

# Comunicación



# RS-232



## Introducción

A medida que la popularidad de la comunicación serial asíncrona fue ampliamente aceptada por la industria, el estándar RS-232 obtuvo una aceptación muy amplia. El uso de este estándar es visible en casi todas las industrias: aplicaciones portátiles, de escritorio, de adquisición de datos y de medición que utilizan un puerto serie para comunicación. Aunque el estándar especifica una velocidad de datos máxima para RS-232 de 20 kbps, la necesidad de aplicaciones de mayor velocidad es muy solicitada. Hoy en día, cada vez más aplicaciones requieren al menos 120 kbps para admitir Laplink, un popular software de comunicación utilizado por computadoras portátiles/de escritorio para transferencia rápida de archivos entre dos computadoras. Los controladores y receptores de tipo RS-232 también deben admitir esta mayor tasa de datos para ser compatible con Laplink.

## Historia

El estándar EIA/TIA-232-E se introdujo en 1962 y desde entonces ha sido actualizado cuatro veces para cumplir con las necesidades cambiantes de las aplicaciones de comunicación en serie. El nombre oficial del estándar EIA/TIA-232-E es "Interfaz entre equipo terminal de datos (DTE) y equipo de terminación de datos (CTE), empleando intercambio de datos binarios en serie". Hay una definición más simple: comunicación de datos en serie entre un sistema anfitrión (DTE) y un sistema periférico (CTE). Ver figura 1.

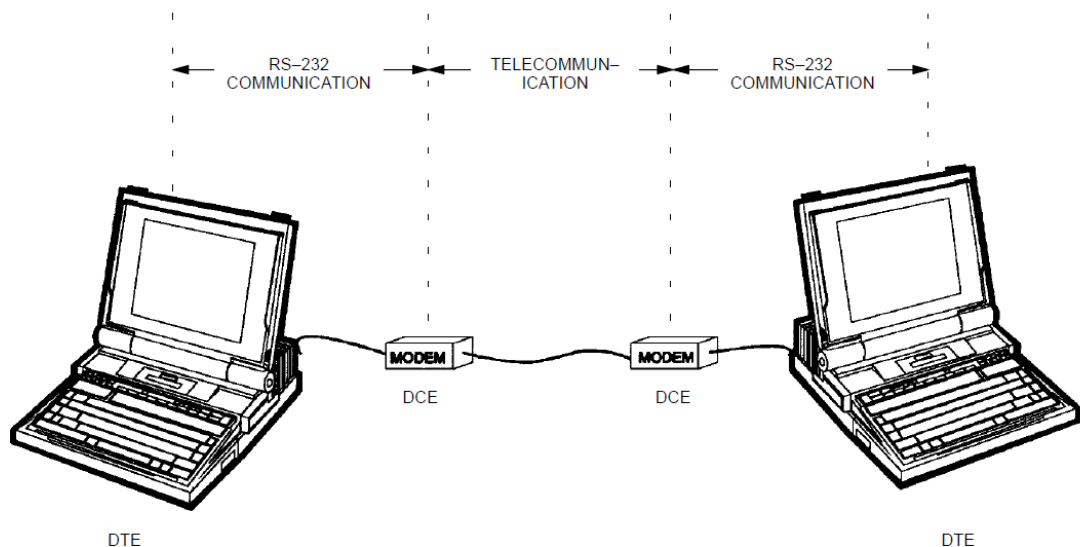


Figura 1: Ejemplo de enlace RS-232



## Historial de aplicaciones

Históricamente, la comunicación en serie RS-232 se usaba para interconectar computadoras con periféricos como módems, impresoras, teclados, joysticks y un ratón. La mayoría de estas aplicaciones ahora se han convertido a otros protocolos de comunicación como el bus serie universal (USB).

Hoy en día, la comunicación en serie RS-232 se utiliza en aplicaciones como GPS, medidores de glucosa, código de barras escáneres, telemática automotriz, decodificadores, juegos y muchos otros que requieren bajo costo y baja velocidad.

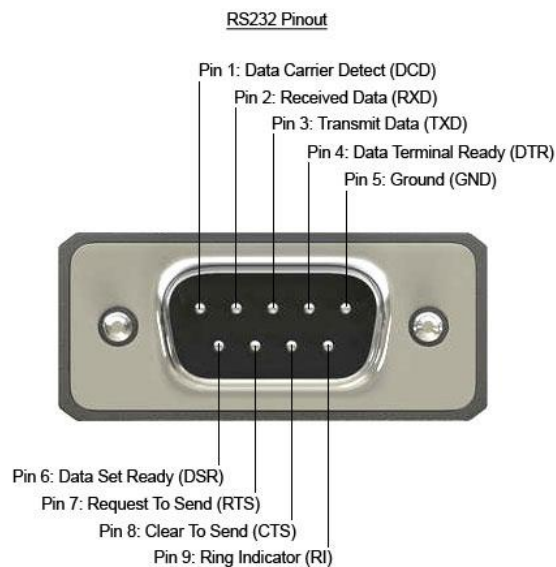


Figura 2: Pinout de un conector DB9

## Características eléctricas

La sección de características eléctricas del RS-232 estándar incluye especificaciones sobre niveles de tensión, tasa de cambio de niveles de señal e impedancia de línea. El estándar RS-232 original se definió en 1962. Como esto fue antes de los días de la lógica TTL, no debería ser sorprendente que el estándar no use 5 volts. La salida se define como de +5 a +15 volts para un nivel alto, y para un nivel bajo para la salida del controlador se define entre -5 y -15 volts.

