

EI1062 – IR2162

Diseño de sistemas empuotrados y de tiempo real

Tema 4 – Comunicaciones
en sistemas empuotrados (I)

Grado en Ingeniería Informática

Grado en Inteligencia Robótica

[Introducción]

- Sistemas empotrados locales
- Sistemas empotrados distribuidos
 - Comunicaciones entre controladores
 - Comunicaciones con sensores y actuadores
- Comunicaciones inalámbricas

[Buses]

- Si se usan los buses de direcciones y datos para conectar periféricos, da lugar a gran cantidad de pistas en el PCB y circuitería adicional de decodificación
- Los protocolos SPI (Motorola) e I²C (Philips) pretenden simplificar la conexión de periféricos locales al microcontrolador

[Buses - comunicaciones]

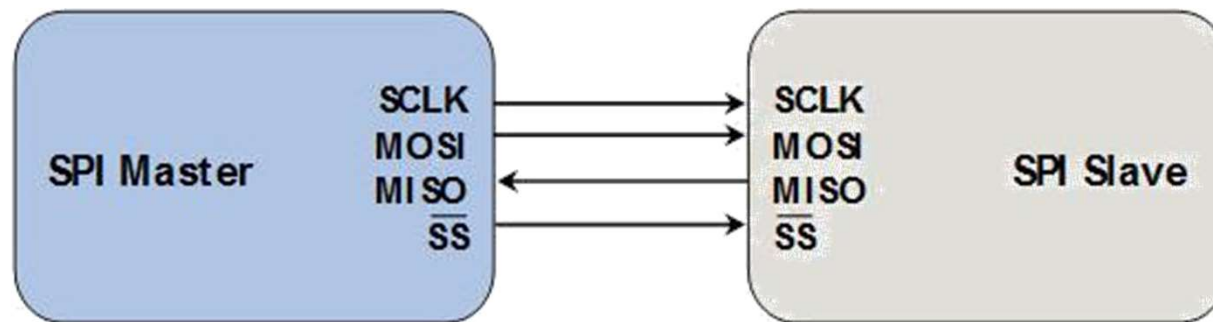
- SPI
- I²C
- 1-wire
- CAN
- RS232 (UART)
- WiFi
- Bluetooth

[SPI: Serial Peripheral Interface]

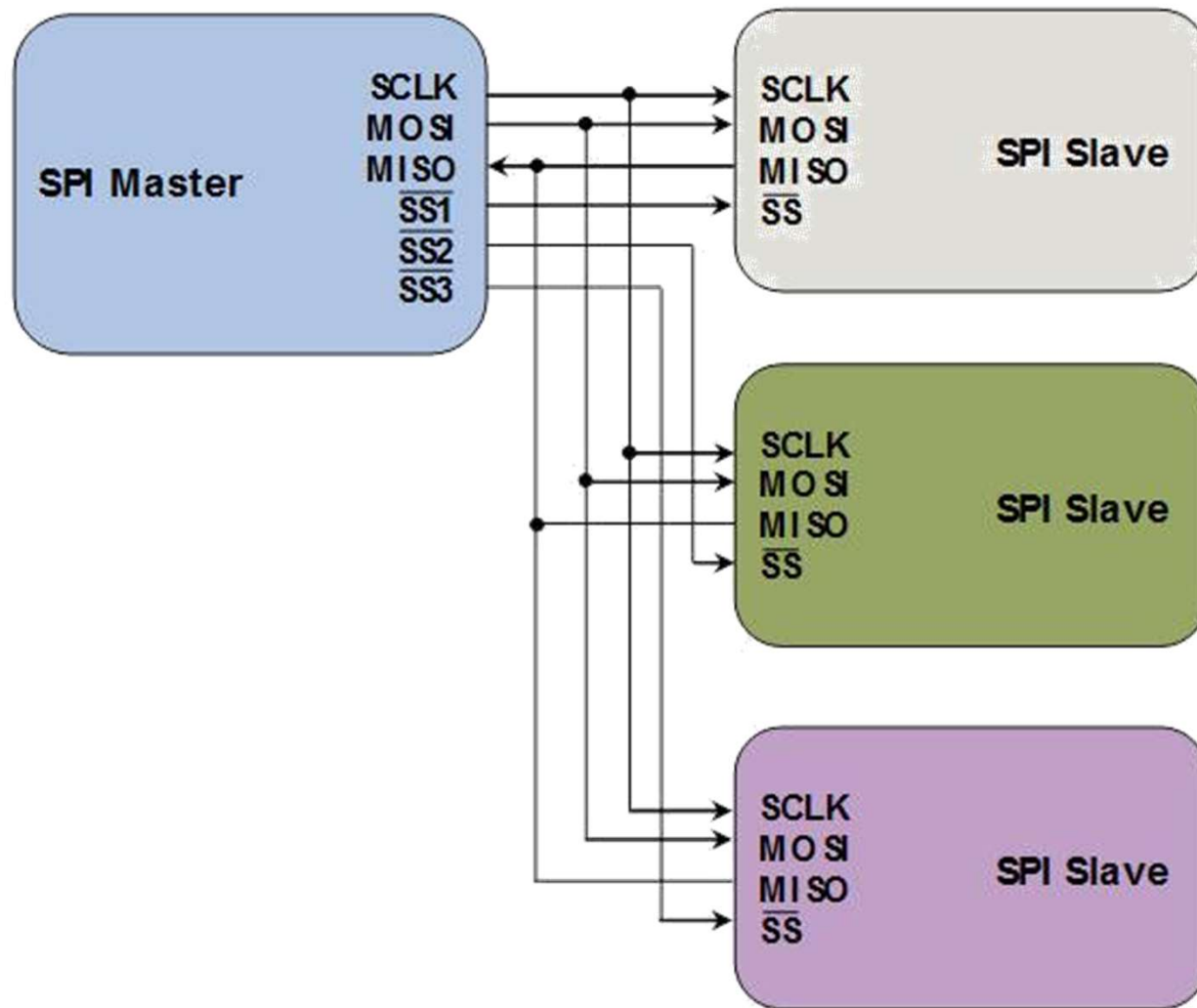
- SPI fue desarrollado por Motorola para el primer microcontrolador derivado del M68000 en 1979
- Define cuatro señales para la comunicación:
 - SCLK: Reloj de sincronismo
 - MOSI: Master output slave input
 - MISO : Master input slave output
 - SSx: Slave select X
- El uso de las líneas SSx permite multiplexar las restantes

[SPI: Serial Peripheral Interface]

- Modelo de bus para la comunicación entre un master y un solo esclavo



[SPI: Serial Peripheral Interface]

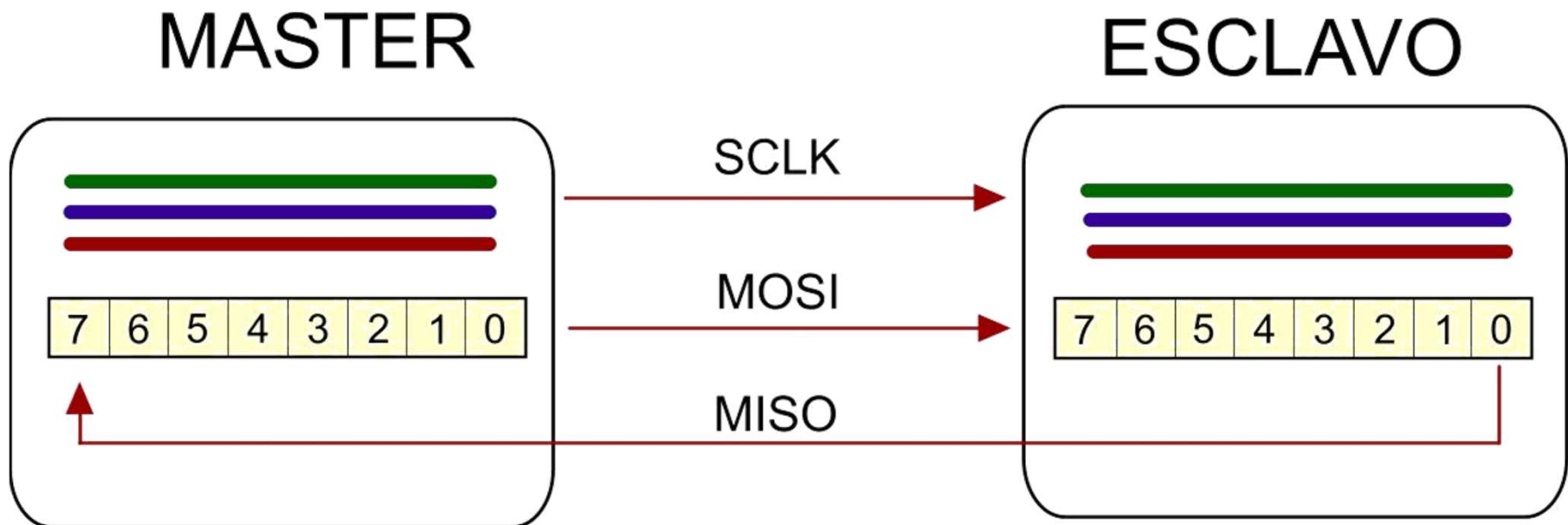


[SPI: Serial Peripheral Interface]

