HW Butterfly FFT Sine/Cosine Generator

20170805

Wilson ROM based Sine/Cosine Generation

[24] Fu & Willson Sine / Cosine Generation

ROM-based

For high resolution, ROM size grows exponentially

quater-wave symmetry

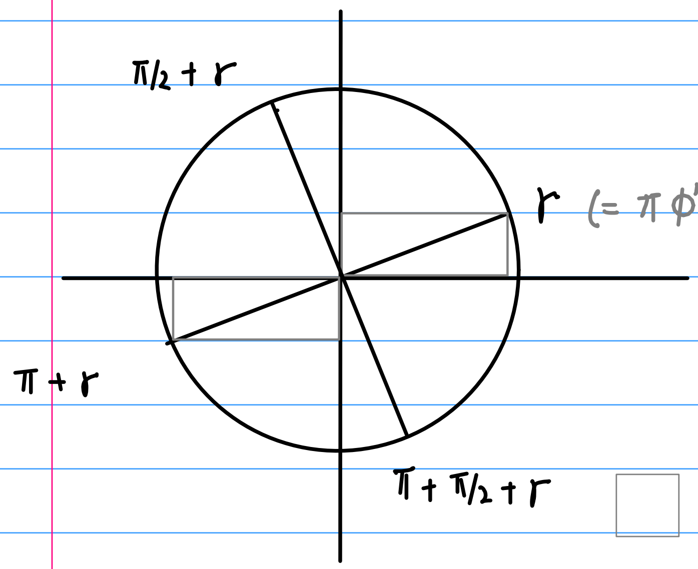
$$\sin\theta = \cos\left(\frac{\pi}{2} - \theta\right)$$

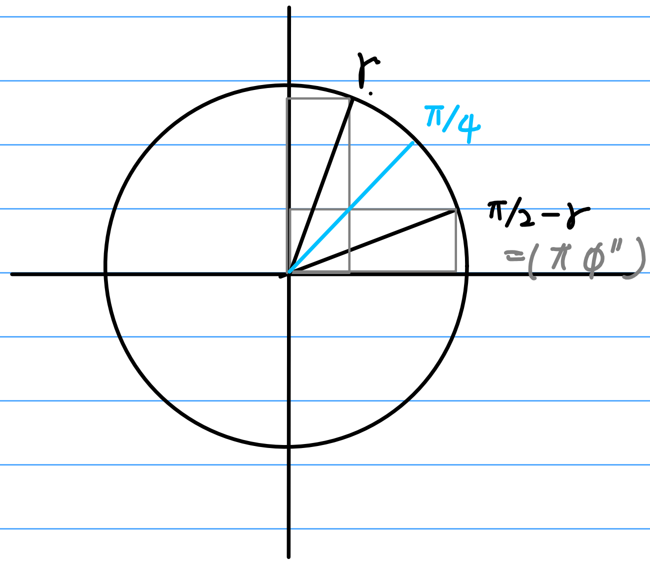
ϕ $[0, 2\pi] \longrightarrow [0, \frac{\pi}{4}]$

conditionally interchanging inputs X_0 & Y_0
Conditionally interchanging and negating outputs X & Y

$$X = X_0 \cos\phi - Y_0 \sin\phi$$
$$Y = Y_0 \cos\phi + X_0 \sin\phi$$

Madisetti VLSI arch





for frequency synthesis

argument: signed normalized by π angle $[-1, 1]$

binary representation of a radian angle required

$[-1, 1] \xrightarrow[\phi]{} [0, \pi/4] \xrightarrow[\theta]{}$ sine/cosine generator $\rightarrow \pi\phi$

① a phase accumulator ϕ $[-1, 1]$
② a radian converter $\phi \rightarrow \theta$
③ a sine/cosine generator $\sin\theta$, $\cos\theta$
④ an output stage $\sin\theta$, $\cos\theta$

$\sin\theta \rightarrow \sin\pi\phi$, $\cos\theta \rightarrow \cos\pi\phi$

.

output stage $\quad \sin\theta \to \sin\pi\phi \quad [-\pi, +\pi]$

$\cos\theta \to \cos\pi\phi$

negation / interchange

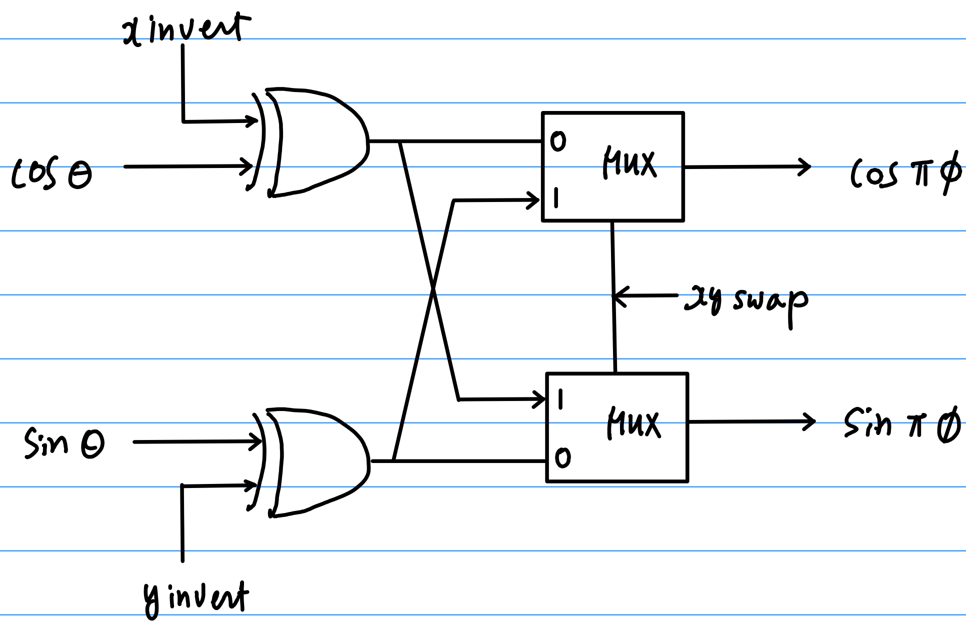
MSB1, MSB2, MSB3 → x invert, y invert, xy swap

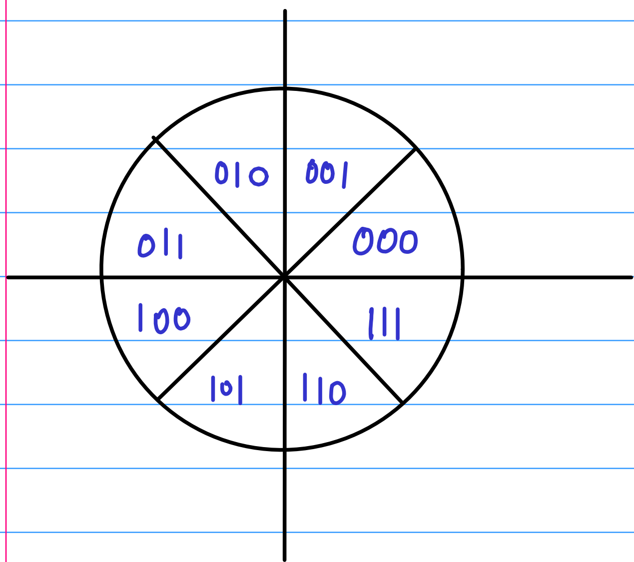
x invert: the negation of $\cos\theta = X_{N+1}$

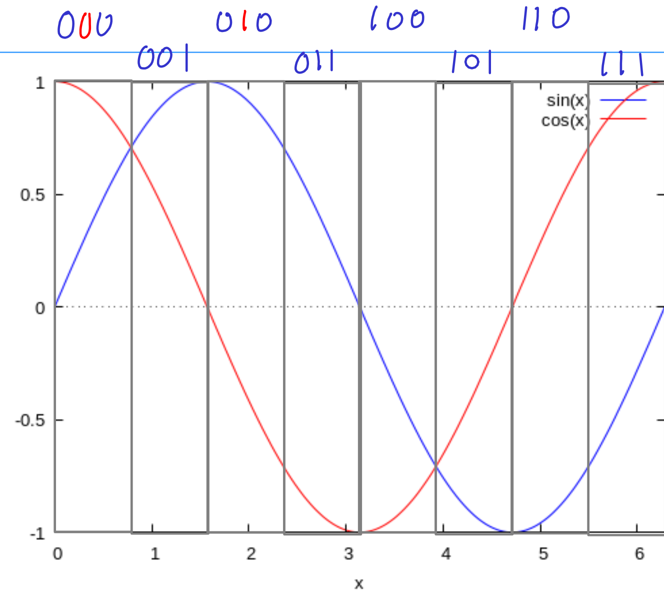
y invert: $\sin\theta = Y_{N+1}$

xy swap: Interchange

negate before swap







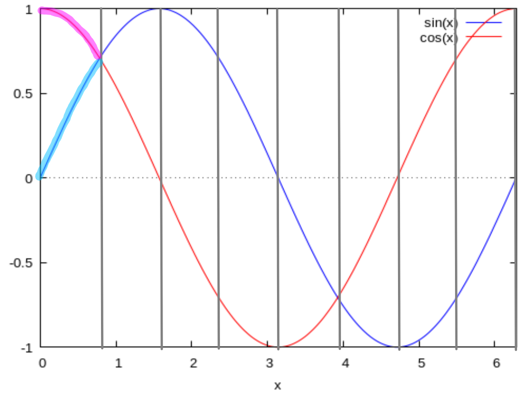
	cos	sin.			
	Xinv	Yinv	swap	$\cos\pi\theta$	$\sin\pi\phi$
000	0	0	0	$\cos\theta$	$\sin\theta$
001	0	0	1	$\sin\theta$	$\cos\theta$
010	0	1	1	$-\sin\theta$	$\cos\theta$
011	1	0	0	$-\cos\theta$	$\sin\theta$
100	1	1	0	$-\cos\theta$	$-\sin\theta$
101	1	1	1	$-\sin\theta$	$-\cos\theta$
110	1	0	1	$\sin\theta$	$-\cos\theta$
111	0	1	0	$\cos\theta$	$-\sin\theta$

