

TEMA 5: Entrada-salida física

Contenidos

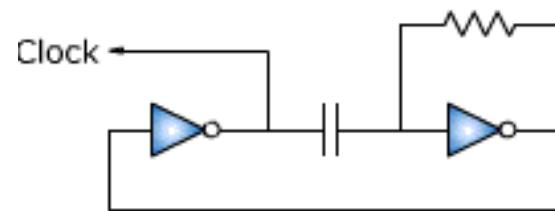
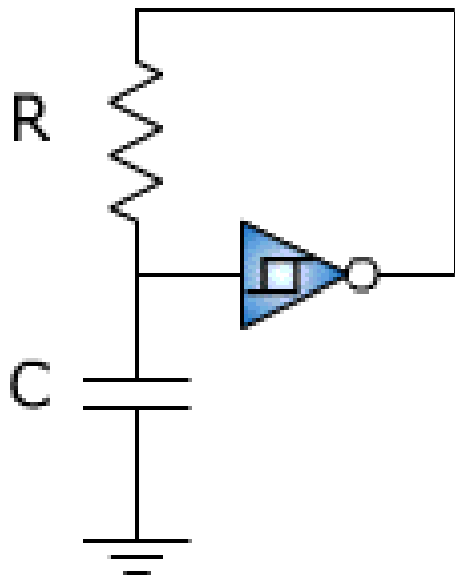
1. Sincronización I. Señales de sincronización
2. Sincronización II. Contadores internos.
3. Sincronización III. Instrucciones de sincronización
4. Interfaces de E/S I. Introducción
5. Interfaces de E/S II. Circuitos básicos
6. Interrupciones. Gestión. PIC y APIC
7. Buses. Introducción.

Objetivo: Métodos y procedimientos de E/S. Buses de comunicación internos y externos.

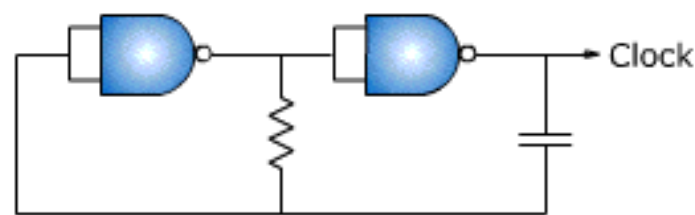
1. Señales de Sincronización

- Objetivo de sincronizar: Hacer posible que diferentes procesos/dispositivos actúen al mismo. Por ejemplo
 - Transmitir y recibir datos
 - Producir efectos simultáneos
 - Aumentar el rendimiento del sistema
- El elemento fundamental es el reloj: sistema digital capaz de producir una señal digital de frecuencia constante (señal de reloj, Clk)

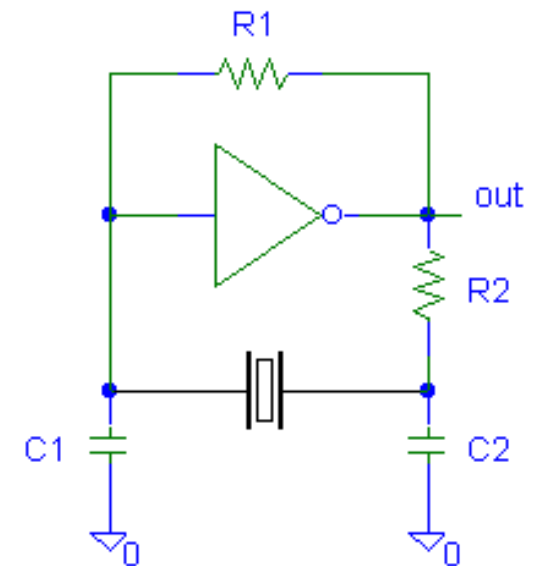
- Dentro de la señal de reloj existe un flanco activo, es que indica el momento en que actúan los procesos



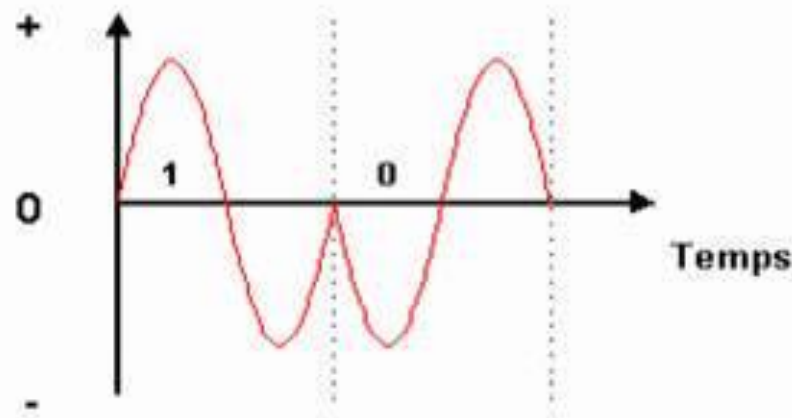
(b)



