UNIVERSITY OF MASSACHUSETTS
Dept. of Electrical & Computer Engineering

Digital Computer Arithmetic
ECE 666

Part 7b
Fast Division - II

Israel Koren

High Radix Division

- * Number of add/subtracts in radix-2 SRT is data-dependent
- * Asynchronous circuit needed to use reduced number of nonzero bits in quotient
- * Increasing number of 0's in quotient - limited practical significance

♦ Number of add/subtracts reduced by increasing radix β

- * $\beta = 2^m$ - m quotient bits generated each step
- * Number of steps reduced to $\lceil n/m \rceil$

♦ Recursive equation for remainder -

♦ $r_i = \beta r_{i-1} - q_i D$

- * Multiply by $\beta=2^m$ - shift left remainder by m bit positions

♦ Digit set for quotient:

- * $0,1,...,\beta-1$ for restoring division
- * Up to $\overline{\beta-1},...,\overline{1},0,1,...,\beta-1$ for high-radix SRT division

High Radix Restoring Division

- All previous division algorithms can use radix > 2
- Restoring division -
  - Initial guess $q_i=1$
  - If remainder $\beta r_{i-1}-D>0$ - increase to $q_i=2$
  - Subtract D from temporary remainder: $\beta r_{i-1}-2D$
  - Repeat until $q_i=j$: temporary remainder $\beta r_{i-1}-jD$ negative
  - Remainder restored by adding D: $\beta r_{i-1}-(j-1)D$; $q_i=j-1$
- Time-consuming - no advantage over binary algorithm
- Can be parallelized by circuits comparing βr_{i-1} to several multiples jD - selecting smallest positive remainder; substantial hardware requirement
- Binary nonrestoring division - similar changes

ECE666/Koren Part.7b.3

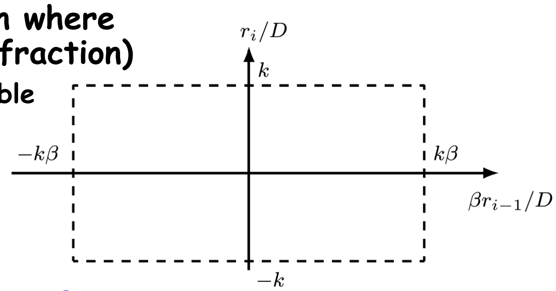
High-Radix SRT Algorithm

- Faster than binary version
- Quotient digit q_i - signed digit in range $\overline{\alpha},\overline{\alpha-1},\ldots,\overline{1},0,1,\ldots\alpha,$ where $\lceil 1/2(\beta-1)\rceil \leq \alpha \leq \beta-1$
- Finding possible choices for α in high-radix division:
- Quotient digit q_i selected so that $|r_i| < |D|$; otherwise, next quotient digit may be β or larger
  - Guarantees convergence of division procedure
  - For maximal remainder $r_{i-1}=D-ulp$ and positive D, largest value for $q_i=\alpha$ should guarantee r_i in allowable region

ECE666/Koren Part.7b.4

Reducing Remainder Range

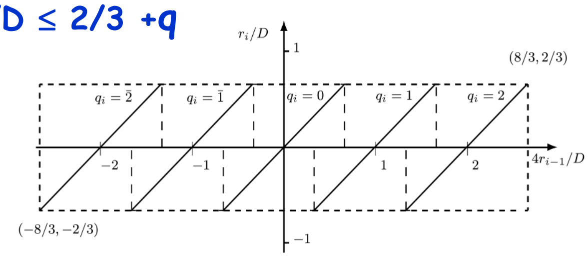
- $r_i = \beta\,(D\text{-}ulp) - \alpha D \le D\text{-}ulp$
- Select for α only maximum value $\beta-1$
- May consider division where $|r_i| \le k|D|$, (k is a fraction)
  - * reduce size of allowable region for remainder:



- $r_i = \beta\, k(D\text{-}ulp) - \alpha D \le k(D\text{-}ulp)$
- $\alpha \ge k(\beta-1) \;\Rightarrow\; k \le \alpha/(\beta-1)$
  - * $1/2 \le k \le 1$ allows selection of any α in $\lceil(\beta-1)/2\rceil \le \alpha \le \beta-1$
  - * Larger $k \Rightarrow$ larger redundancy for quotient