Madisetti & Willson, DDS Freq synthesizer

0

$\cos\theta$

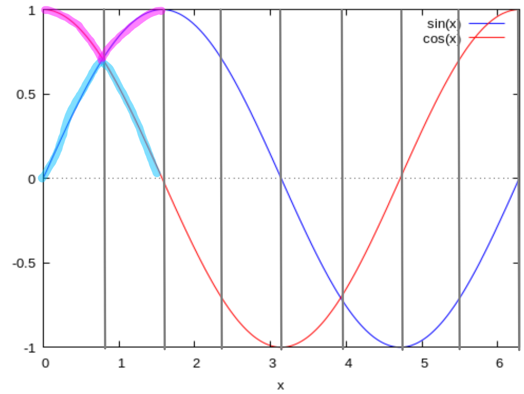
$\sin\theta$



1

$\sin\theta$

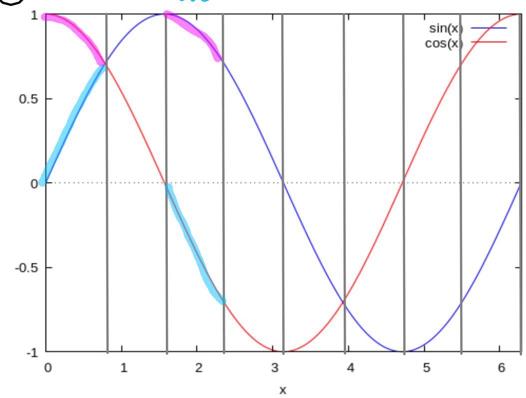
$\cos\theta$



2

$-\sin\theta$

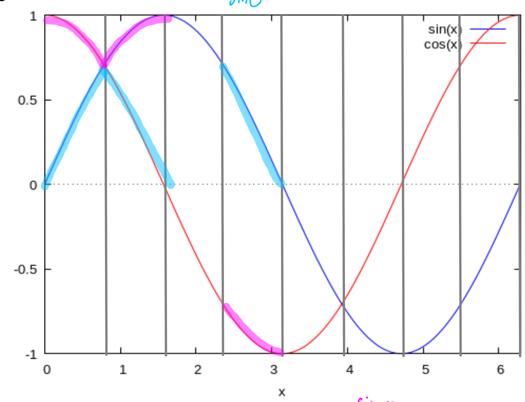
$\cos\theta$



3

$-\cos\theta$

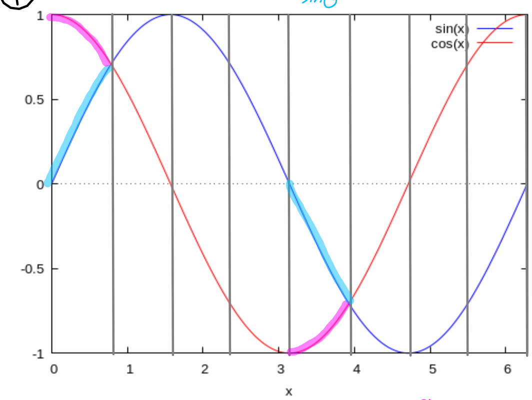
$\sin\theta$



4

$-\cos\theta$

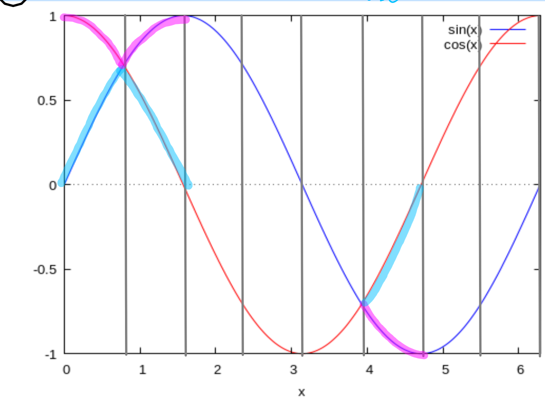
$-\sin\theta$



5

$-\sin\theta$

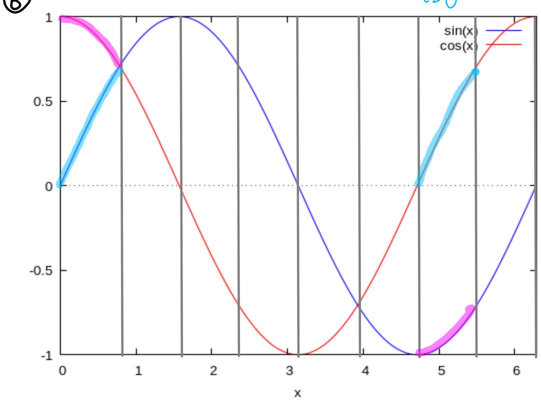
$-\cos\theta$



6

$\sin\theta$

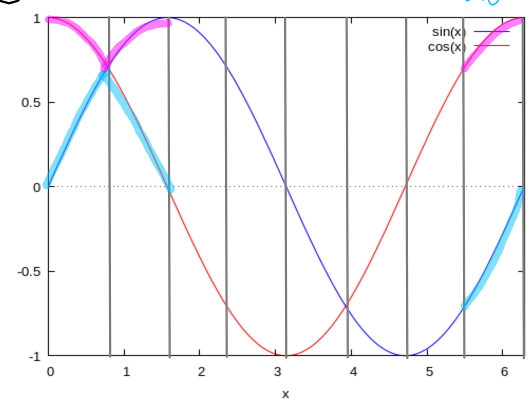
$-\cos\theta$



7

$\cos\theta$

$-\sin\theta$



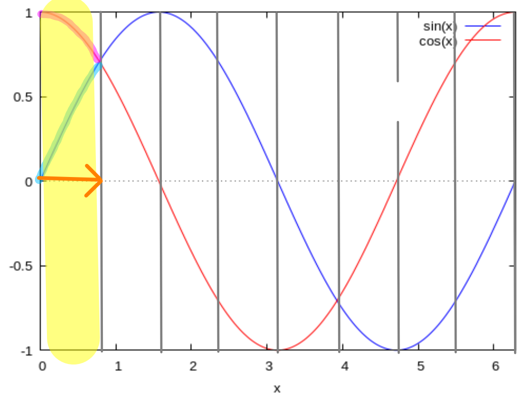
Madisetti & Willson, DDS Freq synthesizer

$$\begin{cases} \cos\phi \\ \sin\phi \end{cases}$$

0

$\cos\theta$

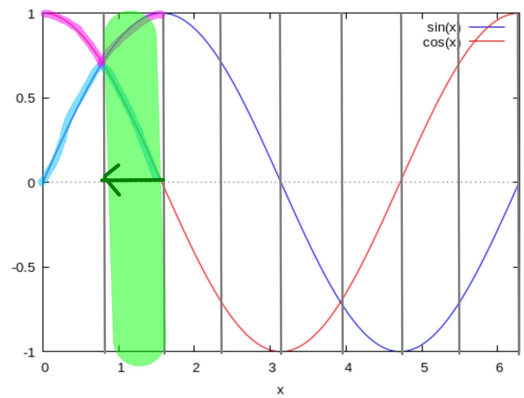
$\sin\theta$



1

$\sin\theta$

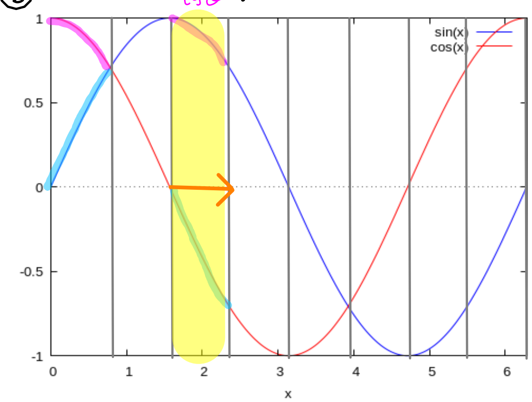
$\cos\theta$



2

$-\sin\theta$

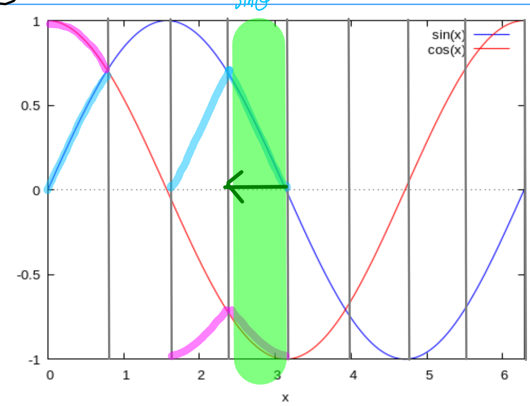
$\cos\theta$



3

$-\cos\theta$

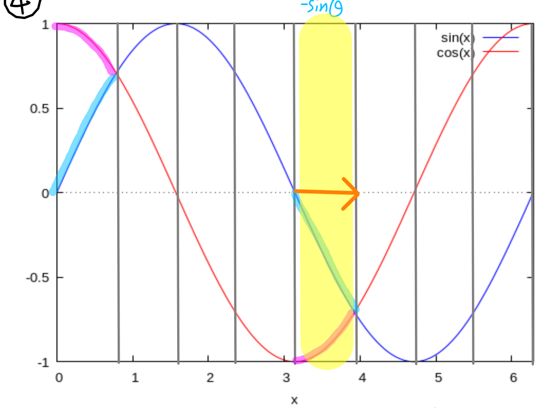
$\sin\theta$



4

$-\cos\theta$

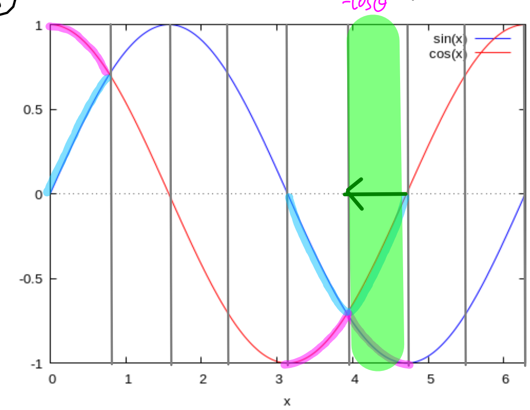
$-\sin\theta$



5

$-\sin\theta$

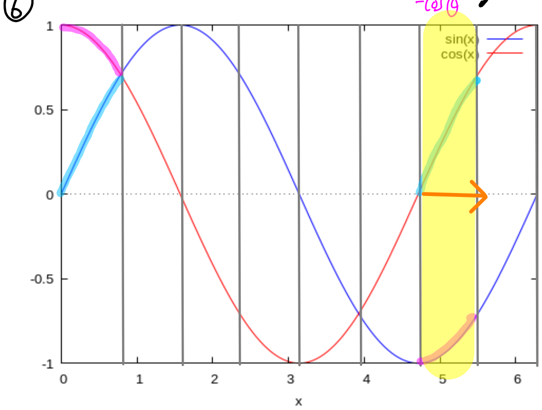
$-\cos\theta$



6

$\sin\theta$

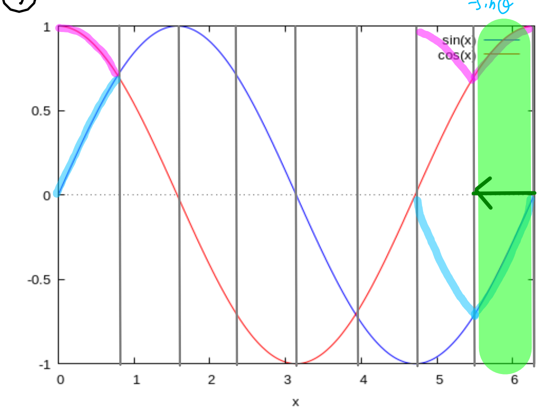
$-\cos\theta$



7

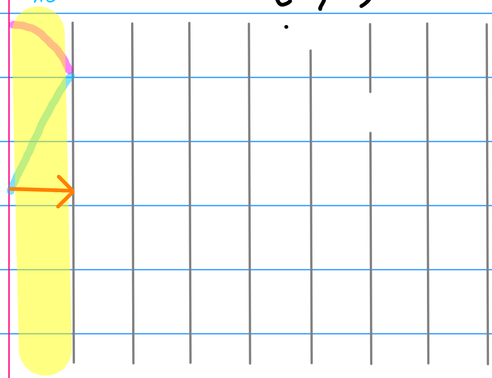
$\cos\theta$

$-\sin\theta$

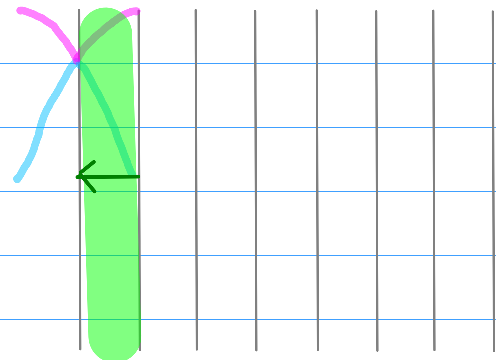


$\sin \phi$

⓪ $\cos\theta$ / $\sin\theta$ + +   (0, 0)

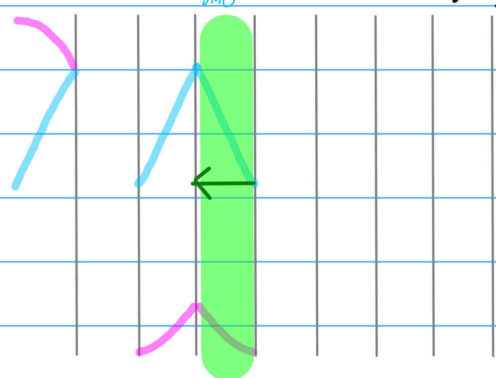


① $\sin\theta$ / $\cos\theta$)   + +   (0, 0)

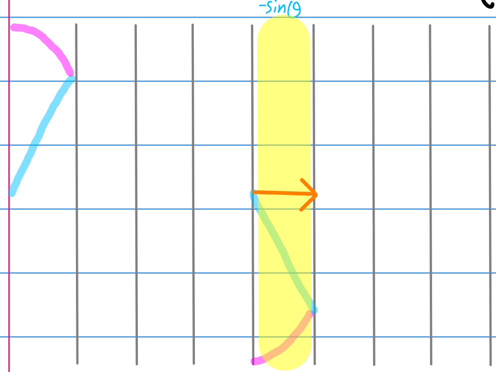


② $-\sin\theta$ / $\cos\theta$)   + −   (0, 1)