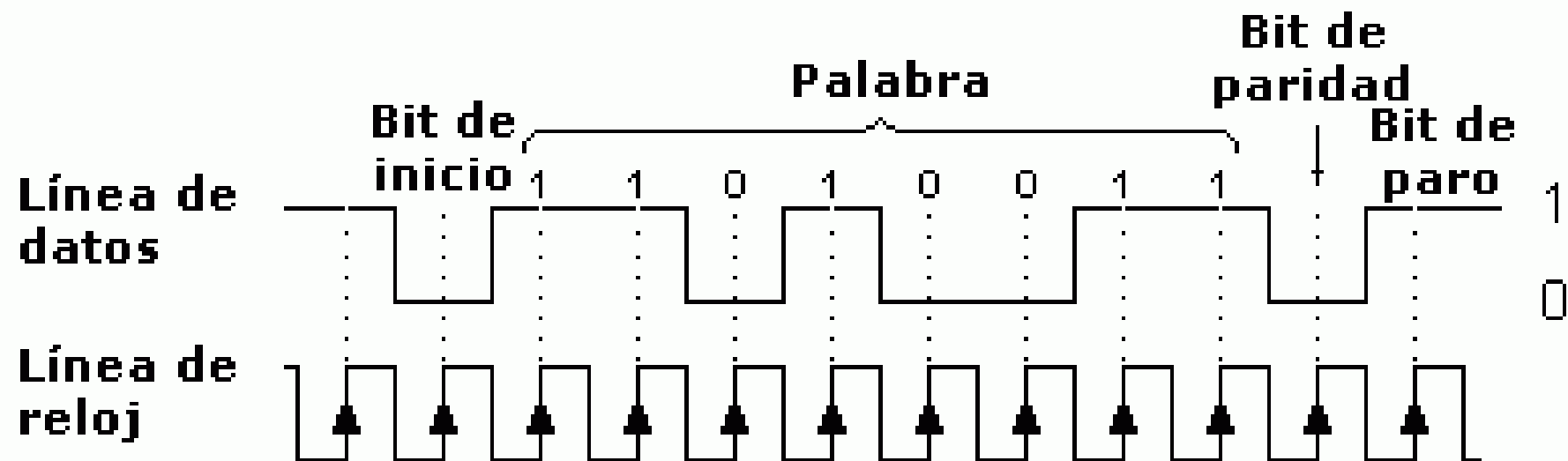
(c)



- La señal de reloj se degrada según separamos a fuente del destino. Depende de las características eléctricas de la línea y de la distancia. Tiende a convertirse en una senoide.
- Para sincronizar sistemas distantes se utilizan los cambios de fase de una señal sinusoidal de frecuencia y amplitud ajustada a las características de la línea (ADSL)

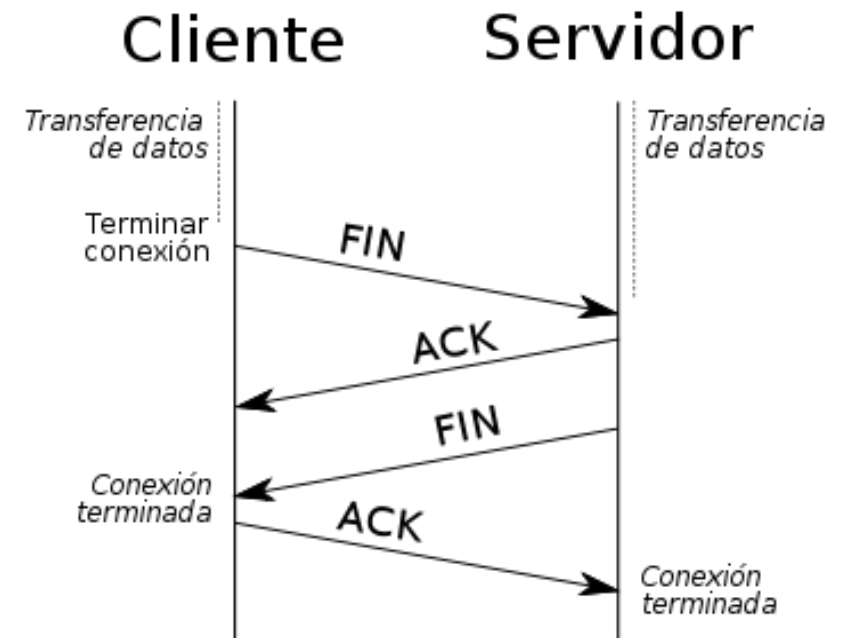


- En sistema donde **no sea posible** utilizar una señal de reloj común (asíncronos), los relojes se sincronizan mediante secuencias previas de bits.



