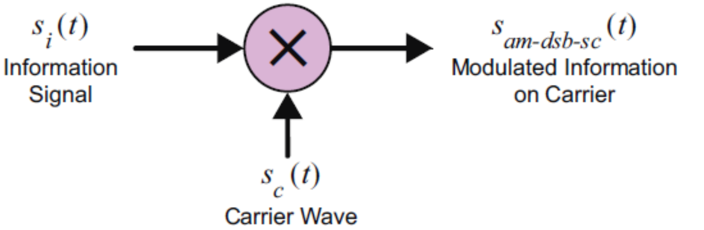
Digital Systems for Telecommunications

Using the Software Defined Radio
for implementing simple analog modulations
The AM case.

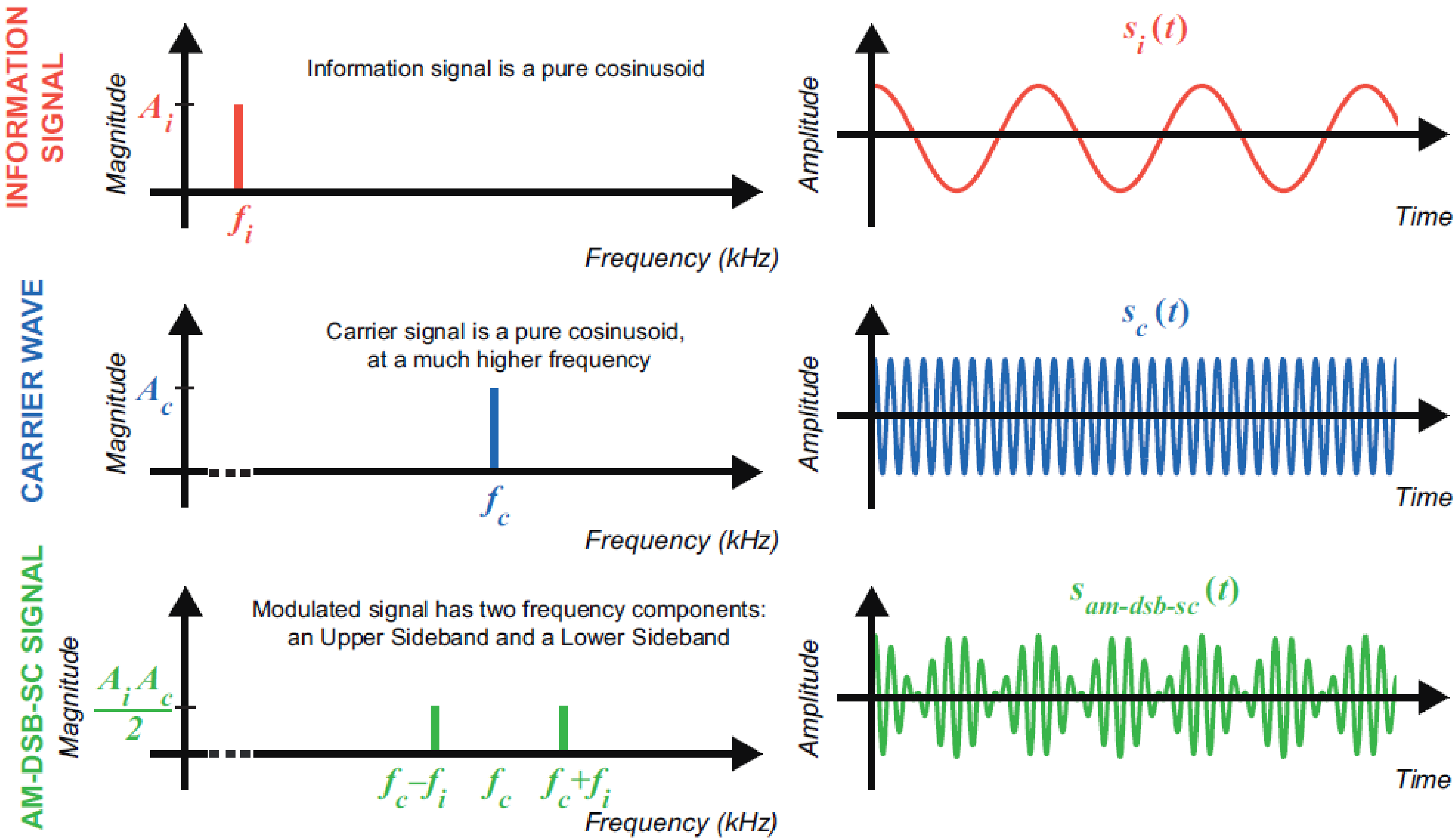
Amplitude Modulation – AM-DSB-SC

- AM is intuitively the simplest modulation method and it also forms the basis of many digital modulation schemes.
- Mathematically, the least complicated of the AM variants is AM-DSB-SC (SC = Suppressed Carrier).
  - An information signal $s_i(t)$ is mixed with a carrier wave $s_c(t)$ to produce the AM signal $s_{am-dsb-sc}(t)$



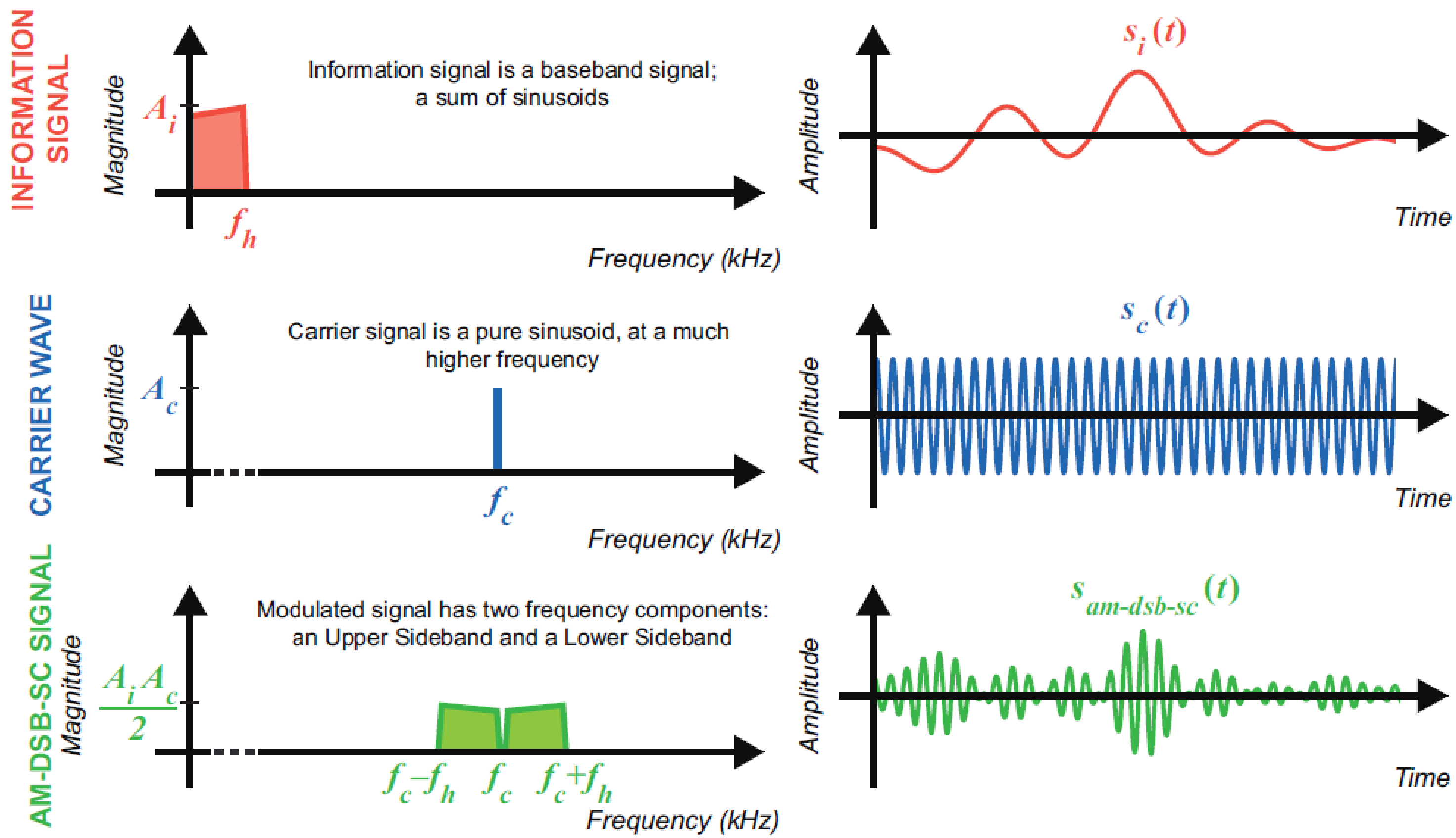
- The maths associated with the modulation of a sinusoidal information signal is the most logical place to start.
  - The information signal is defined as having amplitude Ai and the frequency fi: $si(t) = Ai \cos(2\pi fi\ t) = Ai \cos(\omega i\ t)$
  - The carrier has amplitude A_c and the (higher) frequency f_c: $sc(t) = Ac \cos(2\pi fc\ t) = Ac \cos(\omega c\ t)$
  - Mixing (multiplying) the two signals yields: $s_{am-dsb-sc}(t) = Ai \cos(\omega it)\ Ac \cos(\omega ct) =$
    $= (AiAc/2)[\cos(\omega c - \omega i)t + \cos(\omega c + \omega i)t]$

Amplitude Modulation – AM-DSB-SC



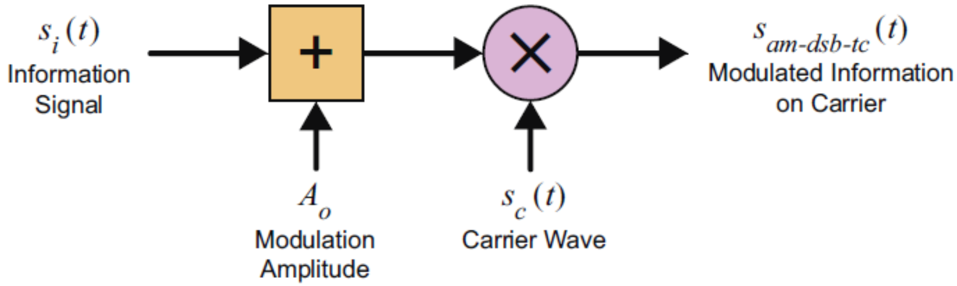
Amplitude Modulation – AM-DSB-SC

- Usually information signals are far more complex.
  - If a baseband information signal had a bandwidth of *fh* Hz, AM-DSB-SC modulating it would result in a signal with a bandwidth of 2*fh* Hz.



Amplitude Modulation – AM-DSB-TC

- AM-DSB-TC (transmitted carrier) is an alternative AM modulation scheme that enables the use of non-coherent demodulators.
  - An information signal si(t) with a DC offset Ao is mixed with a carrier wave sc(t) to produce the AM signal $s_{am-dsb-tc}(t)$.



