materia

Sistemas Dixitais

unidade didáctica 3

Análise e síntese de circuítos lóxicos combinacionais

Diego Rodríguez Martínez

Departamento de Electrónica e Computación

Escola Técnica Superior de Enxeñaría

USC UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

VICERREITORÍA DE ESTUDANTES, CULTURA E FORMACIÓN CONTINUA

titulación

Grao en Enxeñaría Informática

USC
UNIVERSIDADE
DE SANTIAGO
DE COMPOSTELA

unidade **didáctica 3**

Análise e síntese de circuítos lóxicos combinacionais

Diego Rodríguez Martínez
Departamento de Electrónica e Computación
Escola Técnica Superior de Enxeñaría

unidadedidáctica
UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

Copyright © Universidade de Santiago de Compostela, 2012

Deseño
Unidixital

Edita
Vicerreitoría de Estudantes,
Cultura e Formación Continua
da Universidade de Santiago de Compostela
Servizo de Publicacións
da Universidade de Santiago de Compostela

Imprime
Unidixital
Servizo de Edición Dixital da
Universidade de Santiago de Compostela
Dep. Legal: C 1120-2012
ISBN 978-84-9887-903-2

ADVERTENCIA LEGAL: reservados todos os dereitos. Queda prohibida a duplicación, total ou parcial desta obra, en calquera forma ou por calquera medio (electrónico, mecánico, gravación, fotocopia ou outros) sen consentimento expreso por escrito dos editores.

MATERIA: Sistemas Dixitais
TITULACIÓN: Grao en Enxeñaría Informática
PROGRAMA XERAL DO CURSO
Localización da presente unidade didáctica

BLOQUE I. INTRODUCIÓN

Unidade I. Introdución ós sistemas dixitais

- Conceptos de sistema dixital combinacional e secuencial
- Deseño e análise estruturada de sistemas dixitais
- O computador desde a perspectiva estrutural
- Computadores Von Neumann

Unidade II. Sistemas de numeración, aritmética e

- Sistemas de representación numérica de números enteiros
- Conversión entre bases
- Operacións aritméticas en binario
- Códigos BCD, exceso a 3 e alfanuméricos
- Códigos detectores e correctores de erro

BLOQUE II: SISTEMAS DIXITAIS COMBINACIONALES

Unidade III. Análise e síntese de circuítos lóxicos combinacionais

- Fundamentos da álxebra de Boole
- Funcións de conmutación
- Portas lóxicas
- Simplificación de funcións de conmutación
- Deseño e síntese de circuítos lóxicos combinacionais
- Análise de circuítos combinacionais

Unidade IV. Lóxica combinacional modular

- Deseño e utilización de módulos combinacionais básicos: codificadores e decodificadores
- Conversores de código
- Multiplexores e demultiplexores
- Circuítos aritméticos e lóxicos: comparadores, sumadores/restadores
- Unidades aritmético-lóxicas (ALU)
- Síntese de funcións mediante módulos combinacionais

BLOQUE III. SISTEMAS DIXITAIS SECUENCIALES

Unidade V. Sistemas secuenciais

- Concepto de estado
- Diagrama de estados
- Sistemas síncronos e asíncronos
- Biestables
- Modelos de sistemas secuenciais síncronos: máquinas de Mealy e de Moore
- Sistemas síncronos incompletamente especificados
- Deseño e simplificación de sistemas secuenciais síncronos

Unidade V. Lóxica secuencial modular

Deseño de módulos secuenciais: rexistros, contadores, xeradores de secuencia

Tipos básicos de memorias

Tipos de dispositivos lóxicos programables (PLDs): características

ÍNDICE

PRESENTACIÓN

A Unidade Didáctica 3 titulada «Análise e síntese de circuítos lóxicos combinacionais» enmárcase dentro dos contidos relativos á materia de Sistemas Dixitais, de 6 créditos ECTS impartida no primeiro semestre do primeiro curso do Grao en Enxeñaría Informática. Esta é unha materia con marcado carácter introdutorio para a titulación xa que é a primeira materia da área de Arquitectura e Tecnoloxía de Computadores que cursa o alumnado. O obxectivo básico desta materia é que os alumnos adquiran os coñecementos básicos e as habilidades de deseño precisas para estudar outras disciplinas relacionadas con esta área. Ó mesmo tempo, proporciona unha introdución ó estudo dos sistemas de procesamento dixital.

Esta Unidade Didáctica (en adiante, UD) presenta os elementos básicos e as ferramentas matemáticas para poder construír e analizar un circuíto combinacional que realice unha función lóxica. O único requisito previo da UD é ter un dominio básico do sistema binario de representación numérica de números enteiros, que foi presentado en Unidades Didácticas precedentes. Os contidos da UD son fundamentais para o estudo dos módulos combinacionais e dos sistemas secuenciais, que serán abordados en posteriores Unidades Didácticas.

OS OBXECTIVOS

O obxectivo global da UD é que o alumno dispoña dos coñecementos e habilidades necesarias para sintetizar e analizar un circuíto combinacional. Este obxectivo é o resumo dos seguintes obxectivos específicos:

- Coñecer os principios da Álxebra de Boole e comprender o funcionamento das operacións lóxicas.
- Identificar e interpretar as funcións de conmutación.
- Coñecer as distintas representacións das funcións de conmutación.
- Coñecer as portas lóxicas e mailo seu comportamento.
- Aplicar a metodoloxía de simplificación baseada nos mapas de Karnaugh.
- Coñecer e manexar os procesos de deseño e síntese, ó nivel de portas lóxicas, dun circuíto combinacional a partir da especificación verbal dunha función de conmutación.
- Coñecer e manexar o proceso de análise, ó nivel de portas lóxicas, dun circuíto combinacional.

OS PRINCIPIOS METODOLÓXICOS

A metodoloxía utilizada na UD combinará a exposición maxistral dos contidos teóricos co desenvolvemento práctico destes contidos por parte do alumno.

Os principios teóricos e os conceptos fundamentais da UD serán presentados polo profesor nas sesións expositivas. Durante as clases expositivas o profesor tamén desenvolverá diferentes exemplos prácticos, como os que se amosan nos Anexos 1 e 2 da UD, co obxectivo de facilitar a comprensión dos contidos teóricos. O profesor porá a disposición do alumnado o material de apoio utilizado durante estas sesións. Este material proporcionará unha referencia para seguir axeitadamente o fío das explicacións. En calquera caso, estes contidos deben ser desenvolvidos polo alumno, co apoio da bibliografía recomendada, fóra do horario de clases expositivo.

