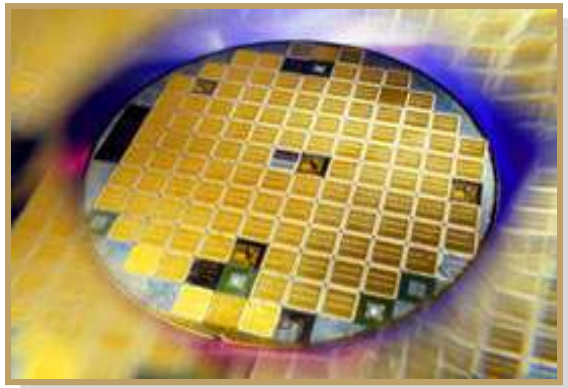
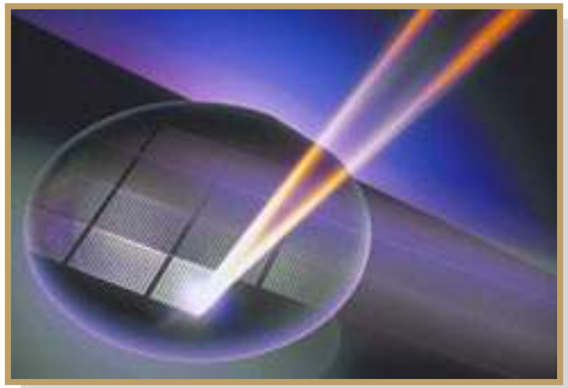
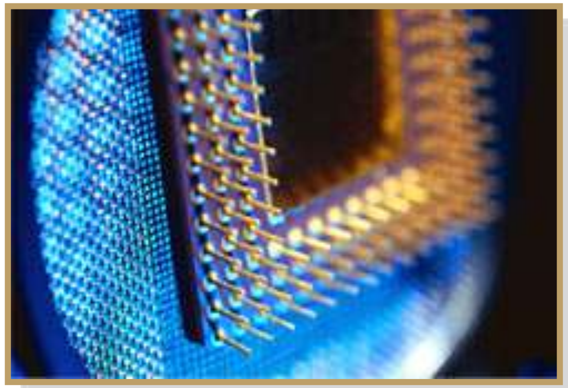
《VLSI数字通信原理与设计》课程 主讲人 贺光辉

