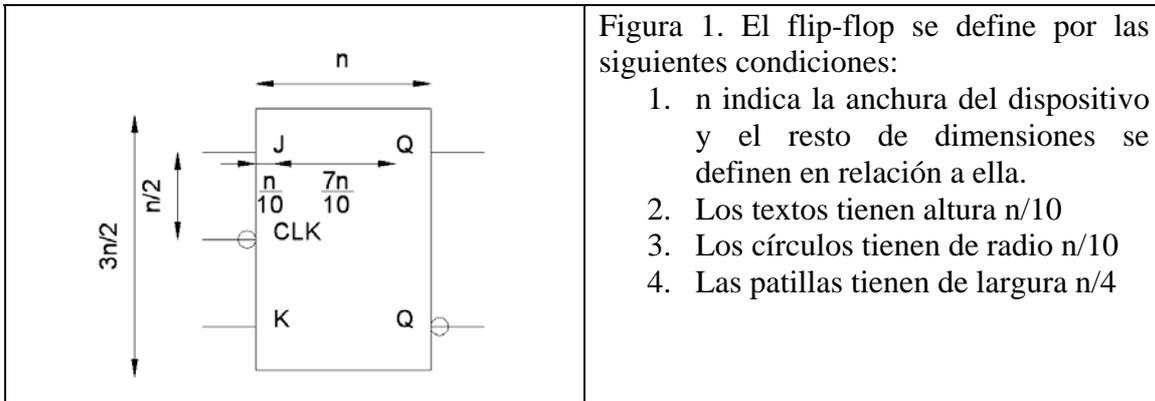


Estructuras Secuenciales. Funciones de Control. Junio 2007

1. Escribir una función **VLISP FJK (N P / ...)** que dibuje un flip-flop JK como se muestra en la figura. Las dimensiones están descritas también en la figura. (1 pt)



2. Para crear un contador asíncrono ascendente es necesario conectar varios flip-flop del tipo anterior en cascada.

Escribir una función **VLISP CONTADOR (n R P /)** que dibuje 2 flip-flop como los descritos en el apartado anterior, conexionados como se muestran en las figuras, dependiendo del valor de su argumento R. Si el argumento R vale TRUE el contador será ascendente (figura 2) y si vale NIL el contador será descendente (figura 3). n indica la anchura del flip-flop y P el punto de inserción. Las dimensiones de las conexiones vienen indicadas por la rejilla, que tiene una separación entre puntos de n/2. Asimismo, las conexiones que diferencian ambos circuitos, vienen marcadas en línea gruesa. (3 pt)