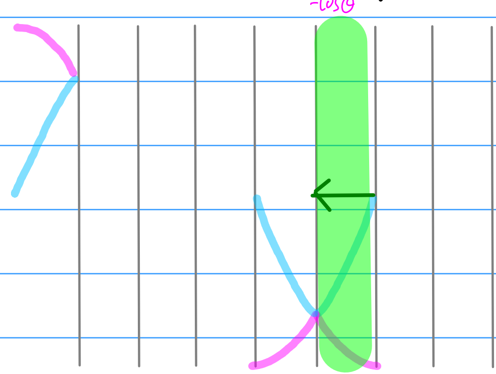
③ $-\cos\theta$ / $\sin\theta$ − +   (1, 0)



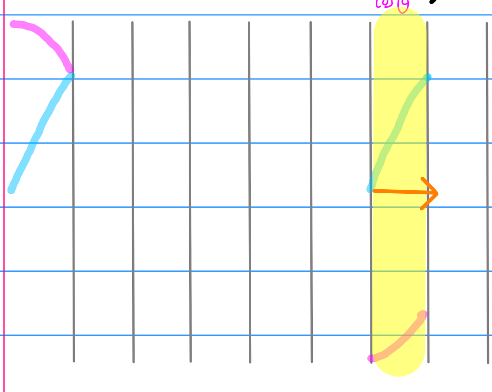
④ $-\cos\theta$ / $-\sin\theta$ − −   (1, 1)



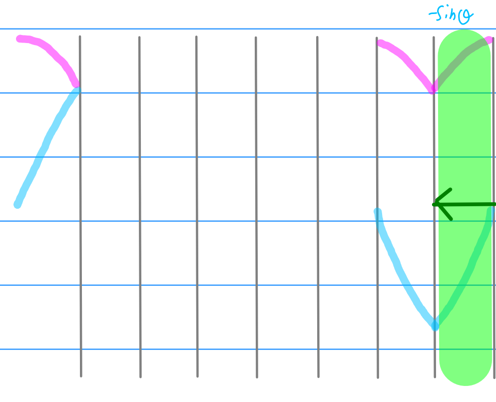
⑤ $-\sin\theta$ / $-\cos\theta$)   − −   (1, 1)



⑥ $\sin\theta$ / $-\cos\theta$)   − +   (1, 0)



⑦ $\cos\theta$ / $-\sin\theta$ + −   (0, 1)



Madisetti & Willson, DDS Freq synthesizer

	xinv	yinv	swap	$\cos\pi\phi$	$\sin\pi\phi$
0 0 0	0	0	0	$\cos\theta$	$\sin\theta$
0 0 1	0	0	1	$\sin\theta$	$\cos\theta$
0 1 0	0	1	1	$-\sin\theta$	$\cos\theta$
0 1 1	1	0	0	$-\cos\theta$	$\sin\theta$
1 0 0	1	1	0	$-\cos\theta$	$-\sin\theta$
1 0 1	1	1	1	$-\sin\theta$	$-\cos\theta$
1 1 0	1	0	1	$\sin\theta$	$-\cos\theta$
1 1 1	0	1	0	$\cos\theta$	$-\sin\theta$

0 / 0	0 / 0
0 / 1	1 / 0
1 / 1	1 / 1
1 / 0	0 / 1

0 0 0 0
0 1 1 0
1 1 1 1
1 0 0 1

Madisetti & Willson, DDS Freq synthesizer

$$\theta = \sum_{k=1}^{N} b_k \theta_k$$

b_k sign + N bit — (N+1) bit fractional b

$b_k \in \{0, 1\}$

$\theta_k = 2^{-k}$

θ is constrained to be positive $\boxed{b_0 = 0}$

$$\theta = \sum_{k=1}^{N} b_k 2^{-k} = \phi_0 + \sum_{k=2}^{N+1} r_k 2^{-k}$$

$r_k \in \{-1, +1\}$ signed digits

ϕ_0 constant

⊕ subrotation by 2^{-k}

2 equal ⊕ half rotations by 2^{-k-1}

⊖ subrotation

2 equal opposite half rotations by $\pm 2^{-k-1}$

Binary Representation

$b_k = 1$: rotation by 2^{-k}

$b_k = 0$: zero rotation

k-th rotation

- fixed rotation by 2^{-k-1}
- pos rotation $\leftarrow b_k = 1$
  neg rotation $\leftarrow b_k = 0$

Combining all the fixed rotations

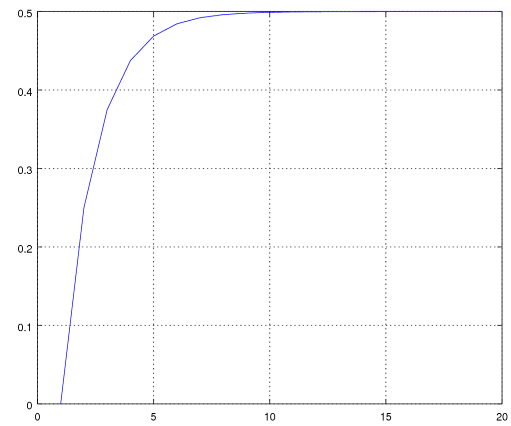
$\rightarrow$ initial fixed rotation

	b_1	b_2	b_3		b_N
	2^{-1}	2^{-2}	2^{-3}		2^{-N}
fixed $\Rightarrow$	$+2^{-2}$	$+2^{-3}$	$+2^{-4}$		$+2^{-N-1}$
	$(b_1=1)$ $+2^{-2}$	$(b_2=1)$ $+2^{-3}$	$(b_3=1)$ $+2^{-4}$		$(b_N=1)$ $+2^{-N-1}$
	$(b_1=0)$ -2^{-2}	$(b_2=0)$ -2^{-3}	$(b_3=0)$ -2^{-4}		$(b_N=0)$ -2^{-N-1}

initial fixed rotation

$$\phi_0 = \frac{1}{2^2} + \frac{1}{2^3} + \cdots + \frac{1}{2^{N+1}}$$

$$= \frac{\frac{1}{2^2}\left(1-\frac{1}{2^N}\right)}{\left(1-\frac{1}{2}\right)} = \frac{1}{2}\left(1-\frac{1}{2^N}\right) = \frac{1}{2} - \frac{1}{2^{N+1}}$$



Signed Digit Recoding

the rotation after recoding

— a fixed initial rotation ϕ_0

a sequence of ⊕/⊖ rotations

$b_k = 1 \quad +\ 2^{-k-1}$ rotation

$b_k = 0 \quad -\ 2^{-k-1}$ rotation

$$r_k = (2b_{k-1} - 1)$$

$2 \cdot 1 - 1 = +1 \qquad b_{k-1} = 1 \longrightarrow r_k = +1$

$2 \cdot 0 - 1 = -1 \qquad b_{k-1} = 0 \longrightarrow r_k = -1$

The recoding need not be explicitly performed

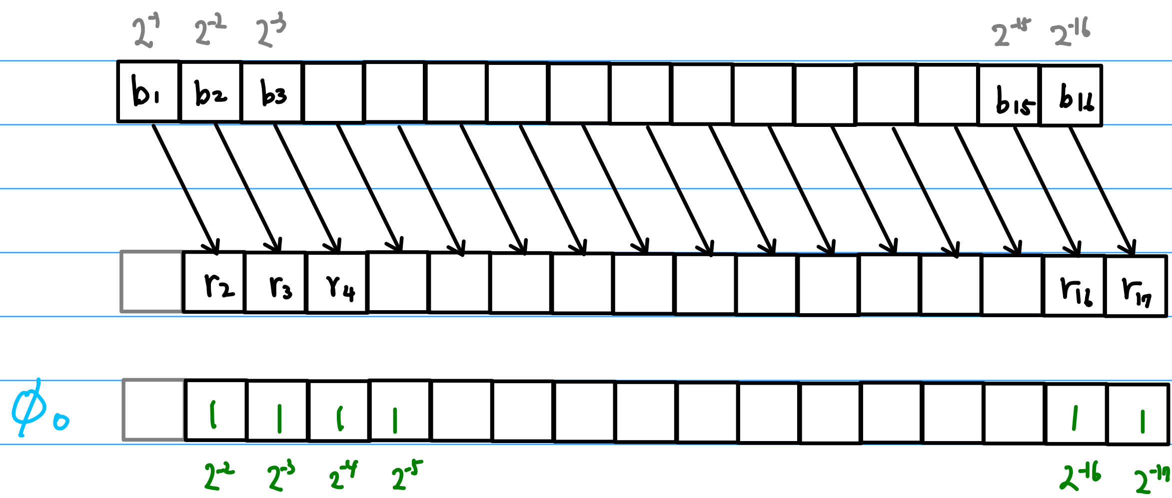
Simply replacing $b_k = 0$ with -1

This recoding maintains

a constant scaling factor K

$$\theta = \sum_{k=1}^{N} b_k 2^{-k} = \phi_0 + \sum_{k=2}^{N+1} r_k 2^{-k}$$

Binary Representation $\{ b_k \}$



Signed Digit Recoding $\{ r_k \}$

The Scaling K.
The initial rotation ϕ_0

rotation Starting point
$$(X_0, Y_0) = (K\cos\phi_0, K\sin\phi)$$

— fixed
— no error buildup

— rotation direction
immediately obtained from the binary representation
→ no need for comparison

the subangles $\boxed{\theta_k = 2^{-k}}$ used in recoding
the subangles $\boxed{\theta_k = \tan^{-1}(2^{-k})}$ used in CORDIC

$\tan\theta_k$ multipliers used
in the first few subrotation stages
Cannot be implemented
as a simple shift-and-add operations

→ ROM implementation

reduced chip area
higher operating speed.

Architecture

① phase accumulator $\phi \in [-1, +1]$

② radian converter $\phi \to \theta \in [0, \frac{\pi}{4}]$

③ sine/cosine generator $\sin(\theta)$ $\cos(\theta)$

④ output stage $\sin(\pi\phi)$ $\cos(\pi\phi)$

Overflowing 2's complement accumulator

normalized by π angle ϕ

need radian angle $\theta \in [0, \frac{\pi}{4}]$

$0 < \theta < 1$ rad

N-bit binary representation of θ

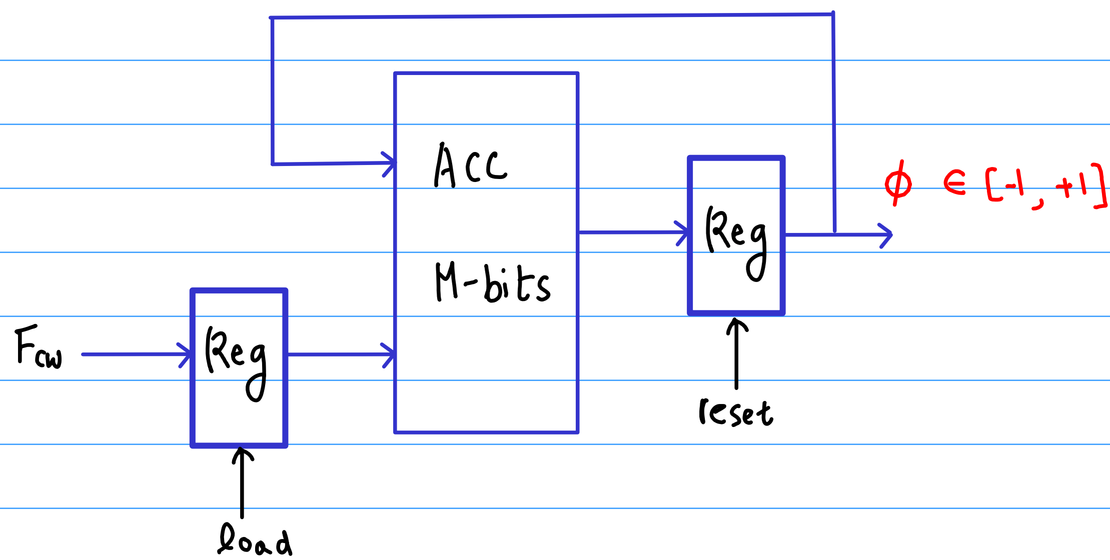
controls the direction of subrotation

N-bit precision of $\cos\theta$ & $\sin\theta$

Output stage

$$\theta \rightarrow \pi\phi$$
$$\sin\theta \rightarrow \sin\pi\phi$$
$$\cos\theta \rightarrow \cos\pi\phi$$

phase accumulator



M-bit adder

repeatedly increments the phase angle by F_{cw} at each clock cycle

frequency control word

at time n, $\phi = n F_{cw} / 2^M$

$$\cos\phi = \cos(n F_{cw} / 2^M)$$
$$\sin\phi = \sin(n F_{cw} / 2^M)$$

Radian Converter

normalized angle ϕ

ϕ

MSB1	MSB2	MSB3					...		

