

Circuitos Digitais

Aula 3

Representação da informação digital

- Escalar
 - $x_i \in \{0, 1\}$ representa 1 bit de informação
- Tuplas
 - $[x_1, x_2, x_3, \dots, x_n]$, conjunto de variáveis binárias.
 - onde $x_i \in \{0, 1\}$
 - Exemplos: $r = 2$
 - $[0, 0], [0, 1], [1, 0], [1, 1]$
- Vetores
 - São r -tuplas que representam informações digitais.
 - $X = [x_1, x_2, x_3, \dots, x_n]$

Onde X é uma variável Vetor

Notação vetorial

- Vetor
 - Os vetores são representados por *letras maiúsculas A, B, MBR*
 - As variáveis escalares que representam cada elemento do vetor são representadas por *letras minúsculas a, b, mbr*
- Dimensão do vetor
 - A dimensão de um vetor é expressa pelo número de suas variáveis escalares.

Representação da Informação

- Informação
 - Transmitida por sinais
 - Armazenada em registradores
- Transmitidas por sinais

• Cada um destes sinais é representado pelos valores atribuídos as variáveis binárias que compõem o vetor.
 • O significado associado a tais sinais depende da tarefa a ser realizada e da decodificação adotada.

Representação da Informação

- Armazenamento em registradores
 - A unidade básica de armazenamento de informação de vetores são os registradores, os quais são compostos de células elementares de memória

O estado de cada célula 0, 1 corresponde a menor unidade de informação digital possível, 1 bit.

Registrador de vetor

- A informação guardada em um registrador de r bits pode ser considerada como um vetor de r bits.
 - $Y = [y_1, y_2, \dots, y_{n-1}, y_n]$
 - As variáveis $y_1, y_2, \dots, y_{n-1}, y_n$ representam o conteúdo de cada célula do registrador e são chamadas variáveis de estado.
 - Exemplo:
 - Se $r=2$ então existem 4 estados distintos, a saber:
 - $[0, 0], [0, 1], [1, 0], [1, 1]$
 - As células podem ser numeradas da esquerda para a direita $Y = [y_1, y_2, y_{n-1}, y_n]$, ou da direita para a esquerda $Y = [y_n, y_{n-1}, \dots, y_2, y_1]$
 - As células podem também ser numeradas de 0 a $r-1$, de 1 a r , ou ainda de $r-1$ a 0 e de r a 1.
 - $A = [a_0, a_1, \dots, a_{r-1}]$ ou $A = [a_1, a_2, \dots, a_r]$
 - $A = [a_{r-1}, a_{r-2}, \dots, a_0]$ ou $A = [a_r, a_{r-1}, \dots, a_1]$

7

Informação simbólica

- Seja S um conjunto de k quantidades distintas, então podemos estabelecer uma relação 1-a-1 entre um valor distinto da r-tupla $x_1, x_2, \dots, x_{r-1}, x_r$ com cada elemento do conjunto desde que $k = 2^r$.
 - Esta correspondência não é única!
 - Existem $(2^r - 1)(2^r - 2) \dots (2^r - k + 1) = (2^r)! / (2^r - k)!$ distintos caminhos nos quais esta correspondência pode existir
 - Exemplo 1: $k = 4$ e $r = 2$
 - a 00 00 00
 - b 01 01 10
 - c 10 11 11
 - d 11 10 01
 - Exemplo 2: O conjunto S consiste de 3 comandos +, *, - os quais podem ser codificados usando
 - 2 bits + [0,0], - [0,1], * [1,1]
 - 3 bits + [1,0,0], - [0,1,0], * [0,0,1]

8

Representação da informação - Lógica

- Lógica
 - Informação lógica é expressa em termos de códigos que indicam se uma declaração é verdadeira ou falsa.
 - Exemplo: A declaração Maria usa blusa azul
 - $x = 1$ se a declaração é verdadeira
 - $x = 0$ se a declaração é falsa
 - x é um escalar
 - Em linguagem de alto-nível dizemos, por exemplo:
 - FORTRAN $x := \text{true}$ ou $x := \text{false}$
 - PASCAL $x := \text{true}$ ou $x := \text{false}$
 - Neste caso, como pode X estar representado internamente?
 - 0,0,0,0,0.....,0 - false
 - 1,1,1,1,1.....,1 - verdadeiro
 - Obs: Essa codificação é ineficiente pois das 2^r possibilidades de codificação, apenas 2 foram usadas. Se o número de variáveis lógicas for pequeno isso pode não ser problema.

