Reconfiguración Dinámica: Fundamentos y Métodos

Miguel A. Aguirre

Departamento de Ingeniería Electrónica

Escuela Superior de Ingenieros

Universidad de Sevilla

INDICE

- Motivación
- Introducción a la RD
- Ganancias sobre el uso de RD
- Sobrecostes de Diseño
- Conclusiones

Motivación

- Hacer un recorrido sobre los aspectos más críticos a la hora de desarrollar un diseño, utilizando Reconfiguración Dinámica
- Explorar, con cierto grado de detalle, los diferentes factores que condicionan la solución de un diseño.

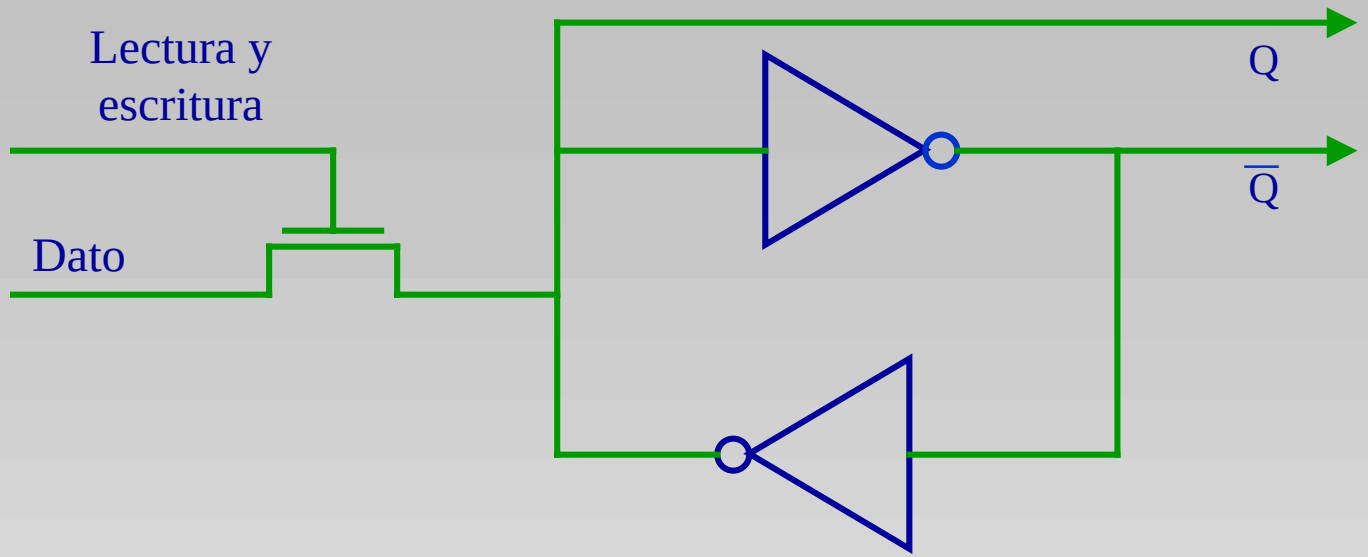
Sistemas Reconfigurables

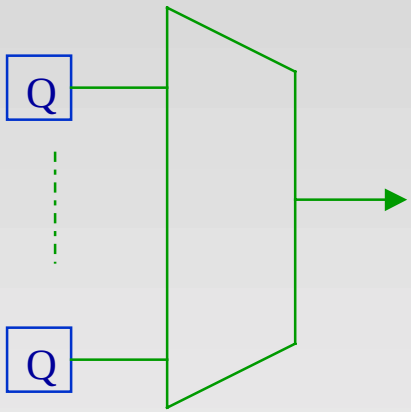
- El origen del concepto de FPGA es un concepto de dispositivo lógico que eliminara toda la circuitería digital tipo SSI que adaptaba los diferentes circuitos LSI.
- Debían tener mayor capacidad que las PLD's.
  - Reducción de área de PCB
  - Mejorar sus prestaciones

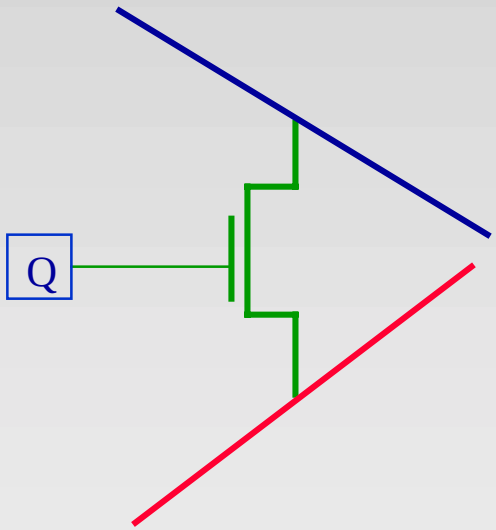
Mecanismos de programación de las primitivas FPGA's

- EEPROM: No volátil, reprogramable mediante un equipo especial, tecnología de puerta flotante.
- Antifusible: No volátil, programable una sola vez y mediante un equipo especial, tecnología ONO.
- SRAM: Volátil, programable mediante una memoria externa

