - Plot temperatures and currents

Automate the Tests
2) Layout of Scripts [2/3]

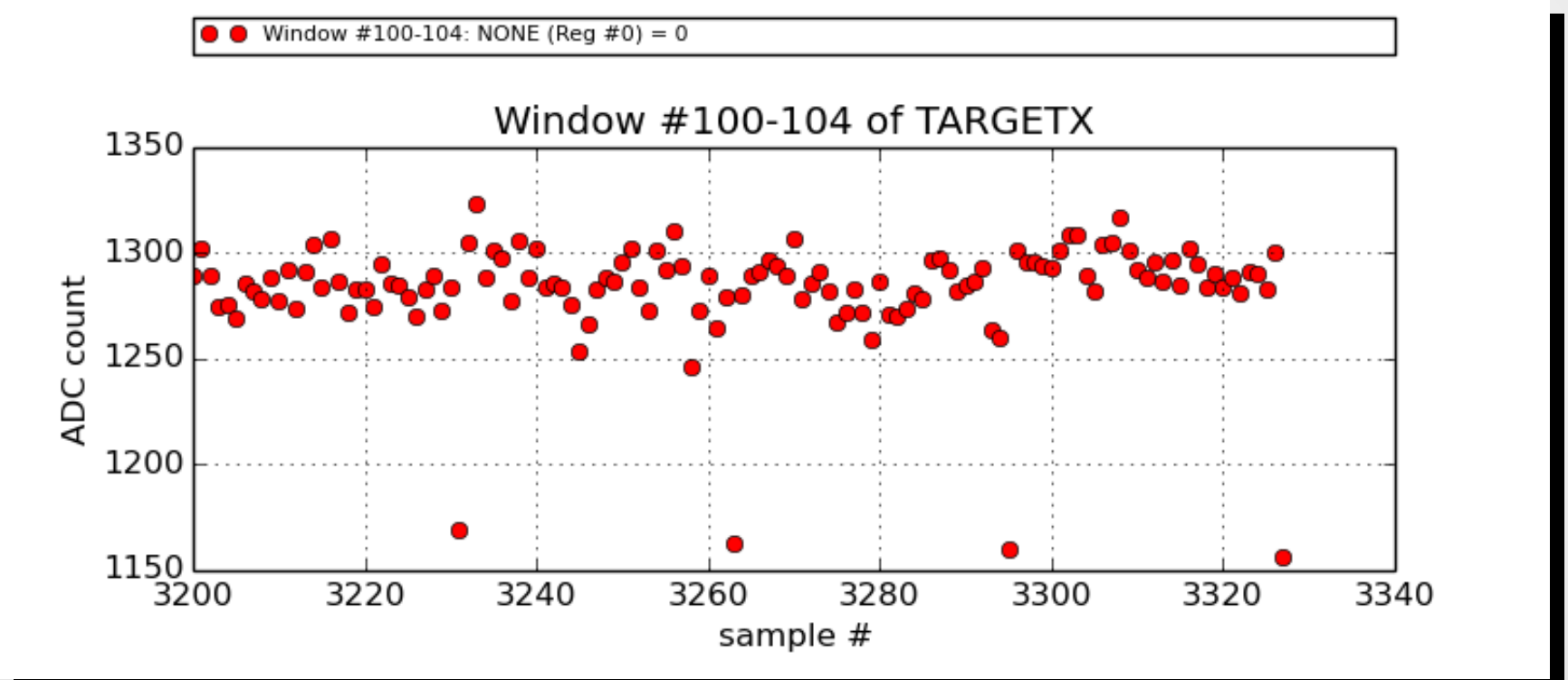


Automate the Tests
3) Layout of Scripts [3/3]

1) **main.py**
    1) User Interface that help ensure right inputs
2) **tx_getdata.py**
3) **tx_plot.py**
4) **tx_db_utility.py**
    1) Upload test data to PostgreSQL database
5) **tx_trigscan.py**
    - TRIG_SCAN
6) **tx_process.py**
    1) Process following tests
        1) OPTIMIZE_BIAS
        2) SINE_SCAN
        3) SINEBURST_SCAN
        4) PEDESTAL_TEST
        5) LINEARITY_ADC_TO_VOLT
        6) TIMING_RESOLUTION_TEST
7) **tx_production.py or tx_production_parallel.py**
    1) PRODUCTION_TEST
        1) Choose ASIC #0-9
        2) Pre-Configure Tests