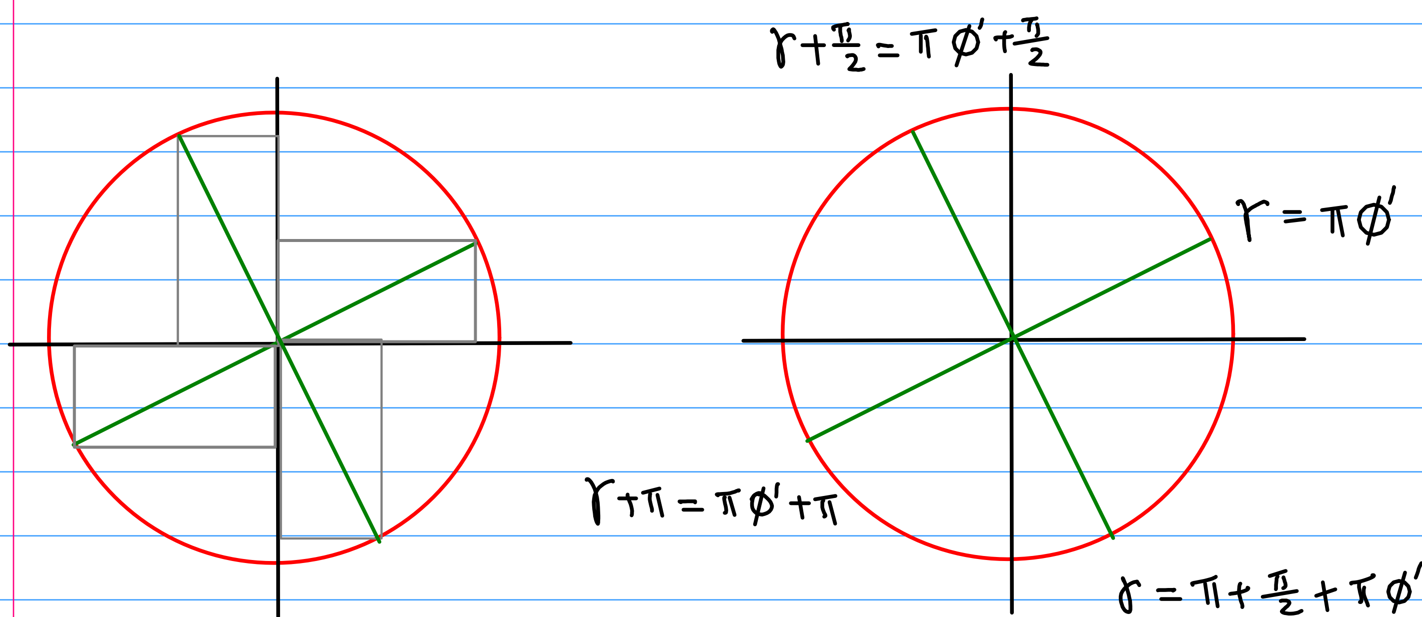
Quadrant of $\pi\phi$

U. half
L. half

Stored to interchange / negate

ϕ'

0	0	MSB3					...		



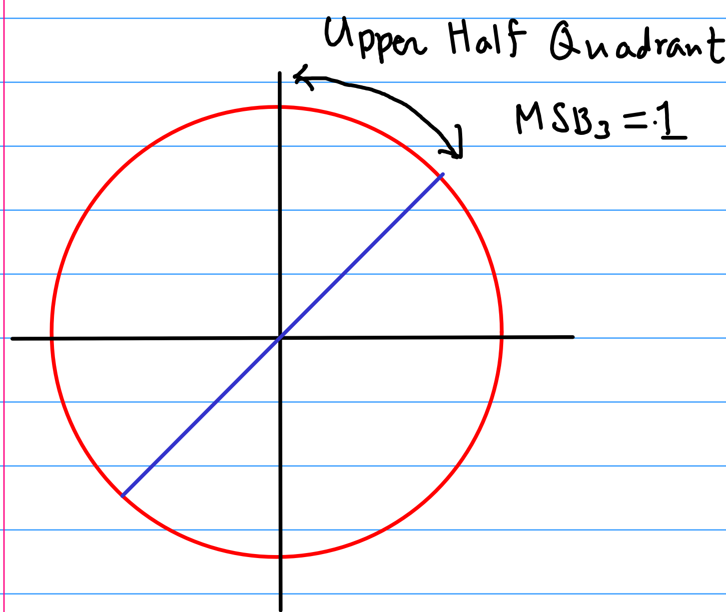
$\phi \rightarrow \phi' \rightarrow$

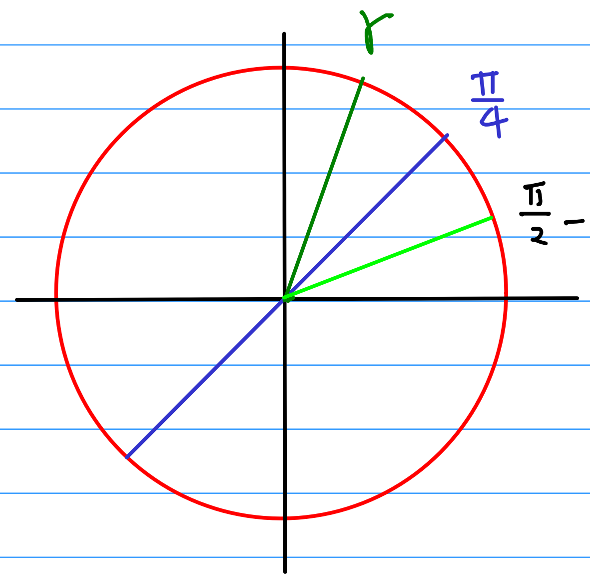
$\pi\phi'$	$+\ 0 \cdot \frac{\pi}{2}$	00
$\pi\phi'$	$+\ 1 \cdot \frac{\pi}{2}$	01
$\pi\phi'$	$+\ 2 \cdot \frac{\pi}{2}$	10
$\pi\phi'$	$+\ 3 \cdot \frac{\pi}{2}$	11

1st Quad

1st Quadrant

ϕ'





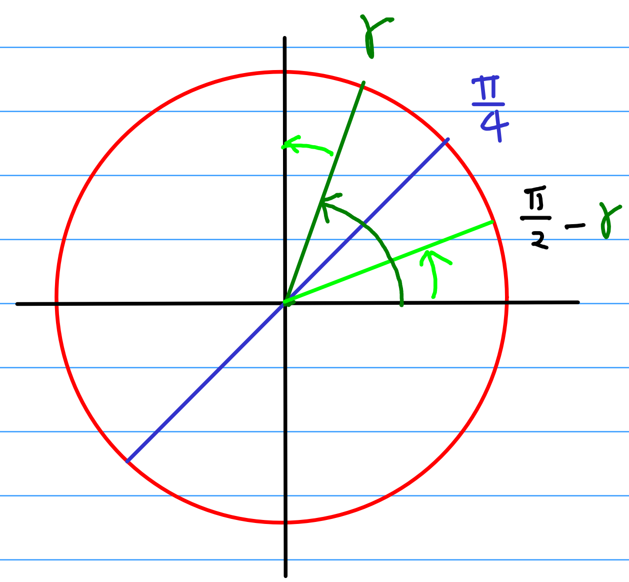
$\gamma > \frac{\pi}{4}$: Upper Half ($MSB_3 = 1$)

$\gamma < \frac{\pi}{4}$: Lower Half ($MSB_3 = 0$)

$$\cos \gamma = \sin\left(\frac{\pi}{2} - \gamma\right)$$

$$\sin \gamma = \cos\left(\frac{\pi}{2} - \gamma\right)$$

$\gamma > \frac{\pi}{4}$ $\quad$ $\frac{\pi}{2} - \gamma < \frac{\pi}{4}$



ϕ'	0	0	MSB3				...	

$MSB3 = 1 \qquad \phi' > \frac{\pi}{4}$

$$\phi'' = \frac{\pi}{2} - \phi'$$

0.5	1	0	0				...	
$\ominus\ \phi'$	0	0	MSB3				...	

$$\begin{cases} MSB3 = 0 & \phi'' = \phi' \\ MSB_3 = 1 & \phi'' = 0.5 - \phi' \end{cases}$$

$\theta = \pi \phi''$ (Hardwired Multiplier)

$0 < \theta < \frac{\pi}{4}$

$$\phi \longrightarrow \phi' \longrightarrow \phi''$$

1st Quad Lower Half

Sine / Cosine Generator

Subrotation

$$X_{k+1} = X_k - (r_k \tan\theta_k) Y_k$$

$$Y_{k+1} = Y_k + (r_k \tan\theta_k) X_k$$

$$\begin{bmatrix} X_\theta \\ Y_\theta \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \end{bmatrix}$$

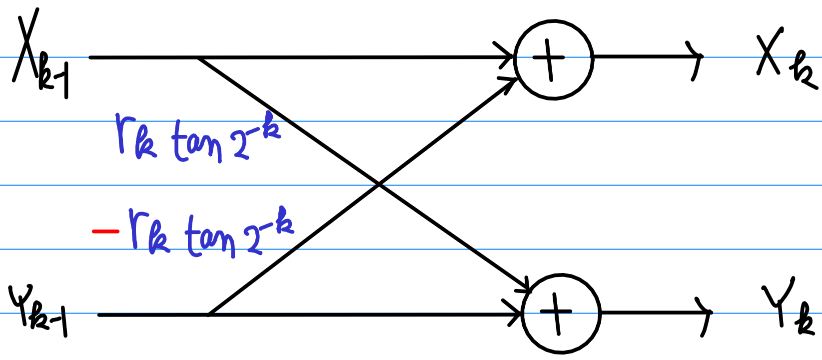
$$= \cos\theta \begin{bmatrix} 1 & -\tan\theta \\ \tan\theta & 1 \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \end{bmatrix}$$

