



# BOARD XSA50

Estructura y  
funcionamiento de la  
placa XSA50

FCEIA

LDD

# Programación de la FPGA

La implementación de un diseño lógico sobre una FPGA, en general, pasa por las siguientes etapas:

- ✓ Se describe un circuito lógico con un HDL (o esquema, o ME, etc.)
- ✓ Se usa un programa sintetizador para transformar el HDL en un *netlist*. La aplicación utilizada (Xilinx Synthesis Technology -XST) , genera un archivo tipo NGC (EDIF + NCF – constricciones-).
- ✓ Se usan herramientas de implementación para mapear las puertas lógicas y sus conexiones (descritos en el *netlist*) sobre la lógica de una FPGA (CLB, I/OB y matriz de interconexiones)
- ✓ Una vez logrado el mapeo, ubicación e interconexión (paso anterior) se extrae el estado lógico de las llaves de interconexión (en las matrices de interconexión o ruteo), generando el *bitstream*.
- ✓ El *bitstream* es bajado a la memoria del chip de la FPGA (si todo anda bien, ésta luego se comportará según la descripción del código VHDL)

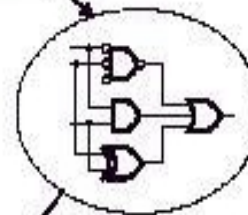
## Código fuente VHDL

```
entity ledded is
  port(
    d: in std_logic_vector(3 downto 0);
    s: out std_logic_vector(6 downto 0);
  );
end;

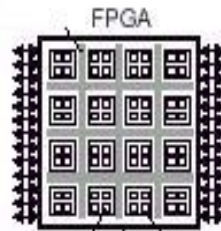
architecture ledded_arch of ledded is
begin
  s <= "1110111" when d="0000" else
       "0010010" when d="0001" else
       "1101101";
end ledded_arch;
```

sintetizador

Netlist



mapeo, ubicación e interconexión



generación del bitstream

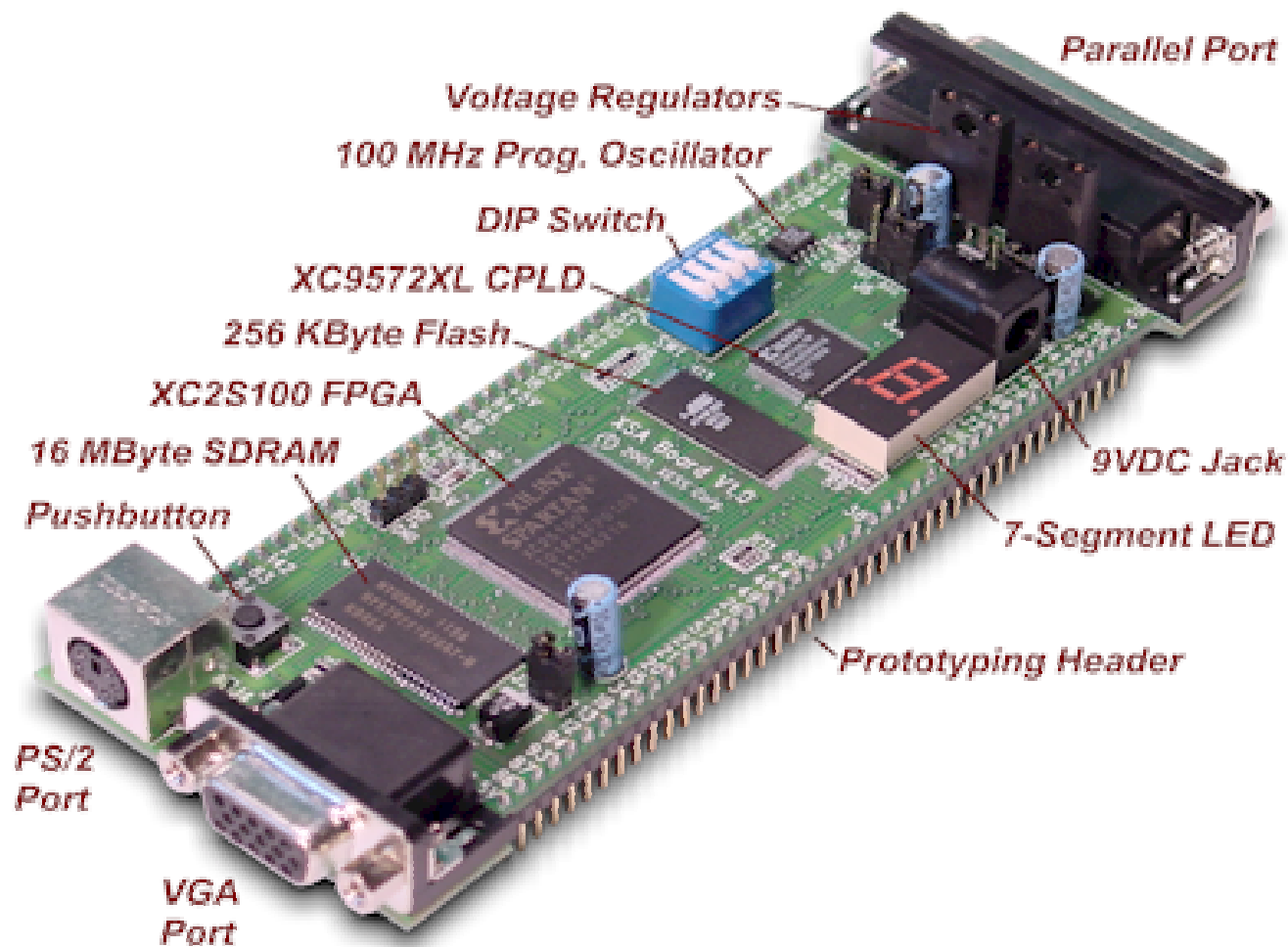
Bitstream

```
101010010101100101
010110100101010101
010110100101101011
010101001010101010
101010101001101010
110110110101001010
110100101011001011
001011001010101001
010101101001101001
011001100010101010
101010100110010101
```