9

Representação da informação - Lógica

- Número de células de um registrador
 - Em princípio o número de bits de um registrador é arbitrário, mas é usual agrupá-los em número padronizado que recebem nome especiais:
 - Nibble (4 bits) - representam algarismos decimais
 - Byte (8 bits) - representam caracteres
 - Word (palavra) - tamanho de registradores padrão de um determinado computador
- Tamanho de palavras

Microprocessadores	Microcomputadores
8080, 8085, Z80, 6800 - 8 bits(byte)	PDP-11 - 16 bits
8086, Z8000 - 16 bits	VAX 11/780 - 32 bits
Main Frames	Microprocessadores RISC
IBM 3033 - 32 bits	SPARC - 32 bits
CDC 6600 - 60 bits	RS6000 - 32 bits
	Alpha AXP - 64 bits

10

Representação da informação - Caracteres

- Todo caracter é devidamente codificado em binário antes de ser processado por um computador digital.
 - Exemplo:
 - Um computador típico permite a codificação dos caracteres:
 - 26 letras minúsculas - 26 letras maiúsculas
 - 10 algarismos decimais - 33 símbolos especiais (+, -, =,
 - 33 caracteres de controle (line feed, deleteem carriage return,...)
 - Ou seja $128 = 2^7$ códigos digitais distintos, que podem ser representados por 7 bits.
 - Existem 128! maneiras diferentes de se codificar tais símbolos, porém alguns padrões são mundialmente usados:
 - ASCII (American Standard Code for Information Interchange) - 7bits
 - EBCDIC (Extended Binary Code Decimal Interchange Code) - 8 bits
 - Exemplo:

Símbolo	ASCII	EBCDIC
A	1000001	11000001
X	1011000	11100111
Y	1011001	11101000
Z	0110010	11110010

11

Representação da informação - Números

- Os números podem ser representados em diferentes sistemas ou bases numéricas, tais como:
 - Exemplo: Base 10 $\rightarrow 305.2$ $N_{10} = 3(10^2) + 0(10) + 5(10^0) + 2(10^{-1})$
 - De maneira genérica podemos representar um número por:

$$N_r = a_{n-1}(r)^{n-1} + \dots + a_0(r)^0 + a_{-1}(r)^{-1} + \dots + a_{-m}(r)^{-m}$$

$N_r = \underbrace{a_{n-1} a_{n-2} \dots a_0}_{\text{Inteiro}} \underbrace{a_{-1} a_{-2} \dots a_{-m}}_{\text{fração}}$
 - Sistemas numéricos usuais:

Base	Nome do sistema	Algarismos
2	Binário	0, 1
3	Ternário	0, 1, 2
8	Octal	0, 1, 2, 3, 4, 5, 6, 7
10	Decimal	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0
16	Hexadecimal	0, 1, 2, ..., 9, A, B, C, D, E, F

12

Codificação de informação numérica

- Localização do ponto binário
 - Se um registrador contém r células, o número binário deve ter no máximo r dígitos. Ou seja, qualquer número inteiro entre 0 e 2^{r-1} pode ser representado.
 - Com números fracionários, não é usual codificar a posição do ponto. Será suposto que o projetista do sistema arbitre a posição do mesmo. Assim, os números 0,101 ou 1,01 ou 10,1 ou 101 poderiam ser todos codificados como 101
- Números negativos
 - Números negativos são representados por um bit de sinal que geralmente é representado pelo bit mais à esquerda do número.
 - Exemplo: para um registrador de r bits teríamos:

sinal	magnitude
1 bit	(r-1) bits
 - Por convenção:
 - 0 - número positivo
 - 1 - número negativo

Codificação de informação numérica

- Exemplo de codificação de números binários negativos: (r=3)

binário	decimal	binário	decimal
000	+0	100	-0
001	+1	101	-1
010	+2	110	-2
011	+3	111	-3
- Codificação de números decimais
 - Embora trabalhem com informações binárias nós vivemos num mundo decimal, e as vezes é desejável manter o caracter decimal de um número mesmo depois de decodificado.
 - De quantas maneiras podemos codificar n dígitos decimais?
 - Precisaremos de no mínimo 4 bits = m
 - Caracteres n = 10, 0 a 9.
 - $n \leq 2^m$
 - Número de possíveis combinações, $NC = 2^m! / (2^m - n)! = 2,9 \times 10^{10}$

