CSE4210
Architecture and Hardware for DSP

Lecture 1

Introduction &

Number systems

`

Administrative Stuff

- CSE4210 Architecture and Hardware for DSP
- Text: VLSI Digital Signal Processing Systems: Design and Implementation. K. Parhi. Wiley Interscience
- Posted articles

Administrative Stuff

- Office hours: Monday 1-2pm TR 3-4pm
- Room 2026 CSEB x40607
- HW 0%
- Quizes 10%
- Midterm 25%
- Projects 25%
- Final 40%

YORK UNIVERSITY CSE4210

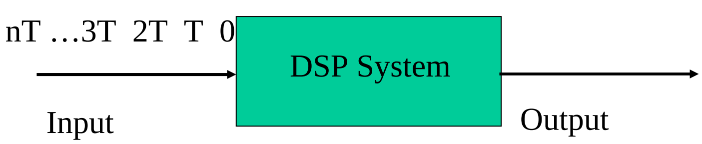
Topics

- Number systems
- Fast arithmetic
- Algorithm representation
- Transformation (retiming, unfolding, folding)
- Systolic arrays and mapping algorithms into hardware
- Low power design

Introduction

- Introduction to DSP algorithms
- Non-terminating programs in real time.
- Speed depends on applications (audio, video, 2-D, 3-D, …)
- Need to design families of architectures for specified algorithm complexity and speed constraints

Typical DSP Programs



3-Dimensional optimization: Area, Speed, Power)

Usually, speed is a requirement, area-power tradeoff

$P = C V^2 F$

Examples

- FIR filter, x(n) is the input, y(n) output

$$y(n)=\sum_{j=0}^{J-1}h(j)x(n-j)$$

- IIR filter

$$y(n)=\sum_{i=1}^{P}a(i)y(n-i)+\sum_{k=0}^{Q}b(k)x(n-k)$$

Examples

- Convolution $y(n)=\sum_{i=0}^{M-1}x(i)h(n-i)=\sum_{j=0}^{N-1}h(j)x(n-j)$

```
For n=1 to M+N-2,
    y(n)=0,
     For i=0:M-1,
     y(n)=y(n)+x(i)*h(n-i)
     end
  end
```

MAC operation (arrow pointing to `y(n)=y(n)+x(i)*h(n-i)`)

More Complex Examples Motion Estimation

- Image (frame) is divided into macroblocks
- Each macroblock is compared to a macroblock in the reference frame using some error measure.
- The search is conducted over a predetermined search area.
- A vector denoting the displacement of the (motion) is sent.

More Complex Examples Motion Estimation

- Many measures of errors could be used.
- The displaced block difference s(m,n) using MAD (Mean Absolute Difference) is defined as

$$s(m,n) = \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \left| x(i,j) - y(i+m, j+n) \right|$$

m,n are in is the search area, N is the macroblock size.
The one with the minimum error is chosen

More Complex Examples Vector Quantization

- Used in compression
- A group of samples (vector) are quantized together
- For example consider k pixels, with W bits.
- That vector is compared to a group of N *codewords*, choose the one with the min. distortion.
- We transmit the index of that codeword

More Complex Examples Vector Quantization

- Compression ratio = KW/$\log_2$N
- Euclidean distance is used as a measure of distortion.

$$d(\mathbf{x},\mathbf{c_j}) = \left\|\mathbf{x}-\mathbf{c_j}\right\|^2 = \sum_{i=0}^{k-1}(x_i - c_{ji})^2$$

$$= \left\|x\right\|^2 - 2(\mathbf{x}\cdot\mathbf{c_j} + e_j), \quad e_j = -\frac{1}{2}\left\|c_j\right\|^2 = -\frac{1}{2}\sum_{i=0}^{k-1}c_{ji}^2$$

Discrete Cosine Transform

- The 1-D DCT is defined as

$$X(K) = e(k)\sum_{n=0}^{N-1} x(n)\cos\left[\frac{(2n+1)k\pi}{2N}\right] \quad , \quad k = 0,1,2,\cdots,N-1$$

$$e(k) = \begin{cases} 1/\sqrt{2} & \text{if } k = 0 \\ 1 & \text{otherwise} \end{cases}$$

More Complex Examples

- Viterbi Decoding
- FFT
- Wavelets and Filter banks
- See the book for details

Requirements

- Consider block matching algorithm, the computational requirement is as follows
- $3(2p+1)^2NMF$
- 3*(2*7+1)*288*352*30=2GOP
- Much higher for higher resolution and bigger frames
- How to achieve these requirements?

hardware

- Microprocessors
- Microprocessors with DSP extension
- DSP
- FPGA
- ASIC

Number System

- Numbers and their representation
- Binary numbers
- Negative numbers
- Unconventional numbers