Los niveles lógicos del receptor se definieron para proporcionar un margen de ruido de 2 volts. Como tal, un alto nivel para el receptor se define como +3 a +15 voltios y un nivel bajo es -3 a -15 volt. La figura siguiente ilustra los niveles lógicos definidos por el estándar RS-232. Es necesario tener en cuenta que, para comunicación RS-232, un nivel bajo (-3 a -15 volt) se define como un 1 lógico y se hace referencia históricamente a



como "marca". Asimismo, un nivel alto (+3 a +15 voltios) es se define como un 0 lógico y se denomina "espaciado".

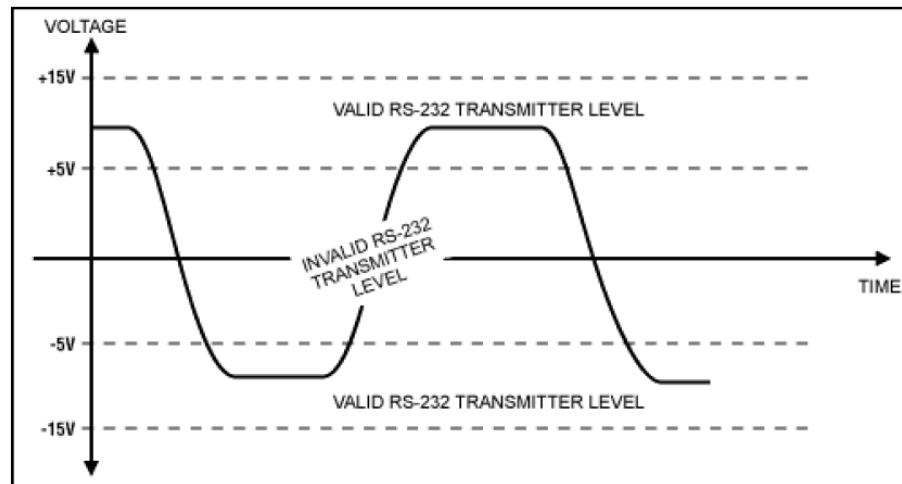


Figura 3: Niveles lógicos RS-232

## Driver RS-232

El controlador RS-232 normalmente está conectado a un cable y un receptor en el otro extremo. Dado que el tiempo de transición del controlador está configurado para ser sustancialmente más largo que el retardo del cable, la carga del cable representa una carga capacitiva concentrada y una resistencia en serie.

La resistencia en serie en cables cortos (< 60 metros) puede despreciarse ya que es muy pequeña en comparación con la resistencia de entrada del receptor. esto significa que el cable se puede modelar como una carga capacitiva concentrada igual a la capacitancia por unidad de longitud multiplicada por la longitud del cable. 1000 pF se usa comúnmente para representar un cable de 6 metros y 2500 pF se usa como la carga de cable máxima especificada. La resistencia de entrada del receptor se especifica entre 3 kΩ y 7 kΩ, 5 kΩ se usa como valor típico y 3 kΩ es el peor de los casos desde el punto de vista de la potencia. Esta carga equivalente se ilustra en la figura 4.

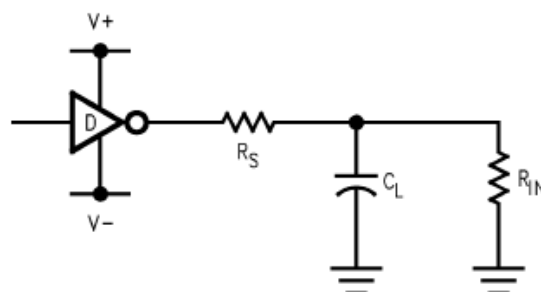


Figura 4: Carga vista por el driver



## Hardware handshake flujo

Un análisis paso a paso del handshake ilustra cómo se utiliza cada circuito para establecer la comunicación entre un sitio local y uno remoto. Para simplificar el tema, se ha supuesto que la transmisión es solo de Local a Remoto. En la figura 5 se muestra el protocolo de enlace de hardware.

