

Problema 1 (7 puntos)

a) Dar la representación en complemento a 2 de estos números con la mínima cantidad de bits con los que se puedan representar los 3 números.

A = 26 B = -98 C = 237

b) Realizar las operaciones (B + C) y (B - C) e indicar si el resultado es correcto en representación complemento a 2 para la cantidad de bits determinada en a. Justificar.

Problema 2 (8 puntos)

La siguiente tabla corresponde a un circuito secuencial modo nivel.

	00	01	11	10	00	01	11	10
a	c	<u>a</u>	b	<u>X</u>		1		
b	d	a	<u>b</u>	<u>b</u>			0	0
c	<u>c</u>	a	<u>c</u>	b	0		1	
d	<u>d</u>	a	b	<u>d</u>	1			1

a) Dar una nueva tabla con una asignación de variables de estado sin carreras críticas. Utilizar la mínima cantidad de variables de estado posible.

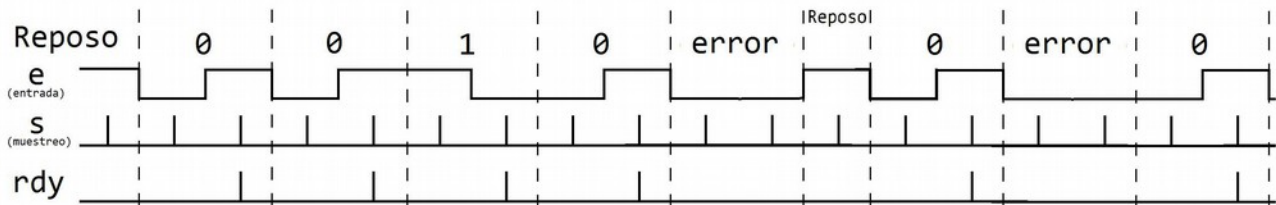
b) Asignar salidas a los inestables de modo de no tener espurios. Asignar solamente en donde sea necesario.

Problema 1 (30 puntos)

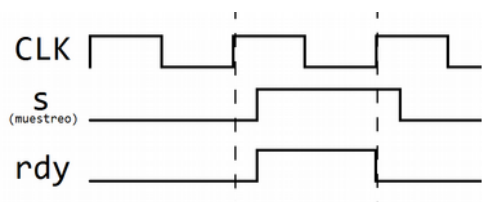
Se reciben datos serie en una entrada **e** cuya codificación se muestra en el siguiente diagrama:



Se desea diseñar un circuito modo reloj que decodifique los datos, para ello se utilizará un reloj **Tck** mucho más rápido que las variaciones de la entrada, y otra entrada de muestreo **s** que indica con un pulso de duración **1 Tck** el momento óptimo para observar la señal **e**. En la siguiente figura se observa el comportamiento del sistema.



En el caso que se detecte un bit válido (en el segundo semiperiodo del bit), el circuito debe de generar un pulso en **rdy** que suba junto a la subida de **s** y baje con el flanco siguiente del reloj del sistema. Junto al pulso de **rdy**, debe de estar disponible en **out** el valor lógico del bit leído. El resto del tiempo no importa el valor de **out**.