Number Systems

- Positional Weight Systems
  - Integer X is represented as

$$X = (X_{n-1}, X_{n-2}, \cdots, X_1, X_0)$$

  - Interpretation $X = \sum_{i=0}^{n-1} X_i W_i$
  - Fixed Radix Number System Special Case

$$W_i = R^i$$

Binary Numbers

- An ordered sequence

$(x_{n-1}, x_{n-2}, \cdots, x_1, x_0)$ $x_n \subset \{0,1\}$

- The value of the number is
- $$X = x_{n-1}2^{n-1} + x_{n-2}2^{n-2} + \cdots + x_1 2 + x_0 = \sum_{i=0}^{n-1} x_i 2^i$$
- The range [X_{min}, X_{max}] is the range of the numbers to be represented, in the previous case [0,2^n-1]

Binary numbers

- The previous representation is non-redundant and weighted (w_i)
- The n-digit number can be partitioned into a fraction part (n-k bits) and an integral part(k bits)

$$(\underbrace{x_{k-1}x_{k-2}\cdots x_1 x_0}_{integral\ part} \ . \ \underbrace{x_{-1}x_{-2}\cdots x_{-m}}_{fractional\ part})_r$$

$$X = x_{k-1}r^{k-1} + x_{k-2}r^{k-2} + \cdots + x_1 r + x_0 + x_{-1}r^{-1} + \cdots + x_{-m}r^{-m}$$

Binary numbers

- Given the length of the operand, n, the weight r^{-m} of the least significant digit indicates the position of the radix point.
- Unit in Last Position ulp=r^{-m}
- Simplifies the discussion and there is no need to partition the number into fractional and integral parts.

Conversion

- Convert 36.4375 into binary

36/2

Division by 2

Quotient	Remainder
18	0
9	0
4	1
2	0
1	0
0	1

Multiplication by 2

Integer	Fraction
0	.875
1	.75
1	0.5
1	0.0

100100.0111

Negative Numbers

- Signed magnitude
- Complement
  - Diminished radix complement (1's complement for binary)
  - Radix complement (2's complement in binary)

Signed magnitude

- The n^{th} bit (digit) is the sign
- n-1 digits for magnitude (k-1 integral and m fractional).
- Largest value 011…11 = $X_{max}=r^{k-1}$-ulp
- Smallest negative value $-(r^{k-1}$-ulp) 11..1
- Two representation for zero

Signed magnitude

- Operations may be more complicated than using complement.
- For example, adding 2 numbers, a positive number X, and a –ve number Y, the result depends on if X>Y or not
- If X > Y the result is X+(-Y)
- If Y>X switch the 2 numbers, subtract, attach minus sign –(Y-X)

Complement representation

- Positive numbers are represented just as signed-magnitude
- Negative numbers are represented as R-number, where R is a constant
- Note that –(-y)=R-(R-Y))=Y
- The choice of R must satisfy 2 conditions
  - Calculating the complement is easy
  - Simplifying or eliminating correction

Complement representation

- For radix complement $R=r^k$
- For the diminished radix complement $R=r^k$ - ulp

YORK UNIVERSITY CSE4210

r=2, k=n=4, m=0, ulp=2^0 =1

Sequence	Two's complement	One's complement	Signed-magnitude
0111	7	7	7
0110	6	6	6
0101	5	5	5
0100	4	4	4
0011	3	3	3
0010	2	2	2
0001	1	1	1
0000	0	0	0
1111	−1	−0	−7
1110	−2	−1	−6
1101	−3	−2	−5
1100	−4	−3	−4
1011	−5	−4	−3
1010	−6	−5	−2
1001	−7	−6	−1
1000	−8	−7	−0

YORK UNIVERSITY CSE4210

Example

2’s complement

6= 0110 -6 = 1010 4=0100 -4 = 1100

6-4	4-6	6+4	-4-6
0110	0100	0110	1100
1100	1010	0100	1010
1 0010	1110	1010	1 0110
+2	-2	-6	+6

Carry in = carry out
Ignore carry out

Carry in $\neq$ carry out
Overflow

YORK UNIVERSITY CSE4210

Example

1’s complement

6= 0110 -6 = 1001 4=0100 -4 = 1011

6-4	4-6	6+4	-4-6
0110	0100	0110	1011
1011	1001	0100	1001
1 0001	1101	1010	1 0100
+2	-2	-5	+4

Carry in = carry out
Add to LSB

Carry in $\neq$ carry out
Overflow

Arithmetic Shift

- Consider the number $\{x_{n-2}, x_{n-3}, .,.,.x_0\}$
- Finite extension of signed magnitude is
- 2's complement $...0,0\{x_{n-2}, x_{n-3}, .,.,.x_0\}0,0,...$
- 1's complement $...x_{n-1}, x_{n-1}\{x_{n-2}, x_{n-3}, .,.,.x_0\}0,0,...$

$$...x_{n-1}, x_{n-1}\{x_{n-2}, x_{n-3}, .,.,.x_0\}x_{n-1}, x_{n-1}$$

Arithmetic Shift

2's Comp		1's Comp	
-6	1010	-6	1001
-12	10100	-12	10011
-24	101000	-24	100111
-6	1010	-6	1000
-3	1101	-3	1100
-2	1110	-1	1110
-1	1111	-0	1111

