

Arquitectura de Computadoras

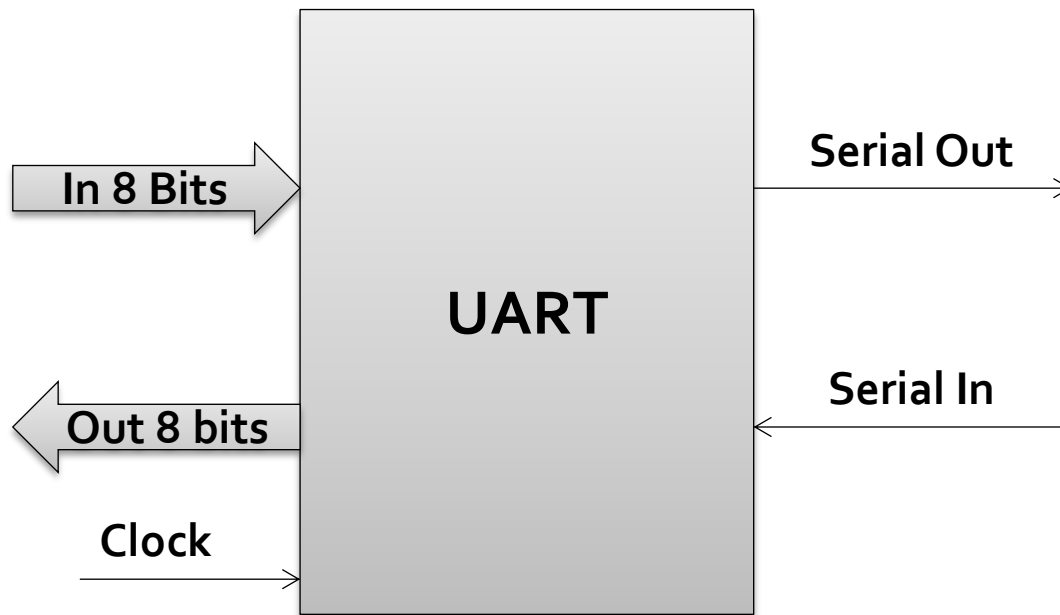
TP N°5: UART

Trabajo practico N°5: UART

- El TP5 xonsiste en realizar un módulo transmisor y receptor UART. El objetivo de este práctico, es el de ser usado como base para la **unidad de debugging en el trabajo final**.
- Fecha de Entrega: **Miércoles 30 de Octubre**.
- Para la evaluación del trabajo, se utilizan las placas Nexys III, que cuentan con un conversor UART-USB para ser leído como puerto COM en la PC. Mediante una terminal tipo HyperTerminal o Putty, se envían caracteres a la unidad UART, y esta simplemente los devuelve, imprimiendo en la consola el mismo carácter que fue enviado.

