EE247
Lecture 13

- Data Converters
  - Data converter testing (continued)
    - Dynamic tests
      - Spectral testing (brief review)
      - Relationship between: DNL & SNR, INL & SFDR
    - Effective number of bits (ENOB)
  - D/A converters: Various Architectures
    - Resistor string DACs
    - Serial charge redistribution DACs
    - Charge scaling DACs
    - R-2R type DACs
    - Current based DACs

Data Converter Testing (So Far)

- Data Converters
  - Static Tests:
    - Measuring DNL & INL
      - Servo-loop
      - Code density testing (histogram testing)
  - Dynamic tests:
    - Spectral testing→ Reveals ADC errors associated with dynamic behavior i.e. ADC performance as a function of frequency
      - Direct Discrete Fourier Transform (DFT) based measurements utilizing sinusoidal signals
      - DFT measurements including windowing

Spectral Testing: DFT Considerations
Integer Cycles versus Windowing

- Sinusoidal inputs with integer number of cycles within the observation window:
  - Signal energy for a single sinusoid falls into single DFT bin
  - Requires careful choice of f_{in} :
    - # of cycles → integer
    - N/cycles = f_s / f_{in} non-integer (choose # of cycles integer and prime #)
  - Ideal for simulations
  - Measurements → need to lock f_{in} to f_s (PLL)- not always possible
- Windowing
  - No restrictions on f_{in} → no need to have the signal locked to f_s
    → Good for measurements w/o having the capability to lock f_{in} to f_s
  - Signal energy and its harmonics distributed over several DFT bins – handle smeared-out harmonics with care!
  - Requires more samples for a given accuracy
  - Note that no windowing is equal to windowing with a rectangular window!

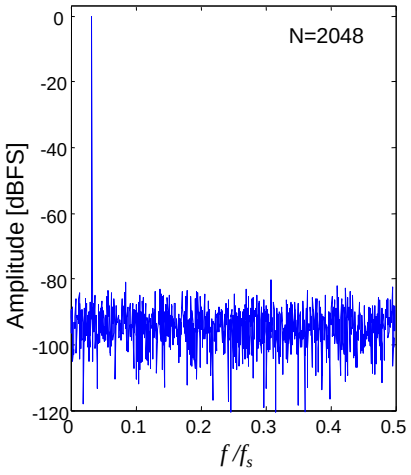
Matlab Example:
ADC Spectral Testing

- ADC with B bits
- Full scale input level=2

```
B = 10;
delta = 2/2^B;
%sampled sinusoid
y = cos(2*pi*fx/fs*[0:N-1]);
%quantize samples to delta=1LSB
y=round(y/delta)*delta;
s = abs(fft(y)/N*2);
f = (0:length(s)-1) / N;
```

ADC Output Spectrum

- $N=2^{11}$
- *Cycles=53 (integer & prime)*
- *N/Cycles= f_s/f_x*
  - → *f_s/f_x =38.64…. (non-integer)*
- Input signal bin:
  - Bx @ bin # ($N * f_x/f_s + 1$) (Matlab arrays start at 1)
  - A_{signal} = 0dBFS
- What is the SNR?

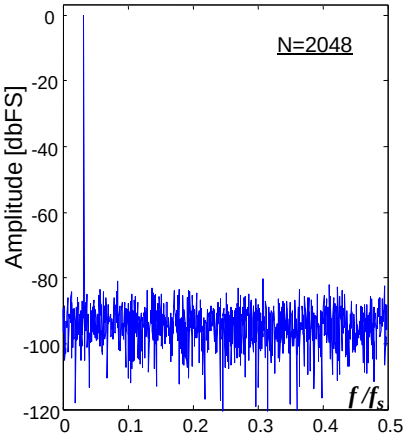


Simulated ADC Output Spectrum

- Noise bins: all except signal bin

```
bx = N*fx/fs + 1;
As = 20*log10(s(bx))
%set signal bin to 0
s(bx) = 0;
An = 10*log10(sum(s.^2))
SNR = As - An
```

- Matlab→SNR = 62dB (10 bits)
- Computed SQNR = 6.02xN+1.76dB=61.96dB



Note: In a real circuit including thermal/flicker noise → the measured total noise is the sum of quantization & noise associated with the circuit

Why is Noise Floor Not @ -62dB ?

- DFT bins act like an analog spectrum analyzer with bandwidth per bin of f_s/N
- Assuming noise is uniformly distributed, noise per bin:

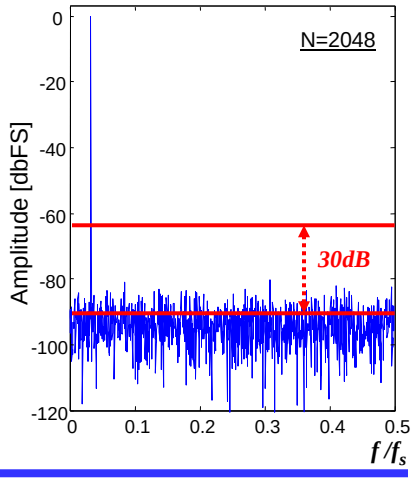
  (Total noise)/N/2

  →The DFT noise floor wrt total noise:

  $-10\log_{10}(N/2)$ [dB]

  below the actual noise floor
- For N=2048:

  $-10\log_{10}(N/2) = -30$ [dB]

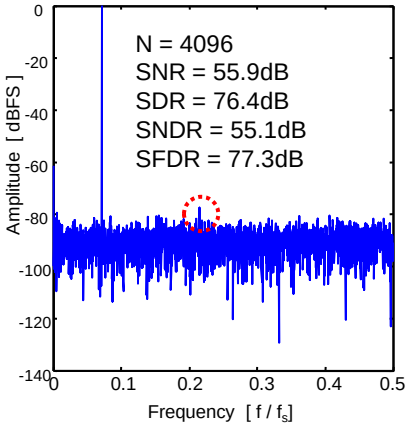


DFT Plot Annotation

- Need to annotate DFT plot such that actual noise floor can be readily computed by one of these 3 ways:
  1. Specify how many DFT points (N) are used
  2. Shift DFT noise floor by $10\log_{10}(N/2)$ [dB]
  3. Normalize to "noise power in 1Hz bandwidth" then noise is in the form of power spectral density

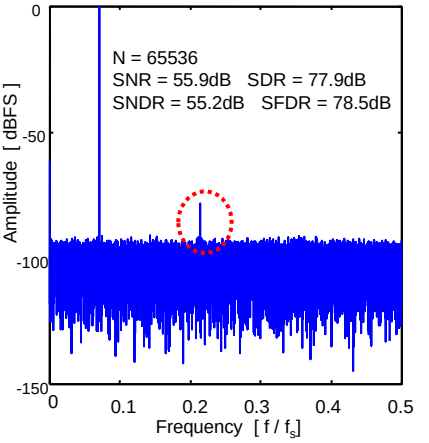
Example:10Bit ADC FFT

- For a real 10bit ADC spectral test results:
- SNR=55.9dB
- A 3rd harmonic is barely visible
- Is better view of distortion component possible?