Ó finalizar as correspondentes sesións expositivas da UD, terán lugar 3 sesións interactivas asociadas á UD. A asistencia a estas sesións interactivas é obrigatoria. Nestas sesións realizaranse actividades que permitirán ó alumno afianzar os conceptos presentados nas sesións expositivas, así como manexar algunhas técnicas de simplificación, síntese e validación relacionadas cos circuítos combinacionais.

OS CONTIDOS BÁSICOS

Nesta UD preséntanse os fundamentos dos procesos de deseño, síntese e análise de sistemas dixitais. En primeiro lugar, describiranse a álxebra de Boole, as funcións de conmutación e as portas lóxicas. A continuación, explicaranse os procesos de deseño e síntese dunha función de conmutación, incluíndo métodos de simplificación de funcións de conmutación. Finalmente, examinarase o proceso de análise de circuítos combinacionais.

1. Fundamentos da álxebra de Boole

A álxebra de Boole é un conxunto matemático distribuído e complementado. Os elementos e propiedades básicos desta álxebra poden resumirse nos seguintes póstulados:

1. Unha álxebra booleana é un sistema alxebraico cerrado, formado por un conxunto B de 2 ou máis elementos e dous operadores, representados cos símbolos «+» e «·».
2. Neste conxunto existe un elemento neutro para cada unha das operacións, representados normalmente como 0 para a operación «+» e 1 para a operación «·». Así, sendo x un elemento de B, cúmprese que:

$$x+0 = x$$

$$x \cdot 1 = x$$

3. Propiedade conmutativa. Sendo x e y elementos de B, cúmprense as seguintes igualdades:

$$x+y = y+x$$

$$x \cdot y = y \cdot x$$

4. Propiedade asociativa. Sendo x, y e z elementos de B, cúmprense as seguintes igualdades:

$$x+(y+z) = (x+y)+z$$

$$x\cdot(y\cdot z) = (x\cdot y)\cdot z$$

5. Propiedade distributiva. Sendo x, y e z elementos de B, cúmprense as seguintes igualdades:

$$x+(y\cdot z) = (x+y)\cdot(x+z)$$

$$x\cdot(y+z) = (x\cdot y)+(x\cdot z)$$

6. Para todo elemento x que pertenza a B existe un elemento complementario $\overline{x}$, de tal xeito que:

$$x+\overline{x} = 1$$

$$x\cdot\overline{x} = 0$$

Ademais, a álxebra de Boole tamén cumpre o principio de dualidade. Este principio di que un teorema ou igualdade, deducido a partir dos postulados descritos anteriormente, pode transformarse nunha expresión válida sen máis que intercambiar as operacións binarias e os elementos identidade. As expresións amosadas nos postulados 2, 3, 4, 5 e 6 son unha mostra deste principio de dualidade.

Na álxebra de Boole pódense deducir, a partir dos postulados anteriores, unha serie de teoremas moi útiles na álxebra booleana. Estes teoremas enuméranse na táboa 1. Nestes teoremas, as letras x, y e z representan a elementos do conxunto B.

No capítulo 2 de Nelson *et ál*. (1996) pode atoparse información máis detallada sobre os conceptos desenvolvidos nesta sección. Este libro

Táboa 1: Teoremas fundamentais da álxebra de Boole

1. Idempotencia $x+x = x$ $x\cdot x = x$	5. $x+\overline{x}\cdot y = x+y$ $x\cdot(\overline{x}+y) = x\cdot y$
2. Elementos neutros $x+1 = 1$ $x\cdot 0 = 0$	6. $x\cdot y+x\cdot\overline{y} = x$ $(x+y)(x+\overline{y}) = x$
3. Involución $\overline{\overline{x}} = x$	7. $x\cdot y+x\cdot\overline{y}\cdot z = x\cdot y+x\cdot z$ $(x+y)\cdot(x+\overline{y}+z) = (x+y)\cdot(x+z)$
4. Absorción $x+x\cdot y = x$ $x\cdot(x+y) = x$	8. Teoremas de Morgan $\overline{(x+y)} = \overline{x}\cdot\overline{y}$ $\overline{(x\cdot y)} = \overline{x}+\overline{y}$

emprega unha notación na que combinación de dous elementos booleanos sen ningún operador é interpretado como a operación «·». Esta notación é moi habitual no eido dos sistemas dixitais polo que, a partir de agora, utilizarase tamén nesta unidade didáctica. Os teorema de DeMorgan –que se amosan na táboa 1– teñen unha especial relevancia nos sistemas dixitais. Estes teoremas son analizados en profundidade, desde unha perspectiva enfocada ós sistemas dixitais, no capítulo 4 de Floyd (2006).

2. Funcións de conmutación

Se o conxunto B soamente ten dous elementos –normalmente representados polos elementos lóxicos 0 e 1–, a álxebra denomínase «álxebra de conmutación». Os circuítos dixitais poden interpretarse como unha implementación física da álxebra de conmutación, xa que se poden identificar os valores lóxicos da álxebra de conmutación coa ausencia ou presenza de magnitudes físicas nestes circuítos. Por iso, os circuítos dixitais tamén se coñecen co nome de circuítos de conmutación.

As funcións de conmutación son funcións de n variables, que están definidas na álxebra de conmutación, polo que, tanto a propia función como as súas variables só poden tomar os valores 0 e 1. É importante darse conta que, dado que as variables son binarias (só poden tomar dous valores), sempre existirán 2^n posibles combinacións diferentes das variables de entrada.

2.1. Representacións

As funcións de conmutación poden representarse mediante mediante formas alxebraicas ou táboas de verdade.

Unha forma alxebraica é unha expresión que indica se unha combinación de valores das variables a función proporciona o valor 1 na saída da función. Esta expresión está formada pola combinación de operacións e literais. Os literais representan ós diferentes valores que poden tomar as variables binarias, polo que comprenden tanto as propias variables como os seus complementados. As formas alxebraicas máis habituais son a forma normal disxuntiva e a forma normal conxuntiva, tamén coñecidas como suma de produtos (*sum of products*, SOP) e produto de sumas (*product of sums*, POS), respectivamente. A suma de produtos é unha expresión onde se realiza a operación «+» sobre termos produto. Cada termo produto está formado por unha ou máis variables da función sobre as que se aplica a operación «·». O produto de sumas é unha expresión onde se realiza a operación «·» sobre termos suma. Cada termo suma está formado por unha ou máis variables da función sobre as que se aplica a operación «+». En realidade, ambas formas son o reflexo do principio de dualidade.

