Proyecto Optativo 1: Gestión de riesgos en MIPS segmentado

Javier Resano, José Luis Briz

GAZ: Grupo de Arquitectura de Computadores UZ

Departamento de
Informática e Ingeniería
de Sistemas

Universidad Zaragoza

Objetivos

- Entender cómo funciona un procesador segmentado y cómo interactúa con memoria y con la entrada/salida
- Aprender a utilizar un entorno de desarrollo profesional
- Adquirir los conceptos básicos de depuración y verificación

Fase I: Preparación

- Entender cómo funciona el Mips segmentado
  - Instalar el simulador, tener los fuentes VHDL a mano, y estudiar los tres videotutoriales que hay en Moodle
- Comprobar que funciona el Mips base. La ISA es retardada:
  - El predictor de saltos está vacío
    - Predice siempre NT = salto 1-retardado!
    - Nadie comprueba las predicciones = No se salta nunca!
  - Uds. de Detec. de Riesgos y Anticipación vacías
    - Todas las productoras retardadas!

Fase II: calentamiento

- Añadir instrucciones **bne** y **`la`**
  - Modificar la UC para gestionar las nuevas instrucciones
  - 1) BNE
  - 2) Load Address:

    la rt, inmed(rs):

    rt ← rs + SignExt(inmed), PC ← PC + 4

Fase III: **Gestión de riesgos de datos**

- Mejorar el rendimiento con anticipación de operandos
  - Diseñar la Unidad de Anticipación (UA)
  - Diseñar el control que detiene el procesador cuando detecta un riesgo

Fase 4: **Gestión de riesgos de control**

- Gestionar el predictor de saltos en IF
  - Usar su salida para actualizar PC
  - Corregir sus errores y actualizar su información

```
entity branch_predictor is Port (
clk : in  STD_LOGIC;
   reset : in  STD_LOGIC;
-- Puerto de lectura
   PC4 : in  STD_LOGIC_VECTOR (7 downto 0);
   branch_address_out : out STD_LOGIC_VECTOR (31 downto 0); --
    prediction_out : out  STD_LOGIC; -- indica si hay que saltar
-- Puerto de escritura
   PC4_ID:  in  STD_LOGIC_VECTOR (7 downto 0);
    branch_address_in : in  STD_LOGIC_VECTOR (31 downto 0);
prediction_in : in  STD_LOGIC;
                  update:  in  STD_LOGIC);
end branch_predictor;
```

Bancos de prueba

- Os damos un ejemplo de un banco de prueba
- Debéis diseñar vuestros propios bancos:
  - Sois responsables de que vuestro diseño funcione correctamente
  - el diseñador debe ser el primer testeador.
  - Cada anticipación, cada parada, o cada caso de predicción debe probarse al menos una vez
  - Os recomendamos comprobar cada paso antes de pasar al siguiente