- If we once again consider a sinusoidal 'information' signal with amplitude and frequency ,

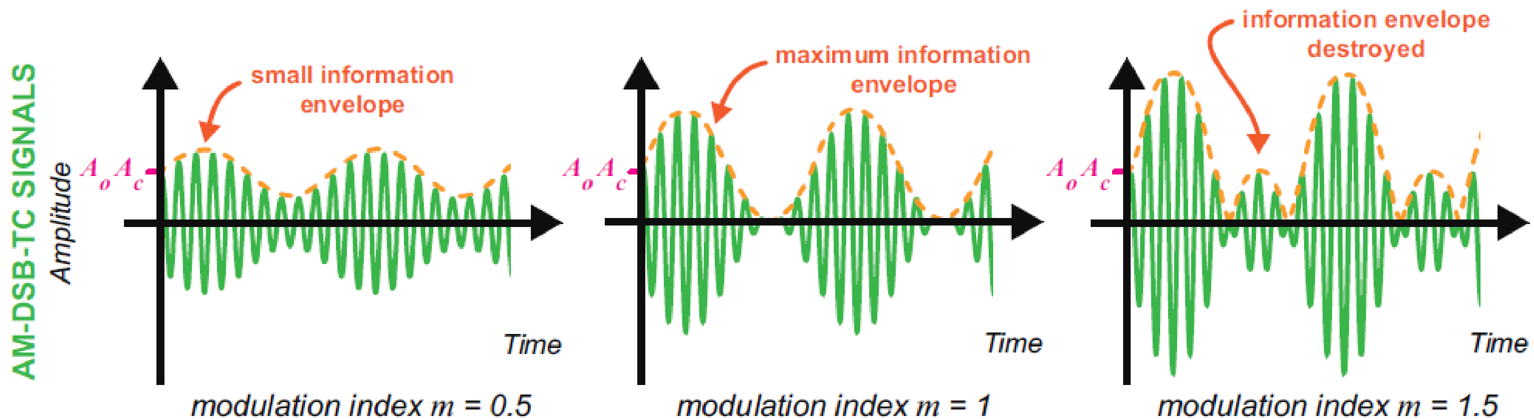
$$
\begin{aligned}
s_{am-dsb-tc}(t) &= \left[A_o + A_i \cos(\omega_i t) \right] A_c \cos(\omega_c t) \\
&= A_o A_c \cos(\omega_c)t + \frac{A_i A_c}{2} \cos(\omega_c - \omega_i)t + \frac{A_i A_c}{2} \cos(\omega_c + \omega_i)t \\
&= A_o A_c \cos(\omega_c)t + \frac{A_i A_c}{2} \left(\cos(\omega_c - \omega_i)t + \cos(\omega_c + \omega_i)t \right) .
\end{aligned}
$$

Amplitude Modulation – AM-DSB-TC

- Sometimes the modulated signal is expressed in terms of the AM modulation index, '*m*'

$$s_{am-dsb-tc}(t) = A_o\left[1 + m\cos(\omega_i t)\right]A_c\cos(\omega_c t) \qquad \text{where: } m = \frac{A_i}{A_o}.$$

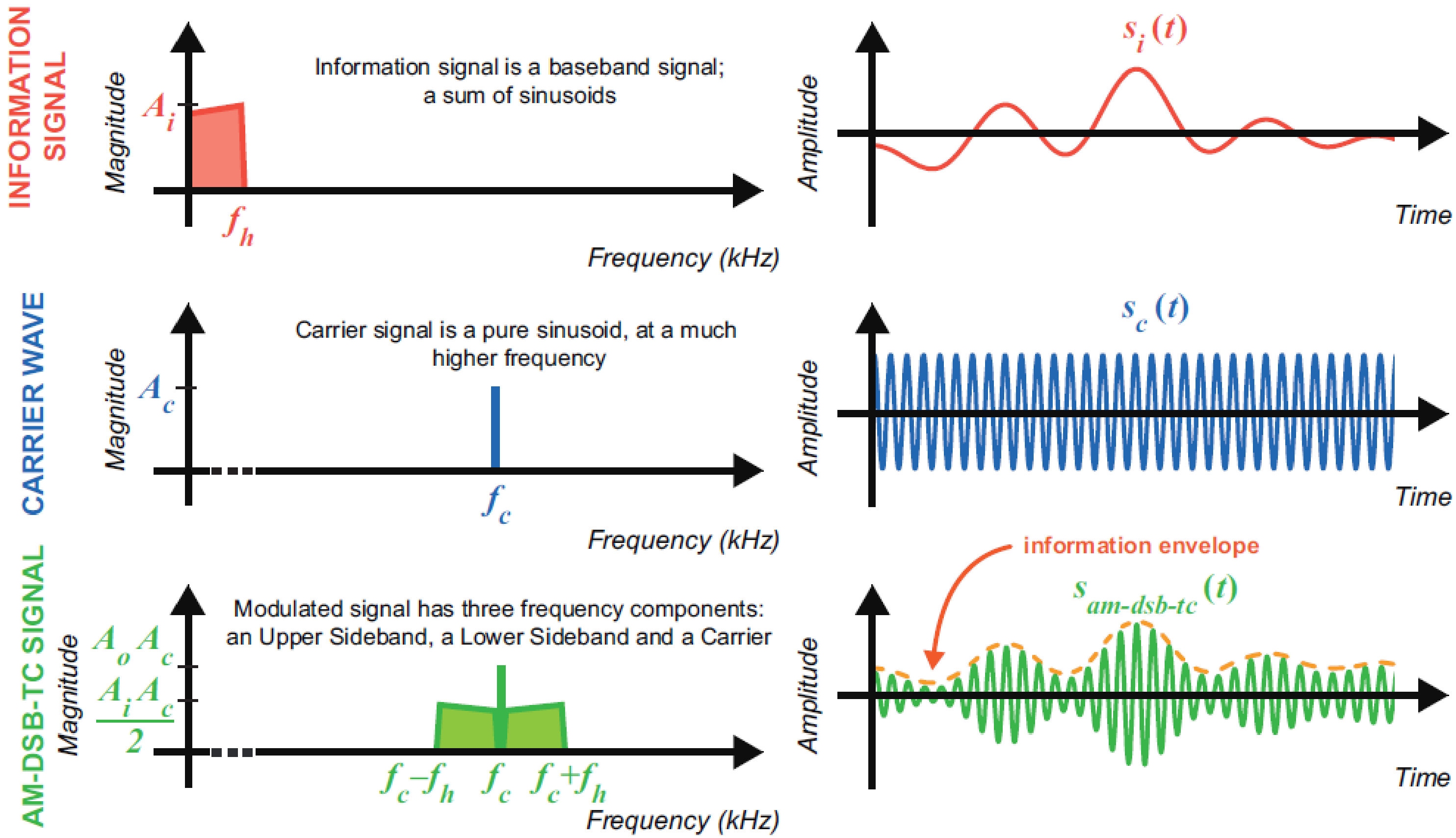
$$= A_o A_c\cos(\omega_c t) + \frac{A_o A_c m}{2}\left(\cos(\omega_c - \omega_i)t + \cos(\omega_c + \omega_i)t\right)$$



- Thus, AM-DSB-TC modulation process results in three sinusoidal terms:
  – a lower sideband at *fc-fi*; an upper sideband at *fc+fi*; and the carrier component at *fc*.

Amplitude Modulation – AM-DSB-TC

- Once again, if a baseband information signal had a bandwidth of *fh* Hz, AM-DSB-TC modulating it would result in a signal with a bandwidth of 2*fh* Hz around the carrier at *fc*.



Receiving and Downconverting AM-DSB-TC Signals

- When an RF $s_{amRF}(t) = s_{am\text{-}dsb\text{-}tc}(t)$ signal is received it is mixed with a complex exponential at frequency flo:

$$s_{bband}(t) = \frac{A_o A_c}{2}\Big[\cos(\omega_c t - \omega_{lo} t) + \cancel{\cos(\omega_c t + \omega_{lo} t)} - j\sin(\omega_c t - \omega_{lo} t) - j\,\cancel{\sin(\omega_c t + \omega_{lo} t)}\Big] + \frac{A_i A_c}{4}\Big[\cos(\omega_c t - \omega_i t - \omega_{lo} t) + \cancel{\cos(\omega_c t - \omega_i t + \omega_{lo} t)}$$
$$+ \cos(\omega_c t + \omega_i t - \omega_{lo} t) + \cancel{\cos(\omega_c t + \omega_i t + \omega_{lo} t)}$$
$$- j\sin(\omega_c t - \omega_i t - \omega_{lo} t) - j\,\cancel{\sin(\omega_c t - \omega_i t + \omega_{lo} t)}$$
$$- j\sin(\omega_c t + \omega_i t - \omega_{lo} t) - j\,\cancel{\sin(\omega_c t + \omega_i t + \omega_{lo} t)}\Big].$$

baseband components

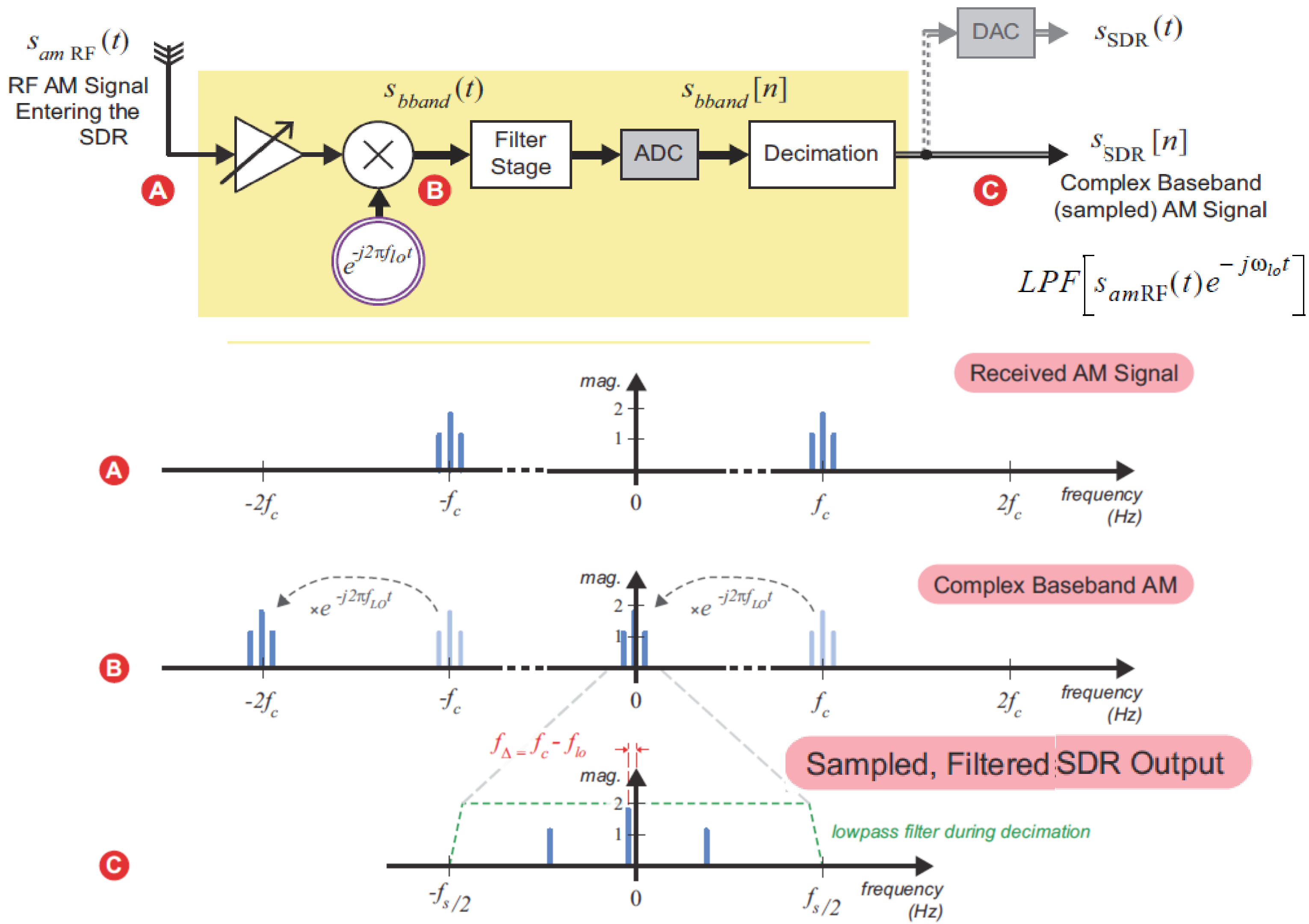
high freq components

- When a frequency offset exists between the modulating carrier and the LO (i.e. $\omega_\Delta t = \omega_c t - \omega_{lo} t$), BB I and Q components are:

$$s_{ip}(t) = \Re e\left[\, s_{bband}(t) \,\right] = \frac{A_o A_c}{2}\cos(\omega_\Delta t) + \frac{A_i A_c}{4}\left[\cos(\omega_\Delta t - \omega_i t) + \cos(\omega_\Delta t + \omega_i t)\right]$$