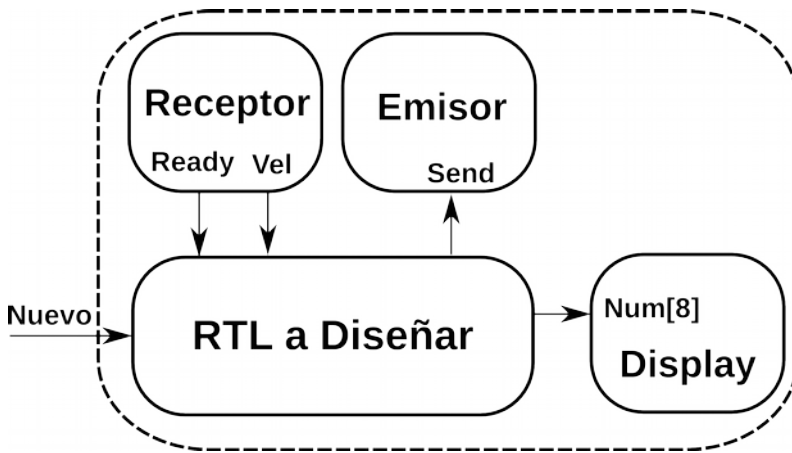


Cuando el sistema no recibe ningún bit (reposo), la entrada **e** se encuentra inactiva en nivel alto. Se garantiza además que luego de un reset la entrada **e** está en reposo y que todas las transmisiones de datos comienzan con un "0 lógico".

En caso de que se reciba un bit erróneo (primer y segundo bit idénticos), se debe descartar sin activar la salida **rdy** y esperar un nuevo inicio de transmisión (**e** = "0"). Observar que si luego del error vienen unos, estos deben ser ignorados y el primer 0 se tomará como un nuevo inicio de transmisión.

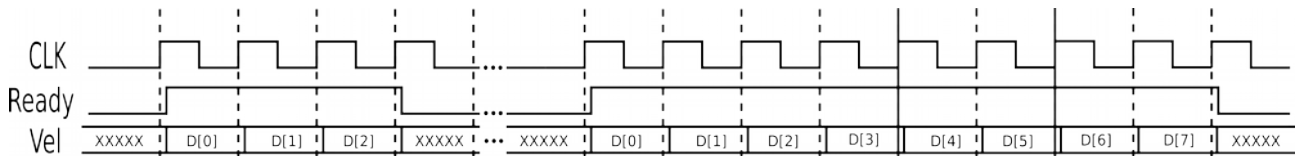
Problema 3 (30 puntos)

Se desea diseñar el módulo de control de un equipo detector de velocidad del saque en partidos de tenis. El equipo calcula la velocidad utilizando el efecto Doppler. Ver figura.



Para esto cuenta con un módulo *Emisor* que mientras tiene en alto su entrada **Send** emite una señal electromagnética a una frecuencia conocida. Por otro lado se cuenta con un módulo *Receptor* que sensa la frecuencia de la señal enviada que cambia al rebotar en un objeto en movimiento, y a partir de esa diferencia calcula la velocidad. Una vez calculada, el *Receptor* envía mediante su salida **Vel** el número binario que representa la velocidad calculada en km/h (entre 0 y 255). El protocolo de transmisión serie es

el siguiente. La velocidad tiene una cantidad de bits variables entre 1 y 8. A su vez el *Receptor* cuenta con una salida **Ready**, sincronizada con el reloj del sistema que cuando sube a '1' indica que se comienza la transmisión de un dato de velocidad y vuelve a cero al finalizar la transmisión. A continuación se muestran dos ejemplos de envío de velocidades por parte del *Receptor*.



El sistema cuenta además con un módulo *Display*, que recibe en su entrada **Num[8]** un número binario de 8 bits que despliega en formato 7 segmentos en tres dígitos.

El módulo de control recibe además la entrada **Nuevo** proveniente de un botón. Cuando el botón es presionado (**Nuevo** vale '1') se debe esperar a que éste se suelte (**Nuevo** vale '0') para comenzar a emitir la señal electromagnética. El botón es presionado por parte de un ayudante cuando está por realizarse el saque.

Si la medida de velocidad no es de 8 bits se debe extender agregando '0's en los bits más significativos. Una vez obtenida la medida, deberá desplegarse en el Display y mantenerse hasta que se obtenga la siguiente. Finalmente, se debe esperar que el botón sea presionado nuevamente para comenzar una nueva medida de velocidad.

Nota:

- Si el botón se presiona mientras se esta recibiendo un dato de velocidad, deberá ignorarse.
- Los datos de velocidad nunca superan los 8 bits.
- La señal **Send** debe permanecer en 1 hasta que se terminen de recibir todos los bits.