

Trabajo Práctico Promocional

Grupo: 3

Integrantes: Francisco Manuel Rodríguez Herrero, Jeremias Souto

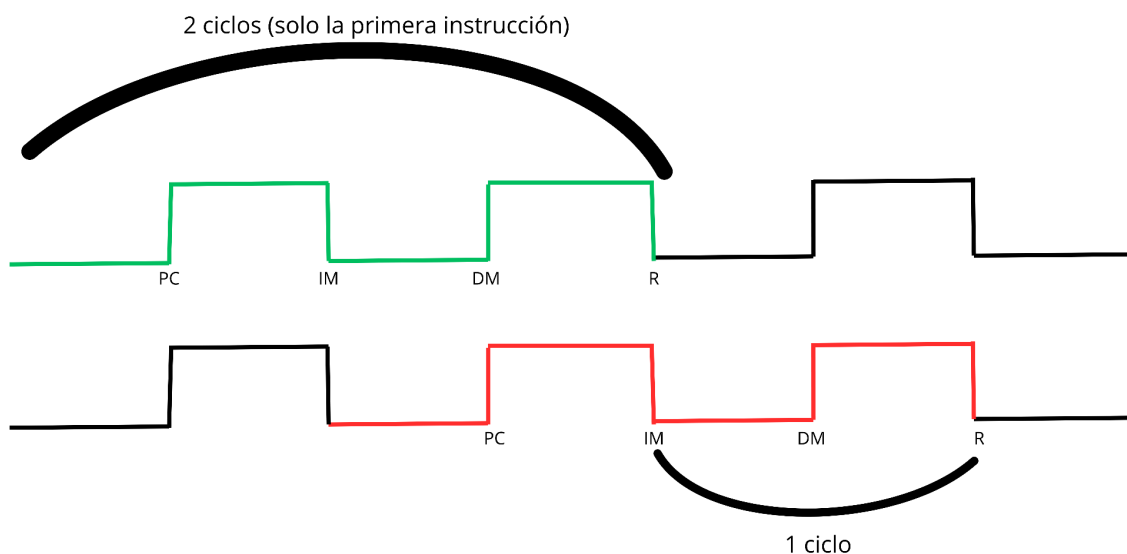
¿Cómo llevamos a cabo el desarrollo del procesador MIPS?

Iniciamos el desarrollo creando un archivo denominado “*entities.vhd*”, en el cual implementamos los componentes faltantes del procesador MIPS, diseñando su respectiva lógica. Una vez finalizada esta etapa, procedimos a la integración con el procesador. Durante este proceso, identificamos que algunos componentes previamente implementados resultaban innecesarios o podían definirse de manera implícita. Además, detectamos errores de sintaxis en ciertas secciones. Esto nos llevó a realizar ajustes y redefinir diversos aspectos del diseño.

Posteriormente, llevamos a cabo simulaciones para verificar el correcto funcionamiento del sistema. En esta fase, se presentaron múltiples errores, como operaciones inválidas, errores de sintaxis y asignaciones incorrectas de datos, entre otros, los cuales fueron corregidos conforme se identificaban. Finalmente, el procesador logró ejecutarse de manera exitosa, obteniendo los resultados esperados.

¿Por qué la Memoria de Programa y el Banco de Registros trabajan en flanco descendente, y la Memoria de Datos y Program Counter (PC) en flanco ascendente?