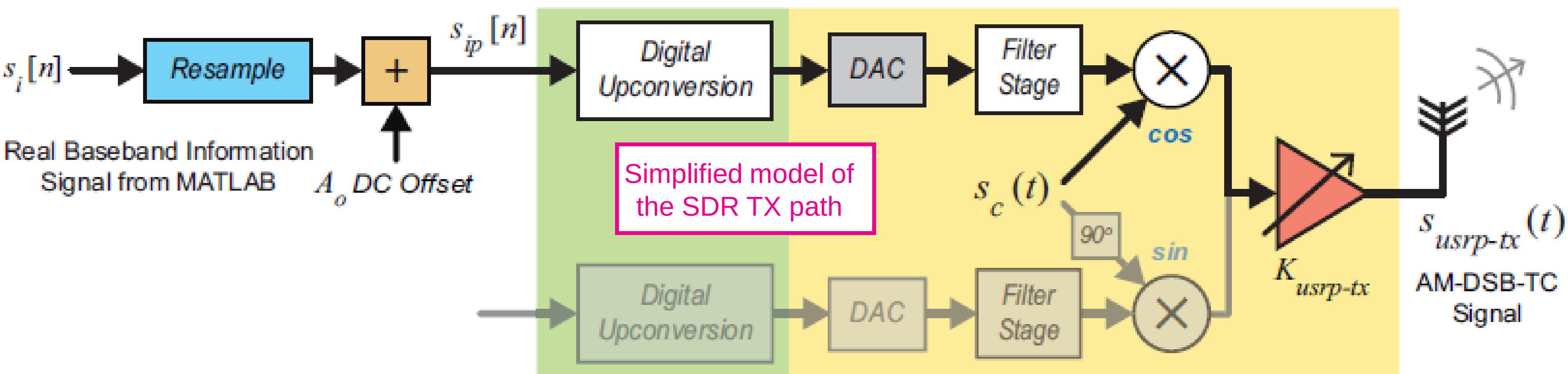
$$s_{qp}(t) = \Im m\left[\, s_{bband}(t) \,\right] = -j\,\frac{A_o A_c}{2}\sin(\omega_\Delta t) - j\,\frac{A_i A_c}{4}\left[\sin(\omega_\Delta t - \omega_i t) + \sin(\omega_\Delta t + \omega_i t)\right]$$

Receiving and Downconverting AM-DSB-TC Signals



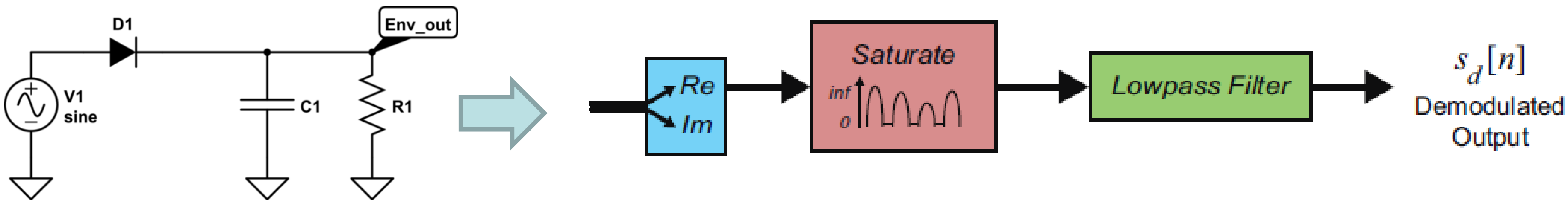
AM modulation TX using SDR

- As any SDR TX path is essentially an AM-DSB-SC modulator, generation of an AM-DSB-SC signal requires only that a real (non-complex) baseband information signal is passed to it!
- For AM-DSB-TC, the DC offset is applied to the information signal before it is mixed with the carrier wave.

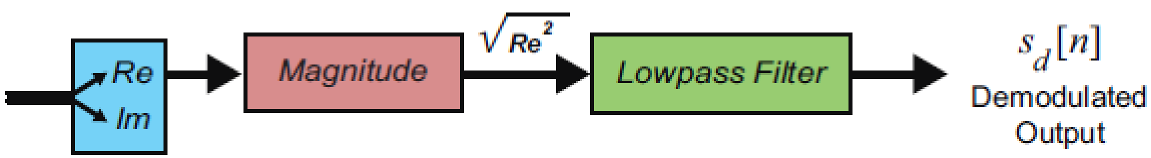


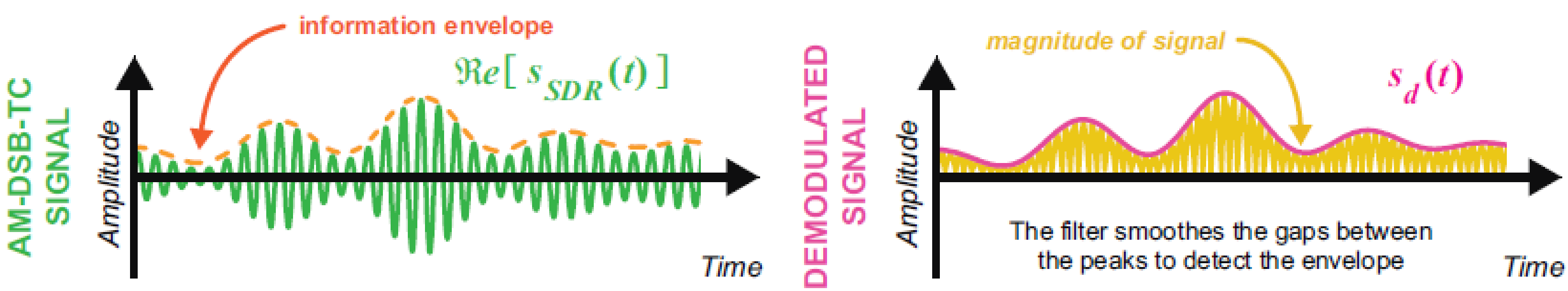
Non-Coherent AM Demod: The Envelope Detector

- A saturating operation can be used to perform the same task as the diode, and an FIR filter can be used to implement the lowpass filter.



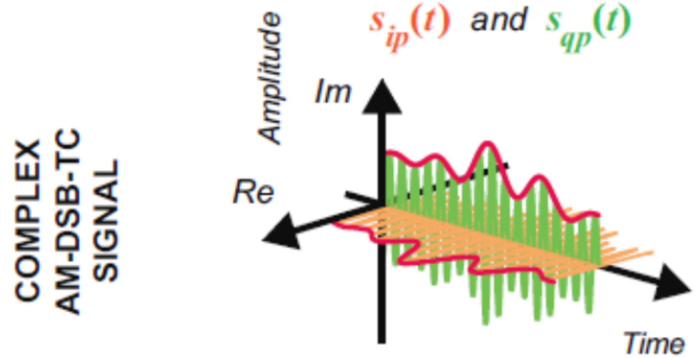
- We can improve the performance by reducing the time gap between the peaks of the carrier, taking the magnitude of the signal.



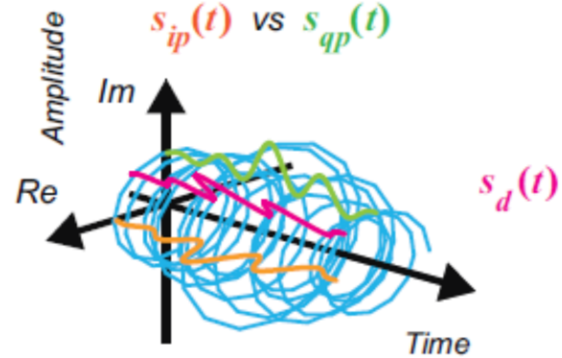


Non-Coherent AM Demod: The Envelope Detector

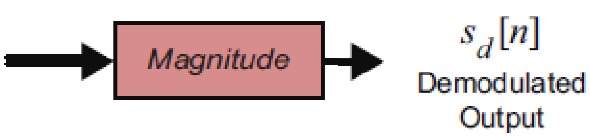
- Both the traditional and the optimised envelope detectors only work with real signals.
- The *complex envelope detector* exploits the magnitude of the complex baseband AM-DSB-TC signal.



When the components of the complex AM signal are plotted individually, they both appear to be AM signals with information envelopes



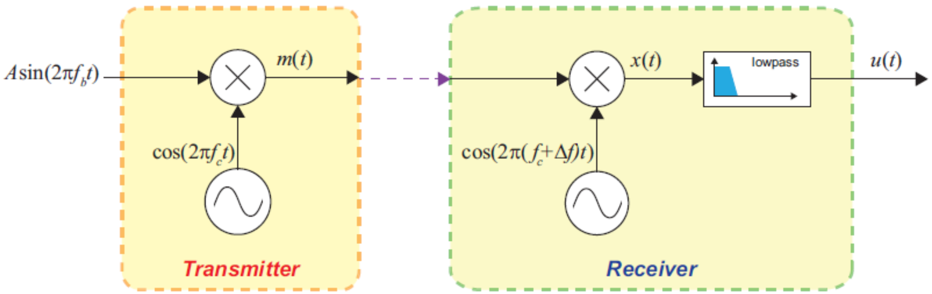
When the complex AM signal is plotted, it appears to look like a spiral travelling through time; the radius of which grows and shrinks in proportion with the envelope of the information signal



The main advantage of this demodulator is that a LPF is not required: finding the magnitude of the complex signal perfectly demodulates it. The output equation of the complex envelope detector is: $s_d[n] = \left| s_{ip}[n] + js_{qp}[n] \right|$

Coherent Demodulation and Carrier Synchrony

- Consider AM-DSB-SC modulation, when demodulation is undertaken using a carrier with a frequency offset *Δf.*



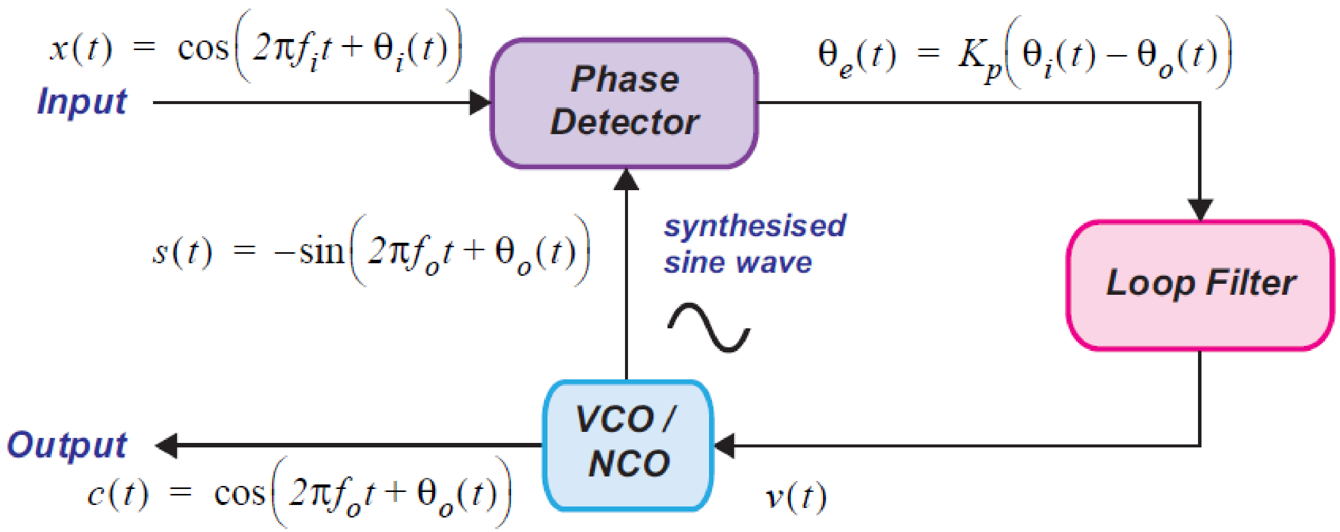
lowpass filtered term

$$\begin{aligned} u(t) &= \frac{A}{4}\sin\left(\cancel{2\pi(2f_c+f_\Delta+f_b)t}\right) - \frac{A}{4}\sin\left(2\pi(f_\Delta-f_b)t\right) \\ &\quad - \frac{A}{4}\sin\left(\cancel{2\pi(2f_c+f_\Delta-f_b)t}\right) + \frac{A}{4}\sin\left(2\pi(f_\Delta+f_b)t\right) \\ &= \frac{A}{4}\left[\sin(2\pi(f_b+f_\Delta)t) + \sin(2\pi(f_b-f_\Delta)t)\right] \end{aligned}$$

- ***Coherent demodulation*** is required, which involves synthesising a sine wave with the same frequency and phase as the carrier of the received signal, and using it to demodulate the received signal.
  - In DSB-TC AM, a coherent receiver extracts the transmitted carrier using a PLL, and uses it to demodulate the received signal to baseband.
  - In DSB-SC AM the receiver has to recover the carrier from the modulated signal that it receives, and then demodulate. The architecture commonly used for this purpose is the Costas Loop.

