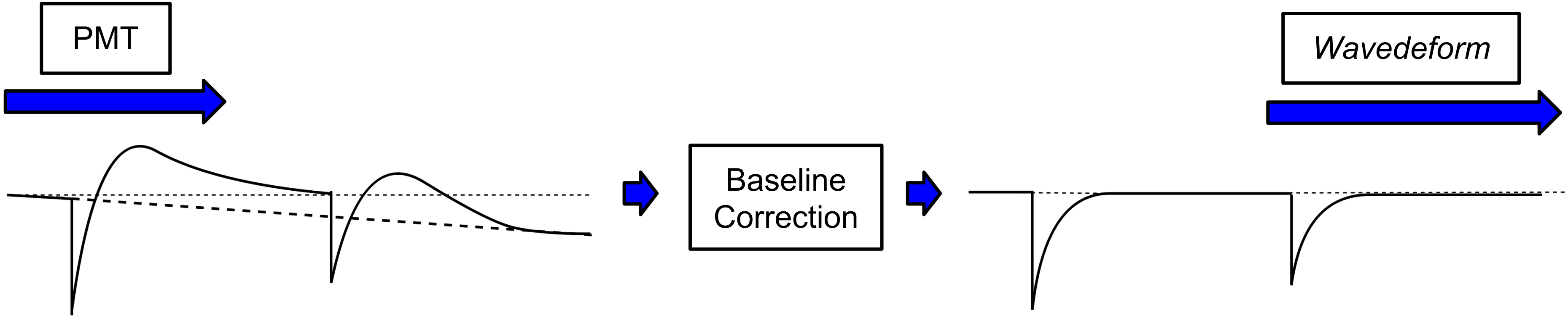
Baseline Correction

<u>Goal</u>: design and verification of FPGA baseline correction module (drift and droop) which can run inside of Gen-2 DOM.

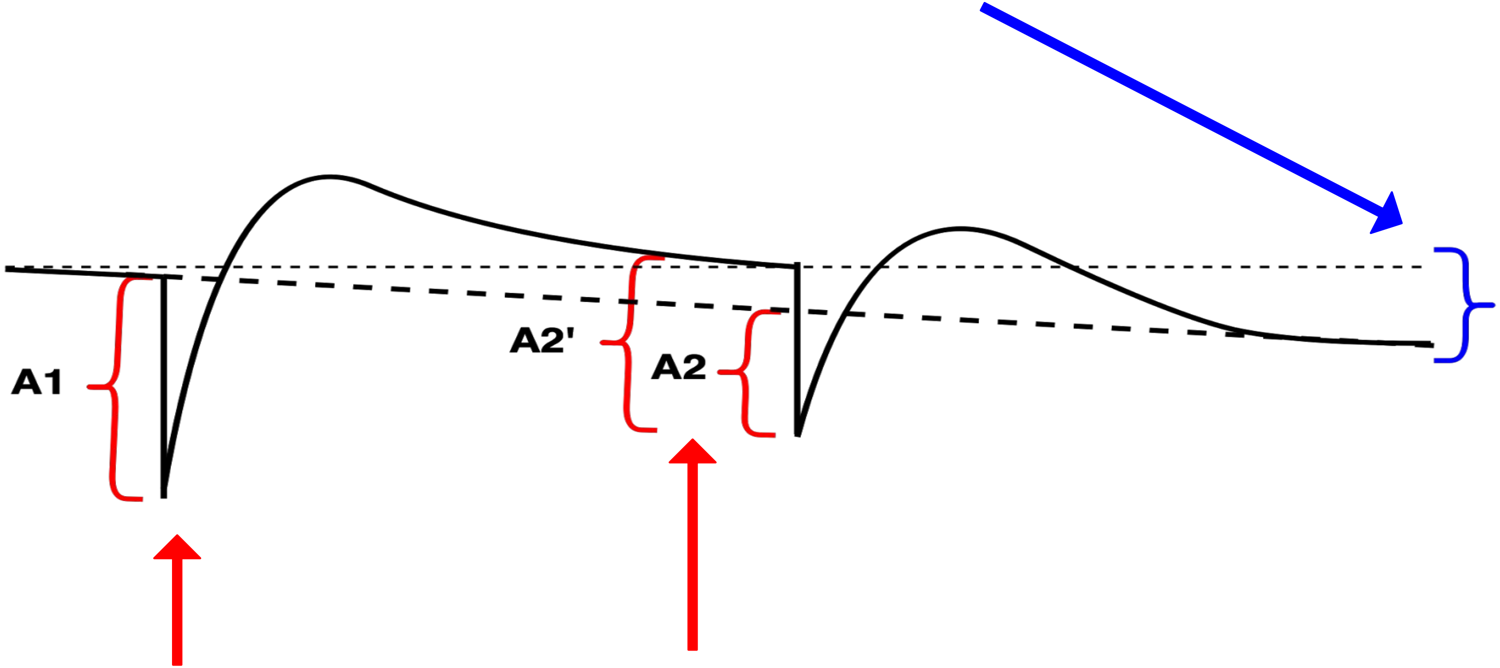
Prerequisite for running *Wavedeform* in the DOM.



Daria Pankova (dup193@psu.edu), Tyler Anderson

Drift and Droop

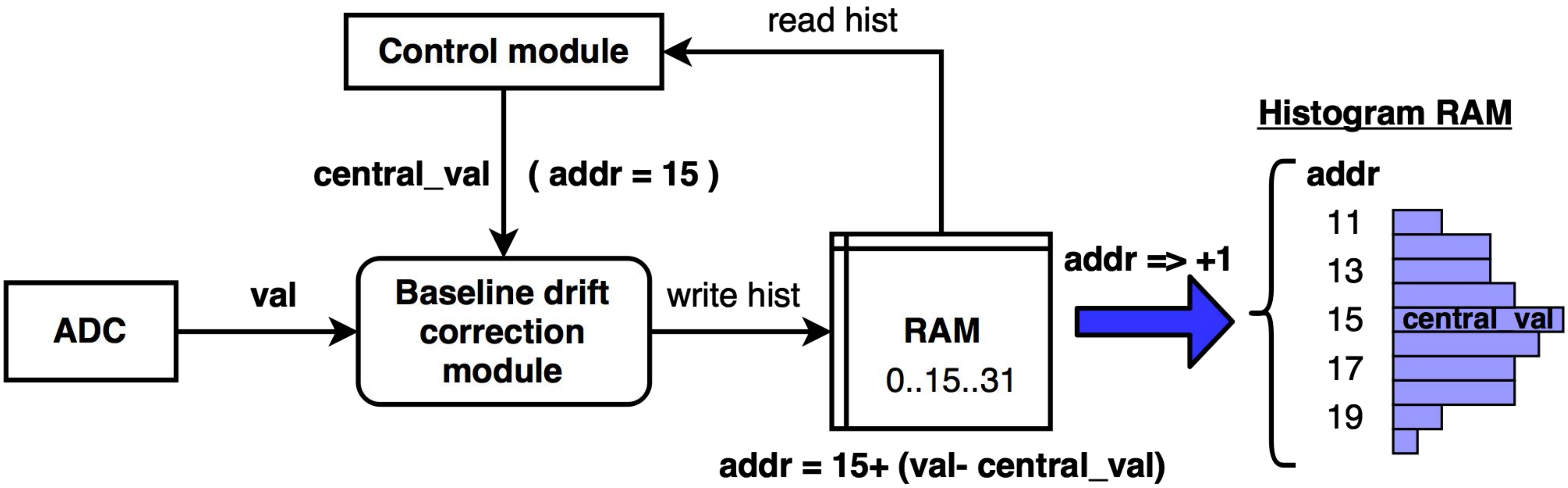
Drift - long term change of baseline due to electronics and environment