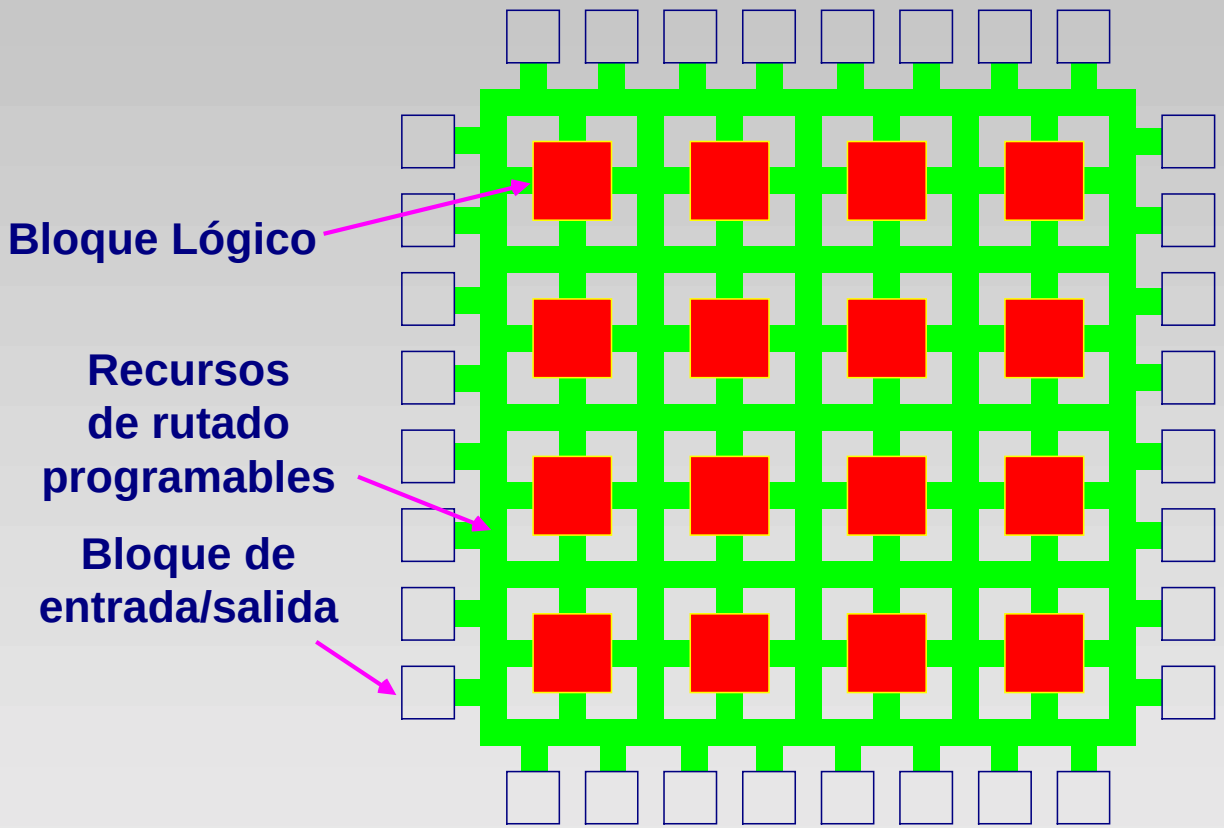
Mecanismo elemental SRAM







Arquitectura de las FPGA tipo isla



FPGA's SRAM

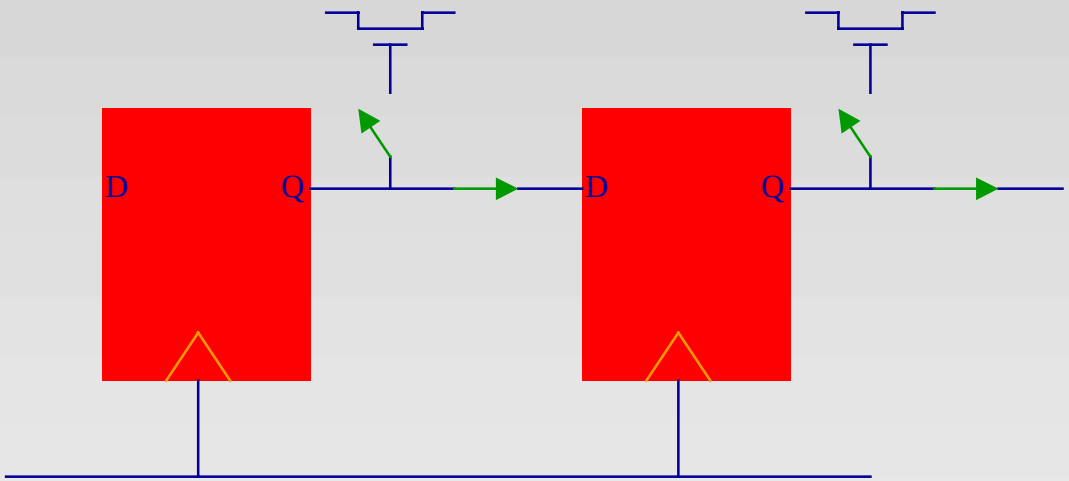
- Tecnología CMOS convencional
- Mayor área por punto de programación
- Independiente del fabricante: "Fabless"
- Programación volátil
- Programación Múltiple

Sin embargo se ha producido ...

- la natural evolución de las tecnologías de fabricación, que ha permitido incrementar la densidad de recursos
- un cambio en el concepto de dispositivo programable en el que el mecanismo de programación forma parte de la funcionalidad