PLL for coherent demodulation of DSB-TC AM

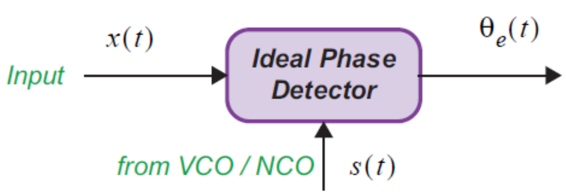
- A PLL includes:
  - A **Controllable Oscillator**, generating the reference sinusoidal output
  - A **Loop Filter**, acting upon the output of the Phase Detector to remove unwanted high frequency terms, and produce the signal that drives the oscillator.
  - A **Phase Detector**, generating a signal that varies in proportion to the difference in phase between the incoming signal, and the locally generated sine wave.



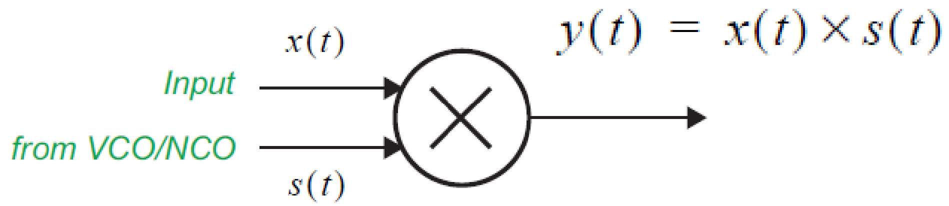
PLL – phase detector

- The role of the phase detector is to generate a signal that varies in proportion to the difference in phase between the reference input signal, and the locally generated sine wave:

$$x(t) = \cos\left(2\pi f_i t + \theta_i(t)\right)$$



$\equiv$



$$y(t) = x(t) \times s(t)$$

$$s(t) = -\sin\left(2\pi f_o t + \theta_o(t)\right) \qquad = \frac{1}{2}\left[\sin\left(2\pi(f_i - f_o)t + (\theta_i(t) - \theta_o(t))\right) - \sin\left(2\pi(f_i + f_o)t + (\theta_i(t) + \theta_o(t))\right)\right]$$

- When fi = fo (syntonized):

$$y(t) = \underbrace{\frac{1}{2}\sin\left(\theta_i(t) - \theta_o(t)\right)}_{\text{low frequency term}} - \underbrace{\cancel{\frac{1}{2}\sin\left(4\pi f_i(t) + \theta_i(t) + \theta_o(t)\right)}}_{\text{high frequency term}}$$

- If the phase difference is small:

$$\sin(x) \approx x$$

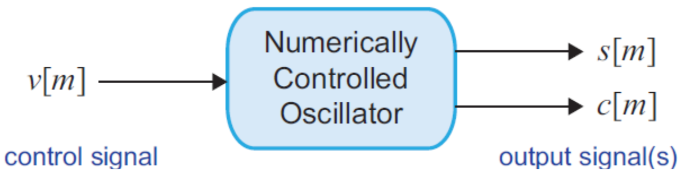
$$y(t) \approx \frac{1}{2}\left(\theta_i(t) - \theta_o(t)\right) \approx \theta_e(t) = K_p\left(\theta_i(t) - \theta_o(t)\right)$$

PLL – controllable oscillator

- The VCO/NCO is an oscillator with a standard or quiescent frequency fo, and a control input v that can adjust the output frequency upwards or downwards.
  - $\hat{\theta}[m] = K_o \sum_{m=0}^{M} v[m]$ is the estimated phase at time t, generated by integrating the control input to the VCO over all time from t = 0
- Let's suppose the phase is incremented by fixed, constant value μ: it determines the number of samples for a 2π phase increment, i.e. the frequency of c[m].

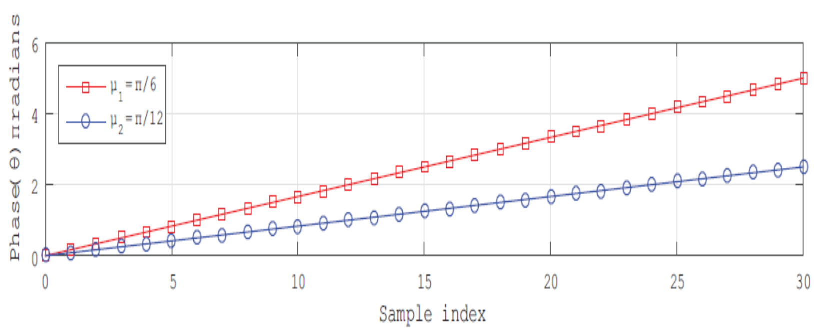
$$c[m] = \cos\left(2\pi f_o m T + \hat{\theta}[m]\right)$$

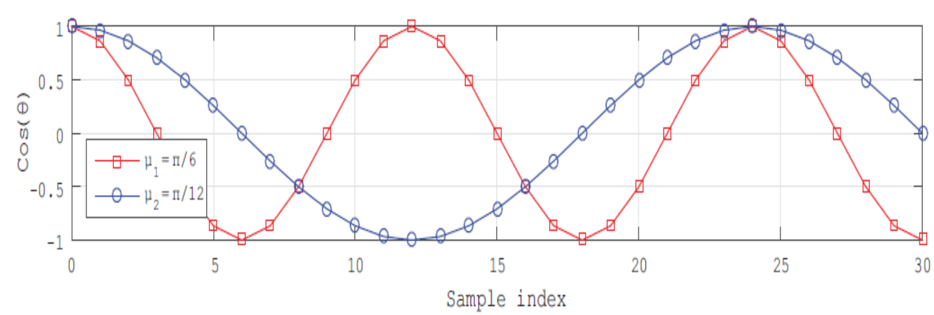
$f_s = 1/T$ Hz



$$\hat{\theta}[m] = K_o \sum_{m=0}^{M} v[m] = \hat{\theta}[m-1] + \mu[m]$$

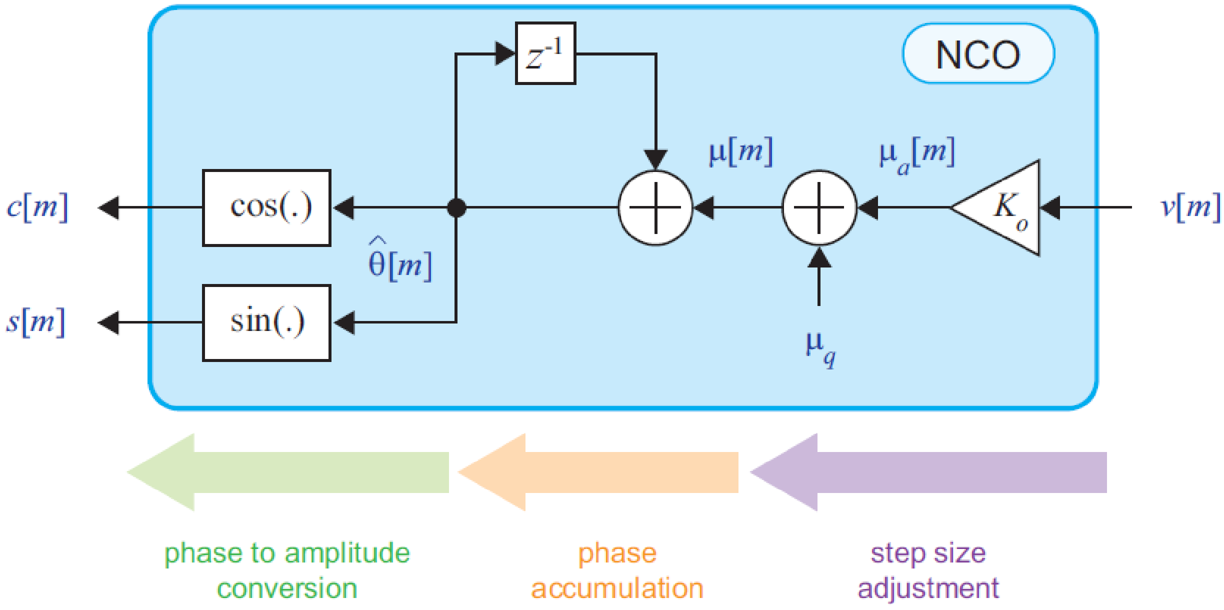
$$\mu = \frac{2\pi f_d}{f_s}$$





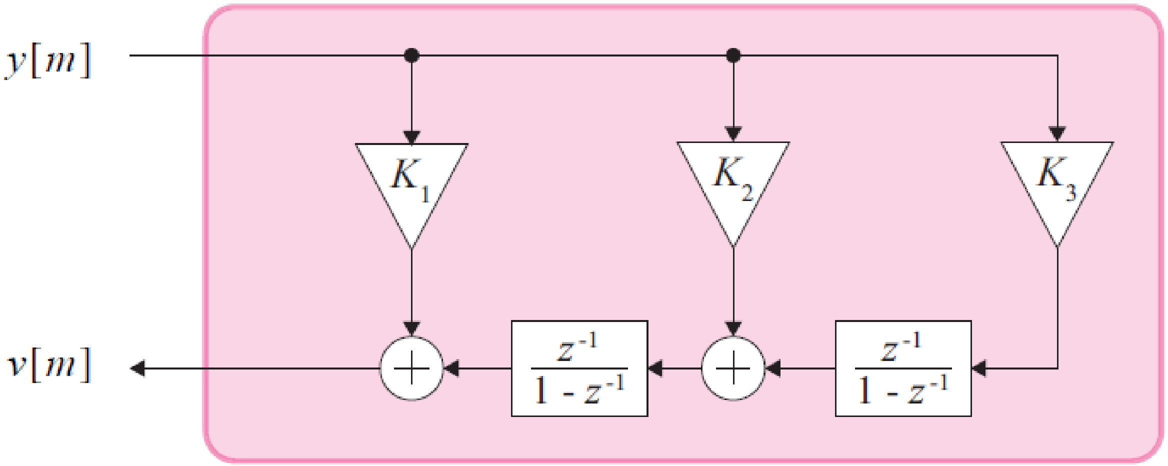
PLL – controllable oscillator

- We need a dynamically controllable cosine wave generator
  - the step size at sample m is formed by summing two components of the step size: μq, a constant value corresponding to the quiescent frequency; and μa, representing the adjustment term which may vary over time $\mu[m] = \mu_q + \mu_a[m]$ $\mu_a[m] = K_o v[m]$
  - Ko is the oscillator gain; larger Ko value makes the NCO more sensitive to changes in the feedback signal



