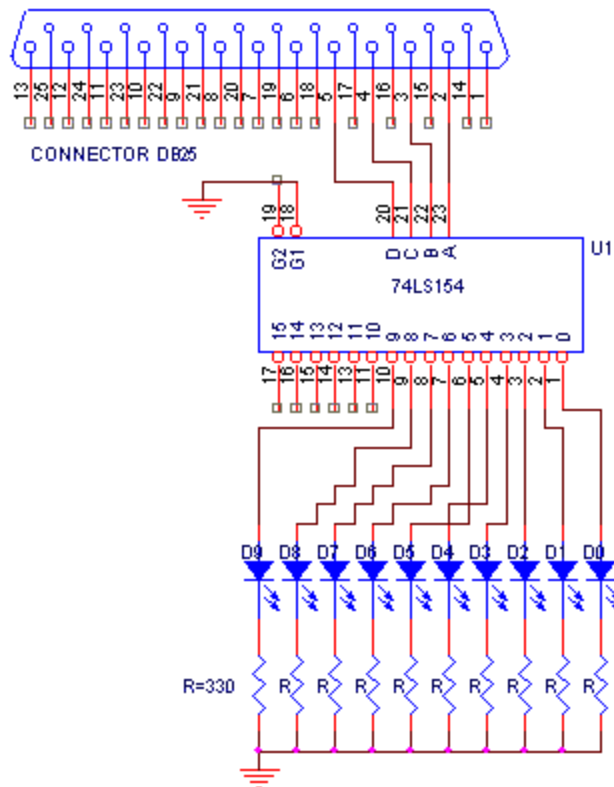


Probador de un Decoder 4 a 16 con 10 Led's

Probador de Luces con un decoder 4 – 16 (74LS154)

;Programa que simula a un decoder (74LS154), para probarlo
;basta con colocar a la salida del puerto paralelo de tu computadora
;el siguiente circuito



Donde cuando a las entrada(D0 D1 D2 D3) del 74LS154 hay un 1, ala salida el LED que le corresponde estara apagado, pero para muestra solo he trabajado con los Diez primeros valores

D3	D2	D1	D0	Entrada	Salida
0	0	0	0	0	111111110
0	0	0	1	1	111111101
0	0	1	0	2	111111011
0	0	1	1	3	111110111
0	1	0	0	4	111101111
0	1	0	1	5	111011111
0	1	1	0	6	110111111
0	1	1	1	7	101111111
1	0	0	0	8	011111111
1	0	0	1	9	011111111

Donde: 1 = LED[ON], 0 = LED [OFF]

Probador de un Decoder 4 a 16 con 10 Led's

```
.286
.model small
.stack 100h
.data
;*****
; Establece la Posición del cursor en el modo texto: 80x25
CURSOR MACRO FILA,COLUMNA
    PUSHA
    MOV DH,FILA
    MOV DL,COLUMNA
    MOV BH,0
    MOV AH,2
    INT 10H
    POPA
ENDM
;*****
; Limpia la pantalla en el modo de texto.
CLS MACRO
    PUSHA
    MOV CX,0
    MOV DX,2479H
    MOV BH,7
    MOV AX,0600H
    INT 10H
    POPA
    cursor 0,0
ENDM
;*****
; Imprime una cadena de caracteres.
CADENA MACRO BUFFER
    PUSHA
    LEA DX,BUFFER
    MOV AH,9
    INT 21H
    POPA
ENDM

espacio equ 20H
linea equ 10,13
titulo db linea
    db "                Uso de un Decoder de 4 a 16 ",linea
    db "                === == == ===== == ===== ",2 dup(linea)
    db "                Manejaremos nuestro decoder",linea
    db "                de manera que segun los datos (dx) se enciendan las

salidas (sx)$"
dec1  db "                $"
dec2  db "                |          s0 | - $"
dec3  db "                -|d0          s1 | - $"
dec4  db "                |          7   s2 | - $"
dec5  db "                -|d1   4   s3 | - $"
dec6  db "                |          L   s4 | - $"
dec7  db "                -|d2   S   s5 | - $"
dec8  db "                |          1   s6 | - $"
dec9  db "                -|d3   5   s7 | - $"
dec10 db "                |          4   s8 | - $"
dec11 db "                |          s9 | - $"
dec12 db "                |          $"
```