- Dos de las tres líneas que participan en la transmisión son las que transfieren los datos (una en cada dirección) y la tercera es la señal de reloj de sincronismo
- Algunos dispositivos solo pueden ser transmisores y otros solamente receptores, aunque los transmisores suelen poder recibir

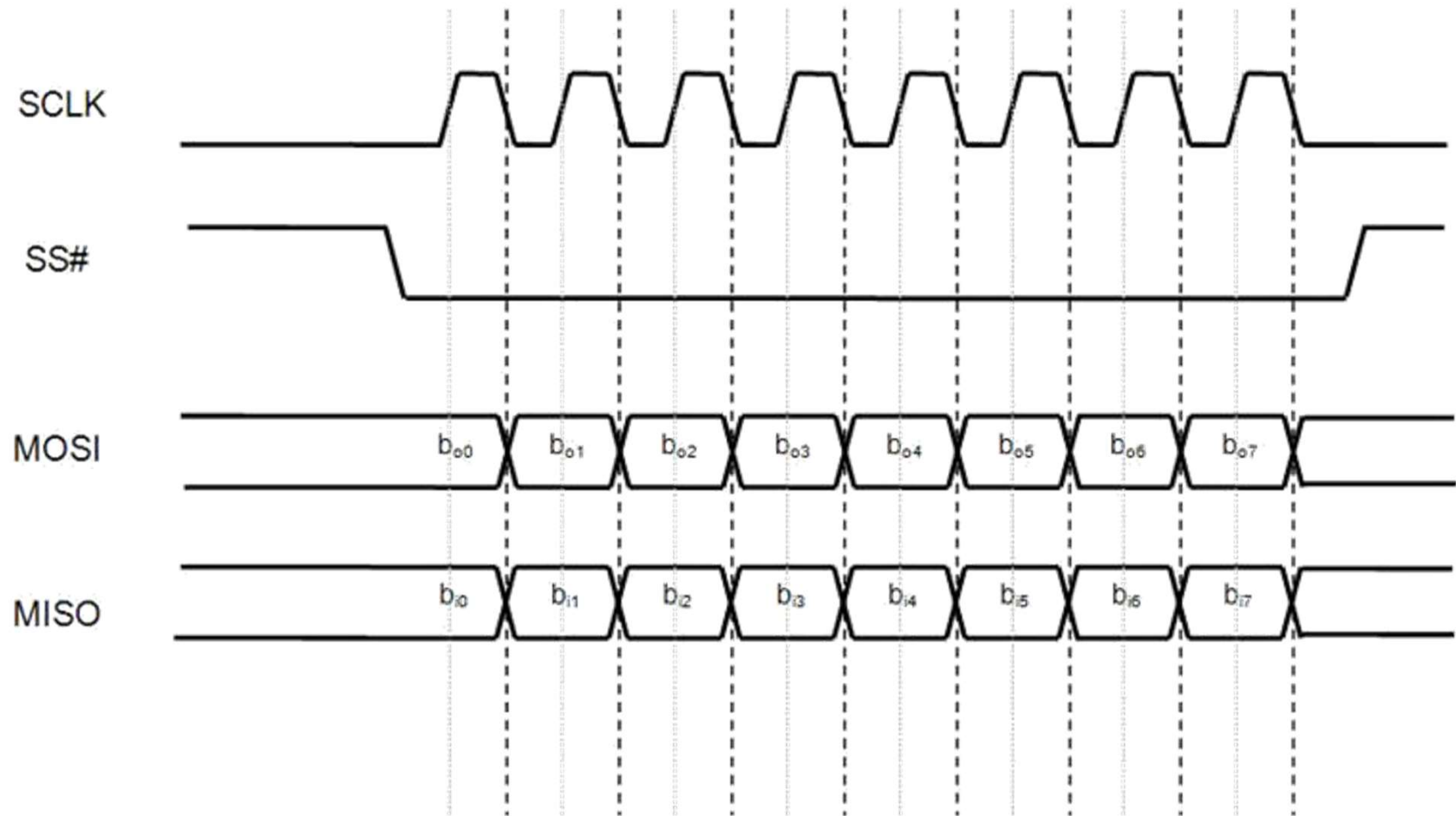
[SPI: Serial Peripheral Interface]

- Orden de transmisión de los bits de cada palabra



[SPI: Serial Peripheral Interface]

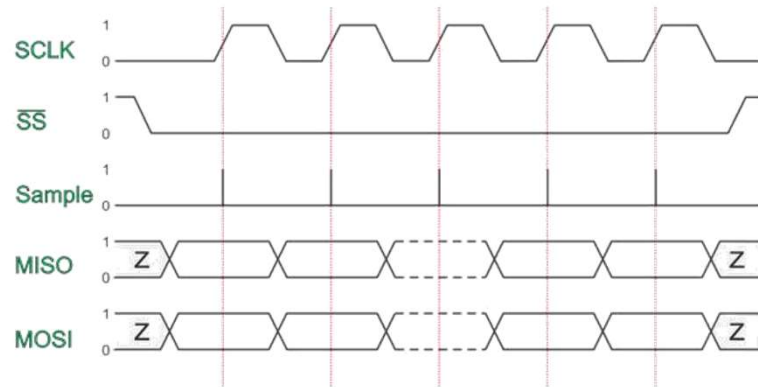
■ Funcionamiento típico



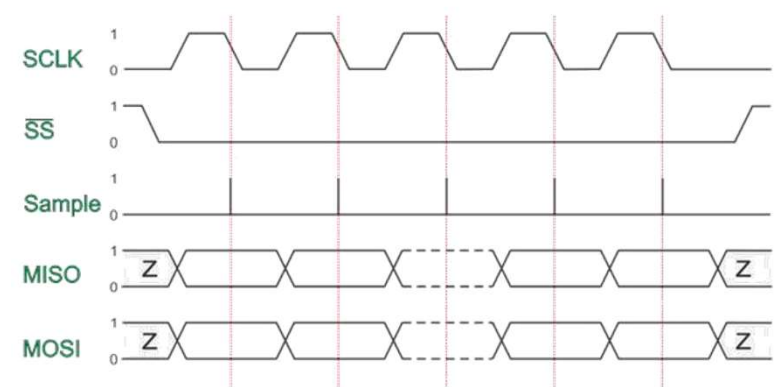
[SPI: Serial Peripheral Interface]

■ Modos de sincronismo

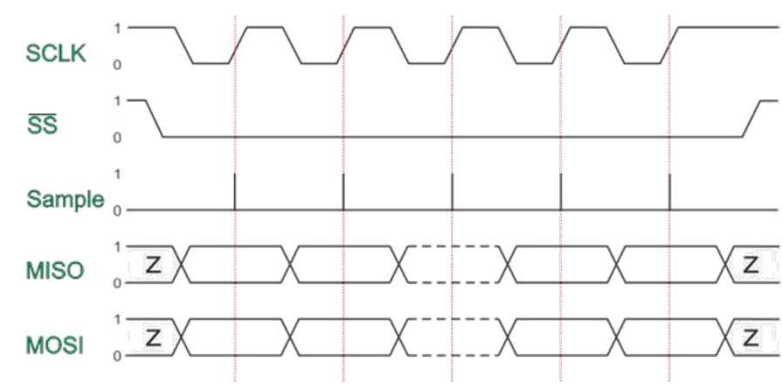
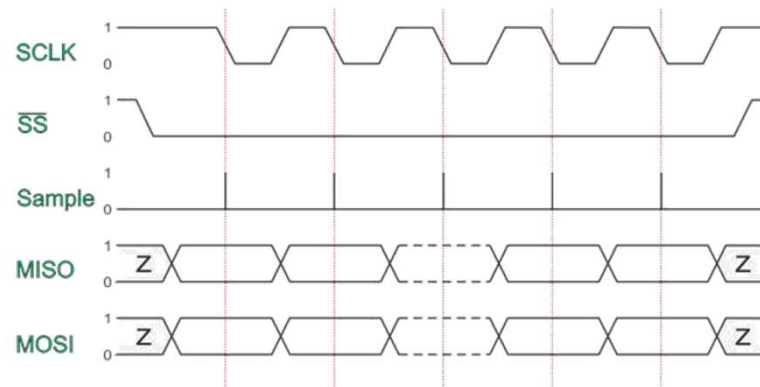
CPHA = 0