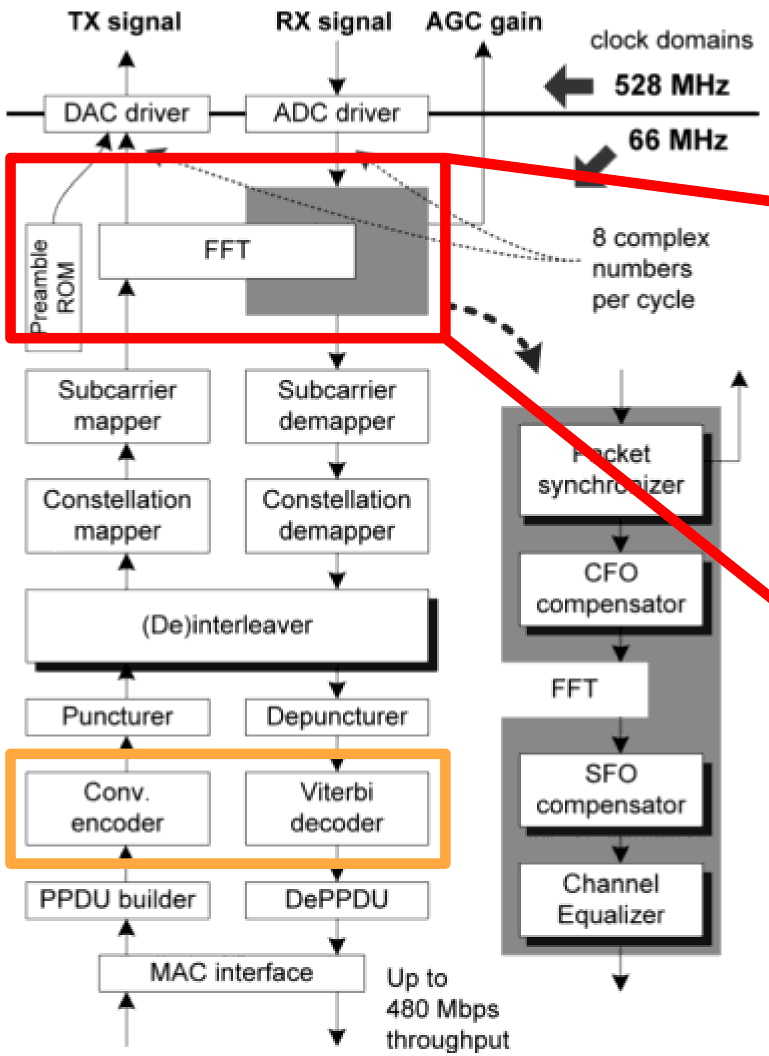
第十一讲：调制解调硬件设计

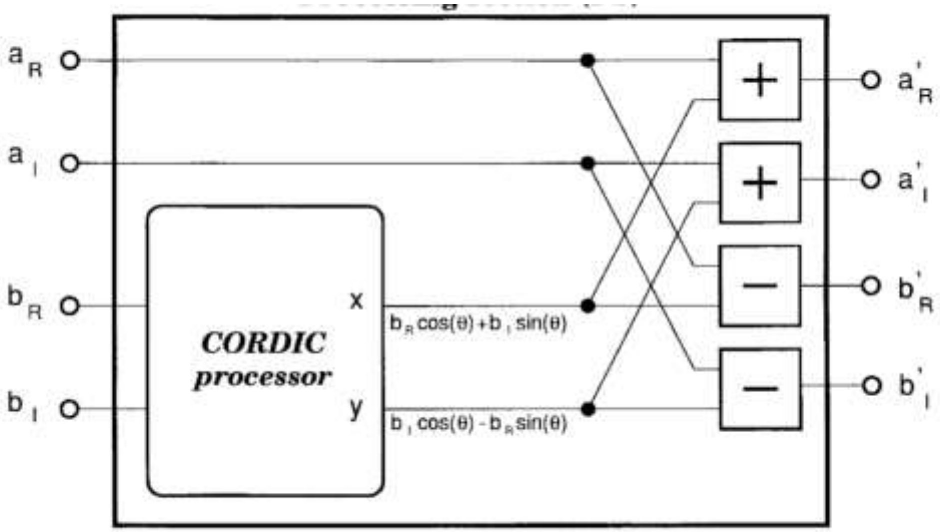






1. 基带收发器系统架构（1）



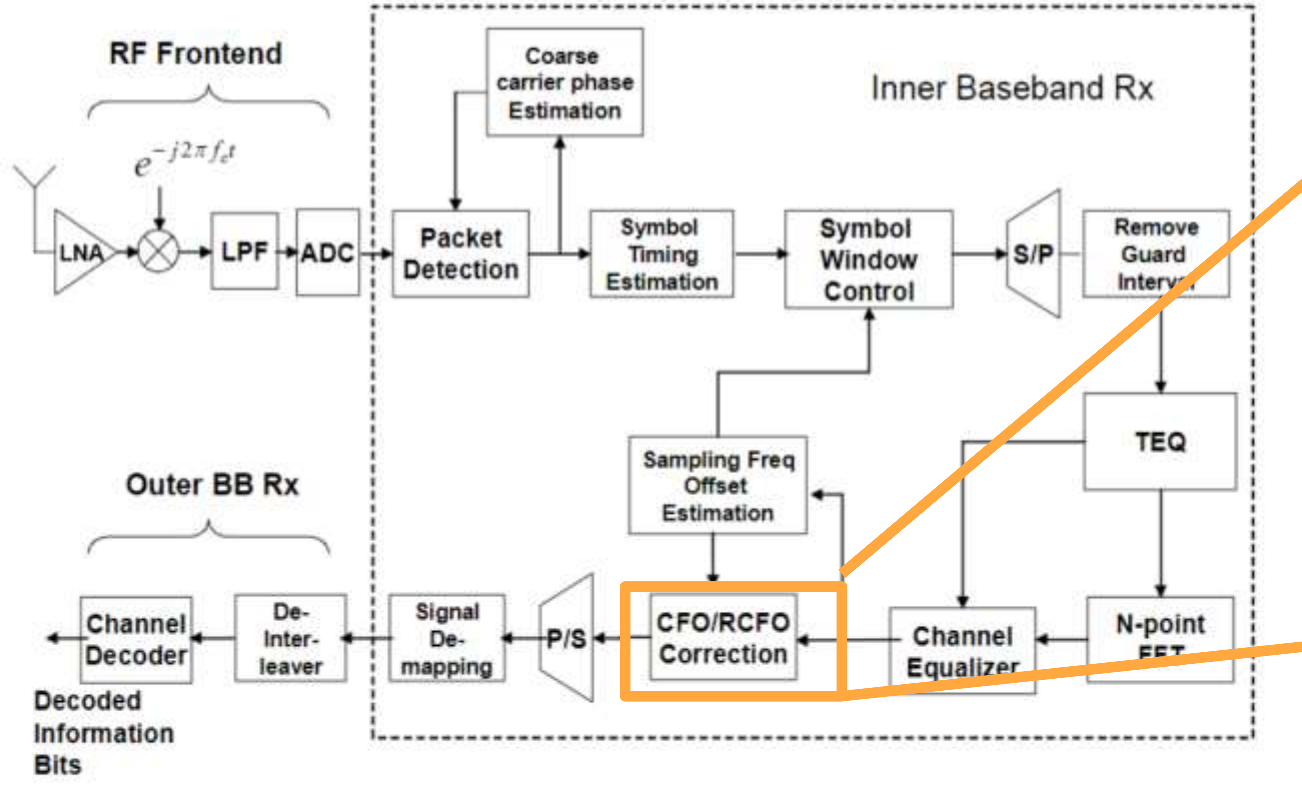


调制技术：OFDM

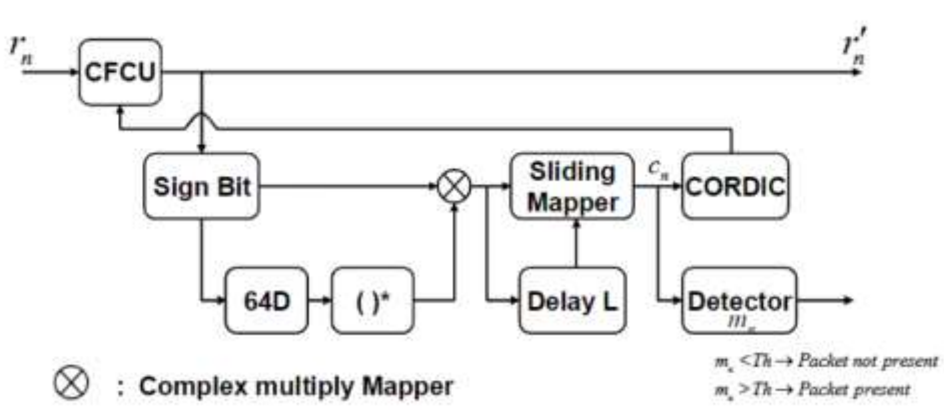
信道编码：卷积码
信道解码：维特比译码

S. J. Hwang, Y. Han, S. W. Kim, J. Park and B. G. Min, "Resource Efficient Implementation of Low Power MB-OFDM PHY Baseband Modem With Highly Parallel Architecture," in IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol. 20, no. 7, pp. 1248-1261, July 2012.