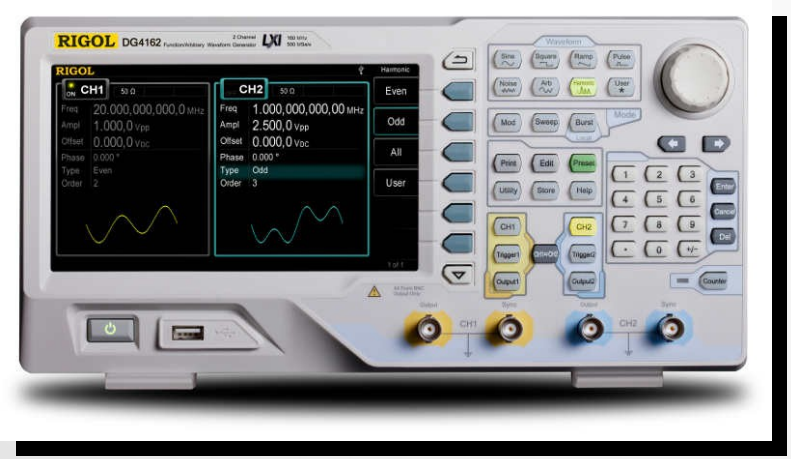
Pedestals
1) What they look like? [1/2]

Must subtract offsets in cells to readout clean waveform



Pedestals
2) How Pedestals are done? [2/2]

- AC Coupled Input
  - Steps
    - Change Bias Register Value
    - Turn OFF Func Gen
    - Generate Pedestals
    - Turn ON Func Gen
    - Get Data



http://www.rigolna.com/products/waveform-generators/dg4162/

- DC Coupled Input
  - Steps
    - Change Bias Register Value
    - Turn ON Func Gen
      - Change Amplitude to 1mVPP (smallest)
    - Generate Pedestals
    - Turn ON Func Gen
      - Change Amplitude back to Default Amplitude
    - Get Data

OPTIMIZE_BIAS
1) Algorithm [1/5]

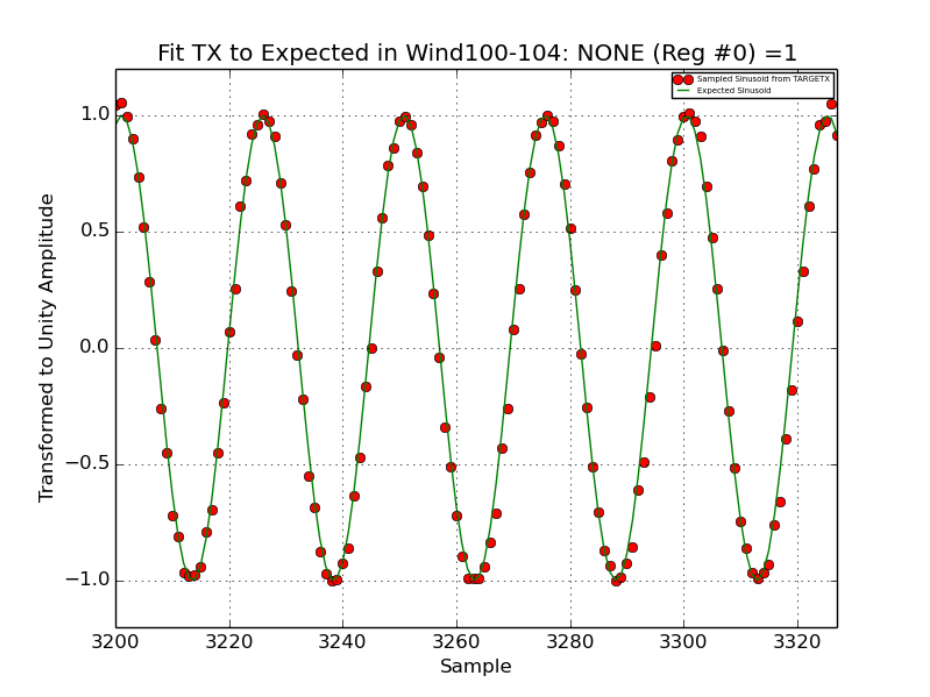
"Fitting and Chi-Squared Test Algorithm" *to choose optimum register value*