Example

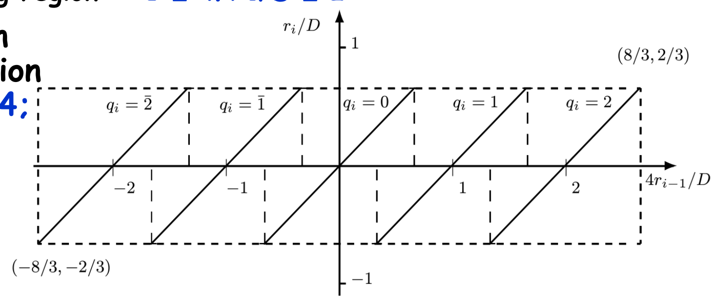
- $\beta=4$; $\alpha=2$; $k=\alpha/(\beta-1) = 2/3$
  - * $|r_i| \le kD = 2/3\ D$; $|\beta r_{i-1}| = |4r_{i-1}| \le 8/3\ D$
    or $|r_i/D| \le 2/3$ and $|4r_{i-1}/D| \le 8/3$
- Digit set for $q_i = \{\bar{2},\bar{1},0,1,2\}$
- Region for selecting q : $-2/3 \le 4r_{i-1}/D - q \le 2/3$
  or $-2/3 + q \le 4r_{i-1}/D \le 2/3 + q$
- Region examples:



  - * For selecting $q_i=2$: $4/3 \le 4r_{i-1}/D \le 8/3$
  - * For selecting $q_i=1$: $1/3 \le 4r_{i-1}/D \le 5/3$
  - * Overlapping region: $4/3 \le 4r_{i-1}/D \le 5/3$
    select either $q_i=1$ or $q_i=2$
  - * Similar overlapping regions exist for any 2 adjoining digits

Measure of Redundancy

- Ratio $k=\alpha/(\beta-1)$ - measure of redundancy in representation of quotient
  - Larger k $\Rightarrow$ larger overlap regions in plot of r_i/D vs $\beta r_{i-1}/D$
- Example: $\alpha=3;\ \beta=4;\ k=1$ - maximum redundancy
  - Region for $q_i=1$: $0 \le 4r_{i-1}/D \le 2$,
  - Region for $q_i=2$: $1 \le 4r_{i-1}/D \le 3$
  - Overlapping region : $1 \le 4r_{i-1}/D \le 2$
- Larger than overlap region for $\alpha=2;\beta=4;\ k=2/3$:

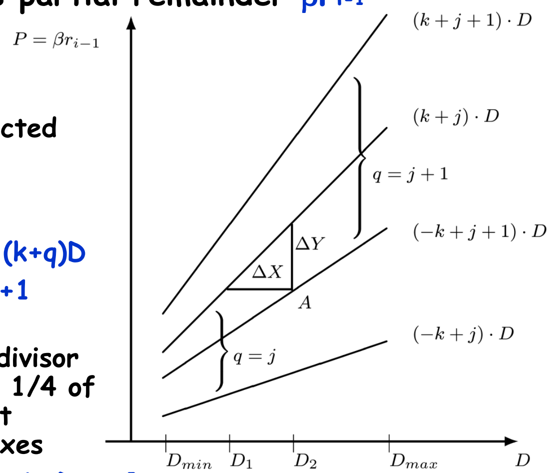


Implication of Overlap Region

- Provides choice of comparison constants for partial remainder and divisor
  - Can be selected to require as few digits as possible
  - Reducing execution time of comparison step when determining quotient digit
- Larger α $\Rightarrow$ larger overlap region $\Rightarrow$ larger choice $\Rightarrow$ fewer digits
- On the other hand, larger α $\Rightarrow$ more αD multiples $\Rightarrow$ extra hardware and/or time required
- For given α - determining number of bits of partial remainder and divisor to be examined is the most difficult step when developing high-radix SRT
  - Can be done numerically, analytically, graphically, or by combination of techniques

