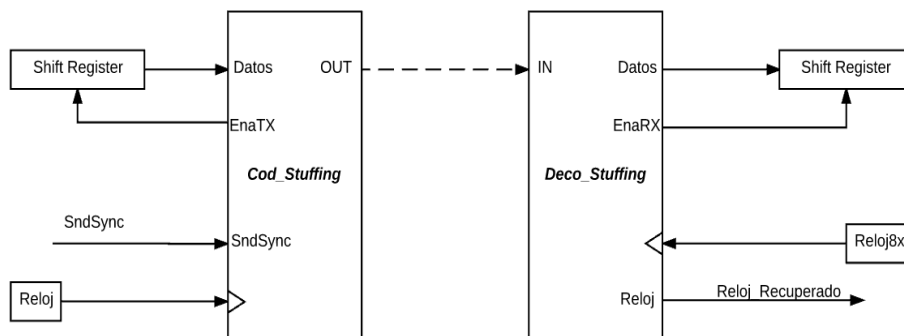


Actividad 4.2

1) Generalidades del sistema

Para ensayar el sistema compuesto por un codificador y decodificador de bit-stuffing se preparó un proyecto que a nivel global se presenta en el siguiente diagrama



Codificador

La codificación por bit-stuffing no es más de lo que debe ser. Para lograr buena sincronización se hizo que inicie con el envío de una cantidad configurable de ceros y una trama de Sync para poder analizar la funcionalidad de manera más fácil al momento de analizar el test-bench. Su funcionalidad en modo normal será realizar un cambio de estados en caso de tener que enviar un '0' o mantener su estado si se envía un '1'. Cada vez que se envía un nuevo dato se pide con EnaTx el próximo dato a ser enviado. En caso de haber mandado 6 '1's, esta señal no será habilitada y se enviará un '0' a modo de bit-stuffing.

Decodificador

En primera instancia se realizó esta sección de manera sincrónica para validar su funcionamiento. En primera instancia su funcionalidad es inversa a la de codificación, solo que al decodificar, se agrega la detección de la trama de Sync (para no incluirla en los datos recuperados) y la posibilidad de detectar un error en la trama cuando, al no ser trama Sync, se detecten 7 '1' seguidos. En este caso no se toma ninguna medida pero se notifica su aparición.

La manera de recolectar datos se realiza por medio de un shift-register que es manejado con la señal EnaRx, ya que en caso de detectar 6 '1', el '0' que le sigue no es incluido en la cadena de datos recuperados.

Como segunda etapa se planteó la recuperación de reloj. Debido a que no todas las transiciones cumplen con períodos de reloj, se tomó la política de solo analizar los flancos de los datos cuando su distancia sea cercana la multiplicación del reloj de referencia. Esto significa que, teniendo una referencia 8x del reloj, si la distancia entre flancos de datos de entrada es mayor a 8 (en realidad se utilizó 10 por posibles corrimientos) esa situación no se tendrá en cuenta, ya que solo se logrará detectar que el reloj debería ir mucho más lento, produciendo sincronizaciones sistemáticas indeseadas.

Teniendo en cuenta esto, solo se analizará la sincronización en momentos donde se envíen '0's consecutivos, y ya que esto simula un sistema con CLK continuo (ya que se desestima el resto del

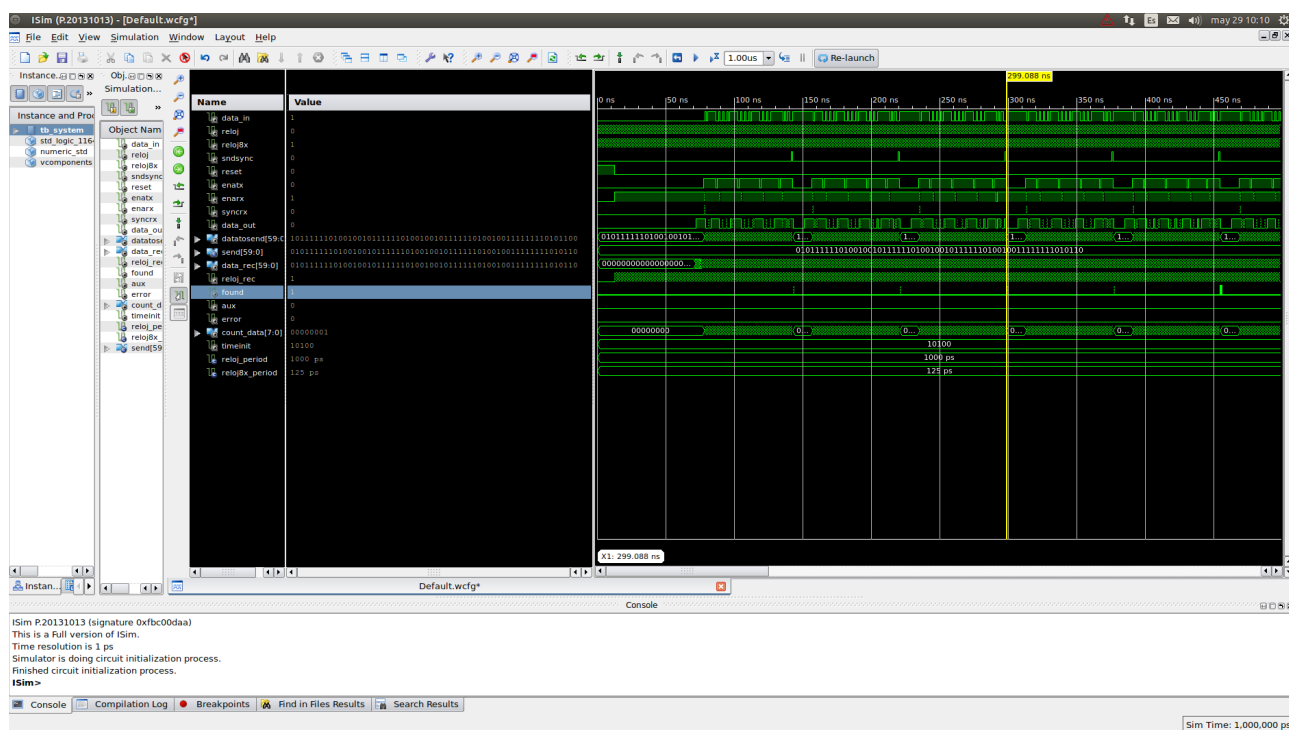
tiempo), se volvió a implementar la máquina de estados del MC4044 logrando muy buenos resultados.

La gran diferencia respecto a su primera utilización, es que en este caso se adaptó a que la condición ideal es la división por 8, por lo que el sistema divisor de CLK varía entre 7,5 y 8,5.

Testing

Para ensayar todo esto se propuso el envío de una trama de 60 bits de manera cíclica, con la perspectiva de encontrar sin errores cada una de estas tramas. Esto permite analizar no solo el funcionamiento del codificador y decodificador del bit-stuffing sino también el sistema de recuperación de reloj, dado que la trama enviada cuenta con cadenas de '0's que dificultan la buena sincronización-

En el Test-bench se propone un reloj principal y uno de referencia 8 veces más rápido, al cual, a modo de ensayo, se lo hizo variar $\pm 4\%$ manteniendo la funcionalidad del sistema.



En esta imagen se ven las señales compartidas globales y las señales de *Found* que se levantan cuando el registro de datos recuperados copia exactamente la palabra que se envió desde el codificador.

Se adjunta el proyecto.