1. Control a function generator to inject a 40MHz Sinusoid with 900mVPP Amplitude
2. Readout and construct waveform "X"
3. Scale amplitude of waveform "X" to unity amplitude
4. Construct an expected sinusoid "E" by sampling (at 1 GSa/s) a 40MHz Sinusoid with unity amplitude
5. Use matched filter to achieve synchronization for fitting with waveforms "X" and "E"
6. Plot synchronized waveforms "X" and "E" onto same plot and call it Fitting#
7. Plot residuals for "X" and "E"
8. Calculate modified-Chi-Squared Result of "X" and "E" and log these raw values
9. Use average of the raw values of the modified-Chi-Squared Results for waveforms with multiple events to determine optimum bias register value

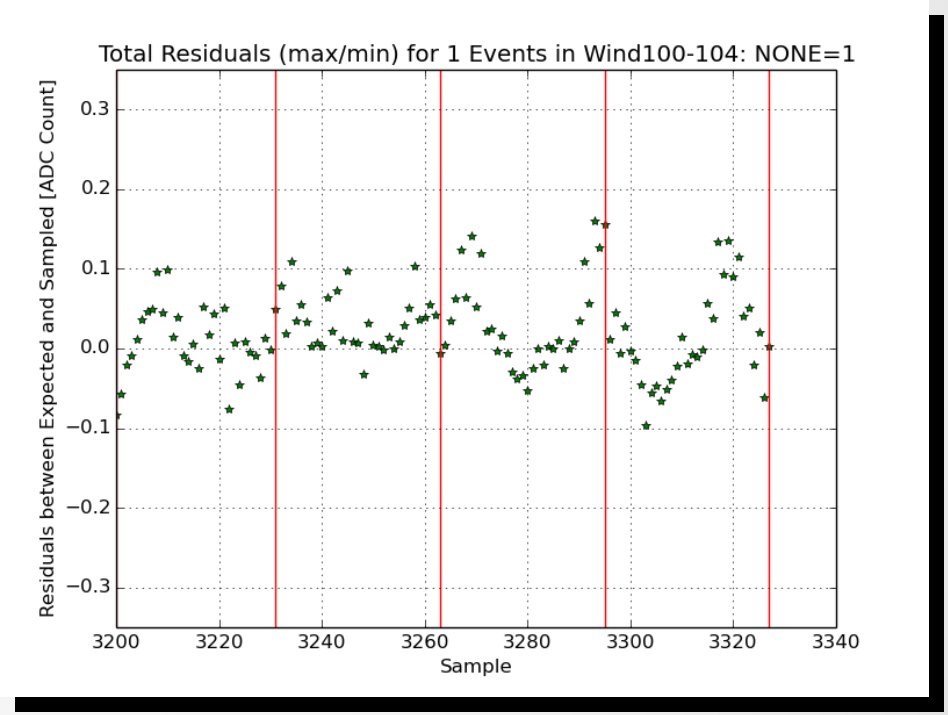
$$\chi^2 = \sum \frac{(observed - expected)^2}{NumSamples}$$

OPTIMIZE_BIAS
2) Fitting [2/5]