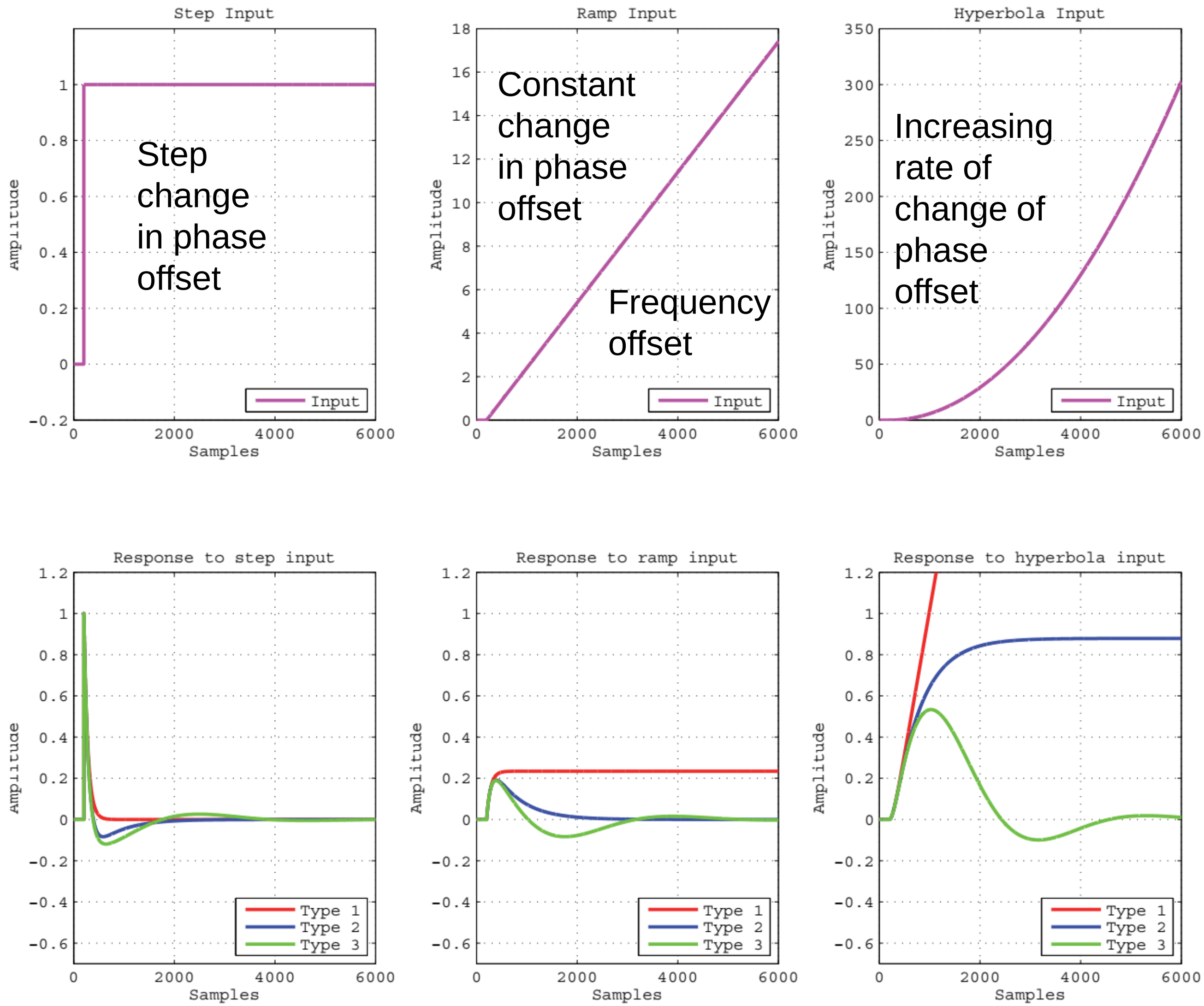
PLL – loop filter

- The Loop Filter has the task of filtering the error signal produced by the phase detector
  - In most cases it is a simple LPF, composed of a proportional path, and one or more integral paths
  - The PLL Type corresponds to the number of integrators in the loop, including the NCO and the loop filter



- **Type 1** — K_1 has a significant value; both K_2 and K_3 are set to zero (or branches are omitted);
- **Type 2** — K_1 and K_2 have significant values; K_3 is set to zero (or branch is omitted);
- **Type 3** — all coefficients have significant values.

PLL – loop filter



- **Only Type 3** can adapt in the presence of an initial phase difference, an initial frequency difference, and a dynamic frequency offset with zero residual phase error

PLL – parameters

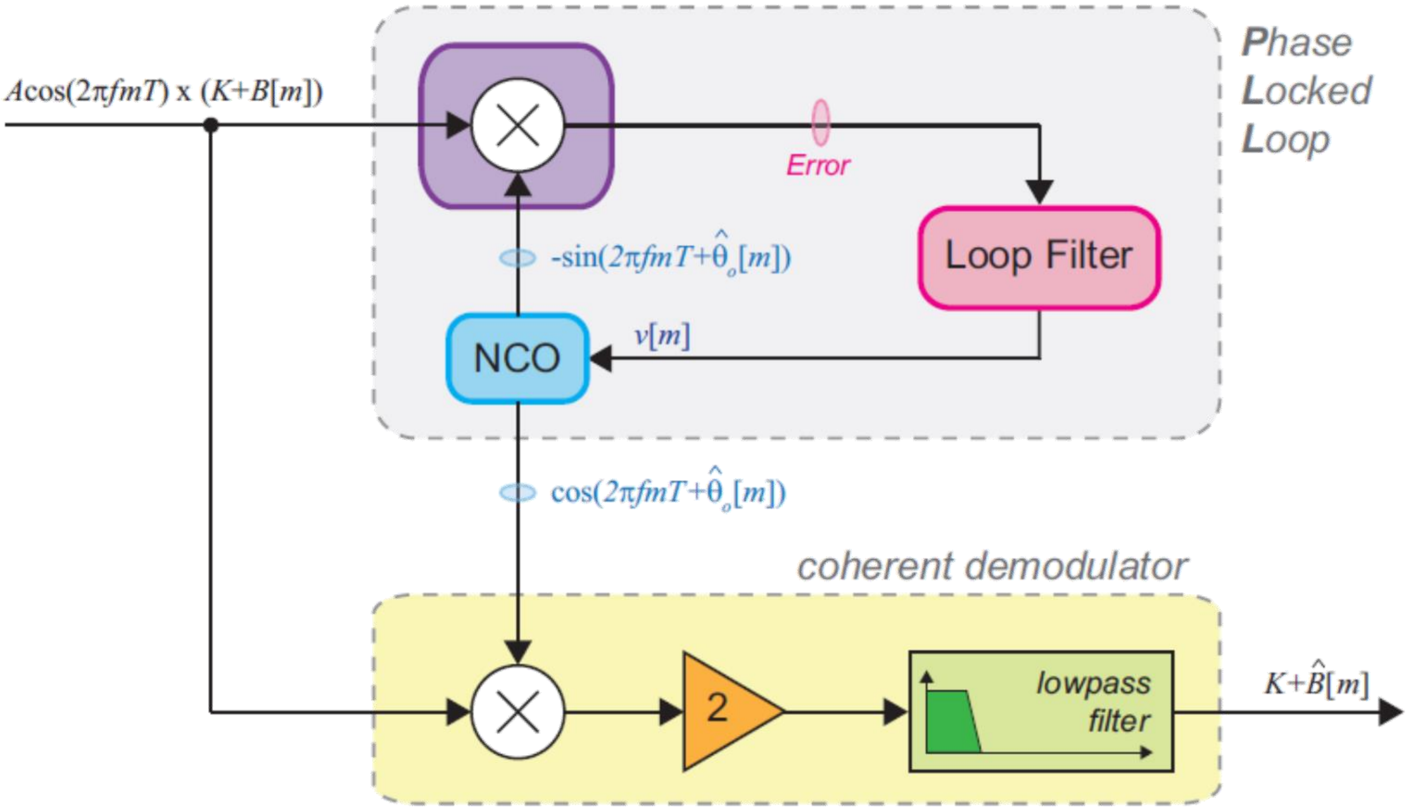
- A few key characteristics of PLLs can be defined:
  - Time to achieve lock; If the PLL has to lock to a frequency far away from the ‘expected’ value, it will take longer; For instance, if it takes 10 symbol periods to achieve lock, this places a minimum length on the preamble (a signal that can be used to ‘wake up’ and help receivers synchronise to incoming information) that precedes information transmissions.

  - Steady state error, influenced by the choice of loop filter; Type 2/3 ensures a zero phase and frequency error in the steady state.

  - Transient behaviour and tracking capability; there is an interaction between the bandwidth of the loop, and the initial deviation between the input and reference frequencies. For any given deviation in frequency, a PLL with a narrower bandwidth will take longer to adapt. The damping ratio of the PLL affects the pattern of the adaption behaviour, including its speed and the extent of overshoots.

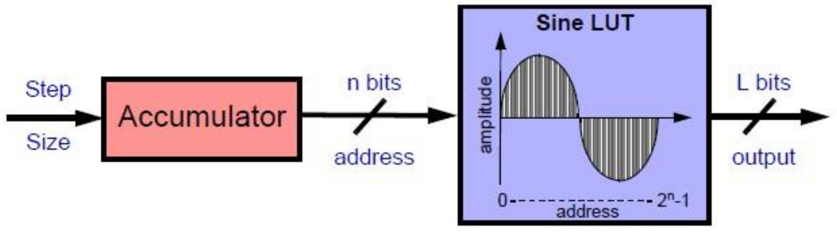
PLL for coherent demodulation of DSB-TC AM

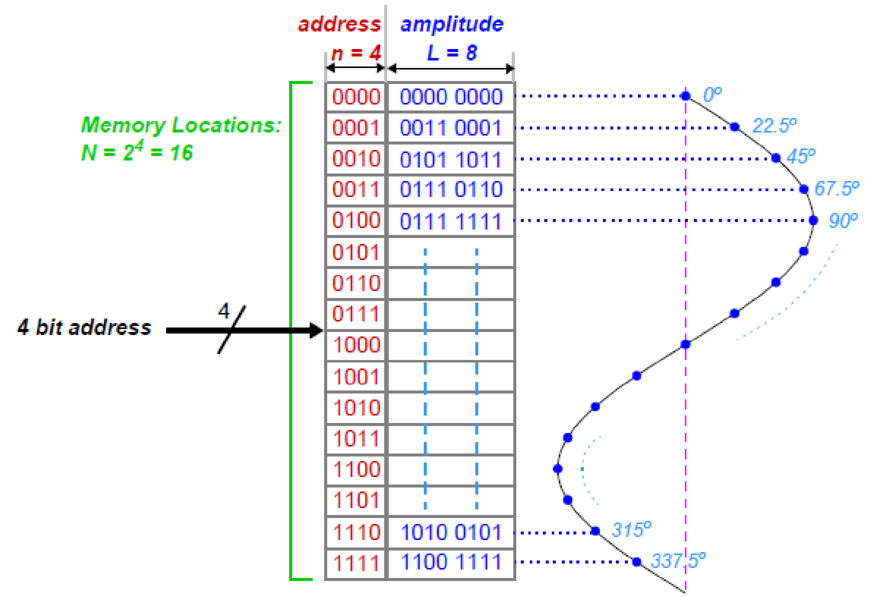
- In DSB-TC AM, a coherent receiver extracts the transmitted carrier component, and uses a PLL to track it and to demodulate the received signal to baseband.



NCO: USE and IMPLEMENTATION ON FPGA

- The simplest operation inside FPGA could be digital mixing, using an NCO and a multiplier.
- Phase to amplitude conversion is usually implemented on LUTs. The LUT has $N=2^n$ size, where n is the numbers of bits that accumulator generate.
- The amplitude resolution of the signal depends on the number of outputs bits L, and the frequency resolution depends on LUT size, N.



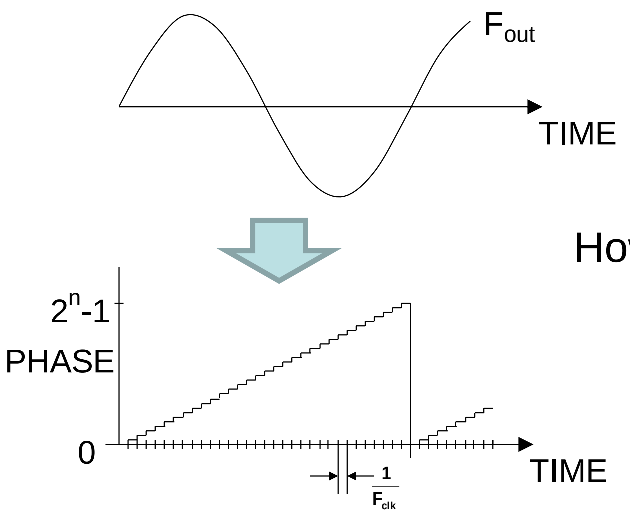


Stored Signal Synthesis

- Concept: fill a delay line with one cycle of the desired waveform, then "loop" the signal over and over to produce the extended output signal
  - Vary the frequency by decimation / interpolation of the stored samples
  - Vary the amplitude by multiplying a slowly varying *envelope* function
- **Structure :**
  - Based on a single crystal oscillator, which generates a reference clock frequency
  - Use of accumulator = Adder + Register
  - ROM to store values
- **Function :**
  - The output of the accumulator takes the form of an address used by a ROM LUT that contains the waveform samples.
  - The number of clock-cycles needed to step through the entire ROM LUT defines the time-period of the waveform along with the step-size Δ_r
  - LUT contents – digital representation of the desired waveform – digital words describing the amplitude of the waveform as a function of phase
  - Address generated by the adder represents the phase value of the waveform