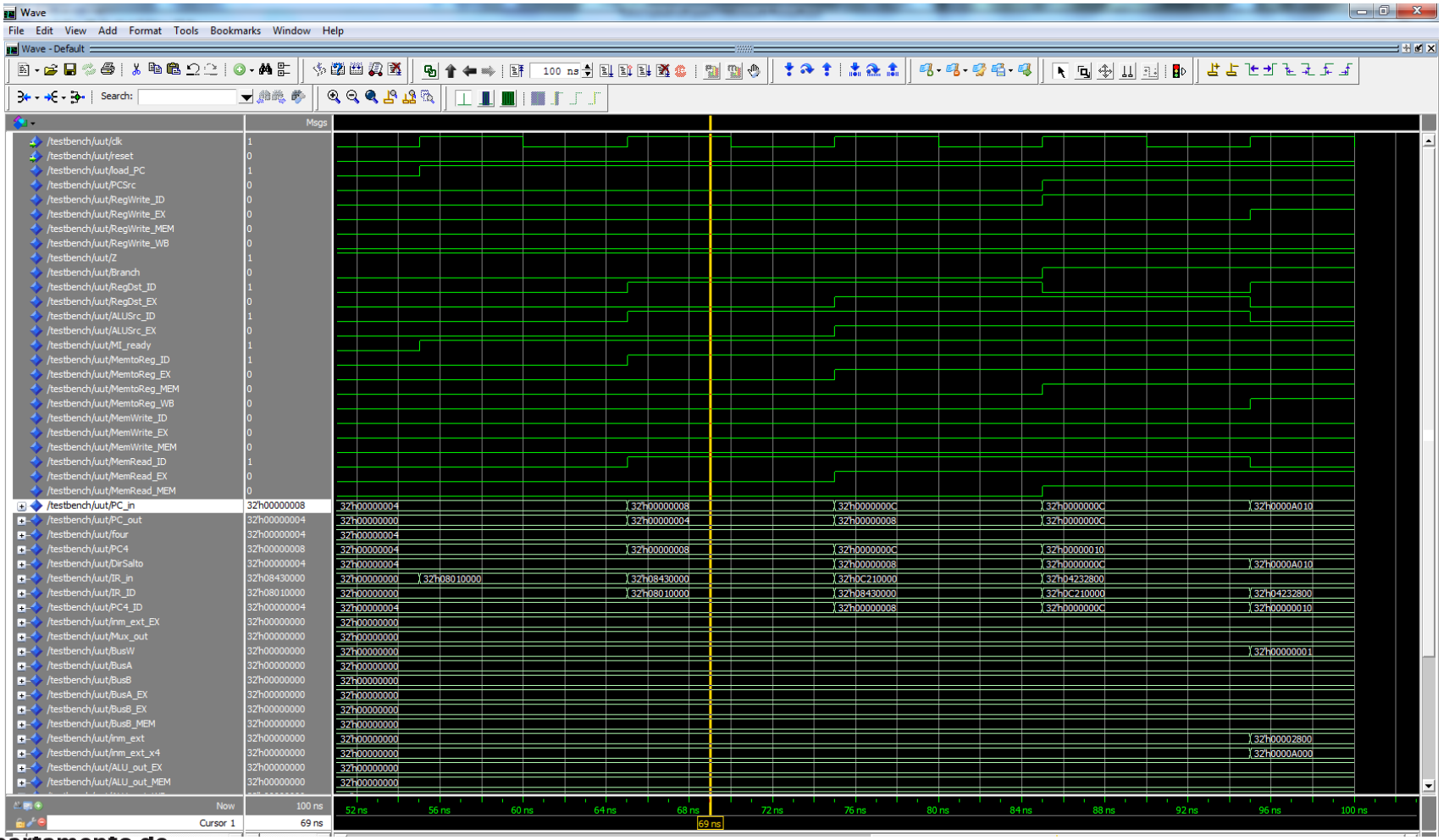
Memoria

- Escribir la memoria con **todas** las partes indicadas en el guión
  - Se puede ir escribiendo mientras se va desarrollando el proyecto
- Miramos y valoramos:
  - Redacción cuidada y concisa
  - Esquemas claros
  - Código de pruebas explicado /comentado
    - Podéis acompañar capturas de pantalla del simulador para aclarar las explicaciones

Entorno de trabajo

- Se puede utilizar cualquier simulador de VHDL
- Recomendamos utilizar Modelsim:
  - http://www.mentor.com/company/higher_ed/modelsim-student-edition
  - Hay que descargarlo, instalarlo y solicitar licencia. Nota: utilizad el mail de la universidad porque sólo es gratuito para estudiantes.
- Necesitaréis también un buen editor de texto

Entorno de trabajo



Defensa del trabajo realizado

- **Entrega 26 de Abril**
- Se entrega a través de Moodle
  - Antes del deadline
  - Comprobad que están todos los ficheros y que todo funciona
- Tras la entrega comprobamos el funcionamiento
  - Si no funciona con el caso base ni con vuestras propias pruebas, el trabajo está suspenso y no se pasa al 2º Proyecto
  - Si no funciona con nuestro testbench interno pero habéis hecho unas pruebas razonables que sí funcionan permitimos subsanar errores en un plazo de uno o dos días

Trabajo en grupo

- El trabajo se puede realizar en parejas
- Antes de formar pareja aseguraos de:
  - Tener horarios compatibles para poder trabajar juntos
  - Tener objetivos académicos similares
- Tras la entrega habrá entrevistas personales:
  - A la entrevista deben acudir los dos miembros del grupo
  - Las preguntas serán individuales
  - Si uno de los estudiantes no es capaz de explicar su diseño y contestar las preguntas del profesor **suspenden los dos miembros de la pareja.**

Carga de trabajo

- Esta es nuestra estimación:
  - Paso 1 y toma de contacto con el simulador: 4 horas
  - Diseño del paso 2: 1 hora
  - Diseño del paso 3: 1 hora
  - Diseño del paso 4: 1 hora
  - **Depuración y ajustes: 10 horas**
  - Memoria: 3 horas