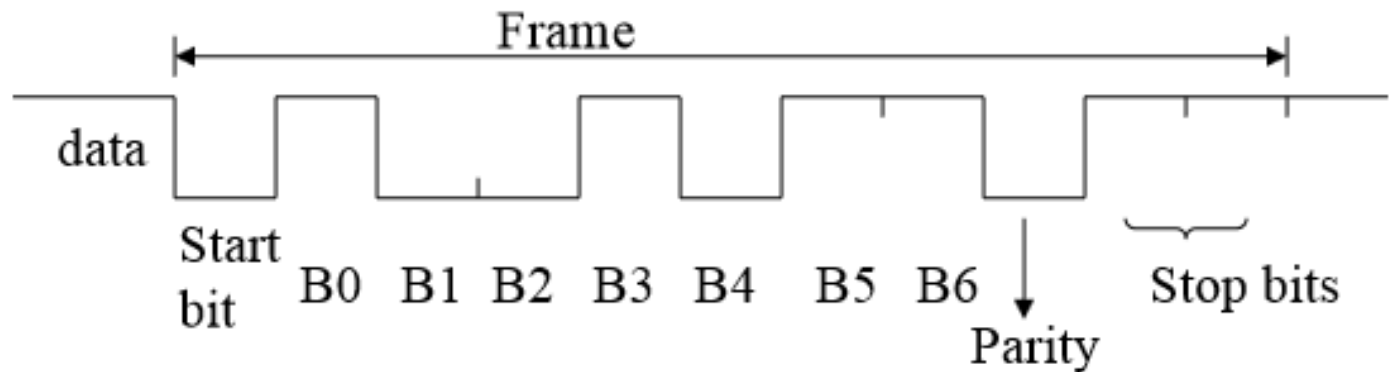
TP: UART

- Universal Asynchronous Receiver and Transmitter

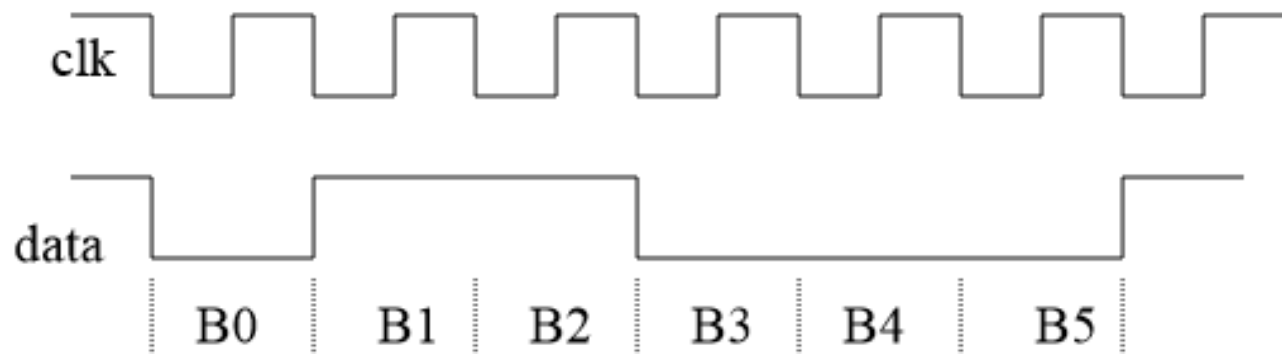


Comunicacion Síncrona vs Asíncrona