- **Señales de handshake:** Es necesario gestionar qué dispositivo envía o recibe información. En cada sistema (bus, periférico o microprocesador) se llaman de diferente manera. Es necesario un protocolo de utilización

- Requerimiento REQ
- Aceptación ACK
- Validación VAL, Ready
- Diferenciación de datos / direcciones (DIR/DAT)
- Parada: STOP, HALT
- Reinicialización: RESET, CLEAR
- Error
- Alimentación



- En buses inteligentes, las señales se sustituyen por mensajes

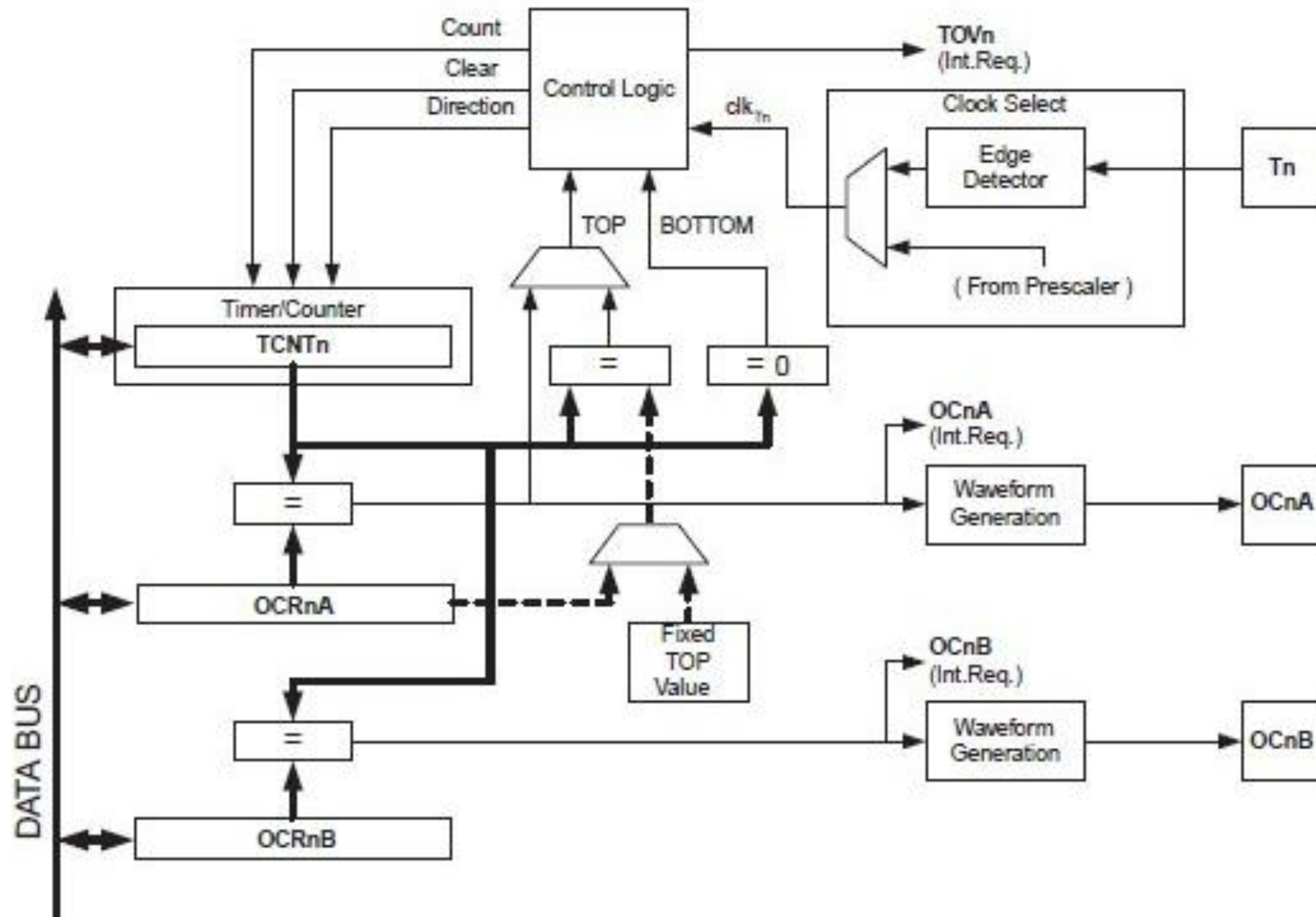
- Tipos de handshaking

- Pulsos. El emisor genera una secuencia de bits se sincronización que permiten al receptor sincronizarse y recibir correctamente los datos. Se utiliza en transmisiones unidireccionales y a baja velocidad
- Simple (REQ/ACK): El emisor solicita en envío (REQ) y el receptor los acepta (ACK) La sincronización se realiza a través de estas dos señales.
- Doble (REQ/ACK/VAL) Similar al anterior, pero el receptor envía una señal de validación final, una vez recibe correctamente el dato. La no generación de VAL, tras un tiempo predeterminado, genera un error y la nueva emisión del dato.