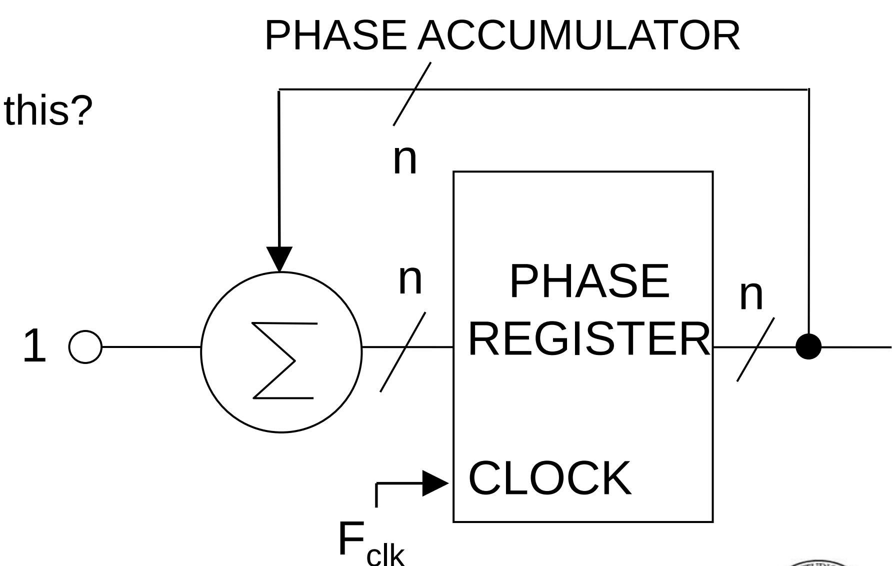
Phase and Time

- Within a period of a sinewave, the phase is directly proportional to time. The same is true in a sampled system, it takes 2^n clock periods to give one output period, being Fsig=Fclk/2^n.
- Frequency is the first derivative of phase, and, correspondingly, phase is the time integral of frequency



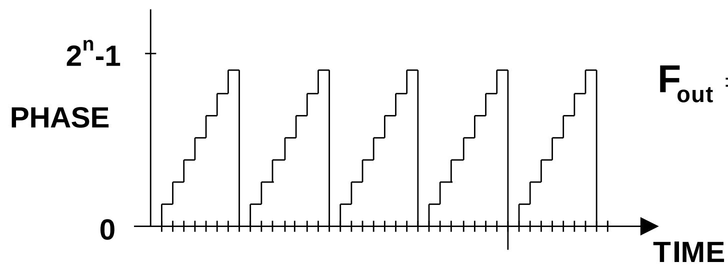
$$F_{out} = \frac{F_{clk}}{2^n}$$

How can we build this?

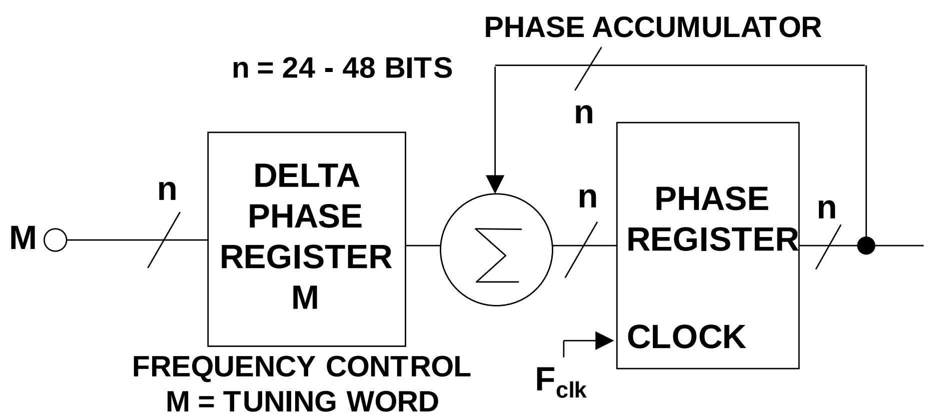


Phase and Time

- We can get a higher frequency by using a larger phase increment.
  - The greater the ***rate*** of change of phase, the greater the frequency
- Note that when we change M, the frequency changes, not the phase (i.e., the accumulator contents).

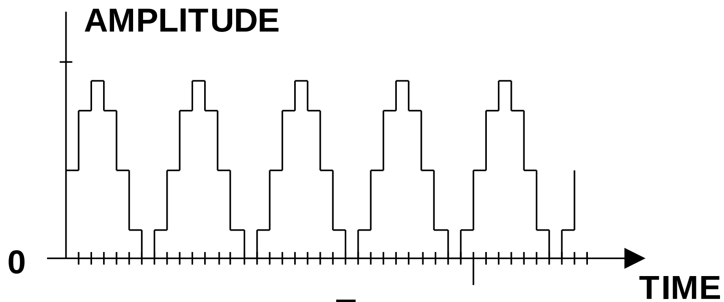


$$F_{out} = M\frac{F_{clk}}{2^n}$$

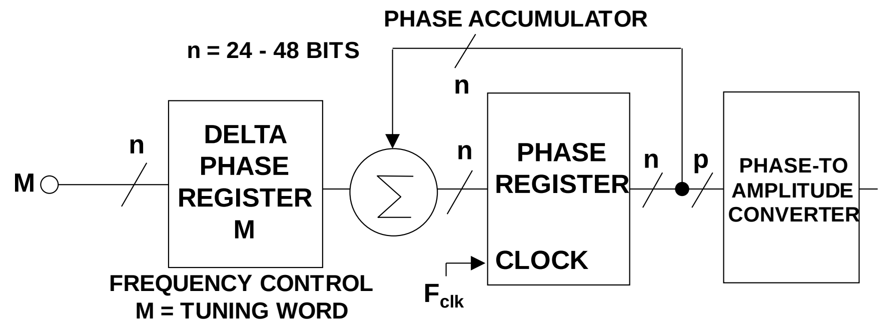


Getting a Sinewave Output

- But of course, we want the output to be amplitude, not phase.
- The phase-to-amplitude converter is often referred to as a sine lookup table but it takes less silicon to perform an algorithm compared to ROM.
  - Still, it takes a lot of silicon to perform this using all n bits (where 24<n<48).
  - Therefore the phase is truncated to p bits first.
  - To convert the amplitude numbers to amplitude voltage, a DAC is placed at the output.

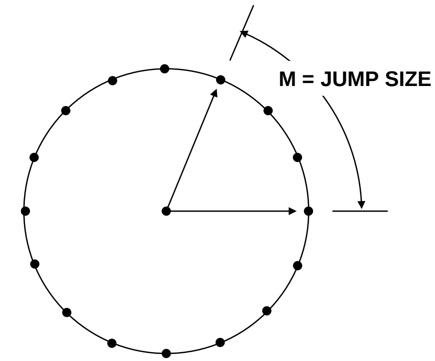


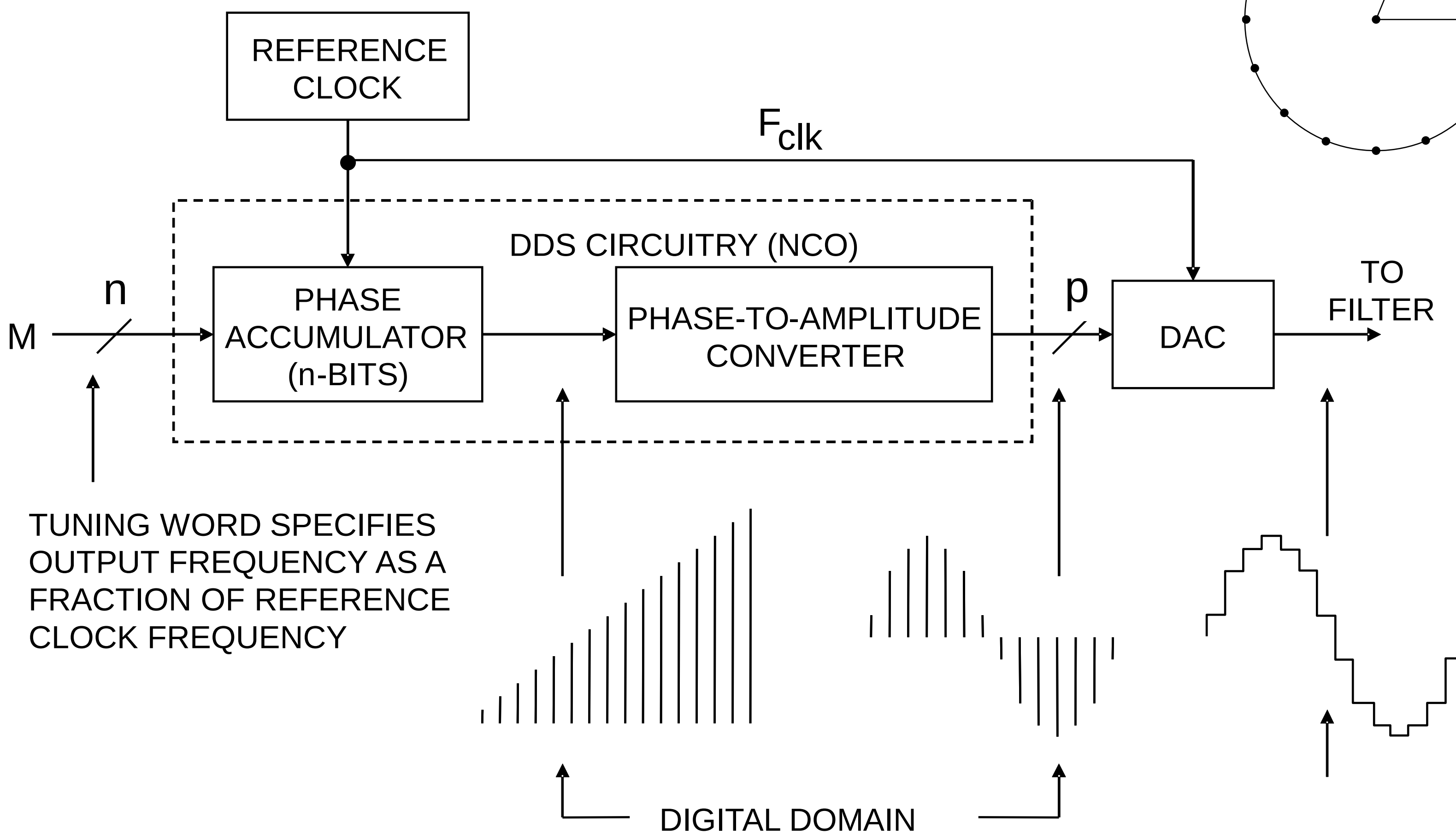
$$F_{out} = M \frac{F_{clk}}{2^n}$$



The DDS architecture

$$F_o = M \frac{F_{clk}}{2^n}$$



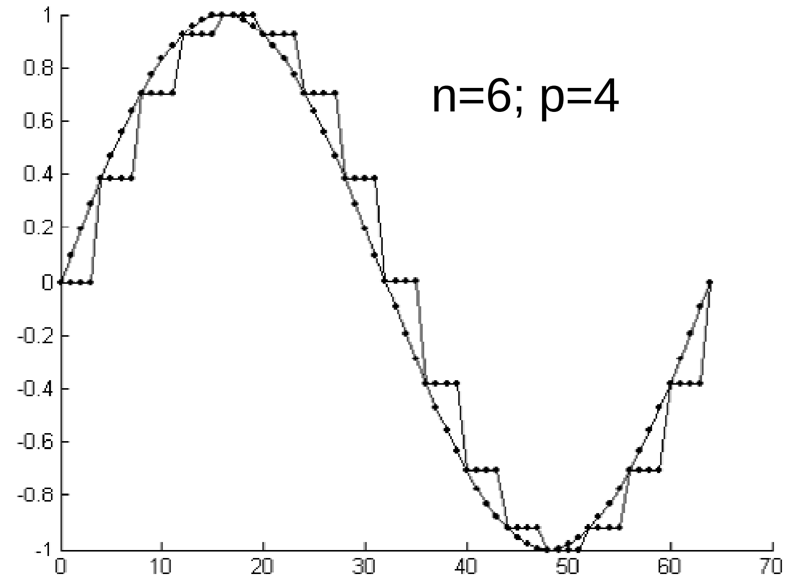


The DDS architecture

- The phase accumulator is actually a modulus 2^n counter.
  - Phase accumulator uses the modulo 2^n property of an n bit accumulator to simulate the modulo 2π property of the sine function
- The magnitude of the increment is determined by the binary input number M in the delta phase register that is summed with the content of the counter→the frequency can be changed instantaneously by simply changing the tuning word M.
  - Nyquist criterion limits the output freq. to half that of the input clock; taking the smallest possible change in M = 1, the freq. resolution can be found as $\Delta F = Fclk / 2^n$
  - Example: Max. output freq. = 2.5 MHz with tuning-step = 1 Hz; Fclk should be 10 MHz
  - Sufficient accumulator width: $2^n = \lceil Fclk / \Delta F \rceil \rightarrow$ smallest n = 24
  - Size of LUT = 2^{24} = 16 MB; having this size of memory on board is costly; LUT size can be reduced by storing only one-fourth the period of the sine-wave and repeating it with necessary sign inversions; still 4 MB
  - Solution – Phase truncation

Phase truncation

- In phase truncation, not all of the n bits of the phase accumulator are used, thereby reducing the size of the lookup table without affecting frequency resolution
- If we choose any p such that p< n, for address of the ROM LUT, then the phase will be held for some time before moving ahead depending upon the value of p.