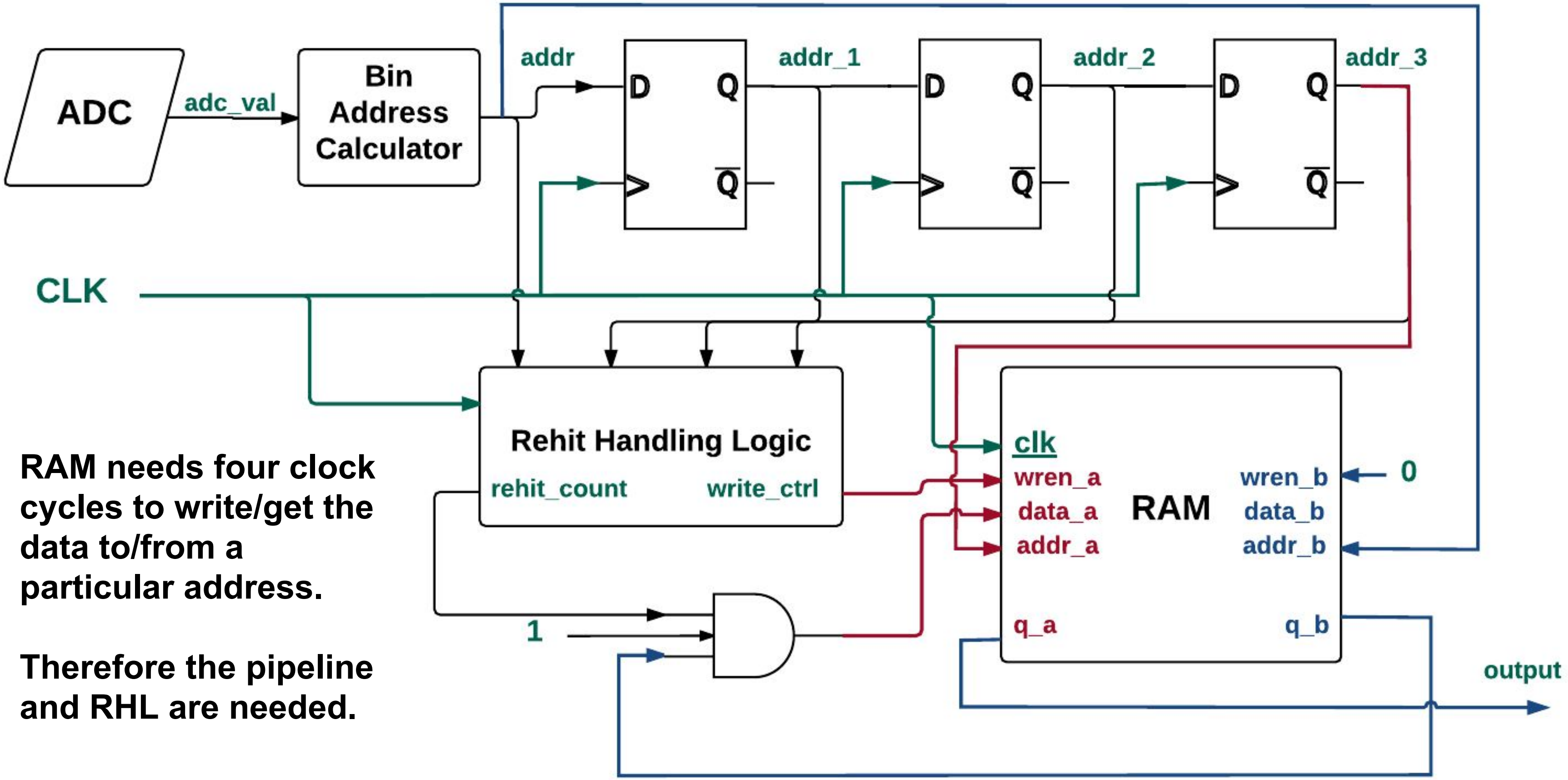
Droop - a change due to toroid, following a signal.

Drift correction.

Idea: **Histogramming. Fill the RAM with ADC values and find the baseline from the mean (central_val).**



Drift correction module diagram

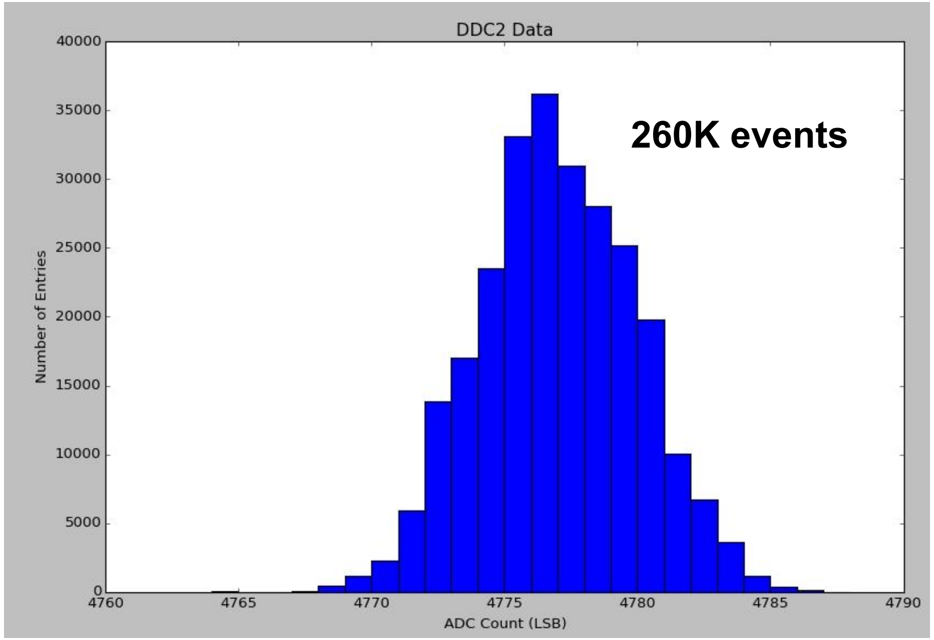


RAM needs four clock cycles to write/get the data to/from a particular address.