2. Contadores internos

- Los contadores internos tienen dos funciones:
 - Generan señales que controlan intervalos de tiempo. Utilizan el modo de funcionamiento síncrono, siendo el reloj del sistema o división, la base de tiempo empleada.
 - Cuentan el número de datos o bloques de datos que se envían o se reciben. Utilizan el modo de funcionamiento asíncrono. Cada dato o bloque de datos generará una señal que incrementará el contador.
 - Se utilizan como punteros, combinando las dos formas anteriores

Esquema de un contador interno



- Señales de control:
 - Count: reloj de cuenta
 - Clear: borrado del registro/contador
 - Direction: incremento/decremento
- Buses:
 - Salida al bus (bidireccional). Se escribe/lee a través de un puerto o dirección de memoria
 - Salida interna: para interactuar con los comparadores
- Registros auxiliares: Se utilizan para almacenar datos que se compararán con la secuencia generada por el contador.
 - OCRna, se utiliza para fijar un tope superior en la secuencia de cuenta.
 - OCRnb, es de carácter general

3. Instrucciones de sincronización

- Instrucciones clásicas:
 - NOP: No hacer nada. Pierde un ciclo
 - Halt: Detiene la ejecución hasta que se produce una interrupción.
 - Wait (t^0): Detiene la ejecución durante un tiempo determinado
- Gestión de interrupciones.
 - Activar (EI) o desactivar (DI) interrupciones
- Comprobación de Bits en registros, direcciones de memoria o de E/S
 - Bit (direcc/reg) -> registro de indicadores Z. Permite la ejecución de saltos condicionales JMPZ o llamadas a rutinas CALLZ

Tareas a realizar:

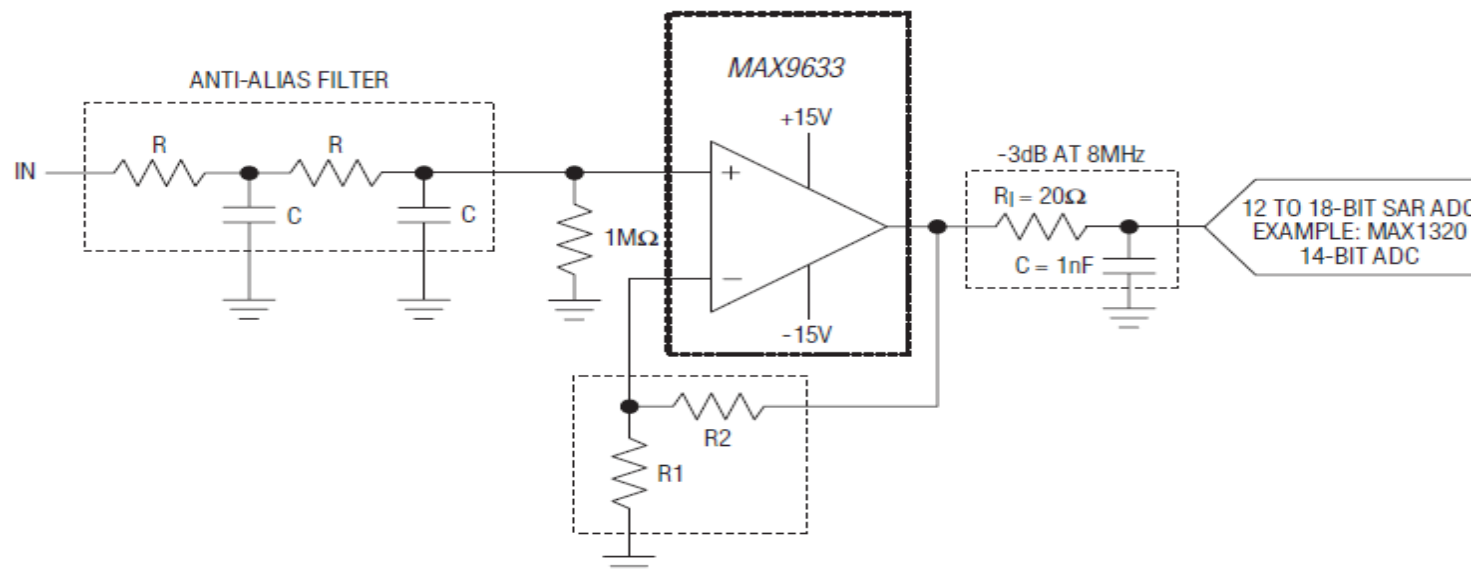
Tomemos como base el circuito desarrollado en la primera práctica en sus diferentes fases. Contestar:

- ¿Qué señales de sincronización existían?
- ¿Qué señales de control estaban disponibles?
- ¿Existían registros internos en los dispositivos? Cuáles y cómo se utilizaban
- No implementó prioridades en la comunicación. ¿Cómo podríamos implementarla de forma que fuese configurable?
- No se consideró que alguno de los dispositivos estuviera “ocupado” cuando se requiriera comunicación. Cómo podríamos implementar esta característica.