Used Matched Filter for Synchronization



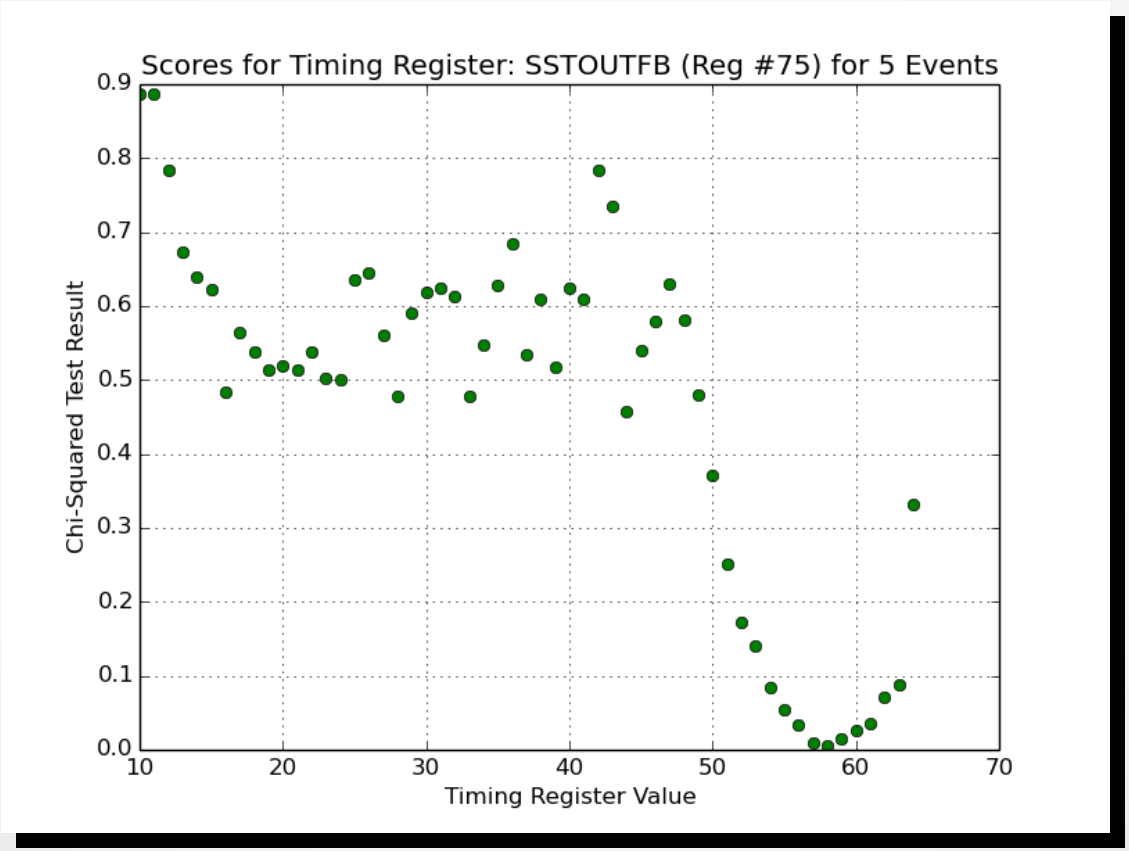
Residuals = observed - expected



OPTIMIZE_BIAS

3) Scanning Bias Register [3/5]