1. 基带收发器系统架构（3）



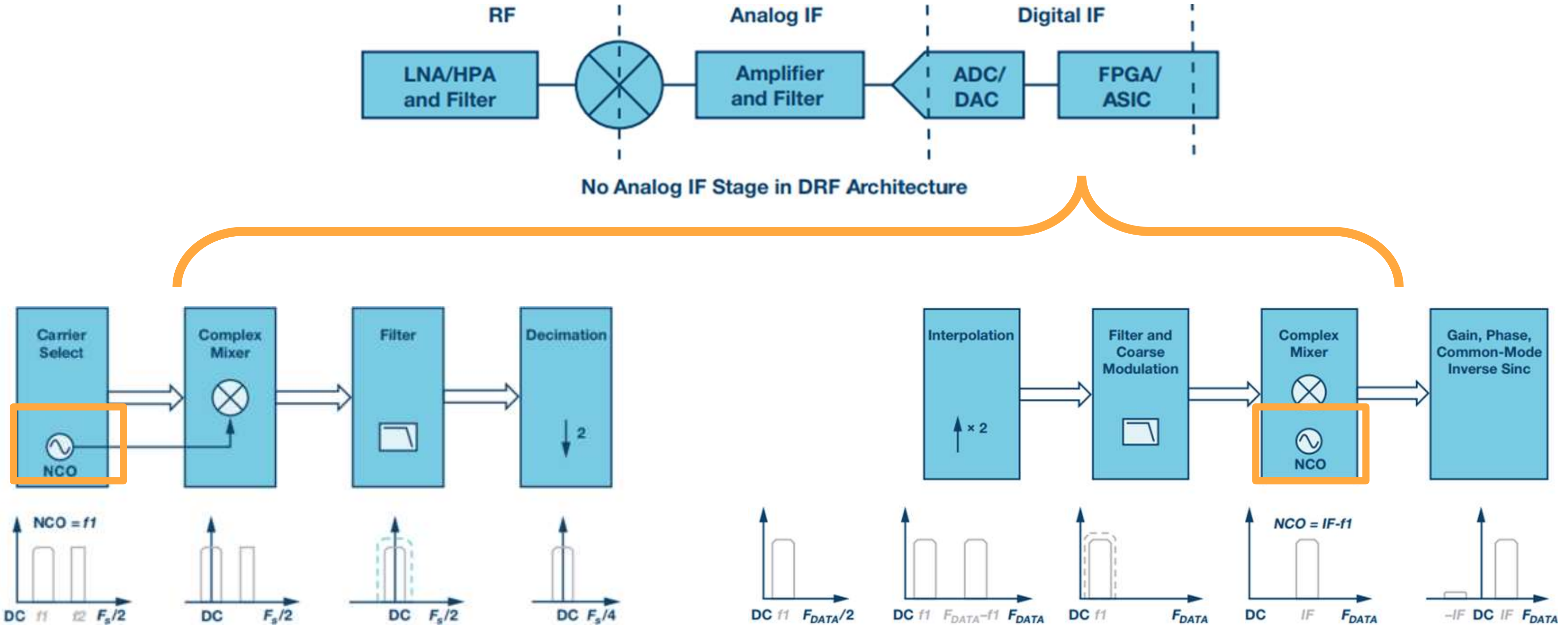
WiMAX receiver system block diagram



Architecture of PD and CFO compensator

Chang C C , Su C H , Wu J M . A low power baseband OFDM receiver IC for fixed WiMAX communication[C]// Asian Solid-state Circuits Conference. IEEE, 2007.

1. 基带收发器系统架构（4）



DDC block diagram

DUC block diagram

2.调制解调中的硬件设计 ——广播电台中的调制与解调

调制

解调

语音信号频率为

300~3400Hz

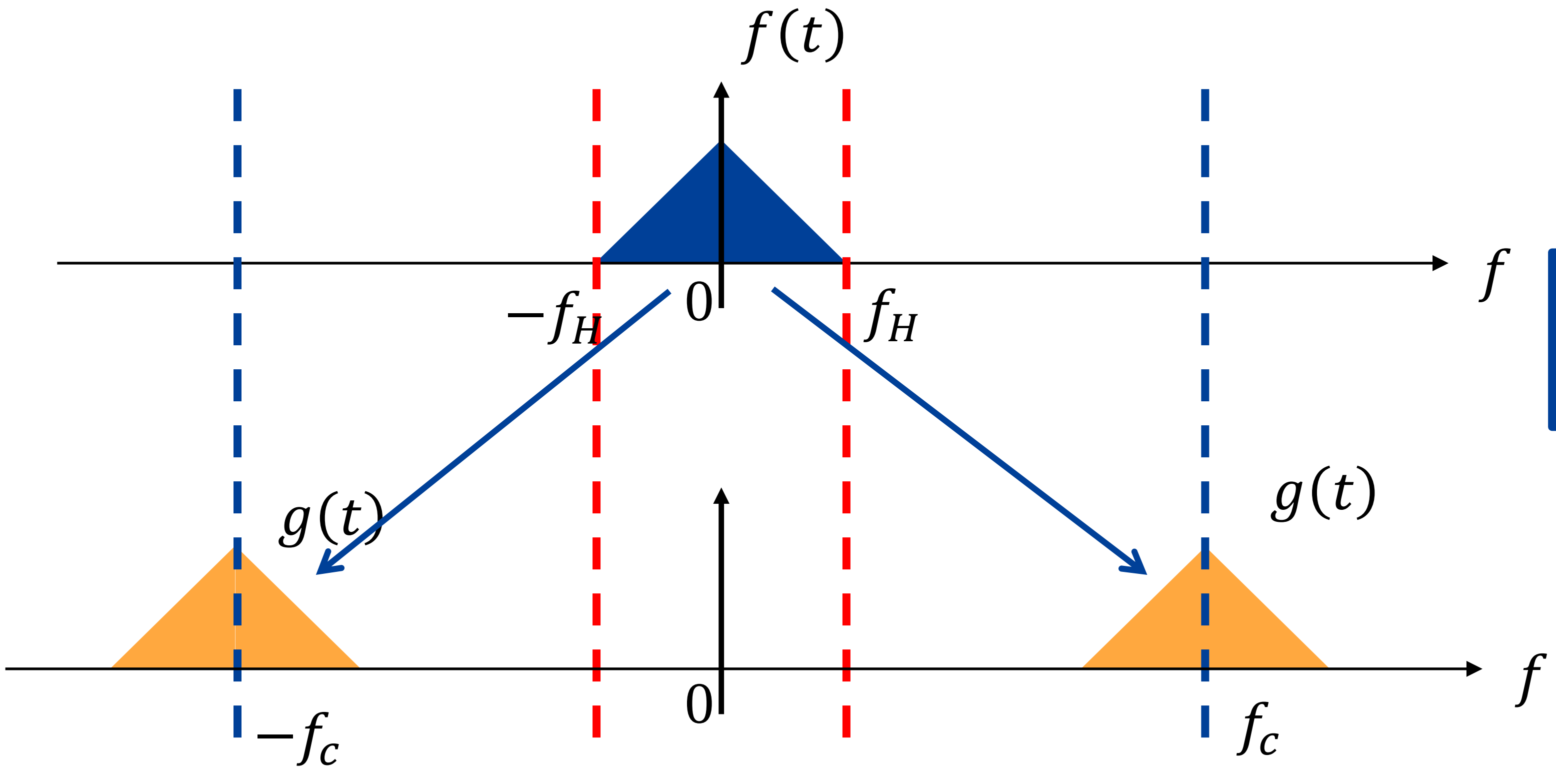
为什么需要**调制**技术?

