ECG 722 – Mixed-Signal Circuit Design: Course Project Report
Dr. R. Jacob Baker

Continuous-Time 8-Path KD1S (8D1S) ADC

By
Dane Gentry

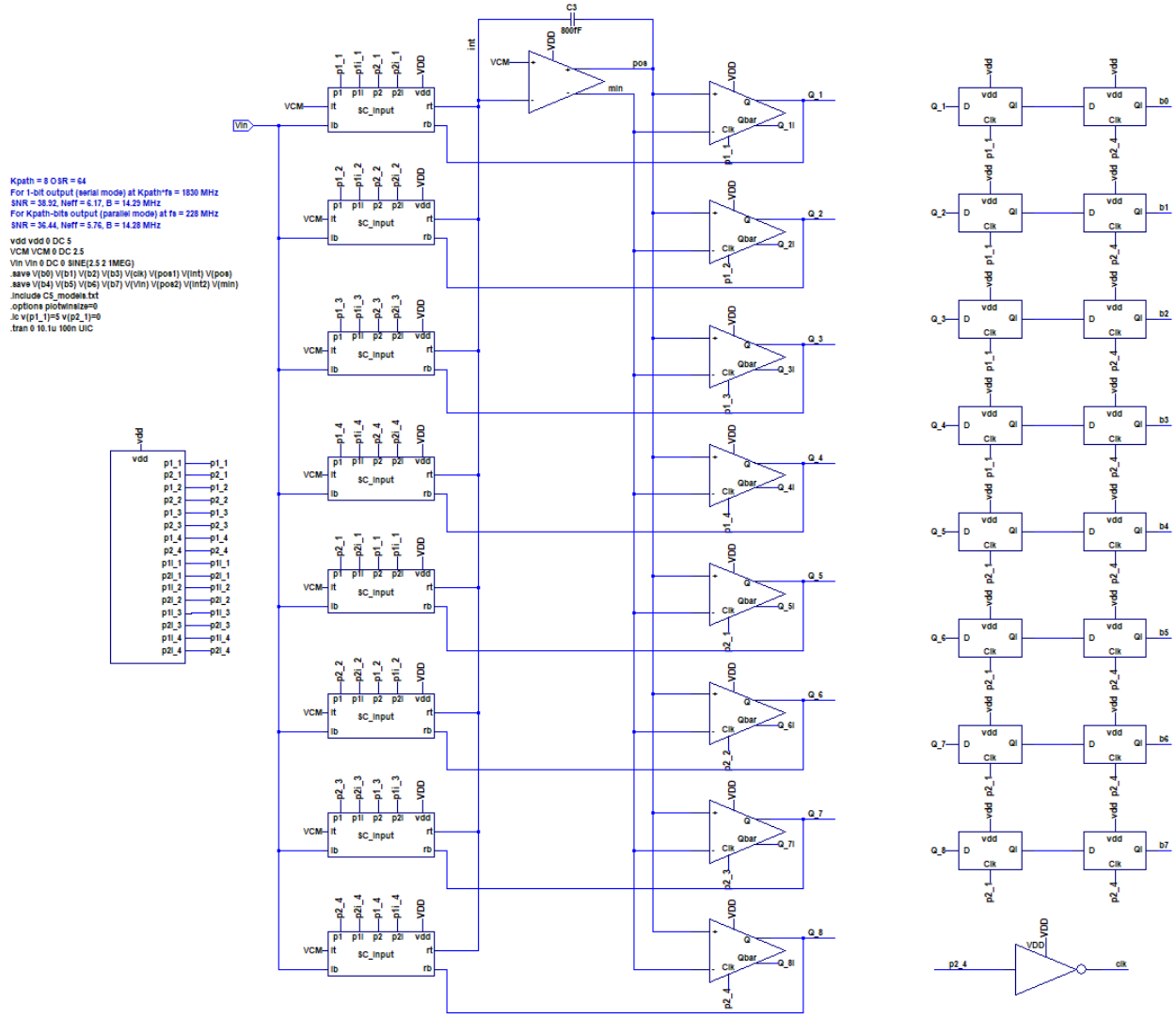
4/30/18

Table of Contents

Design Objective

The goal of this project was to design a replacement for the ideal 8-path delta-sigma (ΔΣ) ADC (K-delta 1-sigma modulator) seen in Fig. 9.32 of the "CMOS Mixed-Signal Circuit Design" textbook using a continuous-time topology (no switched-capacitors and no non-overlapping clock signals) and On Semiconductor's C5 500nm process. Though others were investigated, the topologies included in this report are a first-order, first-order with dither, and second-order.

The LTspice schematic "_Fig9_33_Matlab_MSD" associated with the textbook used an input signal frequency of 3MHz. Throughout this report, however, I will be comparing results using both a 1MHz and 3MHz input signal frequency. The modified schematic of "_Fig9_33_Matlab_MSD" for an input signal frequency of 1MHz is shown below as well as its corresponding MATLAB output results for an oversampling ratio (OSR) of 64, 128, and 256.



"_Fig9_33_Matlab_MSD_1MHz"

```
Kpath = 8 OSR = 64
For 1-bit output (serial mode) at Kpath*fs = 1829 MHz
SNR = 38.00, Neff = 6.02, B = 14.29 MHz
For Kpath-bits output (parallel mode) at fs = 228 MHz
SNR = 34.18, Neff = 5.38, B = 14.28 MHz>>
```

```
Kpath = 8 OSR = 128
For 1-bit output (serial mode) at Kpath*fs = 1829 MHz
SNR = 40.98, Neff = 6.51, B = 7.14 MHz
For Kpath-bits output (parallel mode) at fs = 228 MHz
SNR = 39.48, Neff = 6.26, B = 7.14 MHz>>
```

```
Kpath = 8 OSR = 256
For 1-bit output (serial mode) at Kpath*fs = 1829 MHz
SNR = 54.14, Neff = 8.70, B = 3.57 MHz
For Kpath-bits output (parallel mode) at fs = 228 MHz
SNR = 53.01, Neff = 8.51, B = 3.57 MHz>>
```

The MATLAB output results for "_Fig9_33_Matlab_MSD" with an input signal frequency of 3MHz and OSR's of 64, 128, and 256 are shown below.

```
Kpath = 8 OSR = 64
For 1-bit output (serial mode) at Kpath*fs = 1830 MHz
SNR = 39.97, Neff = 6.34, B = 14.30 MHz
For Kpath-bits output (parallel mode) at fs = 229 MHz
SNR = 38.93, Neff = 6.17, B = 14.28 MHz>>
```

```
Kpath = 8 OSR = 128
For 1-bit output (serial mode) at Kpath*fs = 1830 MHz
SNR = 53.73, Neff = 8.63, B = 7.15 MHz
For Kpath-bits output (parallel mode) at fs = 229 MHz
SNR = 51.18, Neff = 8.21, B = 7.14 MHz>>
```

```
Kpath = 8 OSR = 256
For 1-bit output (serial mode) at Kpath*fs = 1830 MHz
SNR = 58.41, Neff = 9.41, B = 3.57 MHz
For Kpath-bits output (parallel mode) at fs = 229 MHz
SNR = 59.04, Neff = 9.51, B = 3.57 MHz>>
```

The MATLAB output results for an input signal frequency of both 1MHz and 3MHz and OSR's of 64, 128, and 256 summarized in the tables below and will be the basis of comparison for the this report.

Note $f_{s,new} = K_{path} * f_s$ and $K_{path} = 8$ ➔ i.e. $1830 \approx 228*8$

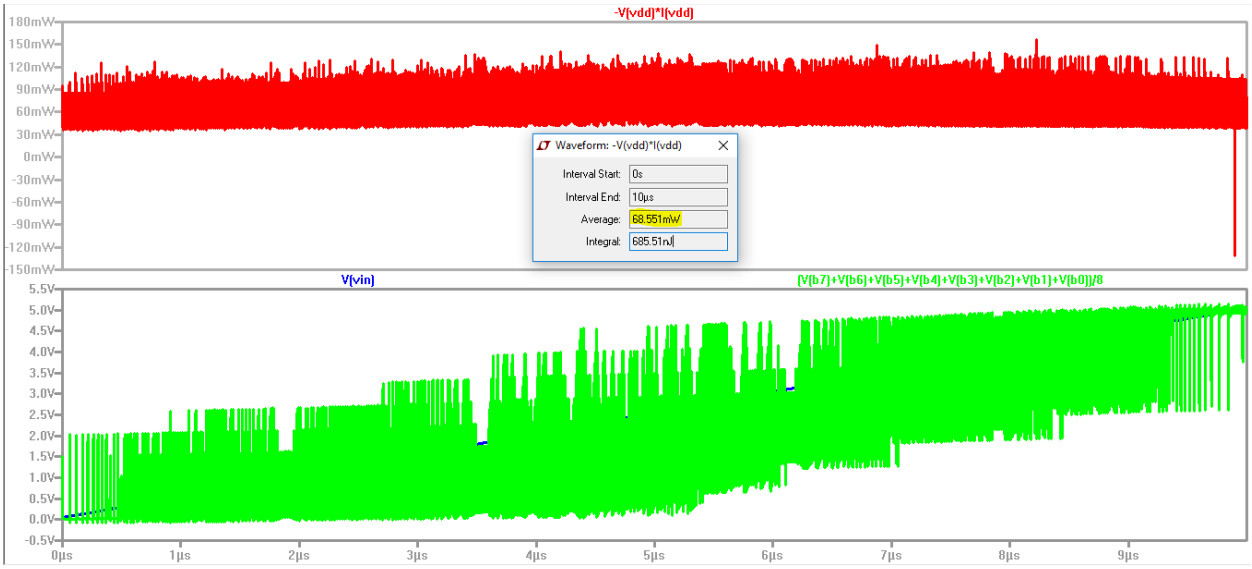
"_Fig9_33_Matlab_MSD_1MHz" (10 Cycles)

	Serial 64	Parallel 64	Serial 128	Parallel 128	Serial 256	Parallel 256
$f_{s,new}$ (MHz)	1829	228	1829	228	1829	228
SNR (dB)	38.00	34.18	40.98	39.48	54.14	53.01
N_{eff} (bits)	6.02	5.38	6.51	6.26	8.70	8.51
Bandwidth (MHz)	14.29	14.28	7.14	7.14	3.57	3.57

"_Fig9_33_Matlab_MSD_3MHz" (10 Cycles)

	Serial 64	Parallel 64	Serial 128	Parallel 128	Serial 256	Parallel 256
$f_{s,new}$ (MHz)	1830	229	1830	229	1830	229
SNR (dB)	39.97	38.93	53.73	51.18	58.41	59.04
N_{eff} (bits)	6.34	6.17	8.63	8.21	9.41	9.51
Bandwidth (MHz)	14.30	14.28	7.15	7.14	3.57	3.57

The power consumption of "_Fig9_33_Matlab_MSD" is shown in the simulation below and was found to be 68.55 mW (average). Also note the absence of dead zones in the output signal associated with a slow-moving ramp input signal.



Design Considerations

Signal-to-Noise Ratio (SNR) Calculation

The ideal signal-to-noise ratio (SNR) for a first and second-order K-path topology are calculated below.

$SNR_{ideal,1st} = 6.02N + 1.76 – 5.17 + 30\log_{10}K$

- 1.5 bit increase in resolution for every doubling of K

$SNR_{ideal,2nd} = 6.02N + 1.76 – 12.9 + 50\log_{10}K$

- 2.5 bit increase in resolution for every doubling of K

where N is the number of bits (here 1) and K is the OSR (here 64, 128, or 256)

K=64 ➔ $SNR_{ideal,1st}$ = 56.80dB & $SNR_{ideal,2nd}$ = 85.19dB

K=128 ➔ $SNR_{ideal,1st}$ = 65.83dB & $SNR_{ideal,2nd}$ = 100.24dB

K=256 ➔ $SNR_{ideal,1st}$ = 74.86dB & $SNR_{ideal,2nd}$ = 115.29dB

Effective Number of Bits (N_{eff}) Calculation

The effective number of bits (N_{eff}) can be calculated below. All of the MATLAB output values of SNR and N_{eff} summarized throughout this report follow this equation.

$$N_{eff} = \frac{SNR_{meas} - 1.76}{6.02}$$

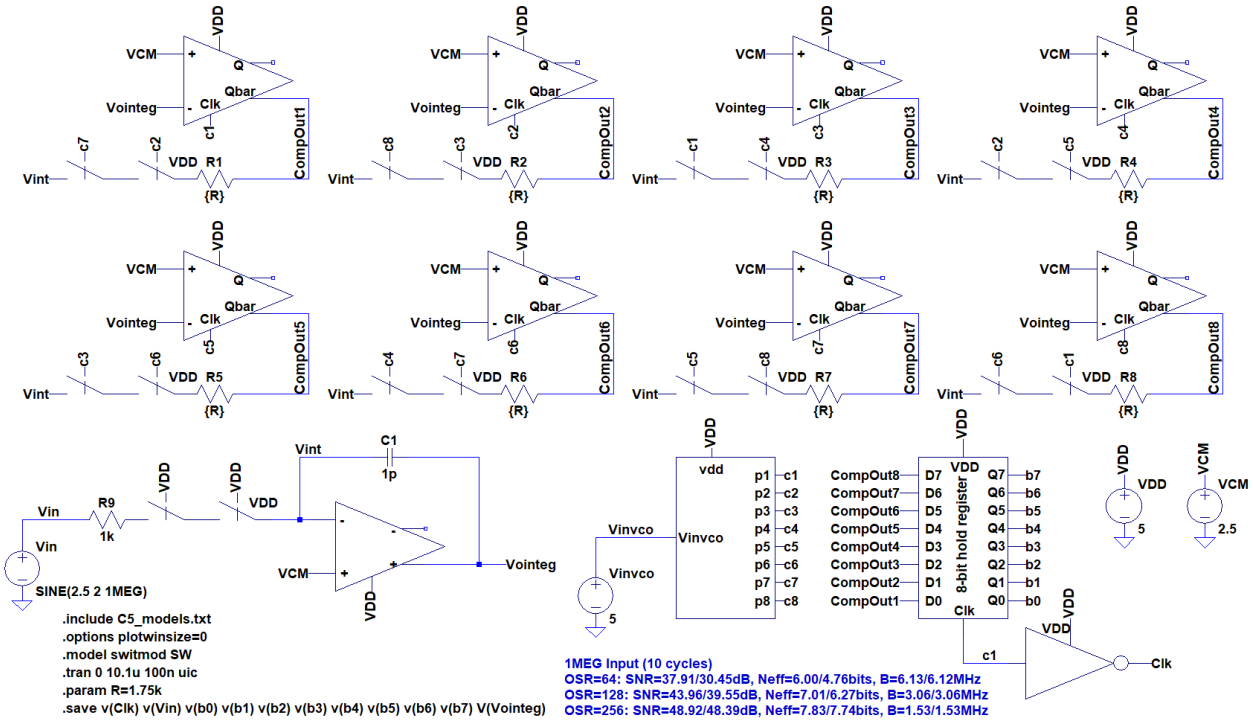
Bandwidth (B) Calculation

The bandwidth (B) can be calculated below. All of the MATLAB output values of $f_{s,new}$ and B summarized throughout this report follow this equation.

$$B = \frac{fs,new}{2K}$$ where $f_{s,new} = K_{path}*f_s$ ($K_{path} = 8$) and K is the OSR (here 64, 128, or 256)

First-Order Topology

The schematic of the first-order topology is shown below and consists of, among other things, eight comparators (8-paths) each feeding back to a single integrator through switches that were implemented using transmission gates. The operation of this topology follows the water-bucket-cup analogy in which the comparator (bucket) is making a decision every clock cycle (implemented by the clock generator) whether the water in the bucket is past a reference line (VCM) and outputs an according fed back signal for the integrator to remove a cup of water from the bucket or not.