Asynchronous
Data transfer



Synchronous
Data transfer



Trama UART

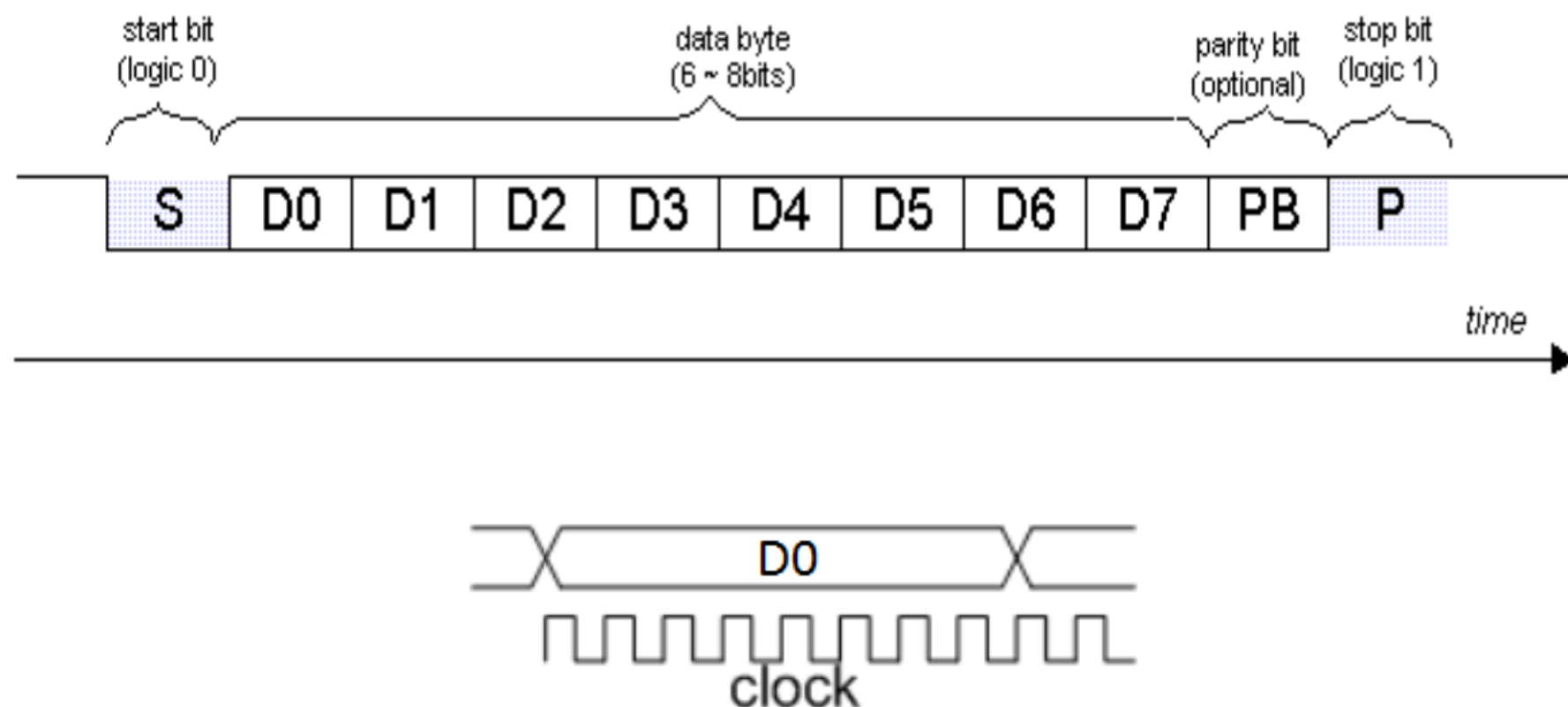
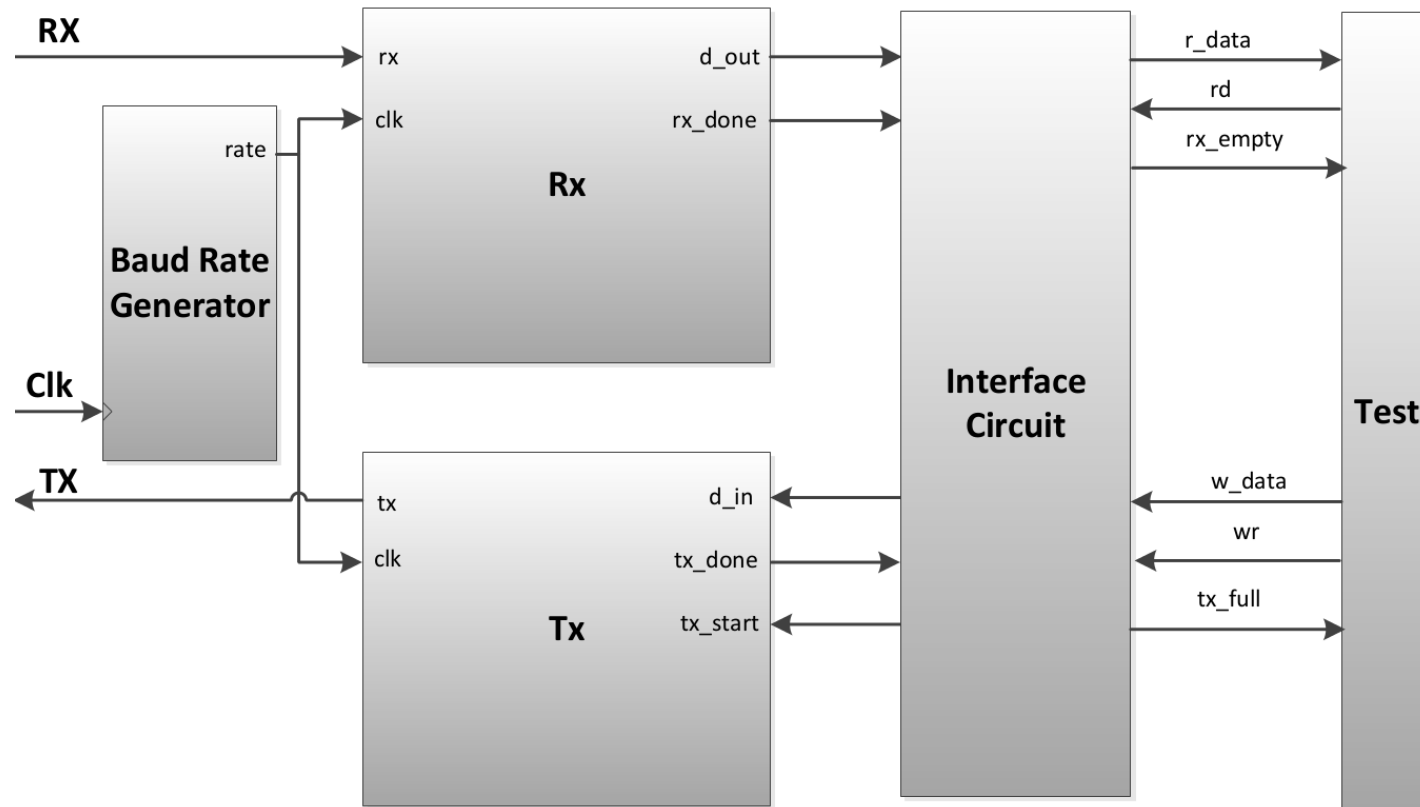