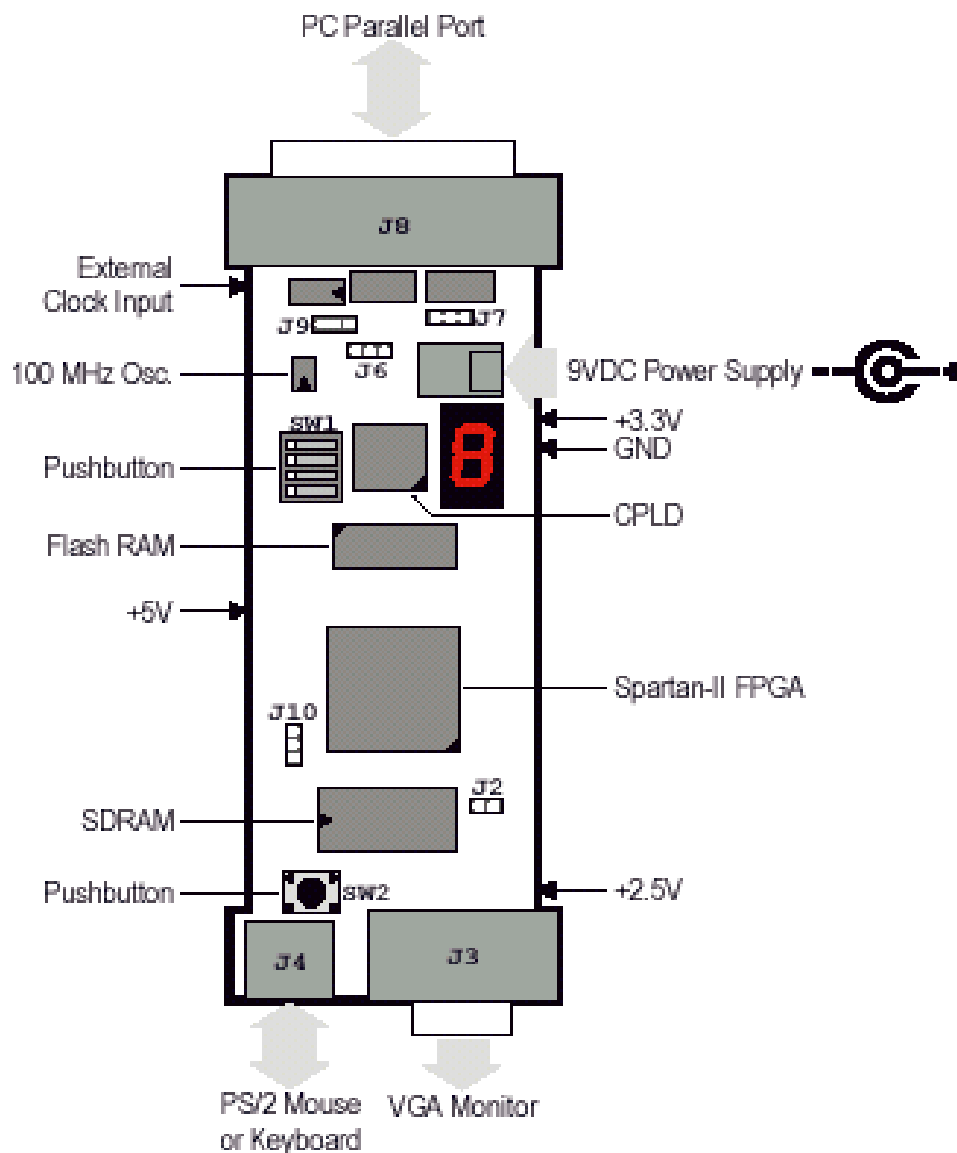
testeo y programación



# Una vista general



# Esquemático



# Componentes principales

- FPGA Spartan XC2S50
- CPLD XC9572 XL
- Generador de Clock Programable DS1075
- Memoria RAM dinámica sincrónica (SDRAM) de 16 Mbyte
- Memoria Flash (EEPROM) de 256K
- algunos “jumpers” que se usan en la fase de configuración

# Periféricos

- 4 micro interruptores
- 1 pulsador
- 1 display de 7 segmentos
- 1 puerto paralelo (interfaceado vía el CPLD)
- 1 salida VGA manejada directamente por la FPGA
- 1 interfase PS2

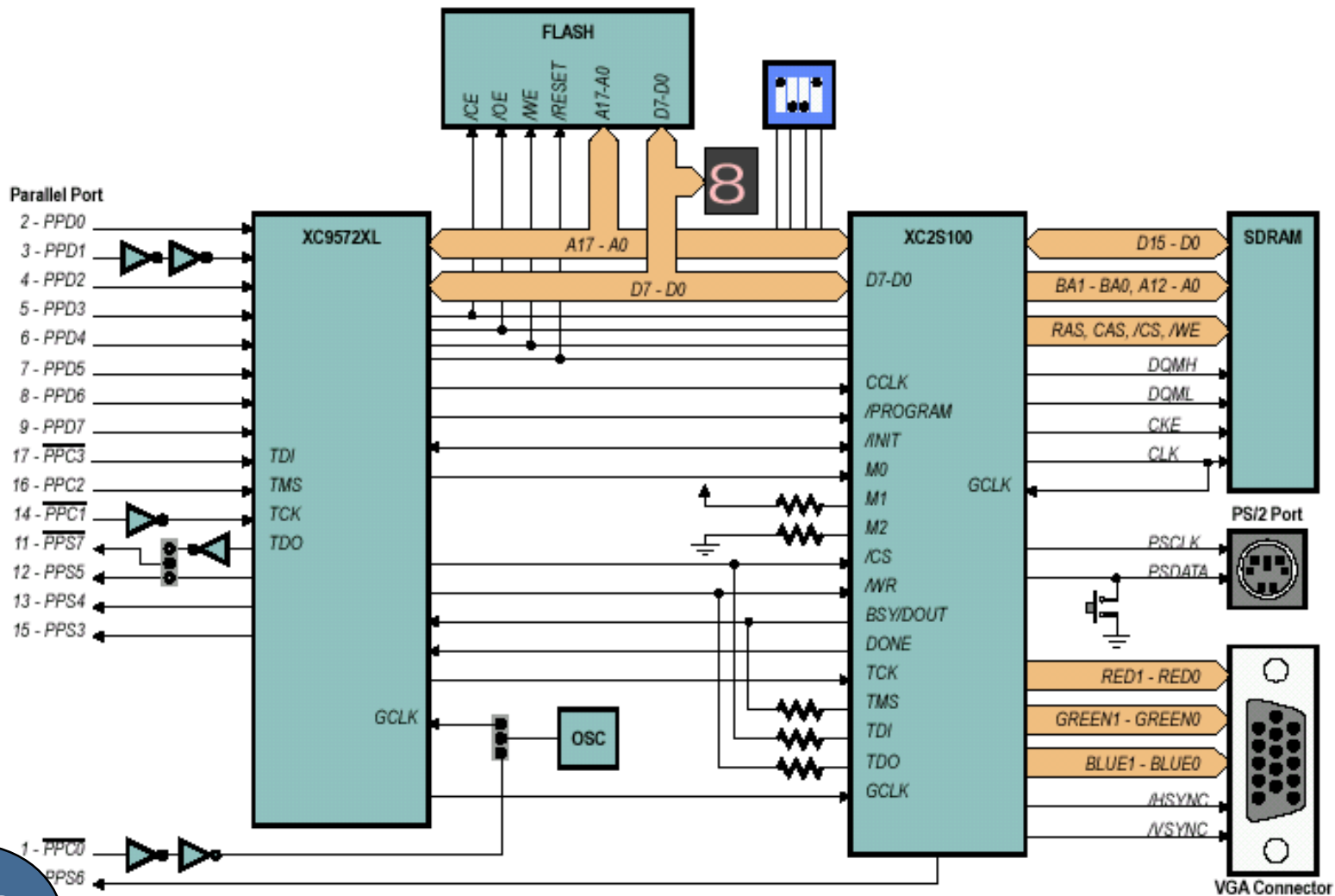
# Seteo de los *jumpers*

- Estas llaves deben manipularse sólo si:
  - Se programa la FPGA usando la aplicación de programación JTAG de Xilinx.
  - Se reprograma la frecuencia del reloj.
  - Se cambia la forma de alimentar con tensión la placa.