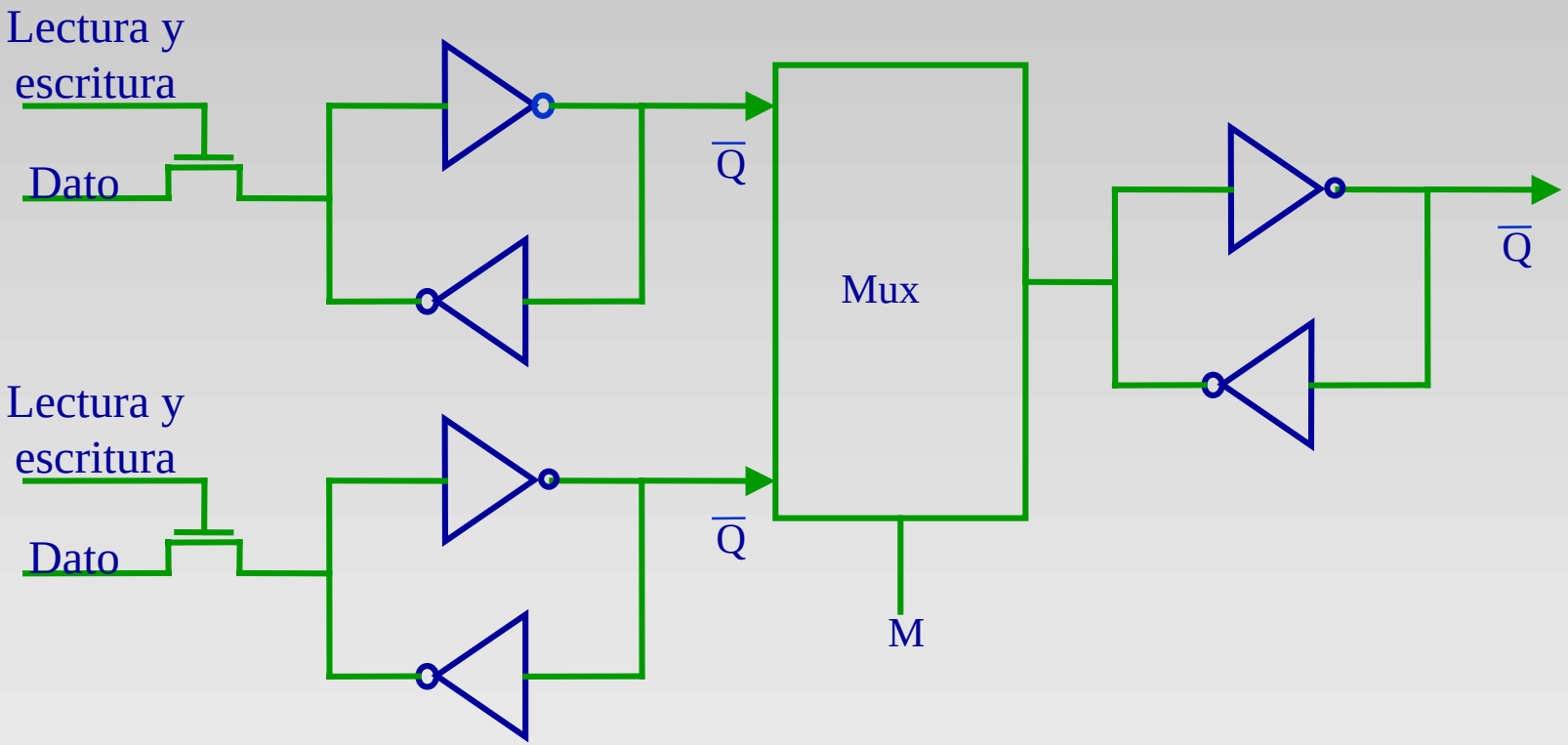
Esquemas de Programación

- Esquema serie monocontexto



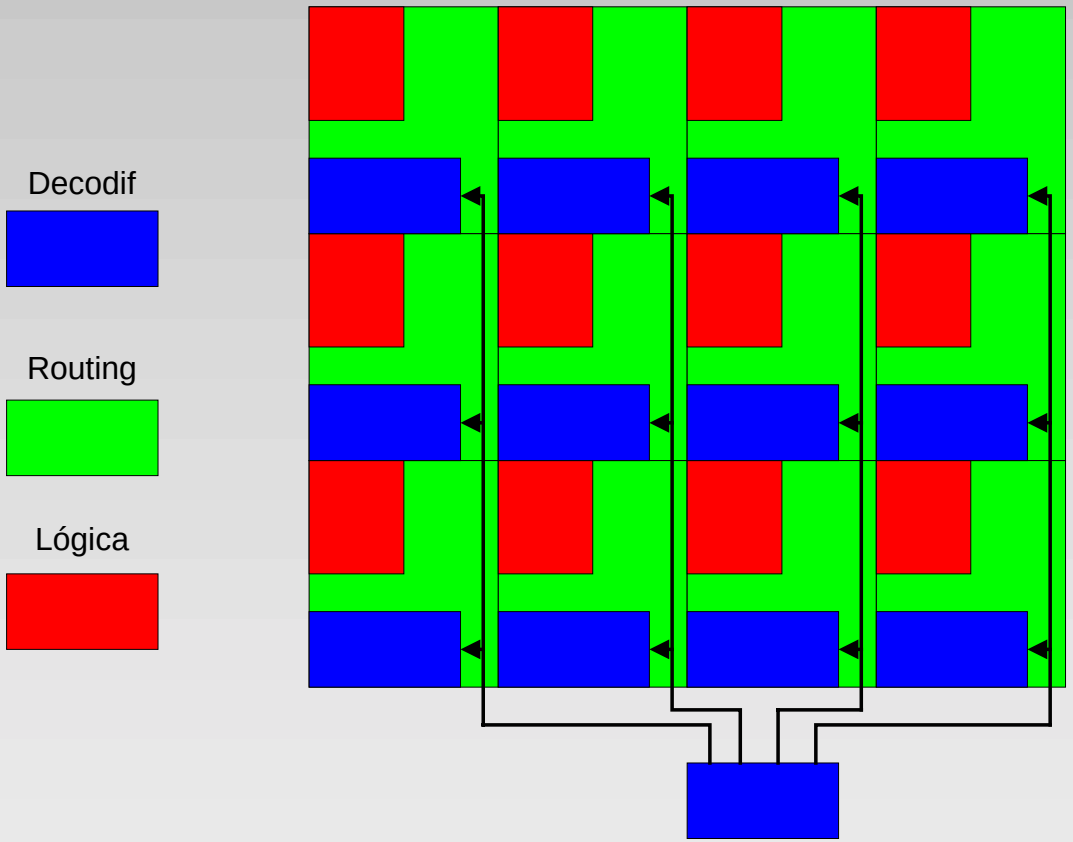
Esquemas de Programación

- Esquema multicontexto



Esquemas de Programación

- Esquema mapa de memoria



Reconfiguración Dinámica

- Reconfiguración Estática. La FPGA detiene su actividad para recibir una nueva configuración
- Reconfiguración Dinámica Parcial: Cambio en la configuración en tiempo de ejecución de de una sección de la FPGA
- Reconfiguración Dinámica Multicontexto: Cambio de configuración activa de la FPGA

No son conceptos excluyentes. En general todos los sistemas RDM son RDP.

FPGA's comerciales

- Xilinx:
  - RE: Series XC y XS
  - RDP: XCV y XC6200
- Atmel
  - Serie AT40K
- ALTERA
  - APEX20 ??

Mejoras introducidas por la RD

- Reusabilidad del Hardware
  - Los recursos no utilizados se pueden reutilizar para soportar nuevas tareas
- Hardware adaptativo
  - Se permite la modificación del hardware de manera que se adapte a las nuevas condiciones del sistema.

Desde el punto de vista de producto

- Podemos actualizar – mejorar – depurar la parte flexible de un hardware
  - Ej: Sistemas multimedia que se reprograma con un nuevo algoritmo de compresión
- Adaptar a diferentes entornos y normativas
  - Ej: Teléfono móvil que puede ser adaptado a diferentes normas.
- Sistema multi-aplicaciones
  - Ej: Controladores de motores para distintos usos
- El costo por puerta se reduce, y el tiempo de vida del producto se ve incrementado
- Optimización del consumo

Aplicaciones

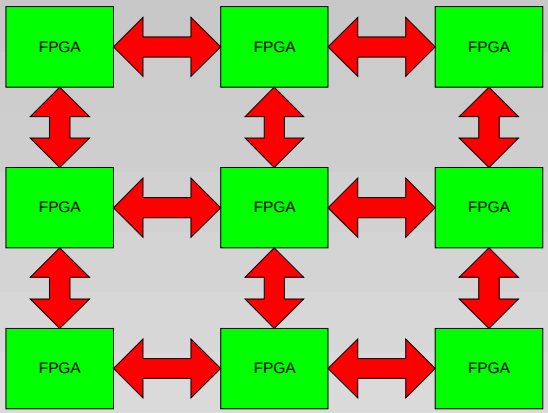
- Tratamiento de imágenes en visión artificial
- Cifrado. En sistemas de comunicación seguros, modificar el algoritmo de codificación/decodificación.
- Control adaptativo y autoajustable de sistemas.
- Sistemas multimedia.
- Sistemas híbridos de computación reconfigurable y codiseño.

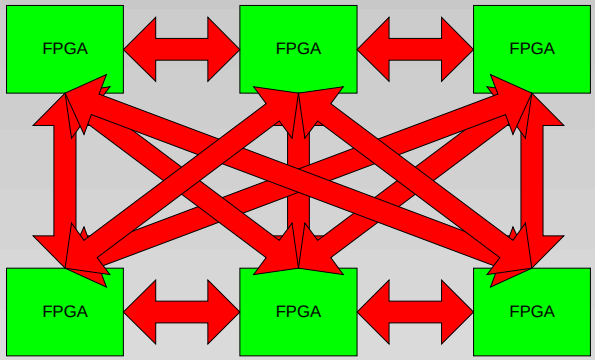
Sobrecostes (Overheads)

- De diseño: Tiempos de desarrollo y herramientas
- De arquitectura interna de los dispositivos
- De selección de operadores.
- De temporización del sistema global

Morfología del Sistema RD

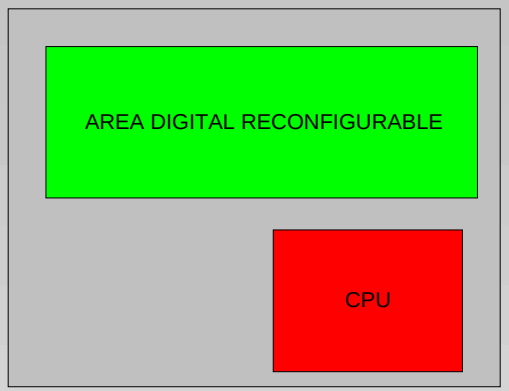
- Sistema programable autónomo
  - Multi FPGA