Other Number System

- Negative radix number system
- A general class of of fixed-radix number system
- Signed-digit number system
- Residue number system

Canonical Systems

- Canonical if $D_i = \{0,1,2,\ldots,R_{i-1}\}$ *with* $D_i = R_i$
  - Binary {0,1}
  - Decimal {0,1,…,9}
  - Hex {0,1,2,…,E,F}
- Range of values $0 \le x \le r^n - 1$ for n-digit radix r number
- No-ncanonical Binary {-1,0,1}
  - {1,1,0,1},{1,1,1,-1} both represent 13

Negative radix Number System

- The radix could be negative, r=-β, β is a positive number.
- Digit set 0,1,…, β-1
- Value of $(x_{n-1}, x_{n-2}, ..., x_0)$
- $$X = \sum_{i=0}^{n-1} x_i (-\beta)^i$$

287_{-10}=200-80+7= 127

Range = $[090,909]_{-10}$, or $[-9,909]_{10}$

1010_{-2} = -8-2= -10

Range=$[1010,0101]_{-2}$ = $[-10,5]_{10}$

Negative radix Number System

- Algorithms do exist for basic operations.
- Not better than 2's complement systems

General Class of Fixed Radix Number System

- Characterized by (n,β,Λ), β is a positive radix, digit set 0,1,… β-1, and a vector of length n $\Lambda=(\lambda_{n-1}, \lambda_{n-2}, \dots \lambda_0)$ $\lambda_i = \{-1,1\}$

$$X = \sum_{i=0}^{n-1} \lambda_i x_i \beta^i$$

- 2's Complement Λ={-1,1,1,….1}

General Class of Fixed Radix Number System

$$P = \{p_{n-1}, p_{n-2}, \dots, p_0\}$$

Max. positive number

$$p_i = \begin{cases} \beta - 1 & \text{if } \lambda_i = +1 \\ 0 & \text{otherwise} \end{cases} \quad = \frac{1}{2}(\lambda_i + 1)(\beta - 1)$$

$$P = \sum_{i=0}^{n-1} 1/2(\lambda_i + 1)(\beta - 1)\beta^i = 1/2\left[\sum_{i=0}^{n-1} \lambda_i (\beta - 1)\beta^i + \sum_{i=0}^{n-1} (\beta - 1)\beta^i\right]$$

$$= 1/2\left[Q + (\beta^n - 1)\right]$$

Where Q is the value of the tuple $(\beta - 1, \beta - 1, \cdots, \beta - 1)$

Find the smallest representable number

$\Lambda = (-1, 1, -1, 1)$

$Q = 9999 = -8181$

$P = \frac{1}{2}[-8181 + 9999] = 909$

Signed Digit Number System

- The digits could be positive or negative
- Redundant (more than one representation for the same number)
- For a radix β, $x_i \in$ {β-1, β-**2,…,1,0,1… β-1**}
- To reduce redundancy,

$$X_i = \left\{ \bar{a}, \overline{a-1}, \ldots, \bar{1}, 0, 1, \ldots a \right\}, \text{ where } \left\lceil \frac{r-1}{2} \right\rceil \le a \le r-1$$

YORK UNIVERSITY CSE4210

Signed Digit Number System

- Example: β=10,a=6,

Breaking the Carry Chain

- Add two numbers X,Y

$$w_i = x_i + y_i - rt_{i+1}$$

$$t_{i+1} = \begin{cases} 1 & \text{if } (x_i + y_i) \geq a \\ \bar{1} & \text{if } (x_i + y_i) \leq \bar{a} \\ 0 & \text{if } |x_i + y_i| < a \end{cases}$$

$$s_i = w_i + t_i \qquad , t_0 = 0$$

Example, a=6,r=10
1634+3366, or
12$\underline{5}$4+36$\underline{2}$1

3645
1456
0111 t
40$\bar{1}$1 w
5101

Breaking the Carry Chain

Breaking the carry chain

- For no carry $|s_i|=|w_i+t_i| \le a$, $\rightarrow |w_i| \le a-1$

$$\text{Case 1} \quad x_i + y_i = 2a \quad \text{(upper bound)}$$

$$w_i = 2a - r(1) = 2a - r \quad \Rightarrow \quad a \le r - 1$$

$$\text{Case 2} \quad x_i + y_i = a \quad \text{(lower bound)}$$

$$w_i = (a) - r(1) = a - r$$

$$|w_i| = r - a$$

$$r - a \le a - 1 \quad \Rightarrow \quad \left\lceil \frac{r+1}{2} \right\rceil \le a$$

$$\left\lceil \frac{r+1}{2} \right\rceil \le a \le r - 1$$

$$2 \le a \le 1$$

$$\left\lceil \frac{2+1}{2} \right\rceil \le a \le 2 - 1$$