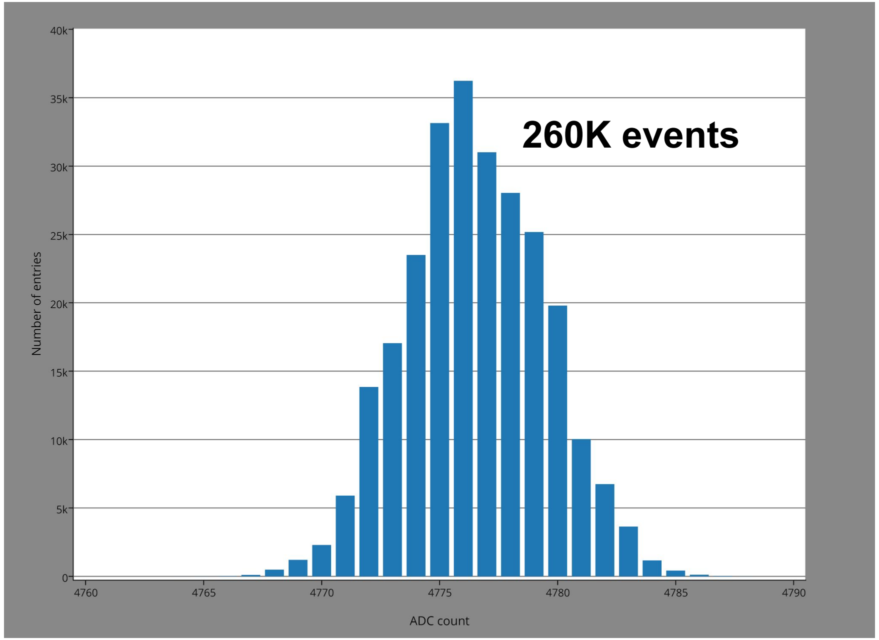
Therefore the pipeline and RHL are needed.

Drift correction testing with *ModelSim*.

They match!

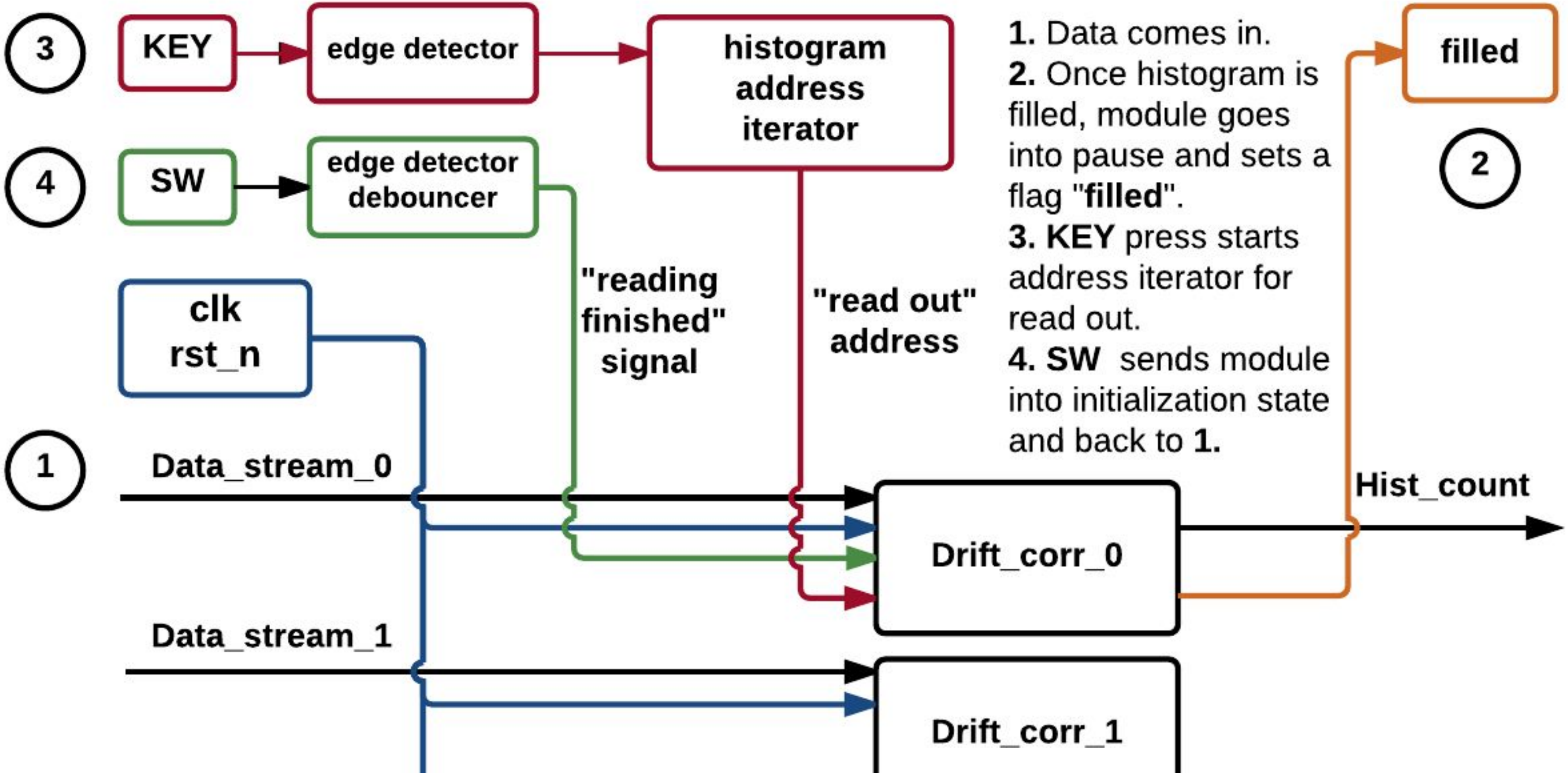


Python. Expected Histogram.