Example:10Bit ADC FFT

- Increasing N, the number of samples (and hence the measurement or simulation time) distributes the noise over larger # of bins
- Larger # of bins → less noise power per bin (total noise stays constant)
- Note the 3rd harmonic is clearly visible when N is increased

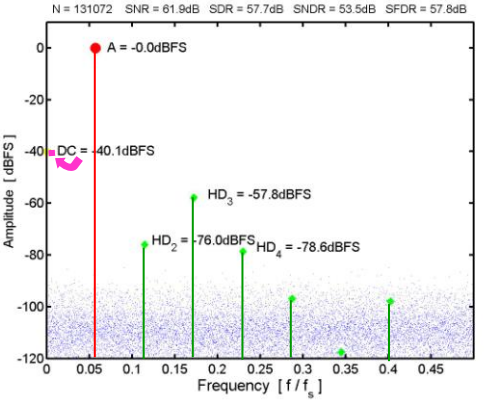


Spectral Performance Metrics
ADC Including Non-Idealities

- Signal S
- DC
- Distortion D
- Noise N

- Ideal ADC adds:
  - Quantization noise

- Real ADC typically adds:
  - Thermal and flicker noise
  - Harmonic distortion associated with circuit nonlinearities

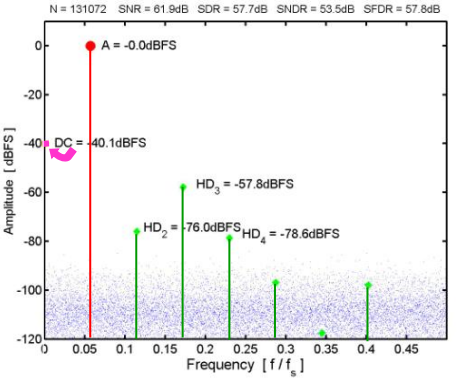


ADC Spectral Performance Metrics
SNR

- Signal S
- DC
- Distortion D
- Noise N

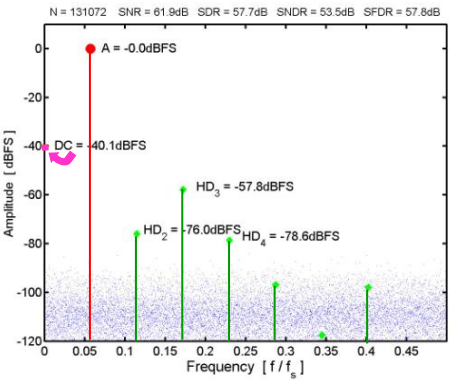
- Signal-to-noise ratio
  SNR = 10log[(Signal Power) / (Noise Power)]

- In Matlab: Noise power includes power associated with all bins except:
  - DC
  - Signal
  - Signal harmonics



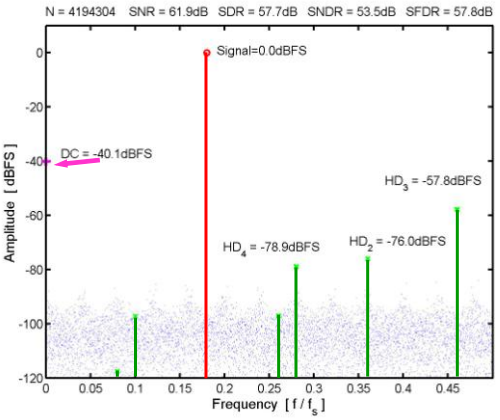
ADC Spectral Performance Metrics
SDR & SNDR & SFDR

- SDR→ Signal-to-distortion ratio
  = 10log[(Signal Power) /
  (Total Distortion Power)]
- SNDR→ Signal-to-(noise+distor
  = 10log[S / (N+D)]
- SFDR→ Spurious-free dynamic range
  = 10log[(Signal)/
  (Largest Harmonic)]
  → Typically SFDR > SDR



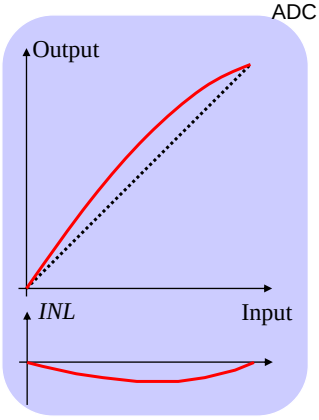
Harmonic Components

- At multiples of f_x
- Aliasing:
  - $f_{signal} = f_x = 0.18\ f_s$
  - $f_2 = 2\ f_0 = 0.36\ f_s$
  - $f_3 = 3\ f_0 = 0.54\ f_s$
    → $0.46\ f_s$
  - $f_4 = 4\ f_0 = 0.72\ f_s$
    → $0.28\ f_s$
  - $f_5 = 5\ f_0 = 0.90\ f_s$
    → $0.10\ f_s$
  - $f_6 = 6\ f_0 = 1.08\ f_s$
    → $0.08\ f_s$

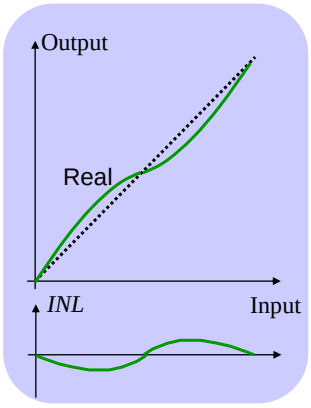


Relationship INL & SFDR/SNDR

ADC Transfer Curve

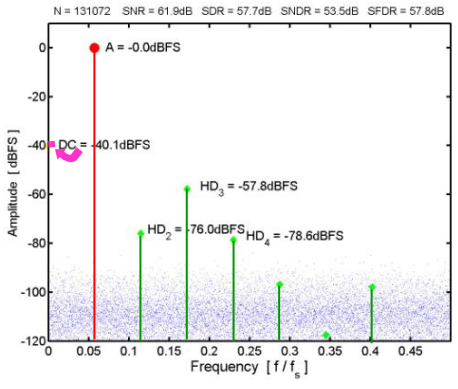


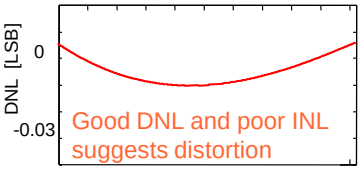
Quadratic shaped transfer function:
→ Gives rise to **even** order harmonics

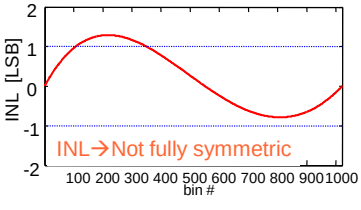


Cubic shaped transfer function:
→ Gives rise to **odd** order harmonics

Frequency Spectrum versus INL & DNL



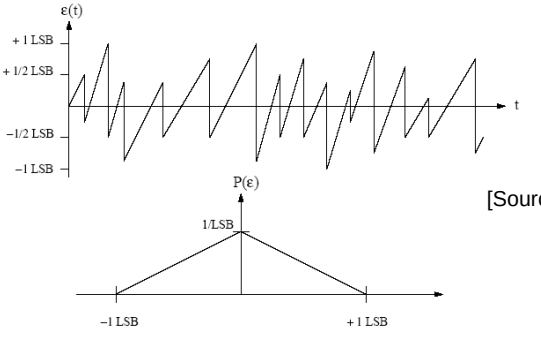




Relationship INL & SFDR/SNDR

- Nature of harmonics depend on "shape" of INL curve
- Rule of Thumb: SFDR ≅ 20log(2^B/INL)
  - E.g. 1LSB INL, 10b→ SFDR≅60dB
- Beware, this is of course only true under the same conditions at which the INL was taken, i.e. typically low input signal frequency