Codificação de informação numérica

- Escolha do código
 - Facilidade para operações aritméticas
 - Redução do espaço de armazenamento
 - Redução de hardware
 - Deteção e correção de erro
 - Simplicidade
- Código com peso
 - A cada bit da 4-tupla $[x_1, x_2, x_3, x_4]$ é associado um peso w_1, w_2, w_3, w_4 , tal que $N_{10} = \sum w_i x_i$
 - Exemplo: Representar algoritmos decimais em BCD(Binary Code Decimal)

Decimal	BCD	Decimal	BCD
0	0000	5	0101
1	0001	6	0110
2	0010	7	0111
3	0011	8	1000
4	0100	9	1001
 - Exemplo: $N_{10} = 675 \Rightarrow 0110\ 0111\ 0101$

6	7	5
---	---	---

Codificação de informação numérica

- Exemplo 2

pesos	2	4	2	1
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
- Exemplo 3

pesos	7	4	-2	-1
0	0	0	0	0
1	0	1	1	1
2	0	1	1	0
3	0	1	0	1
4	0	1	0	0
5	1	0	1	0
6	1	0	0	1
7	1	0	0	0
8	1	1	1	1
9	1	1	1	0
- Exemplo 4

excesso 3	
0	0 0 1 1
1	0 1 0 0
2	0 1 0 1
3	0 1 1 0
4	0 1 1 1
5	1 0 0 0
6	1 0 0 1
7	1 0 1 0
8	1 0 1 1
9	1 1 0 0
- Exemplo 5

BCD o paridade par	
0	0 0 0 0 0
1	0 0 0 1 1
2	0 0 1 0 1
3	0 0 1 1 0
4	0 1 0 0 1
5	0 1 0 1 0
6	0 1 1 0 1
7	0 1 1 1 0
8	1 0 0 0 1
9	1 0 0 1 0

Codificação de informação numérica

- Código Gray
 - É um código que garante a mudança de um bit apenas entre duas unidades consecutivas

Duas formas diferentes de codificar dados de um mesmo sensor.
 - Problema: na mudança do valor 5 para 6 teríamos:
 - No código BCD dois dígitos seriam mudados ao mesmo tempo, o que poderia provocar uma leitura errônea no período de transição $5 \Rightarrow 0101$ para $6 \Rightarrow 0110$
 - Usando código Gray apenas um bit é trocado a cada transição $5 \Rightarrow 0111$ para $6 \Rightarrow 0101$

(Conversão AD/ DA)

Conversão Analógico->Digital

Amplitude

Time Quantized Samples

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

- Encode

- Digital Signal

19

Processo de conversão AD

- A conversão AD é o processo no qual um sinal analógico (mundo contínuo) é transformado para um sinal discreto no tempo (representação simbólica).

Amplitude

Time Quantized Samples

Encode

Digital Signal

- A conversão se processa através do que chamamos amostragem do sinal de entrada (limitado em banda), convertendo o sinal analógico em um sinal discreto no tempo.
- A amplitude de cada sinal amostrado é quantizado dentro de um dos 2^b níveis possíveis, onde b é o número de bits usados para representar uma amostra no conversor AD (ADC).
- Os níveis de amplitude discretos são representados ou codificados em palavras binárias distintas, cada uma de tamanho de b bits.

20

Processo de conversão AD

Filtro passa baixa

Amostragem e armazenamento

Quantizador

Codificador

Entrada analógica $X(t)$

f_s

2^b

Código digital $X(n)$

Circuito lógico

O sinal é contínuo em tempo e amplitude

O sinal é contínuo em amplitude, mas definido apenas em pontos discretos de tempo. Assim, o sinal é zero, exceto no tempo $t = nT$ (instantes de amostragem)

O sinal digital $x(n)$ ($n=0,1,\dots,n$). Este sinal existe apenas em pontos discretos no tempo e em cada ponto do tempo pode haver um dos 2^b valores (valor discreto em tempo discreto)

Tecnologia A/D

- Alta velocidade de amostragem
 - 1500MSps @ 8bit
 - 210MSps @ 12bit (AD9430)
 - Alta taxa de amostragem são alcançadas em osciloscópios digitais.
 - Grande quantidade de dados é gerada – além da capacidade de computação para amostragens contínuas.
- Aplicação: captura de transientes