**Importante: Las placas del Laboratorio están seteadas para trabajar con las siguientes condiciones: 1) programación de la FPGA con la aplicación GXSLoad; 2) reloj con frecuencia de 50 MHz y 3) única fuente de alimentación de 9V DC.**

**¡ NO CAMBIAR!**

# Arquitectura



# Documentación

- Procurarse la información:
  - file: [xsa-manual-v1\\_2.pdf](#) (44 páginas)
  - [XS tools.pdf](#) (18 páginas)
  - sitio: [www.xess.com](#)
- Es útil porque:
  - Describe el funcionamiento del soft de la XESS
  - Describe la disposición y conexionado
    - pinout de los componenetes
    - señales involucradas
    - utilización del CPLD como interface
    - herramientas XS para bajar datos por la interface paralela

# Las XSTOOLS



☐ GXSTEST – Testea la placa



☐ GXSETCLK – Setea el reloj

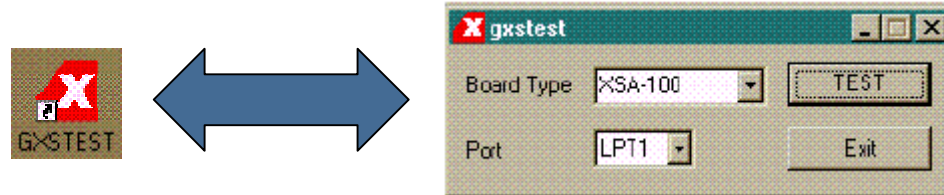


☐ GXSLD – Carga datos en: FPGA, CPLD y FLASH



☐ GXSPORT – Conecta los 8 *pins de datos* del puerto paralelo con la FPGA

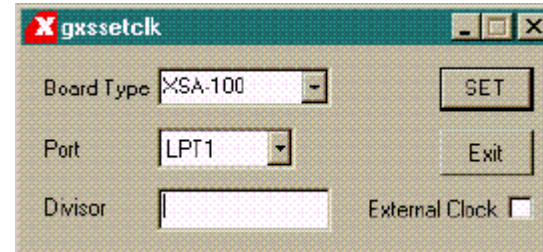
# Test de la placa



- Test de autodiagnóstico de la placa
- Configura primero el CPLD y luego la FPGA
- Si todo es correcto, muestra un “O” en el display
- Caso contrario pone una “E”
- De haber errores, lo más común es problemas con el cable o puerto paralelo.

NOTA: Verificar que esté seteado a 50 Mhz.

# Configuración del CLOCK



- Se puede configurar el reloj para entregar frecuencias desde 100 MHz hasta 48,7 KHz . (divisor entre 1 y 2052)
- Se configura el DS1075 para modificar la frecuencia bajando los datos por el puerto paralelo.
- Sin tensión en la placa, se coloca el *shunt* en las patas 2 y 3 del *jumper* J6 y luego se corre el “gxssetclk”
- La frecuencia *seteada* se memoriza en soporte no volátil

NOTA: No realizar esta operación con las placas del Laboratorio

# PROGRAMACION DE LA PLACA

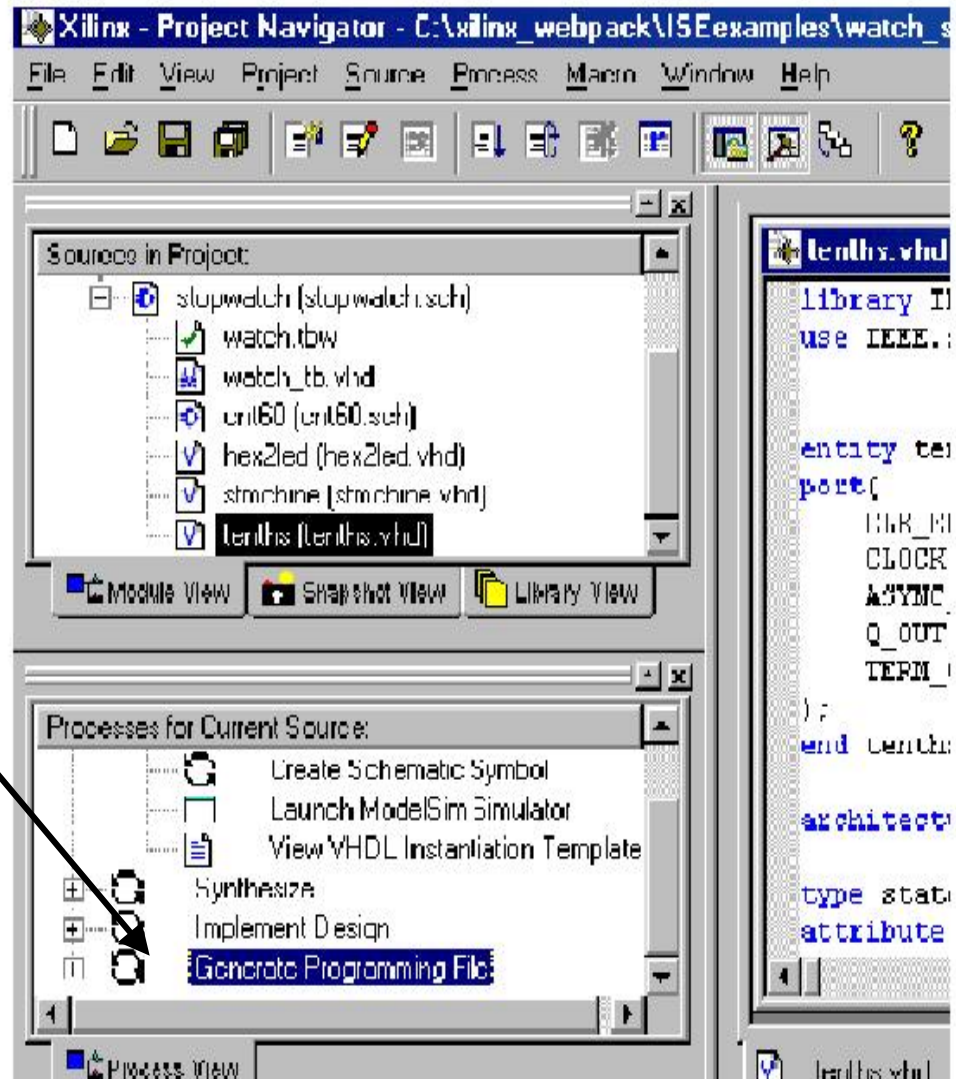
Se puede programar la FPGA o el CPLD

❑ FPGA ⇒ **.BIT**

❑ CPLD ⇒ **.SVF**

# Download a la FPGA

Con el el menú  
Generate Programming  
File, se genera el archivo  
(bitstream) *file.bit*