ModelSim. Histogram result.

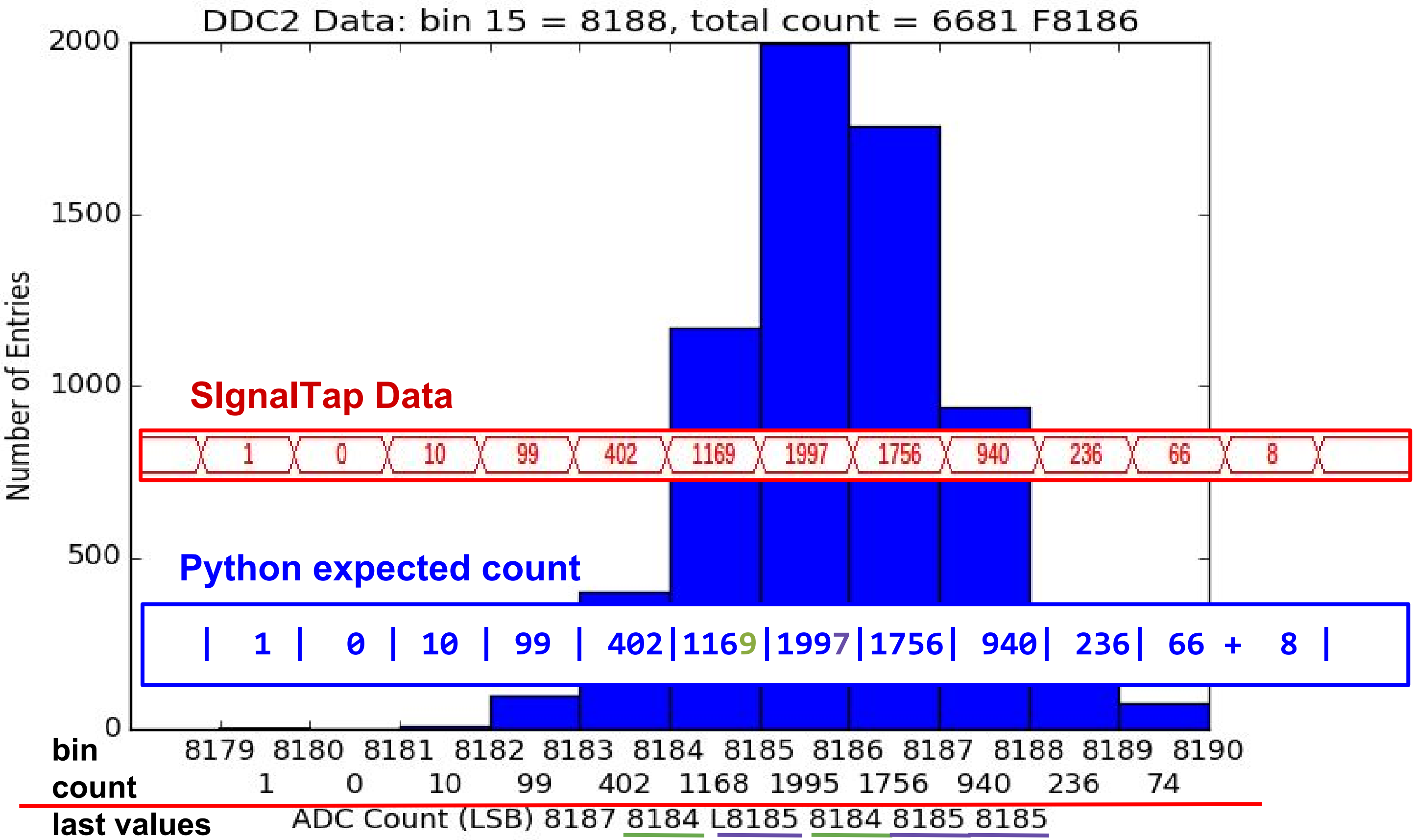
Drift Correction on FPGA.



1. Data comes in.
2. Once histogram is filled, module goes into pause and sets a flag "**filled**".
3. KEY press starts address iterator for read out.
4. SW sends module into initialization state and back to **1.**

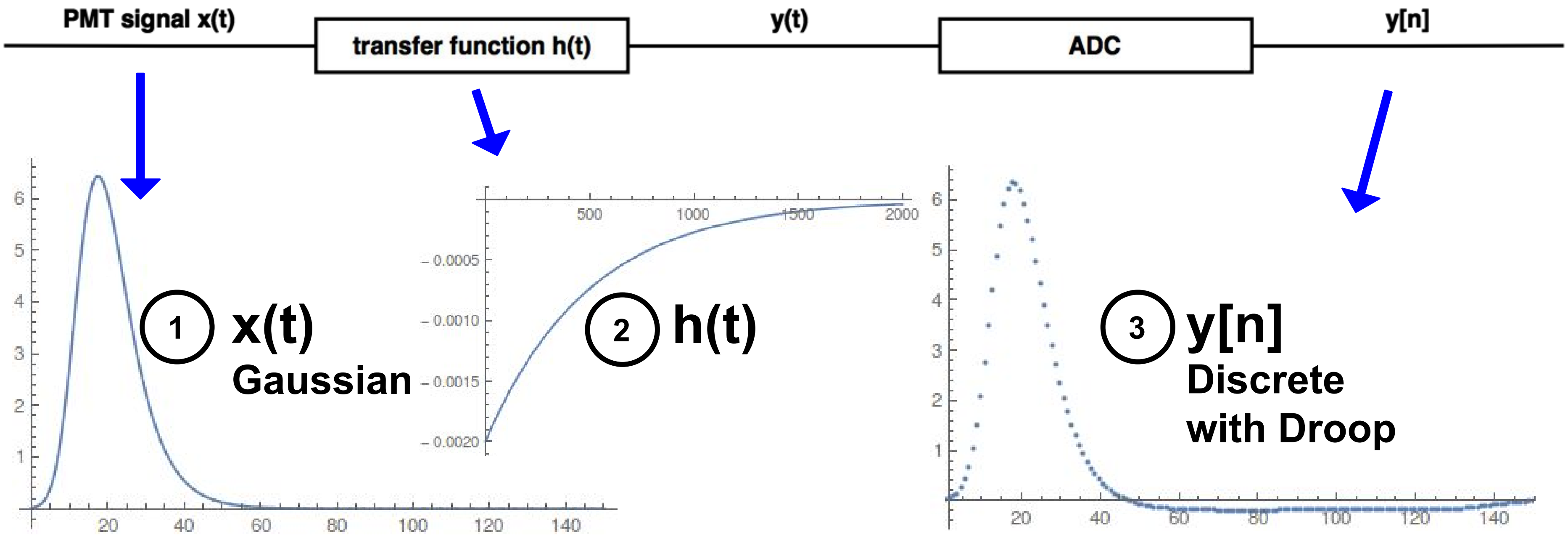
Drift correction testing on FPGA.