Diagrama de bloques



Trabajo practico: UART

Test del modulo:

El submódulo test, debe enviar por Tx, lo mismo que recibe por Rx

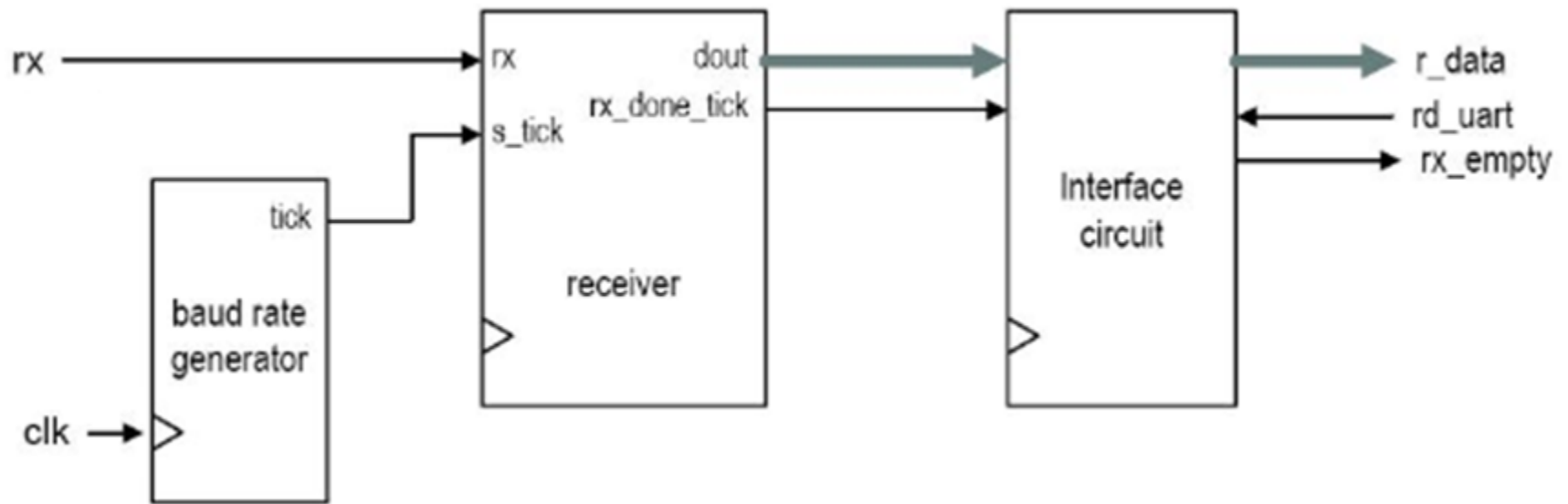
UART: Sobremuestreo

- Es practica común utilizar una velocidad de muestreo de 16 veces el baud rate deseado. Lo que significa que cada bit serial es muestreado 16 veces
- Este esquema utiliza los bits muestreados para estimar el punto medio de cada bit transmitido.