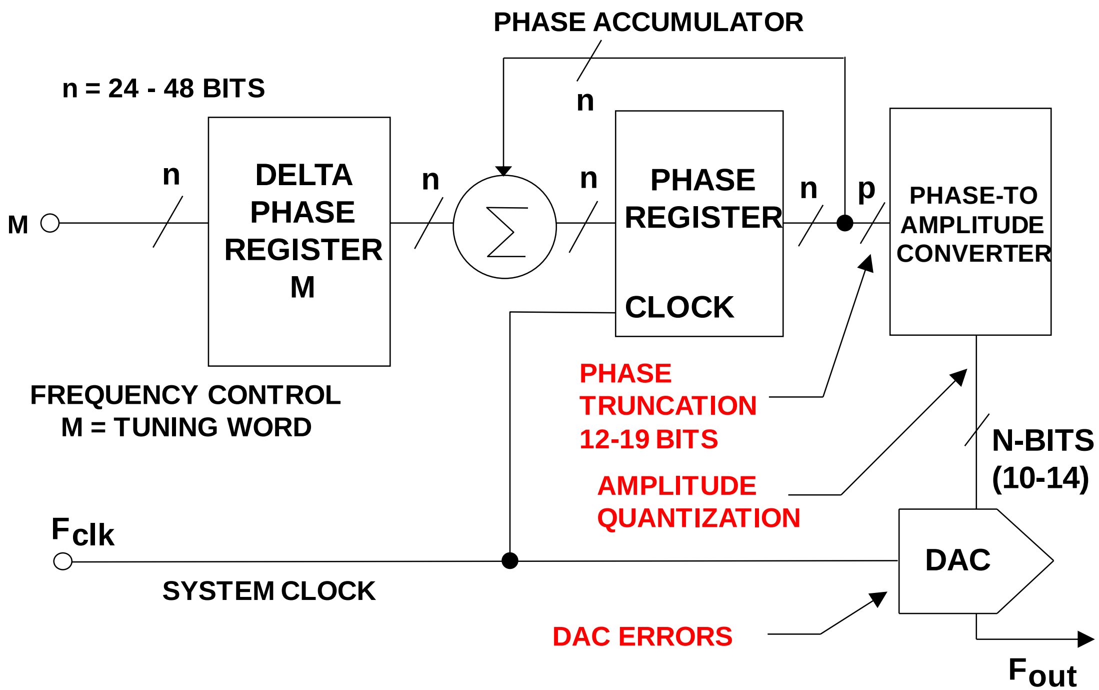


- Example :
  - A 3-bit DDS accumulator, but only 2 of the MSBs (p=2) of the accumulator are used to address a 4*8 ROM

Accumulator Output $N = 3, W = 2$ 3 Bits	Address lines of the ROM (2 MSBs of Accumulator)	Angle (degrees)	Output of the DDS (Contents of ROM)
000	0 0	0	11110000
001	0 0	0	11110000
010	0 1	90	11111111
011	0 1	90	11111111
100	1 0	180	11110000
101	1 0	180	11110000
110	1 1	270	00000001
111	1 1	270	00000001

Errors in a DDS System

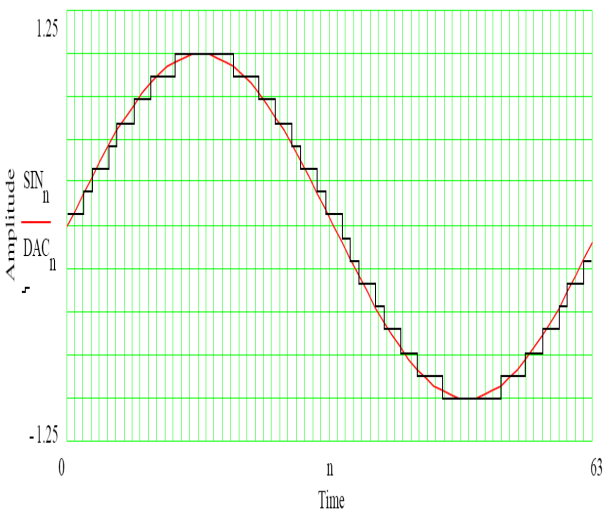
- Three major sources of error :
  - Amplitude truncation because of finite no. of bits
  - Phase truncation; use of truncated no. of bits to address ROM
  - Resolution of DAC; for periodic signals, harmonic spurious frequency components

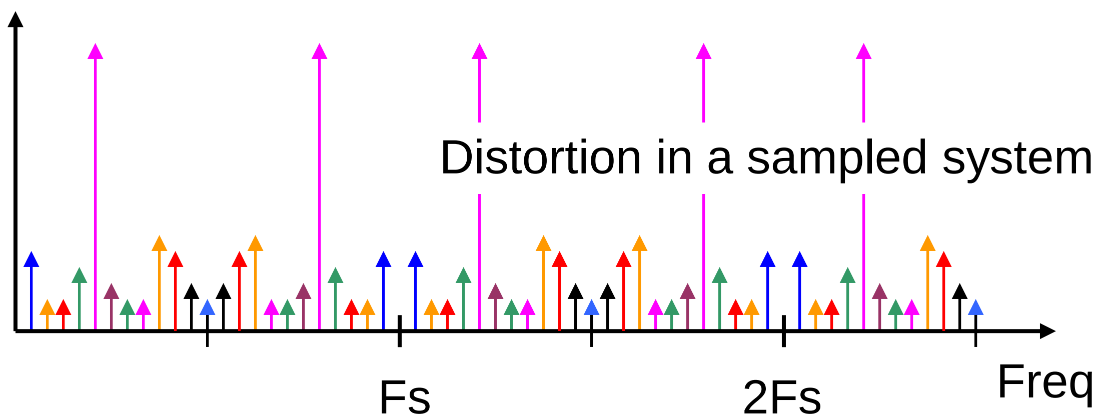


- In general, the phase address information should have two to four bits more resolution than the DAC.
- The objective is to use enough resolution in the lookup table address so that the overall noise and distortion of the analog output signal is limited by the DAC and not the effects of phase truncation

Amplitude errors

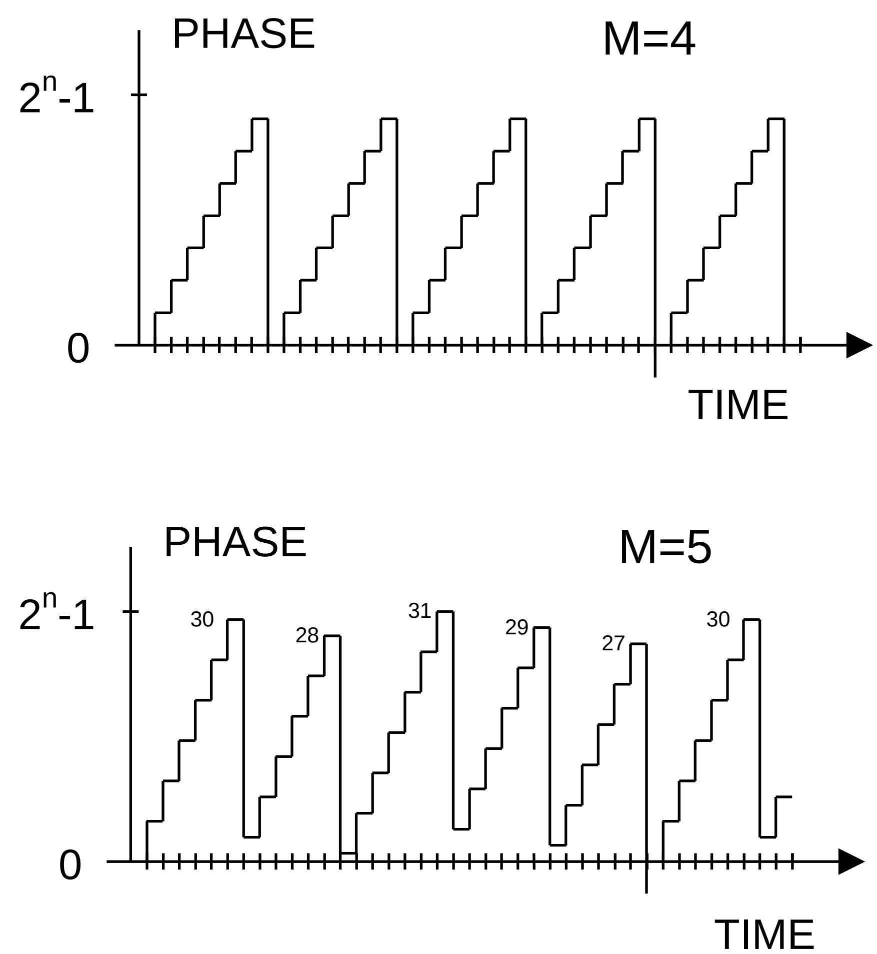
- Quantized waveform ≠ Sinewave
  - Therefore there will be spectral components
  - 6.02n + 1.76 quantization noise is only valid when clock and data are uncorrelated. NOT THE CASE for a DDS!
- DAC non-linearities
  - INL and DNL spurs will alias
  - Harmonics from the analog output stage will NOT alias





Effects of Choosing an Odd Value For M

- For an even value of M, only a few DAC codes are used. Any non-linearity in these codes causes that error to show up over and over and over again (i.e., it will look like a distortion term).
- With an odd value for M, the waveform ‘walks’ thru every possible phase value before it repeats. This exercises ALL the DAC codes and acts to average out the errors (i.e., it acts to dither the signal).

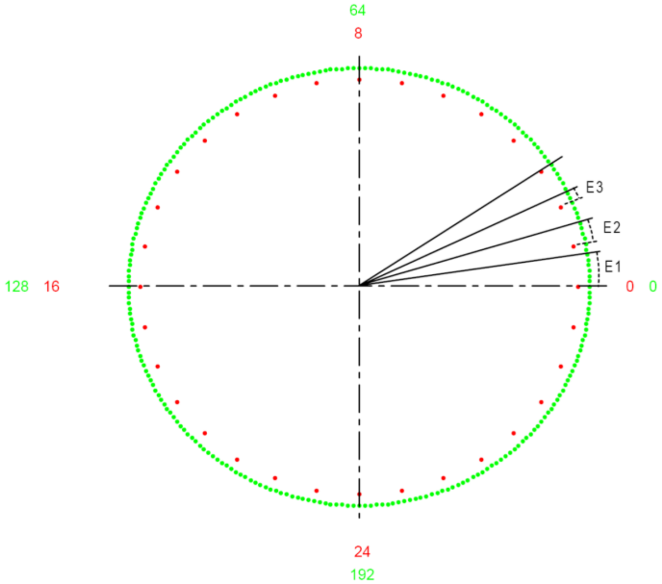


Phase-truncation errors

- No phase-quantization occurs when LSBs which are ignored by ROM are always 0. Thus there are 2^p samples for which phase-quantization does not occur
- Total no. of frequencies generable (Nyquist criteria) vs. Freq. without phase-quantization effect.
  - For n = 32 and p = 14, total combinations possible 2^{31}(at least two samples are needed to represent a periodic waveform), but without phase quantization 2^{13}.
  - As the phase moves around the circle, the error becomes periodic