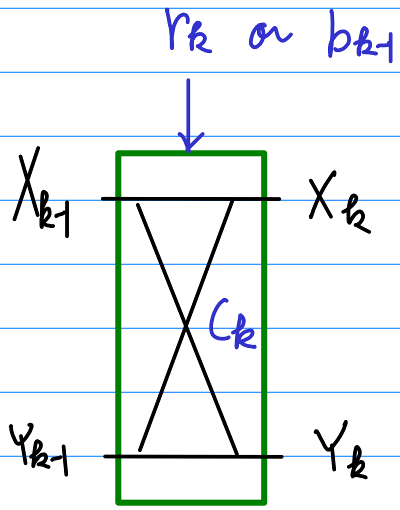
$$\theta = \sigma_0\theta_0 + \sigma_1\theta_1 + \cdots + \sigma_N\theta_N$$

$$\sigma_k = \{-1, 0, +1\}$$

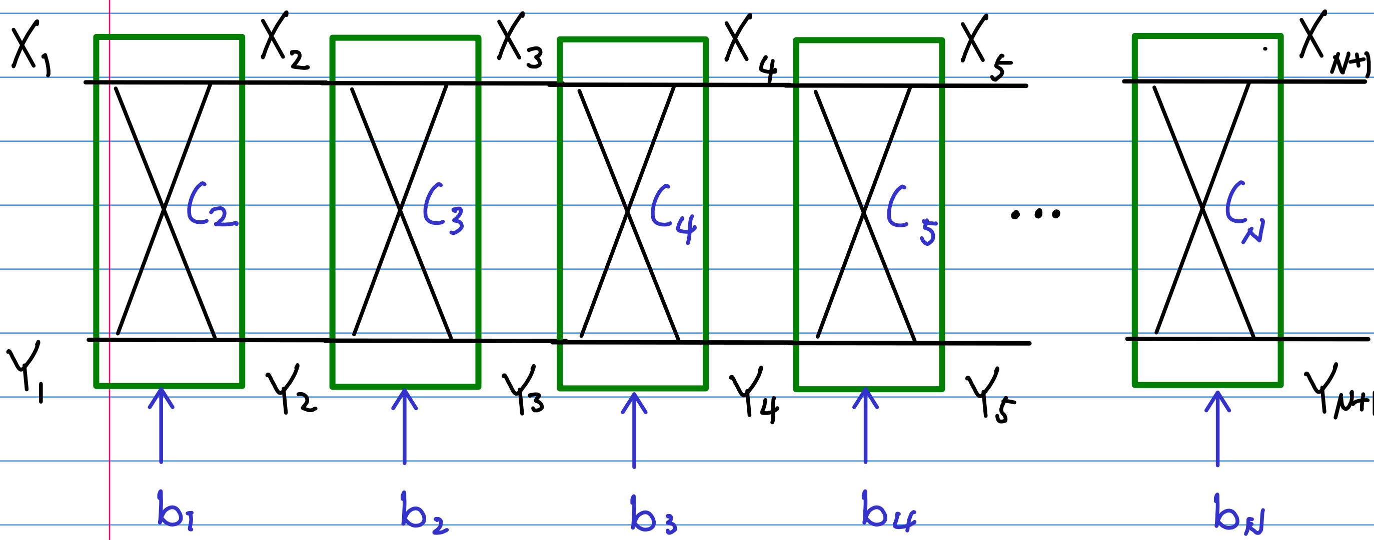
$$\begin{bmatrix} X_\theta \\ Y_\theta \end{bmatrix} = K \begin{bmatrix} 1 & -\tan\sigma_N\theta_N \\ \tan\sigma_N\theta_N & 1 \end{bmatrix} \cdots \begin{bmatrix} 1 & -\tan\sigma_0\theta_0 \\ \tan\sigma_0\theta_0 & 1 \end{bmatrix} \begin{bmatrix} X_0 \\ Y_0 \end{bmatrix}$$

$$K = \cos\sigma_0\theta_0 \cdot \cos\sigma_1\theta_1 \cdots \cos\sigma_N\theta_N$$





Butterfly



$C_2 = \tan\left(\frac{1}{2^2}\right)$ $C_3 = \tan\left(\frac{1}{2^3}\right)$ $C_4 = \tan\left(\frac{1}{2^4}\right)$ $C_5 = \tan\left(\frac{1}{2^5}\right)$

$K\cos\phi_0 \to X_1$ $X_{N+1} \to \cos\theta$

$K\sin\phi_0 \to Y_1$ $Y_{N+1} \to \sin\theta$

$\theta \to \{b_1, b_2, \dots, b_N\}$

the initial (X_0, Y_0) always the same

merge the first $B/3$ butterflies

→ $2^{B/3}$ words ROM implementation

→ no need $\tan\theta_k$ multipliers

→ $\{b_1, b_2, \cdots, b_{B/3}\} \Rightarrow$ address

accesses

$$\cos\left(\phi_0 + \sum_{k=1}^{B/3} b_k 2^{-k+1}\right)$$

$$\sin\left(\phi_0 + \sum_{k=1}^{B/3} b_k 2^{-k+1}\right)$$

Lower Half of the 1st Quadrant

- all positive X_k & Y_k
- no need sign extension
- reduce the loads
- high speed

Merging Butterflies

merge m final butterflies

$$\begin{pmatrix} X_k \\ Y_k \end{pmatrix} \longrightarrow \begin{pmatrix} X_{k+m} \\ Y_{k+m} \end{pmatrix} \quad \text{directly}$$

$$X_{k+m} = X_k - Y_k \sum_{i=k}^{k+m-1} r_i \tan 2^{-i}$$

$$Y_{k+m} = Y_k + X_k \sum_{i=k}^{k+m-1} r_i \tan 2^{-i}$$

valid merging $\quad k \geq (B-1)/2$

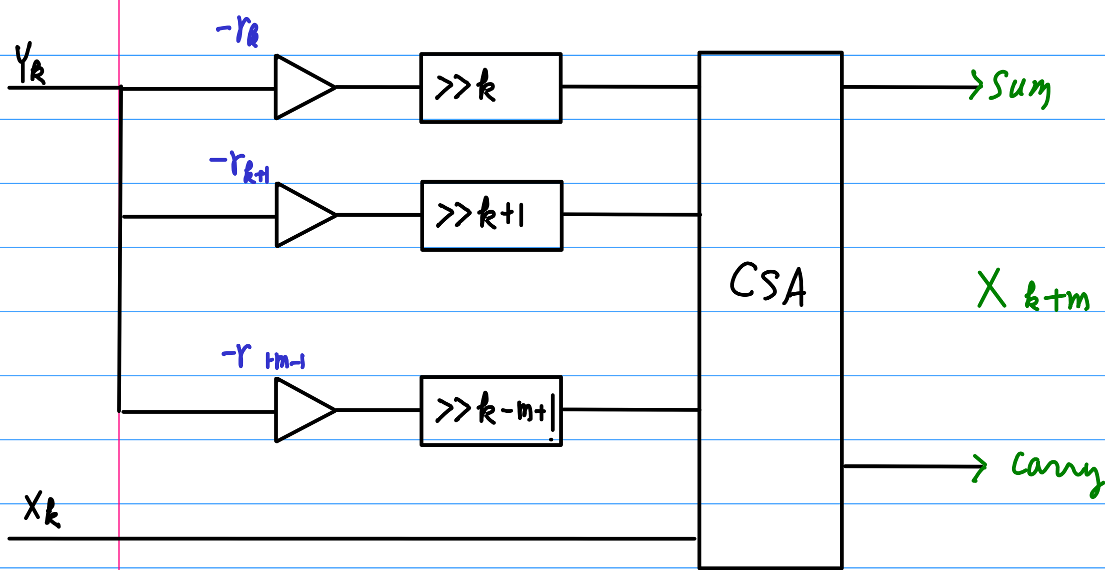
$\tan(2^{-i}) = 2^{-i} \quad k \geq B/3$

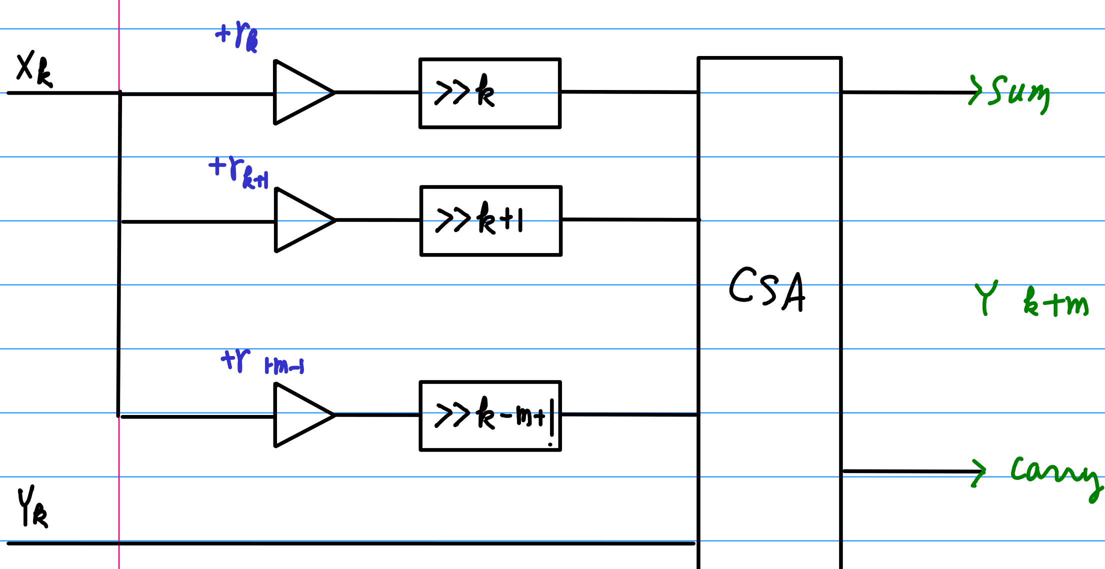
lookahead by m

the individual terms in the summation can be computed independently and summed in parallel

$$X_{k+m} = X_k - Y_k \sum_{i=k}^{k+m-1} r_i \tan 2^{-i}$$

$$Y_{k+m} = Y_k + X_k \sum_{i=k}^{k+m-1} r_i \tan 2^{-i}$$





+ reduced latency
+ reduced routing
+ only the half for a single-ended system.

Output Stage

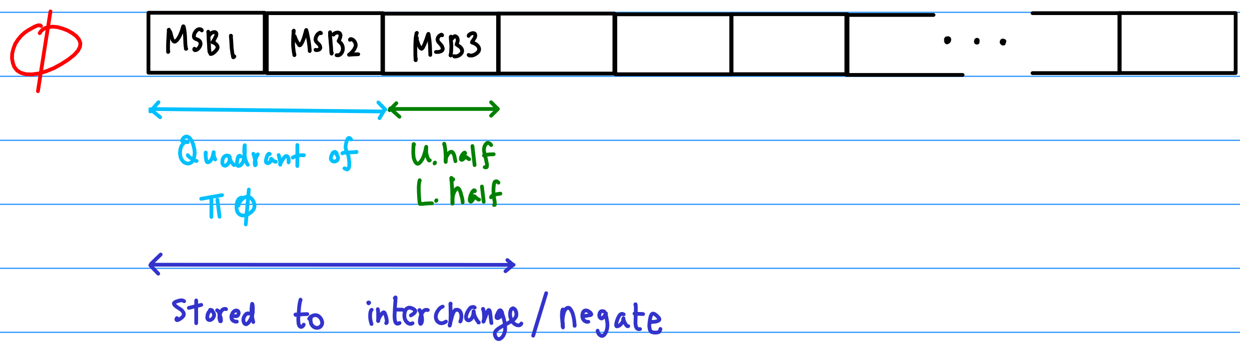
$$\boxed{\begin{array}{l}\sin\ \theta \\ \cos\ \theta\end{array}} \longrightarrow \boxed{\begin{array}{l}\sin \pi\phi \\ \cos \pi\phi\end{array}}$$

$$\theta \in [0, \tfrac{\pi}{4}] \longrightarrow \phi \in [-1, +1]$$

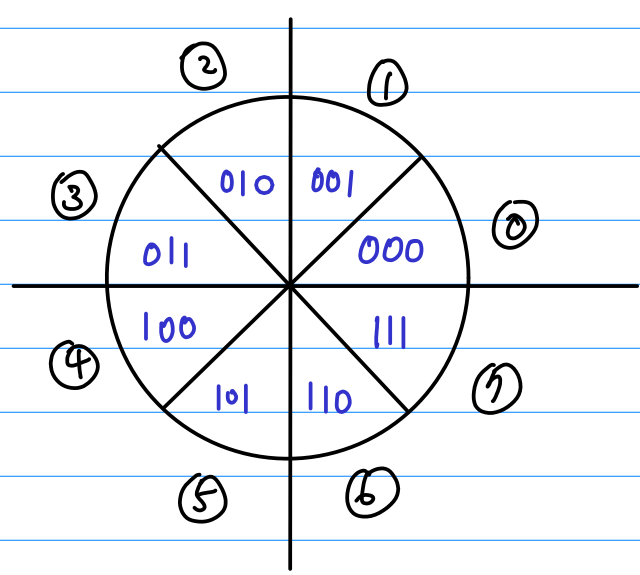
$$\left\{\begin{array}{l}\text{negation} \\ \text{interchange}\end{array}\right.$$

* n-gation before interchange

normalized angle ϕ



MSB of ϕ	ϕ	X_{inv}	Y_{inv}	Swap	$\cos \pi\phi$	$\sin \pi\phi$
0 0 0	0	0	0	0	$\cos\theta$	$\sin\theta$
0 0 1	1	0	0	1	$\sin\theta$	$\cos\theta$
0 1 0	2	0	1	1	$-\sin\theta$	$\cos\theta$
0 1 1	3	1	0	0	$-\cos\theta$	$\sin\theta$
1 0 0	4	1	1	0	$-\cos\theta$	$-\sin\theta$
1 0 1	5	1	1	1	$-\sin\theta$	$-\cos\theta$
1 1 0	6	1	0	1	$\sin\theta$	$-\cos\theta$
1 1 1	7	0	1	0	$\cos\theta$	$-\sin\theta$