4. Interfaces de E/S. Introducción

- Son los elementos (hardware – Software) que permiten la comunicación de entrada y salida del computador, gestionando las señales/códigos necesarios
- Dentro de la Jerarquía de buses o chipsets, son el elemento que permite enlazar/adaptar el exterior e interior del computador

Punto de vista físico/electrónico: Las interfaces de E/S establecen la comunicación entre la CPU y el proceso, filtrando, adaptando y codificando de forma comprensible (para la CPU), las señales procedentes de los elementos de entrada, y decodificando y amplificando las señales generadas durante la ejecución del programa antes de enviarlas a los elementos de salida.



Es interesante considerar:

- Tipo de señal: analógica, digital (binaria o no)
- Medio por el que se transmite la señal: luz, tensión, electricidad, posición, presión, etc.
- Condiciones de la señal (limitaciones): debidas al medio y al tipo de señales que se va a usar:
 - Tensión binaria (V_h , V_l);
 - Intensidad lumínica binaria: (I_v1 , I_v0)
 - Ángulo en sistema ternario (α_2 , α_1 , α_0)
 - Colores en sistema quinario (RGB_0 , RGB_1 , RGB_2 , RGB_3 , RGB_4 ,)
 - Fase de onda
 - Etc.

Tipo de transmisión:

- Paralelo: una línea por bit del dato y todos simultáneos
- Serie: todos los datos a través de la misma línea y multiplexados en tiempo
- Mixtos, que reúnen ambas características

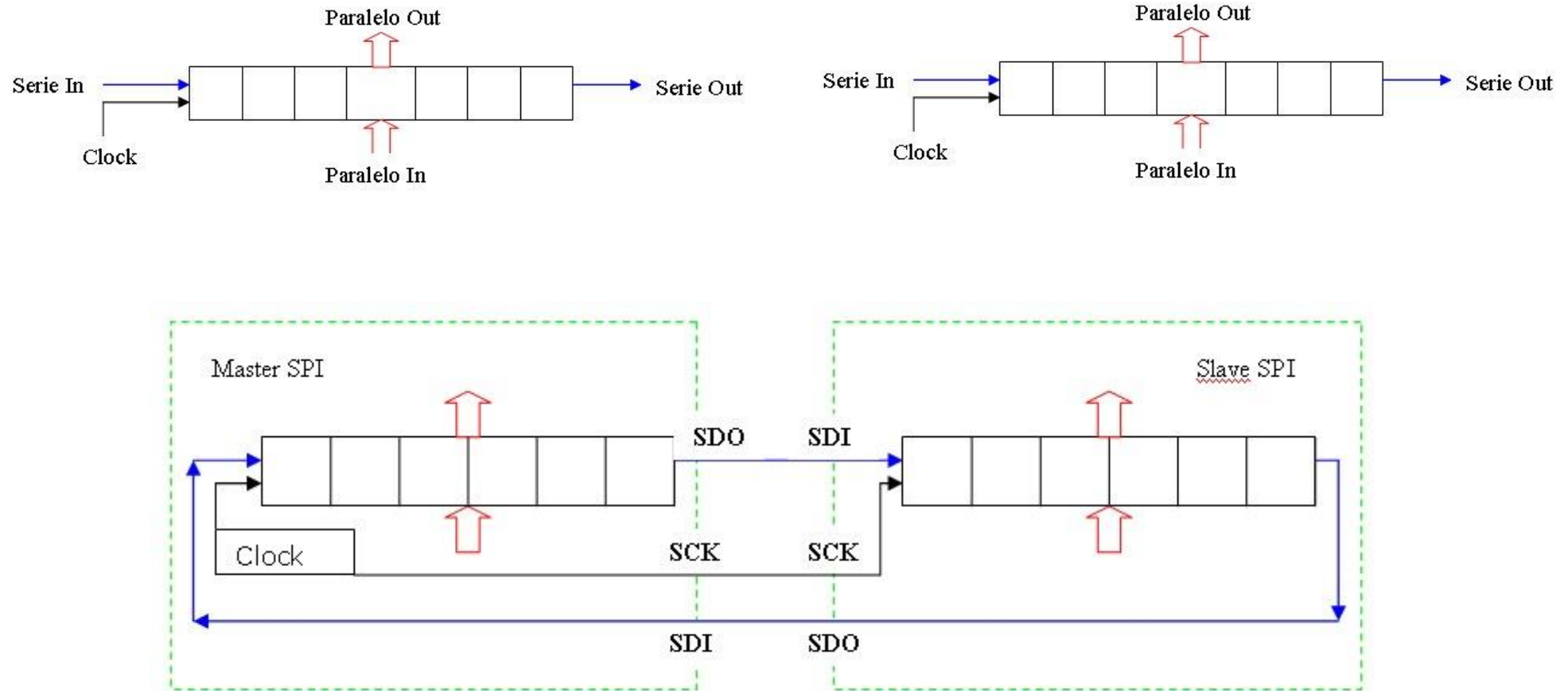
Codificación (tramas), formatos, protocolos, topología del bus...