UART: Explicación

- **Rx:** Es el modulo encargado de obtener los datos recibidos por medio de sobremuestreo.
- **Tx:** Es el modulo encargado de serializar y transmitir los datos.
- **Baud rate genertor:** Modulo que genera los “ticks” de muestreo.
- **Interface circuit:** Modulo que provee la interfaz (Buffer y registro de estado) entre la UART y el sistema que la utilice.

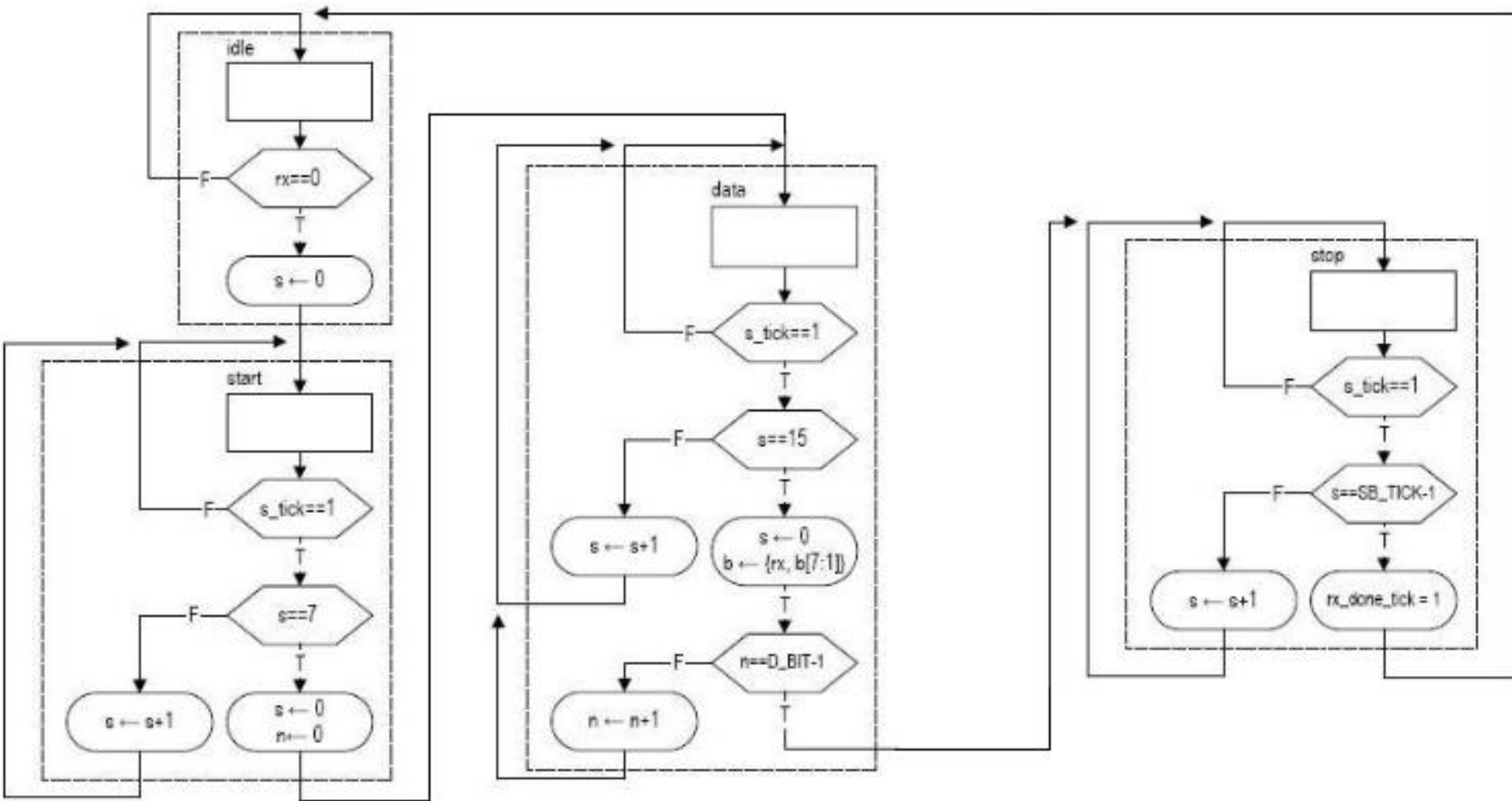
Receptor UART



Secuencia de estados Rx

- Asumiendo N bits de datos, M bits de Stop.
 - 1) Esperar a que la señal de entrada sea 0, momento en el que inicia el bit de Start. Iniciar el Tick Counter.
 - 2) Cuando el contador llega a 7, la señal de entrada está en el punto medio del bit de Start. Reiniciar el contador.
 - 3) Cuando el contador llega a 15, la señal de entrada avanza 1 bit, y alcanza la mitad del primer bit de datos. Tomar este valor e ingresarlo en un shift register. Reiniciar el contador.
 - 4) Repetir el paso 3 $N-1$ veces para tomar los bits restantes.
 - 5) Si se usa bit de paridad, repetir el paso 3 una vez mas.