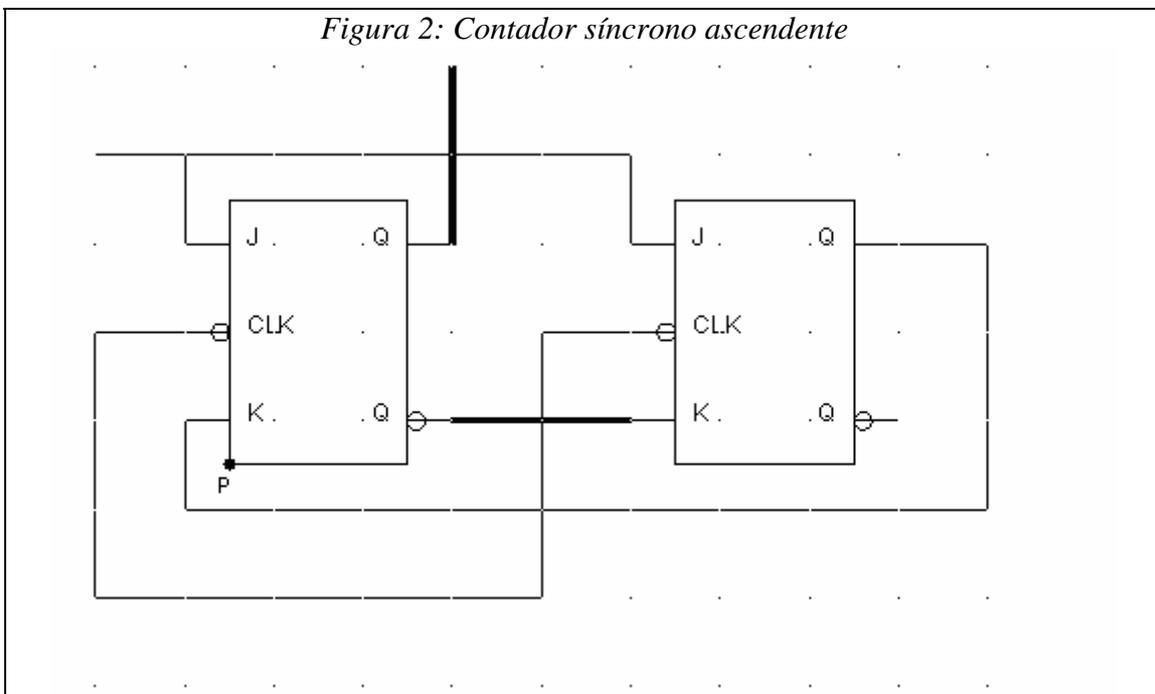
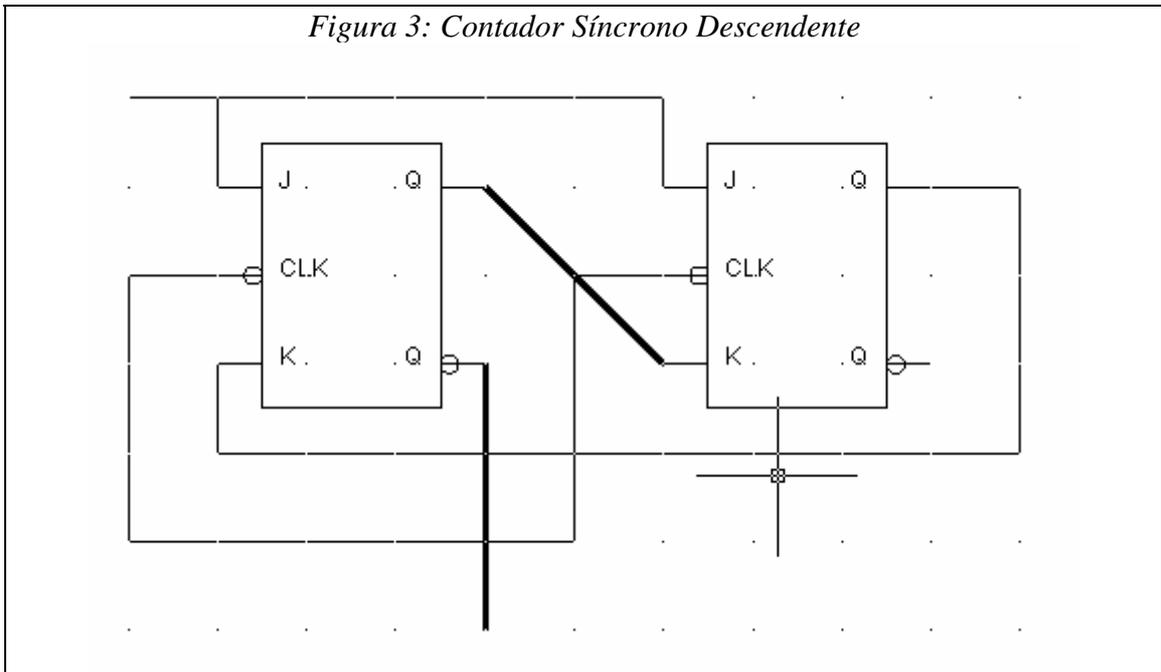


