

## ***GUÍA MÓDULO 6: Sincronización e interfases entre sistemas activos en distintos dominios de reloj***

Así como en el módulo 4 se analizó el caso de sistemas levemente interconectados (como es un enlace de datos serial), en este módulo se trata de presentar el caso de sistemas fuertemente interconectados que trabajan en distintos dominios de reloj. En este caso a la complejidad del uso de gran número de sincronizadores (para evitar la metaestabilidad) se agrega el problema de la pérdida de simultaneidad entre los datos ya sincronizados, para lo que se plantean ciertas posibles soluciones One Hot (tipo demux), Gray, handshake. Cuando la tasa de intercambio se incrementa, una solución es el agregado de sistemas intermediarios basados en memorias (dual port, FIFO) lo que introduce la necesidad de mecanismos de administración de las transacciones (FLAGS y SEMÁFOROS, ARBITROS), y ciertos formalismos de descripción, como las redes de Petri.

Con la aparición de enlaces de ultra velocidad (como PCI Express a 2,5Gbps) estos problemas se agravan y aparecen otras soluciones para recuperación asincrónica de datos, y alineación de datos sincrónicos multi-lane.



### **Actividad 6.1 (Entrega Obligatoria)**

1. Busque información sobre los mecanismos de interacción y sincronización entre procesos de software intra-chip e inter-chip ofrecidos por el lenguaje OCCAM y los Transputers (T414, Inmos, 198x). Realice un informe sobre estos mecanismos.
2. Especifique componentes de VHDL que permitan obtener funcionalidades de sincronización parecida a la de los transputers entre procesos de hardware.
3. Ventajas formales de validación de FSMs basadas en Petri Nets.
4. Describa posibles formas de representar en VHDL componentes que emulen los “lugares” de una red de Petri resuelta por hardware.
5. Elabore en VHDL los componentes “HANDSHAKE”, uno “FLAG” y uno “SEMAFORO BINARIO” que trabajen entre dos sistemas con distintos dominios de reloj.



Utilice el recurso [Actividad 6.1](#) para enviar



### **Actividad 6.2 (Entrega Obligatoria)**

1. Implemente en VHDL el circuito de recuperación asincrónica de datos, propuesto en la transparencia, donde los datos serie se paralelizan en paquetes de 10 bits, aunque sea a nivel de sub-bloques.



Utilice el recurso [Actividad 6.2](#) para enviar

### Consideraciones finales

En un diseño digital moderno de mediana complejidad ha pasado a ser razonable pensar en la integración de varios subsistemas dentro de un único dispositivo, lo que ha dado origen al término SOC (SystemOn a Chip) y al de PSOC, cuando se emplean FPGAs. Con este módulo se pretendió presentar y discutir algunos recursos necesarios para poder realizar esa integración.

### Bibliografía sugerida:

- [http://en.wikipedia.org/wiki/Petri\\_net](http://en.wikipedia.org/wiki/Petri_net)
- <http://www.hindawi.com/journals/ijdsn/2009/758754/>
- [http://www.xilinx.com/support/documentation/application\\_notes/xapp861.pdf](http://www.xilinx.com/support/documentation/application_notes/xapp861.pdf)
- [http://www.radioeng.cz/fulltexts/2010/10\\_01\\_074\\_078.pdf](http://www.radioeng.cz/fulltexts/2010/10_01_074_078.pdf)