

---

# Diseño de Circuitos y Sistemas Electrónicos

## Práctica 2: Implementación de la MINICPU en XILINX ISE

Noviembre 2006



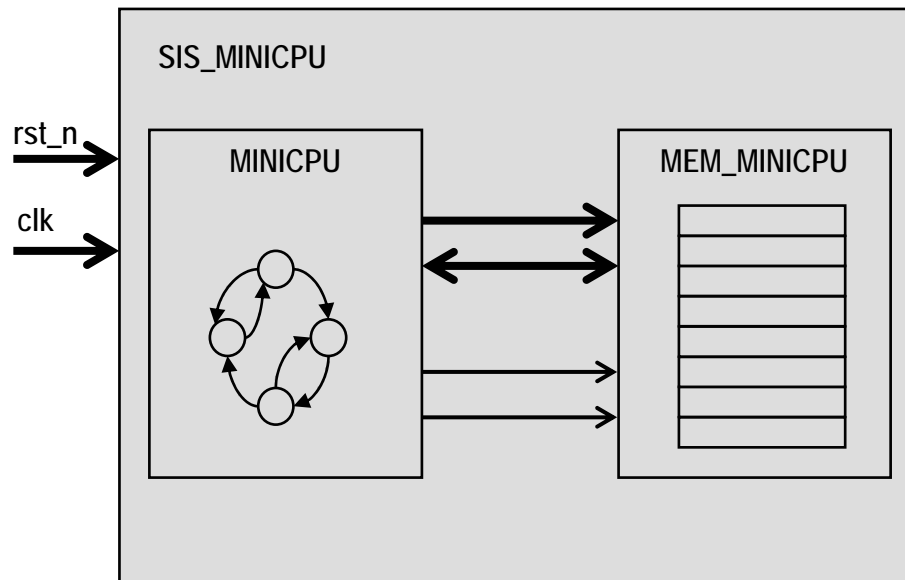
UNIVERSIDAD DE LAS PALMAS DE GRAN CANARIA  
Departamento de Ingeniería Electrónica  
y Automática

Pedro P. Carballo  
*carballo@iuma.ulpgc.es*

Jorge Monagas  
*jmonagas@diea.ulpgc.es*

# Implementación en Xilinx (1)

- Con objeto de realizar la implementación en Xilinx del diseño CPUMINI es necesario:
  1. Modificar el diseño para:
    1. incluir la memoria de programa/datos como una memoria de bloque (BRAM)
    2. Inferir una máquina de estados para implementar la unidad de control



## Implementación en Xilinx (2)

---

- Con objeto de realizar la implementación en Xilinx del diseño CPUMINI es necesario:
  2. Especificar las restricciones correspondientes para la familia SPARTAN-III (dispositivo xc3s1000):
    1. Buffer de reloj
    2. Ubicación de las unidades funcionales (*Floorplan*)
    3. Especificación de E/S para el estándar utilizado (LVTTTL, LVCMOS, etc)
    4. Inicialización de bloques de memoria
    5. Bloques específicos de reset y de programación.

## A realizar (1)

---

1. A partir de la descripción disponible en el servidor web de la MINICPU:
  - a. Modificar la descripción del procesador para que se infiera la utilización de una máquina de estados
  - b. Crear una descripción VHDL para la memoria de programa/datos que infiera y/o instancie una memoria de bloques para Spartan-II y que se pueda inicializar con el programa del MCD desarrollado. La entidad se llamará MEM\_MINICPU
  - c. Crear la descripción del sistema que integre al procesador y a la memoria (SIS\_MINICPU)
  - d. Simular el sistema procesador-memoria utilizando como testbench el programa del Máximo Común Divisor (MCD) indicado.
  - e. Implementar el sistema procesador-memoria en Xilinx ISE incluyendo las restricciones indicadas

## A realizar (y 2.)

---

2. Presentar una memoria en formato Word, que incluya los siguientes apartados:
  1. Introducción
  2. Descripción de la arquitectura
  3. Modelado VHDL
  4. Verificación
  5. Implementación
  6. Resultados
  7. Anexos
3. Fecha de entrega: indicado en el servidor web
4. Modo de entrega: Carga de fichero zip en el servidor de la asignatura.