They match! Using Quartus' SignalTap. <u>150K</u> events (22 batches). Example batch:



Droop Correction.

Signal Transformation:



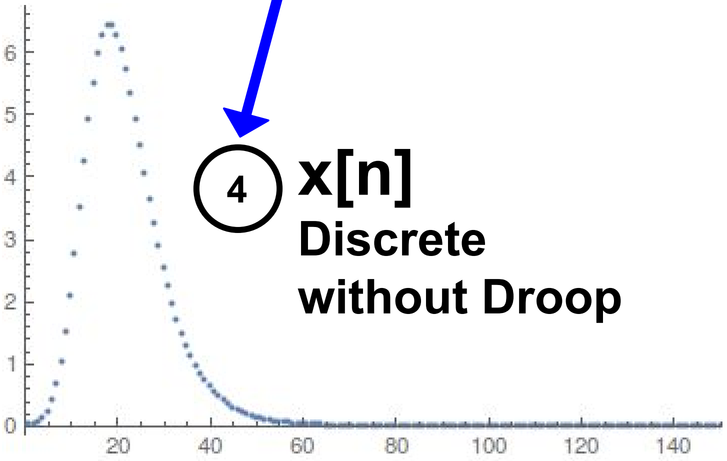
Goal: Find x[n] from y[n]. x[n] is Discrete and without Droop.

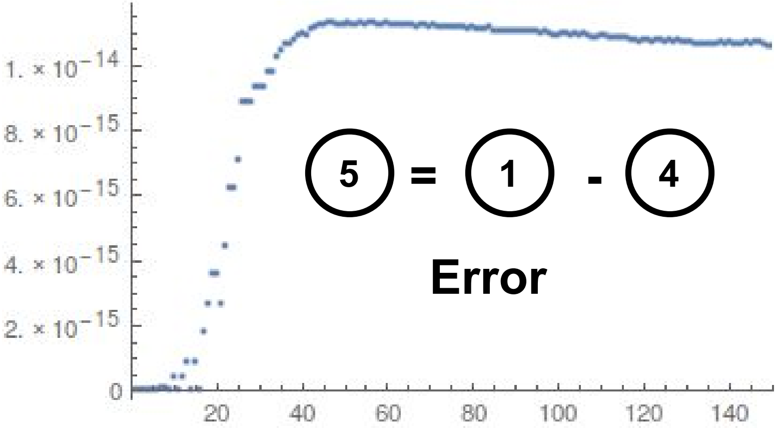
Droop Correction. Possible strategy.

- Calculated possible transfer function h[n] (Impulse invariance).
- Convolution $$y[n] = \sum_{k=-\infty}^{\infty} h[n-k]x[k]$$
- Recursive solution:

$$\boxed{x[n] = a \cdot (b \cdot (y[n-1] - x[n-1]) - y[n])}$$

$$\boxed{x[0] = -a \cdot y[0]}$$





Conclusion

Done

- **Drift monitoring module implemented and tested**

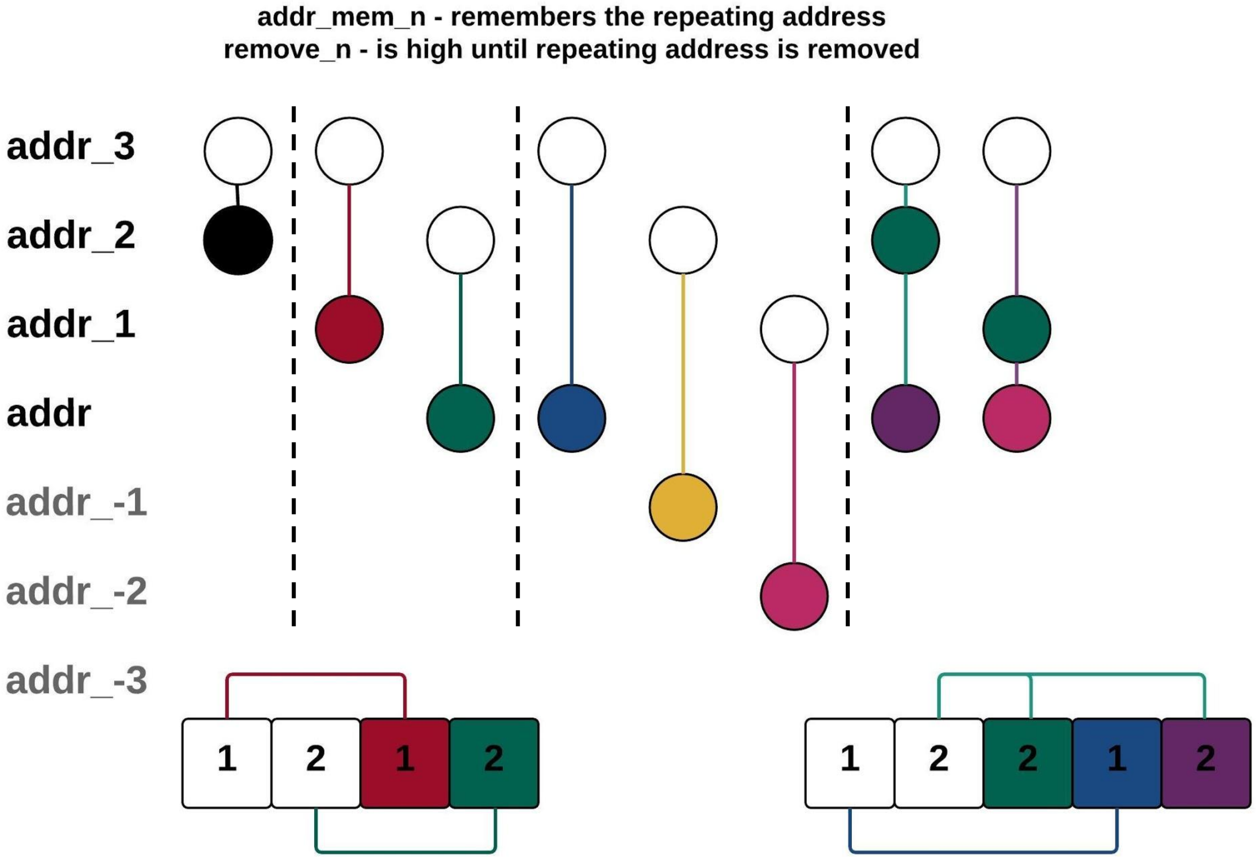
To be done

- **Droop monitoring module implementation and testing**
- **Module to control all separate modules and do correction**