Figura 3: Contador Síncrono Descendente



1. Escribir una función VLISP **NAND (N S P / . . .)** que dibuje una puerta lógica NAND como se muestra en la figura 3. Las dimensiones de la misma están descritas en la figura. (1p)

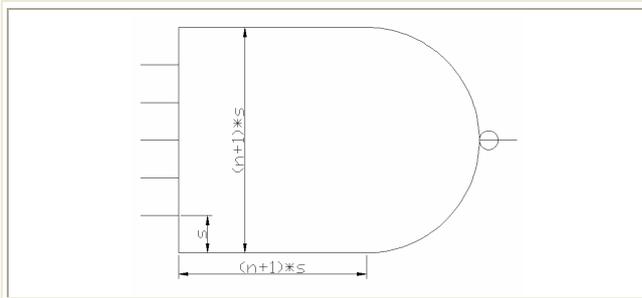


Fig 3. La puerta NAND se define por las siguientes condiciones:

- 1) S indica la separación entre conectores y la longitud de los mismos
- 2) N es el número de conectores
- 3) El círculo de salida tiene de diámetro $s/2$

2. Escribir una función VLISP **CIRCUITO (S L P LC / ...)** que dibuje dos puertas lógicas como las descritas en el apartado anterior. La función además pedirá interactivamente al usuario las conexiones para cada puerta, en formato $LC = ((con1 con2)(con1 con2))$ y dibujará líneas de longitud l en las conexiones correspondientes (3p)

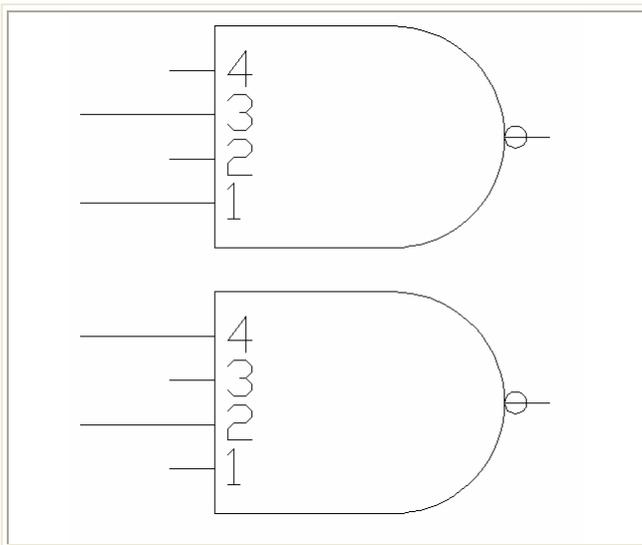


Fig 4.

Resumen de argumentos:

- s : espaciado entre conectores
- l : distancia entre los puntos de inserción de las dos puertas lógicas.
- P : punto de inserción de la primera puerta lógica.
- LC : lista de conexiones.

La distancia entre los puntos de inserción de las dos puertas es $3 * l$

Por ejemplo, en la figura, LC vale $((2\ 4)(1\ 3))$

1. Escribir la función VLISP **DIENTE** (P L / . . .), cuyas características se ofrecen en la figura 1. P es el punto de base de inserción y se sitúa en el extremo inferior izquierdo del diente.

NOTA IMPORTANTE: *junto al código de la función, hacer un croquis con los nombres de las variables auxiliares de punto que se han usado y su posición en el gráfico.* (0,5p)