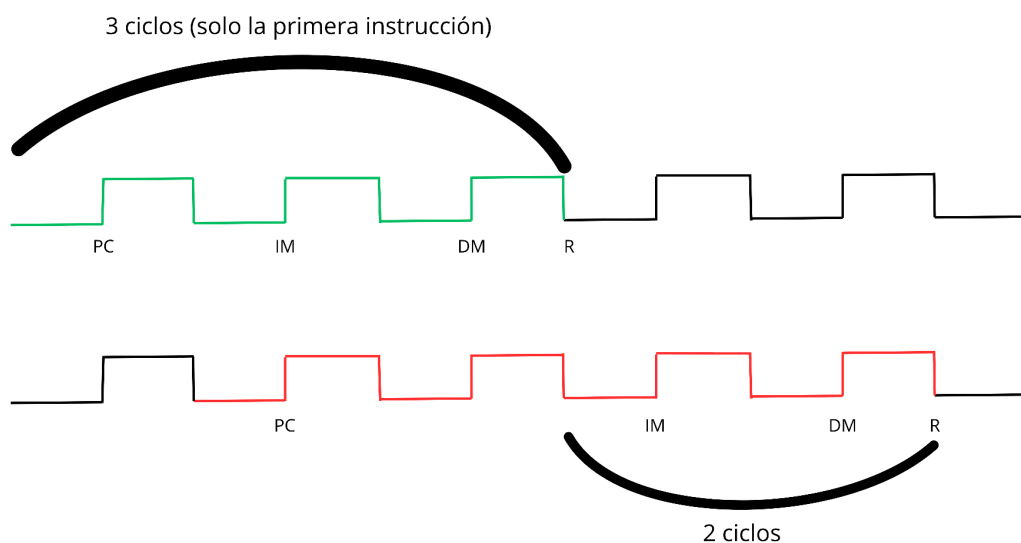
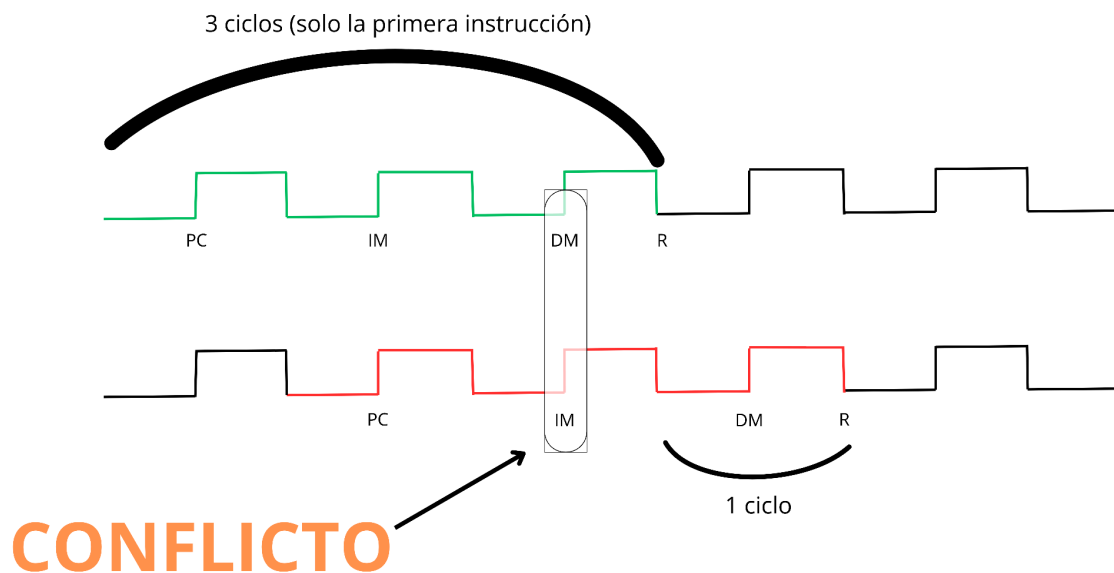
Ejemplo del MIPS con PC y Memoria de Datos en flanco ascendente, y Banco de registro y Memoria de Programa en flanco descendente:



En nuestro análisis, llegamos a la conclusión de que esta implementación fue elegida para reducir el CPI y que sea igual a uno. Instrucciones como las LW y SW deben pasar por los cuatro componentes. Su recorrido es: primero pasan por el PC; segundo por la Memoria de Programa; tercero por el Banco de Registros (lectura solamente); cuarto por la Memoria de Datos; por último, el Banco de Registros (escritura).

El tercer paso lo eliminamos de nuestro análisis, debido a que el Banco de Registros lee en todo momento sin importar los flancos de reloj. Por lo tanto, el camino restante es de cuatro pasos. En cada paso se alterna el flanco. (Paso 1: ascendente, Paso 2: descendente, Paso 3: ascendente, Paso 4: descendente)

Ejemplo del MIPS con PC, Memoria de Programa y Memoria de Datos en flanco ascendente, y Banco de registro en flanco descendente:



Problemas con mismos flancos:

Si dos pasos consecutivos contasen con el mismo flanco de reloj, perderíamos un ciclo, ya que es necesario esperar a que una instrucción termine para poder comenzar con la siguiente.

En caso de no esperar, aparecerían conflictos. Si quisiéramos ejecutar una instrucción por ciclo, dos instrucciones se “ejecutarían” simultáneamente. Al no ser un procesador segmentado, esto no es posible. Los datos serían incontrolables y es imposible asegurar el buen funcionamiento del procesador, a menos que, cada instrucción se ejecutase cada dos ciclos esperando a que la otra termine, para poder acceder a la Memoria de Programa. Esta solución agregaría lógica combinacional, ocuparía un ciclo más y no sería como el uniciclo estudiado en clase.