Technologia A/D

- Conversores A/D de média velocidade
 - 14bit @ 105MSps (AD6645)
 - 16bit @ 20MSps (ET2673)
 - 23bit @ 20MSps (HP)
- Aplicações: comunicação, uso militar, *software radio*
- Baixa velocidade
 - 16+bit @ 200kSps
 - 24bit @ 96kSps
- Aplicações: áudio, brinquedos

23

Exemplo

0000
0001
0010
0011
0010
0001
0000

Conversão D/A

Analogico

Digital

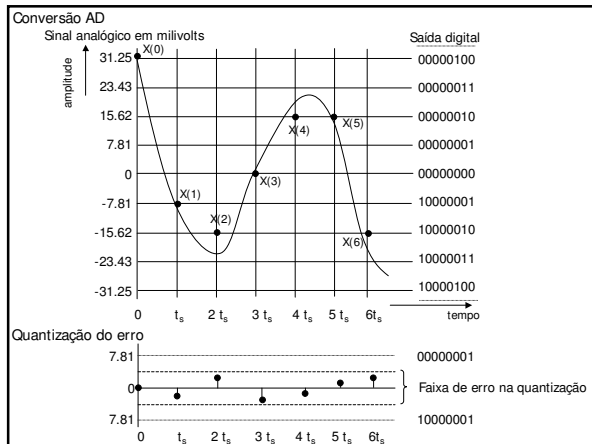
Processamento digital

Conversão A/D

24

Quantificação de erros na conversão A/D

- Conversores AD, em geral, geram palavras binárias de tamanho fixo, 8, 16, 24 bits,
- Conversores comerciais são categorizados pelo tamanho das palavras binárias que podem gerar, 8, 16, 24 bits,
- Uma entrada típica analógica de entrada de um conversor AD é de $-1 a + 1$ volt.
- O bit menos significativo da palavra de conversão (*Least Significant bit - lsb*) quantifica o menor valor a ser codificado pelo conversor.
 - Exemplo:
 - Conversor de 8 bits, entrada analógica de $-1 a + 1$ V.
 - $lsb(\text{valor}) = \text{faixa de tensão na entrada} = \frac{2V}{2^8} = 7.81mV$
- Isto significa dizer que podemos representar tensões contínuas (analógicas) perfeitamente, desde que elas sejam múltiplas de 7.81 mV.



Conversão Analógica / Digital

- Usando apenas 8 bits não é possível representar todos os valores no contínuo \pm o que acarreta erros na informação armazenada.
- O maior erro ocorre quando a entrada está a mesma distância de 2 valores adjacentes.
 - Exemplo:
 - No meio entre 7.81 e 15.62
- Erro de quantificação
 - Este erro é introduzido pela aproximação digital de um sinal contínuo.
 - Como reduzir este erro?
 - Aumentando o número de bits usados para armazenar o sinal digital.

Conversão Analógica / Digital

Exemplo: Range $\pm 0,75$
X = 3 bits

Conversão Analógica / Digital

- Amostragem
 - Em aplicações de conversores A/D pode ser interessante ler o valor do sinal a cada t segundos (tempo de amostragem). A saída será uma série de valores referentes a amostragem do sinal de entrada.
 - Exemplo:
 - $s(t) = \text{sen } \omega t$, onde $\omega = 12 \text{ rad/seg}$
 - Amostre 11 vezes no intervalo de 0 a 20 segundos.
 - Período de uma amostragem será de 2 segundos, $t = nt$, $n = 0, 1, \dots, 10$.

n	nt	sen 12nt	X(nt)
0	0	0,000	00000
1	2	0,500	00100
2	4	0,866	00111
3	6	1,000	01000
4	8	0,866	00111
5	10	0,500	00100
6	12	0,000	00000
7	14	-0,500	10100
8	16	-0,866	10111
9	18	-1,000	11000
10	20	-0,866	10111

Conversão A/D

Funcionamento:
Enquanto V_s é maior que V_d , a saída do comparador é alta e a porta AND (controle de clock) é habilitada, permitindo passagem de pulsos para o contador que conta em ordem crescente até atingir o valor correspondente analógico ou seu valor máximo.

Quando V_d é maior que V_s , a saída do comparador muda para nível baixo (nível lógico '0'), e a porta de controle do clock é desabilitada, impedindo que o contador continue contando. Isto quer dizer que a saída possui a conversão digital correspondente a V_s .