SNR Degradation due to DNL



[Source: Ion Opris]

- Uniform quantization error pdf was assumed for ideal quantizer over the range of: +/- Δ/2
- Let's now add uniform DNL over +/- Δ/2 and repeat math...
  - Joint pdf for two uniform pdfs → Triangular shape

SNR Degradation due to DNL

- To find total noise → Integrate triangular pdf:

$$\overline{e^2} = 2\int_0^{+\Delta}(1-e)\frac{e^2}{\Delta}de = \frac{\Delta^2}{6} \qquad \Rightarrow SNR = 6.02\cdot N - 1.25 \text{ [dB]}$$

3dB

- Compare to ideal quantizer:

$$\overline{e^2} = \int_{-\Delta/2}^{+\Delta/2}\frac{e^2}{\Delta}de = \frac{\Delta^2}{12} \qquad \Rightarrow SNR = 6.02\cdot N + 1.76 \text{ [dB]}$$

→Error associated with DNL reduces overall SNR

SNR Degradation due to DNL

- More general case:
  - Uniform quantization error (ideal) ±0.5Δ
  - Uniform DNL error ± DNL [LSB]
  - Convolution yields trapezoid shaped joint pdf
  - SQNR becomes:

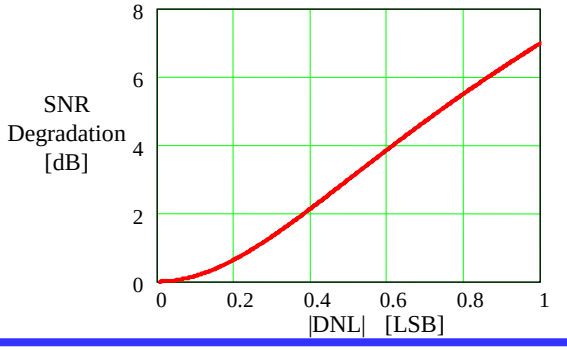
$$SQNR = \frac{\frac{1}{2}\left(\frac{2^N\Delta}{2}\right)^2}{\frac{\Delta^2}{12}+\frac{DNL^2}{3}}$$

SNR Degradation due to DNL

- Degradation in dB:

$$SQNR\_\deg = 1.76 - 10\log\left[\frac{\frac{1}{8}}{\frac{1}{12}+\frac{DNL^2}{3}}\right]$$

Valid only for cases where no missing codes



Summary

INL & SFDR - DNL & SNR

INL & SFDR

- Type of distortion depends on "shape" of INL
- Rule of Thumb:

$$SFDR \cong 20\log(2^B/INL)$$

  - E.g. 1LSB INL, 10b
    → SFDR≅60dB

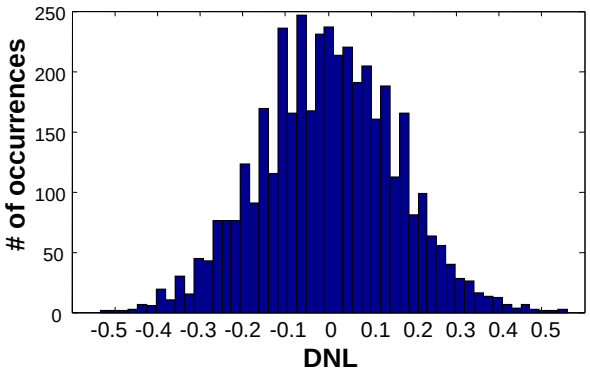
DNL & SNR

Assumptions:

- DNL pdf →uniform
- No missing codes

$$SQNR = \frac{\frac{1}{2}\left(\frac{2^N\Delta}{2}\right)^2}{\frac{\Delta^2}{12}+\frac{DNL^2}{3}}$$

Uniform DNL?



- DNL distribution of measurement for 12-bit ADC test chip
- Not quite uniform...

Effective Number of Bits (ENOB)

- Is a 12-Bit converter with 68dB SNDR really a 12-Bit converter?
- Effective Number of Bits ($ENOB$)→ # of bit of an ideal ADC with the same SQNR as the SNDR of the non-ideal ADC

$$ENOB = \frac{SNDR - 1.76\text{dB}}{6.02\text{dB}}$$

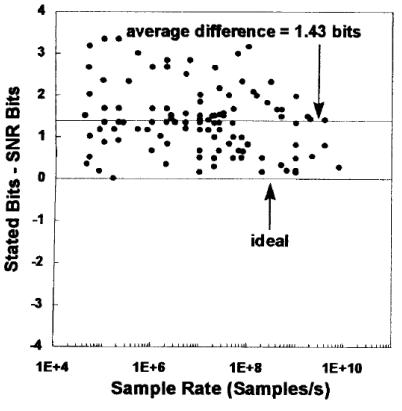
$$= \frac{68 - 1.76}{6.02} = 11.0\text{Bits}$$

→ Above ADC is a 12bit ADC with ENOB=11bits

ENOB

- At best, we get "ideal" ENOB only for negligible thermal noise, DNL, INL
- Low noise design is costly → 4x penalty in power per (ENOB-) bit or 6dB extra SNDR
- Rule of thumb for good performance /power tradeoff: ENOB < N-1

ENOB Survey

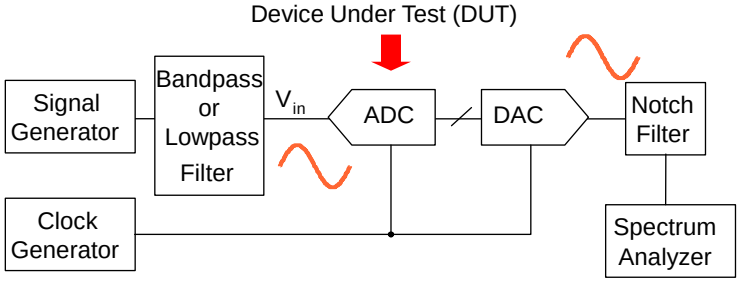


R. H. Walden, "Analog-to-digital converter survey and analysis," *IEEE J. on Selected Areas in Communications*, pp. 539-50, April 1999

Converter Testing Practical Aspects

- Equipment requirements
- Pitfalls

Direct ADC-DAC Test



- Issues to beware of:
  - Linearity of the signal generator output has to be much better than ADC linearity
  - Spectrum analyzer nonlinearities
    - → May need to build/purchase filters to address one or both above problems
  - Clock generator signal jitter