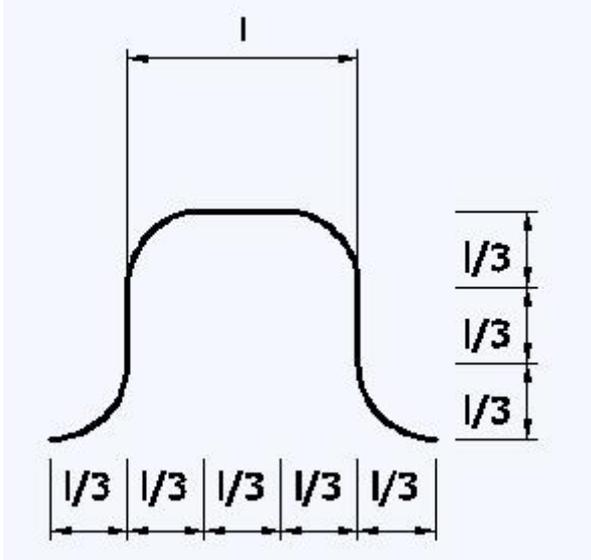
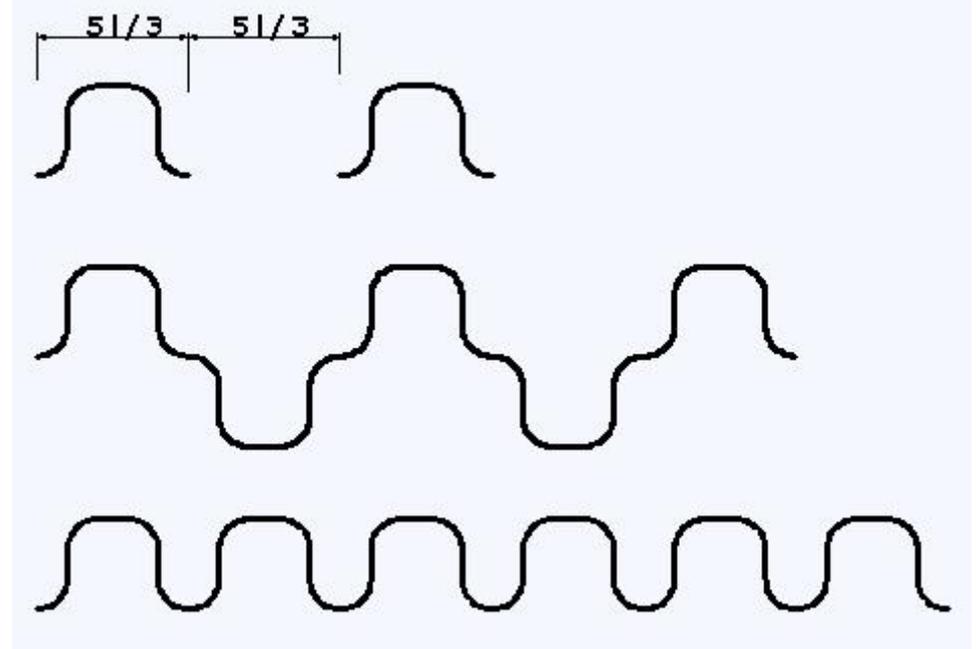
2. Escribir una función VLISP **C:ENGRANA** (P / . . .) cuyas características se ofrecen en la figura 2.