Conversão Digital -> Analógico

Conversão D/A

31

- O conversor DA (DAC) aceita como entrada dados digitais e produz uma saída analógica, a qual é relacionada com o código digital de entrada.

100 = 4	7
101 = 5	6
110 = 6	5
111 = 7	4
110 = 6	3
101 = 5	2
100 = 4	1
010 = 2	0
000 = 0	0
010 = 2	1
100 = 4	2
101 = 5	3

- Um registrado é usada para armazenar a entrada do DAC e assegurar que sua saída fique estável até que o conversor seja alimentado por uma outra entrada digital. O registrador pode ser externo ou fazer parte do DAC.
- Cada palavra digital (número de bits em paralelo) são convertidas no tempo.
- Filtros podem ser usados para suavizar ou restaurar o sinal analógico em sua saída.

Conversão D/A

32

A saída do DAC tem uma forma de escada desde que cada impulso é seguro (hold) por um tempo T(s)

Conversão Digital / Analógica

33

Registrador de n bits

Sinal Digital

- Representação da informação binária

b_u	b_{u-1}	b_{u-2}	b_1	b_0	parte inteira	b_1	b_2	b_v	parte fracionária
-------	-----------	-----------	-------	-------	-------	---------------	-------	-------	-------	-------	-------------------

Exemplo: $r=5$, desde que $u+n+v=1+2+2=r=5$

número digital	saída analógica	número digital	saída analógica
000,00	0,00	100,00	-0,00
000,01	0,25	100,01	-0,25
000,10	0,50	100,10	-0,50
.....
011,11	3,75	111,11	-3,75

← bit de sinal

Conversão Digital / Analógica

34

- Resolução**
 - Resolução de um conversor Digital/Analógica é a diferença entre dois valores consecutivos da saída do conversor D/A.
 - Exemplo:
 - Considerando o exemplo em evidência a resolução do conversor seria de 0,25
- Range ou faixa de atuação**
 - Distância entre o valor mais positivo e o valor mais negativo.
 - Exemplo:
 - Considerando conversor do exemplo anterior teríamos um faixa de atuação de $3,75 - (-3,75) = 7,50$ de faixa de atuação

Conversão Digital/ Analógica

35

Ganho do AMP-OP = $(-3R/2R)$
 $V_o = V_{in} \cdot (-3R/2R)$

V_o (Saída Analógica)

Exemplo:
 Considerando $V_R = +5V$
 Para $X = [1000]$ (entrada)
 $V_{in} = -V_R/3$
 Como $V_o = V_{in} \cdot (-3R/2R)$
 $V_o = (-V_R/3) \cdot (-3R/2R) = V_R/2 \Rightarrow$
 $V_o = 2,5V$

De maneira geral:				
X_3	X_2	X_1	X_0	V_0 (V)
1	0	0	0	$+V_R/2$
0	1	0	0	$+V_R/4$
0	0	1	0	$+V_R/8$
0	0	0	1	$+V_R/16$

Conversão Digital/ Analógica

36

$V_o = V_{in} \cdot (-3R/2R)$

Saída analógica

0,0000
0,3125
0,6250
0,9375
1,2500
1,5625
1,8750
2,1875
2,5000
2,8125
3,1250
3,4375
3,7500
4,0625
4,3750
4,6875

Entrada Digital

Conversão Digital/ Analógica

Valor Digital MSB ← → LSB	Valor analógico V_R (Tensão de referência)	Valor da tensão de saída (V_o)
0000	0	0,0000
0001	$V_R/16$	0,3125
0010	$V_R/8$	0,6250
0011	$V_R/16 + V_R/8$	0,9375
0100	$V_R/4$	1,2500
0101	$V_R/4 + V_R/16$	1,5625
0110	$V_R/4 + V_R/8$	1,8750
0111	$V_R/4 + V_R/8 + V_R/16$	2,1875
1000	$V_R/2$	2,5000
1001	$V_R/2 + V_R/16$	2,8125
1010	$V_R/2 + V_R/8$	3,1250
1011	$V_R/2 + V_R/8 + V_R/16$	3,4375
1100	$V_R/2 + V_R/4$	3,7500
1101	$V_R/2 + V_R/4 + V_R/16$	4,0625
1110	$V_R/2 + V_R/4 + V_R/8$	4,3750
1111	$V_R/2 + V_R/4 + V_R/8 + V_R/16$	4,6875

