Hybrid CORDIC 2.A Sine/Cosine Generator

20170701

Wilson ROM based Sine/Cosine Generation

[24] Fu & Willson Sine / Cosine Generation

ROM-based

For high resolution, ROM size grows exponentially

quater-wave symmetry

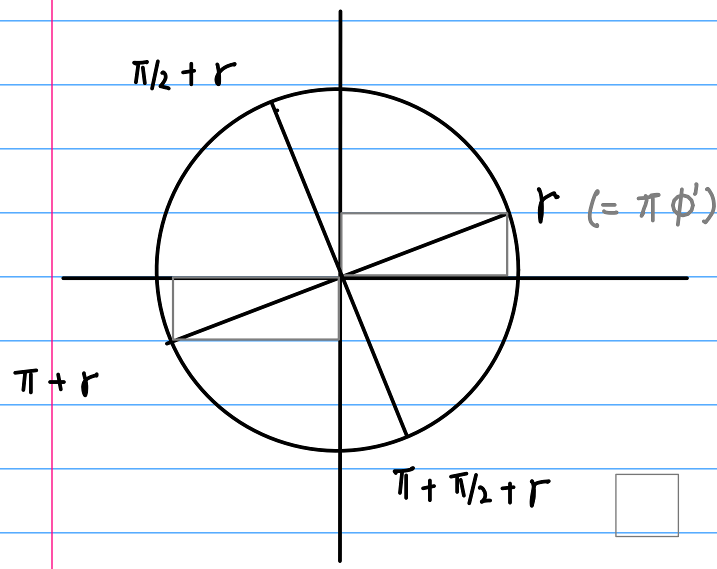
$$\sin\theta = \cos\left(\frac{\pi}{2} - \theta\right)$$

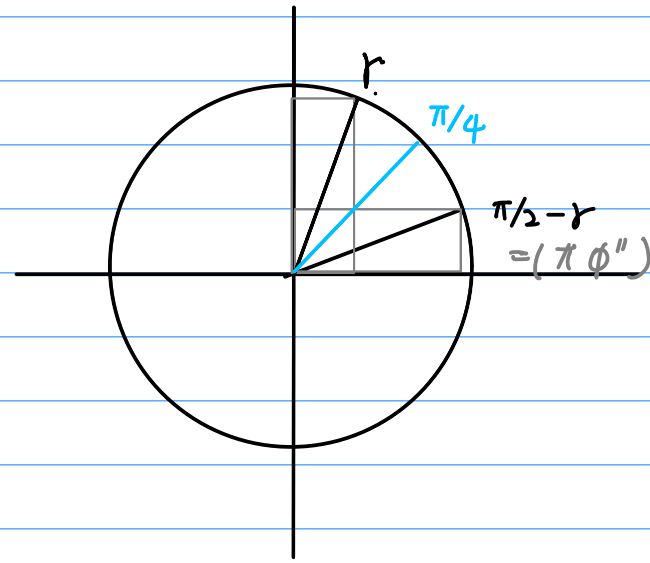
$$\phi \; [0, 2\pi] \longrightarrow [0, \tfrac{\pi}{4}]$$

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

$$\underset{\phi}{[-1, 1]} \longrightarrow \underset{\theta}{[0, \pi/4]} \longrightarrow \text{sine/cosine generator}$$

$$\longrightarrow \pi\phi$$

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 $\sin\theta$, $\cos\theta$

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

2 msb of the normalized angle ϕ

: MSB_1, MSB_2 determine the quadrant of $\pi\phi$
MSB_3 determines the upper / lower half of the quadrant

Control / Interchange

$MSB_1 \leftarrow 0$, $MSB_2 \leftarrow 0$ $\pi\phi'$

$MSB_3 = 1$: the upper half quadrant

$\sin r = \cos(\pi/2 - r)$
$\cos r = \sin(\pi/2 - r)$

the normalized angle below $\pi/4$ ϕ''