- 天线长度限制
- 频段限制

FM收音机频段

87.5~108MHz

2.调制解调中的硬件设计 ——调制的原理



原始信号$f(t)$

调制

$g(t) = f(t) \cdot cos(2\pi f_c t)$

中频/射频信号

$g(t)$

$$g(t) = f(t) \cdot cos(2\pi f_c t)$$

难点是 cos 的计算!

坐标旋转数字计算机

(Coordinate Rotation Digital Computer, CORDIC)

02 调制解调器

2.1 CORDIC基本原理 —— 三角函数计算

查表法：需要大量储存空间

级数展开法

$$cosx = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \cdots$$

$$sinx = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \cdots$$

计算量大

有没有更有效的方法?

角度	正弦值	余弦值
0°	0.0000	1.0000
0.1°	0.0017	1.0000
0.2°	0.0035	1.0000
…	…	…
29.9°	0.4985	0.8669
30.0°	0.5000	0.8660
30.1°	0.5015	0.8652
…	…	…
89.9°	1.0000	0.0017
90.0°	1.0000	0.0000

2.1 CORDIC基本原理 —— 简介

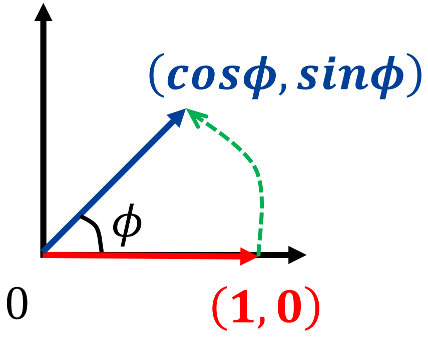
坐标旋转数字计算机（Coordinate Rotation Digital Computer, CORDIC ）

- 通过移位和加减运算，递归计算函数值的算法

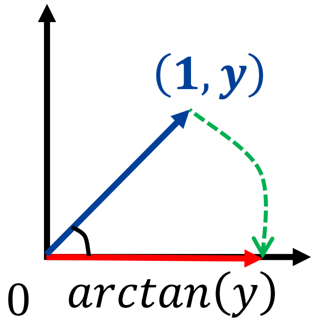
两种模式

旋转化： 将向量(x, y)旋转特定角度ϕ到(x', y')

向量化： 旋转向量(x, y)，使其落在x轴上



如：旋转$(1, 0)$角度ϕ，得到$(cos\phi, sin\phi)$



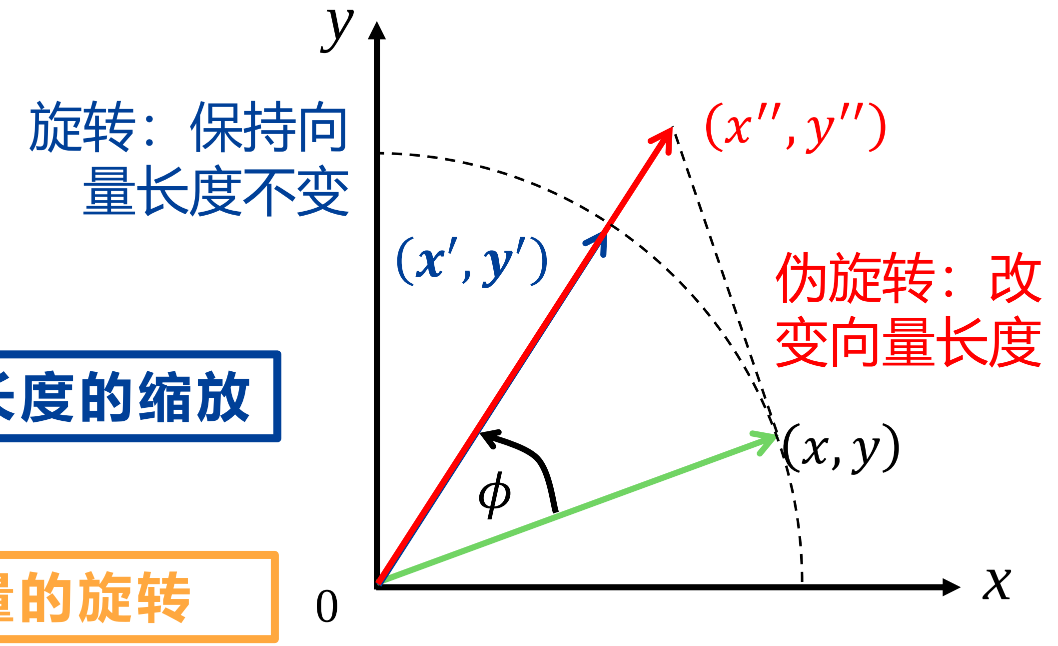
如：旋转$(1, y)$到x轴，旋转角度为$arctan(y)$

J. E. Volder
1959年提出

2.1CORDIC基本原理 —— 向量旋转原理

向量旋转公式：将向量(x, y)逆时针旋转ϕ角度得到新向量(x', y')

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} cos\phi & -sin\phi \\ sin\phi & cos\phi \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$



提出公因子$cos\phi$有

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \boxed{cos\phi} \cdot \boxed{\begin{bmatrix} 1 & -tan\phi \\ tan\phi & 1 \end{bmatrix}} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

对向量长度的缩放

对向量的旋转

为减少计算量，引入伪旋转

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} 1 & -tan\phi \\ tan\phi & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

如何用有限次旋转逼近任意角度？

2.1 CORDIC基本原理 —— 旋转方法

例：如何最快用有限已知角旋转出30°？

二分法!