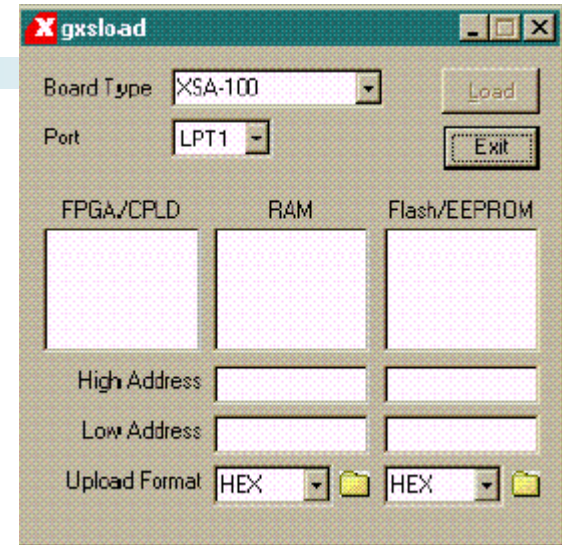


# Download a la FPGA

Utilización de la herramienta GXSLOAD

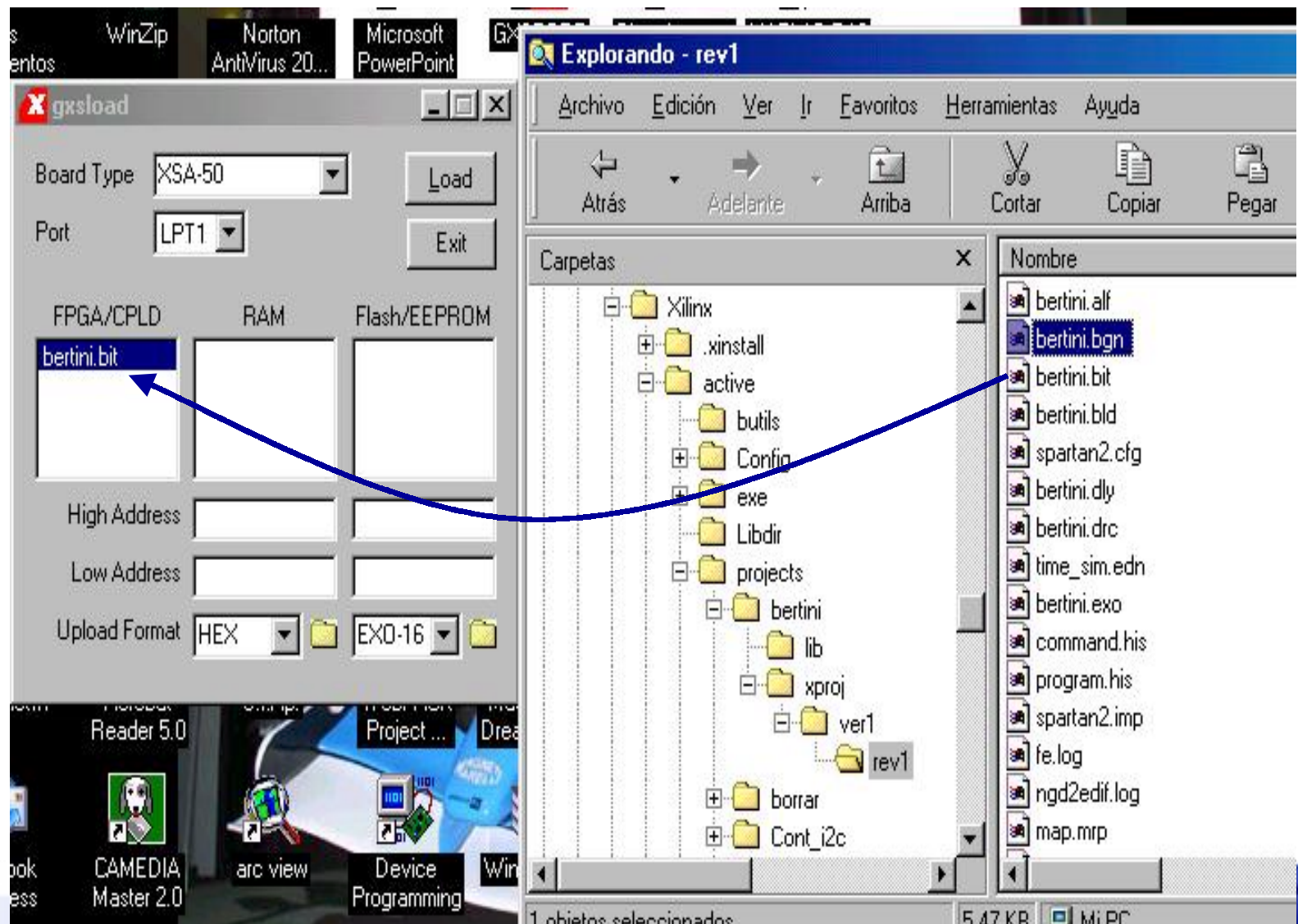


# Download a la FPGA



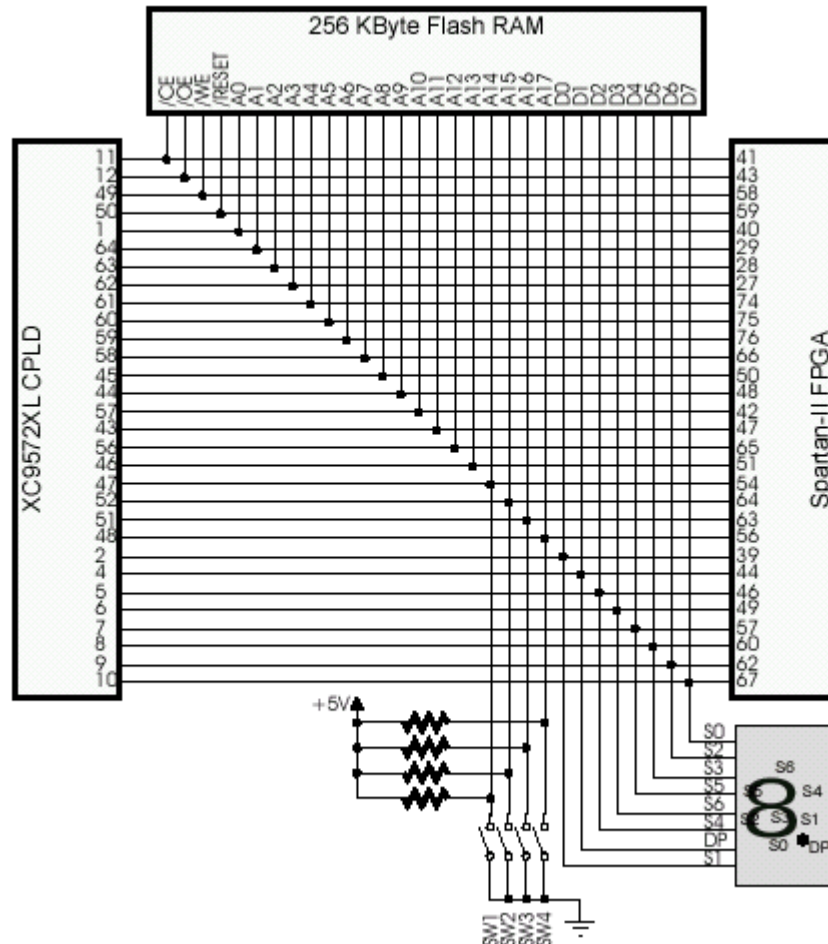
- La herramienta permite:
  - la configuración de la FPGA (carga *file.bit*) y/o del CPLD (*file.svf*)
  - cargar datos en la RAM
  - cargar el programa en la EEPROM (no volatil)
- Todas estas operaciones significan una preconfiguración del CPLD (que hace de interface entre el puerto paralelo y el dispositivo)