CPHA = 1



CPOL = 1



[SPI: Ventajas]

- Comunicación full duplex
- Protocolo flexible sin tamaño preestablecido
- No hay necesidad de arbitraje
- Los esclavos no necesitan reloj
- Se puede implementar solo una de las dos funcionalidades (emisor – receptor)

[SPI: Inconvenientes]

- Usa mayor número de pines
- Una línea de selección por esclavo
- No hay control de flujo por hardware
- No hay señal de aceptación. El master puede estar enviando sin que se reciba
- No es fácil conectar varios masters a un solo bus

[Buses - comunicaciones]

- SPI
- I²C
- RS232 (UART)
- 1-wire
- CAN
- WiFi
- Bluetooth

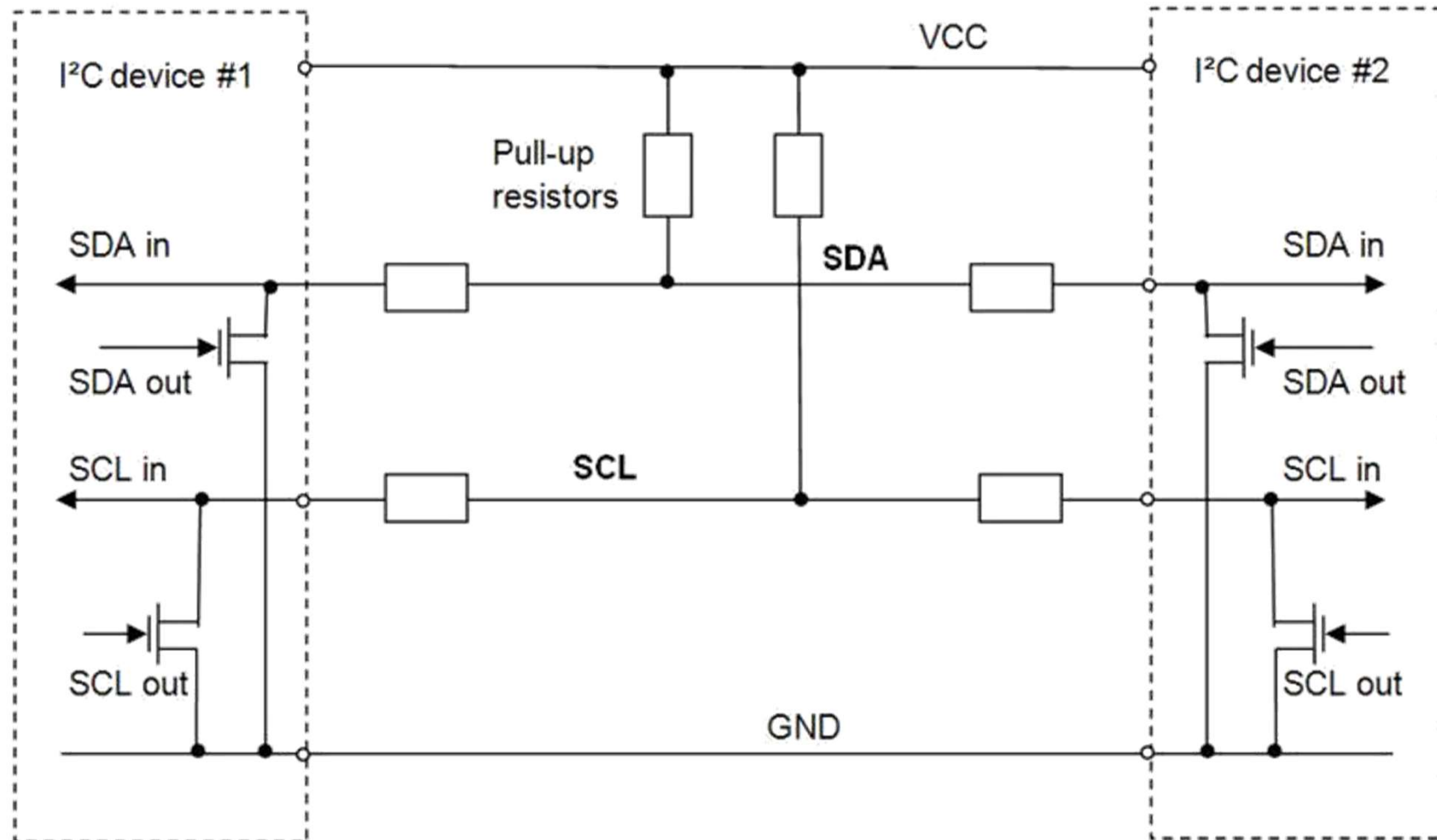
I²C: Inter-Integrated Circuit

- Desarrollado por Philips en 1982 para interconectar los módulos de un TV
- Solamente usa dos cables (serial data – SDA y serial clock – SCL)
- Inicialmente soportaba una velocidad de 100 Kbps que se ha incrementado hasta 3'4 Mbps (1998)

I²C: Inter-Integrated Circuit

- Cualquier número de dispositivos puede conectarse al bus usando el protocolo:
 - Dirección de esclavo de 7 bits
 - Bloques de datos de 8 bits
 - Bits de control: start, end, direction y acknowledgement
- La velocidad se elige entre 100 Kbps (standard) 400 Kbps (fast) y 3'4 Mbps (high speed)

I²C: Inter-Integrated Circuit



I²C: Inter-Integrated Circuit

- El primer módulo en transmitir se considera master y el resto esclavos. El proceso es el siguiente:
 - El master emite una condición START
 - El master emite el identificador del esclavo (ADDRESS) con la indicación de si se trata de una lectura o una escritura
 - El esclavo direccionado genera un ACKNOWLEDGE
 - Al recibir el ACKNOWLEDGE, el master transmite o recibe datos (DATA)
 - Una vez completada la transacción, el master transmite un STOP y el bus se libera

I²C: Inter-Integrated Circuit

Protocolo de comunicación

START	Slave address	Rd/nWr	ACK	Data	ACK	Data	ACK	STOP
1 bit	7 bits	1 bit	1 bit	8 bits	1 bit	8 bits	1 bit	1 bit

Trama de escritura

START	Slave address	0	0	Data	0	Data	0	STOP
1 bit	7 bits	1 bit	1 bit	8 bits	1 bit	8 bits	1 bit	1 bit