As táboas de verdade indican explicitamente o valor da función para calquera posible combinación de valores das variables de entrada. As primeiras n columnas da táboa representan as n variables da función.

A última columna representa o valor da función. Cada fila da táboa representa unha combinación diferente dos valores binarios das n variables (esta combinación tamén se coñece como estado), así como o valor da función asociado a esa combinación. É importante darse conta que, dado que as variables son binarias, unha táboa de verdade de n variables terá sempre 2^n filas. O Anexo 1 da UD contén un exemplo práctico de construción dunha táboa de verdade.

2.2. Formas canónicas

En xeral, existen diferentes formas SOP equivalentes para unha mesma función. Porén, existe unha representación SOP con características especiais que se coñece como forma SOP canónica e que está formada exclusivamente por «mintermos». Un mintermo dunha función de n variables é un termo produto de n literais no cal están presentes (na súa forma complementada ou non complementada) tódalas variables da función. Analogamente, existe unha representación POS canónica formada exclusivamente por «maxtermos». Un maxtermo dunha función de n variables é un termo suma de n literais no cal están presentes presentes (na súa forma complementada ou non complementada) tódalas variables da función.

As formas canónicas teñen unha relación moi estreita coas táboas de verdade, xa que os mintermos da forma SOP canónica están relacionados coas filas da táboa de verdade cun valor da función igual a 1. Así mesmo, as filas da táboa de verdade cun valor igual a 0, están asociadas ós maxtermos da forma POS canónica. Polo tanto, é posible asignar un valor numérico ós mintermos e maxtermos dunha función de n variables, se utilizamos unha orde predeterminada para ordenar as filas dunha táboa de verdade. O método máis amplamente utilizado é asignar ó valor binario correspondente ó estado asociado a cada fila. Esta cuantificación da lugar a diversas representacións das formas canónicas. Por exemplo, na literatura é moi frecuente empregar as variables m_i e M_i para referirse, respectivamente, ó mintermo e ó maxtermo asociados co valor i.

Pode atoparse máis información sobre as diferentes representacións das formas canónicas, así como sobre a súa relación coas táboas de verdade, no capítulo 2 de Nelson *et ál*. (1996). O Anexo 1 da UD contén un exemplo práctico da obtención das formas SOP e POS canónicas.

2.3. Funcións incompletamente especificadas

Se as funcións teñen un valor definido para cada estado, dise que as funcións están completamente especificadas. Polo tanto, en caso contrario, cando existe algunha combinación de valores das variables na que a función toma un valor indefinido ou irrelevante, dise que a función é incompletamente especificada (Llorís Ruíz, 2003: 66-68). Dito doutro xeito, existen funcións que non precisan tódalas posibles filas das táboas de verdade −ou, analogamente, tódolos mintermos/maxtermos das formas canónicas− para estar perfectamente definidas. Estas situacións, que se denominan indiferencias, suceden cando o número de posibles combinacións de variables con significado real (no contexto da función) é menor que 2^n.

As indiferenzas deben indicarse explicitamente, xa que condicionan o xeito de analizar e sintetizar as funcións. Nas formas canónicas as indiferenzas poden indicarse a través do valor binario asociado ós mintermos ou maxtermos para os que a función ten un valor indefinido. Nas táboas de verdade as indiferenzas indícanse cun «X» ou cun «-» na liña correspondente. Cando se emprega a notación m_i ou M_i, as indeterminacións soen indicarse coa letra d.

3. Portas lóxicas

As portas lóxicas son circuítos que realizan funcións lóxica elementais. As portas lóxicas máis comúns en sistemas dixitais son: NOT, AND, NAND, OR, NOR, XOR (ou OR exclusiva) e XNOR (ou NOR exclusiva).

- A porta NOT realiza a operación complemento sobre unha única variable. O seu símbolo máis común móstrase na figura 1(a). Porén, é habitual empregar o círculo como unha simplificación deste símbolo. Esta simplificación emprégase, fundamentalmente, cando a porta NOT está presente nas entradas ou na saída doutras portas lóxicas.
- A porta AND realiza a operación booleana «·» sobre dúas ou máis variables. O seu símbolo máis común, para o caso de dúas entradas, móstrase na figura 1(b).
- A porta NAND realiza o complemento da operación booleana «·». O seu símbolo máis común é igual que o da porta AND pero cun círculo na súa saída, coma o amosado na figura 1(c).
- A porta OR realiza a operación booleana «+» sobre dúas ou máis variables. O seu símbolo máis común, para o caso de dúas entradas, móstrase na figura 1(d).
- A porta NOR realiza o complemento da operación booleana «+». O seu símbolo máis común é igual que o da porta OR pero cun círculo na súa saída, como na figura 1(e).
- A porta XOR coñécese tamén como OR exclusivo. Esta porta implementa a seguinte función de dúas variables:

  $$f(x,y) = \overline{x}y+x\overline{y}$$

  Esta función soe representarse co símbolo $\oplus$ nas expresións alxebraicas. O símbolo máis común desta porta móstrase na figura 1(f).
- A porta XNOR coñécese tamén como NOR exclusivo, e realiza a función XOR complementada. Ó igual que coas portas NAND e NOR, o símbolo máis común da porta XNOR é igual que o da porta XOR pero cun círculo na saída, coma na figura 1(g).

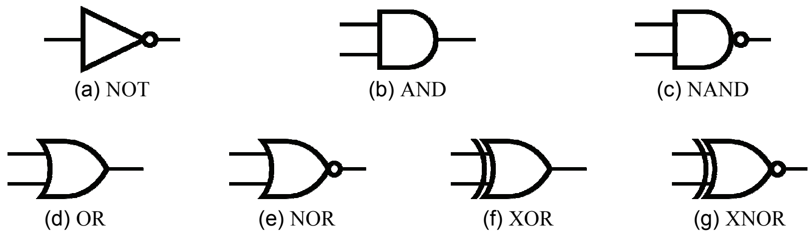


Figura 1: Símbolos máis comúns das portas lóxicas

Na táboa 2 indícanse as táboas de verdade das portas lóxicas descritas. Na figura 1 están representados os símbolos tradicionais, aínda que tamén son comúns os símbolos rectangulares incorporados no estándar ANSI/IEEE 91-1984 (Floyd, 2006: 124-155; Nelson *et ál.*, 1996: 108-120).