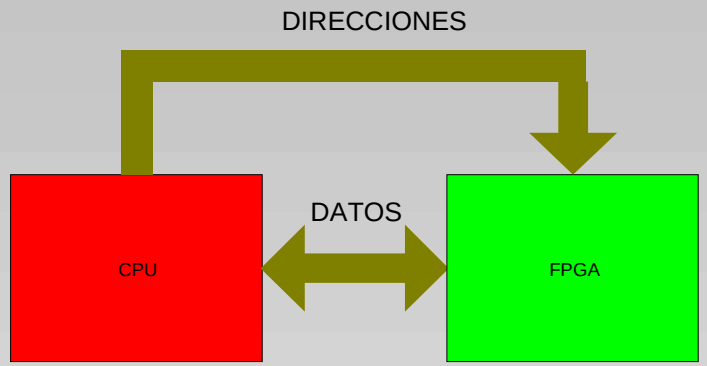


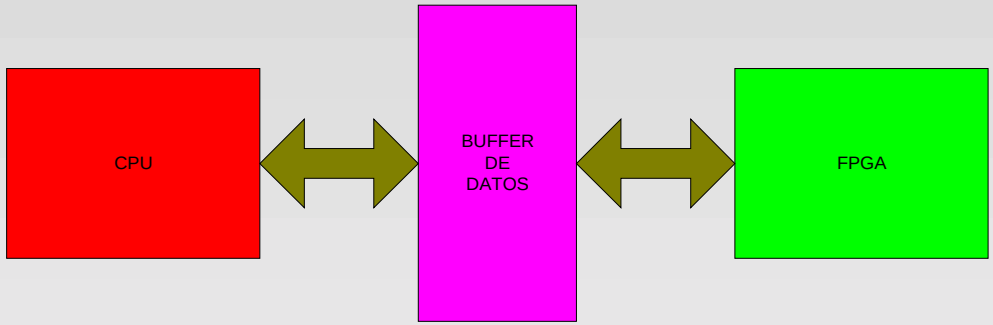


- Sistema híbrido Procesador-Hardware programable.
  - Depende del grado de interrelación entre ambos
  - Condiciona la estrategia de organización del sistema

Grados de interrelación en SH







Codiseño Hardware Software

- De las diferentes morfologías se induce un diferente problema de particionamiento.
- Las diferentes funciones hardware se activan a indicación del PS.
- La reconfiguración se realiza a través del PS
- En un estudio previo se deciden también las secuencias de las tareas

Sobrecostes (Overheads)

Entendemos como sobrecostes todos aquellos aspectos que hay que analizar para diseñar la estrategia de implementación de un sistema Dinámicamente Reconfigurado, tomando como referencia los aspectos análogos de un diseño no reconfigurable.

Sobrecoste en diseño

Diseño convencional con FPGA's:

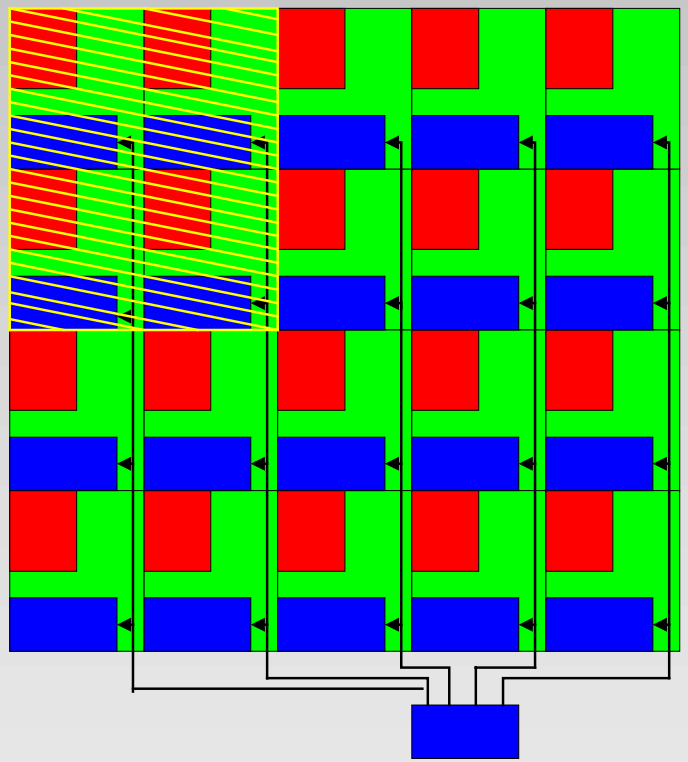
- Alto grado de Automatización
- Tiempos de desarrollo de prototipos muy cortos
- Herramientas muy depuradas con interfaces HM de fácil manejo.
- Se permite un bajo conocimiento del interior de la FPGA

Planificación espacial

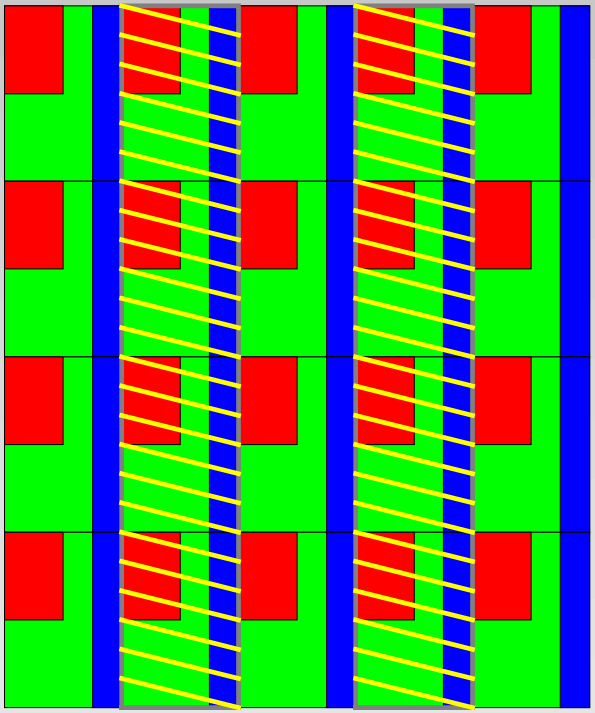
- Planificación de los recursos
  - Uso de recursos especiales: Memorias
- Organización del área que se va a asignar a cada tarea:
  - Condicionada por la arquitectura de reconfiguración
  - Condicionada por las asimetrías de la arquitectura del dispositivo
  - Condicionada por los puntos de conexión entre el área fija y el área modificada.