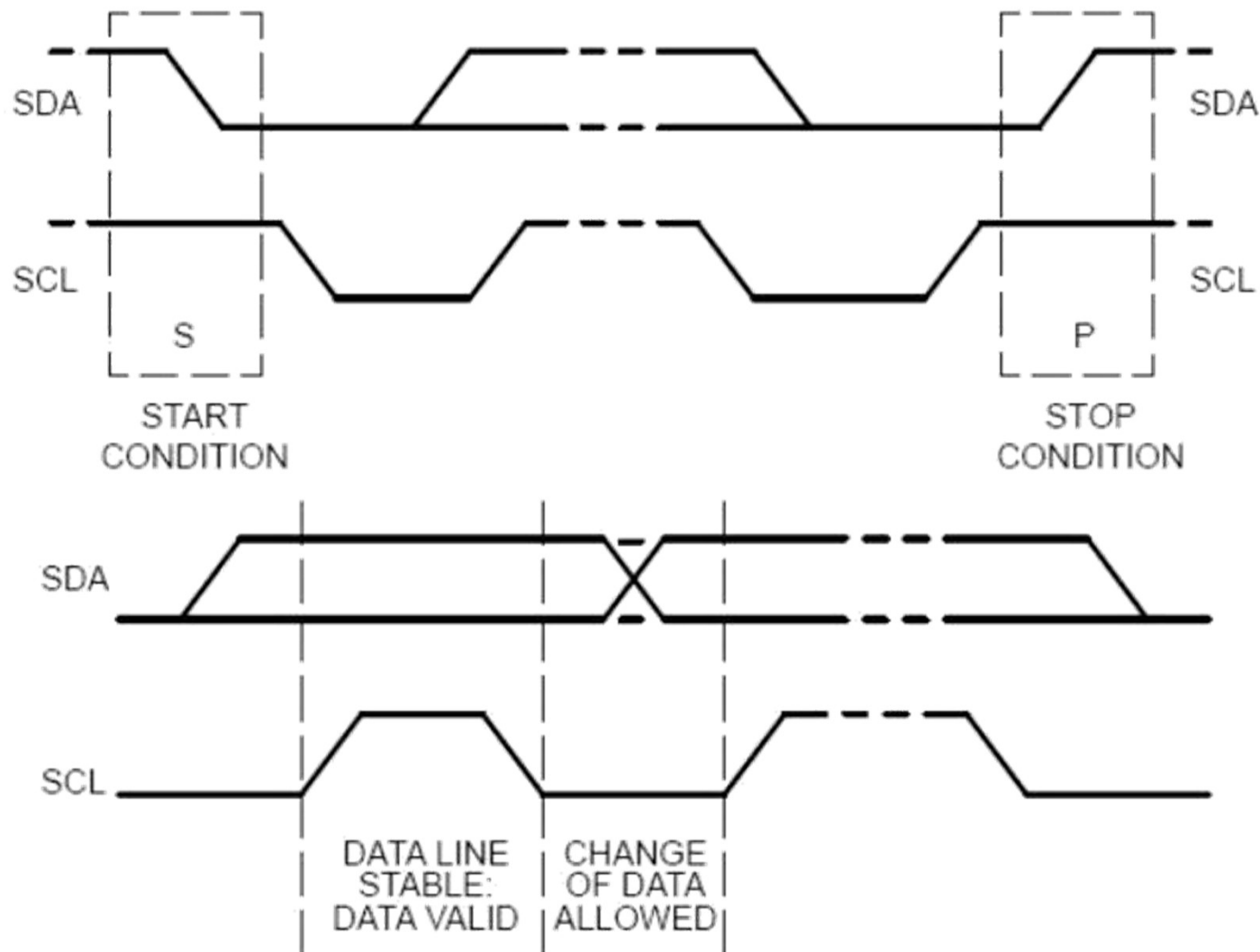
Trama de lectura

(*)

START	Slave address	1	0	Data	0	Data	1	STOP
1 bit	7 bits	1 bit	1 bit	8 bits	1 bit	8 bits	1 bit	1 bit

El master indica al esclavo que quiere finalizar la trama mediante un ACK a 1

I²C: Inter-Integrated Circuit



I²C: Inter-Integrated Circuit

- Al ser señales de colector abierto, no hay conflicto a nivel eléctrico
- Las posibles colisiones se solucionan haciendo que cada master monitorice el valor presente en el bus al tiempo que escribe.
- Los módulos no involucrados permanecen inactivos hasta la próxima condición STOP

I²C: Inter-Integrated Circuit

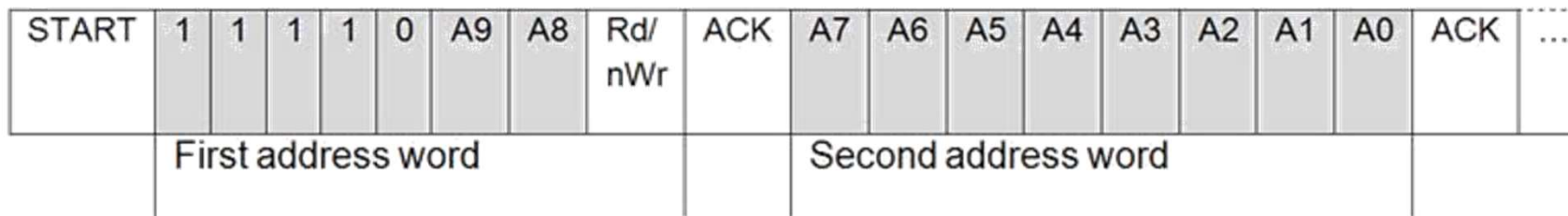
- Con 7 bits de dirección, hay 128 posibles dispositivos I²C
- Por un lado, algunos de estos 128 valores están reservados
- Por otro, para incrementar el número de dispositivos direccionables, se pueden usar direcciones de 10 bits

[I²C: Inter-Integrated Circuit]

Address	Purpose
0000000 0	General Call – addresses all devices supporting the general call mode
0000000 1	Start Byte
0000001 X	CBUS addresses
0000010 X	Reserved for different bus formats
0000011 X	Reserved for future purpose
00001XX X	High-speed Master code
11110XX X	10-bits slave addressing
11111XX X	Reserved for future purposes

I²C: Inter-Integrated Circuit

- Transmisión de una dirección de 10 bits



10 bits address:



I²C: Inter-Integrated Circuit

- Reducción de la velocidad del reloj
 - El esclavo mantiene la señal de reloj a nivel bajo para indicar al master que reduzca la frecuencia del mismo
- Alta velocidad
 - Antes de usar el modo 'high speed', el master debe emitir un comando a velocidad inferior para activar los buffers específicos de E/S de alta velocidad

[Buses - comunicaciones]

- SPI
- I²C
- 1-wire
- CAN
- RS232 (UART)
- WiFi
- Bluetooth

[1-wire]

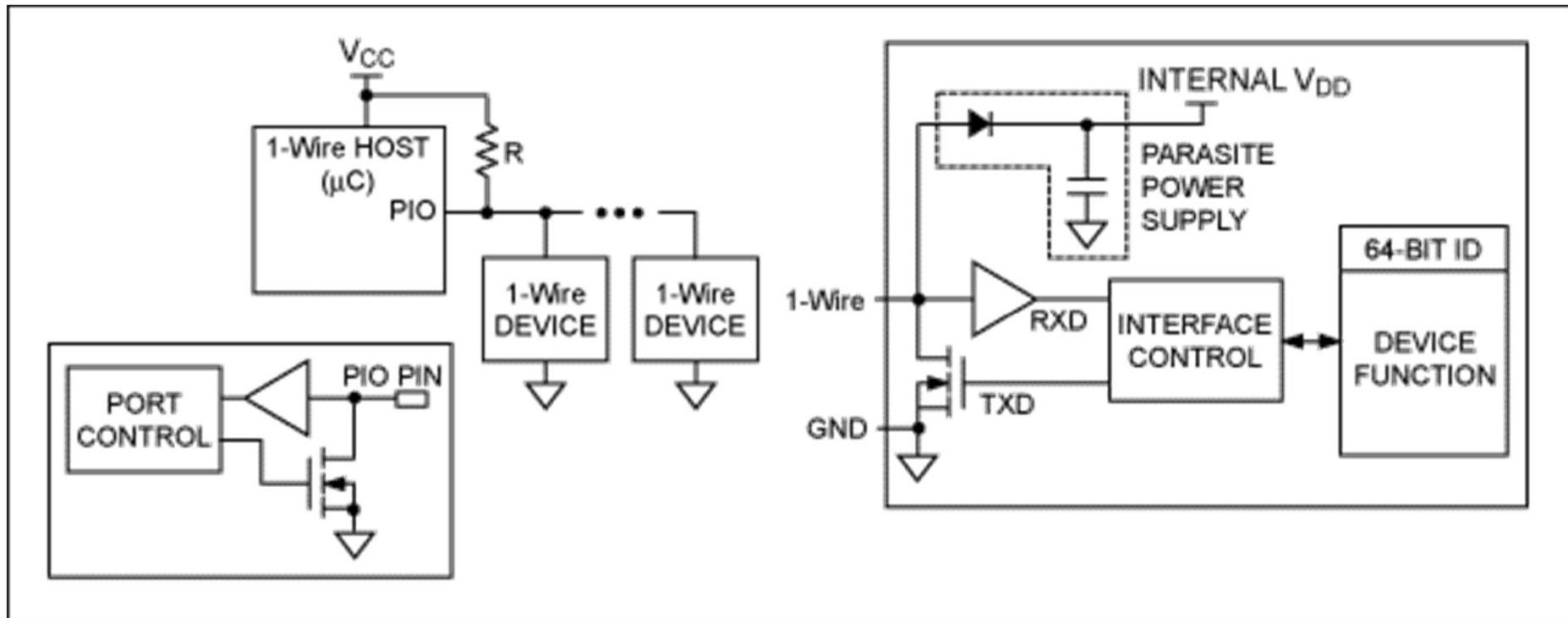
- Desarrollado por Dallas Semiconductor
- Sistema de comunicación de los dispositivos i-button



[1-wire: Descripción general]

- Es un protocolo de comunicación serie
- Un sistema 1-wire consiste en un master y uno o varios esclavos
- La comunicación se basa en el inicio de la comunicación por el master y la sincronización autónoma de los esclavos
- Eléctricamente, la línea presenta un pull-up y todas las salidas son a colector abierto

[1-wire: Descripción general]



[1-wire: Comunicación (1)]

- Un master inicia y controla la comunicación con varios esclavos
- Cada elemento posee un identificador único de 64 bits que se emplea como dirección de red
- Un subconjunto de 8 bits del identificador indica la familia a que pertenece el módulo 1-wire y su funcionalidad

[1-wire: Comunicación (2)]