- 初始：　$\phi = 0^\circ$　　　$(< 30^\circ，\ +)$
- 步骤1：　$\phi = 0^\circ + 45^\circ = 45^\circ$　　　$(> 30^\circ，\ -)$
- 步骤2：　$\phi = 45^\circ - 22.5^\circ = 22.5^\circ$　　　$(< 30^\circ，\ +)$
- 步骤3：　$\phi = 22.5^\circ + 12.25^\circ = 33.75^\circ$　　　$(> 30^\circ，\ -)$
- 步骤4：　$\phi = 33.75^\circ - 5.625^\circ = 28.125^\circ$　　　$(< 30^\circ，\ +)$
- $...$

多次旋转后 $\phi = 0^\circ + 45^\circ - 22.5^\circ + 11.25^\circ - 5.625^\circ + \cdots \approx 30^\circ$

第一次旋转45°，以后逐次减半

步骤	$\Delta\phi$
1	45°
2	22.5°
3	11.25°
4	5.625°
5	2.8125°
6	…

2.1 CORDIC基本原理 —— 旋转方法

例：如何最快用有限已知角旋转出30°？

二分法!

- 初始： $\phi = 0^\circ$ $(< 30^\circ，\ +)$
- 步骤1： $\phi = 0^\circ + 45^\circ = 45^\circ$ $(> 30^\circ，\ -)$
- 步骤2： $\phi = 45^\circ - 22.5^\circ = 22.5^\circ$ $(< 30^\circ，\ +)$
- 步骤3： $\phi = 22.5^\circ + 12.25^\circ = 33.75^\circ$ $(> 30^\circ，\ -)$
- 步骤4： $\phi = 33.75^\circ - 5.625^\circ = 28.125^\circ$ $(< 30^\circ，\ +)$
- $...$

多次旋转后 $\phi = 0^\circ + 45^\circ - 22.5^\circ + 11.25^\circ - 5.625^\circ + \cdots \approx 30^\circ$

n次旋转，需计算$2n$次乘法

$$x'' = x - y \cdot tan\Delta\phi$$

$$y'' = y + x \cdot tan\Delta\phi$$

能否减少乘法开销

CORDIC将$tan\Delta\phi$设计为2^{-i}形式，把乘法转化为移位操作

步骤	$\Delta\phi$	$tan\,\Delta\phi$
1	45°	1
2	22.5°	0.41421
3	11.25°	0.19891
4	5.625°	0.09849
5	2.8125°	0.04913
…		

每次旋转角度减半

步骤	$\Delta\phi$	$tan\,\Delta\phi$
1	45°	1
2	26.565°	0.5
3	14.036°	0.25
4	7.1250°	0.125
5	3.5763°	0.0625
…		

每次旋转正切值减半

当$tan\Delta\phi=2^{-i}$时

$$x''=x-y\cdot tan\Delta\phi$$

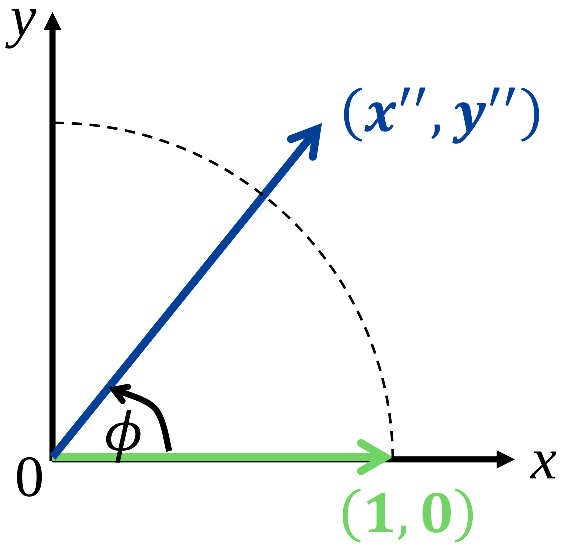
$$y''=y+x\cdot tan\Delta\phi$$

乘法避免
收敛速度降低

2.1 CORDIC基本原理 —— 系数补偿

n次旋转，经过$2n$次移位和加法操作，将$(1,0)$旋转ϕ角度到(x'',y'')

$x''=cos\phi, y''=sin\phi$是否成立



n 次旋转后补偿系数为

$$k=cos\,45^{\circ}\cdot cos\,26.565^{\circ}\cdot cos14.036^{\circ}\cdot\cdots$$

$$\approx 0.60725\ (n\to\infty)$$

伪旋转：改变向量长度

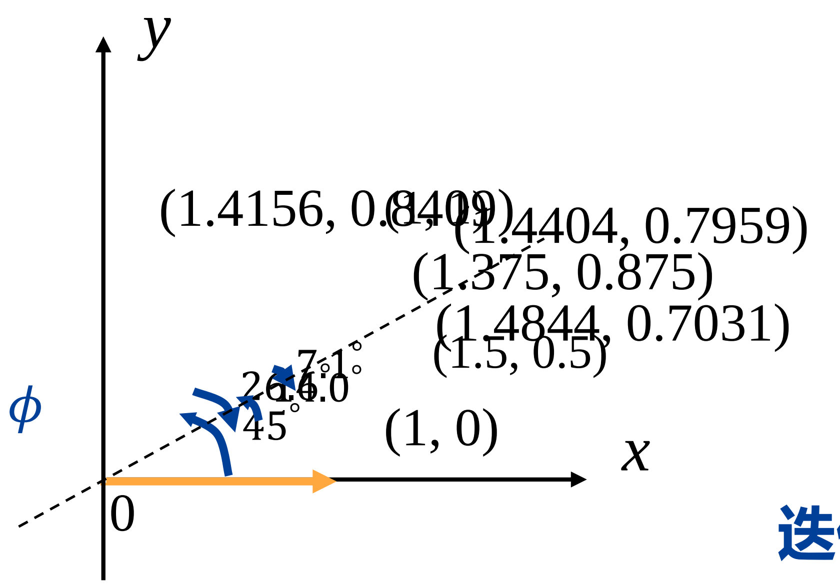
$$x''=x-y\cdot tan\Delta\phi$$
$$y''=y+x\cdot tan\Delta\phi$$