Graphical Approach: P-D Plot

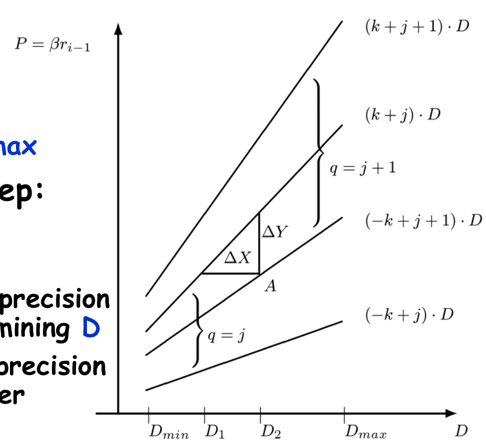
- Basic equation for partial remainder - $\beta r_{i-1} = r_i + q_i D$
- Notation: P = previous partial remainder βr_{i-1}
  - Partial remainder vs. Divisor plot - indicates regions in which given values of q may be selected
  - Limits on P for given q:
  - $-k D \leq r_i \leq k D$
  - Pmin=(-k+q)D ; Pmax=(k+q)D
  - Regions for q=j and q=j+1 overlap
  - Only positive values of divisor and partial remainder - 1/4 of complete P-D plot - plot symmetric about both axes
  - Only values of |D| in [Dmin,Dmax] are of interest - e.g. [0.5,1);[1,2) (IEEE floating-point)



Separating Selection Regions

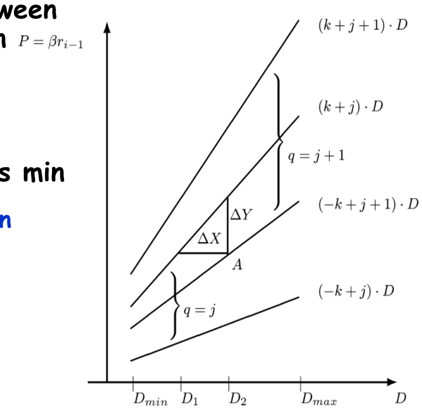
- Value of P separating selection regions of q=j & q=j+1
  - Serves as comparison constant
  - Its number of bits determines necessary precision when examining partial remainder to select q
- Line separating regions is horizontal (P=c ; selection of q independent of D) if and only if

  $(k+j)D_{min} \geq c \geq (-k+j+1)D_{max}$
- Otherwise - line is stairstep:
  - partitioning [Dmin,Dmax) into sub-intervals
  - “Stepping” points determine precision - number of digits - of examining D
  - Height of steps determines precision of examining partial remainder



Determining Precision

- **Notation**: ΔX (ΔY) - maximum width (height) of a step between D_1 and D_2
  - * Horizontal (vertical) distance between the 2 lines defining overlap region



- $\Delta X = D_2 - D_1 = P/(-k+j+1) - P/(k+j) = P(2k-1)/[j(j+1)+k(1-k)]$
  - * ΔX minimal when j is max and P is min
  - * $j_{max}=\alpha-1$; P minimal when $D_1=D_{min}$
  - * $\Delta X_{min}=D_{min}(k+\alpha-1)(2k-1)/[\alpha(\alpha-1)+k(1-k)]$
  - * $\Delta Y=(k+j)D-(-k+j+1)D=(2k-1)D$
  - * ΔY is minimal when $D = D_{min}$
- It is sufficient to consider overlapping region between $q=\alpha$ and $q=\alpha-1$ near D_{min}

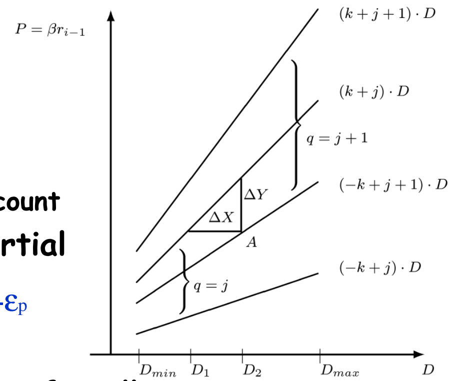
Precision - Cont.

- **Notation**: N_P (N_D) - number of examined bits of partial remainder (divisor) ;

  ε_P (ε_D) - number of fractional bits in N_P (N_D)
- Selecting q - look-up table implemented in a PLA (programmable logic array) with N_P+N_D inputs
- Minimizing size of look-up table speeds up division
- Precision of partial remainder ("truncated" divisor) - $2^{-\varepsilon_P}$ ($2^{-\varepsilon_D}$)
- $2^{-\varepsilon_D} \le \Delta X_{min}$; $2^{-\varepsilon_P} \le \Delta Y_{min}$
- Only upper bounds for precision - the 2 extreme points $\Delta X, \Delta Y$ may require higher precision - more than $\varepsilon_P, \varepsilon_D$ fractional bits

