

# QoS / TCs / VCs y arbitraje en la capa de transacción PCIe

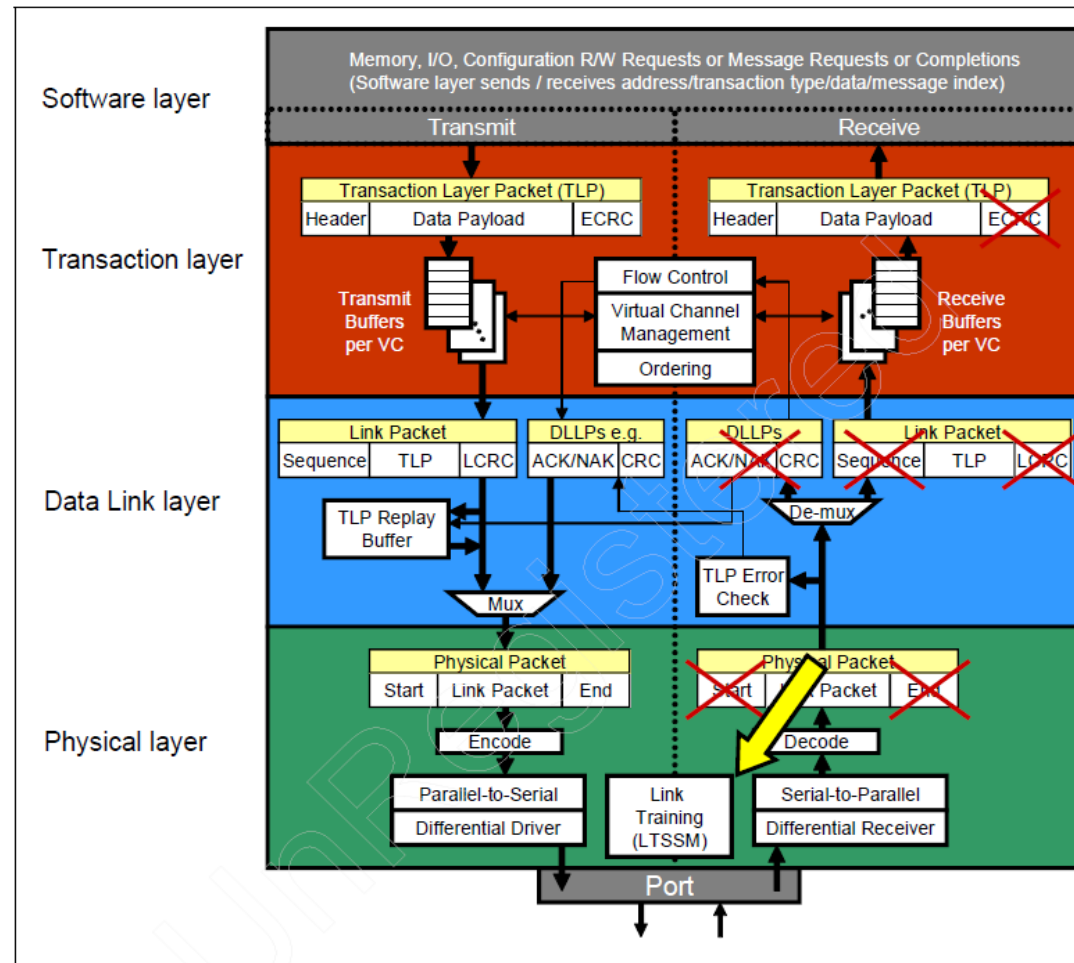
Adaptación de la arquitectura para el proyecto de diseño #2