"Dane_8D1S_1st_Order_1MHz"

Components

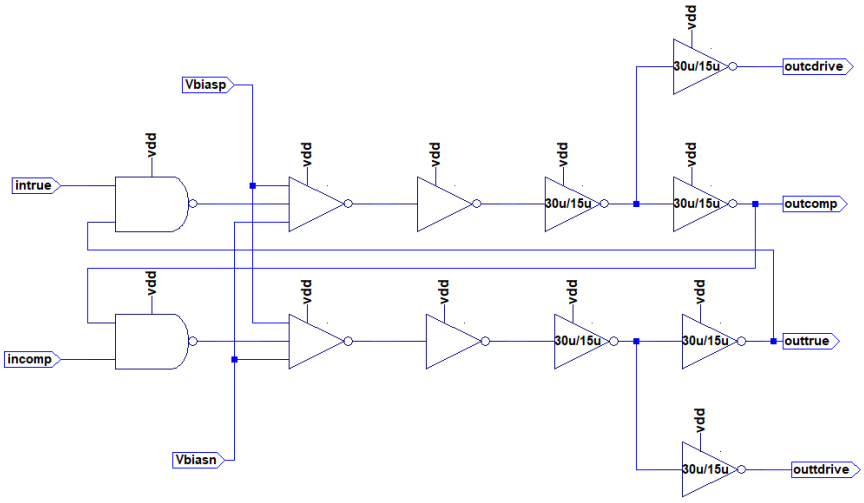
Clock Generator

In order to properly clock the comparators and switches in the ADC, a clock generator needed to be designed to generate eight clock signals with equally-spaced rising/falling edges. This was

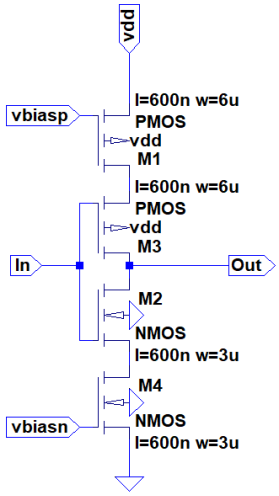
done using a voltage-controlled ring oscillator such that a higher input voltage would generate clock signals with higher frequencies.

The schematic of a single delay stage in clock generator is shown below. In order to achieve closer to a 50% duty cycles for the clocks, the inputs of the NAND gates were chosen to be fed back from the "outtrue" and "outcomp" nodes since the signal pulses on these nodes have faster edges (so the clocks are more decisive) rather than the nodes on the inputs of the first 30u/15u inverters.

The first inverters in the path were also designed to be current starved as shown below such that the bias voltages "Vbiasp" and "Vbiasn" can be used to adjust the delay of the clocks.

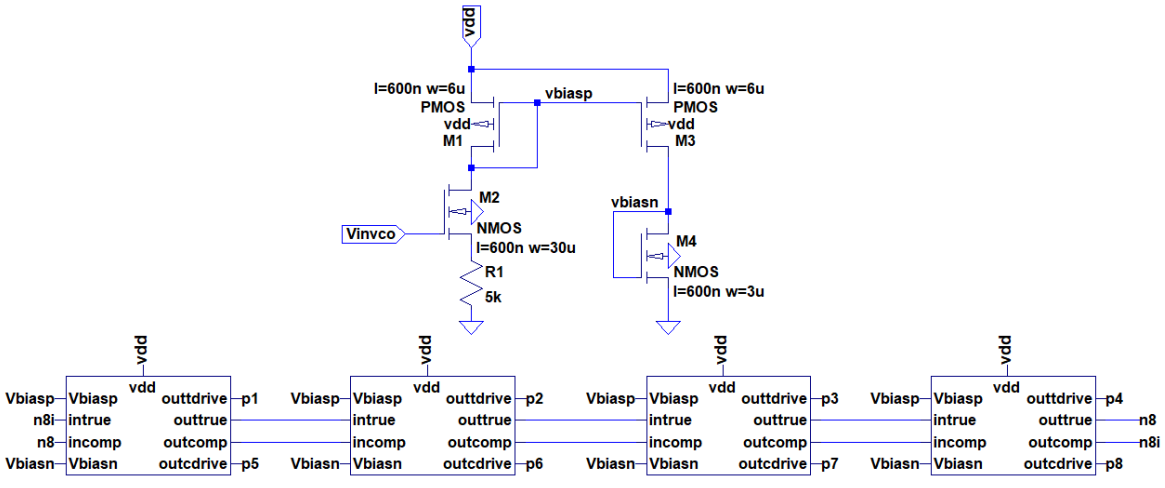


"_Dane_Delay"



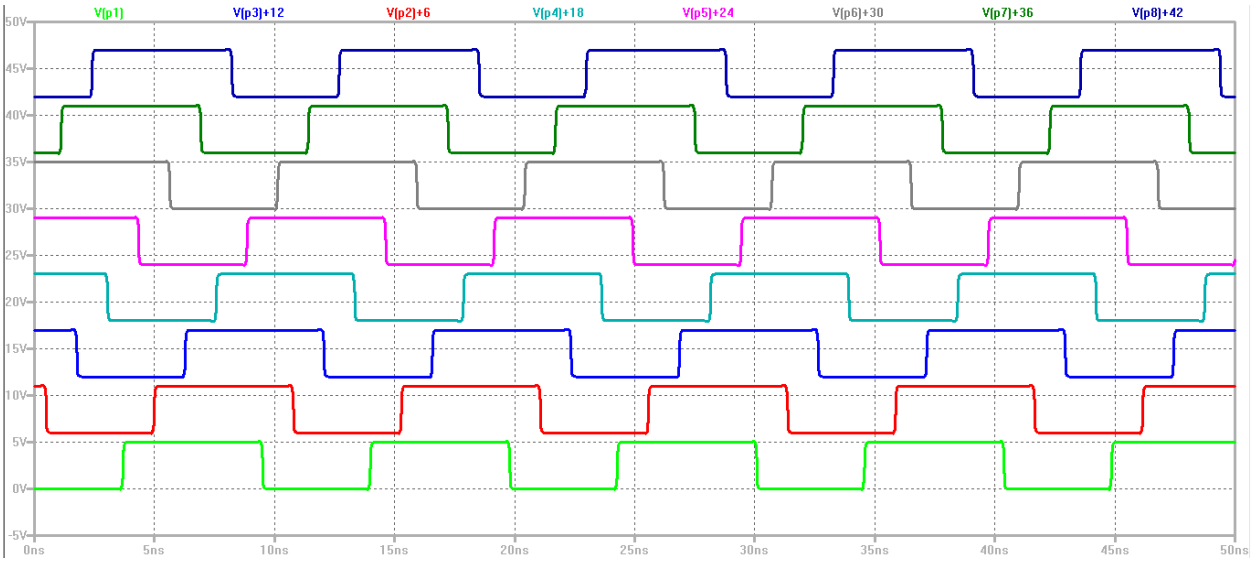
"_Dane_Current_Starved"

The final clock generator is shown below. The bias resistor can be used to adjust the speed of the clocks. The bias resistor was set to 5k which results in the clocks having a period of 10.29ns (frequency of 97.19MHz) for Vinvco = 5V.



"_Dane_Clk_Gen"

The eight equally-spaced clock signals are shown below. Each clock has a period of 10.29ns (frequency of 97.19MHz) for Vinvco = 5V.



"_Dane_Clk_Gen_Sim"

The simulation below shows how the clock frequency changes with differing voltages of Vinvco. The clocks range from a period of 10.29ns (frequency of 97.19MHz) for Vinvco = 5V to a period of 34.88ns (frequency of 28.67MHz) for Vinvco=962.25mV.

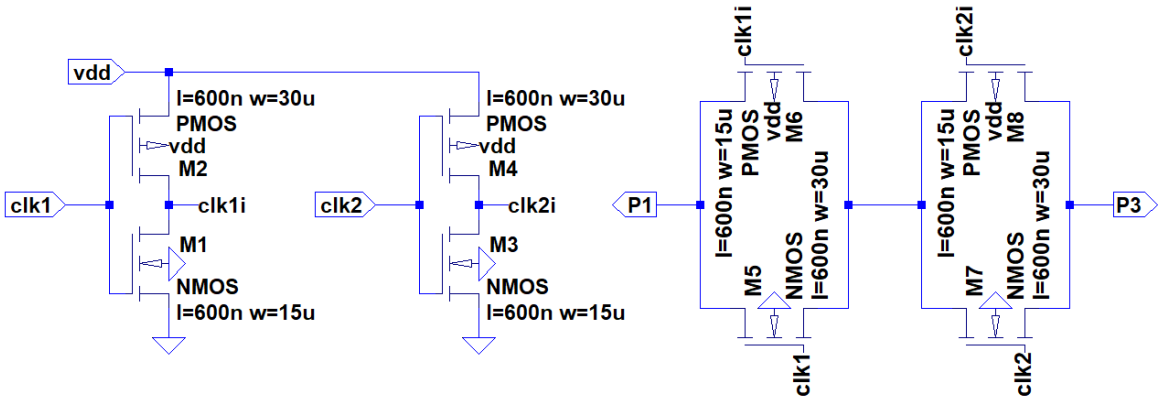
"_Dane_Clk_Gen_Ramp_Sim"

Switches (Transmission Gates)

The switches in this design were implemented using transmission gates (TG's) such that the signal from the output of the comparator would only feedback to the input of the integrator when both switches were closed. Doing this ensured only a "chunk of charge" was fed back for a short

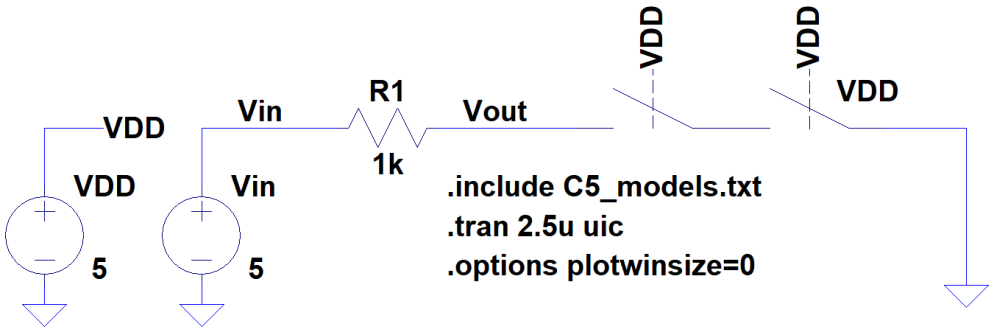
period of time when both switches were closed once the comparator had enough time to make a decision.

The schematic below shows the implementation of the TG switches. Inverters were utilized to generate complementary (non-overlapping) clock signals such that the NMOS/PMOS in the TG would be on/off at the same time. Furthermore, both NMOS and PMOS are used in the TG due to the fact that PMOS can pass a logic high well and NMOS can pass a logic low well.

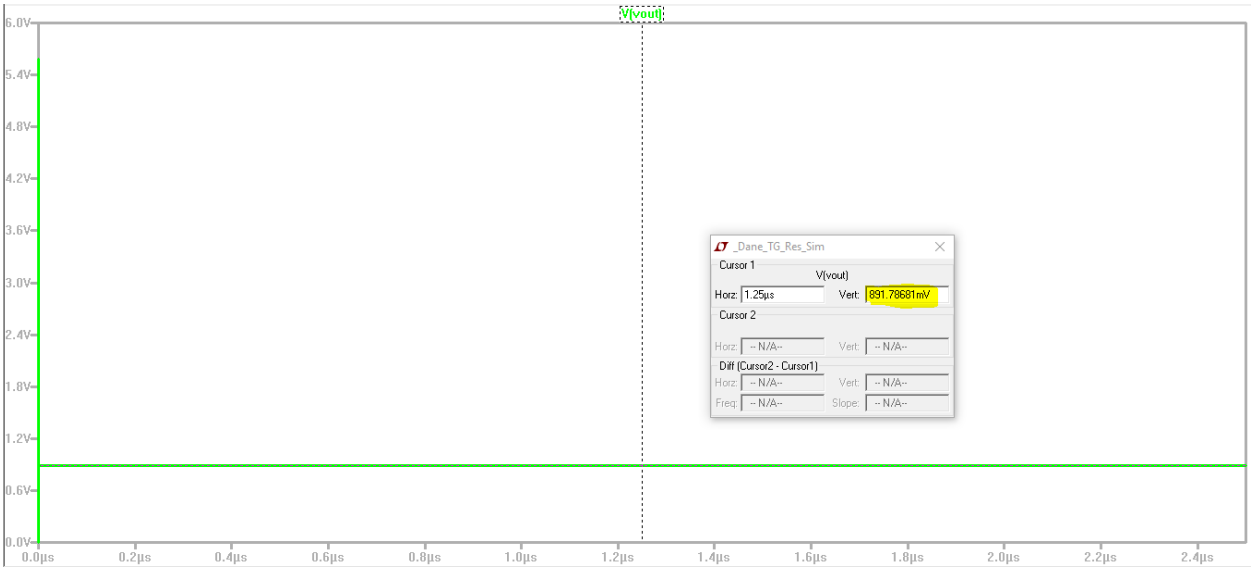


"_Dane_TG"

In order to properly size the TG's, the resistance of the TG's was calculated using the voltage divider schematic/simulation below. The TG was sized such that the resistance of the TG would be small compared to its series resistor in the feedback path.



"_Dane_TG_Res_Sim"



"_Dane_TG_Res_Sim"

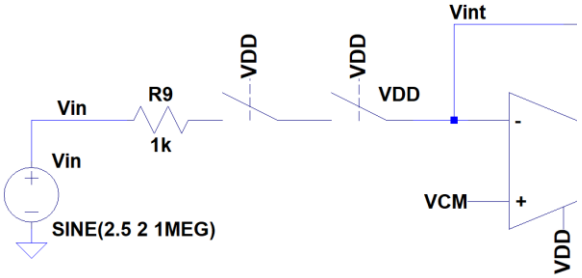
Using the voltage divider above with an output voltage of 891.79mV, the resistance of the TG was calculated to be 217.075 Ω as shown below.

$$Vout = \frac{TG_{Res}}{1k+TG_{Res}} * Vin \rightarrow \frac{Vout}{Vin} = \frac{TG_{Res}}{1k+TG_{Res}} \rightarrow \frac{Vout}{Vin} * (1k + TG_{Res}) = TG_{Res}$$

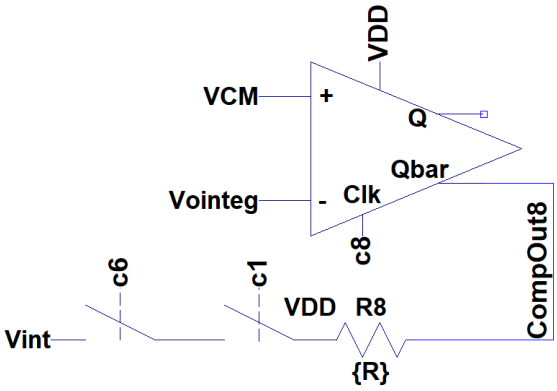
$$\rightarrow \frac{Vout}{Vin} * 1k = TG_{Res} * (1 - \frac{Vout}{Vin}) \rightarrow TG_{Res} = \frac{\frac{Vout}{Vin}*1k}{1-\frac{Vout}{Vin}} \rightarrow TG_{Res} = \frac{\frac{891.79mV}{5V}*1k}{1-\frac{891.79mV}{5V}}$$

$$TG_{Res} = 217.075\ \Omega$$

In order to match the resistance in the feedback path to the resistance in the forward path, a "dummy" switch (shown below) with VDD on its clock inputs (such that the switch is always on) was placed in the forward path. This was done rather than using a resistor equal to the resistance of the TG in order to compensate for changes in the resistance of the TG due to process and temperature variations.