- Phase error = Amplitude error, due to phase-amplitude converter

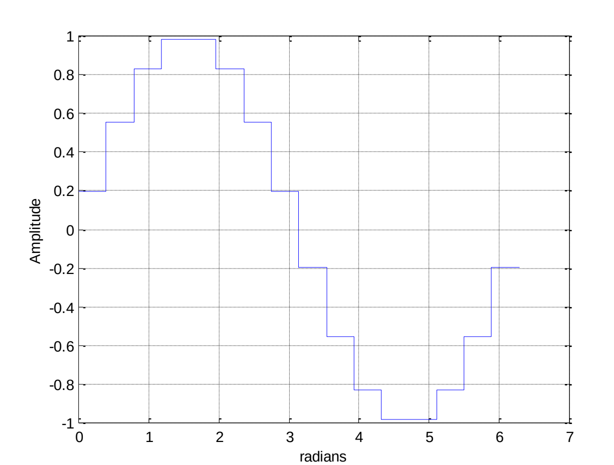
- Periodic phase error = periodic amplitude error = spectral component



Consider n=8; red dots (p=5) allow every 360/32 = 11.25° .
Green dots at M=6 samples the phase every 360/256*6 = 8.4375°

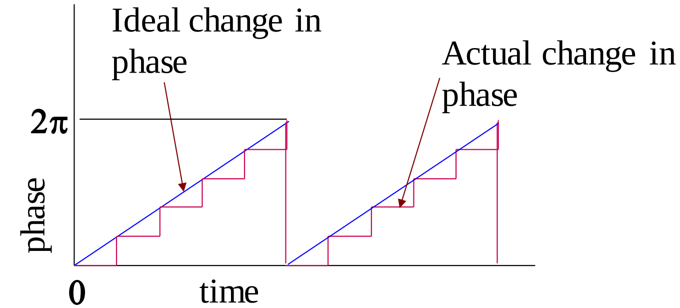
Phase Truncation errors

- Output can be expressed as a series of rectangular pulses
  - Compute the Fourier transform of these pulses
  - Can get very tedious; we will look at some basic analysis



0	000	00	0
1	001		
2	010	01	$\pi/2$
3	011		
4	100	10	π
5	101		
6	110	11	$3\pi/2$
7	111		

$\Delta_r=1$, $n=3$, $Y = 2^3=8$,
$p = 2$, $B = n-p = 1$



Output of DDS can be expressed as

$$y(m) = \sin\left(\frac{2\pi}{2^p}\left[\frac{m\Delta_r}{2^B}\right]\right)$$

Phase Truncation errors

$$y(m) = \sin\left(\frac{2\pi}{2^p}\left\lfloor\frac{m\Delta_r}{2^B}\right\rfloor\right) = \sin\left(\frac{2\pi}{2^{n-B}}\left\lfloor\frac{m\Delta_r}{2^B}\right\rfloor\right)$$

$$= \sin\left(\frac{2\pi 2^B}{2^n}\left(\frac{m\Delta_r}{2^B} - s(m)\right)\right)$$

$$\text{where } s(m) = \frac{m\Delta_r}{2^B} - \left\lfloor\frac{m\Delta_r}{2^B}\right\rfloor \leq 1$$

$$y(m) = \sin\left(\frac{2\pi m\Delta_r}{2^n}\right) - \frac{2\pi 2^B}{2^n} s(m)\cos\left(\frac{2\pi m\Delta_r}{2^n}\right)$$

$$\sin(A - B) = \sin A \cos B - \cos A \sin B$$

Phase Truncation errors

$$y(m) = \sin\left(\frac{2\pi m \Delta_r}{2^n}\right) - \frac{2\pi 2^B}{2^n} s(m) \cos\left(\frac{2\pi m \Delta_r}{2^n}\right)$$

Desired Output

Spurious Component

- Largest spurious amplitude

$$A_{sp} \leq \frac{2\pi 2^B}{2^n} = \frac{2\pi}{2^p}$$

- Detailed calculation of spurious components requires further analysis

Periodicities

- Even in the absence of phase truncation (n = p), periodicities appear in signal depending on the value of Δ_r

$n=4,\ \Delta_r = 2$

$$\underbrace{0,\ 2,\ 4,\ 6,\ 8,\ 10,\ 12,\ 14,}_{\text{first period}}\ \underbrace{0,\ 2,\ 4,\ 6,\ 8,\ 10,\ 12,\ 14,}_{\text{second period}}\ \ldots$$

Perfectly equal periods

Accumulator Values

$n=4,\ \Delta_r = 6$

$$\underbrace{0,\ 6,\ 12,}_{\text{first period}}\ \underbrace{2,\ 8,\ 14,}_{\text{second period}}\ \underbrace{4,\ 10,}_{\text{third period}}\ \underbrace{0,\ 6,\ 12,}_{\text{fourth period}}\ \underbrace{2,\ 8,\ 14,}_{\text{fifth period}}\ 4,\ 10,\ 0,\ \ldots$$

Different period lengths

Location of Spurs

- Time period of spurious components due to periodic jitter alone

$$T_{spur} = k \frac{T_{clk} 2^{n}}{\gcd(\Delta_r, 2^{n})}$$

k is any integer

gcd = greatest common divisor

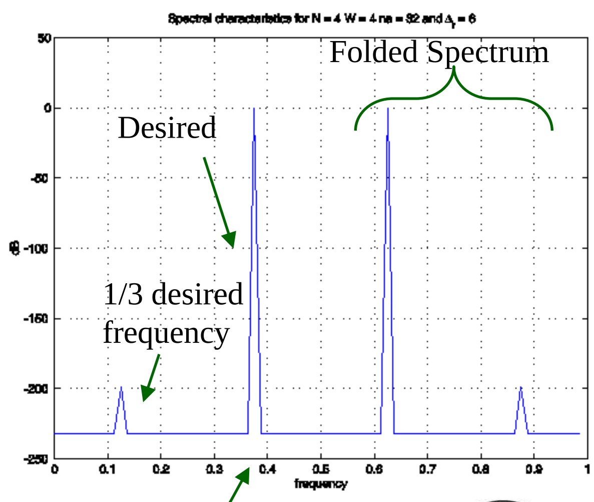
- Example: n=4, Δ_r = 6, p = 4, T_{out} = 16/6T_{clk} *i.e.* T_{clk} = 6/16T_{out}
  - three periods of the fundamental output needed to return to the original state
  - Will create a harmonic at 1/3 of the fundamental
  - Verify from formula:

    Period of spurs = $2^4/\gcd(6,16) T_{clk}$

    = 16/2 T_{clk} = 8T_{clk} = 3*T_{out}

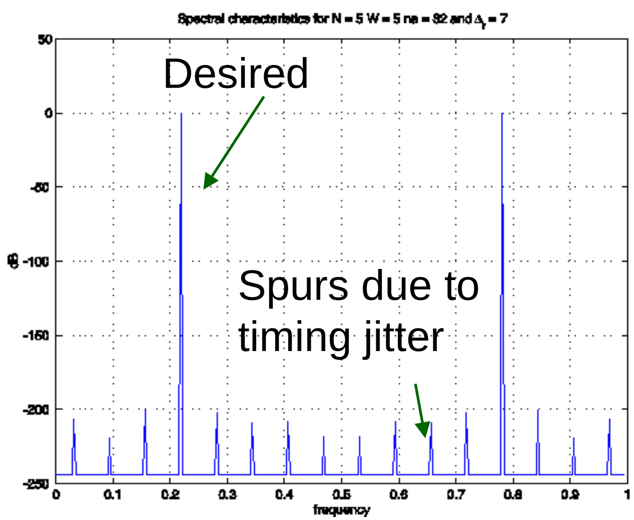
    Thus spur frequency at ±1/3

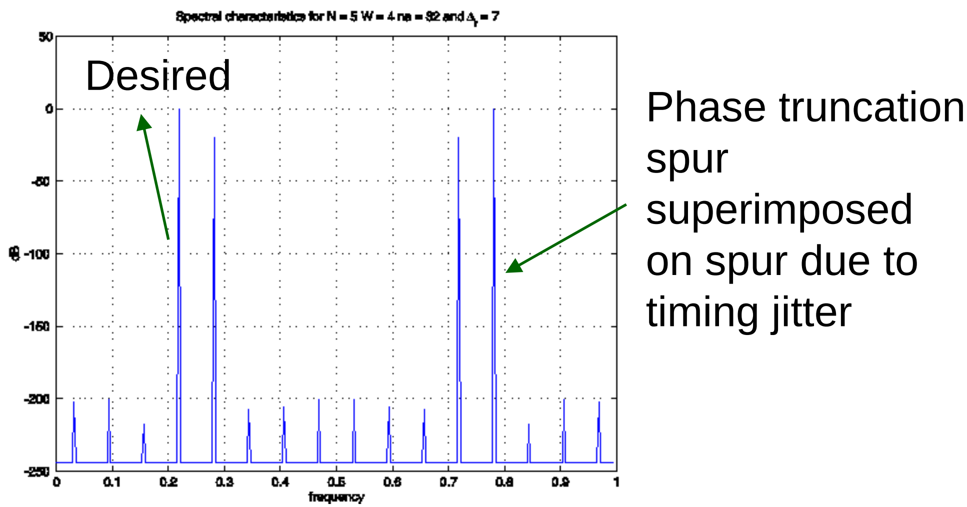
    fundamental and their harmonics exist



Tertiary Periodicities

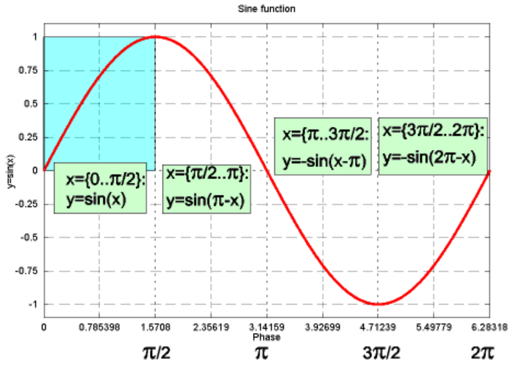
- Presence of a combination of the above three sources of errors could cause additive periodicities which could result in strong spurs
- In the presence of more than one independent set of periodicities, the least common multiple (lcm) of the independent periodicities is another spur frequency
  - Spurs at a particular frequency can be more pronounced than the others

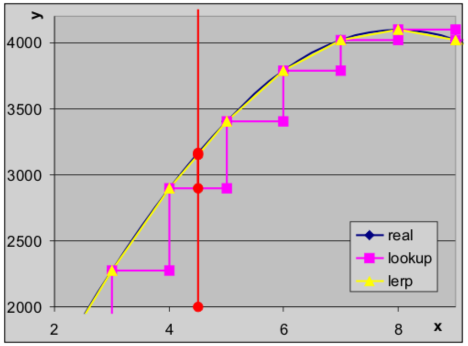




ROM Compression

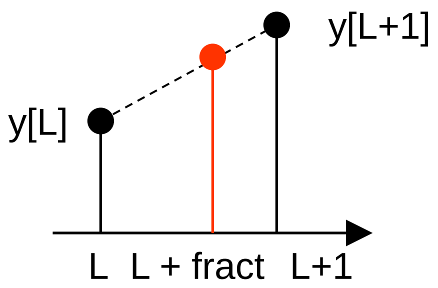
- Spurious signals are one of the main drawbacks of DDS system, especially those caused by phase-truncation – spurious harmonic signals
- Phase-truncation – to avoid a very large ROM
- Phase-truncation can be avoided if it was possible to compress more information into the ROM
- One simple compression approach takes advantage of the symmetry of sine-wave: store only one quadrant of information
  - eliminates 75% of the normal memory requirements
- Other techniques along-with the sine-symmetry – interpolation-based





Linear Interpolation

- Argument (address of LUT) LUA contains desired "location" of next output sample.
  - If LUA:fraction $\neq$ 0, we need to *interpolate* the table.
- Ideally, need to do bandlimited (Shannon) interpolation, but often settle for cheap linear interpolation between adjacent table samples



The computation has three stages:

1. *a=y[L]= LUT[L]=LUT[int(L+fract)]*
2. Slope=*b=LUT[L+1]-LUT[L]*
   =LUT[int(y[L+fract])+1]-a.
3. y[L+fract]=*a+b ✕fract*

A single LUT can be accessed twice and b postcomputed, or two LUT's in parallel where the second LUT contains the precomputed b.

$$y[L+fract]=y[L]+fract\cdot(y[L+1]-y[L])$$

Costas Loop for coherent demodulation of DSB-SC AM

- The Costas loop has two branches (I and Q); an error signal is derived by combining them, then it is filtered to drive the oscillator.