Prof. Jorge Soto

IE-0523 Circuitos Digitales II

# Diagrama de capas PCIE

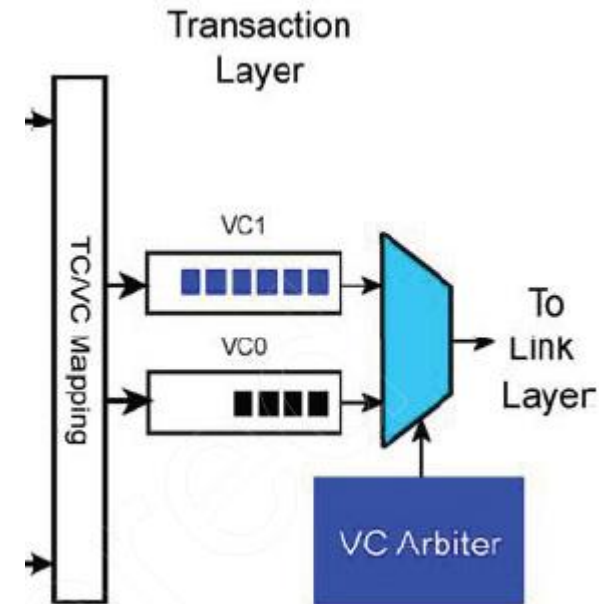
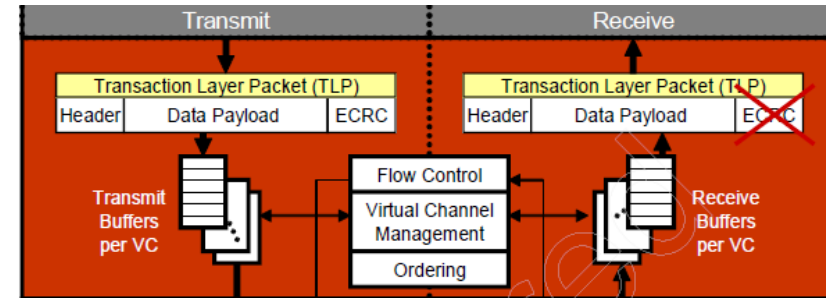
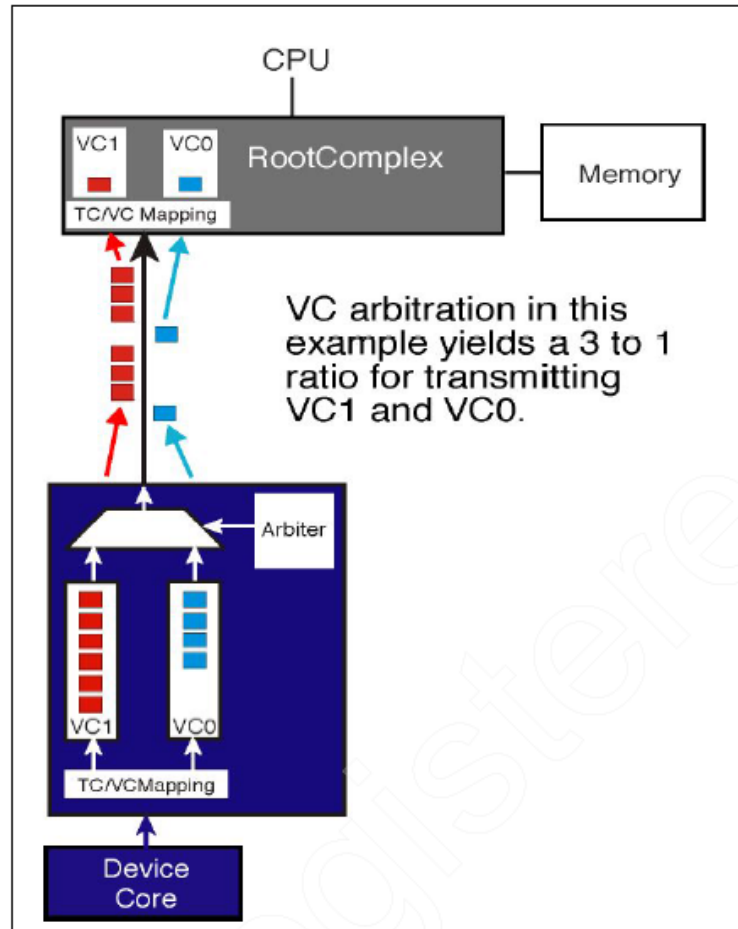
Figure 14-1: Link Training and Status State Machine Location



- Calidad de Servicio (QoS) opera en la capa de transacción
- Por lo general, cada Canal Virtual (VC) tiene una Clase de Tráfico (TC) asociada