# Download a la FPGA



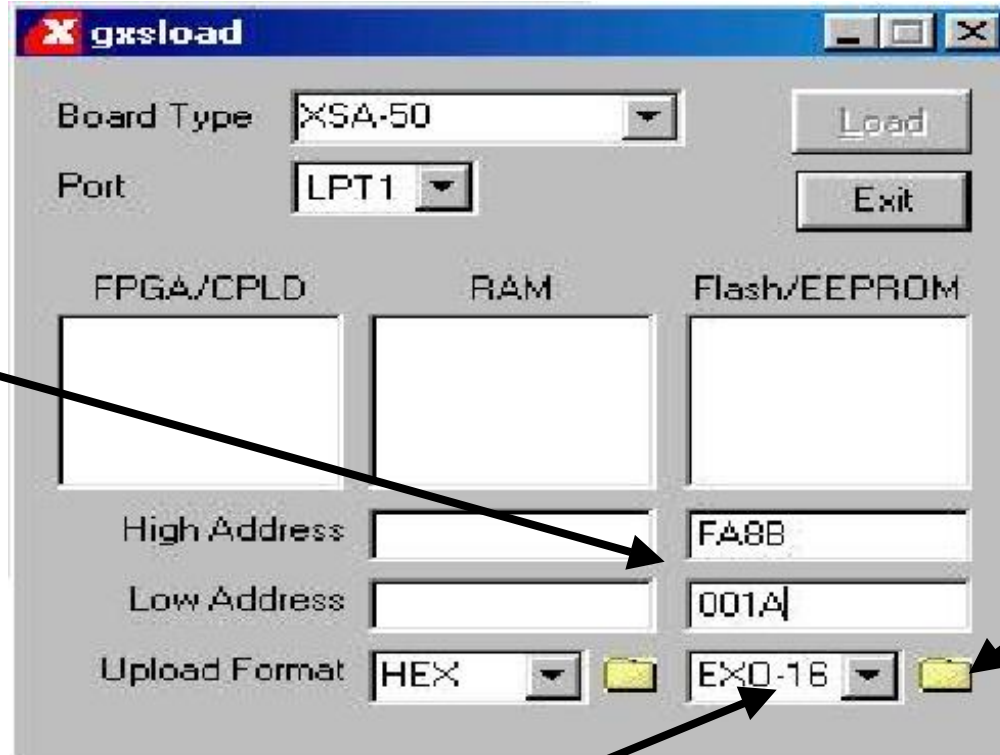
# Configuración de la EEPROM

- Para configurar la EEPROM
  - Verificar que los 4 Switches estén en OFF
  - Bajar archivos .EXO ó .MCS
  - el display de 7 segmentos se iluminará durante la fase de configuración



# Transferir datos de EEPROM a PC

Extremos superior e inferior de las direcciones



click  
acá

elegir la extensión

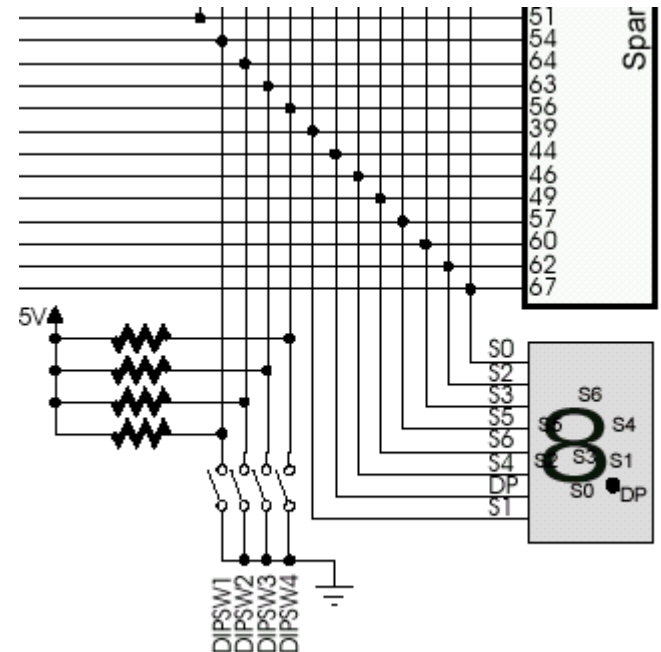
# Display

- El segmento se ilumina cuando entra una señal alta, según el siguiente esquema

| Función | Pin |
|---------|-----|
| S0      | 67  |
| S1      | 39  |
| S2      | 62  |
| S3      | 60  |
| S4      | 46  |
| S5      | 57  |
| S6      | 49  |
| punto   | 44  |

# Switches

- Los switches tienen conectada una resistencia pull up, es decir que entregan un valor alto cuando el switch está en OFF
- La señal de estos, es conducida a FPGA, CPLD, FLASH
- **Cuando no se utilizan es conveniente que los switches estén todos en OFF**

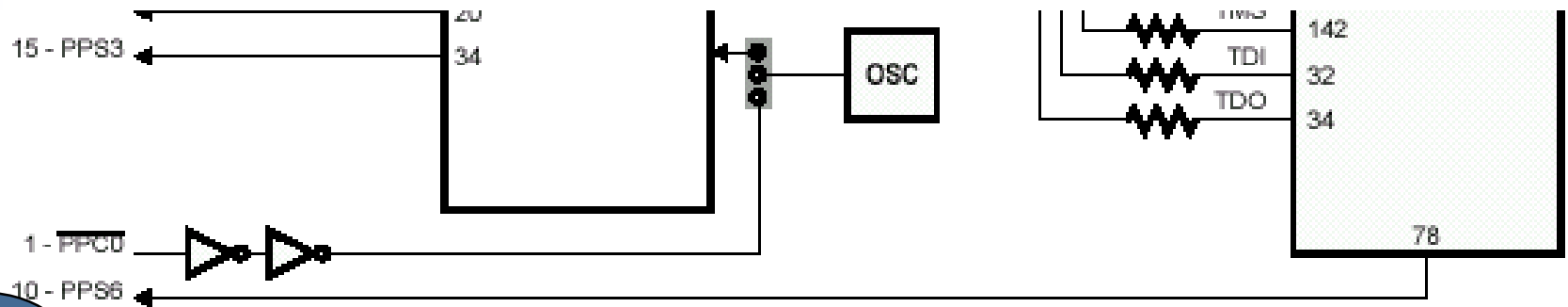


# La interface paralela

- Es el medio de comunicación entre la placa y la PC
  - A través de esta:
    - se configura el clock
    - se configura el CPLD
    - se configura la FPGA
    - se accede a las memorias Flash y SDRAM
  - Se configuran
    - 4 líneas de control C0-C3
    - 8 líneas de dato D0-D7
    - 5 líneas de estado S3-S6
- El CPLD funciona de interface y debe ser configurado convenientemente.