Floorplanning

Atmel

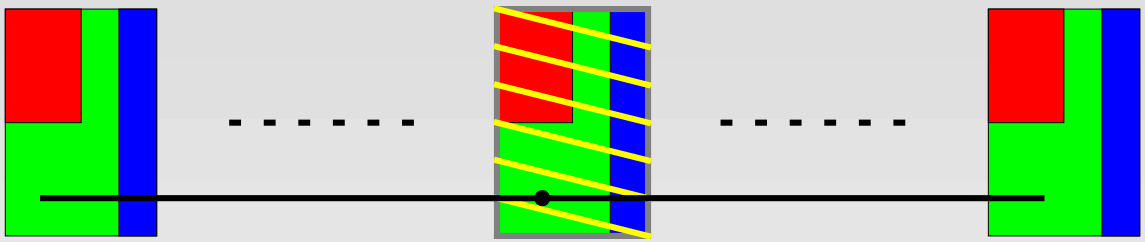


Xilinx-Virtex



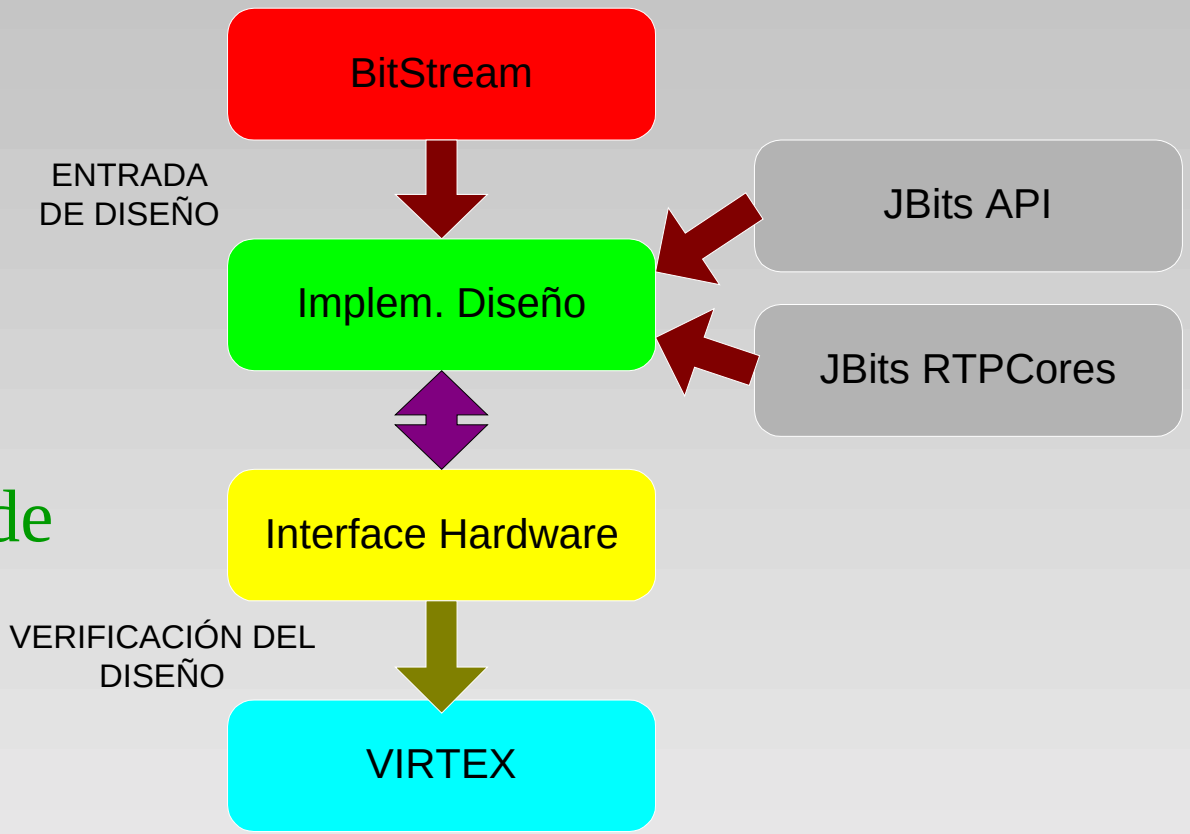
Conexionado

- Se deben cuidar la localización de los puntos de conexión con la parte fija.
  - Ej: Orientación de las entradas-salidas
- Considerar las asimetrías del conexionado
  - Ej: Las líneas de alcance 6 CLBS (Hex lines) de Virtex

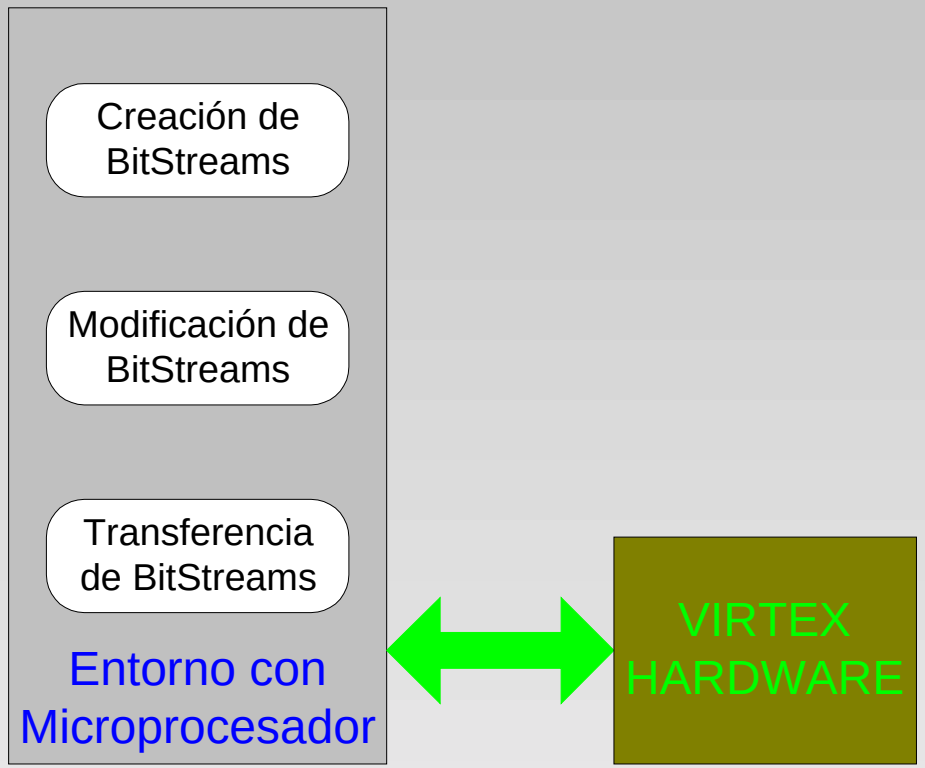


Herramientas

- Bajo grado de automatización. Solamente la creación del BitStream impide que se trabaje a muy bajo nivel

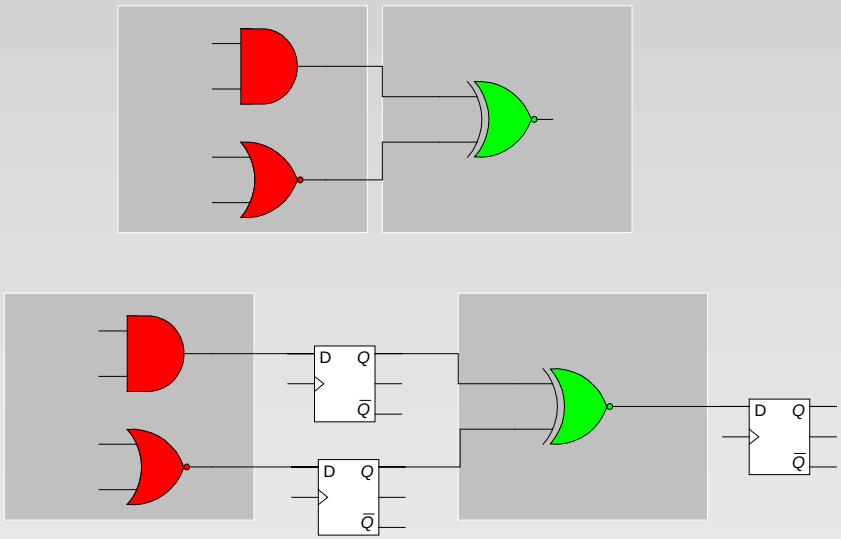


Modelo de Ejecución



RD Multicontexto

- Se permite un cierto grado de automatización si se conmuta entre dos configuraciones cada ciclo de reloj