$\phi'' = 0.5 - \phi'$ $(MSB_3 = 1)$
$\phi'' = \phi'$ $(MSB_3 = 0)$

$\theta = \pi\phi''$ hardwired multiplier
$[0, \pi/4]$

⌈ CORDIC-based architectures
the elementary angles are divided by π

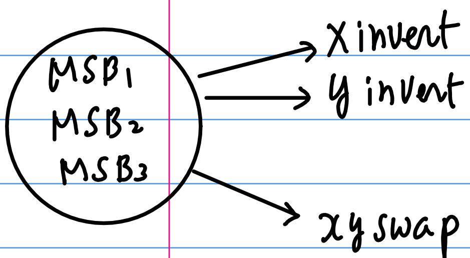
$$\theta_k = \tan 2^{-k} / 2\pi$$

since the directions of the subrotations are controlled only by the sign of the difference between two angles and not the magnitude, the multiplication by π is not required ⌋

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

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

negation / interchange

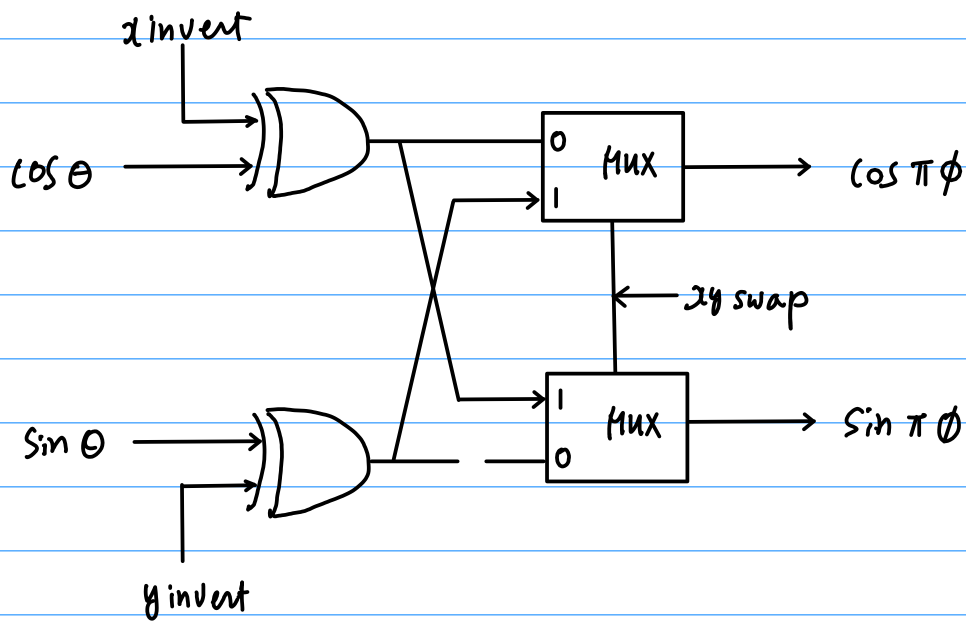