# Arbitraje en los canales virtuales (VC)

Figure 6-7: Conceptual VC Arbitration Example



# Arbitraje en los canales virtuales

Figure 6-8: Strict Arbitration Priority

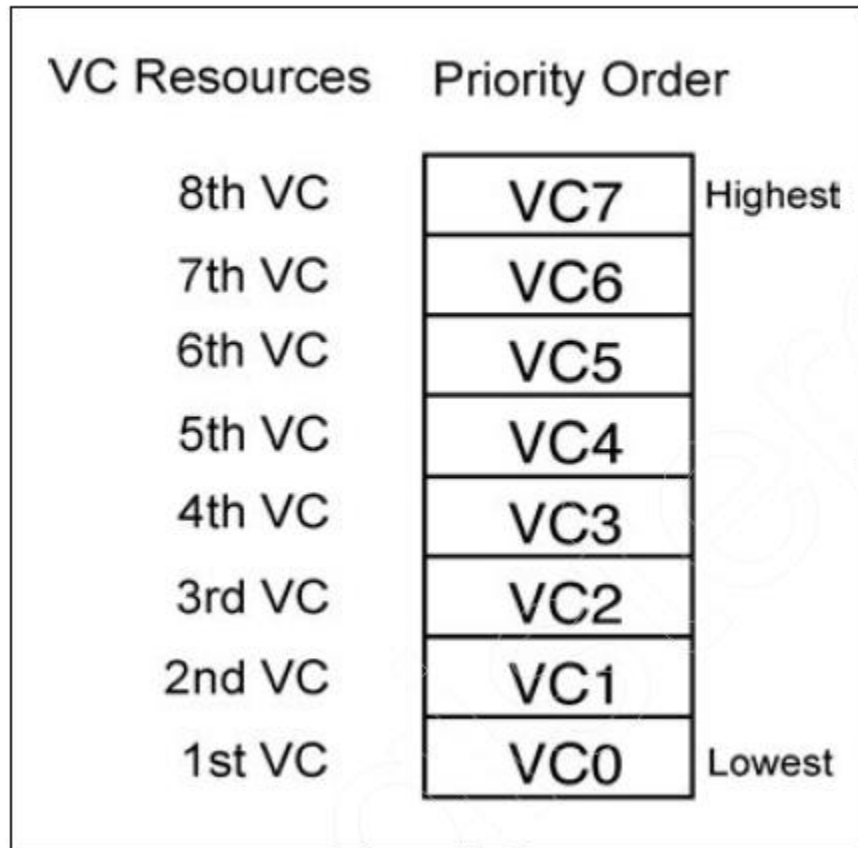
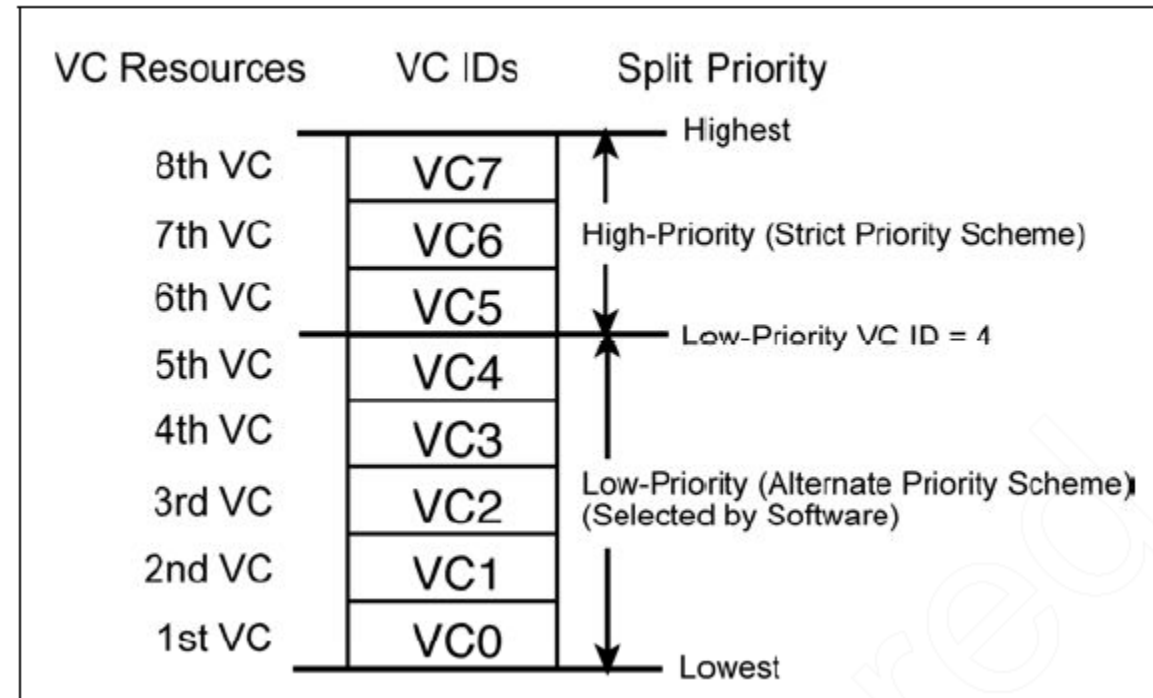
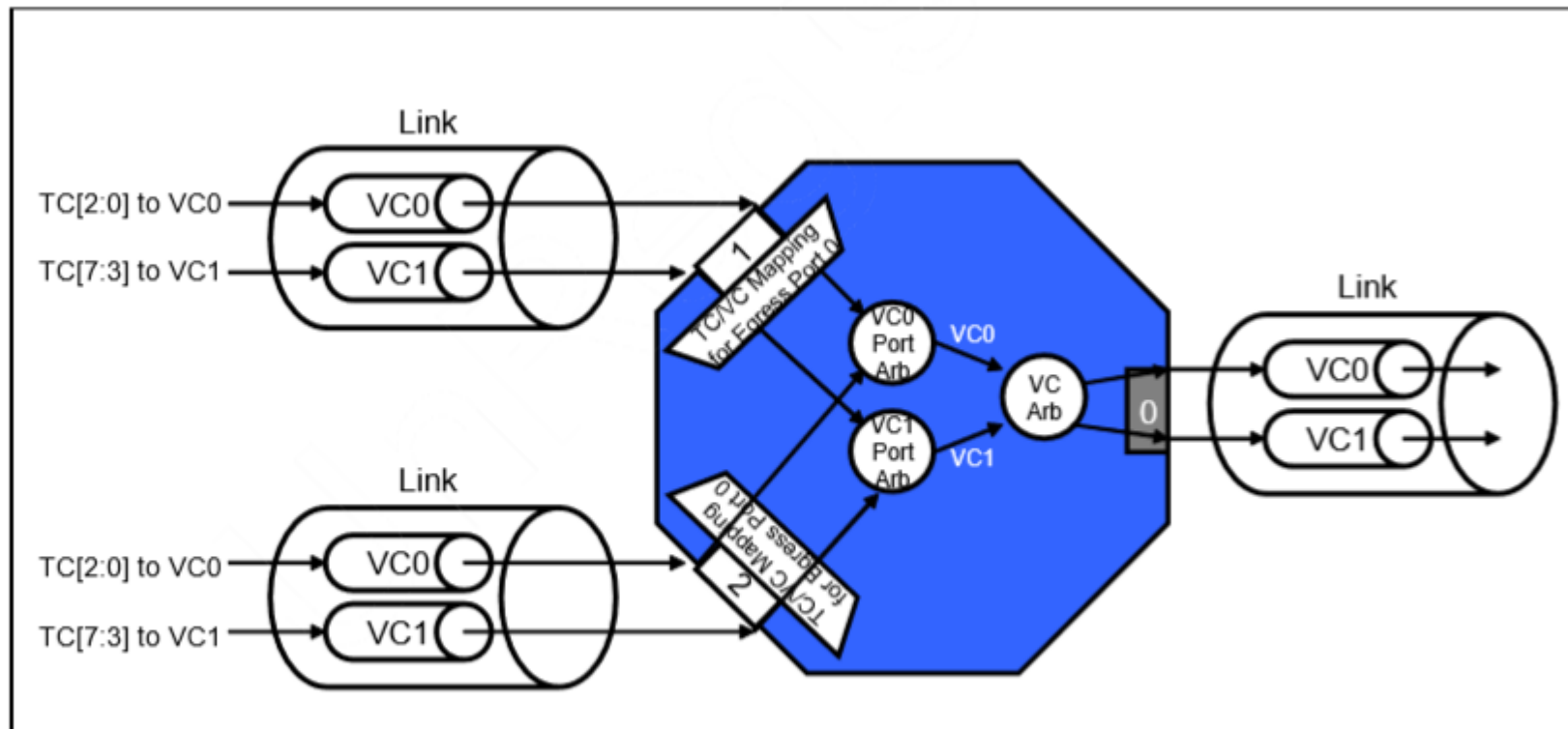