3.1. Equivalencias entre portas

Tendo en conta os postulados e teoremas do álxebra de Boole, é posible obter expresións lóxicas equivalentes ás funcións que realizan as portas lóxicas. Isto significa que as funcións lóxicas (complementario, «·», «+» e ⊕) non se identifican exclusivamente cunha única porta lóxica e, polo tanto, estas funcións poden implementarse con diferentes portas lóxicas. Algunhas das equivalencias máis comúns en sistemas dixitais son:

— Unha porta OR é equivalente a unha porta NAND coas entradas complementadas; figura 2(a).
— Unha porta NOR é equivalente a unha porta AND coas entradas complementadas; figura 2(b).
— Unha porta AND é equivalente a unha porta NOR coas entradas complementadas; figura 2(c).
— Unha porta NAND é equivalente a unha porta OR coas entradas complementadas; figura 2(d).

Táboa 2: Táboas de verdade das portas lóxicas máis habituais.

x	y	NOT x	x AND y	x NAND y	x OR y	x NOR y	x XOR y	x XNOR y
0	0	1	0	1	0	1	0	1
0	1	1	0	1	1	0	1	0
1	0	0	0	1	1	0	1	0
1	1	0	1	0	1	0	0	1

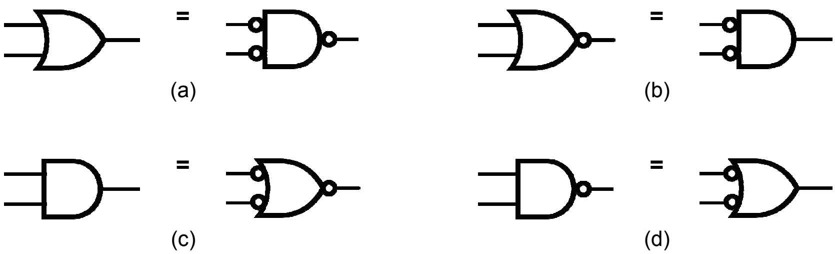


Figura 2: Equivalencias entre diferentes portas lóxicas

En particular, estas equivalencias poden obterse aplicando directamente as leis de De Morgan (táboa 2) sobre as funcións lóxicas que realizan as portas OR, NOR, AND e NAND.

3.2. Conxuntos universais de portas

Un conxunto de portas lóxicas completo, ou conxunto universal, é un conxunto de portas lóxicas mediante o cal se pode realizar calquera función lóxica. O conxunto completo básico, AND-OR-NOT, é o subconxunto formado polas portas AND, OR e NOT. Estas tres portas realizan directamente as operacións básicas da álxebra de Boole, polo que con elas pode realizarse calquera outra función lóxica.

Existen outros conxuntos completos de portas con menos de tres elementos: AND-NOT, OR-NOT, NAND e NOR. Para probar un conxunto é completo, basta con demostrar que pode realizar as funcións básicas do conxunto completo AND-OR-NOT. Por exemplo, as portas NAND son un conxunto completo xa que se cumpren as seguintes igualdades:

NOT x = x NAND x

x AND y = (x NAND y) NAND (x NAND y)

x OR y = (x NAND x) NAND (y NAND y)

Analogamente, podemos obter as correspondentes igualdades entre o conxunto completo AND-OR-NOT e o conxunto completo NOR:

NOT x = x NOR x

x AND y = (x NOR x) NOR (y NOR y)

x OR y = (x NOR y) NOR (x NOR y)

A figura 3 mostra de xeito gráfico estas equivalencias entre os conxuntos completos AND-OR-NOT, NAND e NOR.

No capítulo 5 de Floyd (2006) pode atoparse unha descrición máis detallada dos conxuntos universais NAND e NOR. Neste capítulo tamén se analiza o emprego destes conxuntos universais para sintetizar circuítos lóxicos.

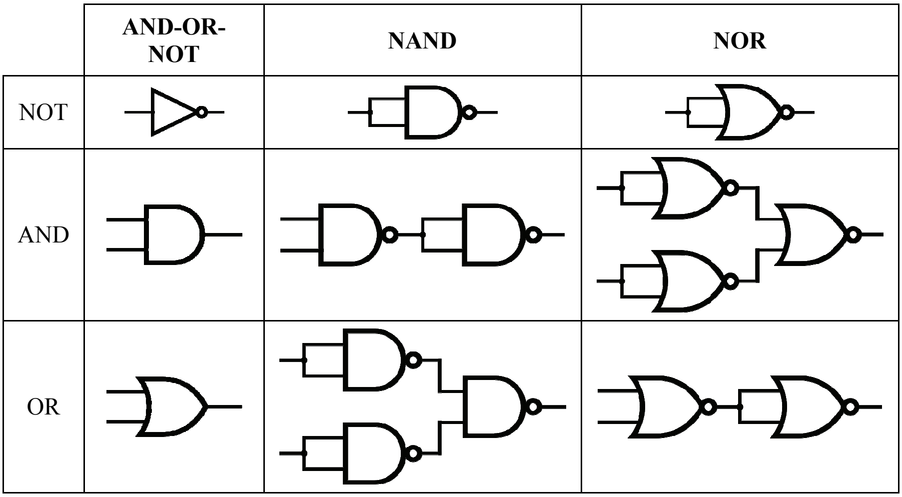


Figura 3: Equivalencias entre conxuntos completos de portas lóxicas

4. Simplificación de funcións de conmutación

A simplificación das funcións de conmutación ten como obxectivo final reducir o custe da implementación de funcións con portas lóxicas. Dependendo dos parámetros de deseño existen diferentes alternativas para reducir estes custes, pero, en xeral, o que se pretende é obter a expresión máis sinxela posible dunha función. Nesta UD centrarémonos exclusivamente na simplificación de funcións lóxicas, obviando outros aspectos para reducir o custe dos circuítos lóxicos. Outros aspectos de deseño (Llorís Ruíz *et ál.*, 2003: 111-150) están fóra dos obxectivos da UD.

A simplificación das funcións lóxicas pode realizarse de diferentes xeitos. Por un lado, uso dos postulados e teoremas da álxebra de conmutación e quizais o método máis inmediato para intentar simplificar a expresión dunha función lóxica. Porén, a capacidade deste método para obter un resultado óptimo depende da habilidade e experiencia do deseñador para escoller adecuadamente o teorema ou postulado que se debe aplicar en cada situación. Polo outro lado, existen métodos sistemáticos que garanten a obtención dunha expresión mínima. Un dos máis coñecidos é o método dos mapas de Karnaugh ou método Veitch-Karnaugh. Este método é adecuado para simplificar manualmente funcións de menos de 5 variables, xa que está baseado en estruturas xeométricas. No resto dos casos, nos que o método de Karnaugh é de difícil aplicación, é preciso utilizar outros métodos manuais máis sofisticados –como o método Quine-McCluskey (Llorís, Prieto e Parrilla, 2003: 91-123)– ou software específico (Angulo Usategui e García Zubía, 2002: 66).