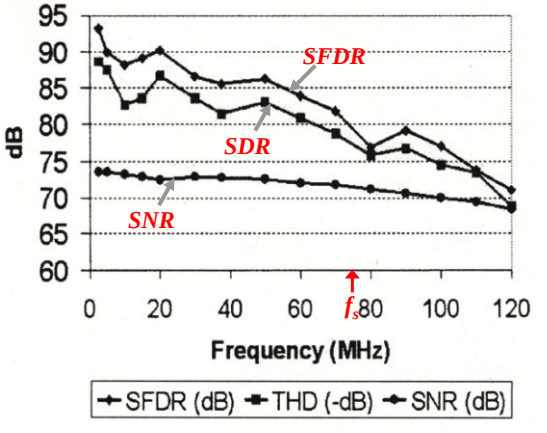
Example: State-Of-The-Art ADC (2001)

0.35micron technology & 3V Supply	
Resolution	14 bits
Conversion Rate	75 MSPS
Input Range	2 V_{pp} differential
SNR @ Nyquist	73 dB → ENOB=12bit
SFDR @ Nyquist	88 dB
DNL	0.6 LSB
INL	2.0 LSB

[W. Yang et al., "A 3-V 340-mW 14-b 75-Msample/s CMOS ADC with 85-dB SFDR at Nyquist input," *IEEE J. of Solid-State Circuits*, Dec. 2001]

- Testing a high performance converter may be just as challenging as designing it!
- Key to success is to be aware of test setup and equipment limitations

Example: ADC Spectral Tests



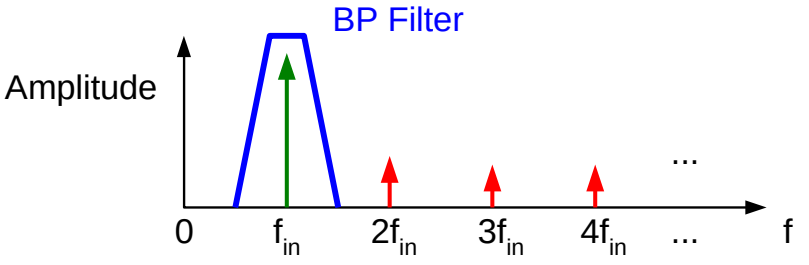
Ref: W. Yang et al., "A 3-V 340-mW 14-b 75-Msample/s CMOS ADC with 85-dB SFDR at Nyquist input," *IEEE J. of Solid-State Circuits*, Dec. 2001

Signal Source

- Typical $40k signal source spec.s:

- f=100kHz...3GHz
- Harmonic distortion (f>1MHz): -30dBc !
- Still need a filter to elliminate harmonic distortion!

Filtering Out Harmonics



- Given HD=-30dBc, we need a stopband rejection > 65dB to get SFDR>95dB

Available Filters

Elliptical Function Bandpass Filters 1kHz to 20MHz

www.tte.com, or
www.allenavionics.com

Stopband to Passband Bandwidth Ratios

Series Number	BWR	*Stopband Attenuation
Q34	4.0:1	-40dBc
Q40	4.0:1	-40dBc
Q36	10.0:1	-60dBc
Q54	2.5:1	-40dBc
Q70	3.5:1	-60dBc
Q56	3.5:1	-60dBc

- Fixed frequency filters!
- Want to test at many frequencies → Need to have many different filters!

Tunable Filter

www.klmicrowave.com

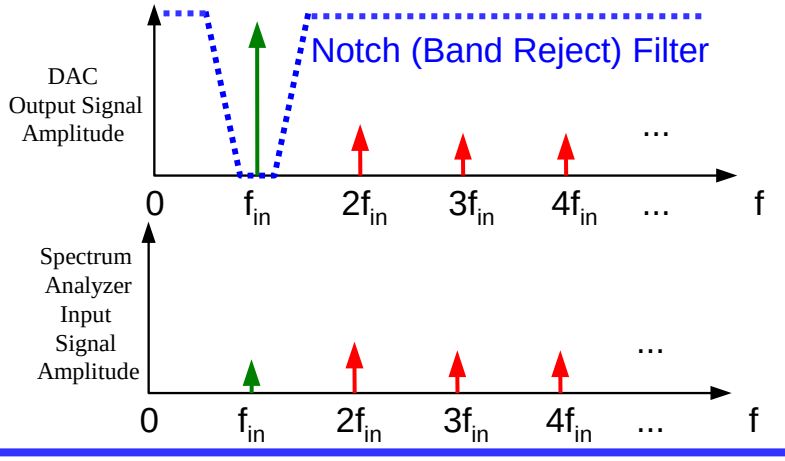
K&L Model	Frequency Range (MHz)	Passband Insertion Loss	Length Inch/mm	Width Inch/mm	Height Inch/mm
5BT-30/76-5-N/N	30-76	1.3 dB Max	9.80/249	5.38/137	2.75/50
5BT-63/125-5-N/N	63-125	1.3 dB Max	9.80/249	5.38/137	2.75/50
5BT-125/250-5-N/N	125-250	1.3 dB Max	9.80/249	5.38/137	2.75/50
5BT-250/500-5-N/N	250-500	1.0 dB Max	9.80/249	5.38/137	2.75/50
5BT-375/750-5-N/N	375-750	1.0 dB Max	9.80/249	5.38/137	2.75/50
5BT-500/1000-5-N/N	500-1000	1.0 dB Max	9.80/249	5.38/137	2.75/50
5BT-750/1500-5-N/N	750-1500	1.0 dB Max	9.80/249	5.38/137	2.75/50
5BT-1000/2000-5-N/N	1000-2000	1.0 dB Max	7.38/187	2.88/73	2.75/50
5BT-1200/2600-5-N/N	1200-2600	1.0 dB Max	7.38/187	2.88/73	2.75/50

Filter Distortion

- Beware: The filters themselves could also introduce distortion
- Distortion is usually not specified, need to contact manufacturer directly!
- Often guaranteed: HD<-85dBc,
- Don't trust your filters blindly...

Filtering Input to Spectrum Analyzer

Prevent Signal Distortion Incurred by Spec. Analyzer



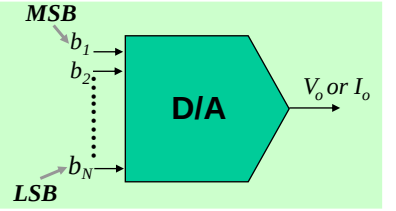
Clock Generator

- The clock signal controls sampling instants – which we assumed to be precisely equi-distant in time (period T)
- Typically, clock signal generators output signal have some level of variability in T called:
  - "Aperture Uncertainty" or "Aperture Jitter "
- Variability in T causes loss of performance in Data Converters
- How much Jitter can be tolerated? (later)

D-to-A Converter Design

D/A Converter Transfer Characteristics

- An ideal digital-to-analog converter:
  - Accepts digital inputs b_1-b_n
  - Produces either an analog output voltage or current
  - Assumption
    - Uniform, binary digital encoding
    - Unipolar output ranging from 0 to V_{FS}



Nomenclature:

$N = \#\ of\ bits$

$V_{FS} = full\ scale\ output$

$\Delta = min.\ step\ size \rightarrow 1LSB$

$$\Delta = \frac{V_{FS}}{2^N}$$

$$or\ N = log_2 \frac{V_{FS}}{\Delta} \rightarrow resolution$$

D/A Converters

- Various D/A architecture
  - Resistor string DAC
  - Charge Redistribution DAC
  - Current source type
- Static performance
  - Limited by component matching
  - Architectures
    - Unit element
    - Binary weighted
    - Segmented
  - Performance improvement via dynamic element matching
- Dynamic performance
  - Limited by timing errors causing glitches

D/A Converters

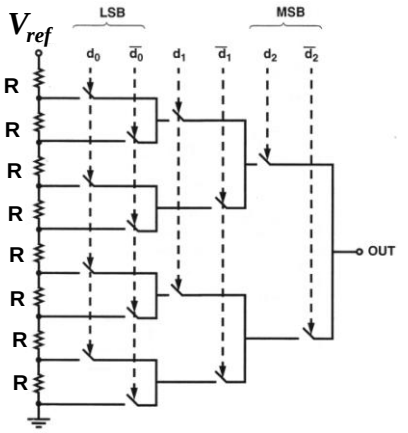
- Comprises *voltage* or *charge* or *current* based elements
- Examples for above three categories:
  - Resistor string → *voltage*
  - Charge redistribution → *charge*
  - Current source type → *current*

Resistor String DAC

Voltage based:

- A *B-bit* DAC requires: 2^B resistors in series
- All resistors values equal

→ Generates 2^B equally spaced voltages ready to be chosen based on the digital input word



3-Bit Resistor String DAC

R-String DAC Example

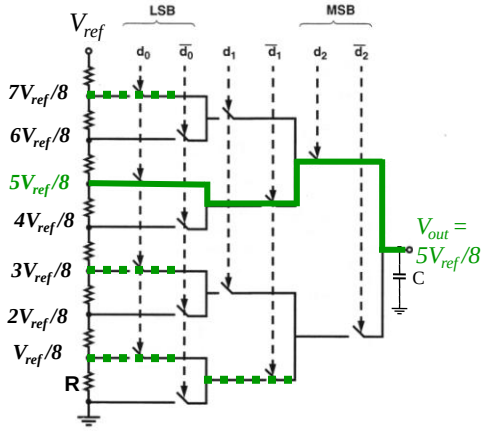
Example:

- Input code: [d2 d1 d0]=***101***

  → $V_{out} = 5V_{ref}/8$

- Assuming switch resistance << ***R:***

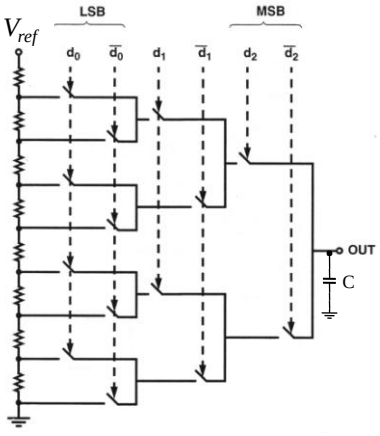
$$\tau_{settling} = (3R||5R) \times C = 0.23 \times 8RC$$



R-String DAC

- Advantages:
  - Takes full advantage of availability of almost perfect switches in MOS technologies
  - Simple, fast for <8-10bits
  - Inherently monotonic
  - Compatible with purely digital technologies
- Disadvantages:
  - 2^B resistors & ~$2x2^B$ switches for B bits → High element count & large area for B >10bits
  - High settling time for high resolution DACs:

    τ_{max} ~ 0.25×2^B RC



Ref:

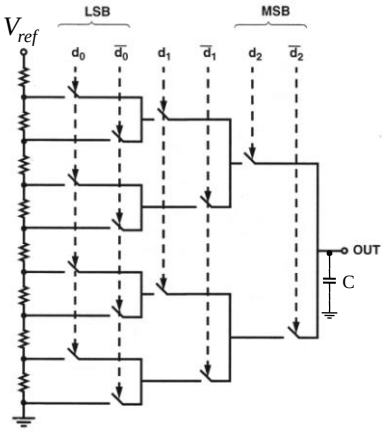
M. Pelgrom, "A 10-b 50-MHz CMOS D/A Converter with 75-W Buffer," JSSC, Dec. 1990, pp. 1347

R-String DAC

- Choice of resistor value:
  - Since maximum output settling time:

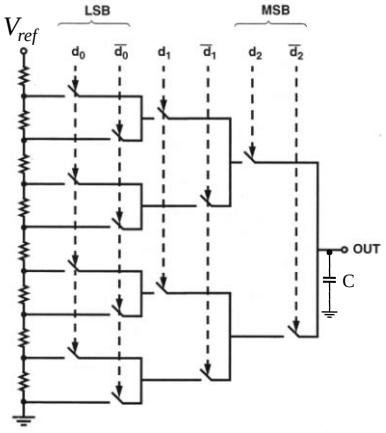
    $\tau_{max} \sim 0.25 \times 2^B RC$
  - Choice of resistor value directly affects DAC maximum operating speed $f_{max} \sim 1/R$
  - Power dissipation: function of $V_{ref}^2/(Rx2^B)$

→ Tradeoff between speed and power dissipation



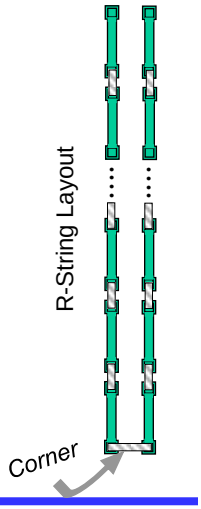
R-String DAC

- Resistor type:
  - Choice of resistive material important
  - Diffusion type R → high temp. co. & voltage co.
    - Results in poor INL/DNL
  - Better choice is poly resistor beware of poly R 1/f noise
  - At times, for high-frequency & high performance DACs, metal R (beware of high temp. co.) or thin film R is used



R-String DAC
Layout Considerations

- Number of resistor segments → 2^B
  - E.g. 10-bit R-string DAC →1024 resistors
- Low INL/DNL dictates good R matching
- Layout quite a challenge
  - Good matching mandates all R segments either vertical or horizontal - not both
  - Matching of metal interconnect and contacts
  - Need to fold the string
    - Difficult to match corner segments to rest
    - Could result in large INL/DNL

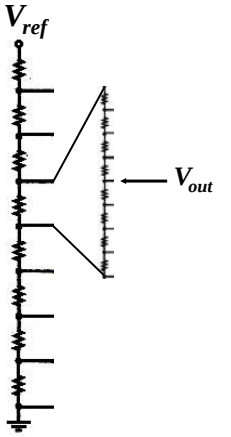


R-String DAC
Including Interpolation

Resistor string DAC + Resistor string interpolator increases resolution w/o drastic increase in complexity
e.g. 10bit DAC→ (5bit +5bit→ $2x2^5=2^6$ # of Rs) instead of direct 10bit→2^{10}

Considerations:

- ❑ Main R-string loaded by the interpolation string resistors
- ❑ Large R values for interpolating string→ less loading but lower speed
- ❑ Can use buffers