the negation of $\cos\theta = X_{N+1}$

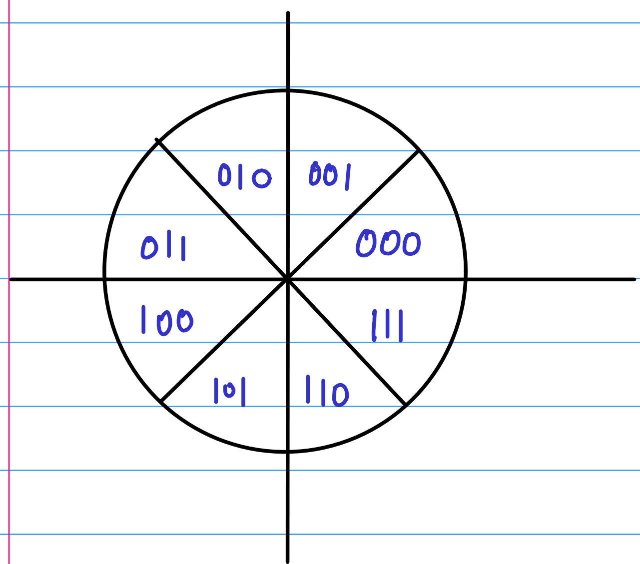
$\sin\theta = Y_{N+1}$

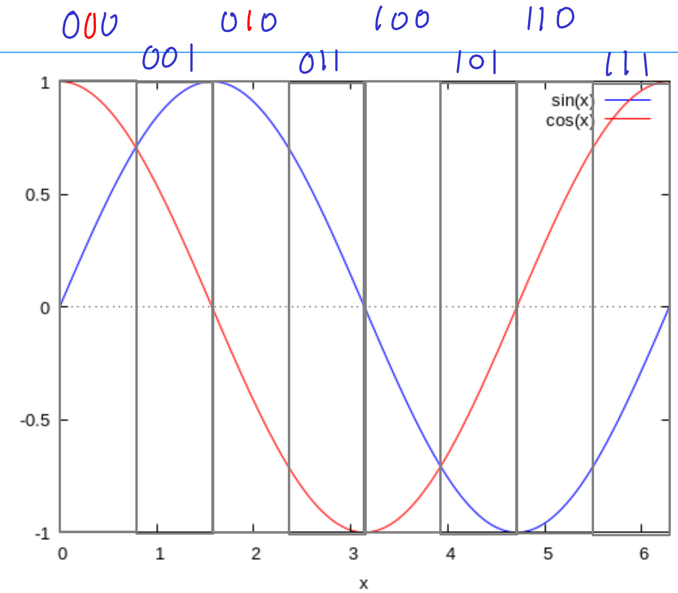
Interchange

negate before swap



Madisetti & Willson, DDS Freq synthesizer



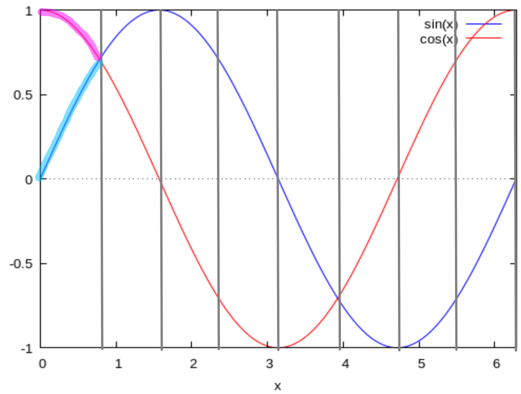


	cos	sin.			
	xinv	yinv	swap	$\cos \pi \theta$	$\sin \pi \phi$
000	0	0	0	$\cos \theta$	$\sin \theta$
001	0	0	1	$\sin \theta$	$\cos \theta$
010	0	1	1	$-\sin \theta$	$\cos \theta$
011	1	0	0	$-\cos \theta$	$\sin \theta$
100	1	1	0	$-\cos \theta$	$-\sin \theta$
101	1	1	1	$-\sin \theta$	$-\cos \theta$
110	1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