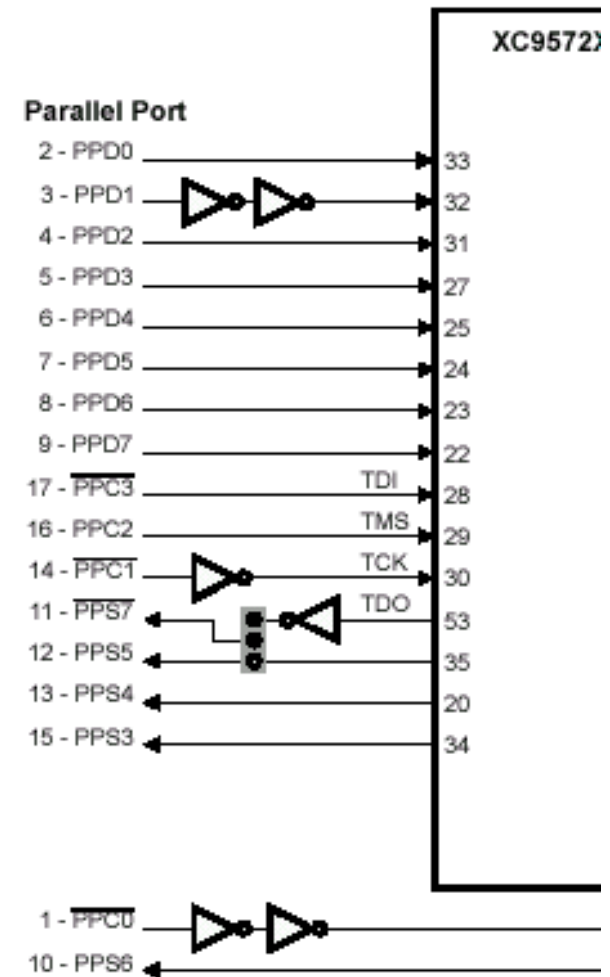
# La interface paralela

- Detalle:
  - C0 es usada para la configuración del oscilador
    - los dos buffer 74LS14 son “trigger Schmitt” ( buffer con histéresis)y son empleados para “limpiar” la señal de eventuales ruidos
  - S6 (linea de estado) es usada como línea directa de comunicación desde la FPGA a la PC
  - Las restante 15 lineas las maneja el CPLD como interface



# La interface paralela

- Detalle:
  - C1,C2,C3 (TCK,TMS,TDI) sirven para configurar el CPLD con el protocolo JTAG (*Join Test Action Group*)
    - “IEEE Std. 1149.1-1990 Test Access Port and Boundary-Scan Architecture”
  - S7 es conectada a TDO (comunicación del CPLD a la PC)
    - tanto TCK como TDO son “bufferadas”
  - Las restantes líneas son de propósito general y serán empleadas para transferir datos entre la FPGA y la PC, con el CPLD actuando como interface.

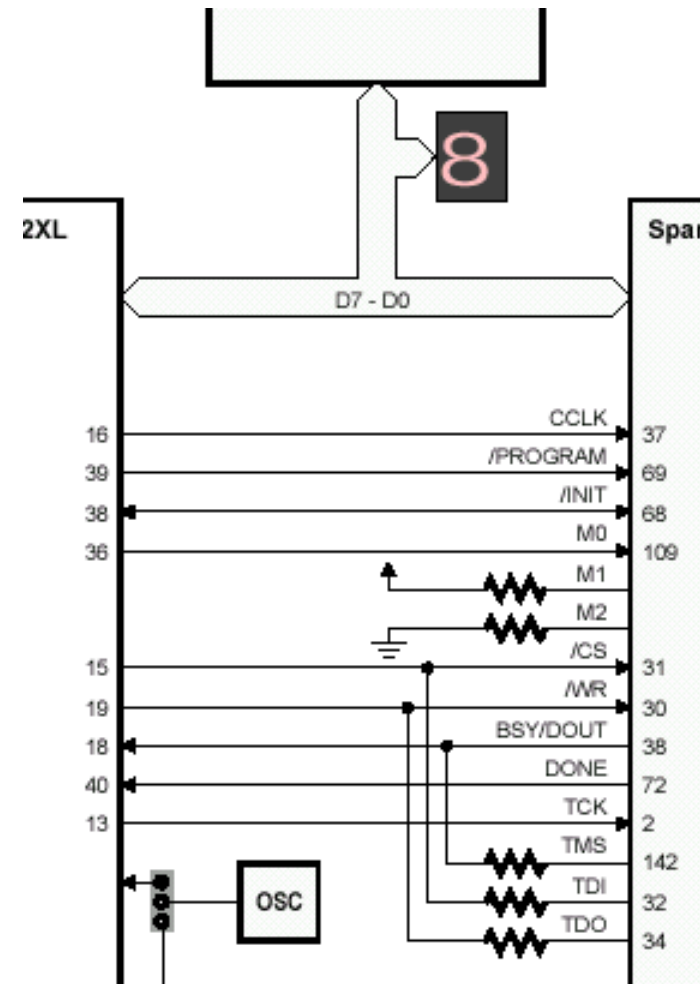


# La interface paralela

- El CPLD será programado como interface paralelo de la FPGA .
  - el archivo **downloadpar.svf** es usado para esto (viene con el sistema XESS)
- la linea D1 es convenientemente bufferada. Esta linea puede ser usado para ingresar un eventual clock externo y ser usado por cualquier máquina de estado programada dentro del CPLD.