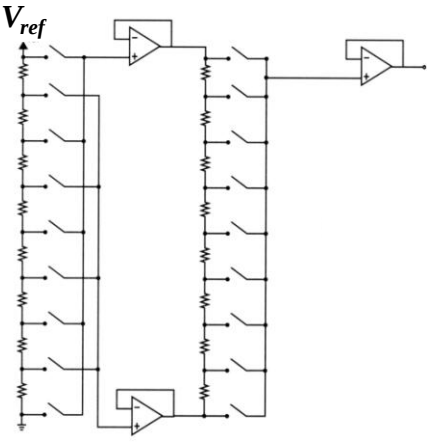


R-String DAC
Including Interpolation

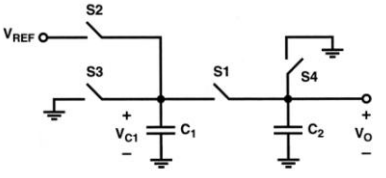
Use buffers to prevent loading of the main ladder

Issues:

- → Buffer DC offset
- → Effect of buffer bandwidth limitations on overall speed



Charge Based: Serial Charge Redistribution DAC

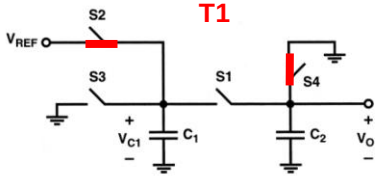


Nominally $C_1=C_2$

- Operation based on redistribution of charge associated with C1 & C2 to perform precise division by factor of **2**

Charge Based: Serial Charge Redistribution DAC
Simplified Operation: Conversion Sequence

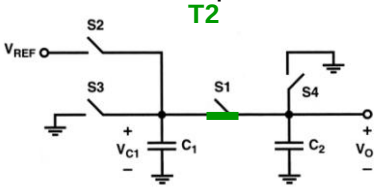
T1



$$Q_{C_1}^{T1} = V_{REF} \times C_1 \quad \& \quad Q_{C_1}^{T1} = 0$$

$$\rightarrow Q_{C_1}^{T1} + Q_{C_2}^{T1} = V_{REF} \times C_1$$

T2



$$Q_{C_1}^{T1} + Q_{C_2}^{T1} = Q_{C_1}^{T2} + Q_{C_2}^{T2} = (C_1 + C_2)V_O$$

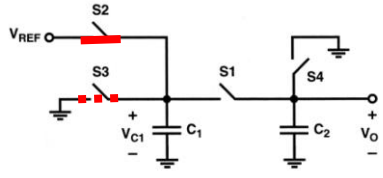
$$V_{REF} \times C_1 = (C_1 + C_2)V_O$$

$$V_O = V_{REF} \times \frac{C_1}{C_1 + C_2}$$

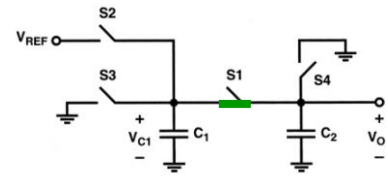
$$Since\ C_1 = C_2 \rightarrow \boxed{V_O = \frac{V_{REF}}{2}}$$

Serial Charge Redistribution DAC
Simplified Operation (Cont'd)

T1



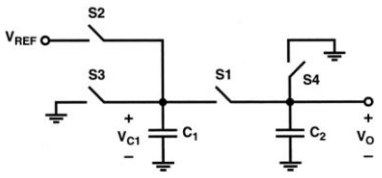
T2



- Conversion sequence:
  - Next cycle
    - If S3 closed V_{C1}=0 then when S1 closes $V_{C1} = V_{C2} = V_{REF}/4$
    - If S2 closed $V_{C1}=V_{REF}$ then when S1 closes $V_{C1} = V_{C2} = V_{REF}/2 + V_{REF}/4$

Serial Charge Redistribution DAC

- Conversion sequence:
  - Discharge C1 & C2→ S3& S4 closed
  - For each bit in succession beginning with LSB, b_1:
    - S1 open- if b_i=1 C1 precharge to V_{REF} if b_i=0 discharged to GND
    - S2 & S3 & S4 open- S1 closed- Charge sharing C1 & C2
      - → ½ of precharge on C1 +½ of charge previously stored on C2→ C2



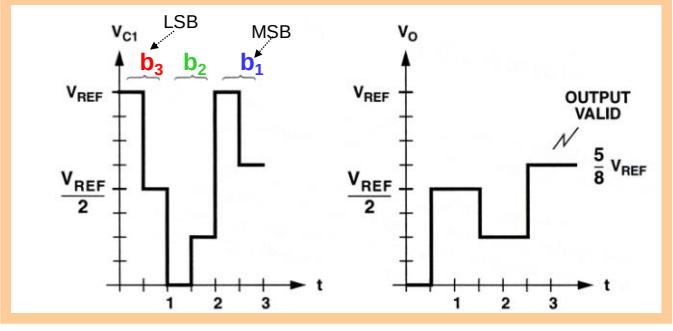
$$V_o(1) = \frac{b_N}{2}V_{REF}$$

$$V_o(2) = \frac{1}{2}\left(b_{N-1} + \frac{b_N}{2}\right)V_{REF}$$

$$V_o(N) = \left(\sum_{i=1}^{N} \frac{b_i}{2^i}\right)V_{REF}$$

Serial Charge Redistribution DAC

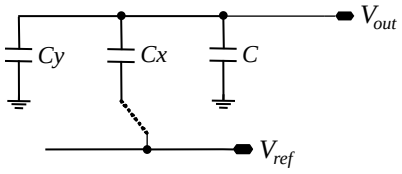
Example: Input Code 101



- Example input code 101→ output (4/8 +0/8 +1/8)V_{REF} =5/8 V_{REF}
- Very small area
- For an N-bit DAC, N redistribution cycles for one full analog output generation → quite slow

Parallel Charge Scaling DAC

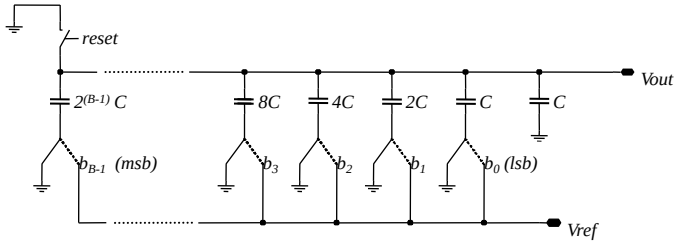
- DAC operation based on capacitive voltage division



$$V_{out} = \frac{Cx}{Cx+Cy+C} V_{ref}$$

→ Make *Cx* & *Cy* function of incoming DAC digital word

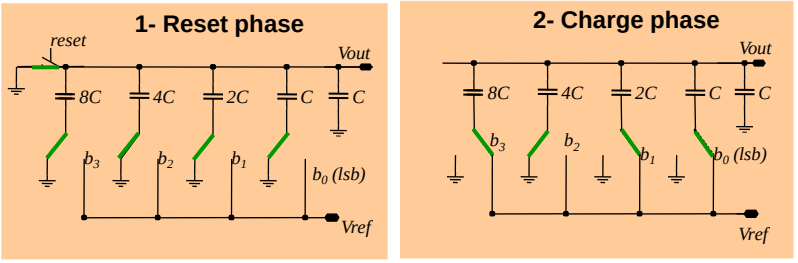
Parallel Charge Scaling DAC



- E.g. "Binary weighted"
- B+1 capacitors & B switches (Cs built of unit elements → 2^B units of C)