Figure 6-11: VC Arbitration with Low- and High-Priority Implementations



# Switches en PCIe

Figure 2-24: Switch Implements Port Arbitration and VC Arbitration Logic

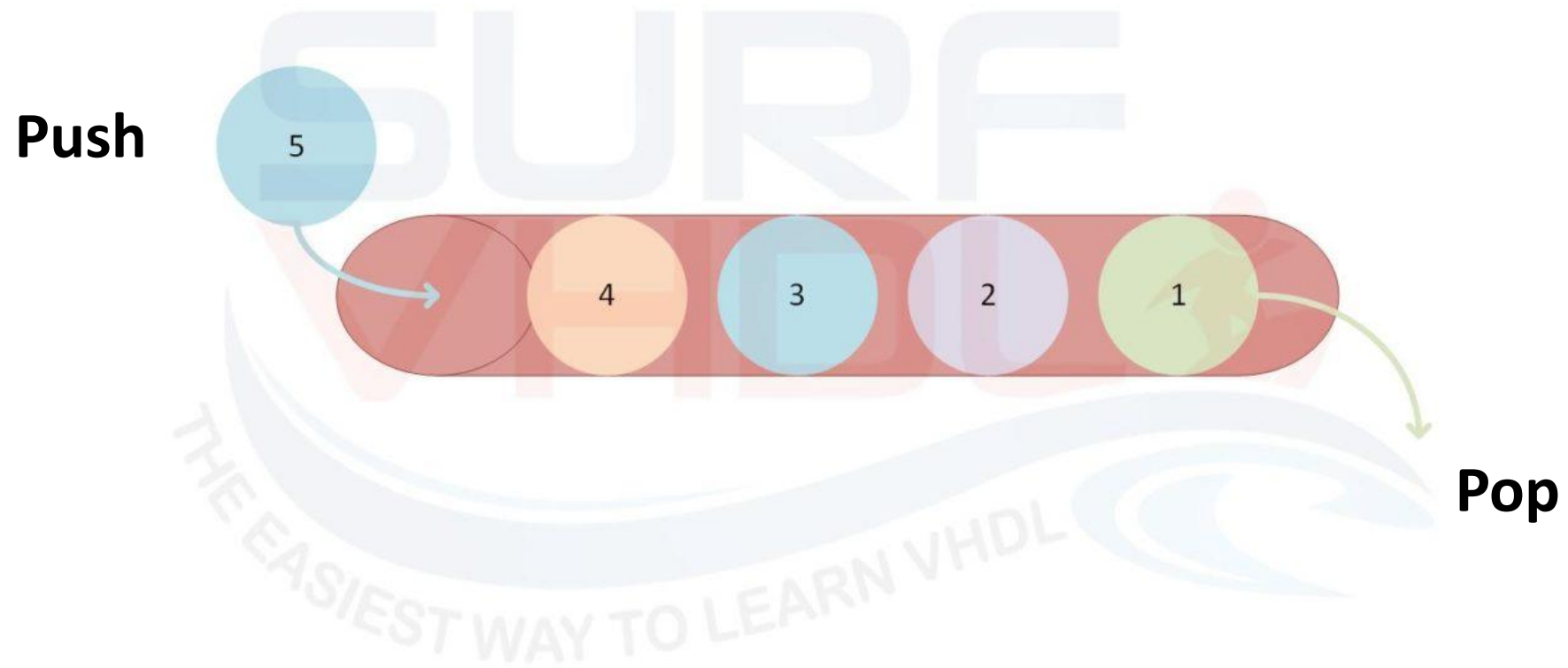


Endpoint devices and a root complex with only one port do not support port arbitration. They only support VC arbitration in the Transaction Layer.

Adaptación de la capa de transacción para el proyecto de diseño #2, IE-0523.

Módulos misceláneos para el  
proyecto de diseño #2

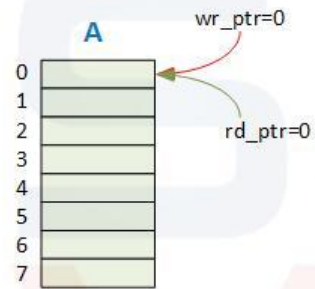
# Buffer/FIFO



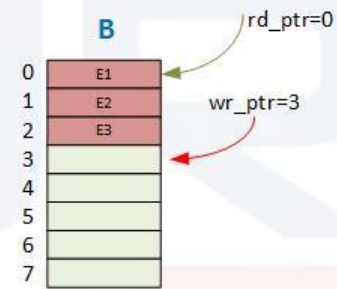
Tomado de: <http://surf-vhdl.com/what-is-a-fifo/>