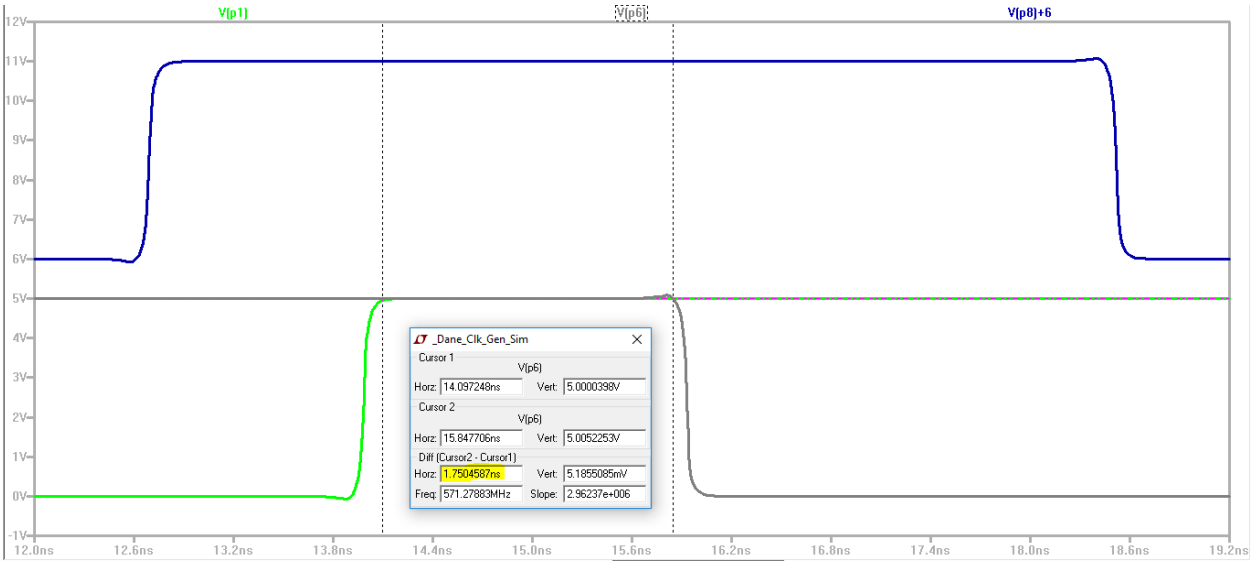


Dummy Switch



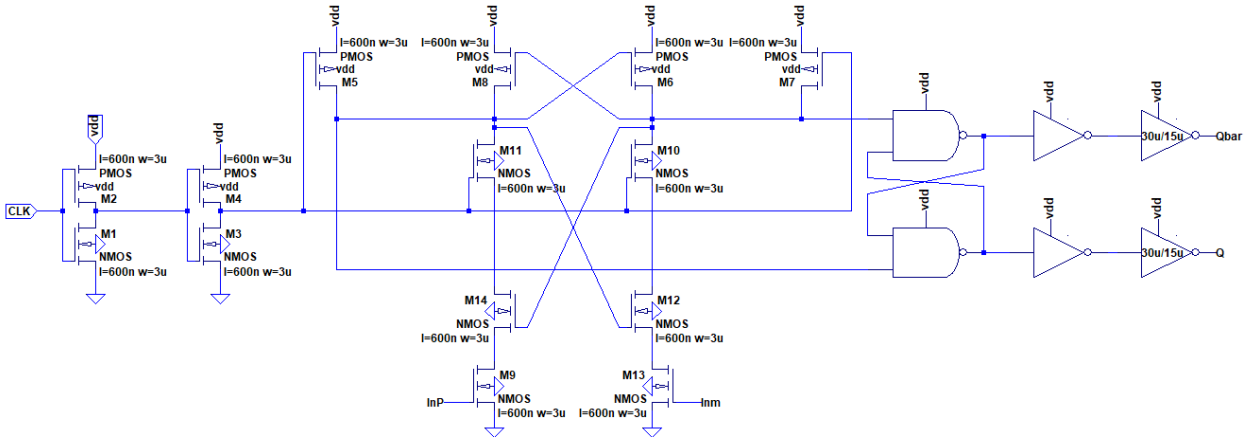
Comparator w/ Feedback

As mentioned earlier, the switches in the feedback path (shown above) were only both closed for a short amount of time. This is shown in the simulation below where the comparator is clocked at c8 and, in the time this clock is high, c1 and c6 overlap for only 1.75ns in order to feedback the comparator output signal after the comparator has had enough time to make a correct decision.



Comparator

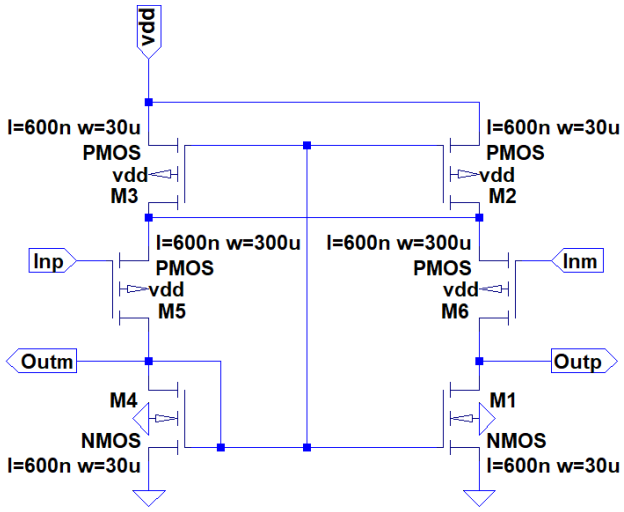
The schematic of the comparator designed for and implemented in each presented topology is shown below.



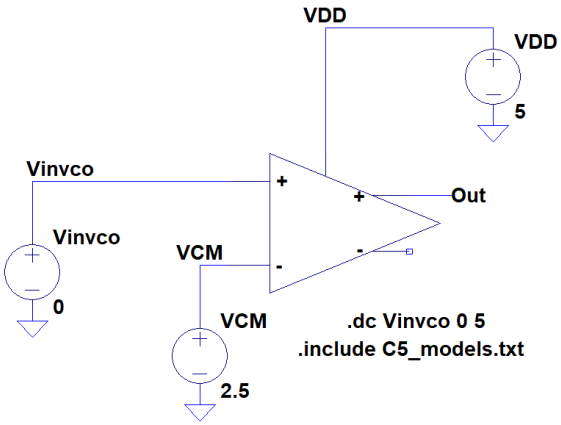
"_Dane_Clk_Comparator"

Integrator

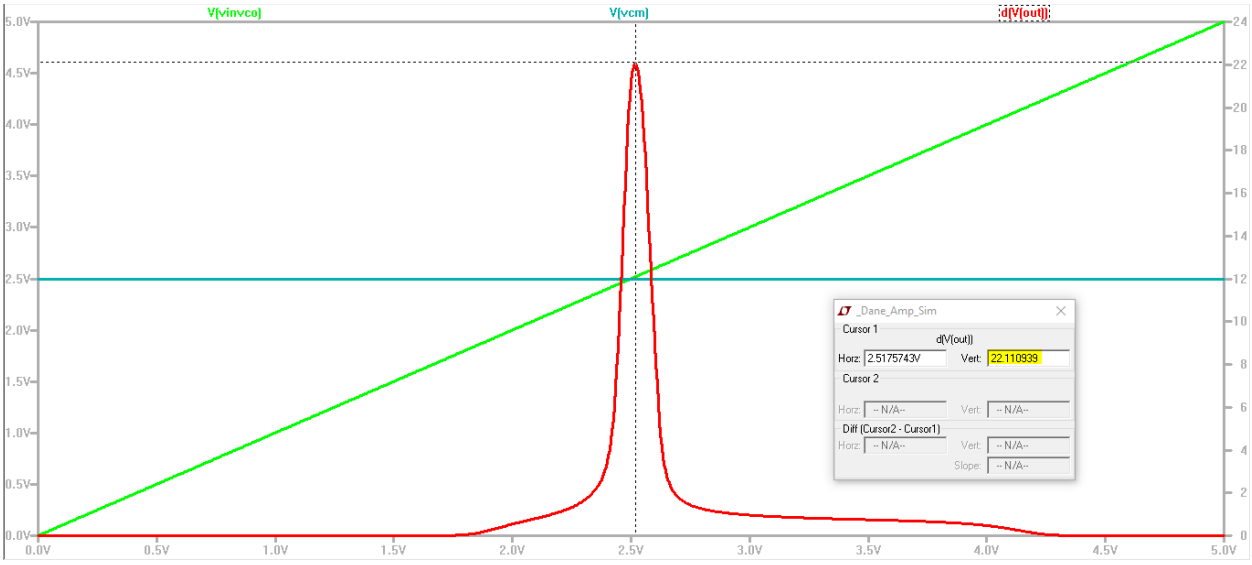
The integrator in this design was implemented using an operational amplifier (opamp) shown below. The simulation of the opamp is shown below, and the opamp was found to have a gain of 22.11.



“_Dane_Amp”



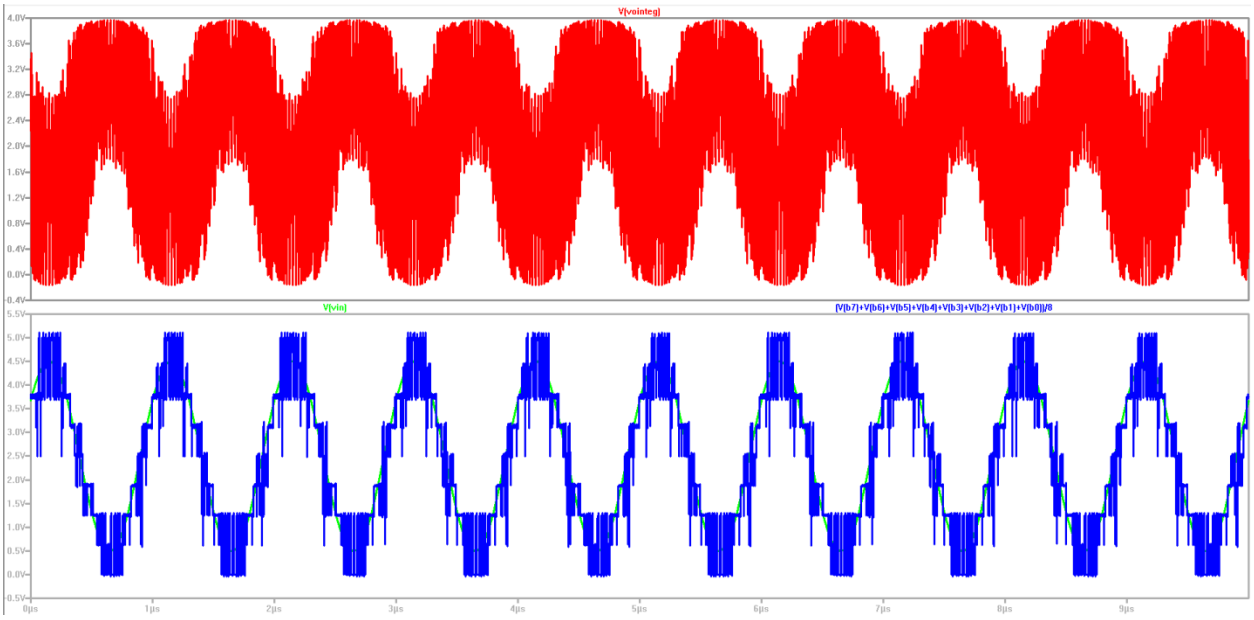
“_Dane_Amp_Sim”



“_Dane_Amp_Sim”

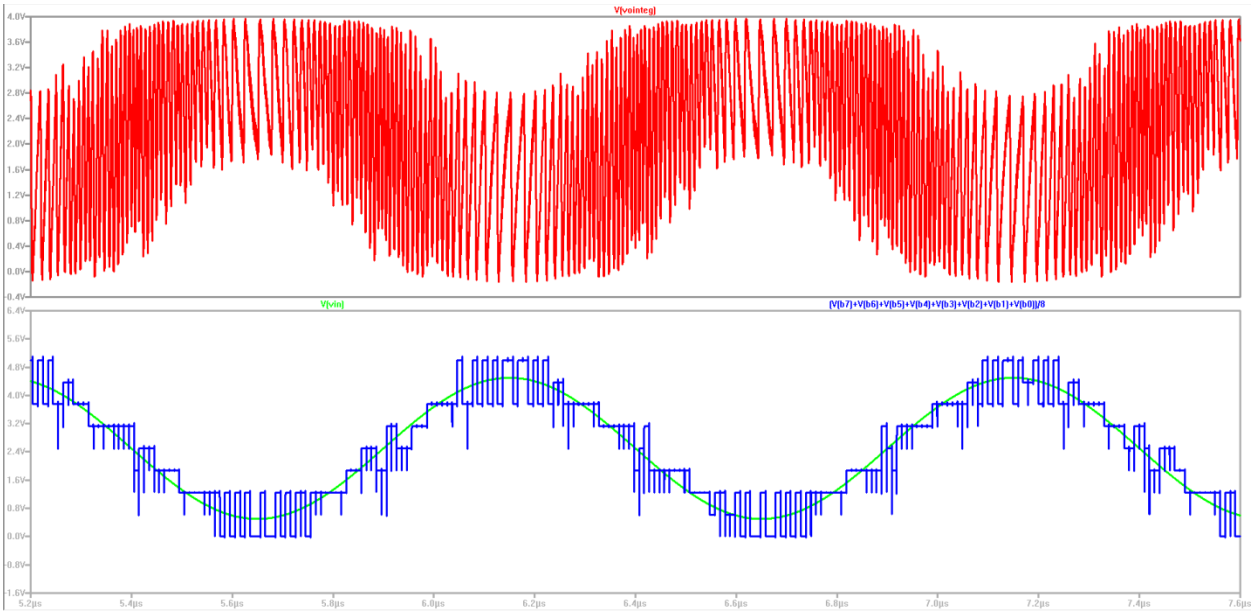
First-Order Topology Results

The simulation plots of the first-order topology with an input signal frequency of 1MHz are shown below.



“Dane_8D1S_1st_Order_1MHz”

Below is a zoomed in version of the simulation above.