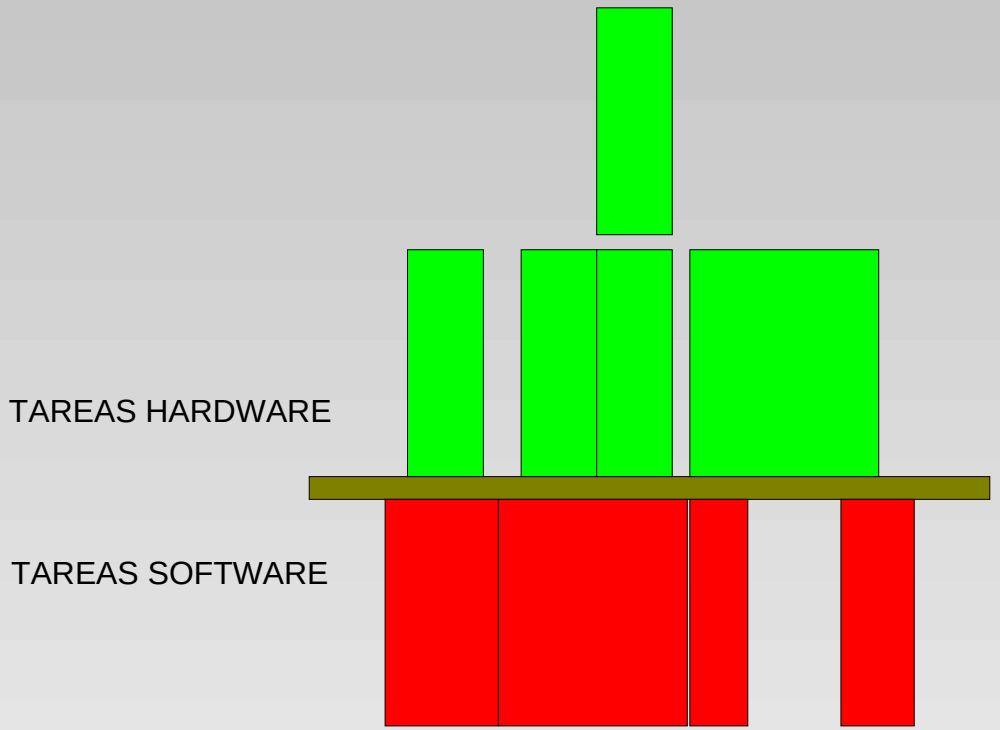
Automatización

- Se trabaja a nivel de bit, no de tarea.
- Se formula como un problema de particionamiento en el que se busca una reducción de los recursos globales reutilizándolos. Se mejora el coste por puerta.

Sistemas Híbridos

El resultado es un reparto de las tareas:

- Criterio global de máximas prestaciones
- Criterios de funcionalidad óptima de cada parte
- Considerar la morfología del sistema

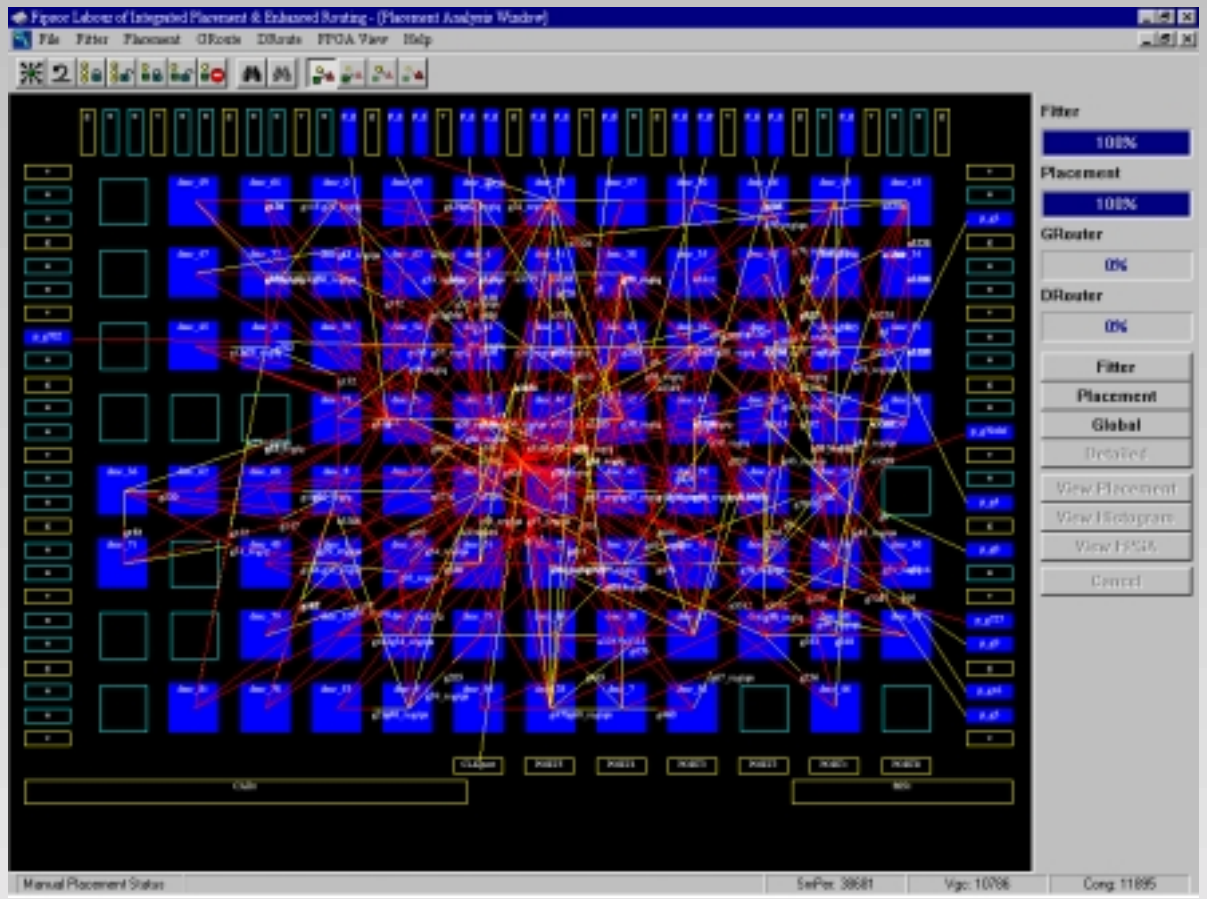


Herramientas

- Comercialmente se requieren herramientas:
  - Sencillas de manejar
  - Alto grado de automatización y pocos controles
  - Que generen soluciones ASAP
  - Entrada HDL o herramientas similares.
- Las herramientas para RDP:
  - Trabajan a nivel de bit
  - Herramientas que trabajan cerca del dispositivo
  - Sugieren el uso de librerías de bloques regulares
- Las técnicas mixtas parecen las más adecuadas

Ejemplo: FLIPER (FIPSOC)

- Entrada alto nivel
- Generación manual de macros pre-colocadas y pre-ruteadas
- *Placement* manual con posibilidades de impedir posiciones para el flujo automático.



Sobrecostes de arquitecturas

Esencialmente se puede medir como el costo por puerta útil. Éste está directamente relacionado con el área de silicio.

Hay que sumar:

- Área del mecanismo de programación
- Área de la funcionalidad y rutado

Se asume que el área de programación es un 25% del área total, en un dispositivo convencional no RD.