Sample optimization scan



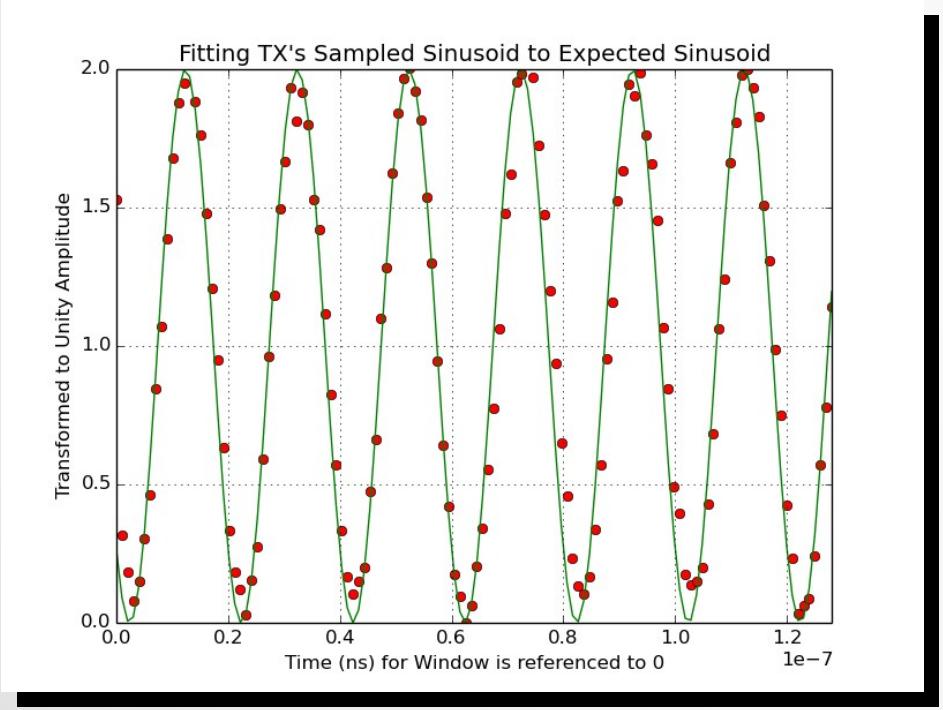
OPTIMIZE_BIAS
4) BiasRev #1.1 [4/5]

Signal	Register #	Default Value (old)	Run values	Default Value (new)
SSTIN_N				
VADJ_P	56	1152		
VADJ_N	58	2235		
ISEL	50	2650		
Vdischarge	49	0		
WL_CLK_p			63.7 MHz	
CMPbias2	76	737		
Pubias	77	3112		
Qbias	52	1500		
VANbuff	57	0		
Vqbuff	53	1062		
VtrimT	54	1209		
CMPbias	78	1152		

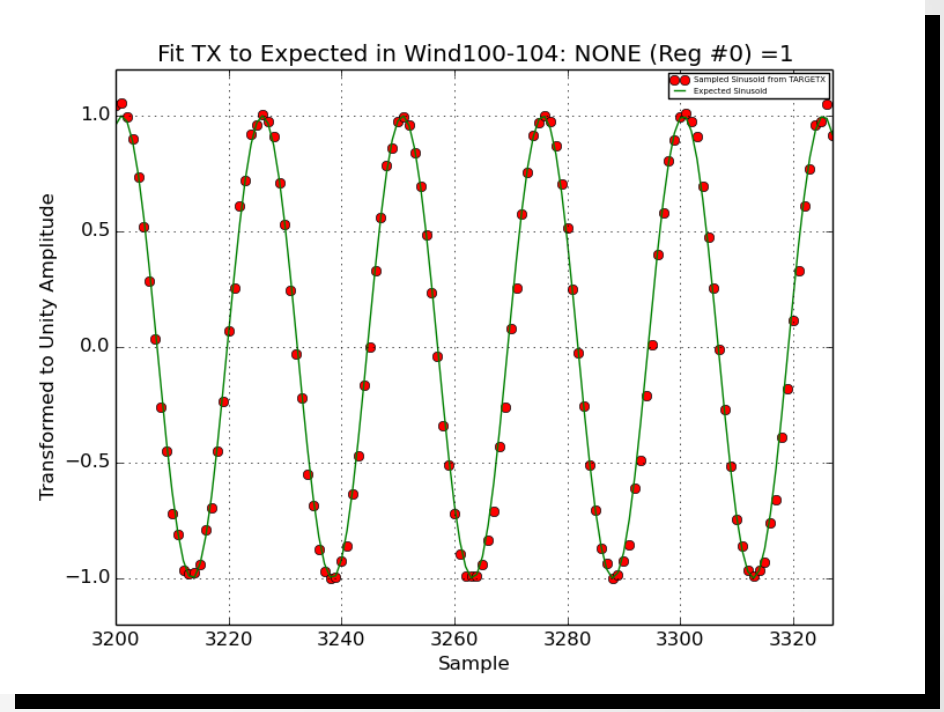
Signal	Register #	Default Value (old)	Run values	Default Value (new)
Mon_Timing_SEL	74	40		
WR_ADDR_Incr1 LE	66	13		
WR_ADDR_Incr1 TE	67	33		
WR_STRB1 LE	68	20		
WR_STRB1 TE	69	40		
WR_ADDR_Incr2 LE	70	33		
WR_ADDR_Incr2 TE	71	53		
WR_STRB2 LE	72	56		
WR_STRB2 TE	73	12		
SSToutFB	75	58		
VBIAS	61	1130		