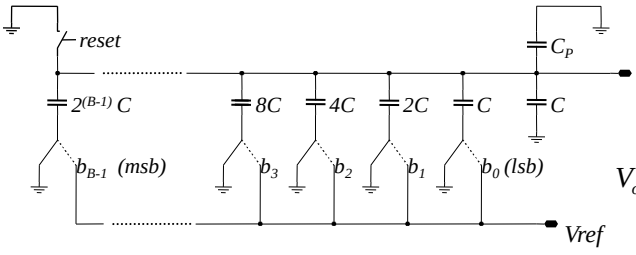
$$V_{out} = \frac{\sum_{i=0}^{B-1} b_i 2^i C}{2^B C} V_{ref}$$

Charge Scaling DAC
Example: 4Bit DAC- Input Code 1011



$$V_{out} = \frac{2^0C + 2^1C + 2^3C}{2^4C} V_{ref} = \frac{11}{16} V_{ref}$$

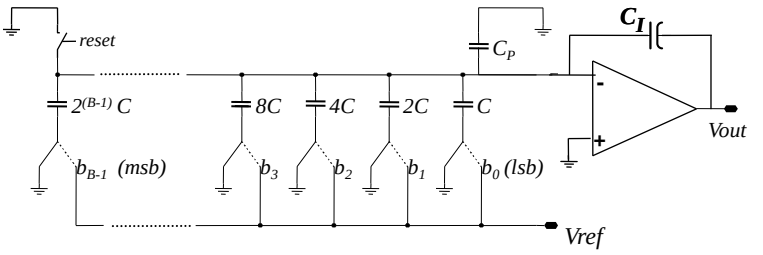
Charge Scaling DAC



$$V_{out} = \frac{\sum_{i=0}^{B-1} b_i 2^i C}{2^B C + C_P} V_{ref}$$

- Sensitive to parasitic capacitor @ output
  - If C_p constant → gain error
  - If C_p voltage dependant → DAC nonlinearity
- Large area of caps for high DAC resolution (10bit DAC ratio 1:512)
- Monotonicity depends on element matching (more later)

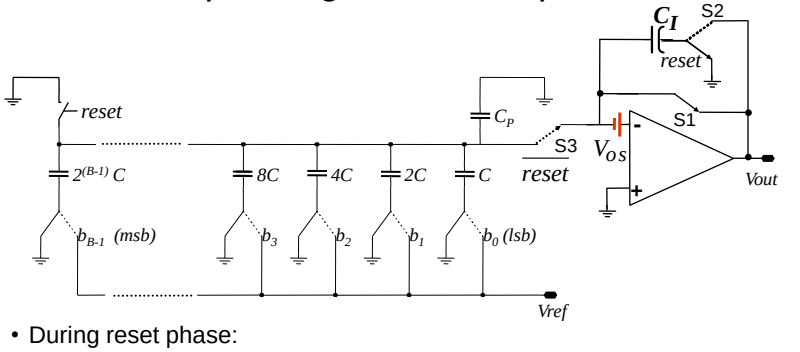
Parasitic Insensitive Charge Scaling DAC



$$V_{out} = -\frac{\sum_{i=0}^{B-1} b_i 2^i C}{C_I} V_{ref} \quad , \quad C_I = 2^B C \rightarrow V_{out} = -\frac{\sum_{i=0}^{B-1} b_i 2^i}{2^B} V_{ref}$$

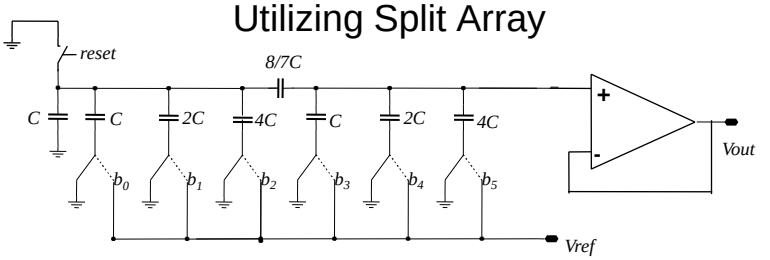
- Opamp helps eliminate the parasitic capacitor effect by producing virtual ground at the sensitive node since C_P has zero volts at start & end
  - Issue: opamp offset & speed- also double capacitor area

Charge Scaling DAC Incorporating Offset Compensation



- During reset phase:
  - Opamp disconnected from capacitor array via switch S3
  - Opamp connected in unity-gain configuration (S1)
  - C_I Bottom plate connected to ground (S2)
  - $V_{out} \sim - V_{os} \rightarrow V_{CI} = -V_{os}$
- This effectively compensates for offset during normal phase

Charge Scaling DAC Utilizing Split Array



$$C_{series} = \frac{\sum all\ LSB\ array\ C}{\sum all\ MSB\ array\ C} C$$

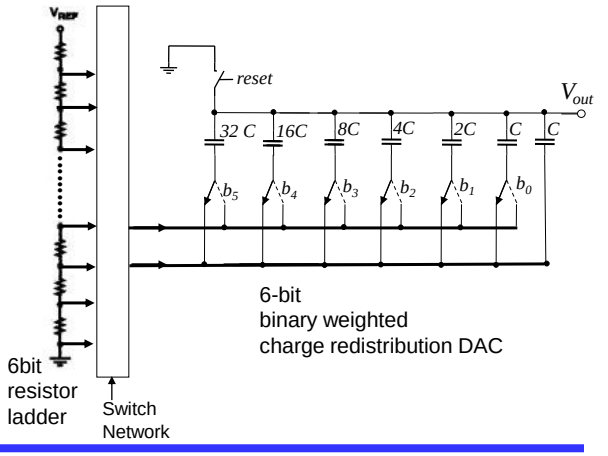
- Split array→ reduce the total area of the capacitors required for high resolution DACs
  - E.g. 10bit regular binary array requires 1024 unit Cs while split array (5&5) needs 64 unit Cs
  - Issue: Sensitive to series capacitance parasitic capacitor

Charge Scaling DAC

- Advantages:
  - Low power dissipation → capacitor array does not dissipate DC power
  - Output is sample and held → no need for additional S/H
  - INL function of capacitor ratio
  - Possible to trim or calibrate for improved INL
  - Offset cancellation almost for free
- Disadvantages:
  - Process needs to include good capacitive material → not compatible with standard digital process
  - Requires large capacitor ratios
  - If binary-weighted Cs used then not inherently monotonic (more later)

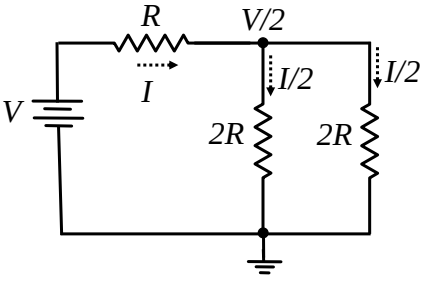
Segmented DAC
Resistor Ladder (MSB) & Binary Weighted Charge Scaling (LSB)