Probador de un Decoder 4 a 16 con 10 Led's

```
var1 db 0,1,2,3,4,5,6,7,8,9
var db ?
numeros dw 0000h,0001h,0010h,0011h,0100h,0101h,0110h,0111h,1000h,1001h
numero dw ?

tcontinua db "Para salir pulsa [S] otro para continuar ...$"
.code
inicio:
    mov ax,@data
    mov ds,ax

    cls
    cadena titulo
    cursor 7,20;alto, ancho
    cadena dec1
    cursor 8,20
    cadena dec2
    cursor 9,20
    cadena dec3
    cursor 10,20
    cadena dec4
    cursor 11,20
    cadena dec5
    cursor 12,20
    cadena dec6
    cursor 13,20
    cadena dec7
    cursor 14,20
    cadena dec8
    cursor 15,20
    cadena dec9
    cursor 16,20
    cadena dec10
    cursor 17,20
    cadena dec11
    cursor 18,20
    cadena dec12

    mov di,0
    mov si,0
    mov cx,10
    mov dx,378h
repite: mov al,var1[di]
        out dx,al
        mov var,al
        mov ax,numeros[si]
        mov numero,ax
        call visual
        inc di
        inc si
        inc si
        push cx
dem1:   mov cx,0ffffh
        push cx
        mov cx,07ffh
dem2:   loop dem2
```

Probador de un Decoder 4 a 16 con 10 Led's

```
        pop cx
        loop dem1
        pop cx
        call limpia
        loop repite

        cursor 20,10
        cadena tcontinua
        mov ah,7
        int 21h
        cmp al,'s'
        jz salir
        cmp al,'S'
        jz salir
        jmp inicio
salir:  cls
        mov ah,4ch
        int 21h

visual proc near
pusha
mov dl,20
mov bh,0
mov ah,2
mov dh,9;cursor en 9,20
int 10h
mov Dx,numero;DCBA
AND Dx,0fh
ADD D1,30h
mov ah,2
int 21h

mov dl,20
mov bh,0
mov ah,2
mov dh,11;cursor en 11,20
int 10h
mov Dx,numero;DCBA
shr dl,4
ADD D1,30h
mov ah,2
int 21h
mov dl,20
mov bh,0
mov ah,2
mov dh,13;cursor en 13,20
int 10h
mov Dx,numero;DCBA
xchg dh,dl
AND Dx,0fh
ADD D1,30h
mov ah,2
int 21h

mov dl,20
mov bh,0
mov ah,2
```

Probador de un Decoder 4 a 16 con 10 Led's

```
mov dh,15;cursor en 15,20
int 10h
mov Dx,numero;DCBA
xchg dh,dl
shr dl,4
ADD DL,30h
mov ah,2
int 21h

mov dl,24h
mov bh,0
mov ah,2
mov dh,8
add dh,var
int 10h
mov dl,'*'
mov ah,2
int 21h

popa
ret
visual endp

limpia proc near
    pusha
    mov dh,8
    mov cx,10
limp1:  mov dl,24h
        mov bh,0
        mov ah,2
        int 10h;cursor en 8,23h
        mov dl,' '
        mov ah,2
        int 21h
        inc dh
        loop limp1
        mov dh,8

        mov dh,9
        mov cx,4
limp2:  mov dl,20
        mov bh,0
        mov ah,2
        int 10h
        mov dl,' '
        mov ah,2
        int 21h
        add dh,2
        loop limp2
    popa
ret
limpia endp
end inicio
```

Probador de un Decoder 4 a 16 con 10 Led's

El Programa en Funcionamiento es de la siguiente Forma:

Uso de un Decoder de 4 a 16
=== == == ===== == =====

Manejaremos nuestro decoder
de manera que segun los datos (dx) se enciendan las salidas (sx)

			s0	-	
1	-	d0	s1	-	
		7	s2	-	
1	-	d1	4	s3	-
		L	s4	-	
1	-	d2	8	s5	-
		1	s6	-	
0	-	d3	5	s7	-*
		4	s8	-	
			s9	-	

Donde el '*'=s7 es equivalente a la entrada 0111=7; el '*'=Led [OFF]