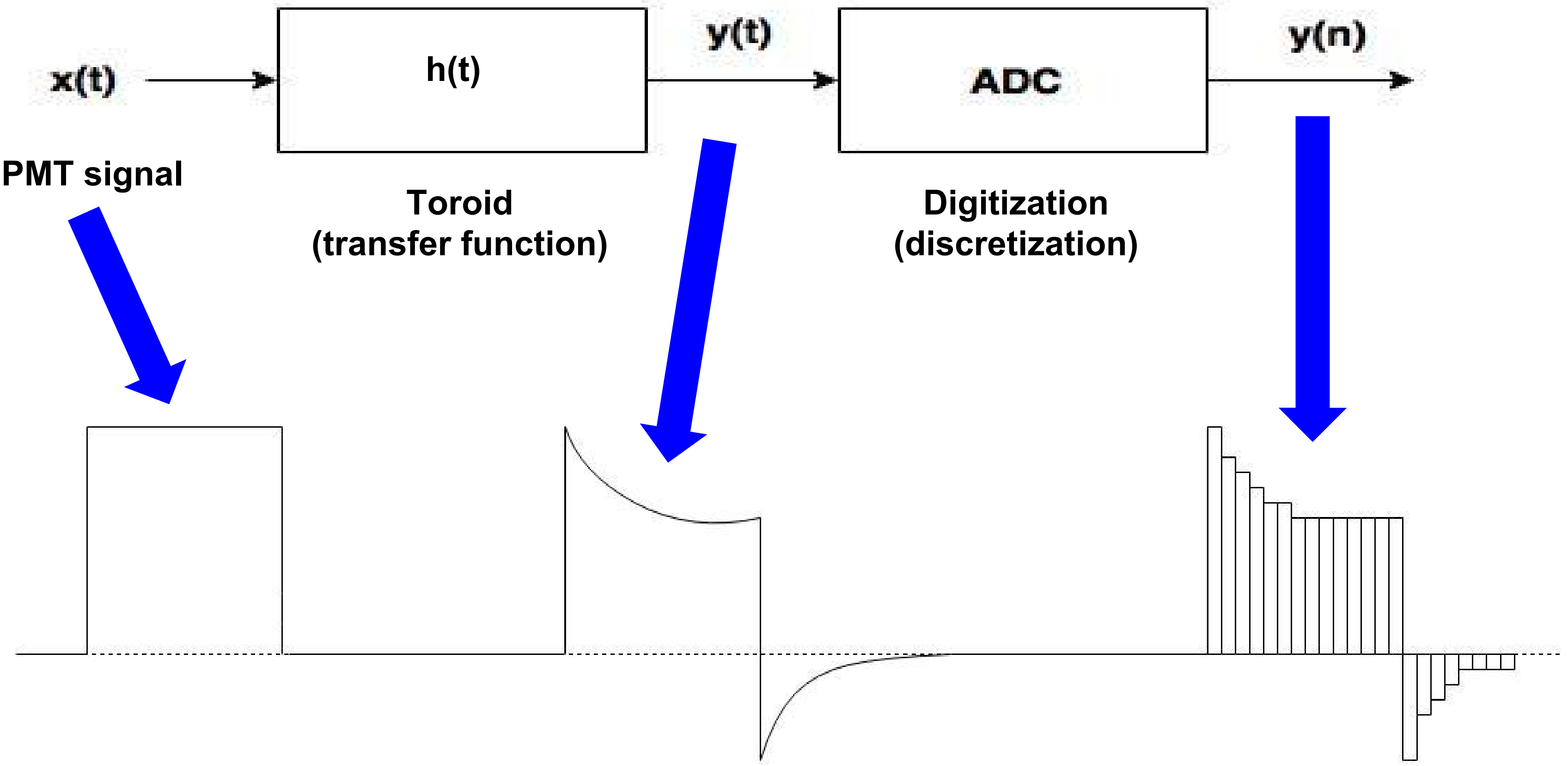
Back up slides

Rehit Handling strategy

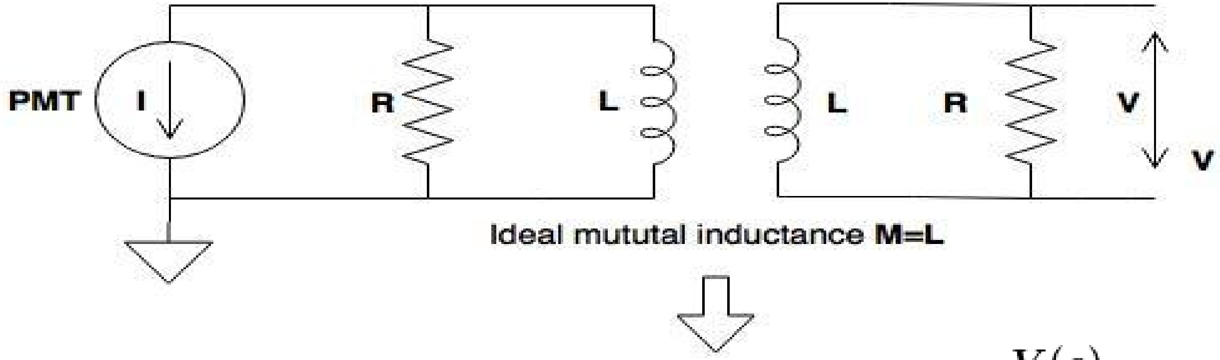
Due to RAM limitations in Pipeline, same bin can't be rehit within 4 cycles