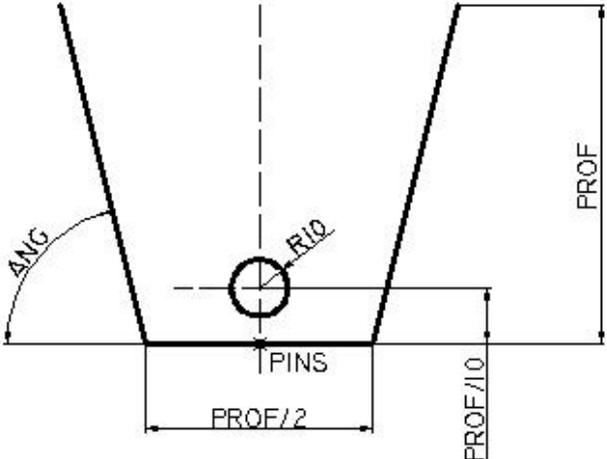
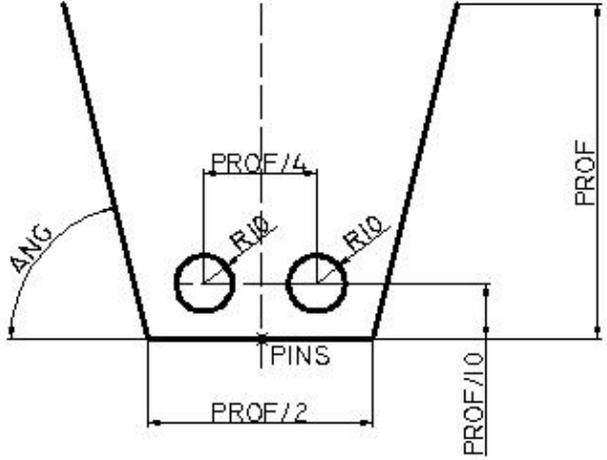
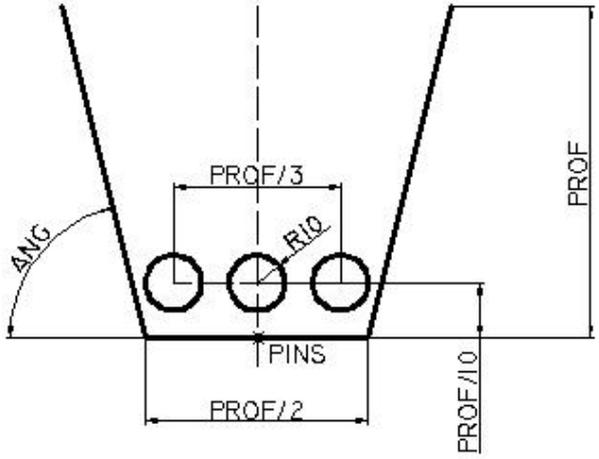
El parámetro N controla la separación entre dientes de la siguiente manera:

- Si vale 2, los dientes están separados un valor $5L/3$.
- Si vale 5, los dientes están juntos pero se dibujan alternativamente hacia arriba y hacia abajo.
- Si vale 6, los dientes se dibujan correlativos
- Si N recibe otros valores, se corta la corriente del programa y se avisa al usuario.

P es el punto de base de inserción y se sitúa en el extremo inferior izquierdo del primer diente.

IMPORTANTE: *crear preferentemente funciones auxiliares distintas para cada opción: se puntuará fundamentalmente que todo el código se haya distribuido en funciones simples que sean invocadas adecuadamente.* (2,5p)

<p align="center"><u>FUNCIÓN A DEFINIR</u></p>	<p align="center"><u>RESULTADO GRÁFICO</u> SÓLO LO QUE ESTÁ EN TRAZO GRUESO</p>
<p>NOMBRE: DIENTE (L / ...)</p> <p>L: ancho total</p> <p>EFECTO: dibuja las tres líneas y los cuatro arcos que se representan en trazo grueso en la figura de la derecha</p> <p>DEVUELVE: NIL</p> <p align="right">FIG 1 →</p>	
<p>NOMBRE: C:ENGRANA (/ ...)</p> <p><u>DATOS INTERACTIVOS:</u></p> <p>L: ancho total N: número de dientes. Puede valer 2, 5 ó 6. Ver enunciado.</p> <p>EFECTO: dibuja el tramo dentado tal como se representa en la figura de la derecha</p> <p>DEVUELVE: NIL</p> <p align="right">FIG 2 →</p>	