- El voltaje empleado abarca el rango comprendido entre los 2'8 V y los 5'25 V
- Muchos dispositivos 1-wire no poseen terminal de alimentación. Toman la alimentación del bus (suministro parásito). En este caso poseen un condensador interno de 800 pF
- En el caso de ciertos dispositivos es necesaria una alimentación adicional externa

[1-wire: Comunicación (3)]

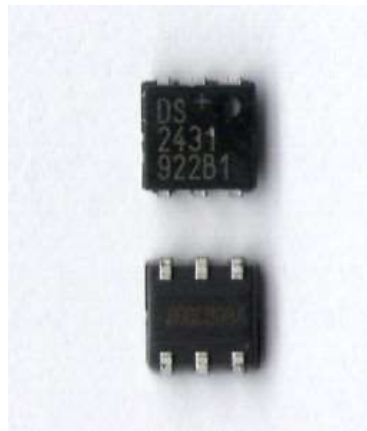
- Los primeros dispositivos 1-wire se comunicaban con el ordenador a través de la UART y tenían una velocidad de bus de hasta 16.3 Kbps (estándar)
- Para acceder al contenido de un ibutton de 64Kb en menos de 1 s se desarrolló la velocidad 10 veces superior (overdrive)

[1-wire: Formatos (1)]

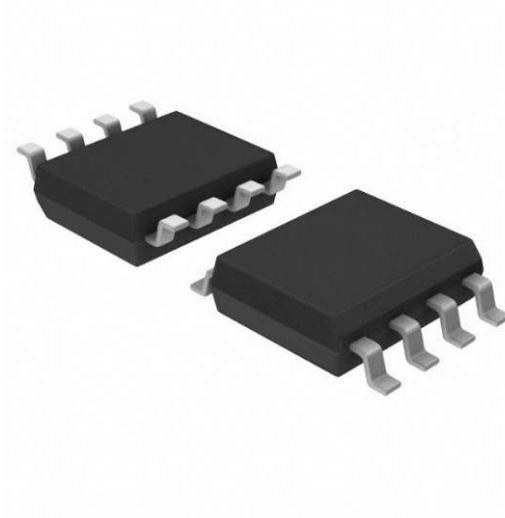
- Los dispositivos 1-wire se presentan en diferentes formatos:
 - Encapsulados TO-92, TSOC, SOIC, SOT23



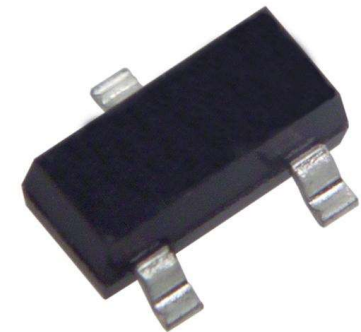
TO92



TSOC



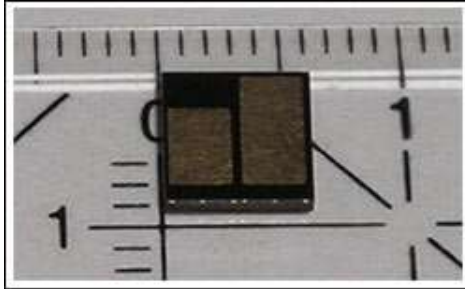
SOIC



SOT23

[1-wire: Formatos (2)]

- SFN, ibutton, flip chip, UCSP



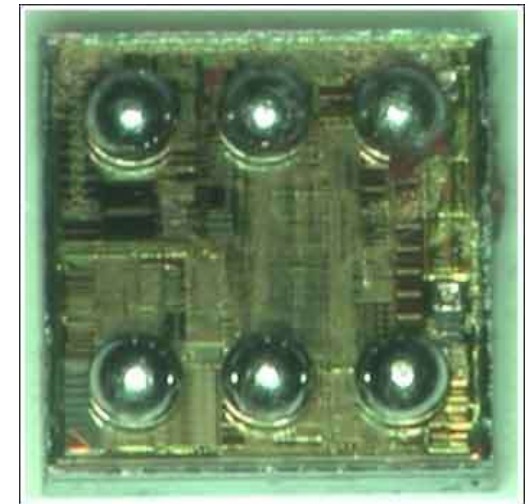
SFN



ibutton



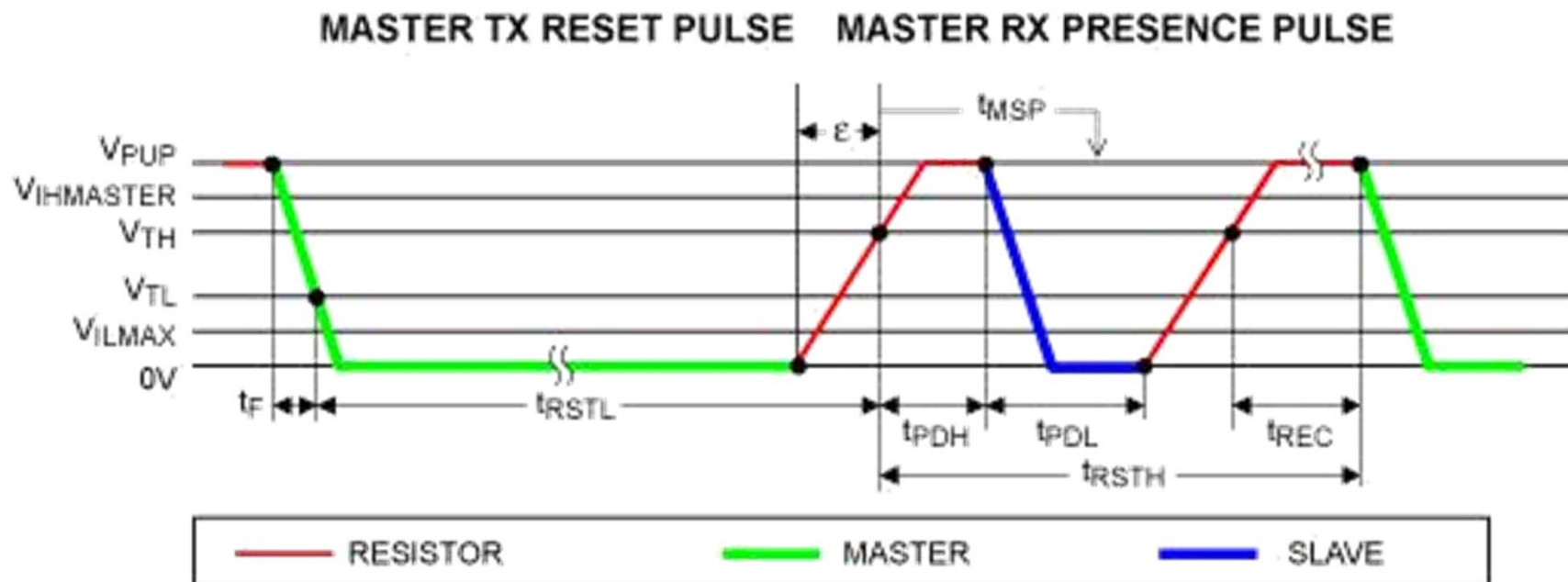
Flip chip



UCSP

[1-wire: Protocolo]

- La comunicación se inicia con un pulso de reset/presencia



- La duración del pulso influye en la velocidad de comunicación

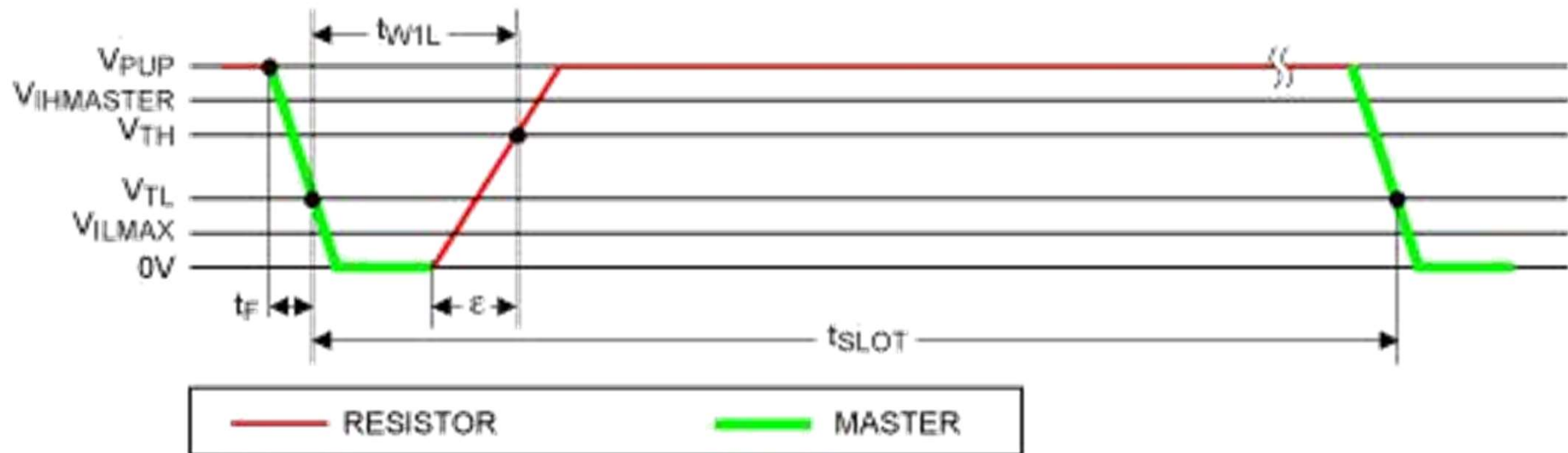
[1-wire: Protocolo]