Fundamentalmente incide el área de programación

Comparación de áreas (Compton-00)

- Dispositivo no RD:

  Area = nº Bits * Area Bit Prog;

- Dispositivo RDP:

  Area = nº Bits * Area Bit Prog/2 + Area Dec Filas +
  Area Dec Col + Area Buffers

- Dispositivo RDMC:

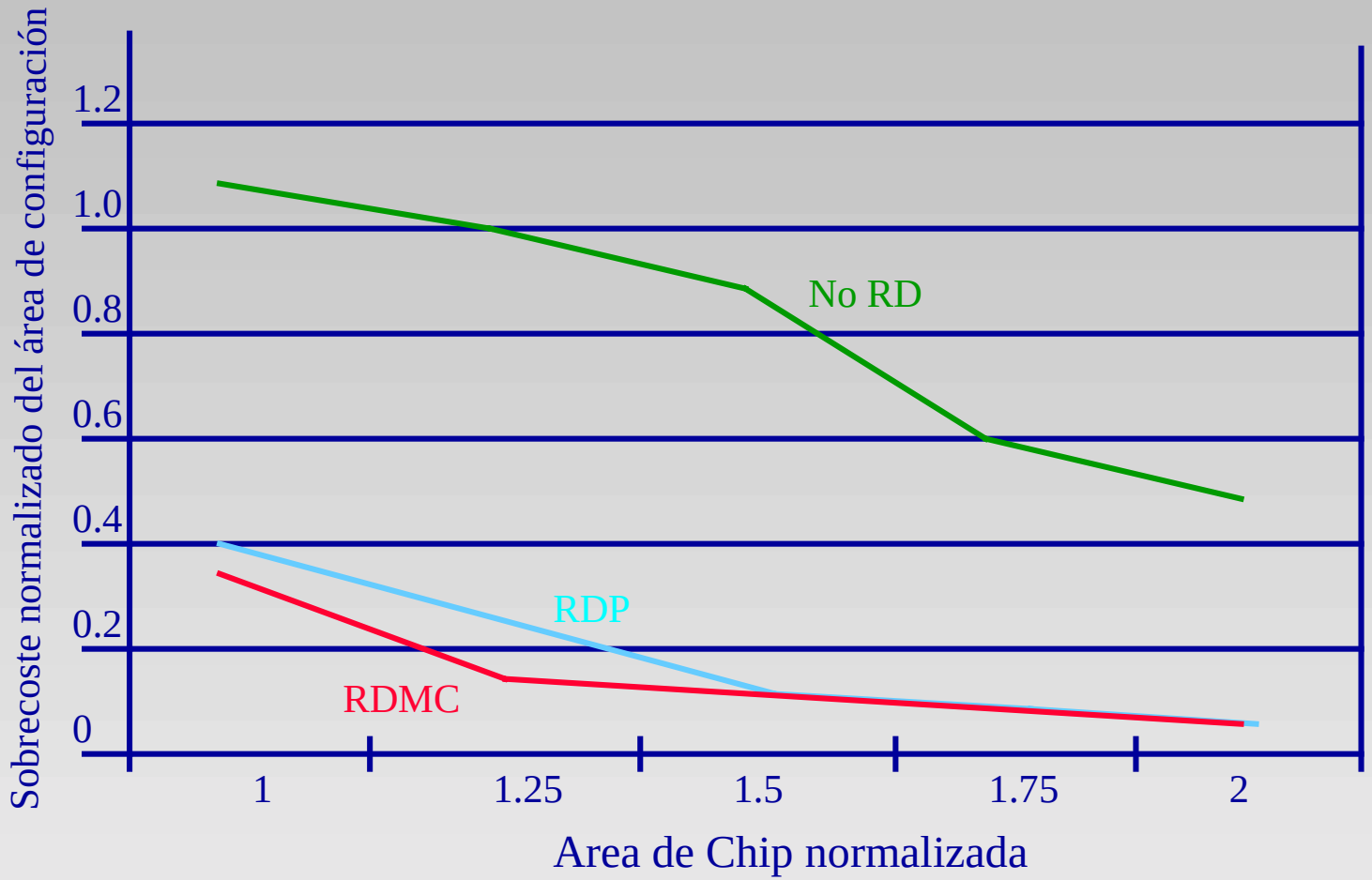
  Area = nº Bits * Area Bit Prog/2 * NC + Area Dec Filas +
  Area Dec Col + Area Dec Contexto + Area Buffers

Comparativa

Estructura de Programación	Area para 1Mb
No RD	9.544 λ^2
RDP	10.547 λ^2
RDMC-2	16.694 λ^2
RDMC-4	20.855 λ^2
RDMC-8	29.273 λ^2

Las curvas para cada estructura se comportan con crecimiento exponencial con el número de bits de programación. (Compton-99)

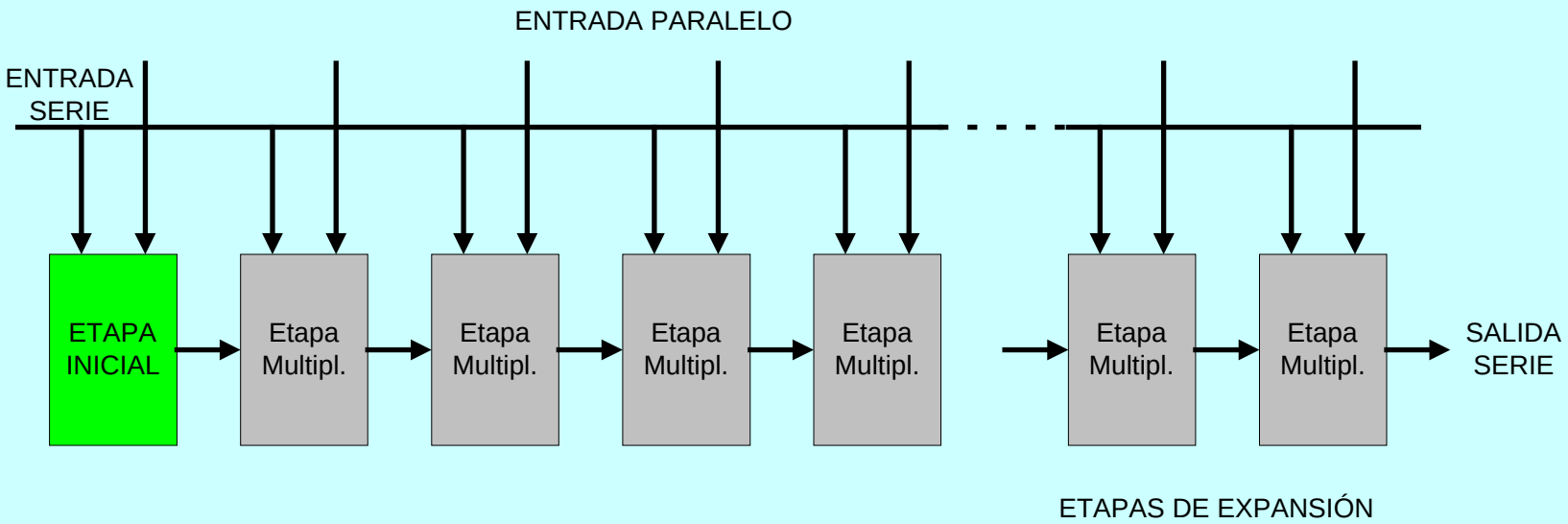
Comparativa (Li-00)



Sobrecoste de operadores

- Las técnicas de diseño para RDP se basan, fundamentalmente en la creación de macrobloques programados a bajo nivel.
- Se aconseja, para reducir tiempos de desarrollo utilizar arquitecturas de elementos expandibles.
- Esto reduce enormemente la capacidad de selección de arquitecturas de operadores

Ej: Multiplicador Paralelo-Serie



Sin embargo...

Las prestaciones obtenidas por los operadores son óptimas, tanto en comportamiento como en ocupación, frente a los mismos operadores sintetizados.