0

cos θ

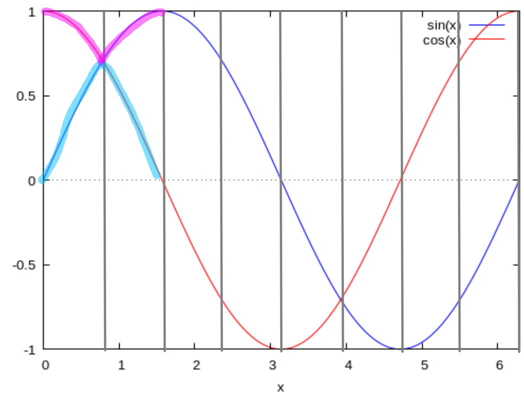
sin θ



1

sin θ

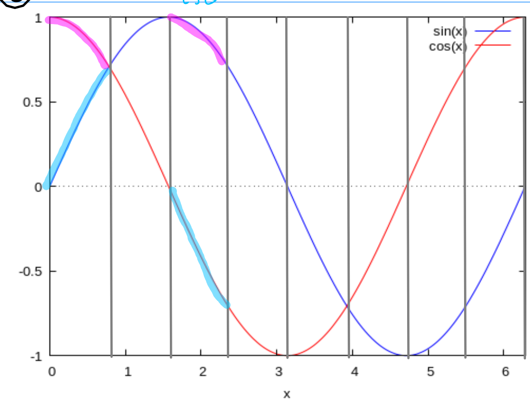
cos θ



2

−sin θ

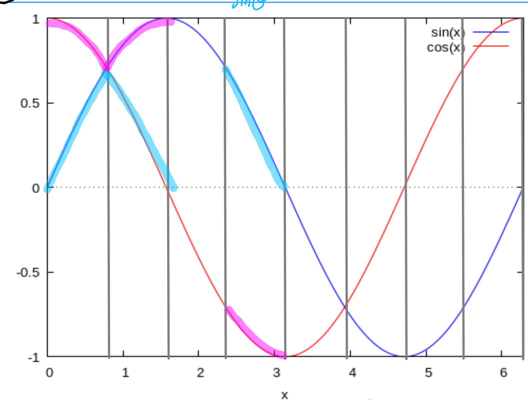
cos θ



3

−cos θ

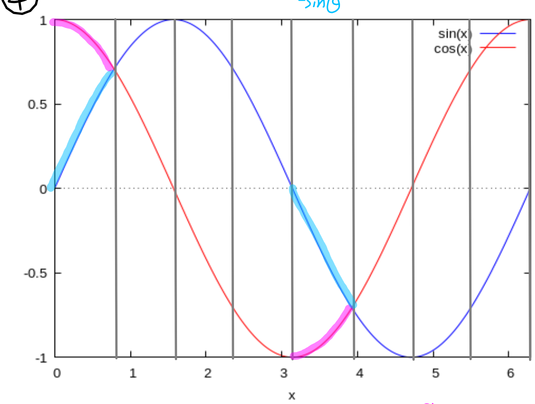
sin θ



4

−cos θ

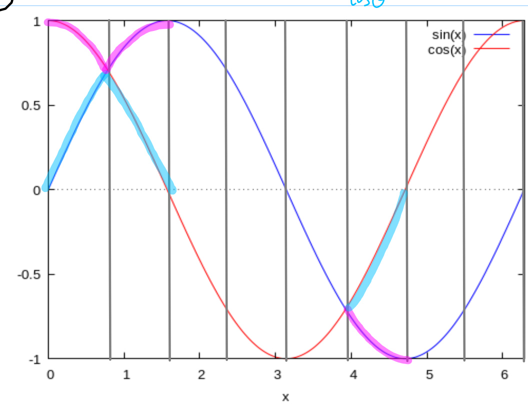
−sin θ



5

−sin θ

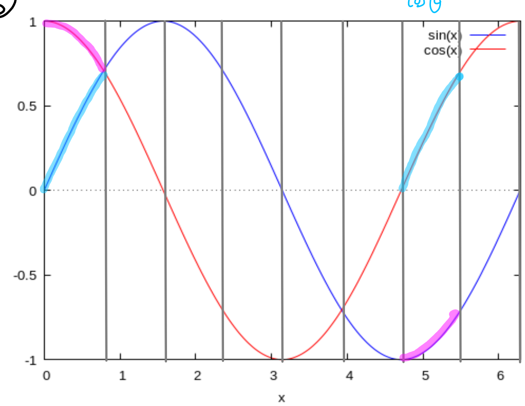
−cos θ



6

sin θ

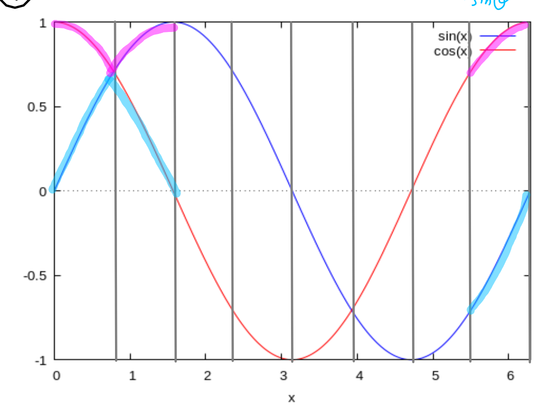
−cos θ



7

cos θ

−sin θ



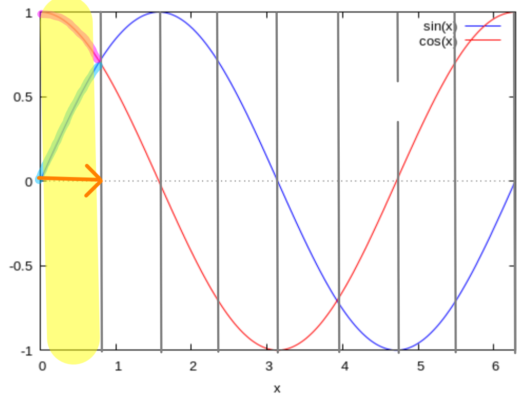
Madisetti & Willson, DDS Freq synthesizer