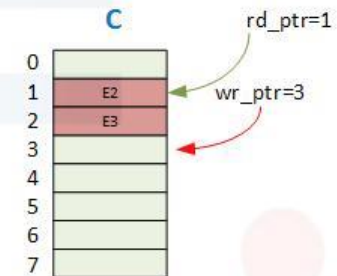
# Buffer/FIFO



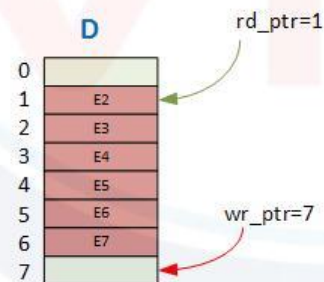
Fifo Empty



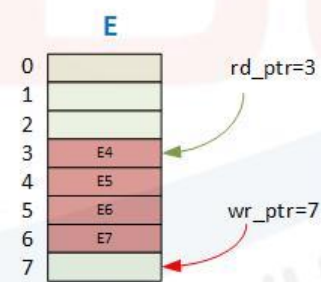
3 element written  
1 element read from



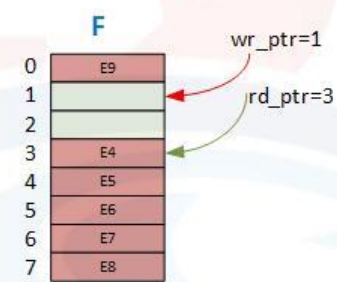
3 element written  
1 element read from



Write 4 element more

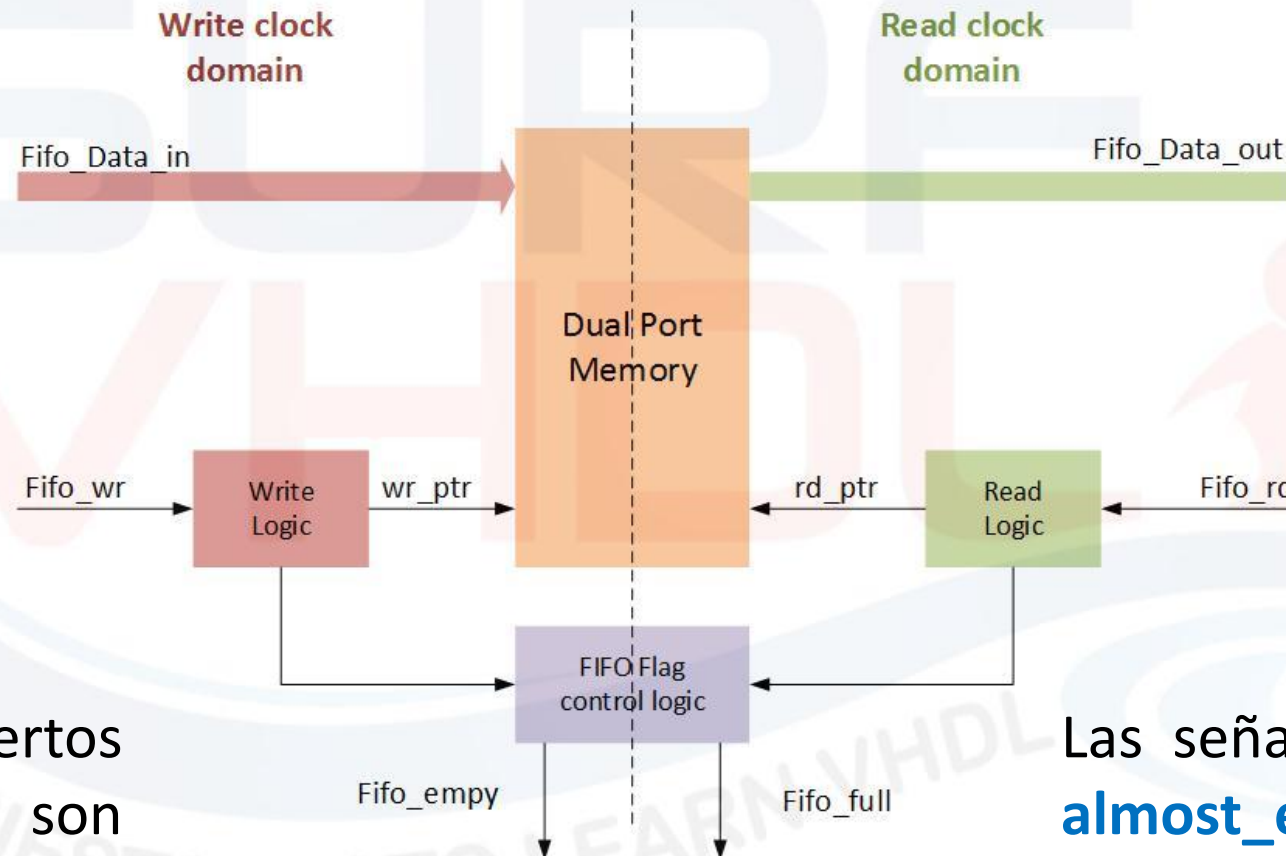


read 2 element more



write 2 element more  
Write pointer wrap around  
6 element in the FIFO

# Buffer/FIFO

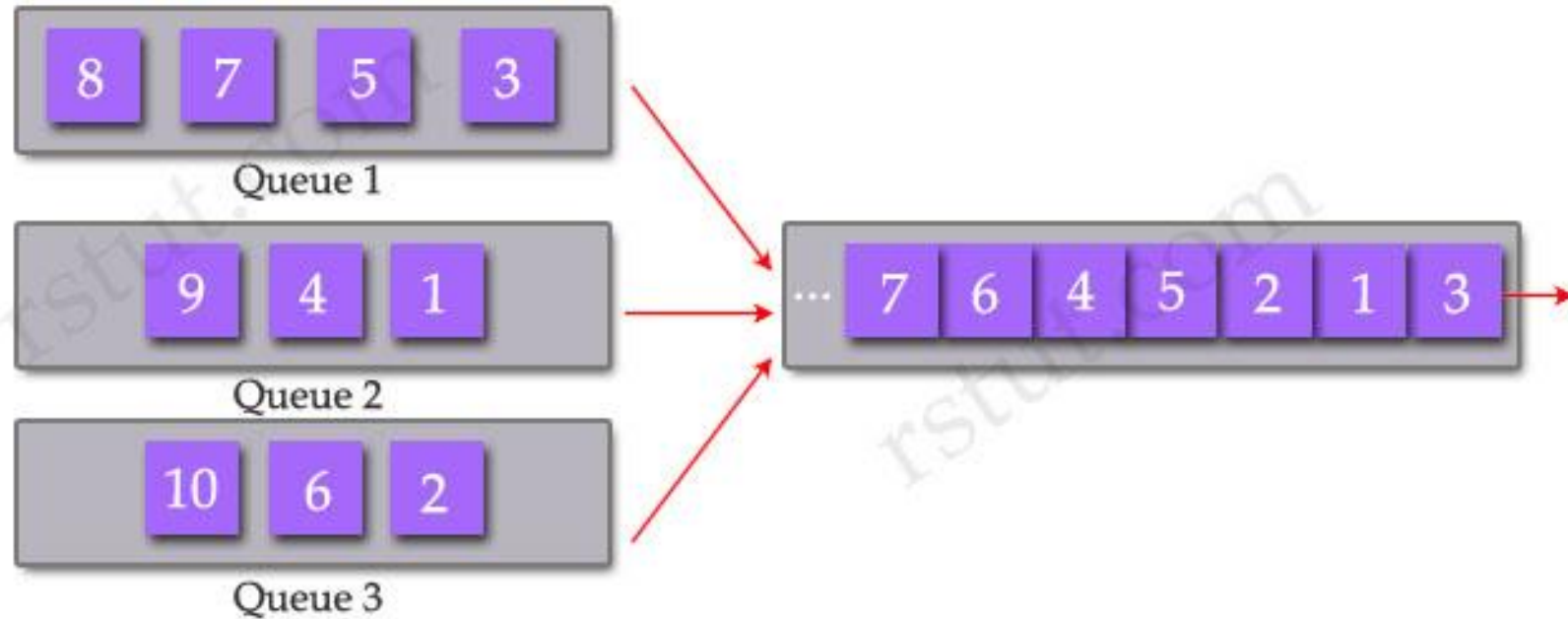


Una escritura y no lectura en full es un error (señal de salida)

¡Cuidado! Los puertos “inout” no son sintetizables.