Elementos que intervienen en los interfaces. Capas/niveles:

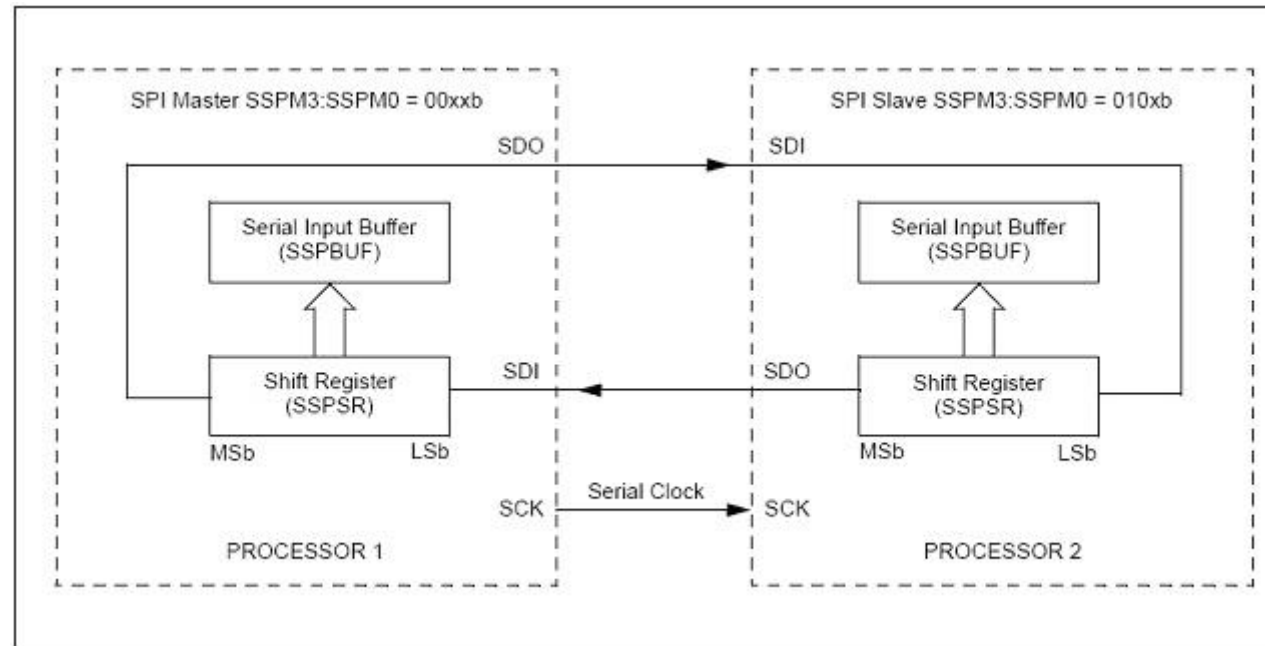
- Registros paralelo o serie, directamente desde el bus de E/S del sistema.
- Circuitos especializados en la gestión de comunicación paralelo (PIO/A) o serie (SIO/A). (Punto a punto)
- Controladores de buses (elementos compartidos)
- Controladores de Jerarquía (buses compartidos)
- Chipsets

A mayor complejidad en el sistema, se utilizarán más niveles. Niveles que tienen parte hardware y software.

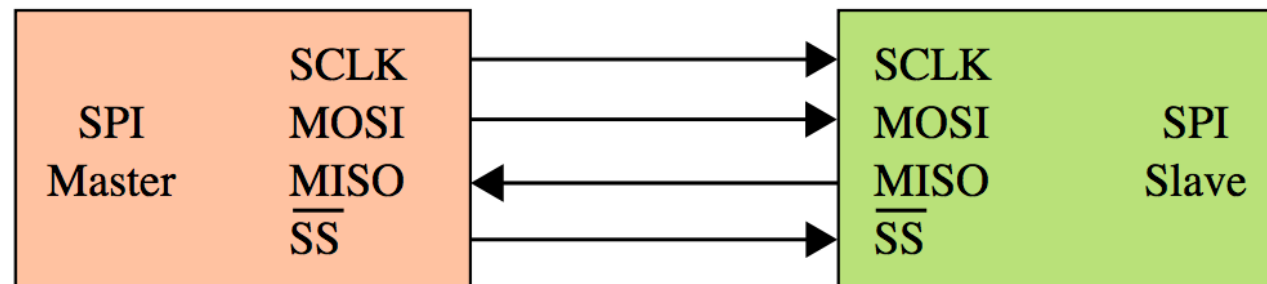
Ejemplo de bus: SPI (Serial Peripheral Interface)