IC Implementation

clock : 100 MHz
acc : 36-bit (22-bit + 14-bit)
precision : 16-bit
advantage over traditional ROM lookup table approach

accumulator :: 36-bit = 22-bit + 14 bit
carry select adder

Speed & layout consideration

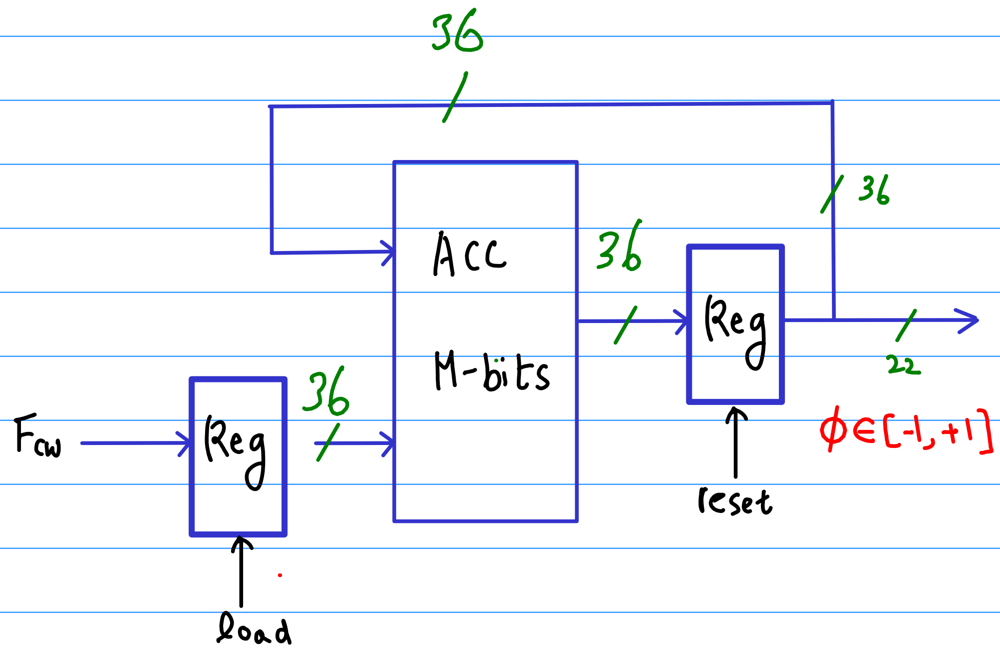
36-bit output → truncated to 22-bit

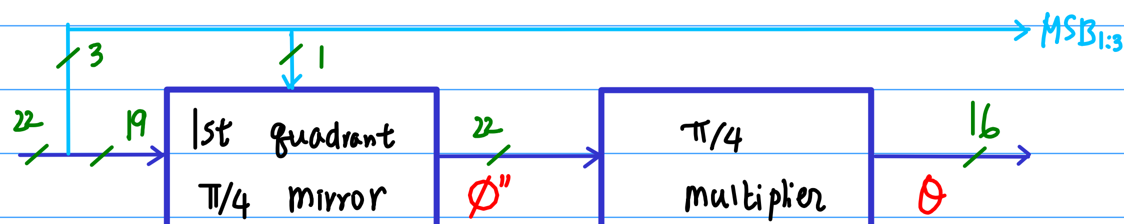
22-bit radian converter
$\pi/4$ multiplier
SDFR ≥ 100 dBc

19-bit ROM address
for the similar performance
⇒ 2^{19} words HUGE!

Coarse / fine ROM arch

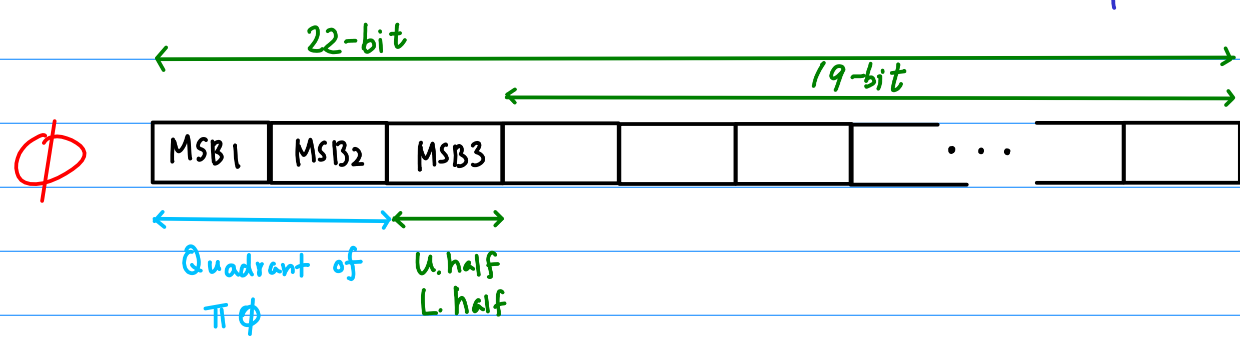
[2] H. T. Nicholas and H. Samueli, "A 150 MHz direct digital frequency synthesizer in 1.25-μm CMOS with −90 dBc spurious performance," *IEEE J. Solid-State Circuits,* vol. 26, pp. 1959–1969, Dec. 1991.

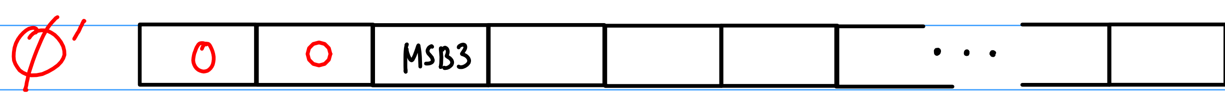




$\theta = \pi \phi''$

$0 < \theta < \frac{\pi}{4}$





$\gamma > \frac{\pi}{4}$: Upper Half $(MSB_3 = 1)$ $\phi'' = \phi'$

$\gamma < \frac{\pi}{4}$: Lower Half $(MSB_3 = 0)$ $\phi'' = 0.5 - \phi'$

radian converter $\theta = \pi \phi''$

all internal angle
~ represented as
fractional binary 2's complement numbers

$\theta = (\pi/4)(4\phi'')$

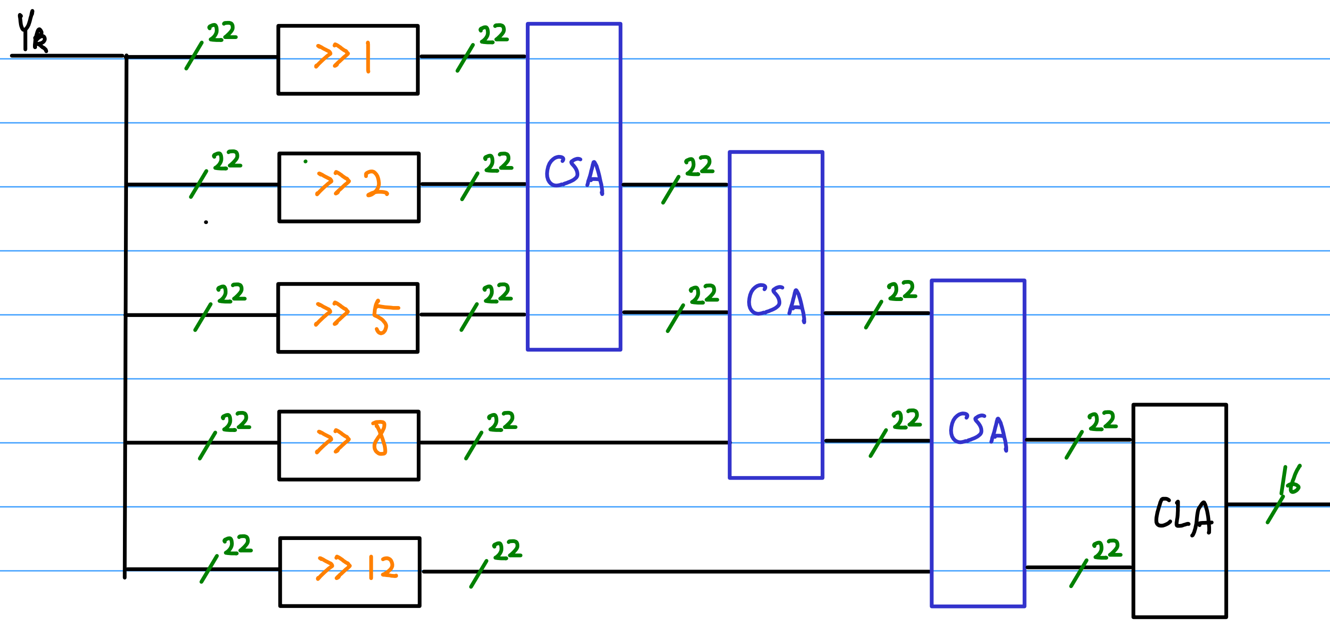
$(\pi/4) = 2^{-1} + 2^{-2} + 2^{-5} + 2^{-8} + 2^{-12}$

1	2	3	4	5	6	7	8	9	10	11	12
1	1	0	0	1	0	0	1	0	0	0	1

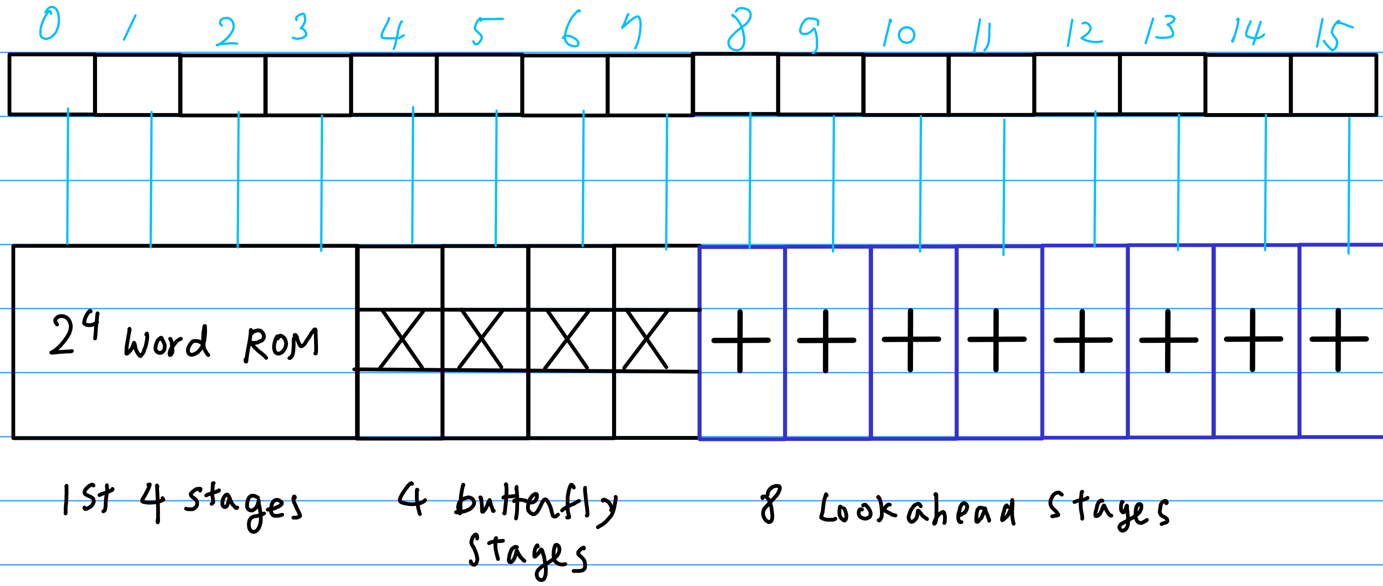
$2^{-1} = 0.5$
$2^{-2} = 0.25$
$2^{-5} = 0.3125$
$2^{-8} = 0.00390625$
$2^{-12} = 0.0002441\!\!41$