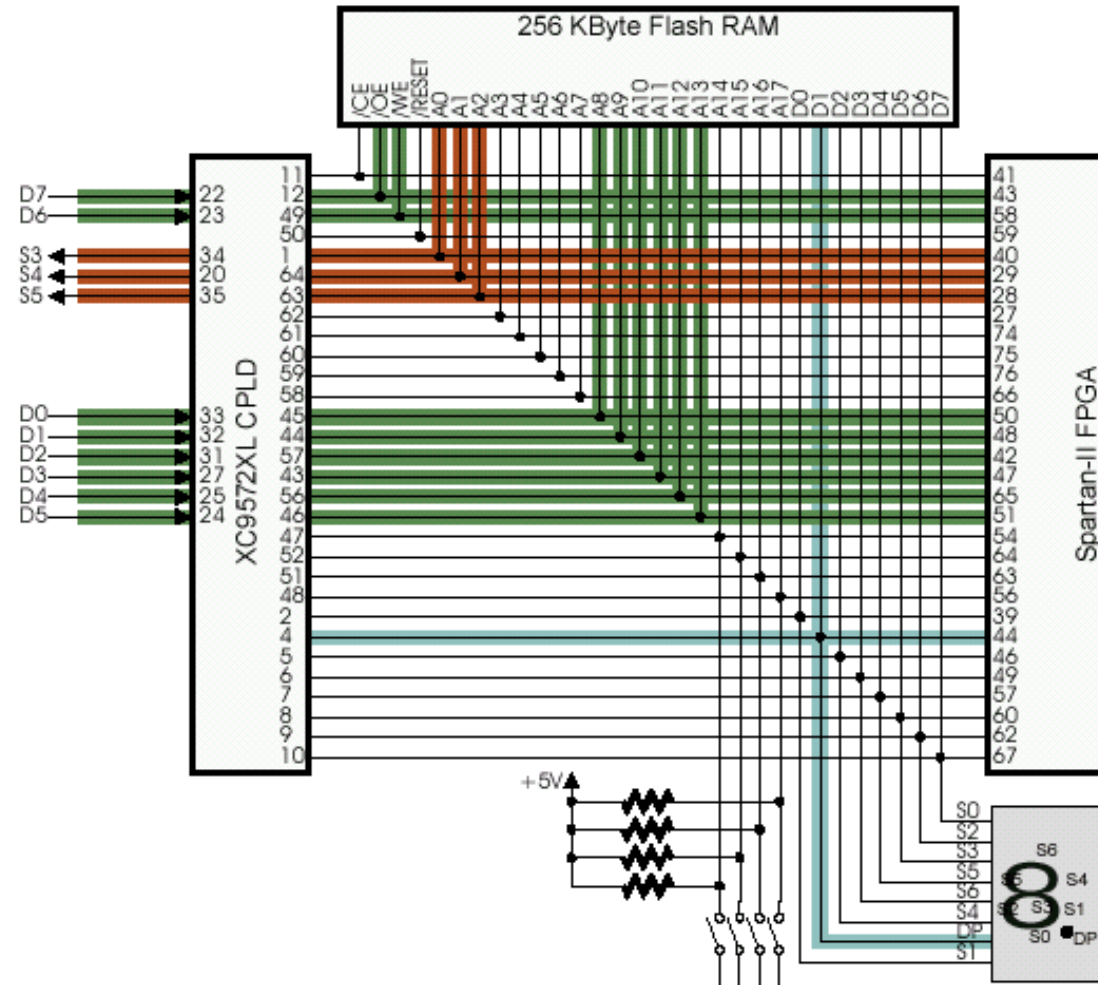
# La interface paralela

- El dato viene configurado sobre un bus de 8 bit, que “interfacea” el CPLD a la FPGA
  - (este bus es común para la Flash y el Display)
- M1 y M2 conectan a VCC y GND
- M0 configura
  - slave serial
  - master select (SelectMAP - slave parallel)
- El CPLD accede a la señal JTAG (TCK, TMS, TDI, TDO) de la FPGA (para eventual testeo o para la programación con la aplicación de software JTAG Programmer de Xilinx)



# La interface paralela

- Cuando la FPGA ha sido configurada, la señal DONE cambia a nivel alto, entonces:
  - el CPLD cambia su función y se vuelve “transparente” entre el puerto y la FPGA
  - Datos (D's) a la FPGA y señal de estados (S's) hacia la PC



# La interface paralela

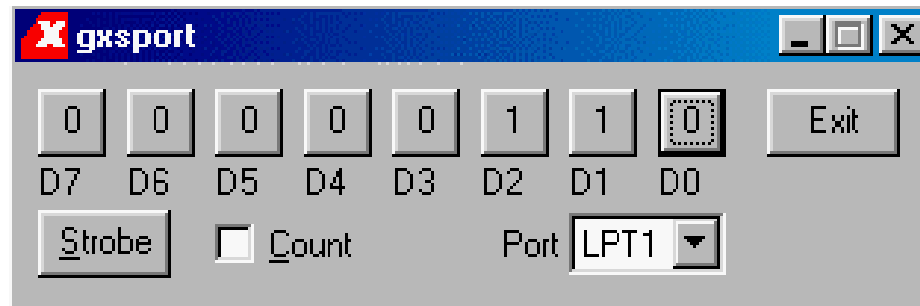
Utilización de la herramienta GXSPORT



GXSPORT.lnk

# La interface paralela

- Se puede emplear esta herramienta de Xess para bajar datos desde la PC a la FPGA, seleccionando los valores, unos y ceros, en el teclado de abajo



# Otras configuraciones del CPLD

---

- Emulación del cable JTAG de XILINX
- Escritura de datos sobre la Flash
- Sistema de control para leer datos de la flash y proveerlos a la FPGA (configuración como esclavo paralelo)

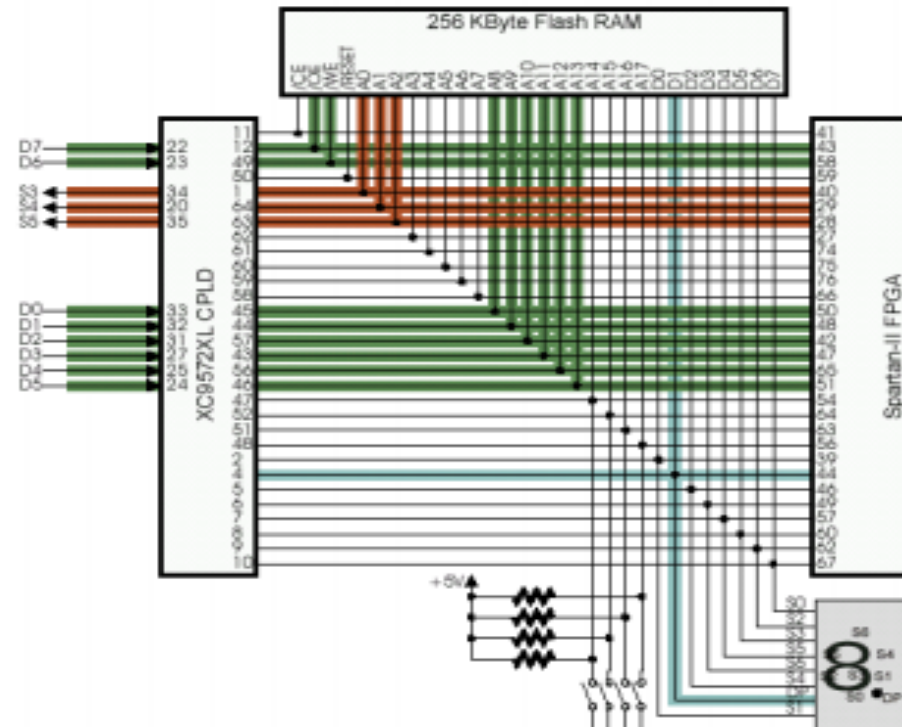
# La interface paralela

## PRECAUCIÓN 1

- La FPGA envía datos al PP a través de S3,S4,S5 (pins 40,29,28)
- La FPGA recibe señal del PP a través de D0-D7 (pins 50,48,42,47,65,51,43,58) por lo tanto estos pines