Using P-D Plot

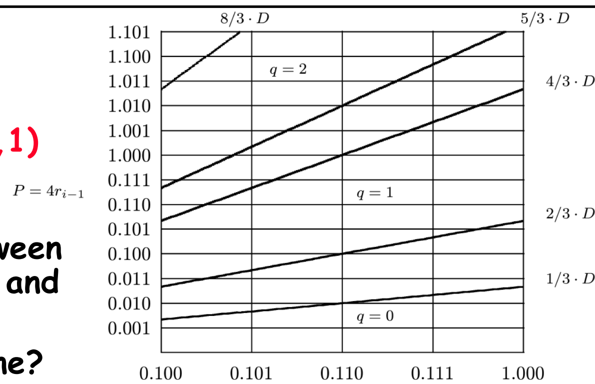
* To determine precision
* To select q for each P,D when truncated to N_P, N_D bits
* Limited precision taken into account



- Point (P,D) represents all partial remainder-divisor pairs with $P \le$ partial remainder $\le P+2^{-\varepsilon_P}$ $D \le$ divisor $\le D+2^{-\varepsilon_D}$
- Selected q must be legitimate for all pairs in range
- Example: Point A
- Divisor=D_2 - select $q=j+1$; divisor=$D_2+2^{-\varepsilon_D}$ - select $q=j$
- Conclusion: do not select $q=j+1$ for point A or any other point in overlap region whose horizontal distance from the line $(-k+j+1)D \le 2^{-\varepsilon_D}$

Example:

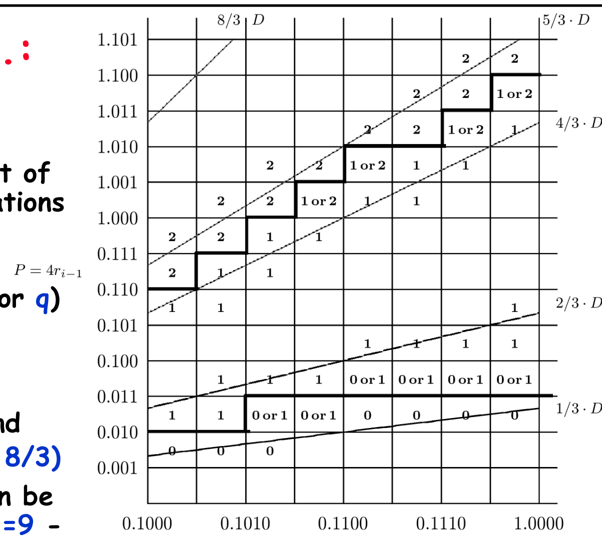
P-D Plot for $\beta=4, \alpha=2, D\in[0.5,1)$



- Overlapping region for $q=1, q=2$ - between $P=(k+\alpha-1)D=5/3\ D$ and $P=(-k+\alpha)D=4/3\ D$
- Single horizontal line?
  * $(k+1)D_{min}=5/6 < (-k+2)D_{max}=4/3$ - no single line
- Smallest horizontal and vertical distances:
  * $\Delta X_{min}=D_{min}\cdot 5/3\cdot 3/20=1/8=2^{-3}$ $\Rightarrow \varepsilon_D \ge 3$
  * $\Delta Y_{min}=D_{min}\cdot 1/3=1/6$ $\Rightarrow \varepsilon_P \ge 3$
- For pair (0.110,.100) - no q legitimate for all points in corresponding rectangle - resolved by $\varepsilon_D=4$

Example Cont.:

$\varepsilon_p=3$; $\varepsilon_D=4$



- Heavy lines - one out of many possible separations
- Designer can select solution to minimize PLA (look-up table for q)
- PLA has $N_D+N_P=4+6$ inputs - 3 more bits needed for integer part of remainder and its sign $(-8/3 \geq P \geq 8/3)$
- Number of inputs can be reduced to $N_P+N_D-1=9$ - most significant bit of D is always 1 - can be omitted
- $2/3 \cdot 1/2 \leq 1/3 \Rightarrow$ single line possible between q=1,q=0 - requires high-precision comparison of partial remainder - divisor interval partitioned into two subintervals

Example

- $X=(0.00111111)_2=63/256$; $D=(0.1001)_2=9/16$
- Comparison constants - 1/4 (0.010) , 7/8 (0.111)