$\pi/4 = \boxed{0.785}398163 = \boxed{0.7854}00391$

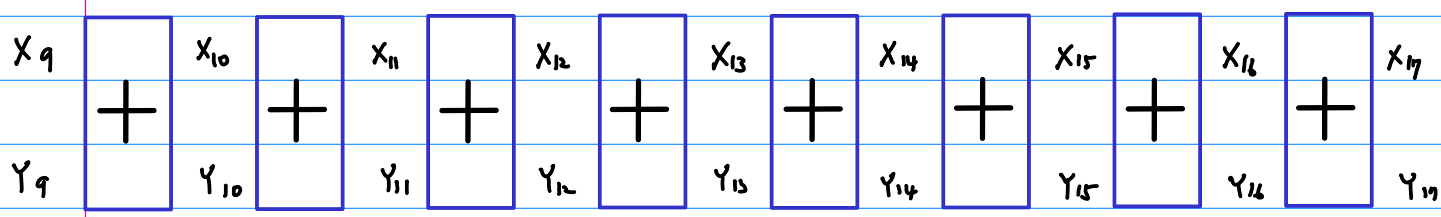
only 1st 5 partial products

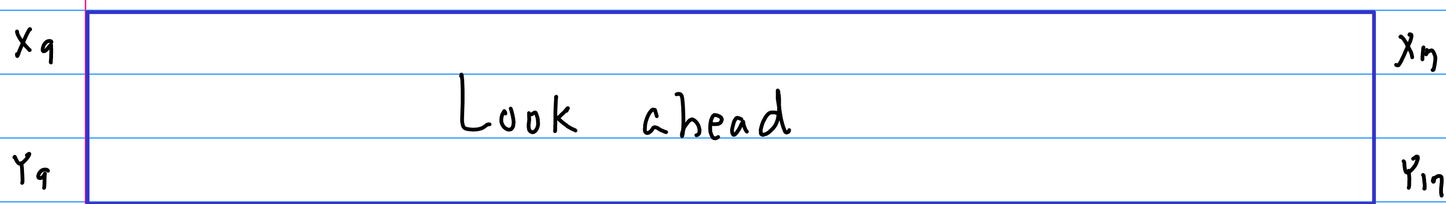


Y_R
22
>> 1
>> 2
>> 5
>> 8
>> 12
CSA
CSA
CSA
CLA
16



0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
2^4 Word ROM
1st 4 stages
4 butterfly stages
8 Lookahead stages


X9
Y9
X10
Y10
X11
Y11
X12
Y12
X13
Y13
X14
Y14
X15
Y15
X16
Y16
X17
Y17

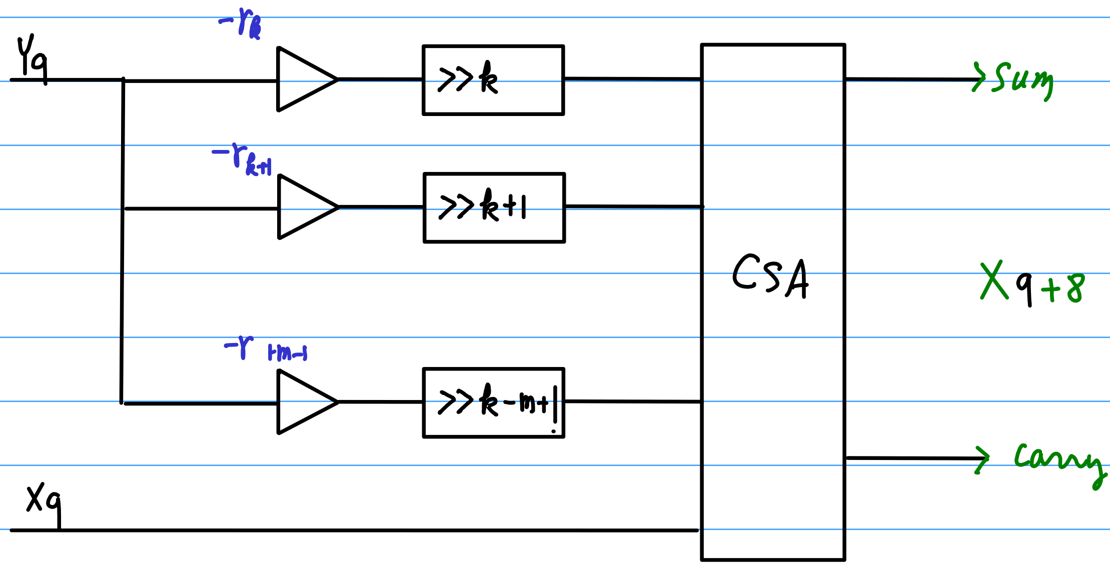

X9
Y9
Look ahead
X17
Y17

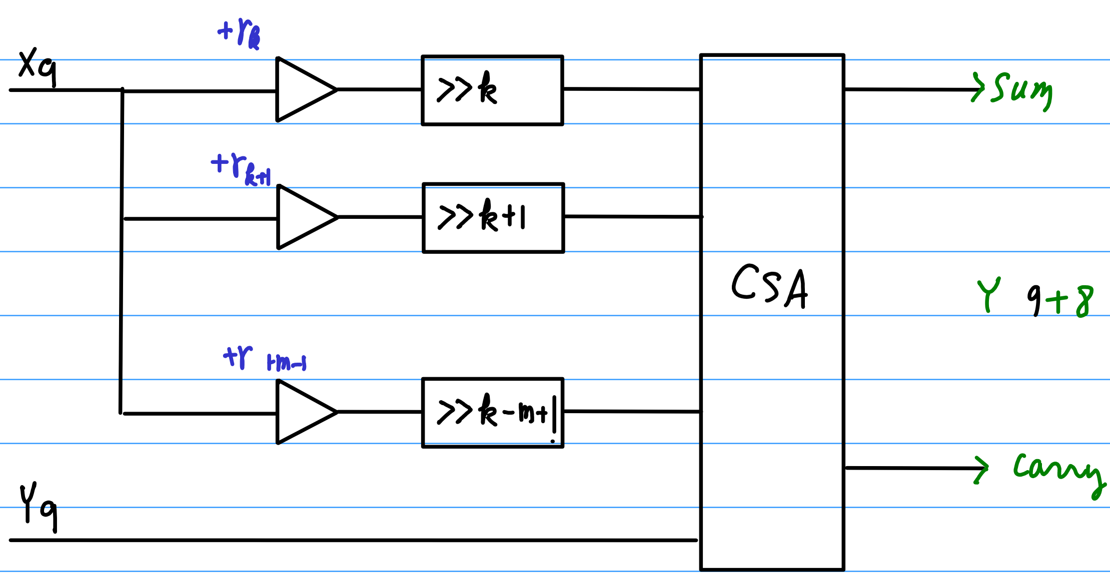
$9 + 8 = 17$

Look ahead

$$X_{k+m} = X_k - Y_k \sum_{i=k}^{k+m-1} r_i \tan 2^{-i}$$

$$Y_{k+m} = Y_k + X_k \sum_{i=k}^{k+m-1} r_i \tan 2^{-i}$$





k	θ_k 22-bit binary $\tan\theta_k$ 22-bit binary																						θ_k decimal $\tan\theta_k$ decimal
		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
1			1																					0.500000
			1			0	1	0	1	1	1	1	0	1	1	0	1	0	0	1	1	1	1	0.546302
2				1																				0.250000
				1						1	0	1	0	1	1	1	1	0	0	0	0	1	0	0.255342
3					1																			0.125000
					1					0	0	0	1	0	1	0	1	0	1	1	1	0	1	0.125655
4						1																		0.062500
						1								0	0	1	0	1	0	1	0	1	0	0.062582
5							1																	0.031250
							1											0	1	0	1	0	1	0.031260
6								1																0.015625
								1														1	0	0.015626
7									1															0.007812
									1															0.007813
8										1														0.003906
										1														0.003906
9											1.													0.001953
											1													0.001953

```
>> t'
ans =

    0
    1
    2
    3
    4
    5
    6
    7
    8
    9
   10
   11
   12
   13
   14
   15

>> t'/16
ans =

   0.00000    0
   0.06250    1
   0.12500    2
   0.18750    3
   0.25000    4
   0.31250    5
   0.37500    6
   0.43750    7
   0.50000    8
   0.56250    9
   0.62500   10
   0.68750   11
   0.75000   12
   0.81250   13
   0.87500   14
   0.93750   15
```

no need to be saved

```
>> pi/4
ans =  0.78540
>>
```

Contents of 2^4-words ROM

cos X-value

0000	0111 1111 1110 1010 1010 10
0001	0111 1111 0111 1010 1100 10
0010	0111 1110 0110 1011 1000 10
0011	0111 1100 1110 1101 1110 10
0100	0111 1010 1111 0011 0110 01
0101	0111 1000 0111 1101 1111 01
0110	0111 0101 1001 0000 0001 01
0111	0111 0010 0010 1100 1010 11
1000	0110 1110 0101 0111 0010 00
1001	0110 1010 0001 0011 0100 10
1010	0110 0101 0110 0101 0110 01
1011	0110 0000 0101 0010 0010 01
1100	0101 1010 1101 1110 1001 10

binary point

sin Y-value

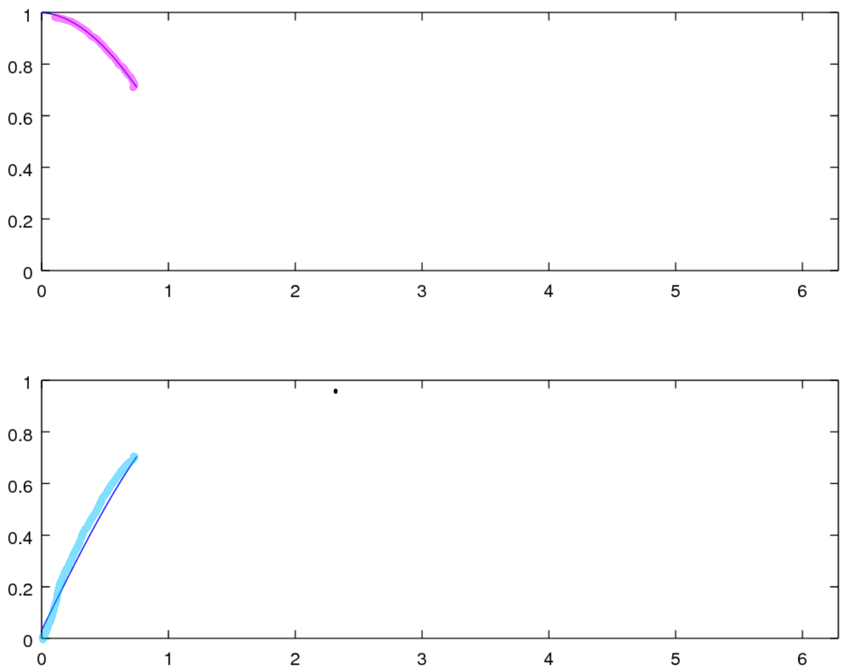
0000	0000 0011 1111 1111 1010 10
0001	0000 1011 1111 1011 0000 00
0010	0001 0011 1110 1010 0101 11
0011	0001 1011 1100 0101 1101 00
0100	0010 0011 1000 0101 0111 11
0101	0010 1011 0010 0001 1010 11
0110	0011 0010 1001 0010 1011 11
0111	0011 1001 1101 0001 0011 11
1000	0100 0000 1101 0101 1111 00
1001	0100 0111 1001 1001 1101 01
1010	0100 1110 0001 0110 0010 01
1011	0101 0100 0100 0100 0110 01
1100	0101 1010 0001 1110 0110 01