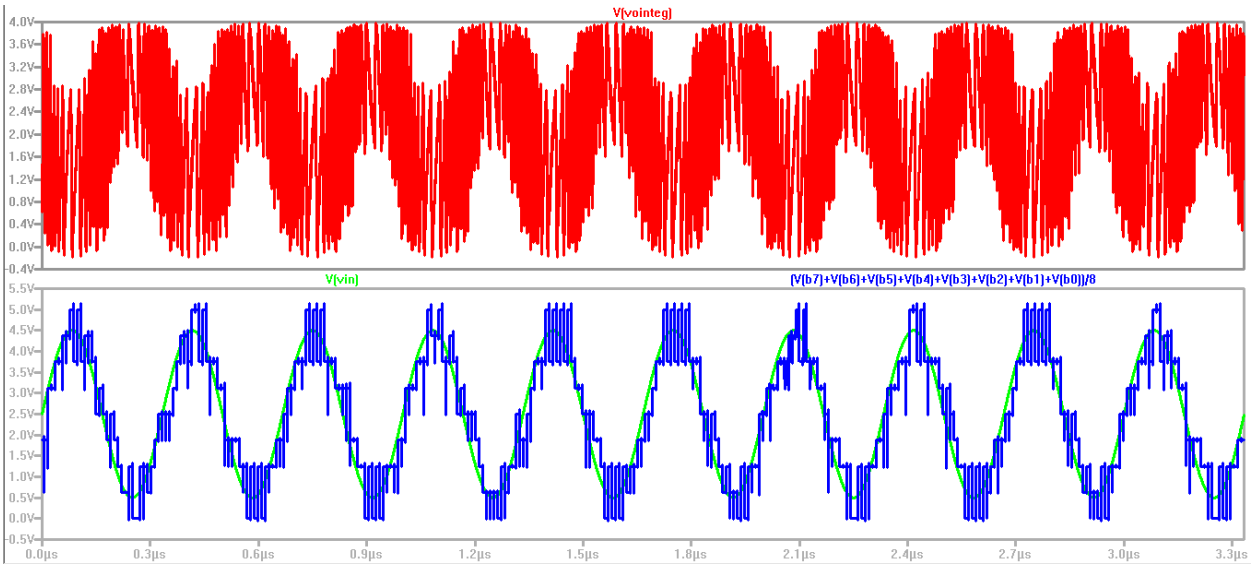
The MATLAB output results for with an input signal frequency of 1MHz and OSR’s of 64, 128, and 256 are shown below.

```
Kpath = 8 OSR = 64
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 37.91, Neff = 6.00, B = 6.13 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 30.45, Neff = 4.76, B = 6.12 MHz>>
```

```
Kpath = 8 OSR = 128
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 43.96, Neff = 7.01, B = 3.06 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 39.55, Neff = 6.27, B = 3.06 MHz>>
```

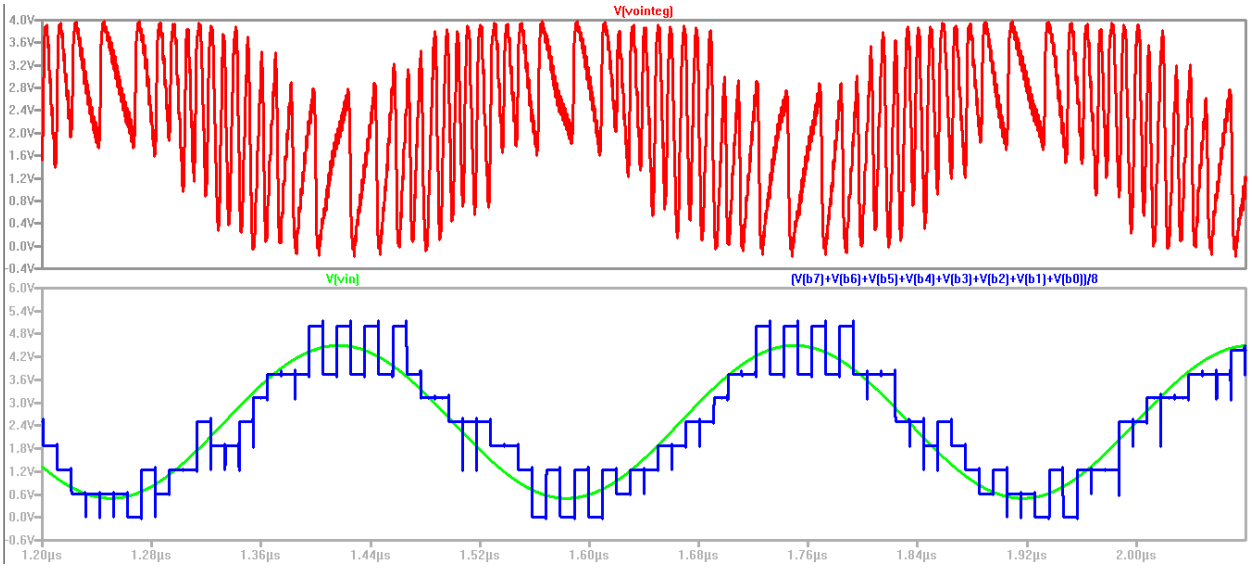
```
Kpath = 8 OSR = 256
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 48.92, Neff = 7.83, B = 1.53 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 48.39, Neff = 7.74, B = 1.53 MHz>>
```

The simulation plots of the first-order topology with an input signal frequency of 3MHz are shown below.



"Dane_8D1S_1st_Order_3MHz"

Below is a zoomed in version of the simulation above.



The MATLAB output results for with an input signal frequency of 3MHz and OSR's of 64, 128, and 256 are shown below.

```
Kpath = 8 OSR = 64
For 1-bit output (serial mode) at Kpath*fs = 785 MHz
SNR = 39.66, Neff = 6.29, B = 6.13 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 33.26, Neff = 5.23, B = 6.12 MHz>>
```

```
Kpath = 8 OSR = 128
For 1-bit output (serial mode) at Kpath*fs = 785 MHz
SNR = 44.89, Neff = 7.16, B = 3.07 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 42.21, Neff = 6.72, B = 3.06 MHz>>
```

```
Kpath = 8 OSR = 256
For 1-bit output (serial mode) at Kpath*fs = 785 MHz
SNR = 48.13, Neff = 7.70, B = 1.53 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 45.95, Neff = 7.34, B = 1.53 MHz>>
```

The MATLAB output results for an input signal frequency of both 1MHz and 3MHz and OSR's of 64, 128, and 256 summarized in the tables below.

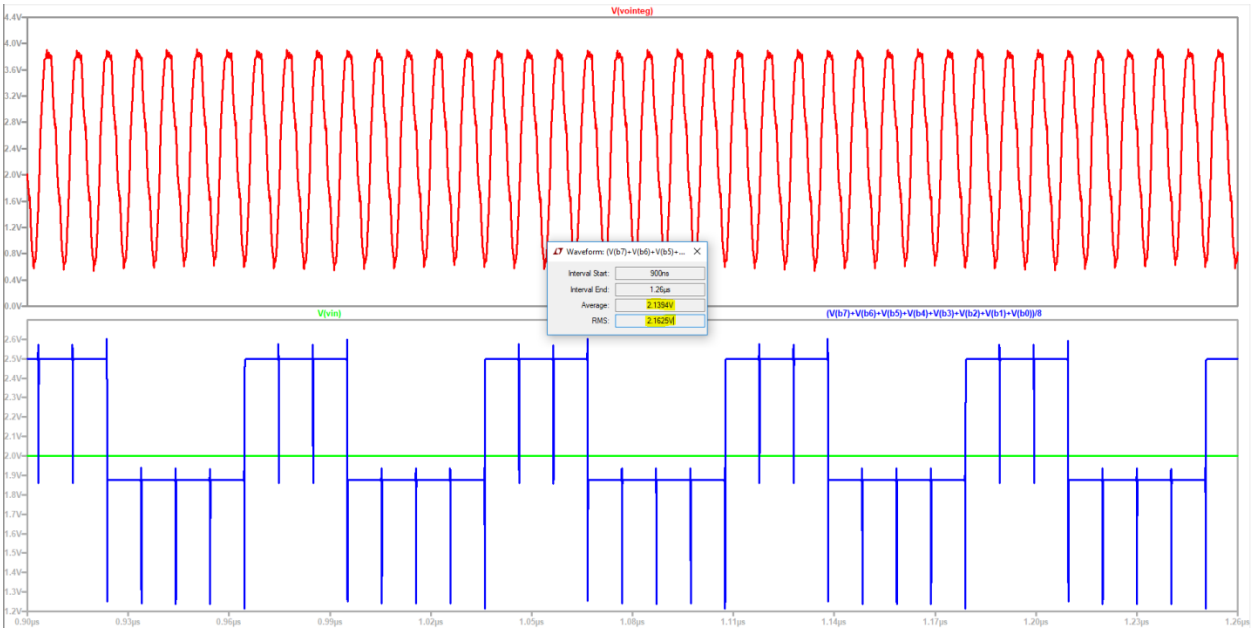
"Dane_8D1S_1st_Order_1MHz" (10 Cycles)

	Serial 64	Parallel 64	Serial 128	Parallel 128	Serial 256	Parallel 256
$f_{s,new}$ (MHz)	784	98	784	98	784	98
SNR (dB)	37.91	30.45	43.96	39.55	48.92	48.39
N_{eff} (bits)	6.00	4.76	7.01	6.27	7.83	7.74
Bandwidth (MHz)	6.13	6.12	3.06	3.06	1.53	1.53

"Dane_8D1S_1st_Order_3MHz" (10 Cycles)

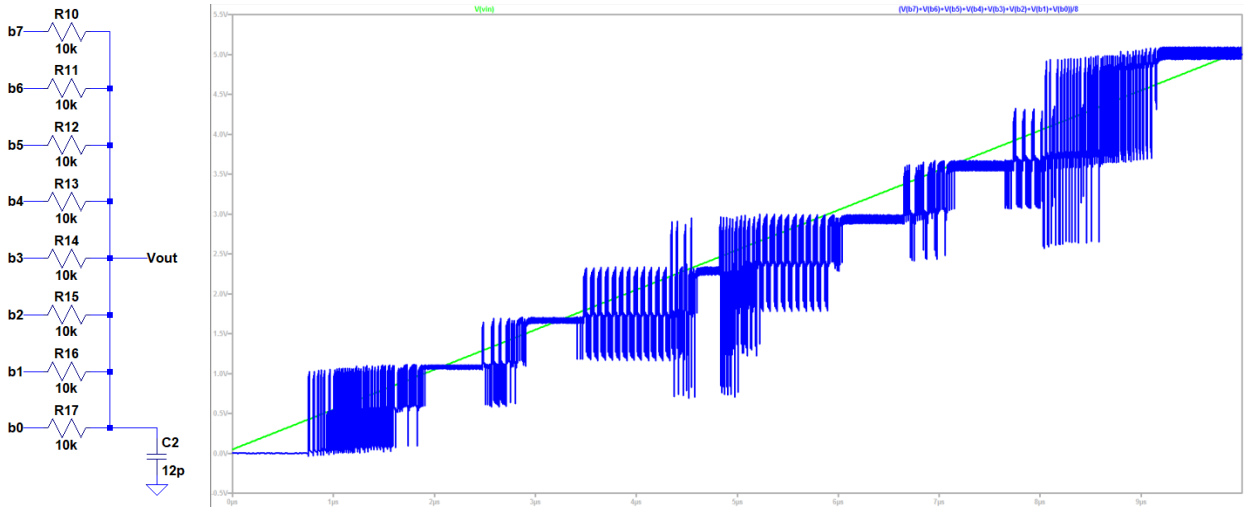
	Serial 64	Parallel 64	Serial 128	Parallel 128	Serial 256	Parallel 256
$f_{s,new}$ (MHz)	785	98	785	98	785	98
SNR (dB)	39.66	33.26	44.89	42.21	48.13	45.95
N_{eff} (bits)	6.29	5.23	7.16	6.72	7.70	7.34
Bandwidth (MHz)	6.13	6.12	3.07	3.06	1.53	1.53

The simulation plot of the first-order topology with a DC input signal is shown below. Note that the output averages to 2.14V which is approximately equal to the DC input signal of 2V.



"Dane_8D1S_1st_Order_DC"

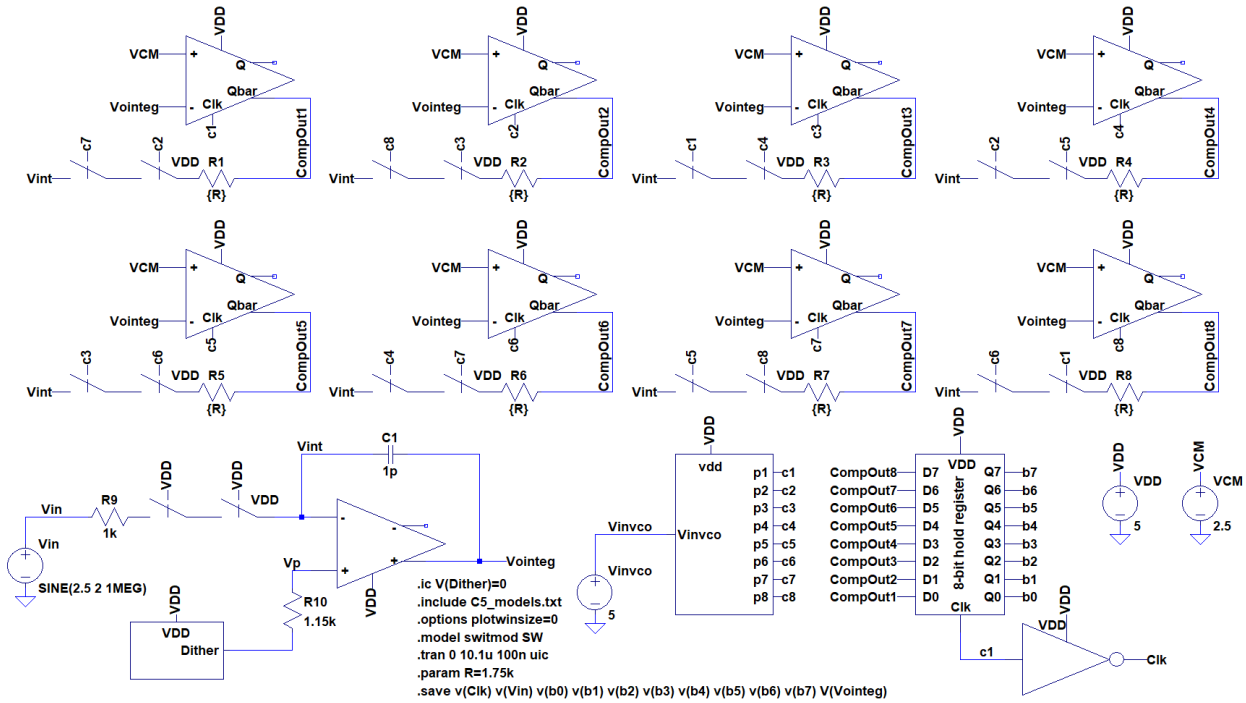
The simulation plot of the first-order topology with a slow-moving ramp input signal is shown below. An RC filter (shown below) was used to filter the output. Note the significant presence of dead zones.