4.1. Mapas de Karnaugh

O fundamento do método de simplificación baseado nos mapas de Karnaugh é o de analizar as adxacencias das posibles combinacións de entrada ou estados da función. Dous estados son adxacentes cando os mintermos ou maxtermos asociados ó estado só se diferencian nun literal. Polo tanto, dous mintermos adxacentes que aparezan na forma SOP canónica dunha función de n variables poden simplificarse, aplicando os postulados 5 e 6, nunha expresión de n-1 literais. Pode facerse un razoamento análogo para o caso de formas POS canónicas.

Un mapa de Karnaugh é unha matriz de celas, na que cada cela está asociada con un número binario, do mesmo xeito que as filas dunha táboa de verdade. Polo tanto, o mapa de Karnaugh dunha función de n variables terá 2^n celas. En realidade, o mapa de Karnaugh contén a mesma información que unha táboa de verdade, pero organizada de xeito que a celas asociadas a mintermos ou maxtermos adxacentes sexan tamén adxacentes na estrutura matricial. Para lograr este tipo de adxacencias, utilízase unha organización baseada no código Gray (Nelson *et ál*., 1996: 62; Floyd, 2006: 96). A figura 4(a) e 4(b) mostran a estrutura matricial dun mapa de Karnaugh para unha función de 3 e para una función de 4 variables, respectivamente. Nestas figuras, o número situado na esquina superior esquerda das celas indica o número binario asociado ó estado correspondente. Debe terse en conta que nun mapa de Karnaugh as celas só son adxacentes en vertical ou horizontal (nunca en diagonal), coma os movementos dunha torre no xadrez. Hai que ter en conta tamén que as celas da fila superior son adxacentes ás celas da fila inferior, como se o mapa de Karnaugh se dobrase como un cilindro. Sucede o mesmo para as celas das columnas da esquerda e da dereita (Nelson, 2006: 230).

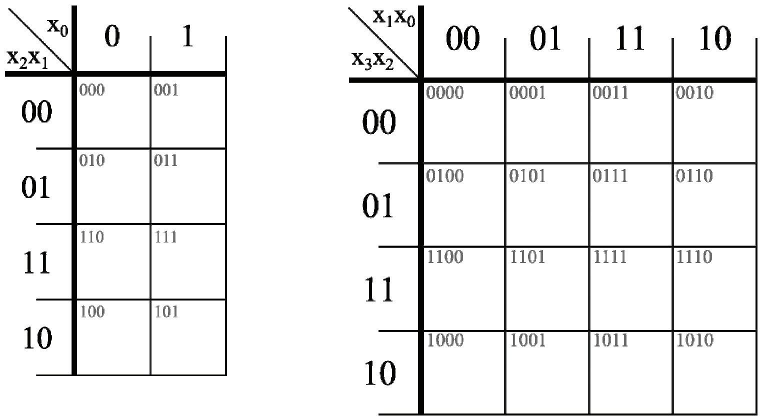
A continuación explicarase o método de simplificación baseado nos mapas de Karnaugh para obter formas SOP mínimas de funcións de menos de 5 variables. Esta análise pódese realizar tamén para obter forma POS mínimas (Nelson *et ál*., 1996: 197-203) e tamén para funcións de 5 variables (Floyd, 2006: 247).

Antes de explicar as regras do método de Karnaugh, é preciso establecer algunhas definicións:

Implicante: é un termo produto da función. Nos mapas de Karnaugh identificase como un conxunto de mintermos situados en celas adxacentes, formando un cadrado ou un rectángulo cun número de mintermos que sexa potencia enteira de 2 (1, 2, 4, 8 ou 16 elementos).

Implicante primo: implicante que non está contido completamente dentro dun implicante de maior tamaño.

Implicante primo esencial: implicante primo que contén un ou máis mintermos que non pertenzan a ningún outro implicante.



(a) f(x2,x1,x0) (a) f(x3,x2,x1,x0)

Figura 4: Estruturas dos mapas de Karnaugh para funcións de 3 e 4 variables

As regras do método de simplificación de Karnaugh son:

1. Representar os mintermos da función nun mapa de Karnaugh, poñendo un 1 na celas que teña a mesma numeración que os mintermos da función. Por exemplo, se a forma canónica SOP dunha función de catro variables contén o mintermo m_4, entón deberá poñerse un 1 na cela 0100.
2. Agrupar tódolos mintermos da función co mímino número de implicantes.
   - En primeiro lugar deben seleccionarse os implicantes primos esenciais. Hai que buscar celas que só pertenzan a un implicante, e seleccionar ese implicante.
   - A continuación, selecciónanse os implicantes primos por orde de tamaño, favorecendo sempre ós implicantes máis grandes, ata que tódolos mintermos estén contidos, como mínimo, nun implicante. Unha cela pode pertencer a varios implicantes se con iso logramos implicantes de maior tamaño.
3. Cando unha función é incompletamente especificada, as indeterminacións poden incluírse como elementos dos implicantes se iso xera implicantes de maior tamaño. Isto é posible porque estamos a forzar que a función teña un 1 nun estado que, en calquera caso, é irrelevante (Floyd, 2006: 240).
4. Cada implicante seleccionado correspóndese cun termo produto composto por tódolos literais dos mintermos asociados ó implicante, excepto aqueles que presenten unha mesma variable de forma complementada e non complementada. Noutras palabras, o termo produto conten tódolos literais que sexan comúns a tódolos mintermos contidos no implicante.

O Anexo 2 da UD contén un caso práctico de simplificación dunha función de catro variables utilizando o método dos mapas de Karnaugh.