OPTIMIZE_BIAS
5) Results from Optimization [5/5]

BEFORE (1/17/15): *Fitted 50MHz Sinusoid with Bias Rev #0*

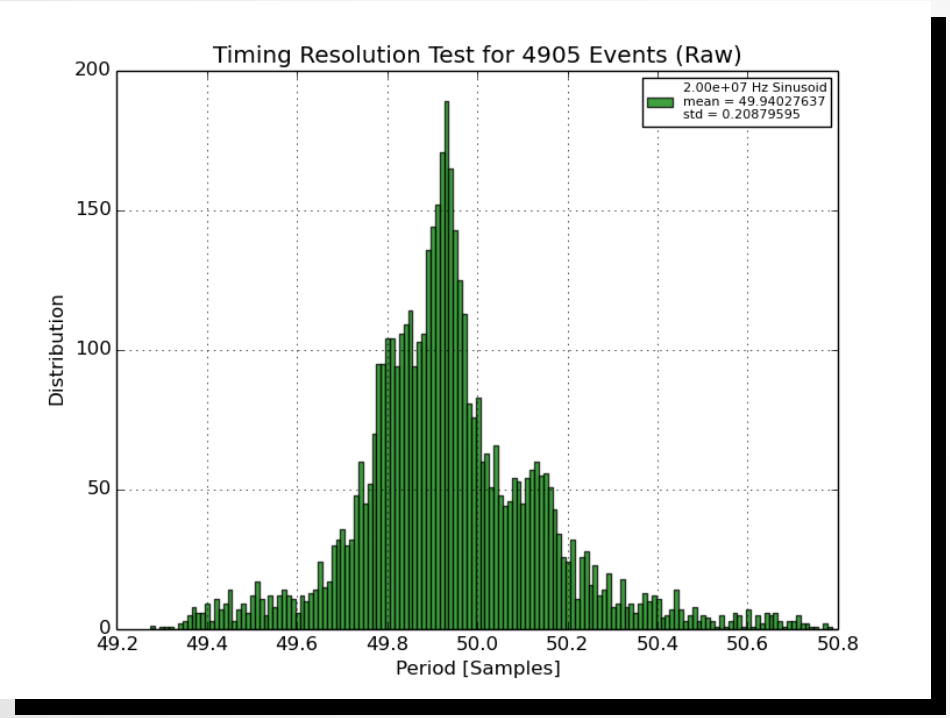


AFTER (4/12/15): *Fitted 40MHz Sinusoid with BiasRev #1.1*

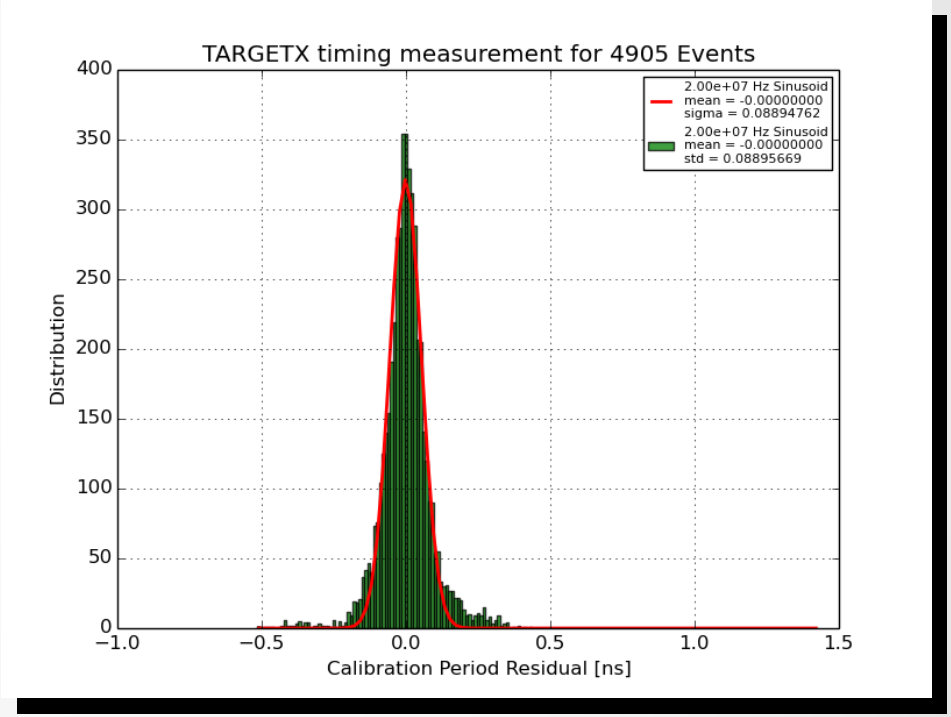


TIMING_RESOLUTION_TEST

1) Timing Calibration [1/3]



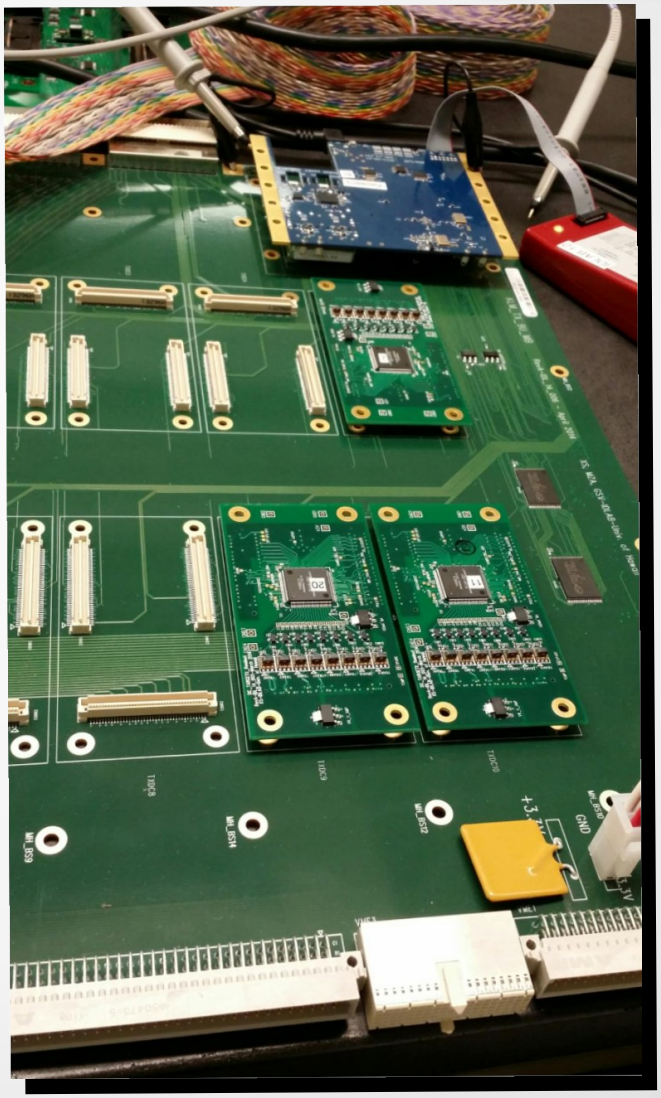
*Estimated TARGETX Sampling Rate = 20MHz*49.94027637 =* ***0.9988055 GSa/s****)*



After correcting for spatial dependence on starting position time, we get ***88.95 ps time resolution****.*

PRODUCTION_TEST

- Pre-configured Tests
  - OPTIMIZE_BIAS
  - SINE_SCAN
  - PEDESTAL_TEST
  - TRIG_SCAN
  - SIPM
- Sequential/Parallel Processing
- Hierarchal Folder Storage
- Logging and storing in database



Lessons Learned