 - 6) Reperir el paso 3 M veces, para obtener los bits de Stop.

Diagrama de estados Rx



Baud Rate Generator

- Genera un Tick 16 veces por Baud Rate
- Si baud rate es 19.200 ciclos por segundo, la frecuencia de muestreo debe ser $19.200 * 16 = 307.200$ ticks por segundo. Si el clock de la placa es 50 Mhz, hay que generar un tick cada 163 ciclos de reloj.

$$\frac{Clock}{BaudRate * 16} \cong 163$$

- El Baud Rate Generator es un contador módulo 163.

UART: Interface circuit

- Tiene 2 funciones:
 - Provee un sistema de dato valido
 - Provee un buffer entre el sistema que utiliza la UART y la misma
- Existen por lo general 3 esquemas:
 - Flag FF
 - Flag FF y buffer de una palabra de capacidad
 - Buffer FIFO

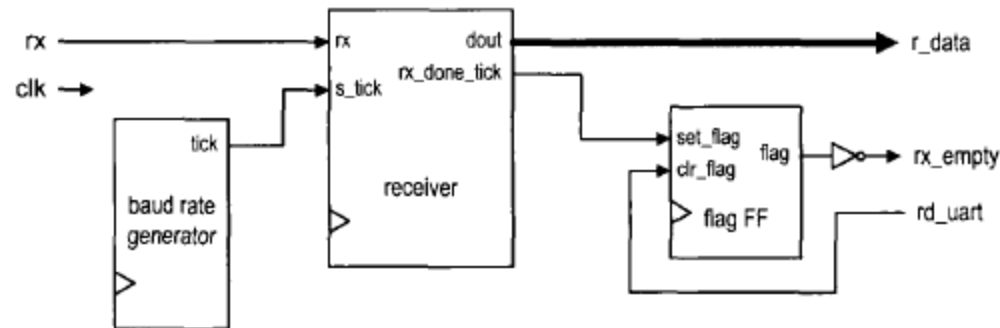
Interface circuit: Flag FF

- Contiene 2 señales
 - set_flag conectada a tx_ready_tick → El sistema chequea este flag (pooling) para ver cuando un dato ha arriado
 - clr_flag → se activa 1 ciclo despues de que un dato valido ha llegado
- Para evitar problemas de *overrun* se puede agregar un buffer de dimension 1