5. Deseño e síntese de circuítos combinacionais

Nesta sección descríbese o proceso de deseño e síntese dun circuíto combinacional a partir da especificación verbal do circuíto. Este proceso pode resumirse nos seguintes puntos:

1. <u>Esquema básico do circuíto</u>. En primeiro lugar, compre debuxar un esquema básico do circuíto que se desexa implementar. Este esquema serve para identificar claramente cales son as entrada e as saídas do circuíto.
2. <u>Codificación binaria das entradas e saídas</u>. Para cada unha das entradas do circuíto é preciso identificar cada posible situación e asignarlle unha codificación binaria. O número de posibles valores dunha entrada determinará o número de bits (e, polo tanto, o número de variables) que se precisan para codificar esa entrada. Tamén é preciso identificar os posibles situacións da saída, realizando a correspondente codificación binaria. Neste caso, cada bit da codificación binaria da saída será tratado como unha función de conmutación independente.
3. <u>Construción das táboas de verdade das funcións de saída</u>. Para cada unha das funcións de saída constrúese a correspondente táboa de verdade e obtense a expresión das diferentes funcións de conmutación na súa forman SOP canónica.
4. <u>Simplificación da función de conmutación</u>. O obxectivo é obter a expresión mínima da función de conmutación. É recomendable utilizar un método sistemático que garanta o resultado óptimo. Para funcións de menos de 5 variables recoméndase a utilización do método de mapas de Karnaugh.
5. <u>Implementación da función conectando portas lóxicas</u>. A partir da expresión mínima obtida anteriormente, a función de conmutación impleméntase nun circuíto lóxico. As diferentes operacións lóxicas da expresión son realizadas mediante portas lóxicas adecuadamente interconectadas. Recoméndase construír o circuíto de xeito ordenado e cunha estrutura regular, que permita unha inspección axeitada en caso de ser preciso algún tipo de rectificación.
6. <u>Verificación da implementación do circuíto</u>. O último paso consiste na verificación da implementación particular que se fixo do circuíto. Para iso, é preciso verificar que a saída do circuíto, para un estado particular das entradas, se corresponde cos valores de saída indicados nas táboas de verdade correspondentes.

6. Análise de circuítos combinacionais

A análise dos circuítos combinacionais é o proceso inverso da síntese de circuítos combinacionais (Nelson *et ál.*, 1996: 120). Partindo da implementación con portas lóxicas, o obxectivo principal da análise é obter a expresión alxebraica da función de conmutación, ou calquera outra forma de descrición do seu comportamento. O método alxebraico de análise consiste en substituír, unha a unha, cada porta lóxica pola correspondente función de conmutación. Neste método, tódalas saídas das portas lóxicas son consideradas como variables temporais do circuíto. O proceso comeza pola porta máis próxima á saída do circuíto, obtendo unha primeira expresión da función de conmutación buscada. Deste xeito, esta primeira aproximación correspóndese coa función de conmutación da primeira porta substituída. A continuación, as variables temporais que vaian aparecendo nesta expresión son substituídas polas funcións de conmutación das portas que xeran esas variables temporais. O proceso repítese ata que non existan variables temporais na expresión alxebraica da función de conmutación. Polo tanto, tódalas variables na expresión final da función de conmutación correspóndense con variables de entrada do circuíto.

Ademais da función lóxica do circuíto, a análise dun circuíto combinacional permite avaliar certas características físicas que dependen do número e configuración das portas dun circuíto (Nelson *et ál.*, 1996: 125). Un exemplo é a análise dos retardos de propagación, xa que as portas lóxicas teñen un determinado tempo de resposta fronte ós estímulos nas entradas. O tempo de resposta dun circuíto indica o tempo mínimo que a saída tarda en responder ante un cambio nalgunha da súas entradas. Nunha primeira aproximación, o tempo de resposta dun circuíto é proporcional ó número máximo de portas que apareza nalgún dos diferentes camiños que unen cada entrada coa saída do circuíto.

ACTIVIDADES PROPOSTAS

As actividades asociadas a esta UD teñen como obxectivo afianzar os contidos teóricos presentados nas clases expositivas e familiarizarse cos procesos de deseño, síntese e análise de sistemas combinacionais.

Por unha parte, durante o desenvolvemento da UD, os alumnos terán a súa disposición unha listaxe de exercicios para que, de xeito autónomo, practiquen os conceptos e métodos presentados nas clases expositivas. O Anexo 3 da UD contén un mostra representativa deste tipo de exercicios. Estes exercicios poderán ser utilizados polo alumno como un mecanismo de autoavaliación continua da UD. En calquera caso, o alumnado pode complementar esta listaxe con outros exercicios que se atopen na bibliografía. Neste senso, recoméndase especialmente consultar libros que conteñan exercicios resoltos (García Zubía, 2003; Gascón de Toro et ál., 1990; González García, 1996; Velasco Ballano e Otero Arias, 1995).

Pola outra banda, a UD ten asociadas 3 sesións interactivas, de carácter obrigatorio, nas que se traballarán tamén os conceptos e métodos presentados na UD. En particular, o traballo destas sesións distribuirase do seguinte xeito:

- Na primeira sesión realizarase un exercicio en grupos reducidos. O obxectivo do exercicio é afianzar os conceptos teóricos asociados á UD e practicar algúns métodos de simplificación de funcións de conmutación. O exercicio consistirá nunha competición entre os grupos para resolver exercicios propostos polo profesor. As normas concretas da competición serán explicadas polo profesor ó comezo da mesma.
- Nas dúas últimas sesións, os procesos de deseño, simplificación, síntese e validación de circuítos de conmutación serán exercitados a través de problemas propostos polo profesor. Estes problemas deben ser resoltos en parellas. A parte de deseño ten que realizarse en papel, namentres que a síntese e verificación farase nun entorno software axeitado para tal fin. Ó comezo da primeira destas dúas últimas sesións, o profesor realizará un pequeno exercicio práctico para amosar as principais características e funcionalidades deste contorno software.

AVALIACIÓN DA UNIDADE DIDÁCTICA

A avaliación realizarase durante o desenvolvemento da unidade didáctica nas clases interactivas (avaliación procesual) e ó seu remate (avaliación final).

A avaliación procesual realizarase durante as clases interactivas asociadas á unidade didáctica. Os criterios de avaliación que se aplicarán neste caso son:

- Os exercicios en grupos reducidos, serán avaliados en función da cantidade de exercicios resoltos correctamente polo grupo, de acordo coas normas establecidas polo profesor. Así mesmo, o profesor tamén valorará a cohesión e participación dos compoñentes do grupo.
- Os exercicios en parellas das clases interactivas avaliaranse mediante o ensino dos diferentes exercicios propostos. O profesor valorará a súa correcta implementación e verificación, así como os coñecementos do alumno mediante preguntas acerca do proceso de deseño e síntese.

A avaliación final realizarase mediante un exame escrito, ó remate das sesións interactivas asociadas á unidade didáctica. Este exame permitirá ó estudante mostrar o grao de coñecemento adquirido, no que se refire á comprensión dos conceptos e técnicas expostos na unidade.

ANEXOS

Anexo 1: representación de funcións de conmutación

Dada a función $f(x,y,z) = \overline{x}y + xz$, calcular:

a) A táboa de verdade.
b) A forma SOP canónica.
c) A forma POS canónica.