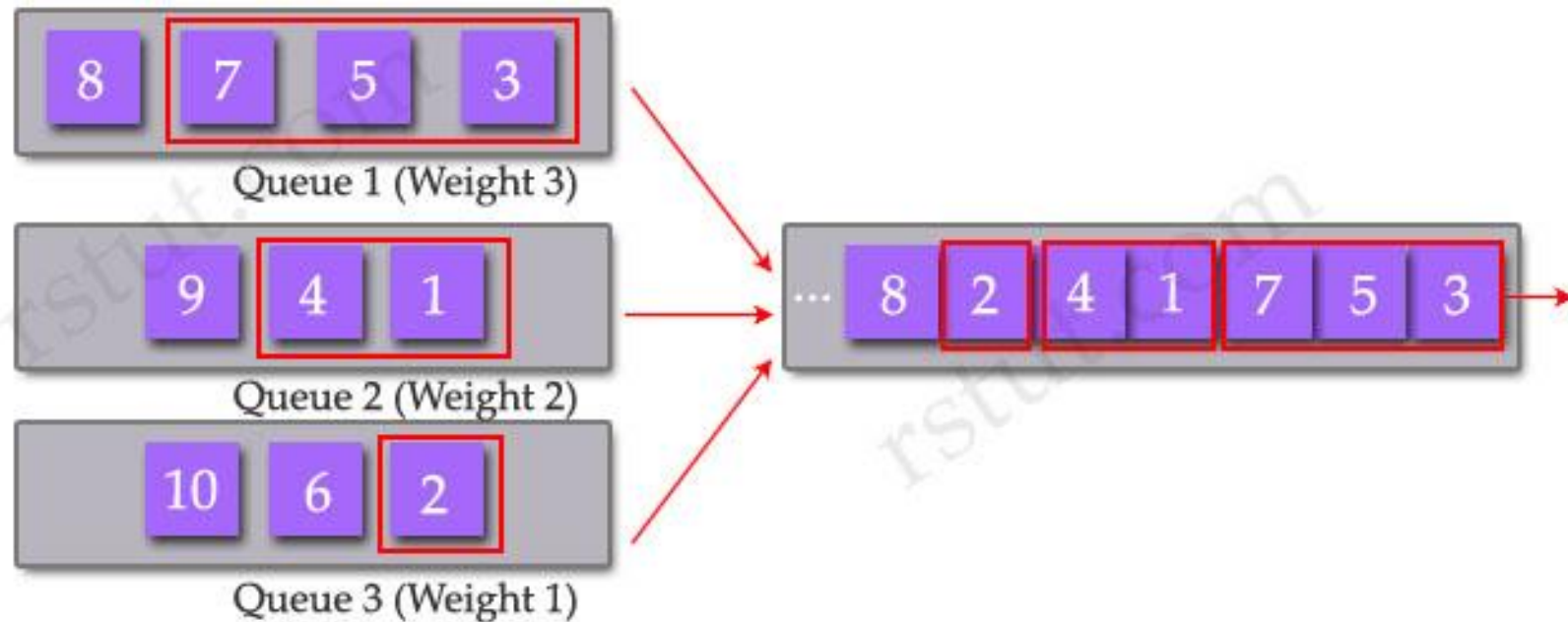
Las señales **almost\_full** y **almost\_empty** del FIFO se utilizan en Flow Control