- El pulso de reset debe durar 480 μ s o más para establecer la velocidad estándar y menos de 80 ms para permanecer en velocidad overdrive
- Una vez el esclavo está activo, el master inicia “time slots” que pueden ser de tres tipos:
 - Escritura de un ‘1’
 - Escritura de un ‘0’
 - Lectura

[1-wire: Protocol]

- Time slot de escritura de un '1'

Write-One Time Slot



- La línea debe permanecer a nivel bajo (desde que el flanco de bajada cruza el nivel V_{TL} hasta que el de subida cruza V_{TH}) como máximo un tiempo t_{W1LMAX}

[1-wire: Protocol]

- Time slot de escritura de un '0'

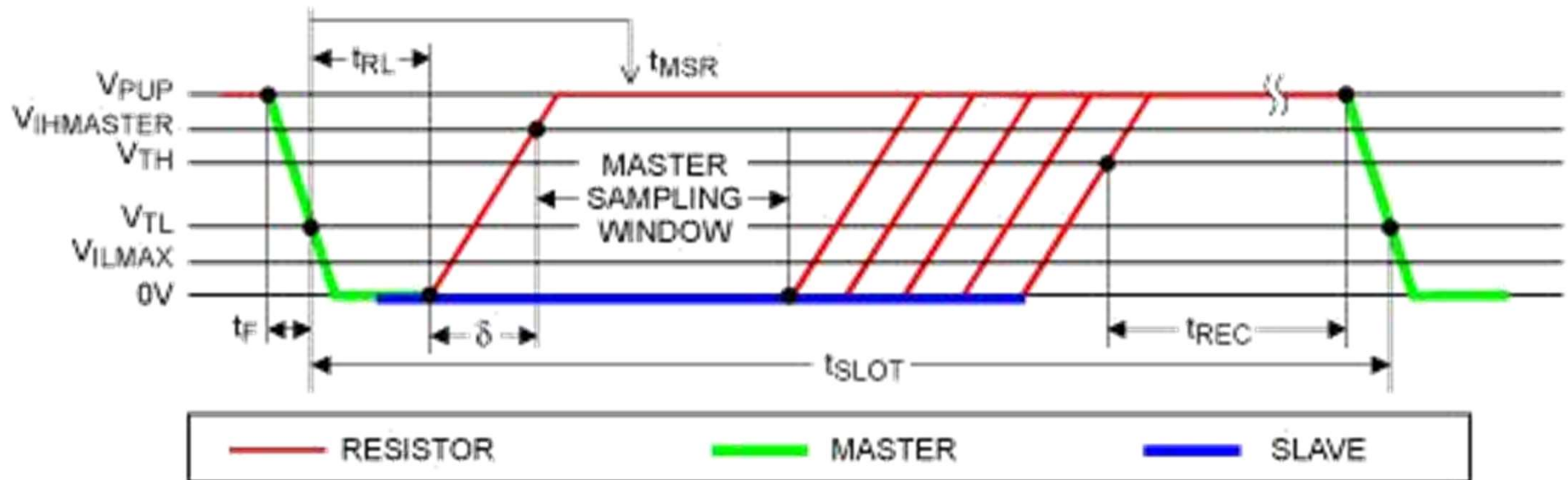


- La línea debe permanecer a nivel bajo al menos un tiempo t_{W0LMIN}

[1-wire: Protocol]

- Time slot de lectura

Read-Data Time Slot



- El esclavo mantiene la línea a nivel bajo (cero - azul) o alto (uno - rojo)

[1-wire: Protocolo]

- Cada bit transmitido requiere un *time slot*
- El tipo de *time slot* (lectura o escritura) viene determinado por el desarrollo de la comunicación. El master emite comandos y espera respuestas
- El protocolo de arbitraje se basa en la prevalencia del '0' frente al '1' en la línea de comunicación con pull-up y transmisores a colector abierto

[1-wire: Dispositivos]

- Identification only
- Id plus control
- Id plus temperature
- Id plus time
- Id plus NV SRAM
- Id plus (one time programmable) OTP EPROM
- Id plus EEPROM
- Id plus SHA-1 secure EEPROM
- Id plus logging

[Buses - comunicaciones]

- SPI
- I²C
- 1-wire
- CAN
- RS232 (UART)
- WiFi
- Bluetooth

[CAN: Computer Area Network]

- CAN (Controller Area Network) es un bus industrial caracterizado por su gran robustez y fiabilidad
- Estándar ISO 11898
- Garantiza la comunicación en los ambientes más ruidosos
- Ampliamente utilizado en automoción y en la industria

[CAN: Computer Area Network]

- Ante el incremento del número de dispositivos electrónicos en los automóviles, las necesidades de cableado y su complejidad aumentaron.
- Pronto se vio la posibilidad de conectar todos los dispositivos a un bus que debía de ser fiable, robusto, alta inmunidad al ruido, etc.
- El bus debía poder permitir altas velocidades de transmisión en entornos difíciles por la temperatura, vibraciones, interferencias, etc.
- Además del automóvil, encuentra gran utilidad como bus de campo en diversas aplicaciones industriales.

[CAN: Computer Area Network]

- Patentado por la compañía Robert Bosch en 1982.
- Diseñado como bus de campo, se aplicó en el sector del automóvil.
- El Mercedes Clase E fue el primer coche en incorporar el bus CAN en 1992.
- Fue diseñado para permitir la comunicación fiable entre centralitas electrónicas basadas en microprocesador, ECUs (“Electronic Control Unit”) y reducir cableado.
- En Europa se ha convertido en un estándar “de facto”, con carácter internacional y documentado por normas ISO (11898).
- Ventajas: reducción de costes, mejora flexibilidad

[CAN: Computer Area Network]

- Protocolo serie asíncrono de tipo CSMA/CD (“Carrier Sense Multiple Access with Collision Detection”).
- El bus es un medio compartido (multiplexado). Comunicación “Multicast”. Todos pueden transmitir y deben monitorizar.
- “CSMA”: Cada nodo monitoriza el bus. Si no detecta actividad, puede transmitir.
- “CD”: Si dos o más nodos de la red comienzan a transmitir un mensaje, detectan la colisión. Un arbitraje basado en prioridades resuelve el conflicto.

[CAN: Computer Area Network]

- Par de cables trenzados (bus diferencial). En ocasiones apantallado.
- Alta inmunidad a las interferencias electromagnéticas.
- Impedancia característica de línea del orden de 120Ω . Se colocan resistencias de este valor en los extremos para evitar reflexiones.
- Longitud máxima de 1000m (a 40Kbps).
- Velocidad máxima de 1Mbps (con una longitud de 40m). En automoción se utiliza a 125kbit/s y a 500kbit/s.