Resolución:

a) Primeiro, hai que construír a estrutura da táboa de verdade (TV). Xa que a función ten 3 variables, a TV ten oito (2^3) filas e catro (3+1) columnas. Hai unha columna para cada variable e unha columna para os valores da función. As columnas correspondentes ás variables hai que enchelas coas 2^3 posibles combinacións diferentes de valores. Para non esquecerse de ningunha combinación, recoméndase seguir unha orde. Neste caso, por exemplo, cada combinación reflicte a codificación binaria de números enteiros correspondente ó número da fila (comezamos en 0):

x	y	z	f(x,y,z)
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Para encher a última columna, hai que avaliar o valor da función para cada estado. Por exemplo:

$$f(0,1,0) = 1\cdot 1 + 0\cdot 0 = 1$$

$$f(1,1,0) = 0\cdot 1 + 1\cdot 0 = 0$$

O resultado final é:

x	y	z	f(x,y,z)
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	1

b) Para obter a forma SOP canónica podemos utilizar a TV calculada anteriormente. Os mintermos da función están asociados coas combinacións de valores nos que a función ten un 1. Polo tanto, podemos indicar a forma SOP canónica como:

$$f(x,y,z) = m_2 + m_3 + m_5 + m_7$$

Esta expresión pode expresarse de xeito máis condensado indicando soamente os números dos mintermos (o símbolo Σ indica que se realiza a operación «suma» sobre os termos produto):

$$f(x,y,z) = \Sigma m\ (2,3,5,7)$$

Para atopar a expresión alxebraica é preciso substituír cada mintermo polo seu correspondente termo produto. O termo produto dun mintermo obtense a partir da combinación de valores asociada. Se a variable presenta un 0 nesa combinación, o termo produto do mintermo correspondente conterá esa variable complementada. Se, pola contra, a variable presenta un 1, o termo produto conterá esa variable sen complementar. Por exemplo, o mintermo m_2 está asociado á combinación $x,y,z = 010$, polo que o termo produto asociado será $\overline{x}y\overline{z}$. Polo tanto, a forma SOP canónica é:

$$f(x,y,z) = \overline{x}y\overline{z} + \overline{x}yz + x\overline{y}z + xyz$$

Poderiamos obter esta mesma expresión a partir da expresión inicial, utilizando o teorema 6 do álxebra de Boole (véxase a táboa 2).

c) Para obter a forma POS canónica podemos utilizar tamén a TV calculada anteriormente. Os maxtermos da función están asociados coas combinacións de valores nos que a función ten un 0. Polo tanto, podemos indicar a forma POS canónica como:

$$f(x,y,z) = M_0 \cdot M_1 \cdot M_4 \cdot M_6$$

Esta expresión pode expresarse de xeito máis condensado indicando soamente os números dos maxtermos (o símbolo Π indica que se realiza a operación «produto» sobre os termos suma):

$$f(x,y,z) = \Pi M\ (0,1,4,6)$$

Para atopar a expresión alxebraica é preciso substituír cada maxtermo polo seu correspondente termo suma. O termo suma dun maxtermo tamén se obtén a partir da combinación de valores asociada, pero de xeito inverso a como ser fai no caso dos mintermos. Se a variable presenta un 1 nesa combinación, o termo suma do maxtermo correspondente conterá esa variable complementada. Se, pola contra, a variable presenta un 0, o termo suma conterá esa variable sen complementar. Por exemplo, o maxtermo M_1 está asociado á combinación $x,y,z = 001$, polo que o termo suma asociado será $x+y+\overline{z}$. Polo tanto, a forma SOP canónica é:

$$f(x,y,z) = (x+y+z)(x+y+\overline{z})(\overline{x}+y+z)(\overline{x}+\overline{y}+z)$$

Do mesmo xeito que na forma SOP canónica, poderiamos obter esta mesma expresión a partir da expresión inicial, utilizando o teorema 6 do álxebra de Boole (véxase a táboa 2).

Anexo 2: simplificación de funcións de conmutación

Simplificar mediante o método de mapas de Karnaugh a función lóxica que identifica se un número binario de 4 bits é múltiplo de 2 ou de 3.

Resolución:

A forma SOP canónica da función que nos están pedindo é:

$$f(x_3,x_2,x_1,x_0) = \Sigma m\ (2,3,4,6,8,9,10,12,14,15)$$

onde x_i reflicte o bit de peso i do número binario.

Tendo en conta as regras do método de simplificación, en primeiro lugar hai que situar os mintermos da función nas correspondentes celas dun mapa de Karnaugh de catro variables:

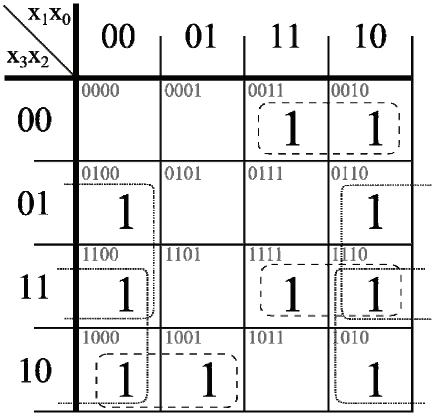
x_3x_2 \ x_1x_0	00	01	11	10
00	0000	0001	0011 1	0010 1
01	0100 1	0101	0111	0110 1
11	1100 1	1101	1111 1	1110 1
10	1000 1	1001 1	1011	1010 1

A continuación hai que seleccionar os implicantes axeitados:

— Neste caso, hai tres implicantes primos esenciais, que son os formados polas parellas de celas 3-2, 15-14 e 8-9. Estes implicantes primos son precisos para incluír ós mintermos 3, 15 e 9.

— Non existe unha posibilidade única para incluír ó resto de mintermos que quedan por cubrir. Polo tanto, hai que escoller os implicantes primos con maior número de elementos, aínda que se solapen celas. Neste caso, o máis axeitado é escoller o implicante formados polas celas 4-6-12-14 e o implicante formado polas celas 12-14-8-10.

Graficamente, o resultado é:



Os termos produto que se corresponden con cada implicante son:

Celas do implicante	Termo produto
3-2	$\overline{x}_3\overline{x}_2x_1$
15-14	$x_3x_2x_1$
8-9	$x_3\overline{x}_2\overline{x}_1$
4-6-12-14	$x_2\overline{x}_0$
12-14-8-10	$x_3\overline{x}_0$

Polo tanto, a expresión mínima da función é:

$$f(x_3,x_2,x_1,x_0) = \overline{x}_3\overline{x}_2x_1 + x_3x_2x_1 + x_3\overline{x}_2\overline{x}_1 + x_2\overline{x}_0 + x_3\overline{x}_0$$

Anexo 3: propostas de exercicios

A continuación, preséntase unha mostra de exercicios asociados á UD. Os obxectivos que se pretenden acadar en cada exercicio están indicados ó final do mesmo.