旋转：不改变向量长度

$$x'=\boxed{cos\Delta\phi}\cdot(x-y\cdot tan\Delta\phi)$$
$$y'=\boxed{cos\Delta\phi}\cdot(y+x\cdot tan\Delta\phi)$$

2.1 CORDIC基本原理 —— CORDIC旋转示例（1）

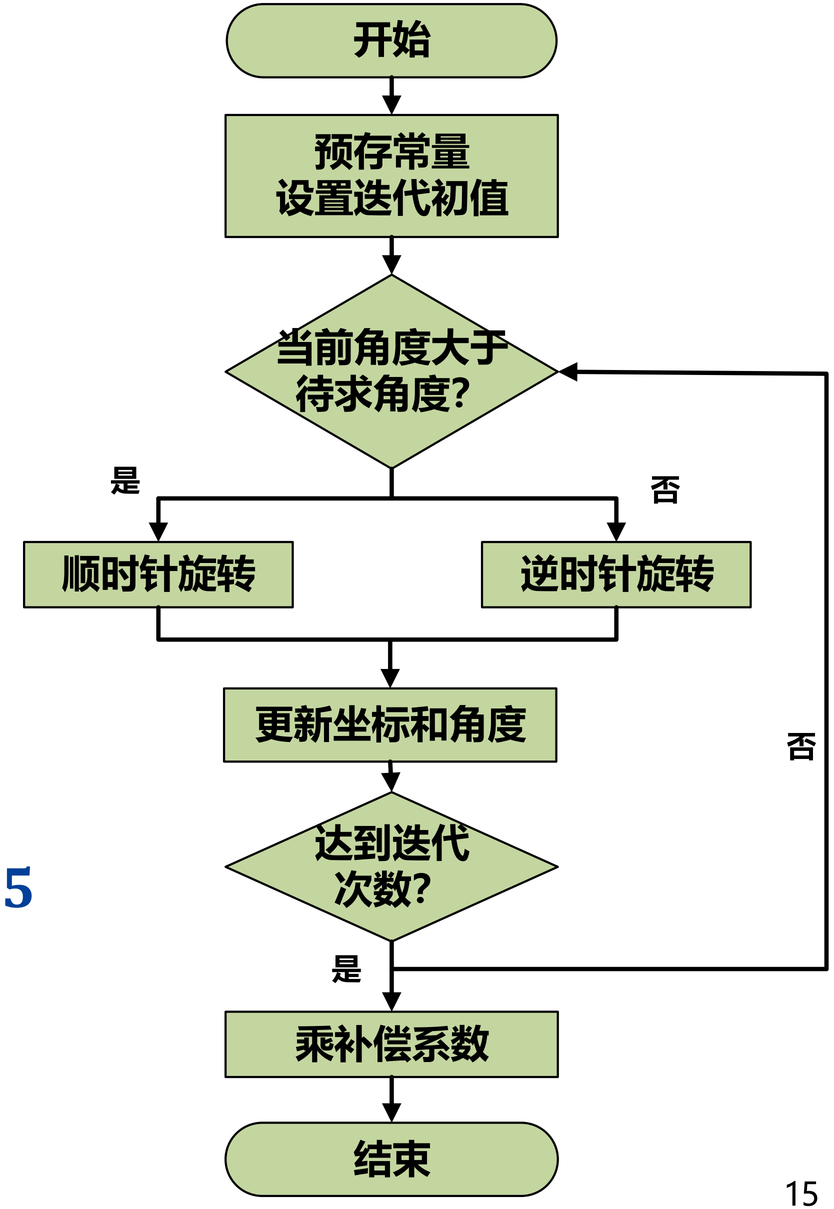
工作过程



ϕ	$\Delta\phi$	x	y
0°	+45°	1	0
45°	-26.565°	1	1
18.435°	+14.036°	1.5	0.5
32.471°	-7.1250°	1.375	0.875
25.346°	+3.5763°	1.4844	0.7031
28.923°	+1.7899°	1.4404	0.7959
30.712°	-0.8952°	1.4156	0.8409

迭代6次时，补偿系数$k = 0.60735$

$$\cos 30^\circ = k \cdot x = 0.8598$$
$$\sin 30^\circ = k \cdot y = 0.5107$$



随着迭代次数增加，计算精度提升

CORDIC能表示的角度最大范围为

$$\phi = 45^{\circ} + 26.6^{\circ} + 14.0^{\circ} + 7.1^{\circ} + \cdots = 99.7^{\circ}$$

如何计算360°范围的角度?

$$\cos(\phi + 90^{\circ}) = -\sin\phi$$
$$\sin(\phi + 90^{\circ}) = \cos\phi$$
$$\cos(\phi + 180^{\circ}) = -\cos\phi$$
$$\sin(\phi + 180^{\circ}) = -\sin\phi$$

可以将任意角度转换到0°~90°求解

迭代次数	x	y
7	0.86762	0.49724
8	0.86370	0.50400
9	0.86567	0.50062
10	0.86664	0.49892
11	0.86616	0.49977
12	0.86591	0.50020
…		