“Dane_8D1S_2nd_Order_Ramp”

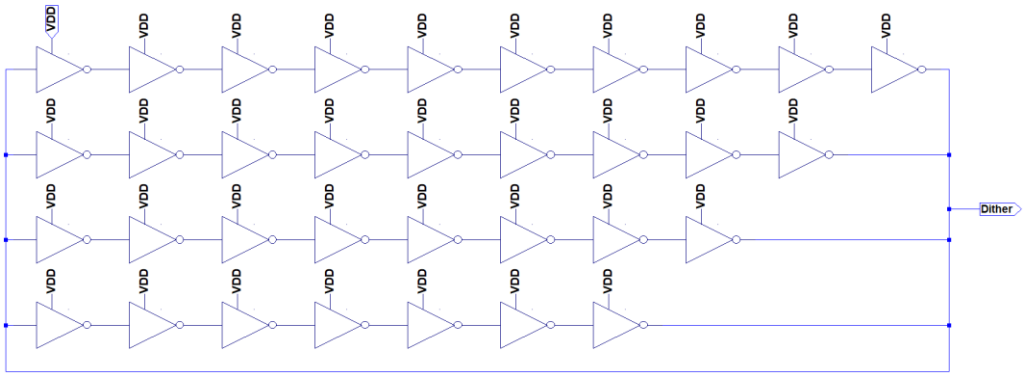
First-Order w/ Dither Topology

In order to improve the performance of the first-order topology for slow-moving signals, a dither was added to the first-order topology. The dither acts as a kind of noise in order to keep the input busy and, therefore, adhere to Bennett’s criteria. The schematic of the first-order with dither topology is shown below.



“Dane_8D1S_1st_Order_Dither_1MHz”

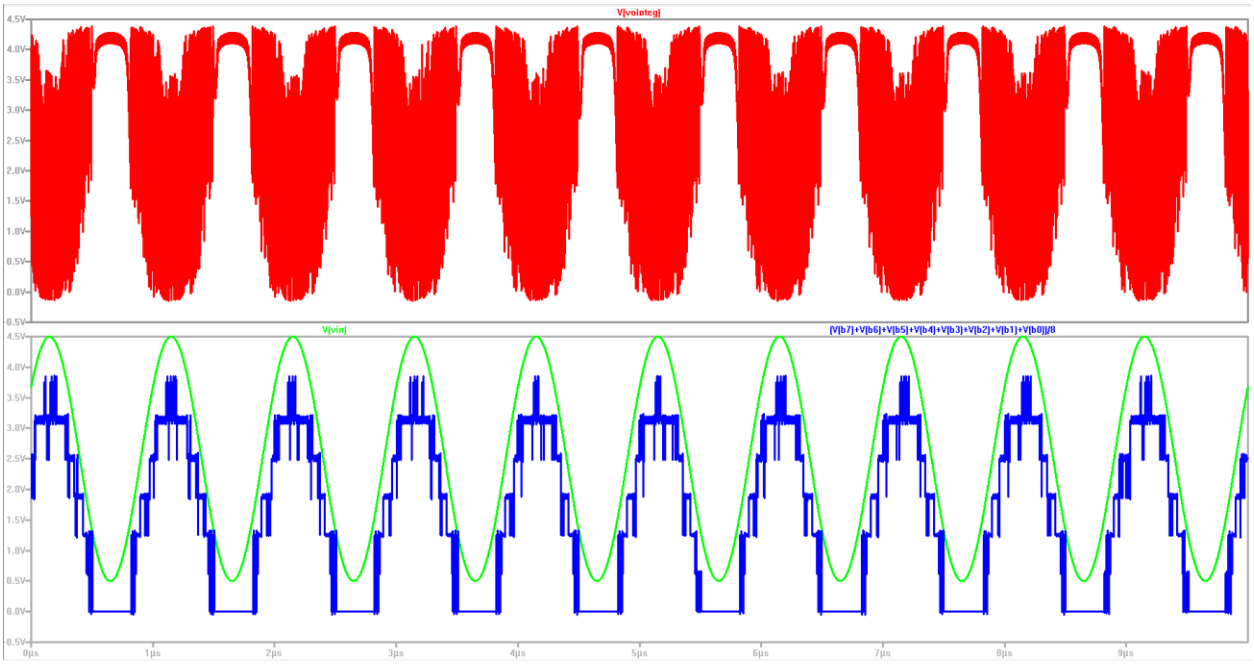
The symbol schematic of the dither is shown below. The outputs of each row of inverters asynchronously contradict one another and cause the dither output to oscillate at abnormal voltage levels other than VDD and GND. This is seen in the simulation plot of the dither output voltage shown below.



"_Dane_Dither"

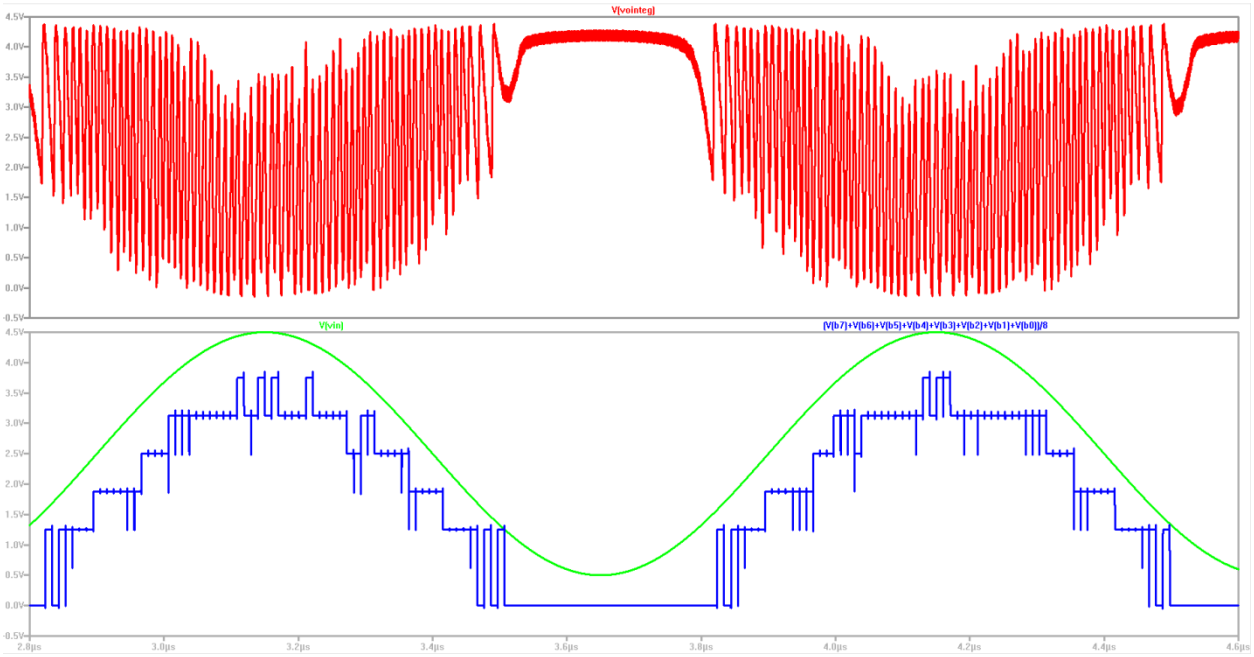
"Dane_8D1S_1st_Order_Dither_1MHz"

The simulation plots of the first-order with dither topology with an input signal frequency of 1MHz are shown below.



“Dane_8D1S_1st_Order_Dither_1MHz”

Below is a zoomed in version of the simulation above.



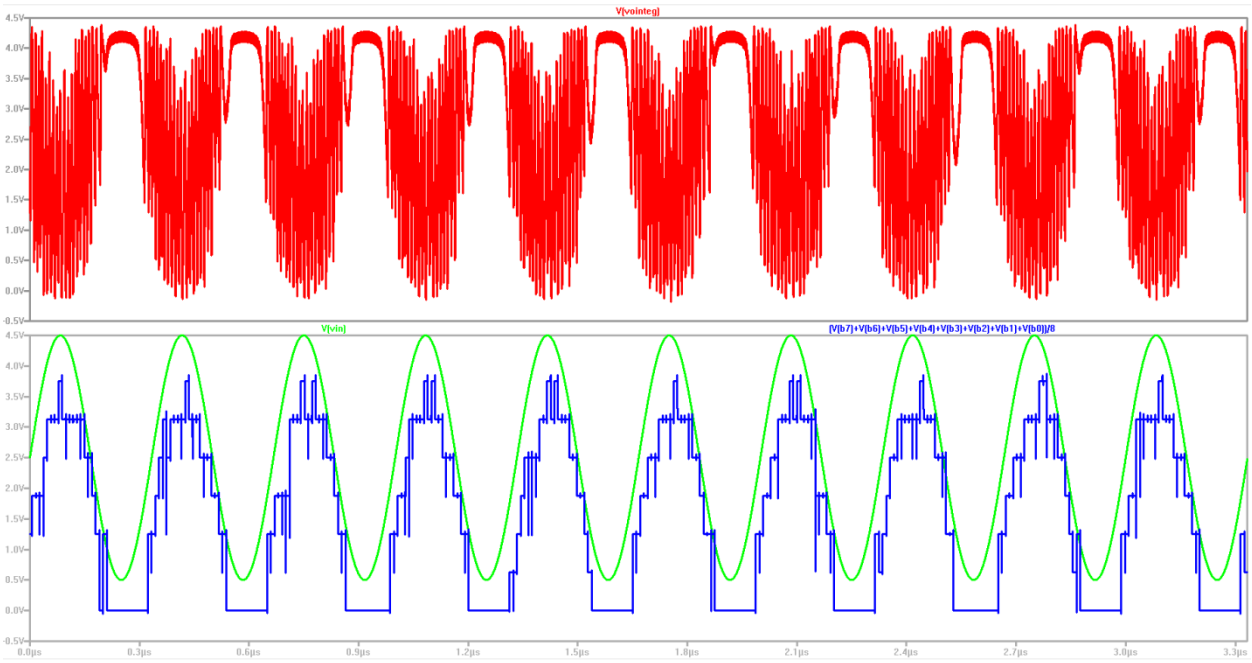
The MATLAB output results for with an input signal frequency of 1MHz and OSR’s of 64, 128, and 256 are shown below.

```
Kpath = 8 OSR = 64
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 23.95, Neff = 3.68, B = 6.13 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 24.04, Neff = 3.70, B = 6.12 MHz>>
```

```
Kpath = 8 OSR = 128
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 49.43, Neff = 7.92, B = 3.06 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 45.79, Neff = 7.31, B = 3.06 MHz>>
```

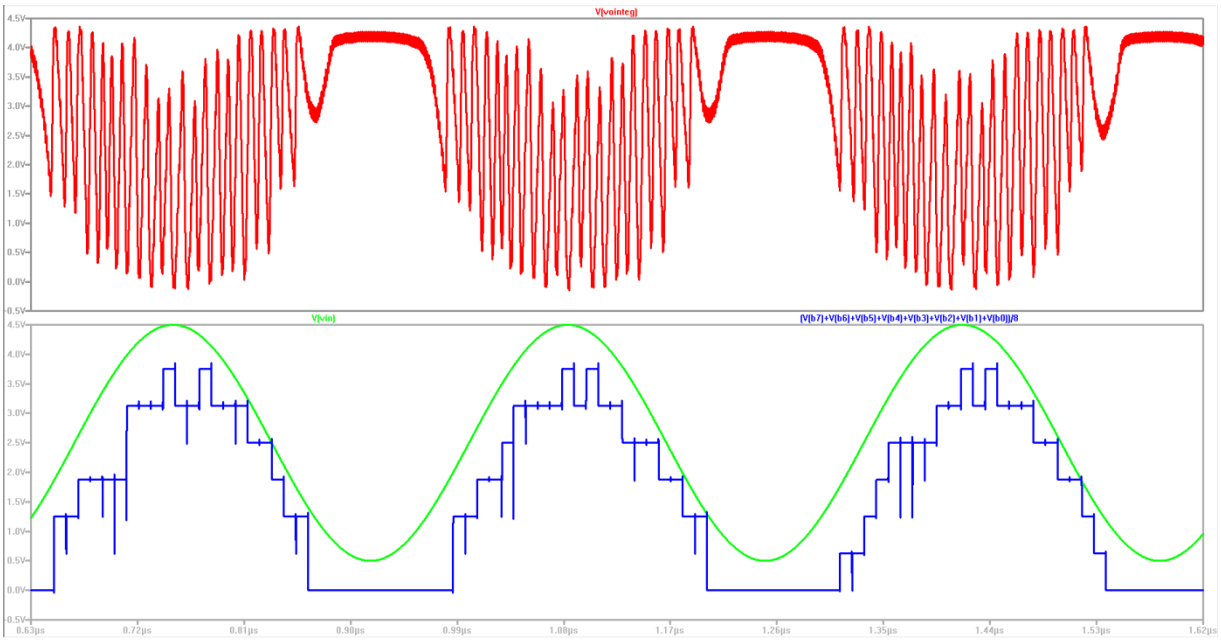
```
Kpath = 8 OSR = 256
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 52.47, Neff = 8.42, B = 1.53 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 52.58, Neff = 8.44, B = 1.53 MHz>>
```

The simulation plots of the first-order with dither topology with an input signal frequency of 3MHz are shown below.



“Dane_8D1S_1st_Order_Dither_3MHz”

Below is a zoomed in version of the simulation above.



The MATLAB output results for with an input signal frequency of 3MHz and OSR's of 64, 128, and 256 are shown below.

```
Kpath = 8 OSR = 64
For 1-bit output (serial mode) at Kpath*fs = 785 MHz
SNR = 33.37, Neff = 5.25, B = 6.13 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 27.38, Neff = 4.25, B = 6.12 MHz>>
```

```
Kpath = 8 OSR = 128
For 1-bit output (serial mode) at Kpath*fs = 785 MHz
SNR = 34.71, Neff = 5.47, B = 3.07 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 34.59, Neff = 5.45, B = 3.06 MHz>>
```

```
Kpath = 8 OSR = 256
For 1-bit output (serial mode) at Kpath*fs = 785 MHz
SNR = 36.07, Neff = 5.70, B = 1.53 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 36.02, Neff = 5.69, B = 1.53 MHz>>
```

The MATLAB output results for an input signal frequency of both 1MHz and 3MHz and OSR's of 64, 128, and 256 summarized in the tables below.