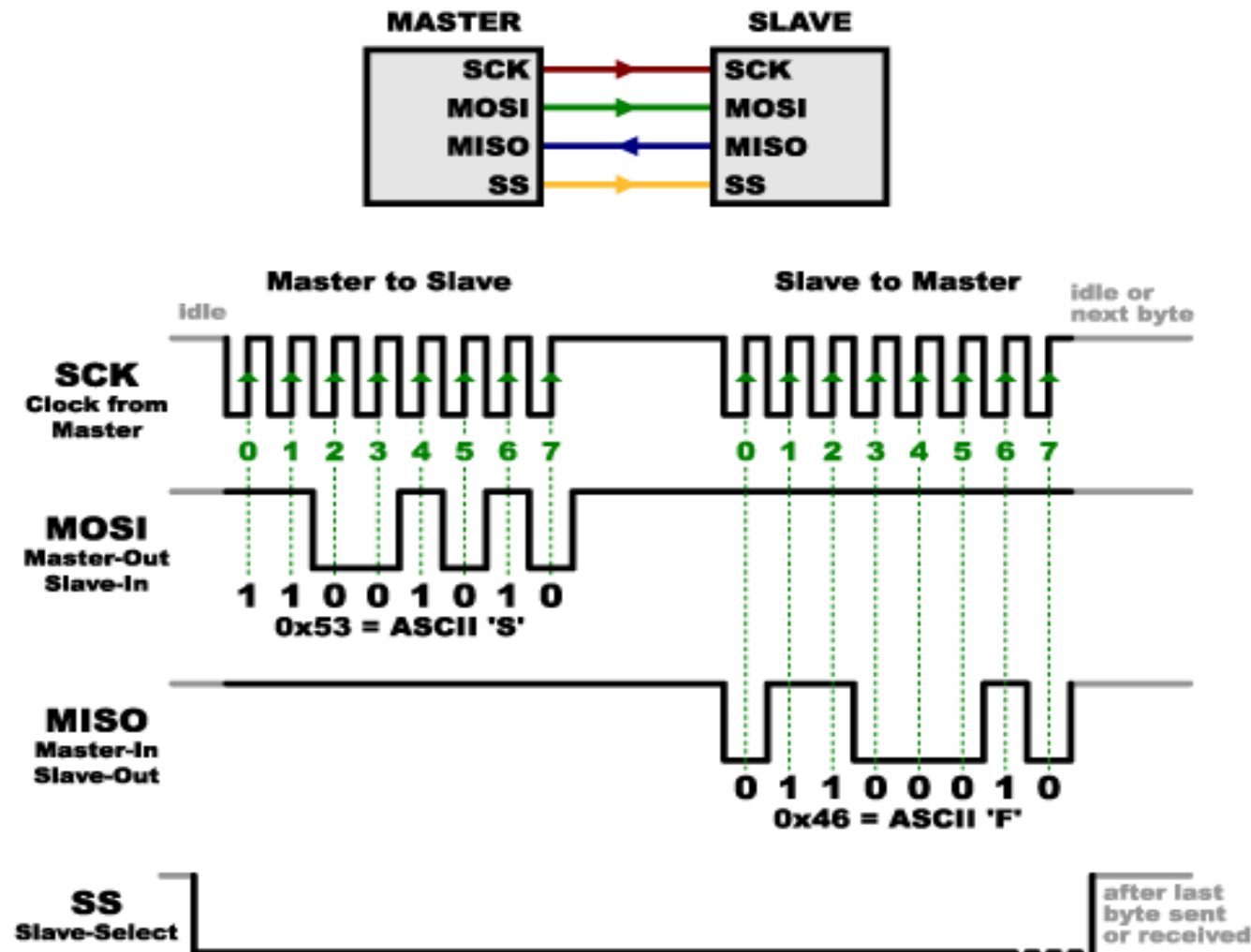


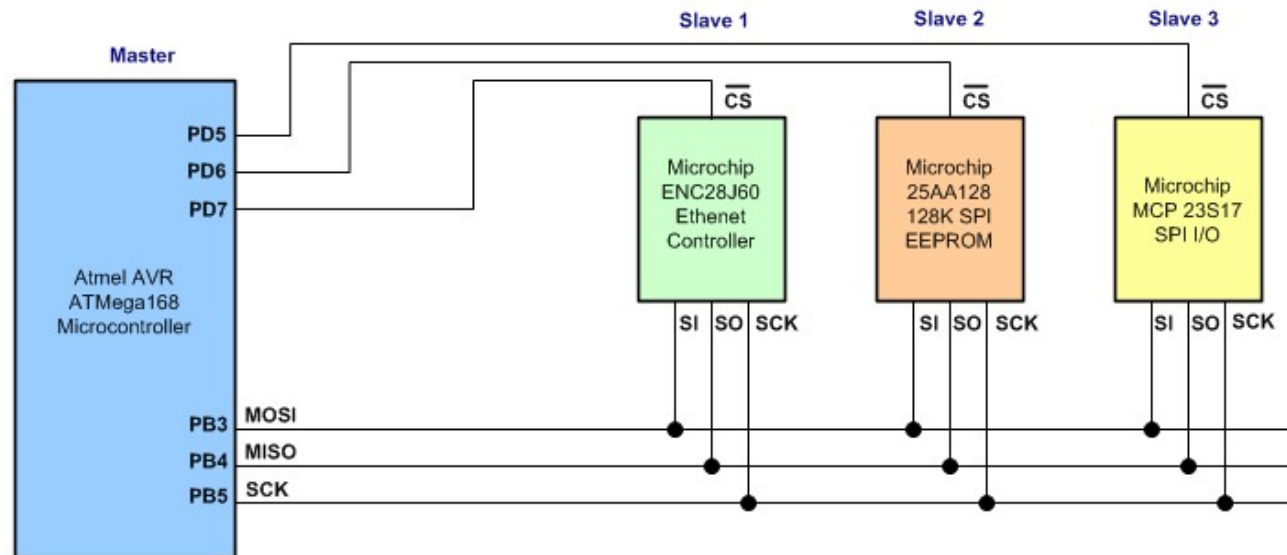
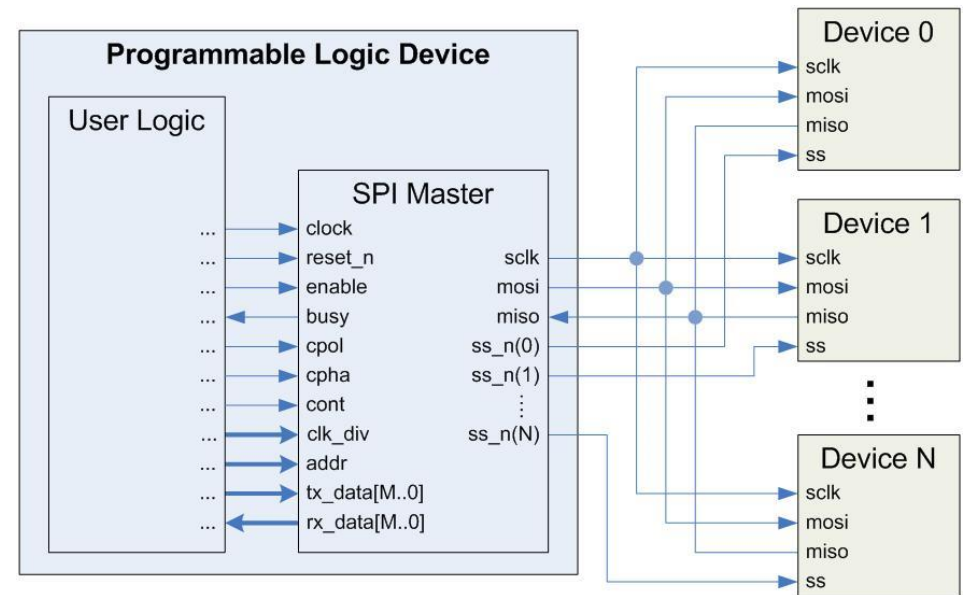
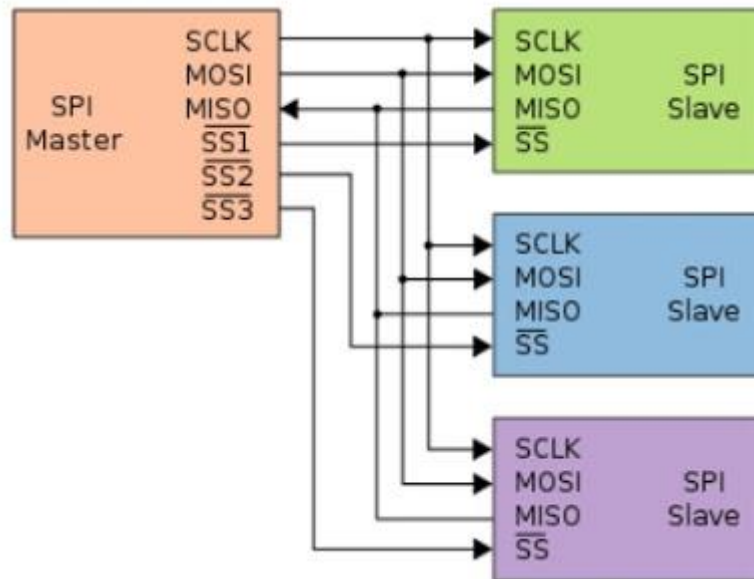
Sistema Maestro-esclavo



Para más de un dispositivo esclavo (SS):







Typical SPI Master with Multiple SPI Slave Device Connection

Tareas a realizar

Objetivo: conocer un poco más del bus SPI y de sus capas:
Responder:

- Límites físicos del bus
- N° de periféricos que puede controlar
- ¿Qué sucede si se producen errores al transmitir/recibir? ¿Cómo se gestionan?
- ¿Qué sucede si el dispositivo receptor no está disponible para recibir?
- ¿Qué sucede si el dispositivo esclavo necesita transmitir un dato al maestro?
- ¿Existe forma de enviar un mismo dato a varios esclavos?
- ¿Existe la posibilidad de tener varios dispositivos maestros? ¿Cómo se gestionaría esta circunstancia?