$\cos 30^{\circ} = 0.86603$
$\sin 30^{\circ} = 0.5$

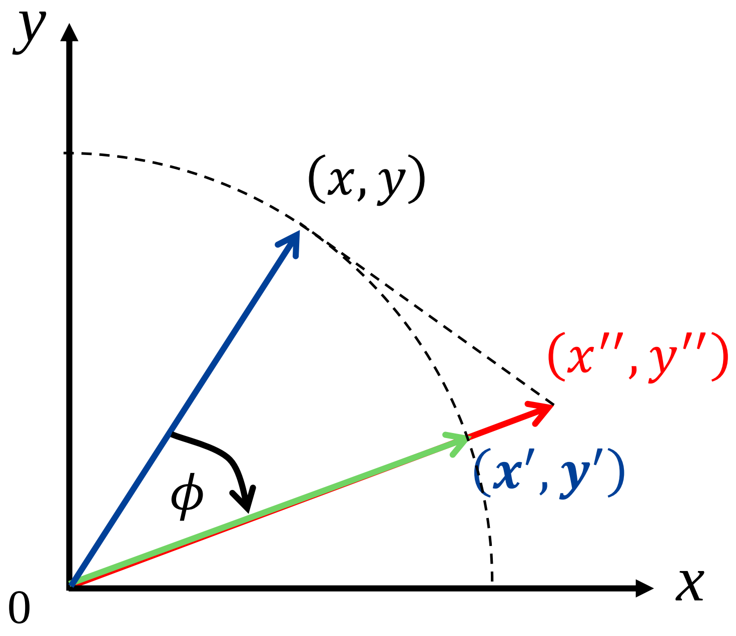
2.1 CORDIC基本原理—— 向量方法

向量方法：

顺时针旋转向量（x , y）使其落在x轴上

向量旋转公式：

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\phi & \sin\phi \\ -\sin\phi & \cos\phi \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix} = \cos\phi \cdot \begin{bmatrix} 1 & -\tan\phi \\ \tan\phi & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$



伪旋转：

$$\begin{bmatrix} x'' \\ y'' \end{bmatrix} = \begin{bmatrix} 1 & \tan\phi \\ -\tan\phi & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \end{bmatrix}$$

符号与旋转方法相反

口通过旋转使得纵坐标y值趋近于0，可用于计算arctan（y/x）

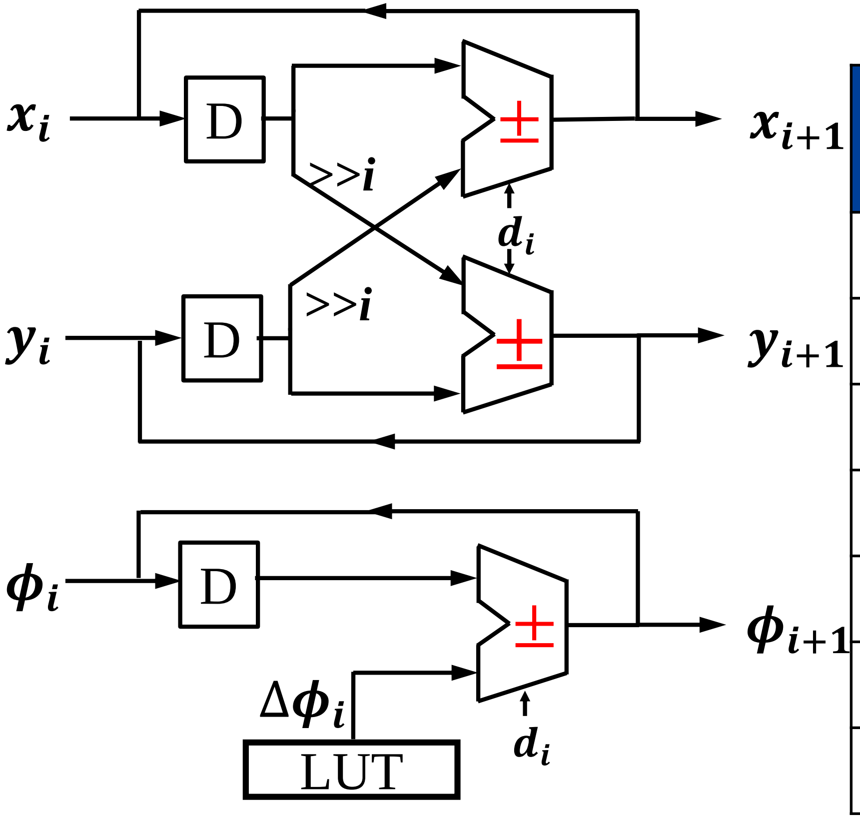
2.2 CORDIC硬件结构 —— 迭代结构

$$x'' = x - y \cdot tan\Delta\phi$$
$$y'' = y + x \cdot tan\Delta\phi$$

$$\phi_{i+1} = \phi_i + d_i \cdot \Delta\phi_i$$
$$x_{i+1} = x_i - y_i \cdot d_i \cdot 2^{-i}$$
$$y_{i+1} = y_i + x_i \cdot d_i \cdot 2^{-i}$$