"Dane_8D1S_1st_Order_Dither_1MHz" (10 Cycles)

	Serial 64	Parallel 64	Serial 128	Parallel 128	Serial 256	Parallel 256
$f_{s,new}$ (MHz)	784	98	784	98	784	98
SNR (dB)	23.95	24.04	49.43	45.79	52.58	52.47
N_{eff} (bits)	3.68	3.70	7.92	7.31	8.44	8.42
Bandwidth (MHz)	6.13	6.12	3.06	3.06	1.53	1.51

"Dane_8D1S_1st_Order_Dither_3MHz" (10 Cycles)

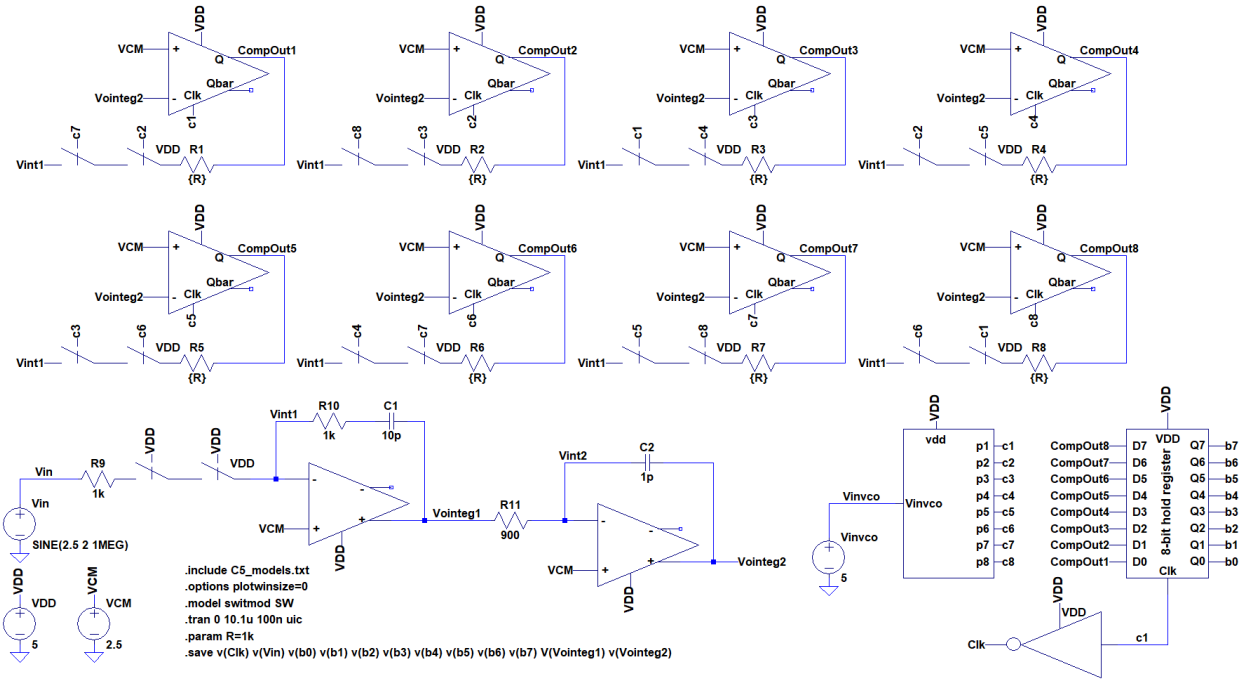
	Serial 64	Parallel 64	Serial 128	Parallel 128	Serial 256	Parallel 256
$f_{s,new}$ (MHz)	785	98	785	98	785	98
SNR (dB)	33.37	27.38	34.71	34.59	36.07	36.02
N_{eff} (bits)	5.25	4.25	5.47	5.45	5.70	5.69
Bandwidth (MHz)	6.13	6.12	3.07	3.06	1.53	1.53

Second-Order Topology

The second order-topology was implemented with a second integrator. All other components except for the opamp (as well as resistor/capacitor values) are identical to the first-order.

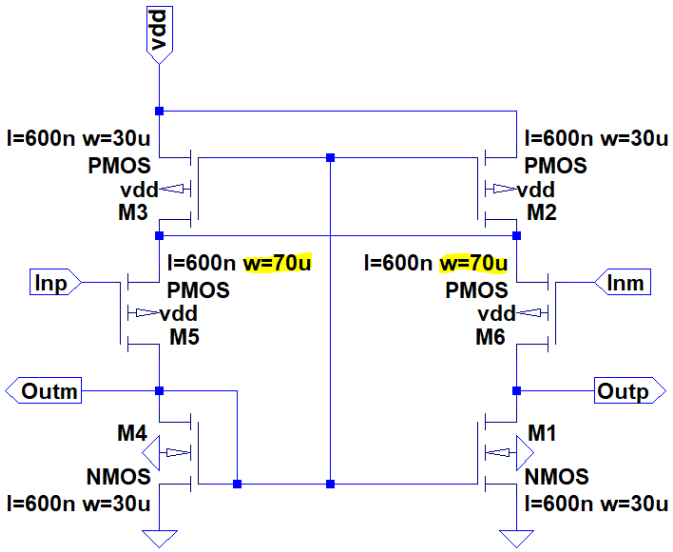
In order to design a second-order topology that performed better than the first-order, the gain of the opamp was decreased to prevent the outputs of the integrators from saturating. In addition, much experimentation was doing with the values of the resistors and capacitors in order to achieve optimal performance.

The schematic of the second-order topology is shown below.

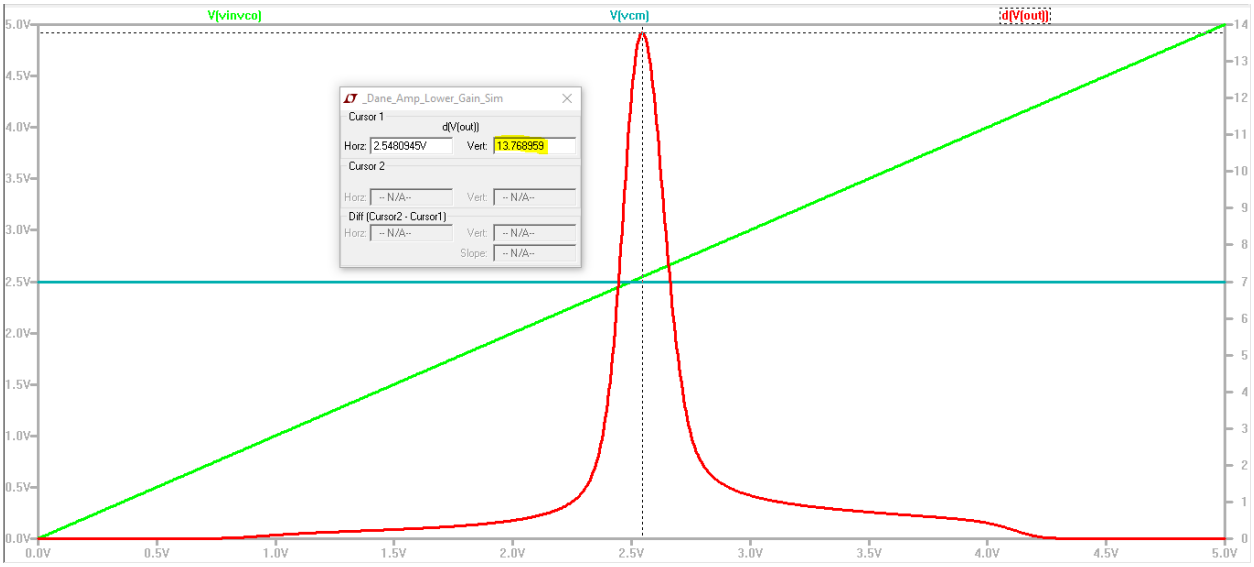


“Dane_8D1S_2nd_Order_1MHz”

The gain of the opamp was reduced by dropping the width of the input PMOS’s (M5 & M6) down from 300u to 70u as shown in the schematic of the opamp below. This resulted in lowering the gain of the opamp from 22.11 to 13.77 (as shown in the simulation below) and yielded optimal performance of the second-order topology.

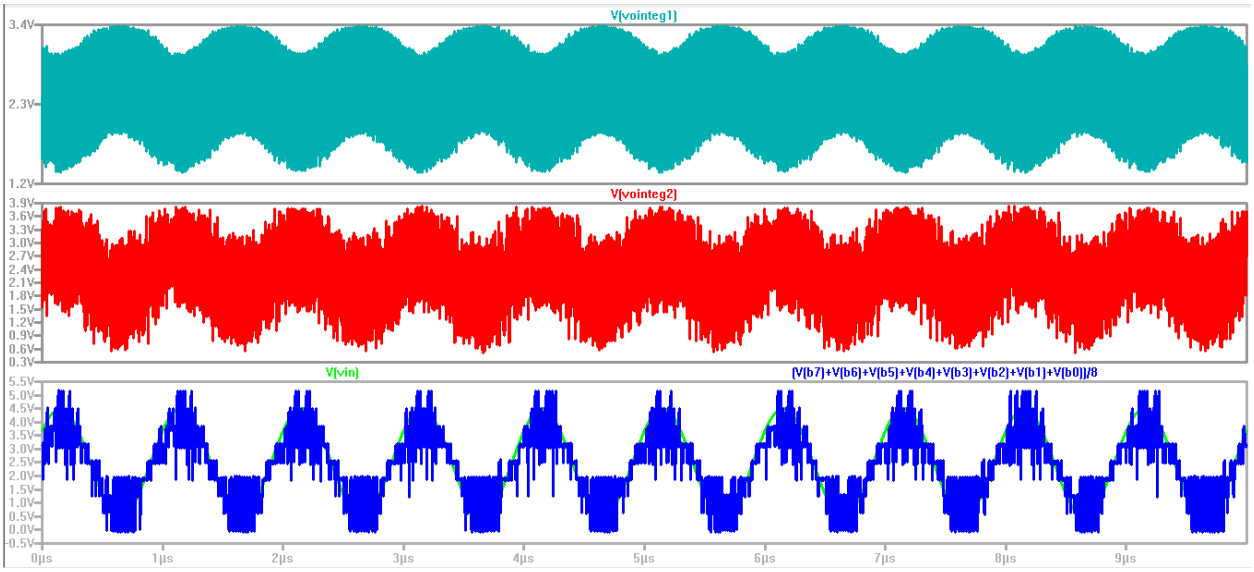


“_Dane_Amp_Lower_Gain”



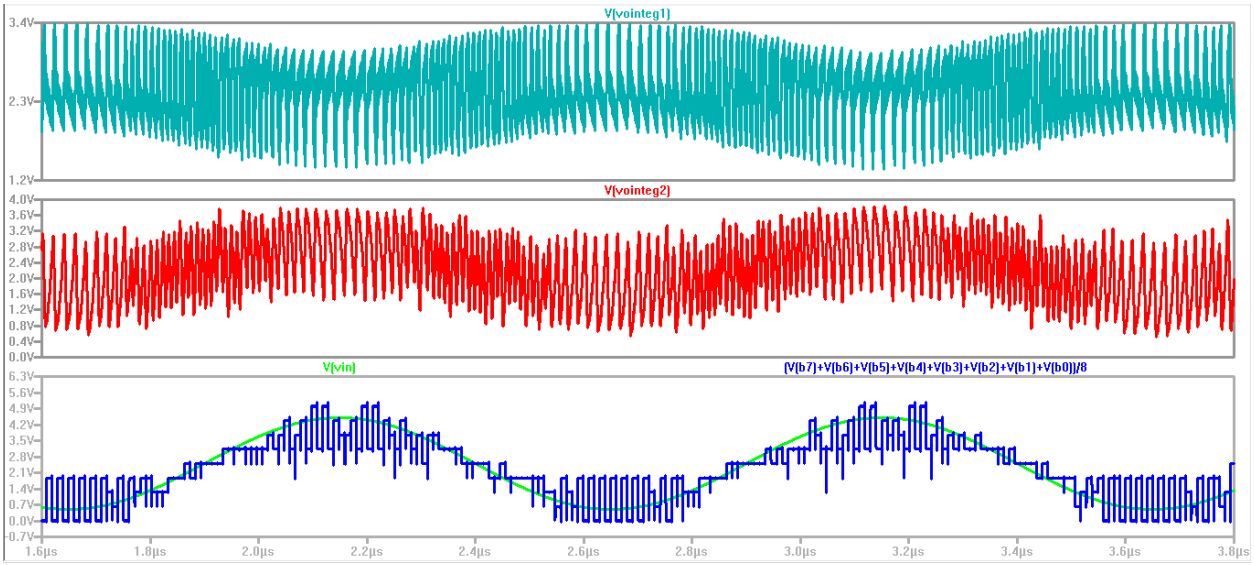
"_Dane_Amp_Lower_Gain_Sim"

The simulation plots of the second-order topology with an input signal frequency of 1MHz are shown below.



"Dane_8D1S_2nd_Order_1MHz"

Below is a zoomed in version of the simulation above.



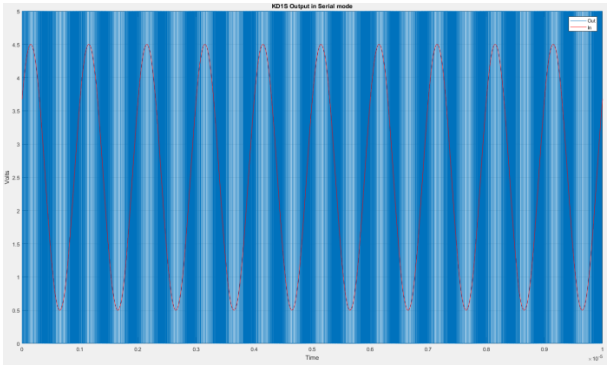
The MATLAB output results for with an input signal frequency of 1MHz and OSR's of 64, 128, and 256 are shown below.

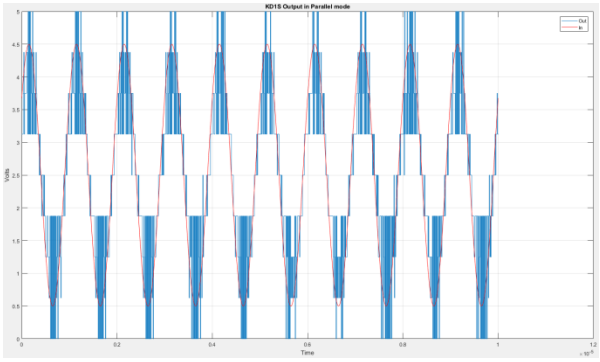
```
Kpath = 8 OSR = 64
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 41.90, Neff = 6.66, B = 6.13 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 26.26, Neff = 4.07, B = 6.12 MHz>>
```

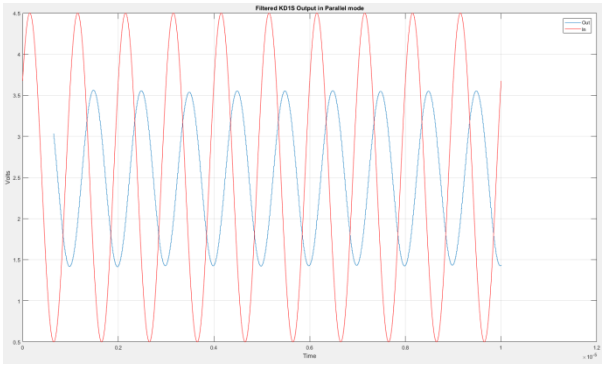
```
Kpath = 8 OSR = 128
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 47.98, Neff = 7.68, B = 3.06 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 38.87, Neff = 6.16, B = 3.06 MHz>>
```

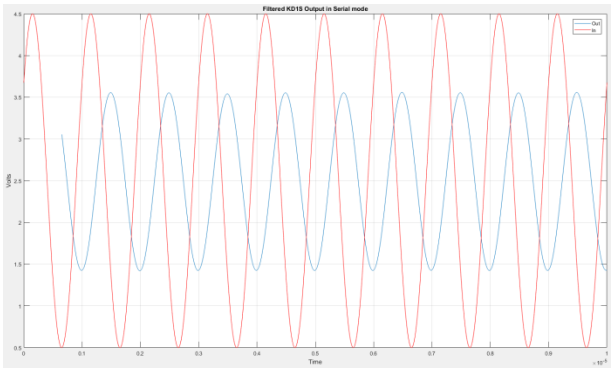
```
Kpath = 8 OSR = 256
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 52.38, Neff = 8.41, B = 1.53 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 48.75, Neff = 7.80, B = 1.53 MHz>>
```

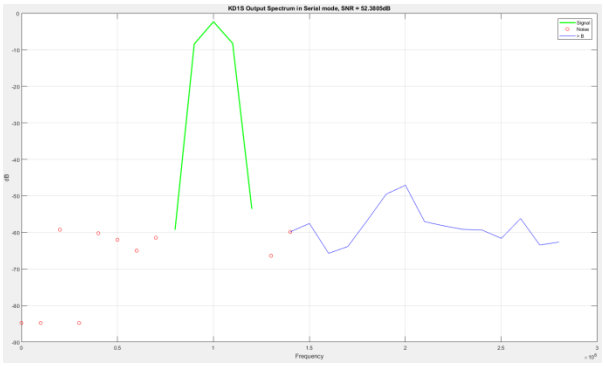
The MATLAB output plots for with an input signal frequency of 1MHz and an OSR of 256 are shown below.