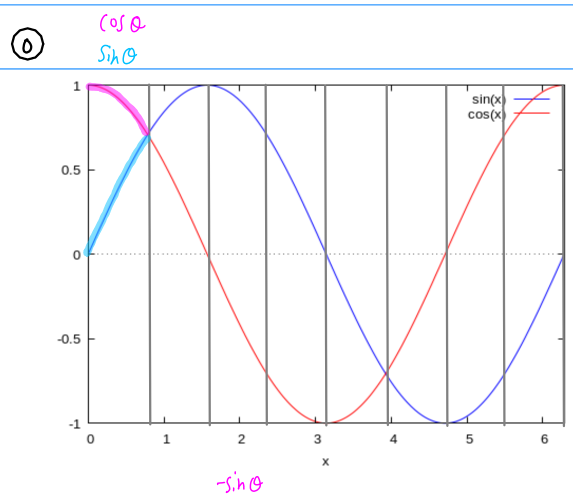
binary point

4-bit address

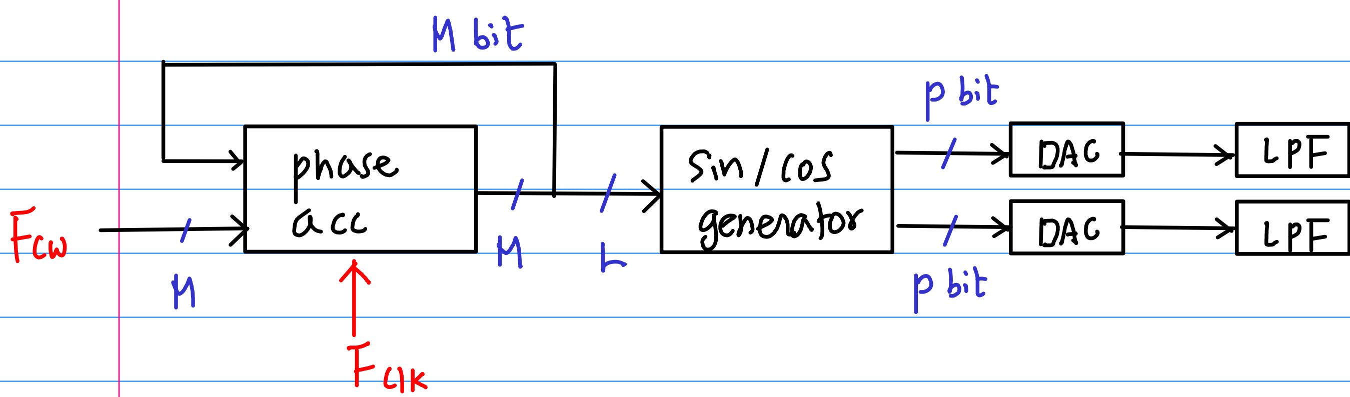
```
C = [ "0111111111101010101010" ;
      "0111111101111010110010" ;
      "0111111001101011100010" ;
      "0111110011101101111010" ;
      "0111101011110011011001" ;
      "0111100001111101111101" ;
      "0111010110010000000101" ;
      "0111001000101100101011" ;
      "0110111001010111001000" ;
      "0110101000010011010010" ;
      "0110010101100101011001" ;
      "0110000001010010001001" ;
      "0101101011011110100110" ]


S = [ "0000001111111111101010" ;
     "0000101111111011000000" ;
     "0001001111101010010111" ;
     "0001101111000101110100" ;
     "0010001110000101011111" ;
     "0010101100100001101011" ;
     "0011001010010010101111" ;
     "0011100111010001001111" ;
     "0100000011010101111100" ;
     "0100011110011001110101" ;
     "0100111000010110001001" ;
     "0101010001000100011001" ;
     "0101101000011110011001" ]


CV = zeros(rows(C), 1);
Cn = 2^rows(C') / 2;
for i=1:rows(C)
  CV(i) = bin2dec(C(i, :)) / Cn;
end

SV = zeros(rows(S), 1);
Sn = 2^rows(S') / 2;
for i=1:rows(S)
  SV(i) = bin2dec(S(i, :)) / Sn;
end
```

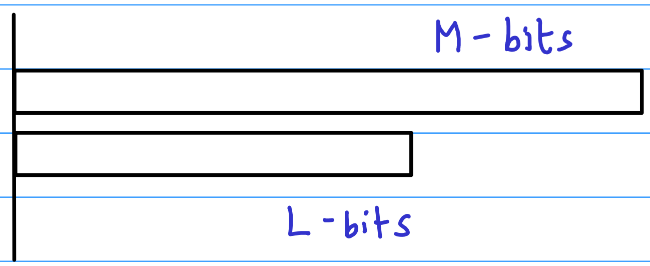
```
>> subplot(2,1,1)
>> axis([0, 2*pi 0 1])
>> plot(x, CV)
>> axis([0, 2*pi 0 1])
>> subplot(2,1,2)
>> plot(x, SV)
>> axis([0, 2*pi 0 1])
>>
```





DDFS (Direct Digital Frequency Synthesis)



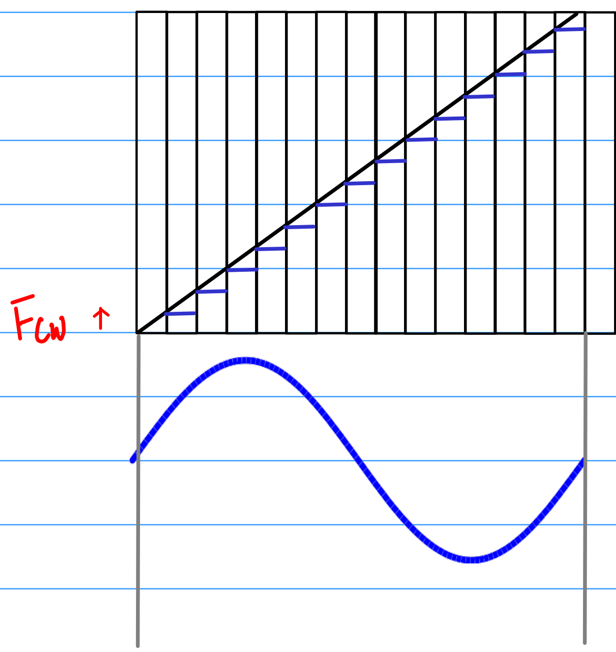


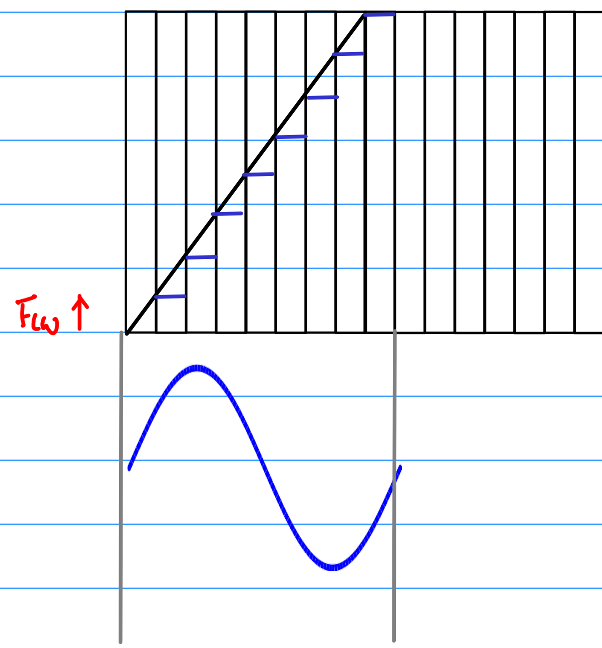
Fcw   Frequency Control Word

at each Fclk, the phase accumulator increments by Fcw until it overflows and wraps

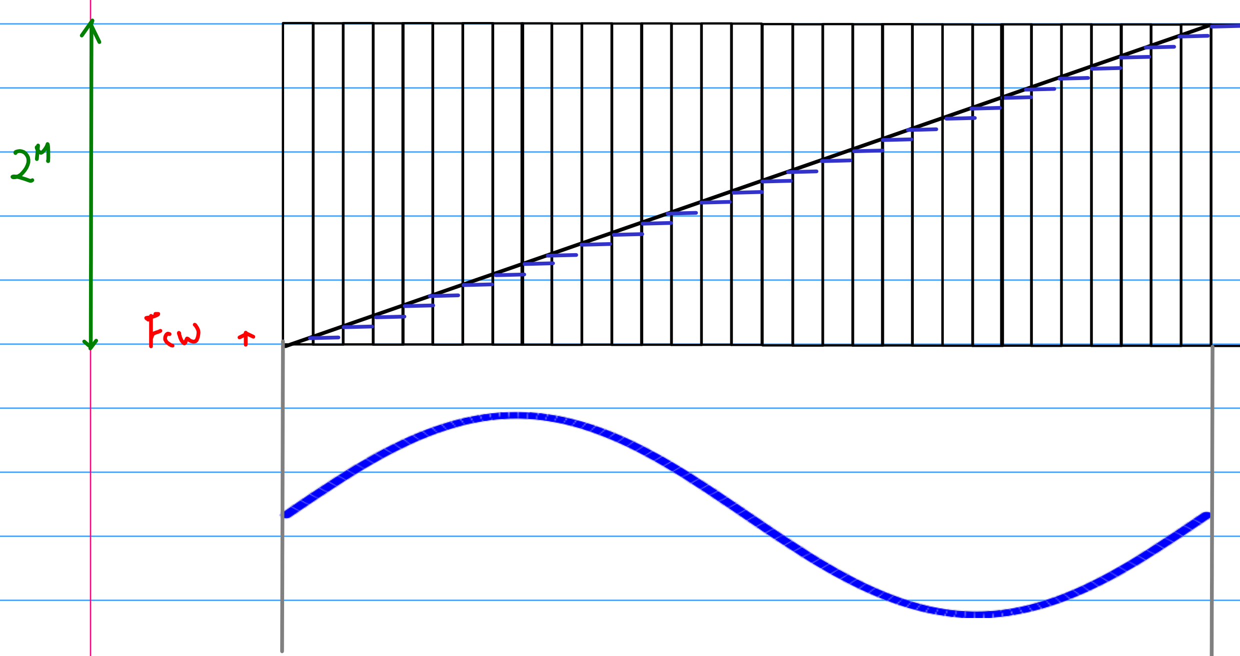
↳ one period of a sine wave

Fcw controls the rate at which the accumulator overflows controls the frequency of the sine or cosine wave.





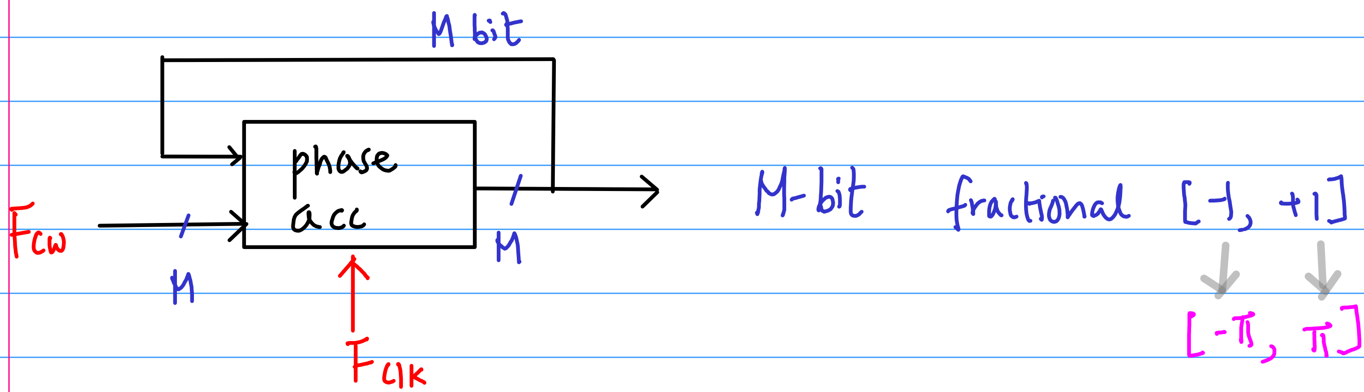
$\leftrightarrow \quad T_{clk} = 1/F_{clk}$



$T_o = (2^M / F_{cw})\, T_{clk}$ $\qquad F_o = \frac{F_{cw}}{2^M} F_{clk}$

Output frequency

/



$$F_0 = \frac{d\theta}{dt} = \frac{F_{clk} F_{cw}}{2^M}$$

Nyquist theorem : at least 2 samples / clock cycle

$$\text{Max } F_{cw} = 2^M/2 = 2^{M-1}$$

$$\text{Max } F_0 = \frac{F_{clk} F_{cw}}{2^M} = \frac{F_{clk}\, 2^{M-1}}{2^M} = \frac{F_{clk}}{2}$$

in practice $F_0 \leq \dfrac{F_{clk}}{3}$ ← $\boxed{\text{DAC}}$

The spectrum of the DAC's output signal

images at $n F_{clk} \pm F_o$

the closest image freq $\Rightarrow$ $F_{clk} - 2F_o$