其中d_i表示旋转方向，逆时针旋转为1，顺时针旋转为 -1

迭代串行结构



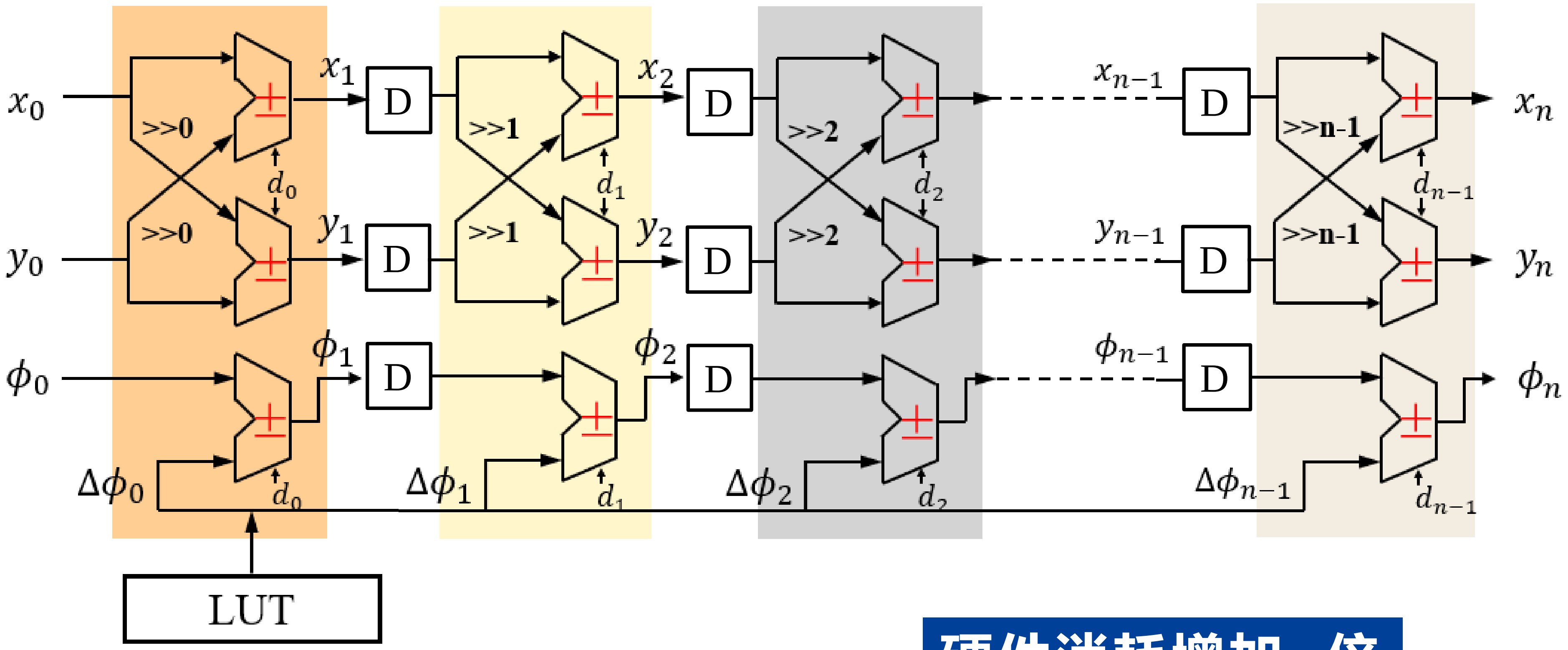
LUT储存的定点值

$tan\Delta\phi$	$\Delta\phi$ 浮点值	$\Delta\phi$ 定点值
2^{-0}	45°	16’h500
2^{-1}	26.565°	16’h3521
2^{-2}	14.036°	16’h1c12
2^{-3}	7.1250°	16’h0e40
2^{-4}	3.5763°	16’h0727
2^{-5}	1.7899°	16’h0394
…		

注：定点值用16位表示

迭代次数较多时，计算速度慢

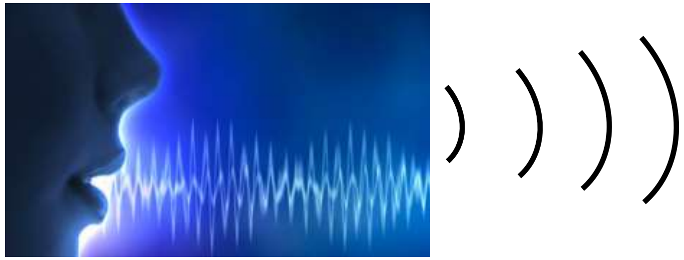
n级流水线结构如下



硬件消耗增加n倍
处理速度提升n倍

2.3 CORDIC应用实例 —— 调制技术的硬件实现

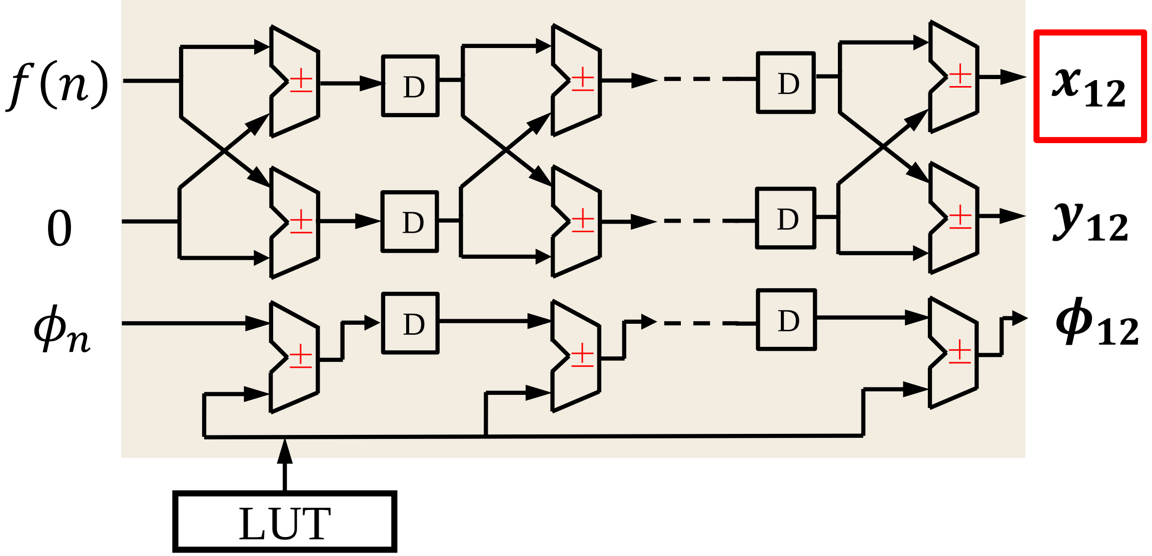
例：语音信号$f(t)$频率为 $f_H = 1k\mathrm{Hz}$，
而广播电台频段为 $f_c = 97.9\mathrm{MHz}$。
如何通过CORDIC实现调制？



步骤1： 以$f_s = 2f_H$的频率对$f(t)$采样得到信号$f(n)$，$n = 0, 1, 2 \ldots\ldots$

步骤2： 确定角度值 $\phi_n = 2\pi f_c n / f_s$，并映射到 $-90^\circ \sim 90^\circ$

步骤3： 用12级流水线结构CORDIC电路计算$g(n) = f(n) \cdot cos(\phi_n)$
输入$x_0 = f(n)$，$y_0 = 0$，$\phi_0 = \phi_n$



总结

调制解调中的硬件设计：CORDIC

- 使用CORDIC计算三角函数的步骤为：
  - 设置迭代初值
  - 确定旋转方向
  - 更新旋转角度和坐标
  - 补偿系数
- CORDIC算法易于硬件实现，通过流水线能提升处理速度