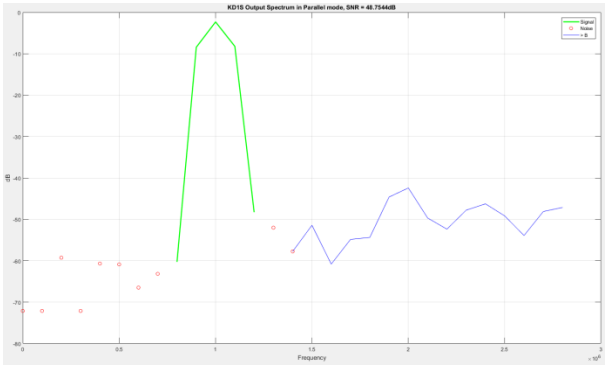


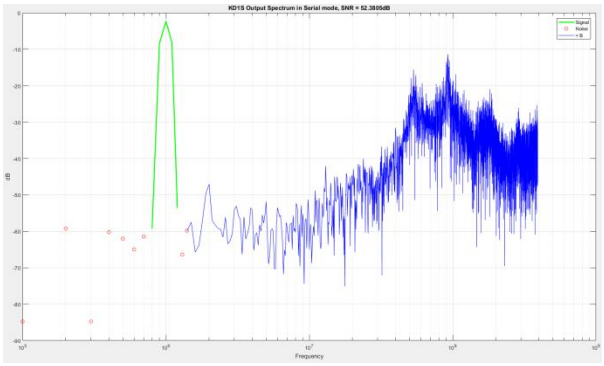


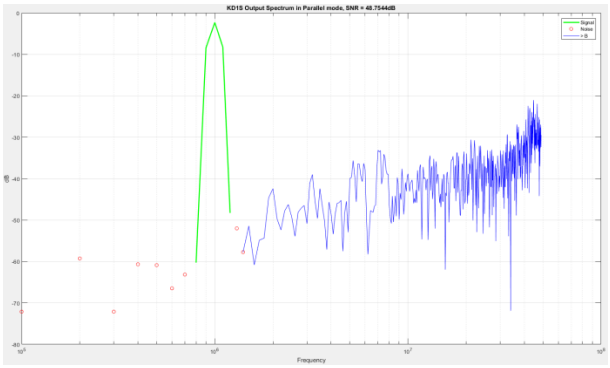




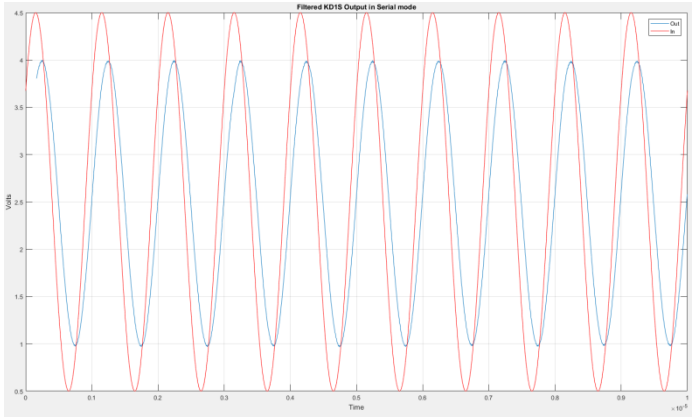




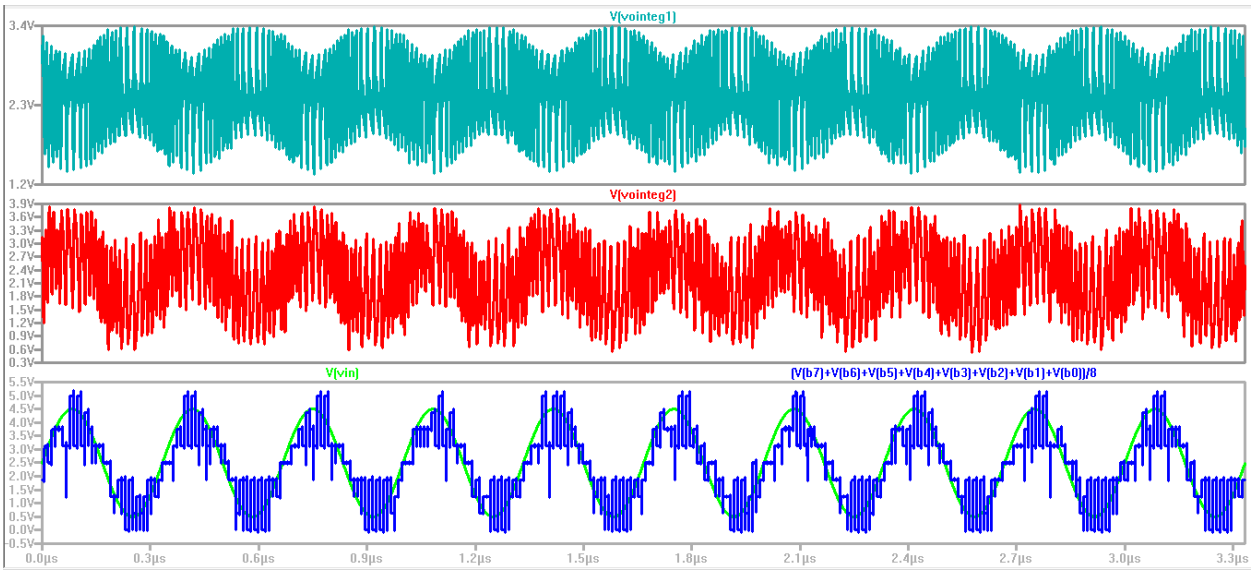




Note the MATLAB output plot of the “Filtered KD1S Output” for an OSR 64 (shown below) has much less attenuation on the input signal than for an OSR 256. This is true for both serial and parallel.

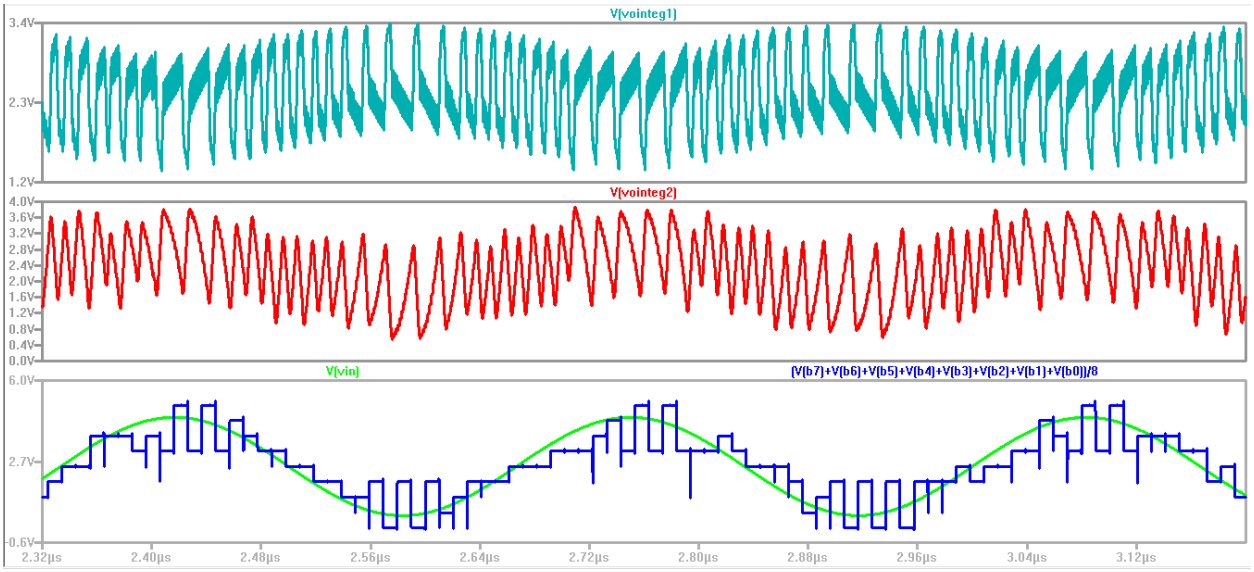


The simulation plots of the second-order topology with an input signal frequency of 3MHz are shown below.



"Dane_8D1S_2nd_Order_3MHz"

Below is a zoomed in version of the simulation above.



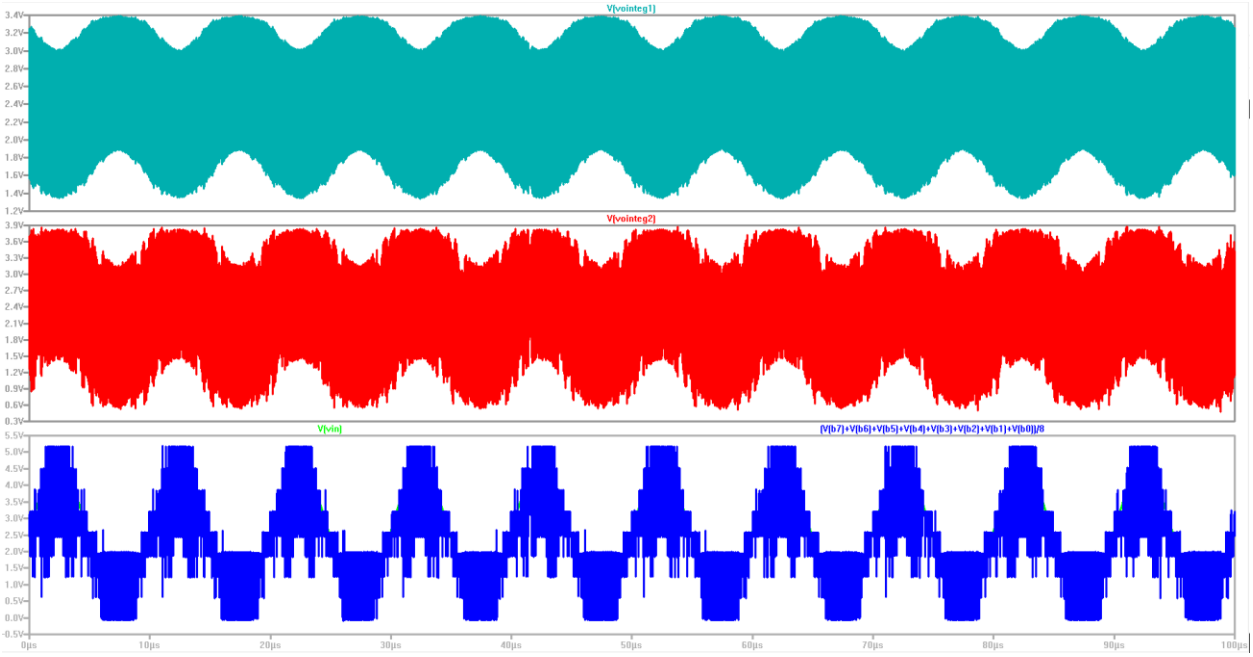
The MATLAB output results for with an input signal frequency of 3MHz and OSR's of 64, 128, and 256 are shown below.

```
Kpath = 8 OSR = 64
For 1-bit output (serial mode) at Kpath*fs = 785 MHz
SNR = 49.33, Neff = 7.90, B = 6.13 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 30.66, Neff = 4.80, B = 6.12 MHz>>
```

```
Kpath = 8 OSR = 128
For 1-bit output (serial mode) at Kpath*fs = 785 MHz
SNR = 54.86, Neff = 8.82, B = 3.07 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 41.85, Neff = 6.66, B = 3.06 MHz>> Dane_8D1S_2nd_Order_3MHz
```

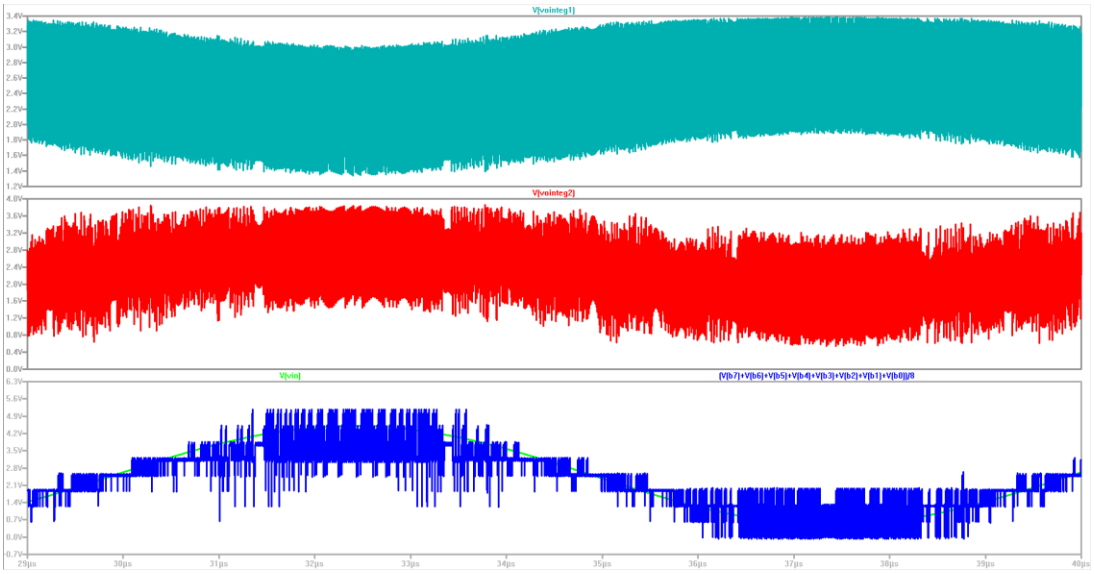
```
Kpath = 8 OSR = 256
For 1-bit output (serial mode) at Kpath*fs = 785 MHz
SNR = 59.67, Neff = 9.62, B = 1.53 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 52.99, Neff = 8.51, B = 1.53 MHz>> Dane_8D1S_2nd_Order_3MHz
```

A slow moving input signal with a frequency of 100KHz was applied to the second-order topology to observe its performance and see if it yields a better comparison to hand calculations. Unfortunately, as we will see shortly, it does not. The simulation plots of the second-order topology with an input signal frequency of 100KHz are shown below.



“Dane_8D1S_2nd_Order_100KHz”

Below is a zoomed in version of the simulation above.



The MATLAB output results for with an input signal frequency of 100KHz and OSR's of 64, 128, and 256 are shown below.

```
Kpath = 8 OSR = 64
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 38.42, Neff = 6.09, B = 6.12 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 25.77, Neff = 3.98, B = 6.12 MHz>>
```

```
Kpath = 8 OSR = 128
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 41.83, Neff = 6.65, B = 3.06 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 35.71, Neff = 5.64, B = 3.06 MHz>>
```

```
Kpath = 8 OSR = 256
For 1-bit output (serial mode) at Kpath*fs = 784 MHz
SNR = 44.49, Neff = 7.10, B = 1.53 MHz
For Kpath-bits output (parallel mode) at fs = 98 MHz
SNR = 42.13, Neff = 6.70, B = 1.53 MHz>>
```

The MATLAB output results for an input signal frequency of 100KHz, 1MHz, and 3MHz and OSR's of 64, 128, and 256 summarized in the tables below.