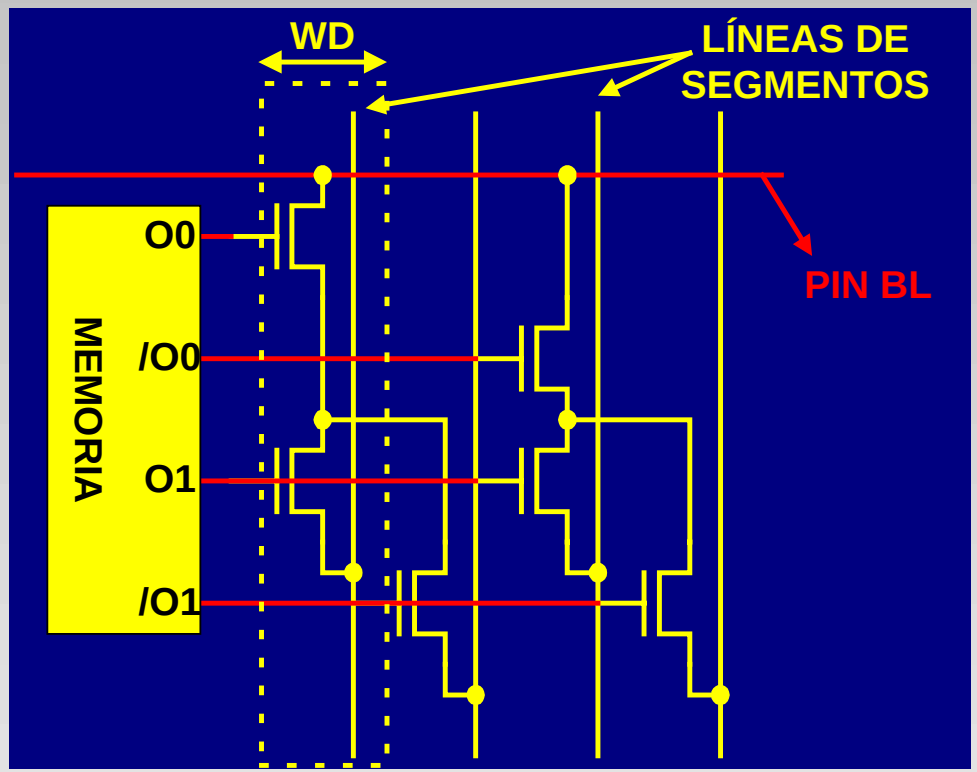
Sobrecostes en tiempos de ejecución

- El aspecto más crítico es el tiempo de reconfiguración.
- Las técnicas de reconfiguración parcial exigen una previsión de estos tiempos para evitar detener la ejecución de las tareas.
- Se mejora notablemente con la RPMC, cuando se utiliza con técnicas de preprogramación.
- Para reducir tiempos:
  - Técnicas de compresión de configuraciones
  - Técnicas de reducción mediante codificación

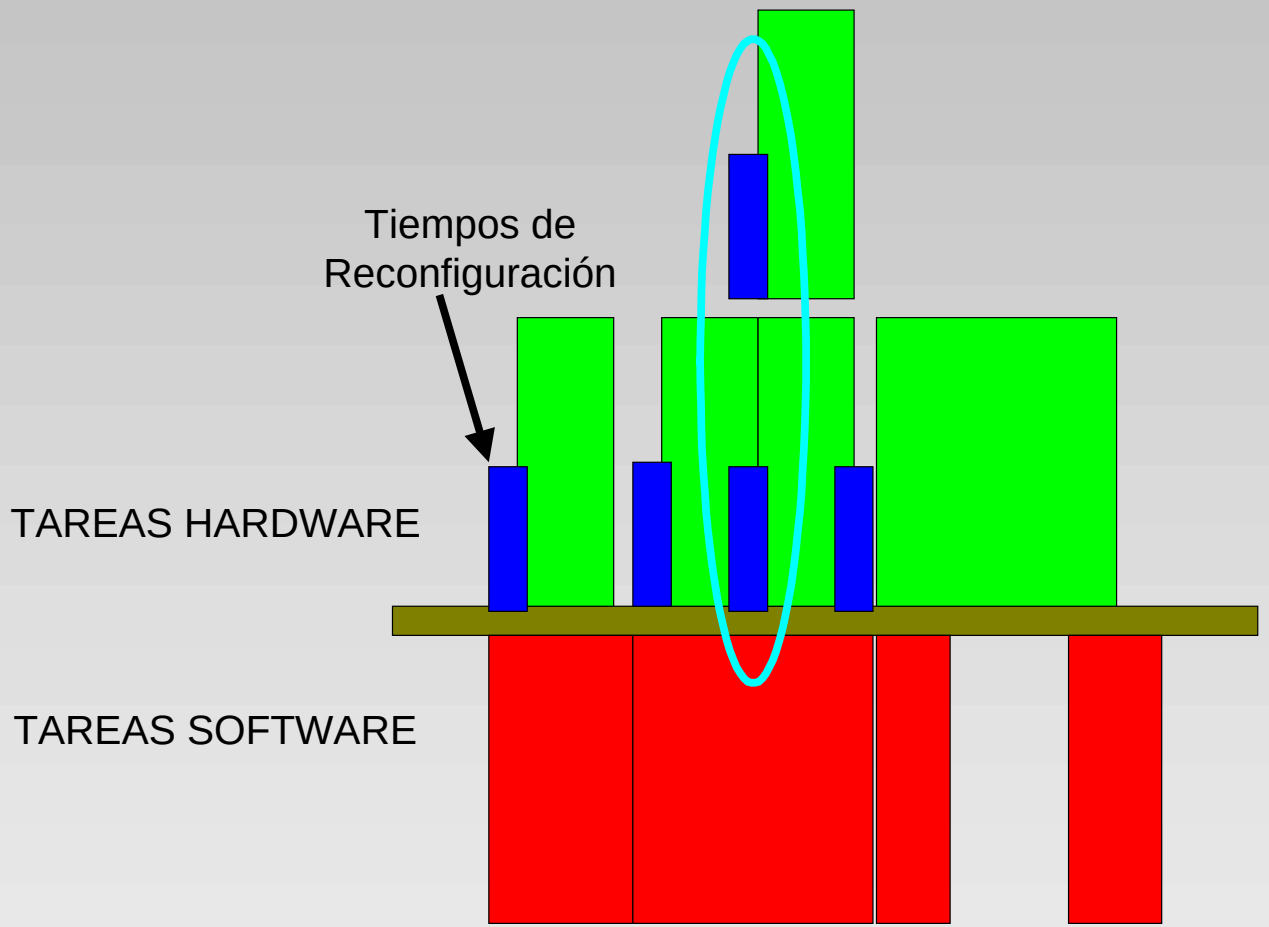
Técnicas de codificación (Baena-01)

Uso de decodificadores:

- Es común a las entradas
- A las salidas:
  - Reduce la rutabilidad
  - Reduce el área
  - No varía significativamente el retraso
  - Sin embargo complica en mucho el software



Conflictos por decisiones en el flujo



Conclusiones

- Diseñar sistemas con capacidad RD resulta ventajoso, si:
  - Si se precisa flexibilidad en el diseño
  - Si hay ganancia en determinadas tareas hardware junto con su tiempo de reconfiguración
  - Si se admite un reparto ventajoso:
    - Costo en área ☺
    - Costo en tiempos de desarrollo ☹
    - Costos de tiempos de ejecución 😐