[CAN: Computer Area Network]

- El protocolo CAN está basado en mensajes (no tiene direccionamiento de nodo a nodo).
- La priorización y el direccionamiento están contenidos en los datos transmitidos
- Todos los nodos del sistema reciben toda la información transmitida
- Cada nodo decide si el mensaje o trama debe ser descartado o aceptado
- Un único mensaje puede ir destinado a un nodo en particular o a varios
- Un nodo tiene la posibilidad de pedir información de otros nodos (Remote Transmit Request)
- Pueden añadirse nodos adicionales sin necesidad de cambios

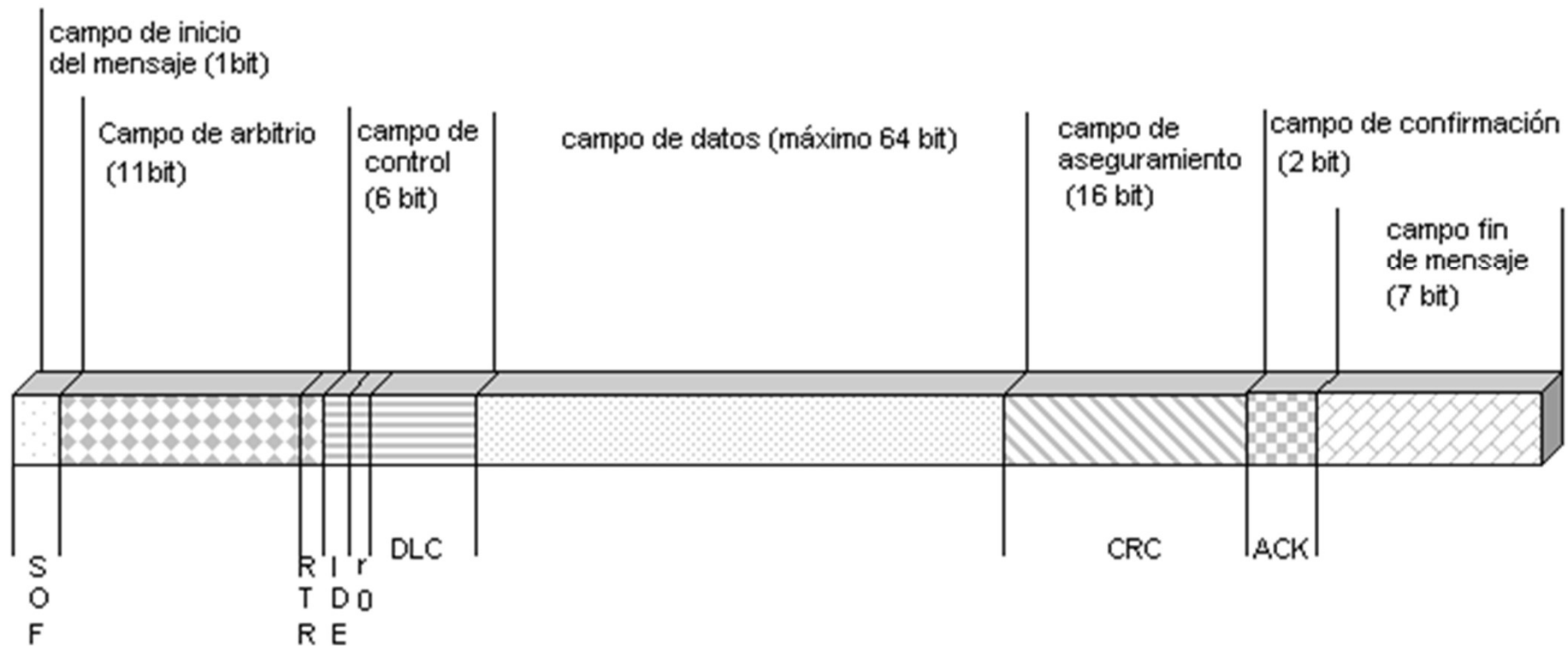
[CAN: Computer Area Network]

- El protocolo CAN define cuatro tipos de mensajes:
 - **Tramas de datos (Data Frame):** Transmite información de un nodo a cualquiera de los restantes
 - **Trama remota (Remote Frame):** Es una trama de datos con el bit RTR=1
 - **Tramas de error:** Generadas por nodos que detectan cualquiera de los errores de protocolo definidos por CAN.
 - **Tramas de overload:** Generadas por nodos que necesitan más tiempo para procesar los mensajes recibidos.

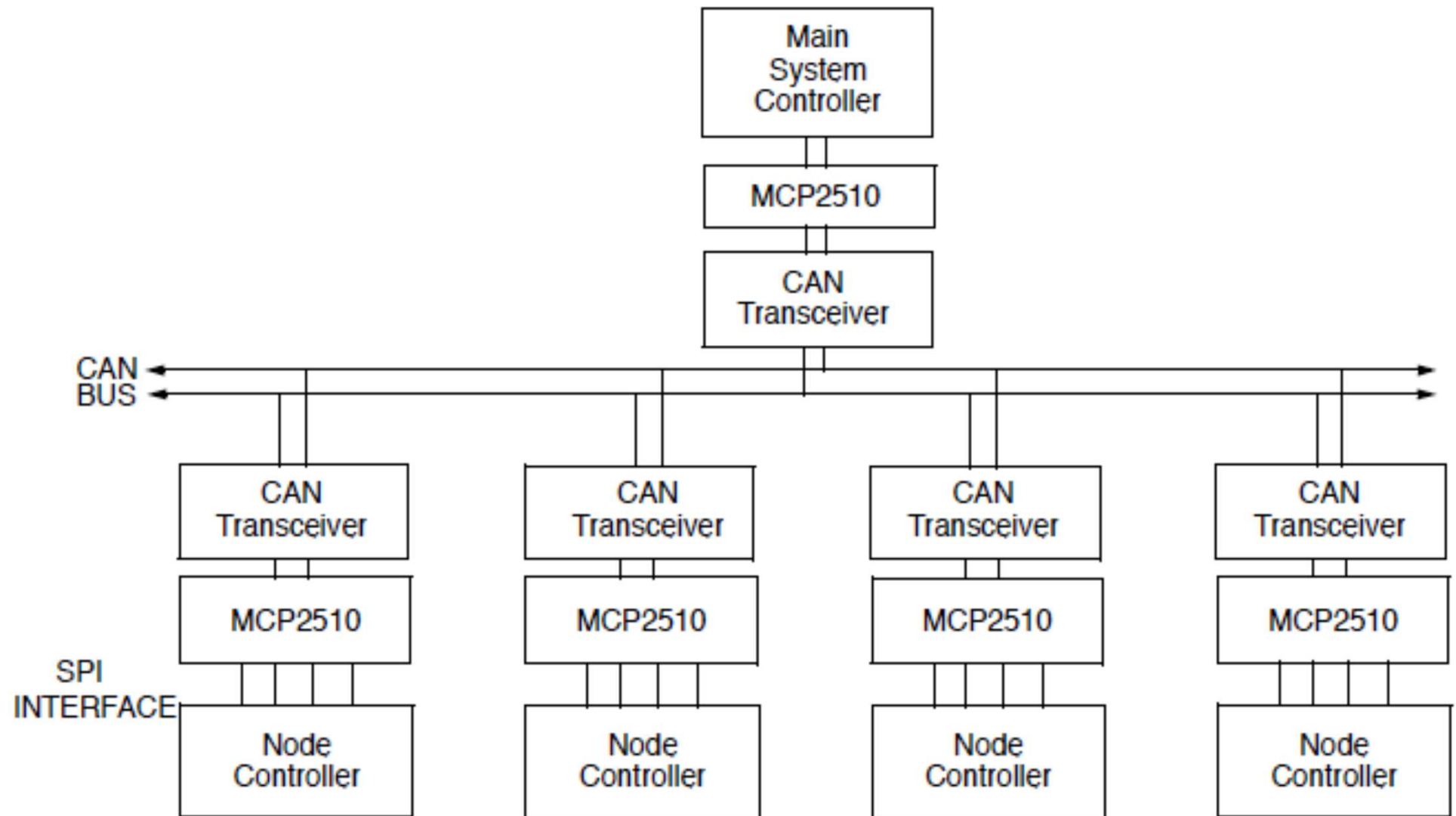
[CAN: Computer Area Network]

- La trama de datos está formada por los siguientes campos:
 - **Campo de arbitraje.** Se utiliza para priorizar los mensajes en el bus. Formado por 12 o 32 bits
 - Trama estándar: 11 bits de identificación y un bit RTR
 - Trama extendida: 29 bits de identificación, 1 bit para definir el mensaje como trama extendida, un bit SRR no usado, un bit RTR
 - **Campo de control.** Formado por 6 bits.
 - El bit IDE (o RB1 en extendida) de mayor peso, significa trama extendida
 - El bit RB0 es reservado
 - Los cuatro bits de menor peso definen la longitud de los datos (DLC).
 - **Campo de datos.** N° de bytes determinado por DLC. Los RF no tienen campo de datos.
 - **Campo de CRC.** 15 bits y un delimitador CRC. Utilizado por receptores para detección de errores de transmisión
 - **Campo de confirmación (ACK).** El nodo receptor indica recepción correcta del mensaje, poniendo un bit dominante en el flag ACK de la trama.