**NO DEBEN SER CONFIGURADOS COMO SALIDAS !!**

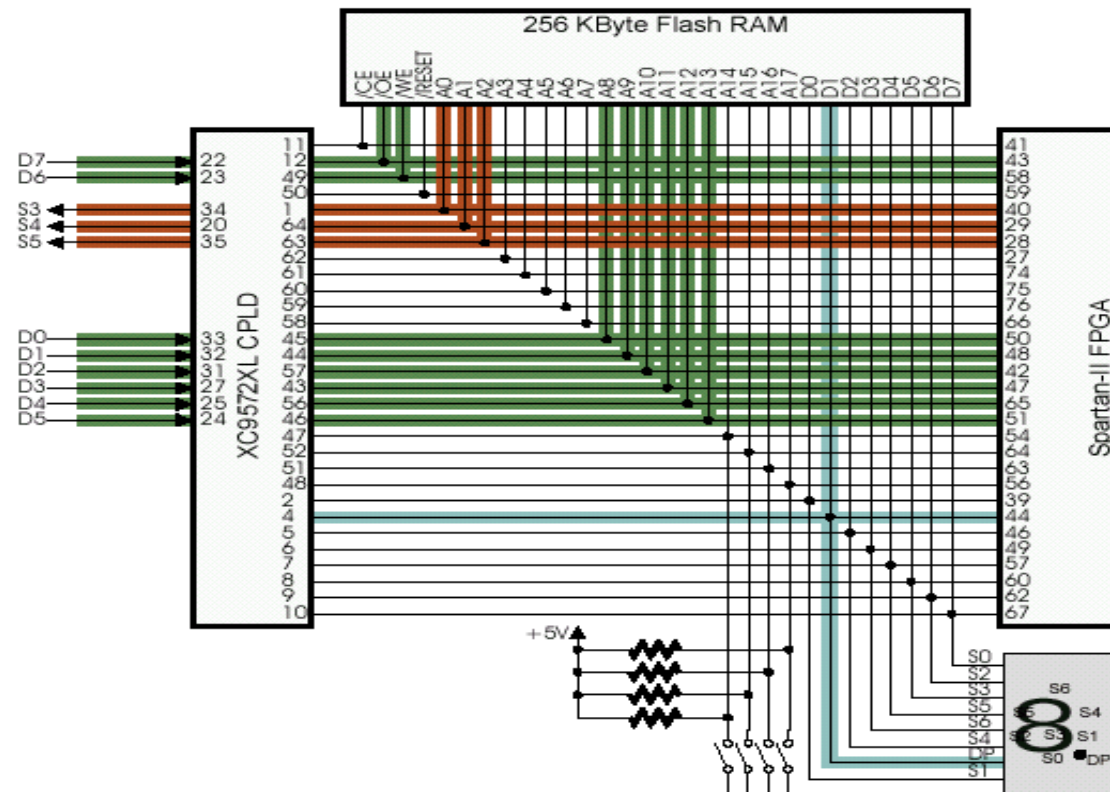
**NOTA:** La FPGA usa estos pines para acceder a la FLASH, pero en este caso pone un nivel bajo en CE (pin 41) y el CPLD responde liberando la linea (pone alta impedancia)



# La interface paralela

## PRECAUCIÓN 2

- La línea D7 del PP es usada tambien para manejar la linea **PROG** de la FPGA
- Por lo tanto si se “baja” esta linea se obtiene ua “**desprogramación**” de la FPGA

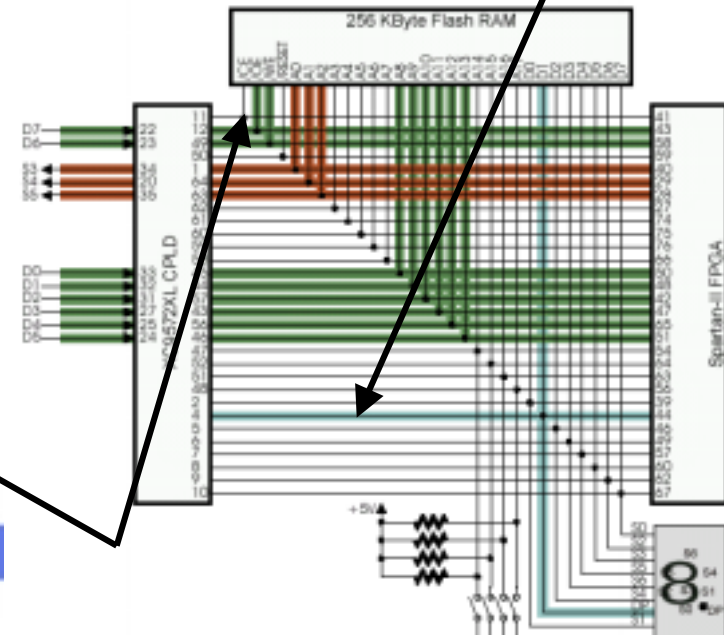


# La interface paralela

## PRECAUCIÓN 3

- El CPLD señala con un punto decimal sobre el display cuando una configuración válida se ha bajado sobre la FPGA
- Por lo tanto el pin 44 de la FPGA (que es usado para esto),  
**NO DEBE CONFIGURARSE COMO SALIDA**  
**(DAÑO SOBRE LA FPGA O CPLD)**

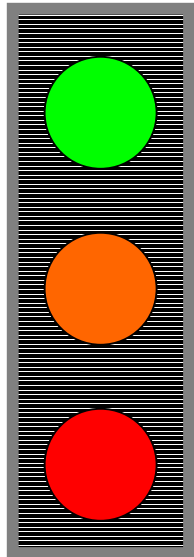
**NOTA:** La FPGA usa este pin para acceder a FLASH, pero en este caso, al poner un nivel bajo en CE (pin 41) el CPLD responde, liberando la línea



# Conexionado usado

## PRECAUCION 4

El manual provisto por XESS marca:



**Con 1 asterisco los pines que pueden utilizarse con Propósito General**

**Con 2 asteriscos aquellos que se pueden utilizar sólo despues de una conveniente RECONFIGURACIÓN del CPLD**

**Sin ningún asterisco  
NO PUEDEN SER UTILIZADOS**

# Pines libres

---

Se sugiere usar los siguientes pines

**77, 78, 79, 80, 83, 84, 85, 86, 87**

# Alimentación

- Usar una alimentación de 9 V con el **POSITIVO al centro**. La placa genera las voltages necesarios.
- Se manipulan “Jumpers” si la alimentación es externa

| Voltage | Pin | Note                                                                                                                                                          |
|---------|-----|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| +5V     | 2   |                                                                                                                                                               |
| +3.3V   | 22  | Remove the shunt from jumper J7 if you wish to use your own +3.3V supply.<br>Leave the shunt on jumper J7 to generate the +3.3V supply from the +5V supply.   |
| +2.5V   | 54  | Remove the shunt from jumper J2 if you wish to use your own +2.5V supply.<br>Leave the shunt on jumper J2 to generate the +2.5V supply from the +3.3V supply. |
| GND     | 52  |                                                                                                                                                               |