# Round Robin



Todos los queues tienen la misma prioridad

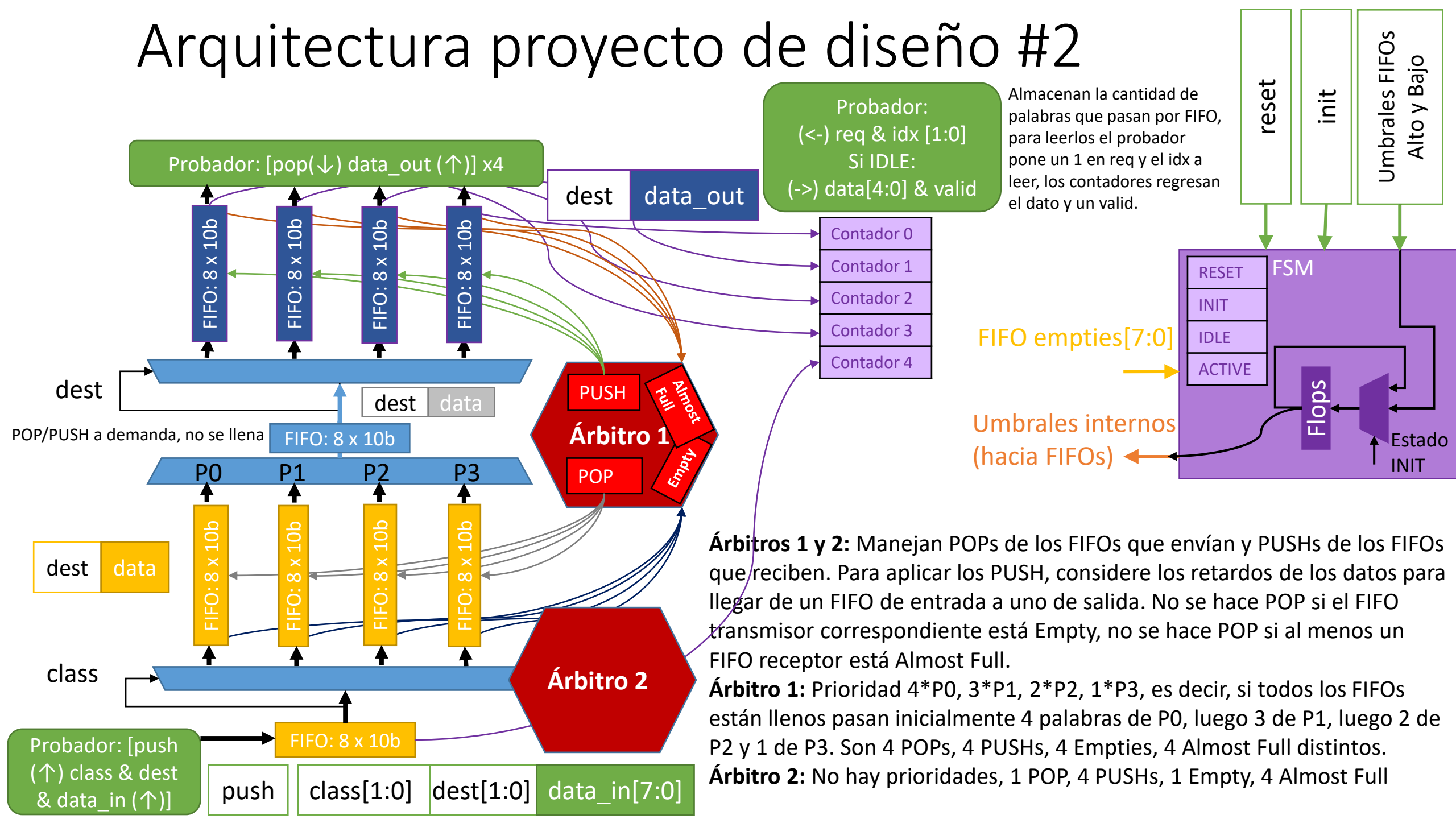
# Weighted Round Robin



Los pesos de cada clase de tráfico son entradas al “round robin”. En el caso del Queue 1, se le asigna un valor de peso o prioridad de 3. Lo mismo para los pesos 2 y 1.

# Arquitectura proyecto #2

# Arquitectura proyecto de diseño #2



# FSM de control

- RESET: Estado de reset, cambia a INIT.
- INIT: Forzado mediante señal “init”, permite la modificación de registros “Umbrales”. Precedencia sobre IDLE, cambia a IDLE.
- IDLE: Todos los FIFOs están vacíos. Salida “idle” en 1 sólo en este estado. Cambia a ACTIVE al tener un FIFO no vacío.
- ACTIVE: Modo de transmisión de datos por defecto.

# Pruebas del bloque completo

1. Saque el bloque de RESET, manteniendo el estado de INIT (señal init)
2. Modifique 2 veces los umbrales altos y bajos de los FIFOs (son el mismo umbral bajo y alto para todos). Libere la señal init.
3. Provoque un Almost Full en todos los FIFOs de salida, el árbitro no le permitirá hacerlo de forma simultánea. Desde el probador, haga la menor cantidad de POPs. Verifique que las palabras que salieron son las mismas que entraron y que salieron por la salida correcta en la prioridad correcta.
4. Provoque un Almost Full en todos los FIFOs de entrada. Luego usando POPs del probador deje todos los FIFOs vacíos. Verifique que las palabras que salieron son las mismas que entraron y que salieron por la salida correcta en la prioridad correcta.
5. Lea los contadores de palabras. El contador 4 debe tener el mismo valor que la suma de los contadores 0, 1, 2 y 3.
6. Envíe 16 palabras, 4 a cada FIFO de entrada, y cada set de 4 palabras por FIFO de entrada estén con destino a cada FIFO de salida (las  $4 \times 4 = 16$  combinaciones posibles). Deje todos los FIFOs vacíos. Verifique que las palabras que salieron son las mismas que entraron y que salieron por la salida correcta en la prioridad correcta.
7. Lea los contadores de palabras y valide un aumento de 4 palabras por contador. El contador 4 debe tener el mismo valor que la suma de los contadores 0, 1, 2 y 3.