<p><u>FUNCIÓN A DEFINIR</u></p>	<p><u>RESULTADO GRÁFICO</u> SÓLO LO QUE ESTÁ EN TRAZO GRUESO</p>
<p>NOMBRE: ZANJA1 (PROF PINS ANG / ...)</p> <p>PROF: valor real PINS: lista de punto ANG: ángulo de las paredes en sexagesimales</p> <p>EFFECTO: dibuja las tres líneas y el círculo que se representan en trazo grueso en la figura de la derecha DEVUELVE: NIL</p> <p style="text-align: right;">FIG 1 →</p>	
<p>NOMBRE: ZANJA2 (PROF PINS ANG / ...)</p> <p>PROF: valor real PINS: lista de punto ANG: ángulo de las paredes en sexagesimales</p> <p>EFFECTO: dibuja las tres líneas y los dos círculos que se representan en trazo grueso en la figura de la derecha DEVUELVE: NIL</p> <p style="text-align: right;">FIG 2 →</p>	
<p>NOMBRE: ZANJA3 (PROF PINS ANG / ...)</p> <p>PROF: valor real PINS: lista de punto ANG: ángulo de las paredes en sexagesimales</p> <p>EFFECTO: dibuja las tres líneas y los tres círculos que se representan en trazo grueso en la figura de la derecha DEVUELVE: NIL</p> <p style="text-align: right;">FIG 3 →</p>	

1. Escribir las tres funciones **VLISP ZANJA1**, **ZANJA2** y **ZANJA3** a partir de las especificaciones que se dan en las figuras 1, 2 y 3 en la hoja adjunta. (1p)

NOTA IMPORTANTE: *junto al código de la función, hacer un croquis con los nombres de las variables auxiliares de punto que se han usado y su posición en el gráfico.*

2. Escribir una función **VLISP C:ZANJAS ()** que solicite al usuario tres datos:

* El punto de inserción de la zanja PINS.

* La profundidad de la zanja PROF.

* El número de conducciones N, que sólo podrá valer 1, 2 o 3,

y dibuje las zanjas correspondientes según los gráficos del apartado 1. Este programa utilizará una función COND para distribuir las llamadas a las funciones auxiliares. También, este COND hará que se presente un mensaje si N tiene un valor distinto de 1, 2 ó 3. (1p)

EJERCICIOS DE LISTAS. (Para comprobar en la consola de VLISP)

1. Previo a los ejercicios, estudiar la información y los ejemplos que se adjuntan en los apartados 2.1 y 2.2 completos de AUTOLISP-TEORÍA, en WEBCT.

2. Descargar y estudiar los videos que hay en MATERIAL MULTIMEDIA-TEMAS PUNTUALES LISP – FUNCIONES DEMANIPULACIÓN DE LISTAS

3. Resolver los ejercicios siguientes.

1. Construir en la consola de VLisp la variable lista lis de valor

(2 4 6 (1 3 5) "abc" (10 . 20))

A continuación, accediendo a los elementos de la lista lis,

- Generar la lista (4 3 “abc”)
- Generar la lista (20 (10 . “abc”) (1 . 5))
- Indicar lo que devuelven las siguientes expresiones:
 - (nth 2 (nth 3 lis))
 - (cons (caddr lis) (caddrdr lis))
 - (list (caddrdr lis) lis)
 - (cons (caddrdr lis) lis)
 - (append (caddrdr lis) lis)
 - (cons (car lis) (cons (cadr lis) (car lis)))
 - (setq k (cons (cadr lis) (car lis)))
 - (setq p (cons 2 k))
- ¿ La variable p tiene asociado un par punteado o una lista ? Comprobar, antes de responder, en la consola los siguientes valores:
 - (car p)
 - (cadr p)
 - (caddr p)
 - (last p)

2. Generar una variable de nombre lis_asoc que sea una lista de asociaciones con el siguiente contenido:

Lis_asoc = ((“uno” 10) (“dos” 20) (“tres” 30) (“cuatro” 40))

- Qué devuelve la expresión (assoc 10 lis_asoc) ?
- Qué devuelve la expresión (assoc “dos” lis_asoc) ?
- ¿ Cómo se puede obtener el valor “tres” de esa lista ?
- ¿ Y el valor 30 ?
- Qué devuelve la expresión (subst 300 30 lis_asoc)?
- Qué devuelve la expresión (subst (cons “dos” 200) (assoc 30 lis_asoc) lis_asoc) ?
- Qué devuelve la expresión (subst (cons “dos” 200) (assoc “dos” lis_asoc) lis_asoc) ?

Justificar los resultados.