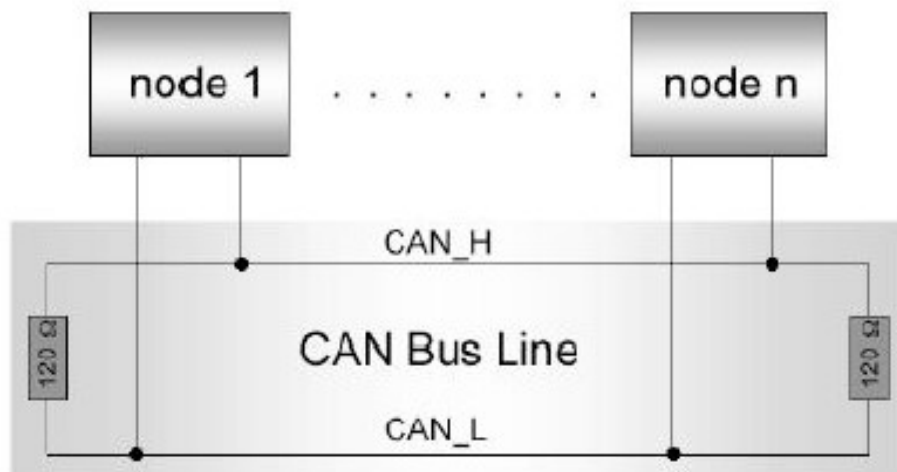
[CAN: Trama estándar]



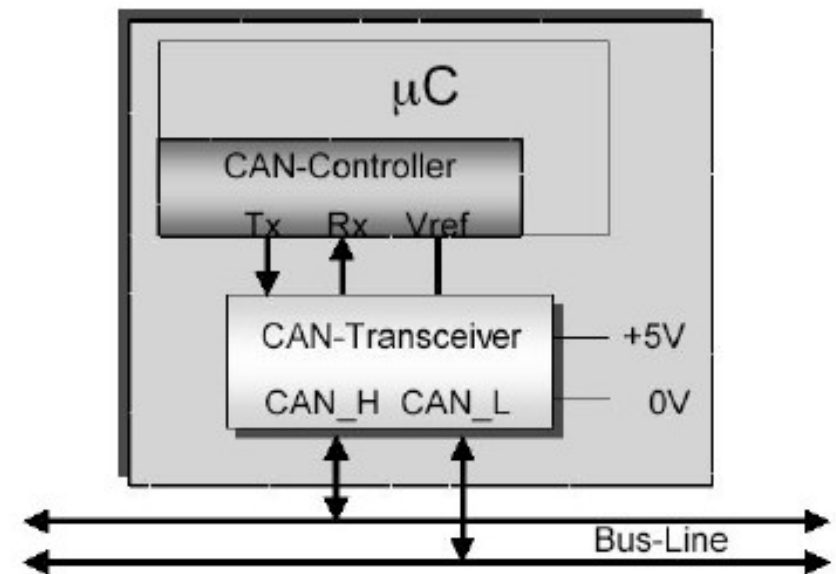
[CAN: Topología]



[CAN: Topología]

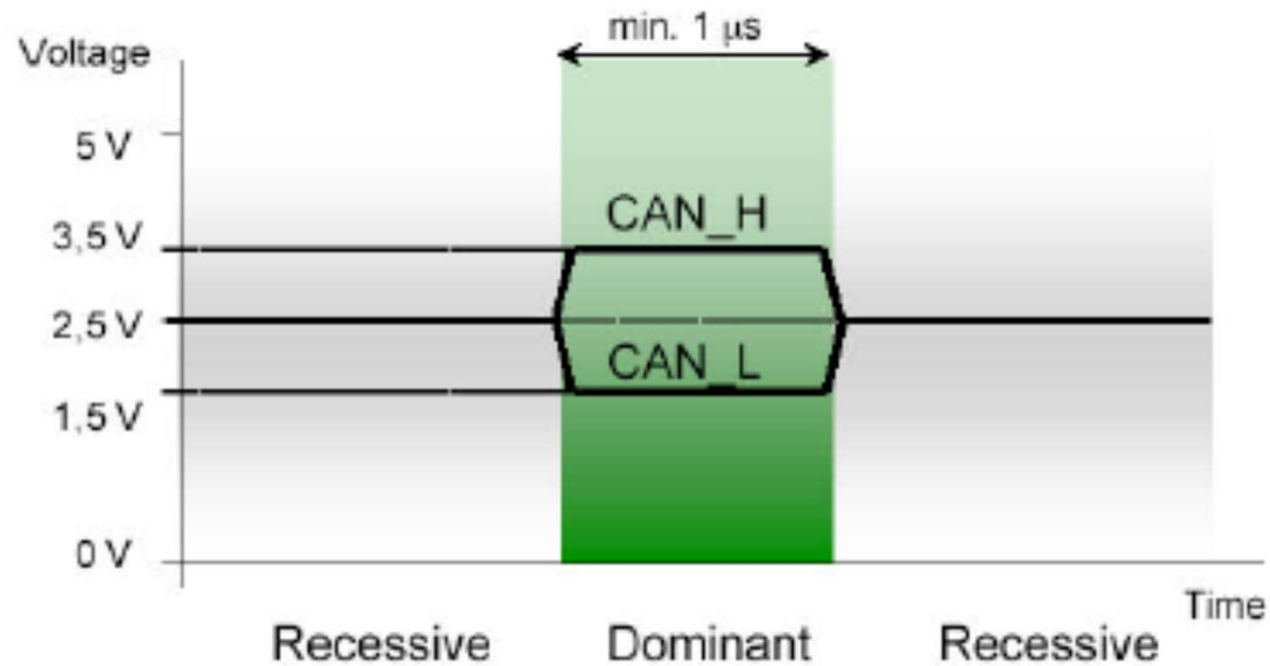


Aspecto del bus



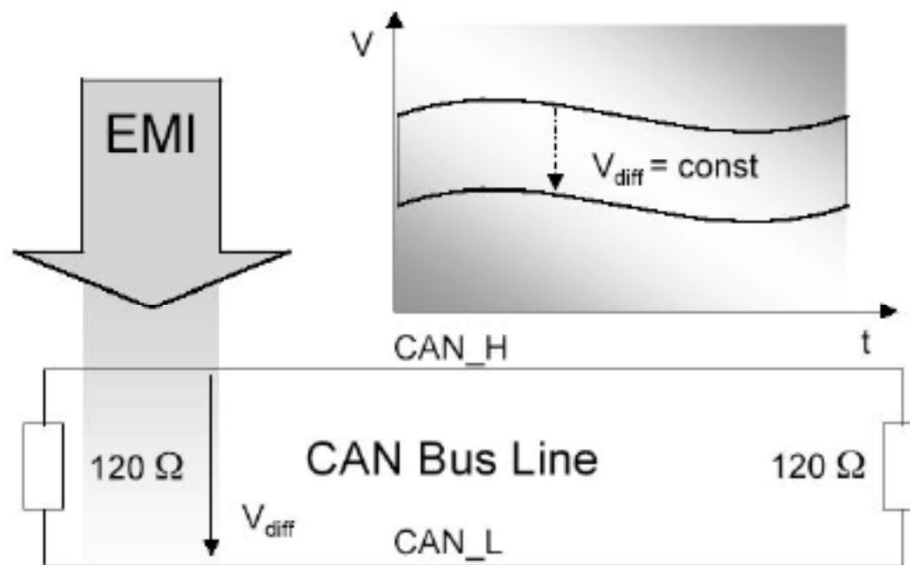
Estructura de cada nodo

[CAN: Topología]

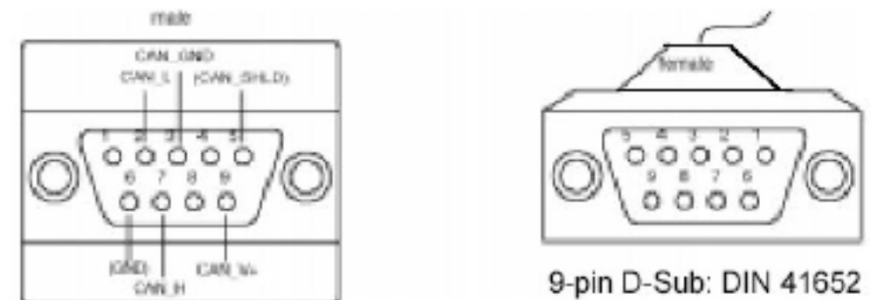


Niveles de tensión nominales

[CAN: Topología]



Protección contra interferencias



Pin	Signal	Description
1	-	Reserved
2	CAN_L	CAN_L bus line dominant low
3	CAN_GND	CAN Ground
4	-	Reserved
5	(CAN_SHLD)	Optional CAN Shield
6	GND	Optional Ground
7	CAN_H	CAN_H bus line dominant high
8	-	Reserved
9	(CAN_V+)	Optional CAN external supply

Asignación de pines

EI1062 – IR2162

Diseño de sistemas empuotrados y de tiempo real

Tema 4 – Comunicaciones
en sistemas empuotrados (I)

Grado en Ingeniería Informática

Grado en Inteligencia Robótica