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

0

$\cos\theta$

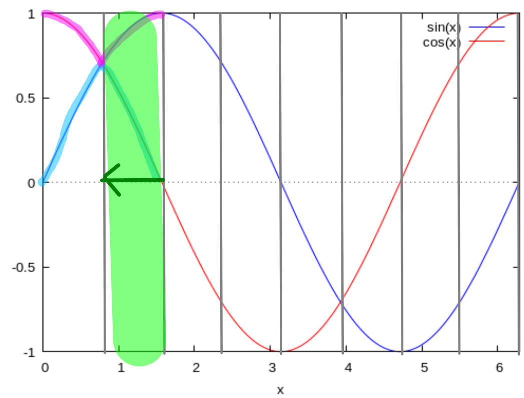
$\sin\theta$



1

$\sin\theta$

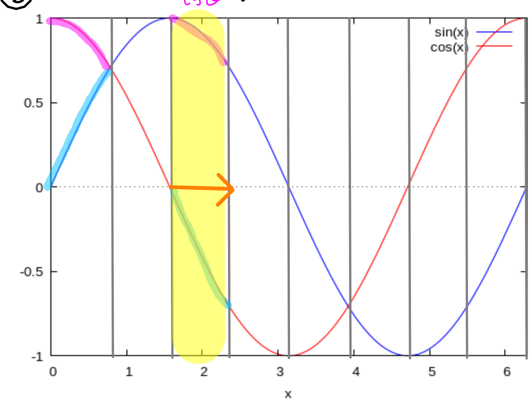
$\cos\theta$



2

$-\sin\theta$

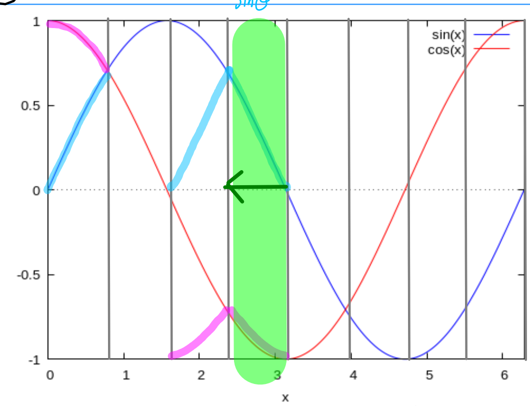
$\cos\theta$



3

$-\cos\theta$

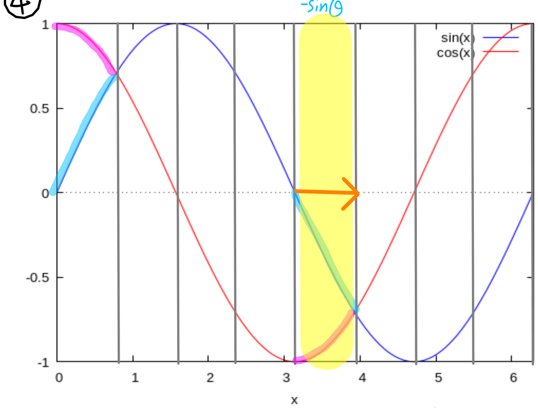
$\sin\theta$



4

$-\cos\theta$

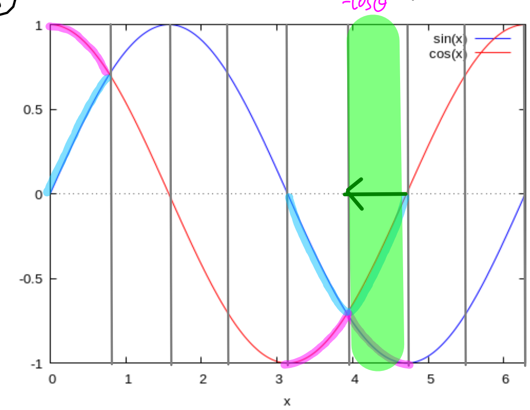
$-\sin\theta$



5

$-\sin\theta$

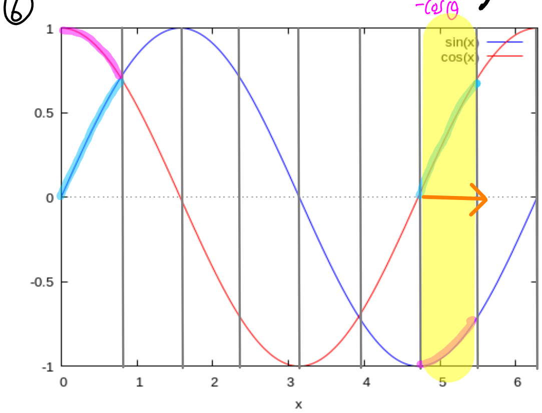
$-\cos\theta$



6

$\sin\theta$

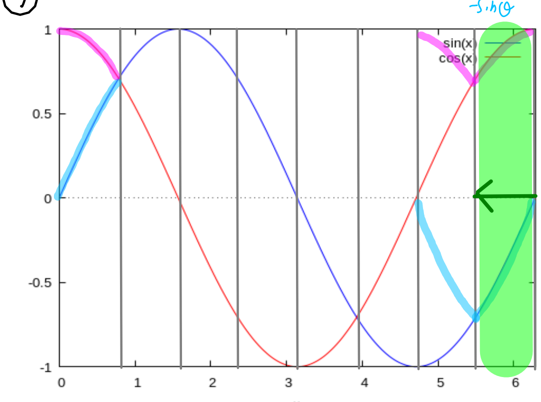
$-\cos\theta$



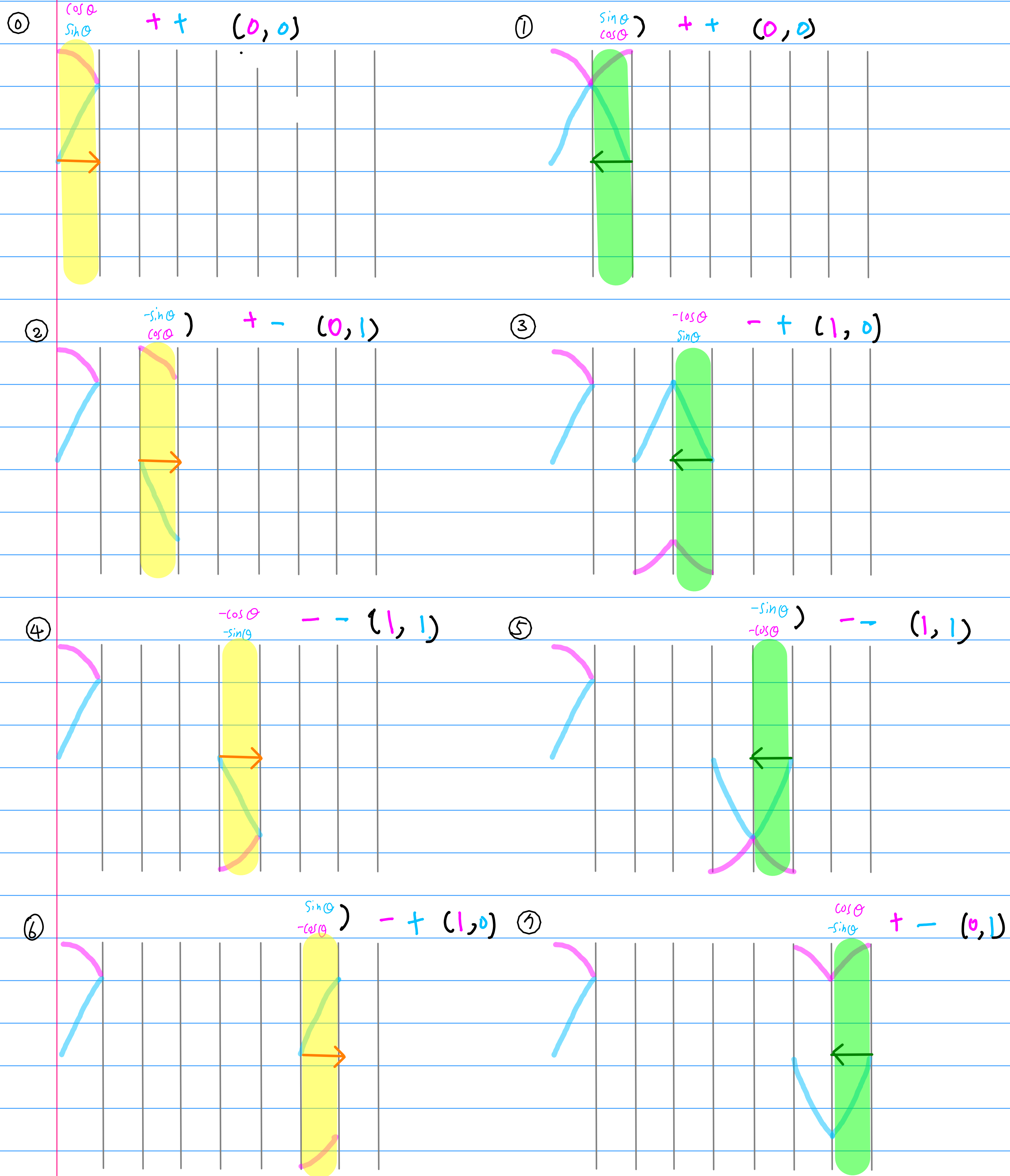
7

$\cos\theta$

$-\sin\theta$



$\sin \phi$


0
cos θ
sin θ
+ +
(0, 0)
1
sin θ
cos θ)
+ +
(0, 0)
2
-sin θ
cos θ)
+ -
(0, 1)
3
-cos θ
sin θ
- +
(1, 0)
4
-cos θ
-sin θ
- -
(1, 1)
5
-sin θ
-cos θ)
- -
(1, 1)
6
sin θ
-cos θ)
- +
(1, 0)
7
cos θ
-sin θ
+ -
(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}

⓪ 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 ← $b_k = 1$
  neg rotation ← $b_k = 0$

Combining all the fixed rotations

→ initial fixed rotation

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

initial fixed rotation

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

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

the rotation after recoding

— a fixed initial rotation ϕ_0

a sequence of $\oplus / \ominus$ rotations

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

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

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

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

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

The recoding need not be explicitly performed

Simply replacing $b_k = 0$ with -1

This recoding maintains

a constant scaling factor 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 $\theta_k = 2^{-k}$ used in recoding
the subangles $\theta_k = \tan^{-1}(2^{-k})$ 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.

positive subrotation by 2^{-k} radians

= sum of 2 equal positive half rotations by 2^{-k-1}

zero subrotations