- Example: 12bit DAC
  - 6-bit MSB DAC→ R- string
  - 6-bit LSB DAC → binary weighted charge scaling
- Component count much lower compared to full R-string
  - Full R string→ 4096 resistors
  - Segmented → 64 R + 7 Cs (64 unit caps)



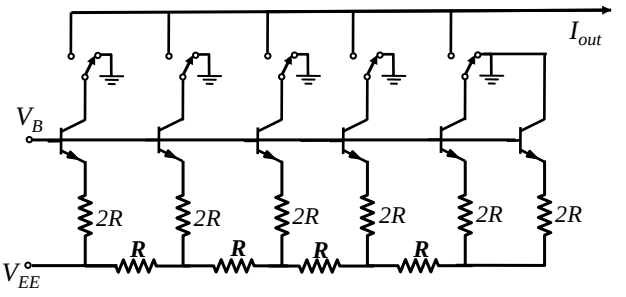
Current Based DACs
R-2R Ladder Type

- R-2R DAC basics:
  - Simple R network divides both voltage & current by 2



Increase # of bits by replicating circuit

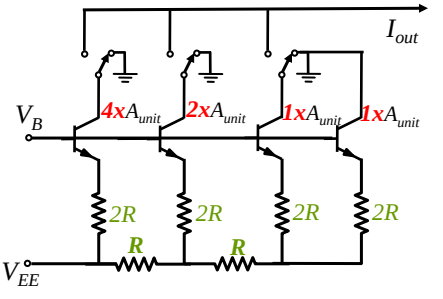
R-2R Ladder DAC



Emitter-follower added to convert to high output impedance current sources

R-2R Ladder DAC
How Does it Work?

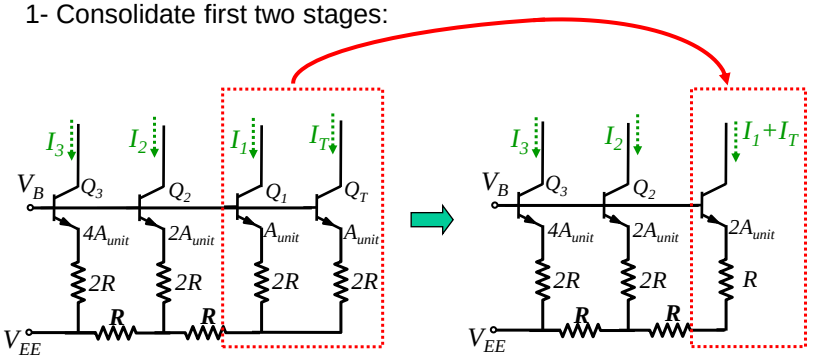
Consider a simple 3bit R-2R DAC:



R-2R Ladder DAC
How Does it Work?

Simple 3bit DAC:

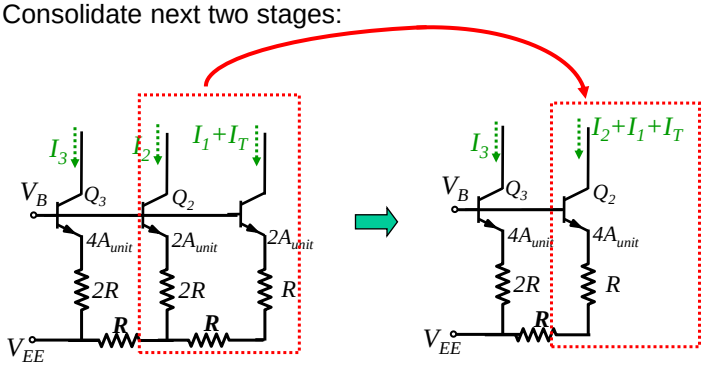
1- Consolidate first two stages:



R-2R Ladder DAC
How Does it Work?

Simple 3bit DAC-

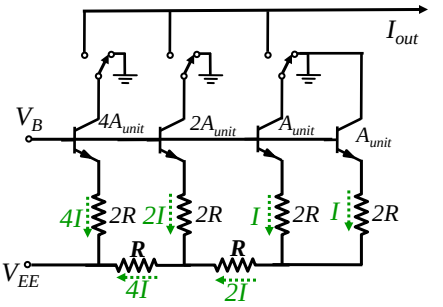
2- Consolidate next two stages:



$$I_3 = I_2 + I_1 + I_T \rightarrow I_3 = \frac{I_{Total}}{2}, I_2 = \frac{I_{Total}}{4}, I_1 = \frac{I_{Total}}{8}$$

R-2R Ladder DAC
How Does it Work?

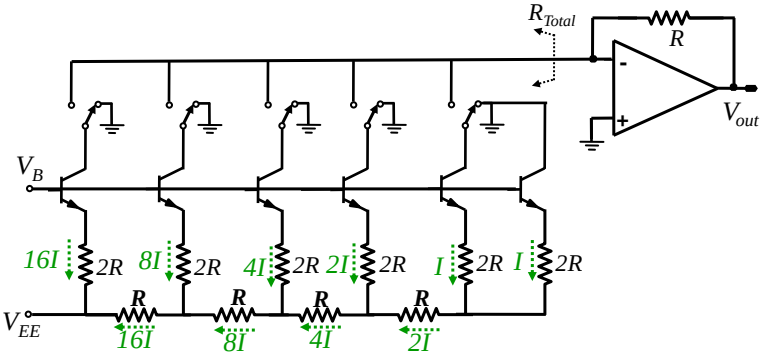
Consider a simple 3bit R-2R DAC:



In most cases need to convert output current to voltage
Note that finite output resistance of the current sources causes gain error only

Ref: B. Razavi, "*Data Conversion System Design",* IEEE Press, 1995, page 84-87

R-2R Ladder DAC



Trans-resistance amplifier added to:

- Convert current to voltage
- Generate virtual ground @ current summing node so that output impedance of current sources do not cause error
- Issue: error due to opamp offset

R-2R Ladder DAC
Opamp Offset Issue

$$V_{os}^{out} = V_{os}^{in}\left(1+\frac{R}{R_{Total}}\right)$$

If $R_{Total} = large,$

$$\rightarrow V_{os}^{out} \approx V_{os}^{in}$$

If $R_{Total} = not\ large$

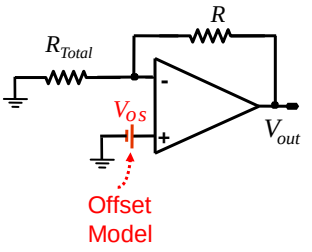
$$\rightarrow V_{os}^{out} = V_{os}^{in}\left(1+\frac{R}{R_{Total}}\right)$$

Problem:

Since R_{Total} *is code dependant*

$\rightarrow V_{os}^{out}$ *would be code dependant*

$\rightarrow$ *Gives rise to INL & DNL*



R-2R Ladder
Summary

- Advantages:
  - Resistor ratios only x2
  - Does not require precision capacitors
- Disadvantages:
  - Total device emitter area → $A_E^{unit} x\ 2^B$
    - → Not practical for high resolution DACs
  - INL/DNL error due to amplifier offset