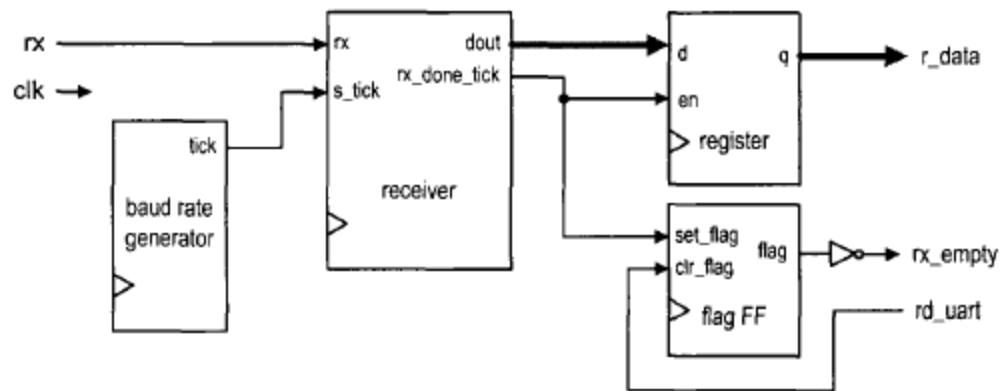
Interface circuit: Buffer FIFO

- Mejor protección contra condiciones de *overflow*
- rx_ready_tick es conectada a la señal **WR** de la FIFO.
- El sistema que utiliza la UART activa la señal **RD** de la FIFO para leer un dato
- Puede utilizar la señal *empty* de la FIFO para saber si hay datos que leer

Interface circuit: Diagramas de bloque (esquemas con flag FF)

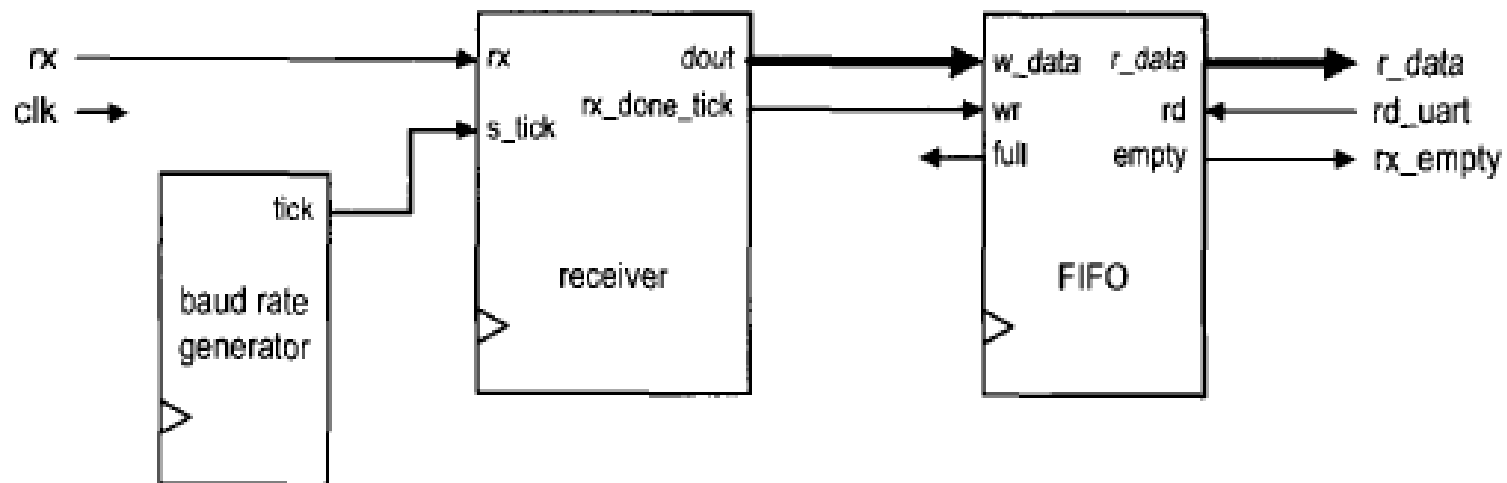


(a) Flag FF



(b) Flag FF and one-word buffer

Interface circuit: Diagramas de bloque (esquemas con FIFO)



(c) FIFO buffer

UART: Subsistema Tx

- Es esencialmente un *shift-register* que funciona a una velocidad determinada
- Dicha velocidad puede estar determinada por 1 ciclo de tick enable generado por el Baud Rate Generator dado que no se necesita sobremuestreo (por lo general se utiliza el mismo generador de ticks que el subsistema de Rx y se mantiene un contador interno para adaptar a las necesidades de Tx)
- Un bit es desplazado fuera del shif-register una vez cada 16 ticks del Baud Rate Generator.
- El diagrama ASM es similar al del caso Rx

TP: UART