1. El DTE local enciende DTR y el módem local marca el número de teléfono del módem remoto.
2. Si DTR en la ubicación remota está ACTIVADO, el RI del módem remoto se ACTIVA/DESACTIVA en secuencia con el teléfono, sonando, indicando que está entrando una llamada.
3. El módem remoto devuelve un tono de respuesta al módem local. Al detectar este tono, el módem local y el módem remoto establece la conexión en línea. En este punto, ambos módems encienden sus pines DSR para indicar que se ha establecido una conexión.
4. El DTE local enciende RTS indicando que está listo para enviar datos. Esta señal se transmite a el circuito DCD de los módems remotos.
5. El módem local verifica que el DCD local esté APAGADO, lo que indica que el módem remoto no tiene el control de la línea de transporte.
6. El módem local luego enciende CTS en el DTE local para informar que puede comenzar a enviar datos. Localmente, el circuito DCD permanece APAGADO. En el módem remoto, DCD permanece encendido. RTS está en ON por el local DTE durante toda la duración de la conexión.
7. El DTE local envía datos a través de TXD al módem para su transmisión.
8. El módem remoto recibe los datos y los envía a su terminal a través del circuito RXD.
9. Cuando finaliza la transmisión de datos, el DTE local descarta el RTS, que descarta el DCD en el módem remoto y CTS en módem local. La transmisión de datos puede interrumpirse colgando la línea telefónica, por el DTE desconectando su circuito DTR, desconectando el cable del módem del DTE.
10. Ahora, cualquiera de los DTE está listo para comenzar de nuevo y obtener el control de la línea de telecomunicaciones.



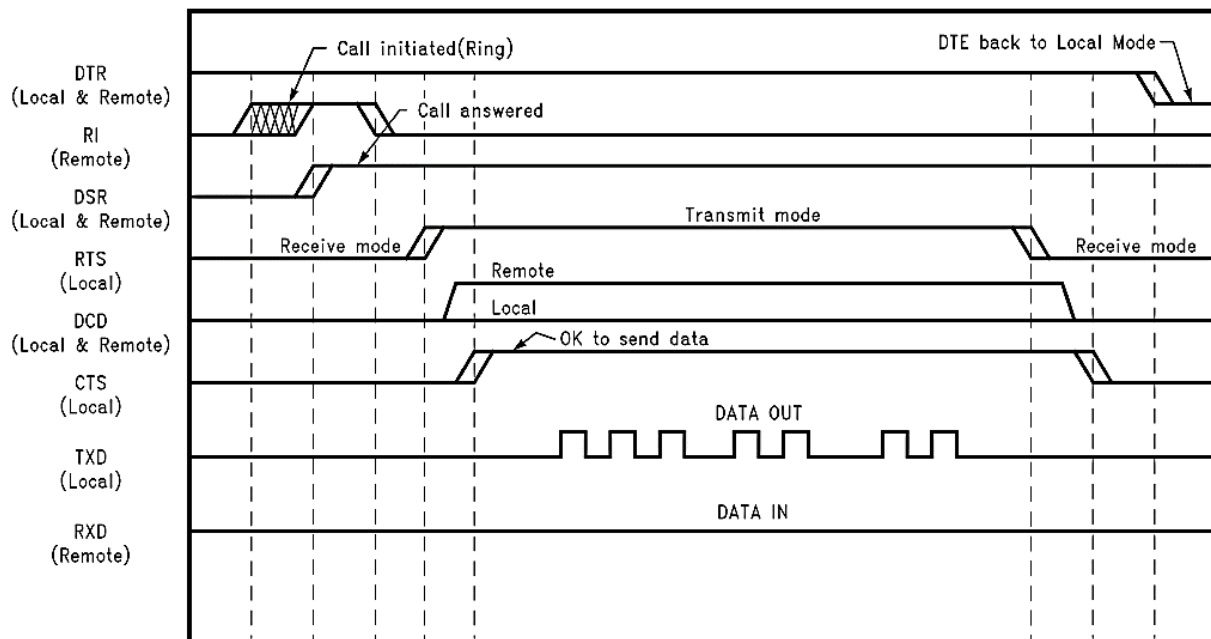


Figura 5: Protocolo de comunicación

## Comunicación SCPI (comandos estándar para programación de instrumentos)

Los comandos estándar para instrumentación programable (SCPI) definen un estándar destinado al control de instrumentación. El SCPI describe un lenguaje que es útil para controlar los instrumentos de prueba. SCPI ofrece una sintaxis estándar, formato de intercambio de datos y estructura de comandos.

El objetivo clave de SCPI es minimizar el tiempo de desarrollo de un programa de equipos de prueba automática (ATE). El objetivo se logra al proporcionar un entorno de programación confiable para el uso de datos y el control del instrumento. Este entorno de programación confiable se obtiene utilizando formatos de datos definidos, mensajes de programa y respuestas de instrumentos en todos los equipos SCPI, independientemente del diseñador.