"Dane_8D1S_2nd_Order_100KHz" (10 Cycles)

	Serial 64	Parallel 64	Serial 128	Parallel 128	Serial 256	Parallel 256
$f_{s,new}$ (MHz)	784	98	784	98	784	98
SNR (dB)	38.42	25.77	41.83	35.71	44.49	42.13
N_{eff} (bits)	6.09	3.98	6.65	5.64	7.10	6.70
Bandwidth (MHz)	6.12	6.12	3.06	3.06	1.53	1.53

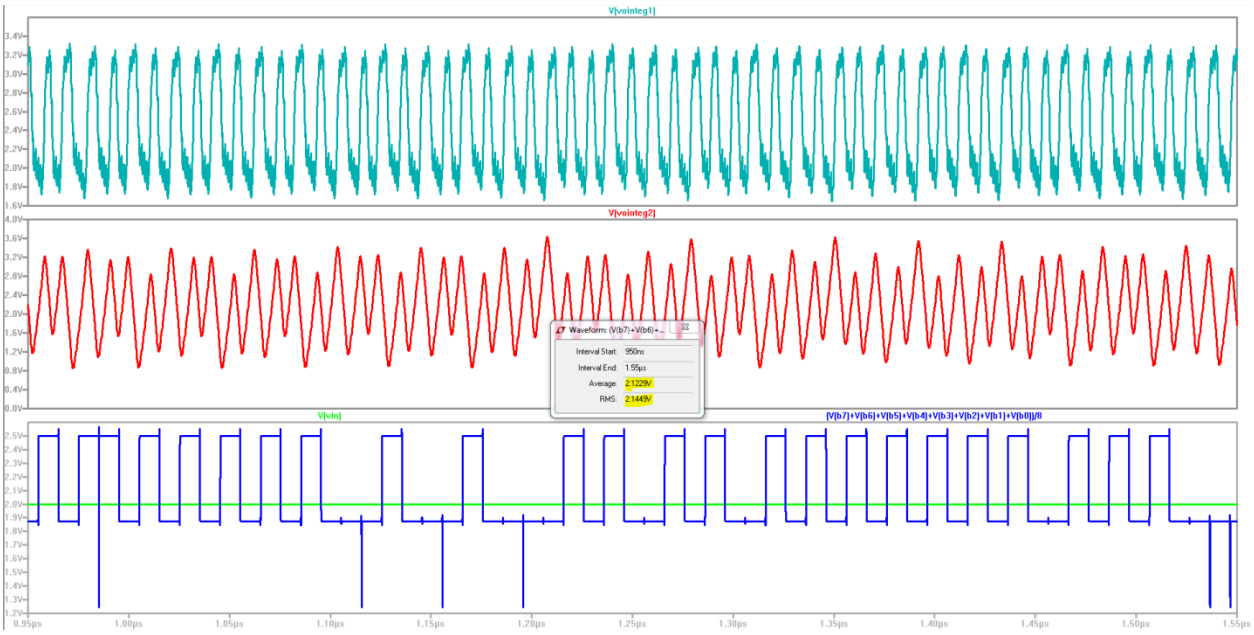
"Dane_8D1S_2nd_Order_1MHz" (10 Cycles)

	Serial 64	Parallel 64	Serial 128	Parallel 128	Serial 256	Parallel 256
$f_{s,new}$ (MHz)	784	98	784	98	784	98
SNR (dB)	41.90	26.26	47.98	38.87	52.38	48.75
N_{eff} (bits)	6.66	4.07	7.68	6.16	8.41	7.80
Bandwidth (MHz)	6.13	6.12	3.06	3.06	1.53	1.53

"Dane_8D1S_2nd_Order_3MHz" (10 Cycles)

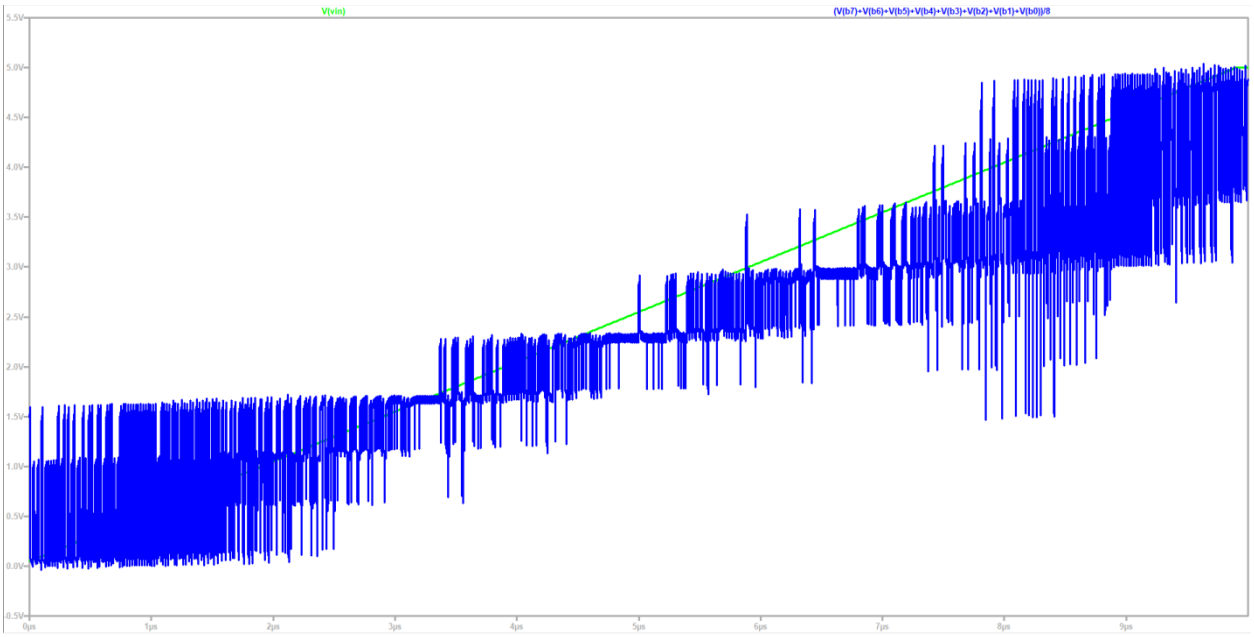
	Serial 64	Parallel 64	Serial 128	Parallel 128	Serial 256	Parallel 256
$f_{s,new}$ (MHz)	785	98	785	98	785	98
SNR (dB)	49.33	30.66	54.86	41.85	59.67	52.99
N_{eff} (bits)	7.90	4.80	8.82	6.66	9.62	8.51
Bandwidth (MHz)	6.13	6.12	3.07	3.06	1.53	1.53

The simulation plot of the second-order topology with a DC input signal is shown below. Note that the output averages to 2.12V which is approximately equal to the DC input signal of 2V.



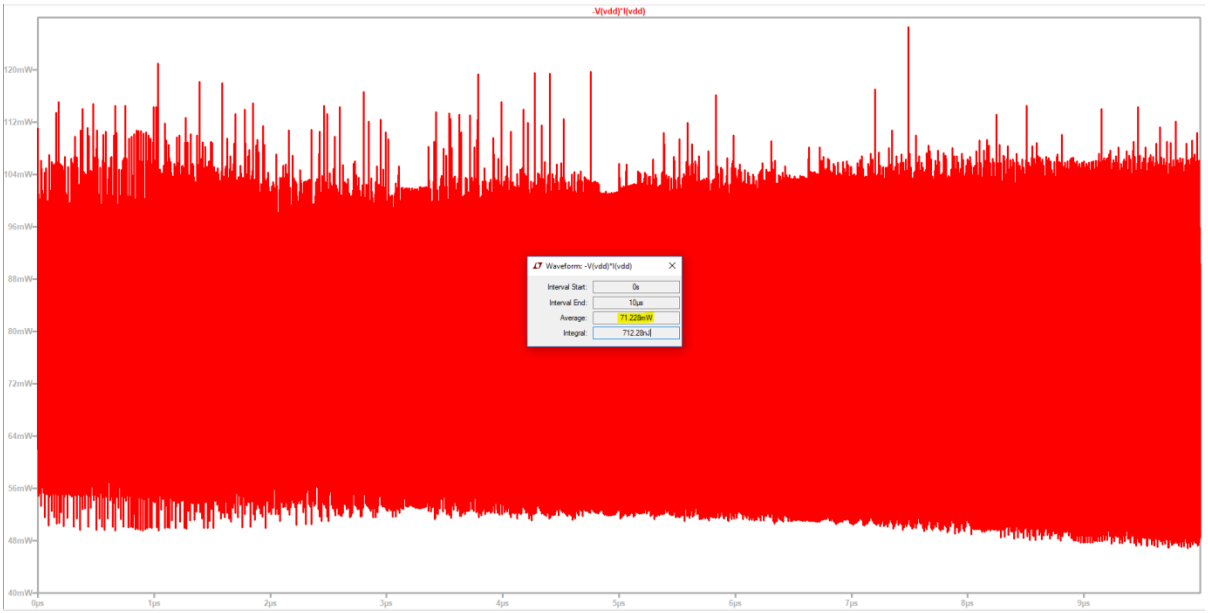
“Dane_8D1S_2nd_Order_DC”

The simulation plot of the second-order topology with a slow-moving ramp input signal is shown below. An RC filter was used to filter the output. Note the presence of dead zones in the first-order topology have almost been completely removed by using a second-order topology.



“Dane_8D1S_2nd_Order_Ramp”

The power consumption of the second-order topology is shown in the simulation below and was found to be 71.23 mW (average). This is very comparable to the power consumption of “_Fig9_33_Matlab_MSD” as it is not much higher than 68.55 mW (average).



Summary

Much experimentation was done in the design of several topologies in addition to the three topologies presented in this report. A 4-path second-order topology was thoroughly investigated and came close to, but did not outperform, the 8-path second-order topology.

Much insight was gained as to the performance of different topologies through the varying of resistor/capacitor values as well the gain of the integrator and the gain of the overall forward path. Even swapping the position of the switches in regard to their respective series resistors yielded slight differences in performance.

The final chosen design was the second-order topology due to its improved performance over the first-order topology and better comparison in the values of SNR, N_{eff}, and B to "_Fig9_33_Matlab_MSD".

Despite higher values of SNR, N_{eff}, and B in some cases for the first-order dither topology compared to the second-order topology, the second-order topology was chosen due to the visibility of saturation in the first-order dither topology. This saturation was able to be eliminated through the manipulation of various components in the design, however, it proved more useful to show how the addition of a dither affected the performance of the first-order topology without adjusting any other parameters and letting the dither remain an independent variable.

Furthermore, the values of SNR, N_{eff}, and B did not improve consistently across designs but rather experienced tradeoffs with sometimes improving for the serial case but worsening for the parallel case. Overall, the second-order topology certainly performed "comparable" to "_Fig9_33_Matlab_MSD" with better values of SNR, N_{eff}, and B for the serial case but slightly less values for the parallel case. In addition, the power consumption of the second-order topology was only slightly more at 71.23 mW (average) than that of "_Fig9_33_Matlab_MSD" at 68.55 mW (average).

Tables are provided below for comparison between the four designs discussed for a fixed frequency and OSR.

100 KHz

"Dane_8D1S_2nd_Order_100KHz" (10 Cycles)

	Serial 64	Parallel 64	Serial 128	Parallel 128	Serial 256	Parallel 256
$f_{s,new}$ (MHz)	784	98	784	98	784	98
SNR (dB)	38.42	25.77	41.83	35.71	44.49	42.13
N_{eff} (bits)	6.09	3.98	6.65	5.64	7.10	6.70
Bandwidth (MHz)	6.12	6.12	3.06	3.06	1.53	1.53

1 MHz

	OSR = 64 (Serial / Parallel)			
	Fig 9_33	**1st Order**	**1st Order Dither**	**2nd Order**
$f_{s,new}$ (MHz)	1829 / 228	784 / 98	784 / 98	784 / 98
SNR (dB)	38.00 / 34.18	37.91 / 30.45	23.95 / 24.04	41.90 / 26.26
N_{eff} (bits)	6.02 / 5.38	6.00 / 4.76	3.68 / 3.70	6.66 / 4.07
Bandwidth (MHz)	14.29 / 14.28	6.13 / 6.12	6.13 / 6.12	6.13 / 6.12

	OSR = 128 (Serial / Parallel)			
	Fig 9_33	**1st Order**	**1st Order Dither**	**2nd Order**
$f_{s,new}$ (MHz)	1829 / 228	784 / 98	784 / 98	784 / 98
SNR (dB)	40.98 / 39.48	43.96 / 39.55	49.43 / 45.79	47.98 / 38.87
N_{eff} (bits)	6.51 / 6.26	7.01 / 6.27	7.92 / 7.31	7.68 / 6.16
Bandwidth (MHz)	7.14 / 7.14	3.06 / 3.06	3.06 / 3.06	3.06 / 3.06

	OSR = 256 (Serial / Parallel)			
	Fig 9_33	**1st Order**	**1st Order Dither**	**2nd Order**
$f_{s,new}$ (MHz)	1829 / 228	784 / 98	784 / 98	784 / 98
SNR (dB)	54.14 / 53.01	48.92 / 48.39	52.58 / 52.47	52.38 / 48.75
N_{eff} (bits)	8.70 / 8.51	7.83 / 7.74	8.44 / 8.42	8.41 / 7.80
Bandwidth (MHz)	3.57 / 3.57	1.53 / 1.53	1.53 / 1.51	1.53 / 1.53

3 MHz

	OSR = 64 (Serial / Parallel)			
	Fig 9_33	**1st Order**	**1st Order Dither**	**2nd Order**
$f_{s,new}$ (MHz)	1830 / 229	785 / 98	785 / 98	785 / 98
SNR (dB)	39.97 / 38.93	39.66 / 33.26	33.37 / 27.38	49.33 / 30.66
N_{eff} (bits)	6.34 / 6.17	6.29 / 5.23	5.25 / 4.25	7.90 / 4.80
Bandwidth (MHz)	14.30 / 14.28	6.13 / 6.12	6.13 / 6.12	6.13 / 6.12

	OSR = 128 (Serial / Parallel)			
	Fig 9_33	**1st Order**	**1st Order Dither**	**2nd Order**
$f_{s,new}$ (MHz)	1830 / 229	785 / 98	785 / 98	785 / 98
SNR (dB)	53.73 / 51.18	44.89 / 42.21	34.71 / 34.59	54.86 / 41.85
N_{eff} (bits)	8.63 / 8.21	7.16 / 6.72	5.47 / 5.45	8.82 / 6.66
Bandwidth (MHz)	7.15 / 7.14	3.07 / 3.06	3.07 / 3.06	3.07 / 3.06

	OSR = 256 (Serial / Parallel)			
	Fig 9_33	**1st Order**	**1st Order Dither**	**2nd Order**
$f_{s,new}$ (MHz)	1830 / 229	785 / 98	785 / 98	785 / 98
SNR (dB)	58.41 / 59.04	48.13 / 45.95	36.07 / 36.02	59.67 / 52.99
N_{eff} (bits)	9.41 / 9.51	7.70 / 7.34	5.70 / 5.69	9.62 / 8.51
Bandwidth (MHz)	3.57 / 3.57	1.53 / 1.53	1.53 / 1.53	1.53 / 1.53