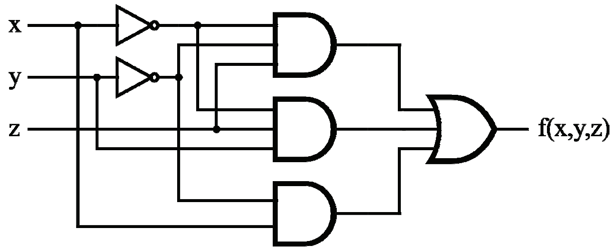
1. Obter a expresión lóxica mínima para que a implementación das seguintes expresións lóxicas teña o menor número de portas. A simplificación deber realizarse utilizando os postulados e teoremas da álxebra de Boole. Supoñer que unicamente se utilizan portas de dúas entradas.
   a) $f(x,y,z) = \overline{x}yz + \overline{x}\,\overline{y}$
   b) $f(x,y,z) = \overline{x}\,\overline{z} + \overline{y}\,\overline{z}$
   c) $f(x,y,z) = x(\overline{z} + \overline{y}) + xy$

   Obxectivo: manexo dos postulados e teoremas da álxebra de Boole.

2. Expresar a función $f(a,b,c,d) = \overline{a}bc + \overline{a}b + a\overline{b}\overline{c}$ como produto de mintermos e como produto de maxtermos.

   Obxectivo: manexo das formas canónicas das funcións de conmutación.

3. Tendo en conta o circuíto da figura:



   a) Obter a expresión lóxica asociada á saída.
   b) Obter a expresión na forma SOP canónica.
   c) Obter a expresión na forma POS canónica.

   Obxectivo: análise de circuítos de conmutación e manexo das formas canónicas das funcións de conmutación.

4. Simplificar a función $f(a,b,c,d) = \Sigma m(0,2,6,11,13,14) + \Sigma d(1,3,9)$ como suma de produtos.

   Obxectivo: simplificación de funcións de conmutación.

5. Deseñar un circuíto combinacional que teña catro chaves de entrada (A, B, C e D) e dúas lámpadas de saída (L_1 e L_0). O circuíto debe comportarse de acordo cos seguintes puntos:
   a) A lámpada L_1 debe alumear cando o número de chaves acesas sexa maior ou igual que o número de chaves apagadas.
   b) A lámpada L_0 debe alumear cando o número de chaves apagadas sexa maior ou igual que o número de chaves acesas.
   c) Non está permitido que tódalas chaves estean simultaneamente no mesmo estado (non poden estar as catro acesas, nin tampouco as catro apagadas).

   Implementar o circuíto utilizando o mínimo número de portas AND, OR e NOT que sexa posible.

   Obxectivo: obtención da función de conmutación a partir dunha especificación verbal, simplificación de funcións de conmutación e síntese de circuítos combinacionais.

6. Homer (H) está na ribeira dun río coa súa filla Maggie (M), co seu can (C) e cun bote de veleno para ratas (V). Homer ten que cruzar o río nunha gamela. O problema é que, como Homer está tan gordo, na gamela só hai sitio para Homer e pouco máis. Polo tanto, Homer ten que facer varias viaxes, e en cada viaxe ten que escoller entre carrexar a filla, carrexar o can ou carrexar o bote de veleno. Agora ben, Homer non pode deixar sos ó can e maila filla, porque o can pode morder á nena. Tampouco pode deixar a Maggie co bote de veleno, porque a nena podería tragar o veleno. Con estes datos, hai que achar a táboa de verdade dunha función, $f(H,M,C,V)$, que lle diga a Homer cales son as potenciais situacións de perigo. Esta función ten que valer 1 se existe perigo, e dicir, se a nena queda soa co can ou co bote de veleno nunha das ribeiras. A función ten que valer 0 en caso contrario. Obter a expresión mínima como SOP e interpretar o resultado. Suxestión: codificar as variables H, M, C e V cun 1 para indicar se está na ribeira esquerda e cun 0 para indicar se está na ribeira dereita.

   Obxectivo: obtención da función de conmutación a partir dunha especificación verbal e simplificación de funcións de conmutación.

BIBLIOGRAFÍA

ANGULO USATEGUI, José Mª e Javier GARCÍA ZUBÍA (2003): *Sistemas digitales y tecnología de computadores*, Madrid: Thomson-Paraninfo.

BAENA OLIVA, Carmen, Manuel Jesús BELLIDO DÍAZ, Alberto Jesús MOLINA CANTERO, María del Pilar PARRA FERNÁNDEZ e Manuel VALENCIA BARRERO (2001): *Problemas de circuitos y sistemas digitales*, Madrid: McGraw-Hill/Interamericana de España.

FLOYD, Tomas L. (2006): *Fundamentos de sitemas digitales*, Madrid: Pearson Educación.

GARCÍA ZUBÍA, Javier (2003): *Problemas resueltos de electrónica digital*, Madrid: Thomson-Paraninfo.

GASCÓN DE TORO, Manuel, Antonio LEAL HERNÁNDEZ e Virginia PEINADO BOLOS (1990): *Problemas prácticos de diseño lógico*, Madrid: Paraninfo.

GONZÁLEZ GARCÍA, Marco Antonio (1996): *Ejercicios resueltos de sistemas digitales*, Santiago de Compostela: Tórculo.

LLORÍS RUÍZ, Antonio, Alberto PRIETO ESPINOSA e Luis PARILLA ROURE (2003): *Sistemas digitales*, Madrid: McGraw-Hill/Interamericana de España.

NELSON, Victor P., H. Troy NAGLE, Bill D. CARROLL e J. David IRWIN (1996): *Análisis y diseño de circuitos lógicos digitales*, México: Prentice Hall.

VELASCO BALLANO, Joaquín e José OTERO ARIAS (1996): *Problemas de sistemas electrónicos digitales*, Madrid: Paraninfo.

unidadesdidácticas
UNIVERSIDADE DE SANTIAGO DE COMPOSTELA

Unha colección orientada a editar materiais docentes de calidade e pensada para apoiar o traballo do profesorado e do alumnado de todas as materias e titulacións da universidade

Impreso en papel 100% reciclado e libre de cloro

ISBN 978-84-9887-903-2
9 788498 879032

USC
UNIVERSIDADE
DE SANTIAGO
DE COMPOSTELA

SERVIZO DE NORMALIZACIÓN
LINGÜÍSTICA