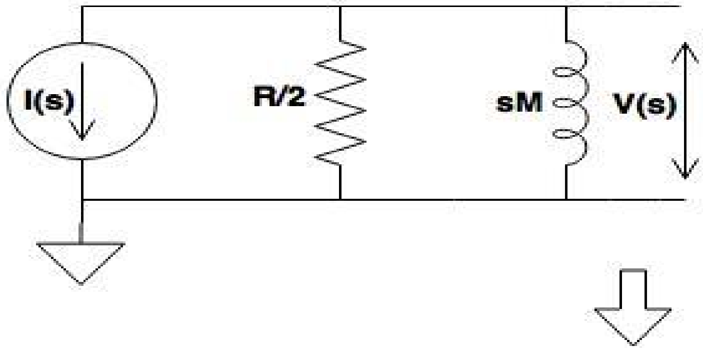


Droop Correction Detail
Signal transformation scheme



Transfer function





$$H(s) = \frac{V(s)}{I(s)} = Z(s)$$

$$\mathrm{H(s)} = \frac{\mathrm{R}}{2} \frac{\mathrm{s}\tau}{1 + \mathrm{s}\tau}$$

Recursive expression

Transfer function

$$h(t) = L^{-1}\{H(s)\} \xrightarrow{\text{Mathematica}} h(t) = \begin{cases} -\dfrac{e^{\frac{-t}{\tau}}}{\tau} + \delta(t), & t \geq 0 \\ 0, & t < 0 \end{cases}$$

Discretization

ADC frequency $f = \dfrac{1}{T}, \Gamma = \dfrac{\tau}{T} \rightarrow h[n] = \begin{cases} -\dfrac{e^{\frac{-n}{\Gamma}}}{\Gamma} + \delta[n], & n \geq 0 \\ 0, & n < 0 \end{cases}$

Convolution

$$y(t) = \int_{-\infty}^{\infty} h(t-t')x(t')dt' \rightarrow y[n] = \sum^{\infty} h[n-k]x[k]$$

$$y[n] = \sum_{k=0}^{n} \left(\frac{-e^{\frac{k-n}{\Gamma}}}{\Gamma} + \delta[n-k] \right) x[k]$$

Expansion

$$\mathbf{y}(0) = -\left(\frac{1}{\Gamma} - 1\right)\mathbf{x}[0]$$

$$\mathbf{y}(1) = -\frac{e^{-\frac{1}{\Gamma}}}{\Gamma}\mathbf{x}[0] - \left(\frac{1}{\Gamma} - 1\right)\mathbf{x}[1]$$

$$\mathbf{y}(2) = -\frac{e^{-\frac{2}{\Gamma}}}{\Gamma}\mathbf{x}[0] - \frac{e^{-\frac{1}{\Gamma}}}{\Gamma}\mathbf{x}[1] - \left(\frac{1}{\Gamma} - 1\right)\mathbf{x}[2]$$

$$\mathbf{y}(3) = -\frac{e^{-\frac{3}{\Gamma}}}{\Gamma}\mathbf{x}[0] - \frac{e^{-\frac{2}{\Gamma}}}{\Gamma}\mathbf{x}[1] - \frac{e^{-\frac{1}{\Gamma}}}{\Gamma}\mathbf{x}[2] - \left(\frac{1}{\Gamma} - 1\right)\mathbf{x}[3]$$

$$\vdots$$

$$\boxed{\mathbf{y}[\mathbf{n}] = -\frac{e^{-\frac{n}{\Gamma}}}{\Gamma}\mathbf{x}[0] - \frac{e^{-\frac{n-1}{\Gamma}}}{\Gamma}\mathbf{x}[1] - \cdots - \frac{e^{-\frac{1}{\Gamma}}}{\Gamma}\mathbf{x}[\mathbf{n}-1] - \left(\frac{1}{\Gamma} - 1\right)\mathbf{x}[\mathbf{n}]}$$

[

Recursive form

$$\mathbf{y(n)} = \underline{-\frac{e^{-\frac{n}{\Gamma}}}{\Gamma}x[0] - \frac{e^{-\frac{n-1}{\Gamma}}}{\Gamma}x[1] - \cdots -} \frac{e^{-\frac{1}{\Gamma}}}{\Gamma}x[n-1] - \left(\frac{1}{\Gamma} - 1\right)x[n]$$

$$y(n-1) = -\frac{e^{-\frac{n-1}{\Gamma}}}{\Gamma}x[0] - \frac{e^{-\frac{n-2}{\Gamma}}}{\Gamma}x[1] - \cdots - \frac{e^{-\frac{1}{\Gamma}}}{\Gamma}x[n-2] - \left(\frac{1}{\Gamma} - 1\right)x[n-1]$$

$$e^{-\frac{1}{\Gamma}}y(n-1) = \underline{-\frac{e^{-\frac{n}{\Gamma}}}{\Gamma}x[0] - \frac{e^{-\frac{n-1}{\Gamma}}}{\Gamma}x[1] - \cdots - \frac{e^{-\frac{2}{\Gamma}}}{\Gamma}x[n-2]} - \left(\frac{1}{\Gamma} - 1\right)e^{-\frac{1}{\Gamma}}x[n-1]$$

$$y[n] = e^{-\frac{1}{\Gamma}}y(n-1) + \left(\cancel{\frac{1}{\Gamma}} - 1\right)e^{-\frac{1}{\Gamma}}x[n-1] - \cancel{\frac{e^{-\frac{1}{\Gamma}}}{\Gamma}x[n-1]} - \left(\frac{1}{\Gamma} - 1\right)x[n]$$

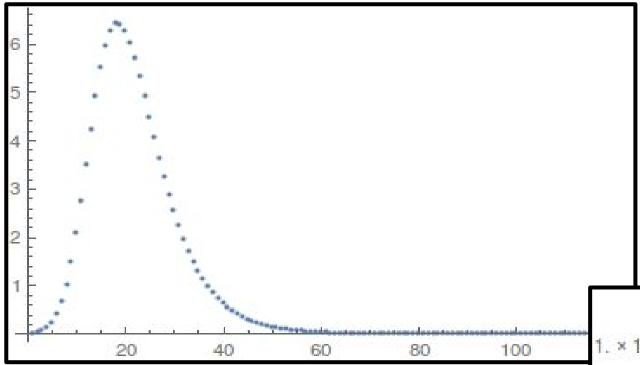
$$y[n] = e^{-\frac{1}{\Gamma}}y[n-1] - e^{-\frac{1}{\Gamma}}x[n-1] - \left(\frac{1}{\Gamma} - 1\right)x[n]$$

$$x[n] = \left(\frac{\Gamma}{1-\Gamma}\right)\left(e^{-\frac{1}{\Gamma}}\left(y[n-1] - x[n-1]\right) - y[n]\right) = a \cdot \left(b \cdot \left(y[n-1] - x[n-1]\right) - y[n]\right)$$