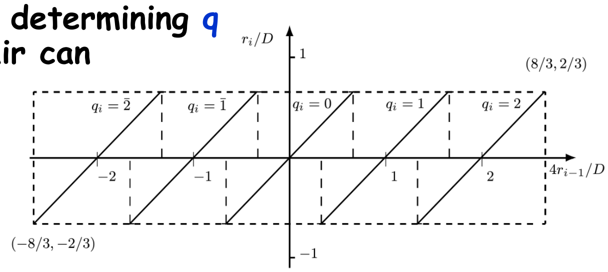
$r_0 = X$		0	.0	0	1	1	1	1	1	1	
$4r_0$	0	0	.1	1	1	1	1	1			$\geq 7/8$ set $q_1 = 2$
Add $-2D$	1	0	.1	1	1	0					
r_1	1	1	.1	1	0	1	1	1			
$4r_1$	1	1	.0	1	1	1					$< -1/4$ set $q_2 = \bar{1}$
Add D	0	0	.1	0	0	1					
r_2	0	0	.0	0	0	0					zero final remainder

- Resulting quotient:

$Q=0.2\bar{1}_4=0.100\bar{1}_2=0.0111_2=7/16$

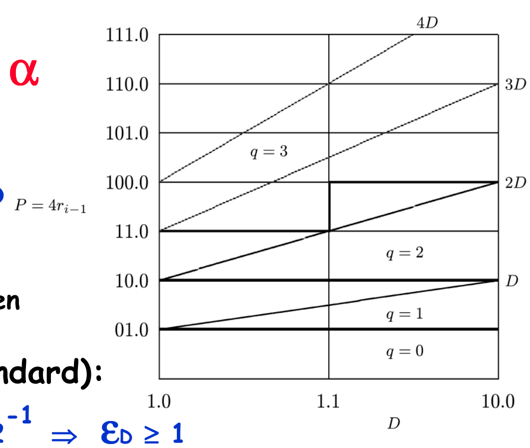
Numerical Calculation of Look Up Table

- **Example-** start with initial guess $\varepsilon_p=\varepsilon_D=3$ - attempt to calculate q for D=0.100 and P=0.110 (worst case)
- Divisor truncated - consider values from 0.100 to 0.101 ; partial remainder from 0.110 to 0.111
- P/D between 0.110/0.101=1.2 (q=1) and 0.111/0.100=1.75 (q=2)
- Insufficient precision - increase number of bits of either divisor or partial remainder - try again
- Numerical search determining q for each (P,D) pair can be programmed



Example - Lower Precision of Higher α

- $\beta=4;\ \alpha=3;\ k=\alpha/(\beta-1)=1$
- Region for q=2 - between P=(k+q)D=3D ; P=(-k+q)D=D
- Region for q=3 - between P=4D ; P=2D
- Overlapping region - between P=3D and P=2D



- For $D\in[1,2)$ (IEEE standard):
  - * $\Delta X\min=D\min\cdot 3\cdot 1/6=3/6=2^{-1} \Rightarrow \varepsilon_D \geq 1$
  - * $\Delta Y\min=D\min\cdot 1=1 \Rightarrow \varepsilon_p \geq 0$
- Based on diagram - $\varepsilon_D=1$; $\varepsilon_p=0$ - $N_D=2$; $N_P=4$ instead of $N_D=4$; $N_P=6$ for α=2
- Simpler quotient selection logic - costly multiple 3D

Example

- X=(01.0101)$_2$=21/16 ; D=(01.1110)$_2$=15/8
- **Partial remainder comparison constants - 1,2,4**

$r_0 = X$				0	1	.0	1	0	1	
$4r_0$		0	1	0	1	.0	1	0		≥ 4.0 set $q_1 = 3$
Add $-3D$	+	1	0	1	0	.0	1	1		
r_1		1	1	1	1	.1	0	1		
$4r_1$		1	1	1	0	.1	0	0		≥ -2.0 set $q_2 = \bar{1}$
Add D	+	0	0	0	1	.1	1	1		
r_2		0	0	0	0	.0	1	1		final remainder $= 3/8 \cdot 2^{-4}$

- **Quotient:** Q=$(0.3\bar{1})_4$=$(0.110\bar{1})_2$ = 11/16
- **Verification:**
  QD+R = 11/16·15/8+3/128 = 168/128 = 21/16=X