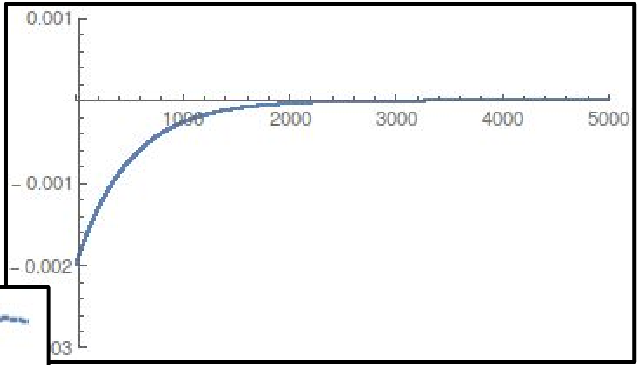
$$x[n] = a \cdot (b \cdot (y[n-1] - x[n-1]) - y[n]]) \qquad a = \frac{\Gamma}{1-\Gamma}, b = e^{-\frac{1}{\Gamma}}$$

$$x[0] = -a \cdot y[0]$$

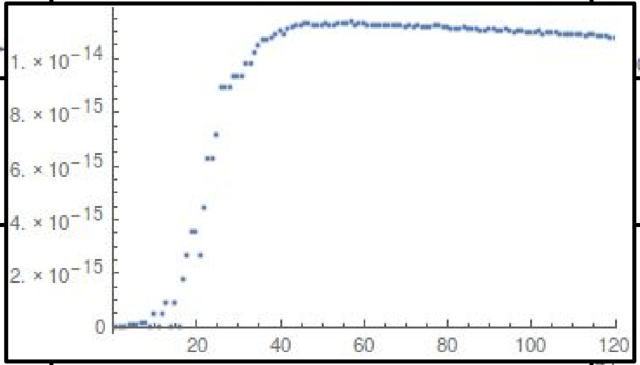
Numerical confirmation



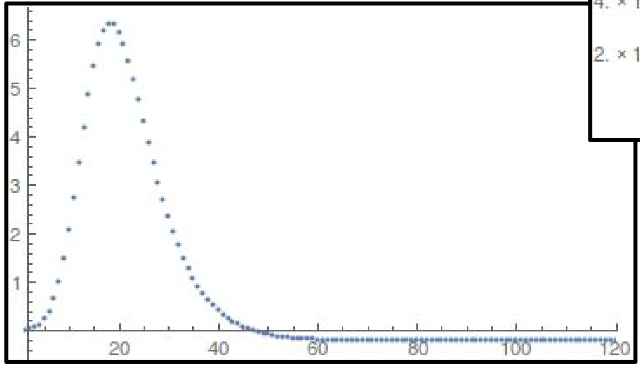
a) Gaussian input x(t)



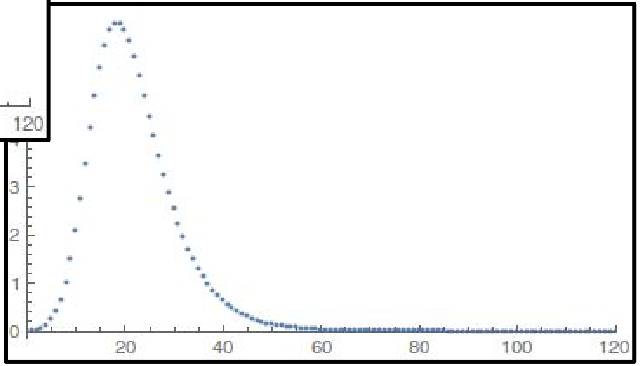
b) Transfer function h(t)



e) Error x(t)-x’(t)

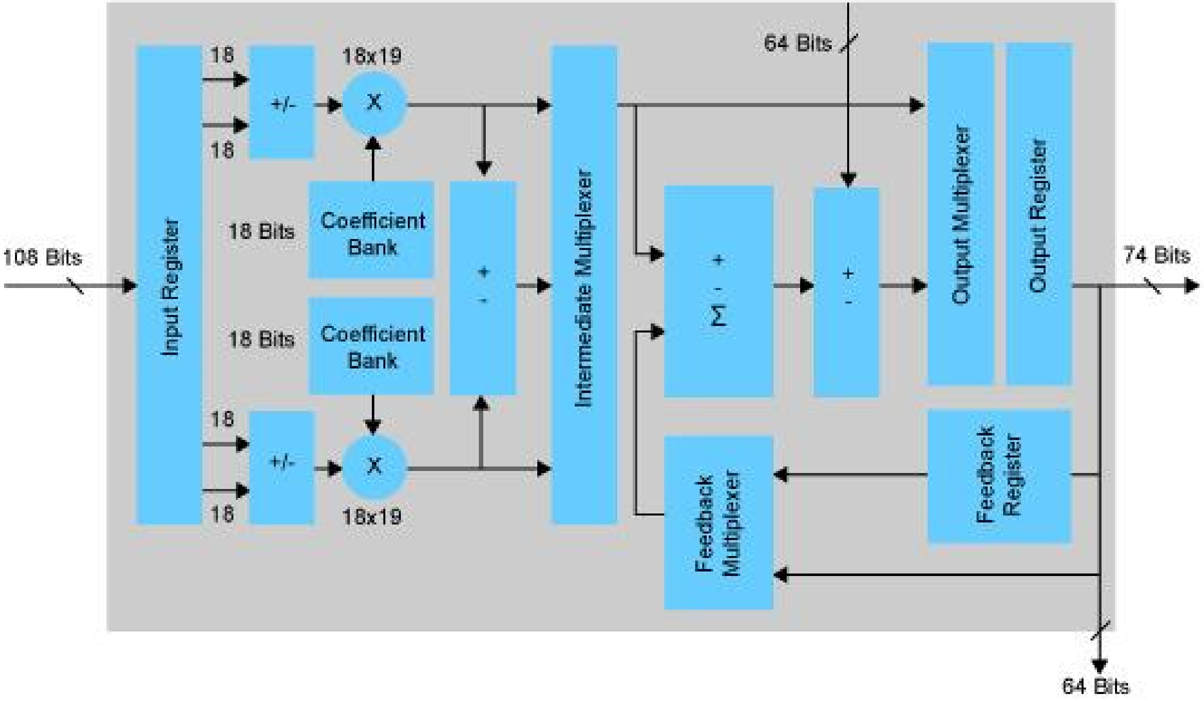


c) After convolution y(t)



d) After the method x’(t)

DSP block in FPGA



Need to:

- **Multiply by a constant**
- **Subtraction**
- **Storage of previous values**