Data Communication and Network

Asynchronous serial communication

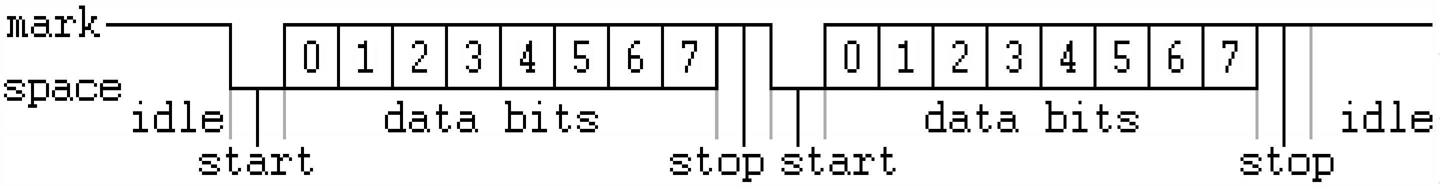
Asynchronous serial communication is a form of serial communication in which the communicating endpoints' interfaces are not continuously synchronized by a common clock signal. Instead of a common synchronization signal, the data stream contains synchronization information in form of start and stop signals, before and after each unit of transmission, respectively.

Asynchronous serial communication

The start signal prepares the receiver for arrival of data and the stop signal resets its state to enable triggering of a new sequence.

A common kind of start-stop transmission is ASCII over RS-232, for example for use in teletypewriter operation.

Asynchronous serial communication



In this diagram, two bytes are sent, each consisting of a start bit, followed by eight data bits (bits 0-7), and one stop bit, for a 10-bit character frame.

Asynchronous serial communication

The last data bit is sometimes used as a parity bit. The number of data and formatting bits, the order of data bits, the presence or absence of a parity bit, the form of parity (even or odd) and the transmission speed must be pre-agreed by the communicating parties. The "stop bit" is actually a "stop period"; the stop period of the transmitter may be arbitrarily long.

Asynchronous serial communication

It cannot be shorter than a specified amount, usually 1 to 2 bit times. The receiver requires a shorter stop period than the transmitter. At the end of each character, the receiver stops briefly to wait for the next start bit. It is this difference which keeps the transmitter and receiver synchronized.

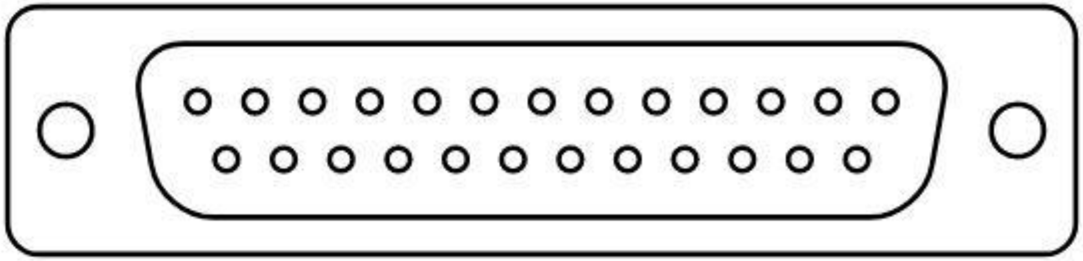
RS232

In telecommunications, RS-232, Recommended Standard 232 is a standard originally introduced in 1960 for serial communication transmission of data. It formally defines signals connecting between a DTE (data terminal equipment) such as a computer terminal, and a DCE (data circuit-terminating equipment or data communication equipment), such as a modem.

RS232

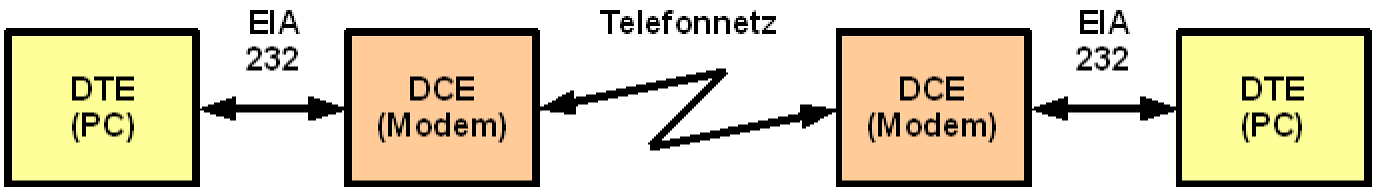
The standard defines the electrical characteristics and timing of signals, the meaning of signals, and the physical size and pinout of connectors. The current version of the standard is TIA-232-F Interface Between Data Terminal Equipment and Data Circuit-Terminating Equipment Employing Serial Binary Data Interchange, issued in 1997. The RS-232 standard had been commonly used in computer serial ports and is still widely used in industrial communication devices.

RS232



A DB-25 connector as described in the RS-232 standard

RS232



Data circuit-terminating equipment (DCE) and data terminal equipment (DTE) network. Telefonnetz refers to a telephone network; EIA-232 is an old name for RS-232, the serial communication standard.

RS232

- A serial port complying with the RS-232 standard was once a standard feature of many types of computers. Personal computers used them for connections not only to modems, but also to printers, computer mice, data storage, uninterruptible power supplies, and other peripheral devices.

RS232

RS-232, when compared to later interfaces such as RS-422, RS-485 and Ethernet, has lower transmission speed, short maximum cable length, large voltage swing, large standard connectors, no multipoint capability and limited multidrop capability. In modern personal computers, USB has displaced RS-232 from most of its peripheral interface roles.

USART(IC 8251A) word format

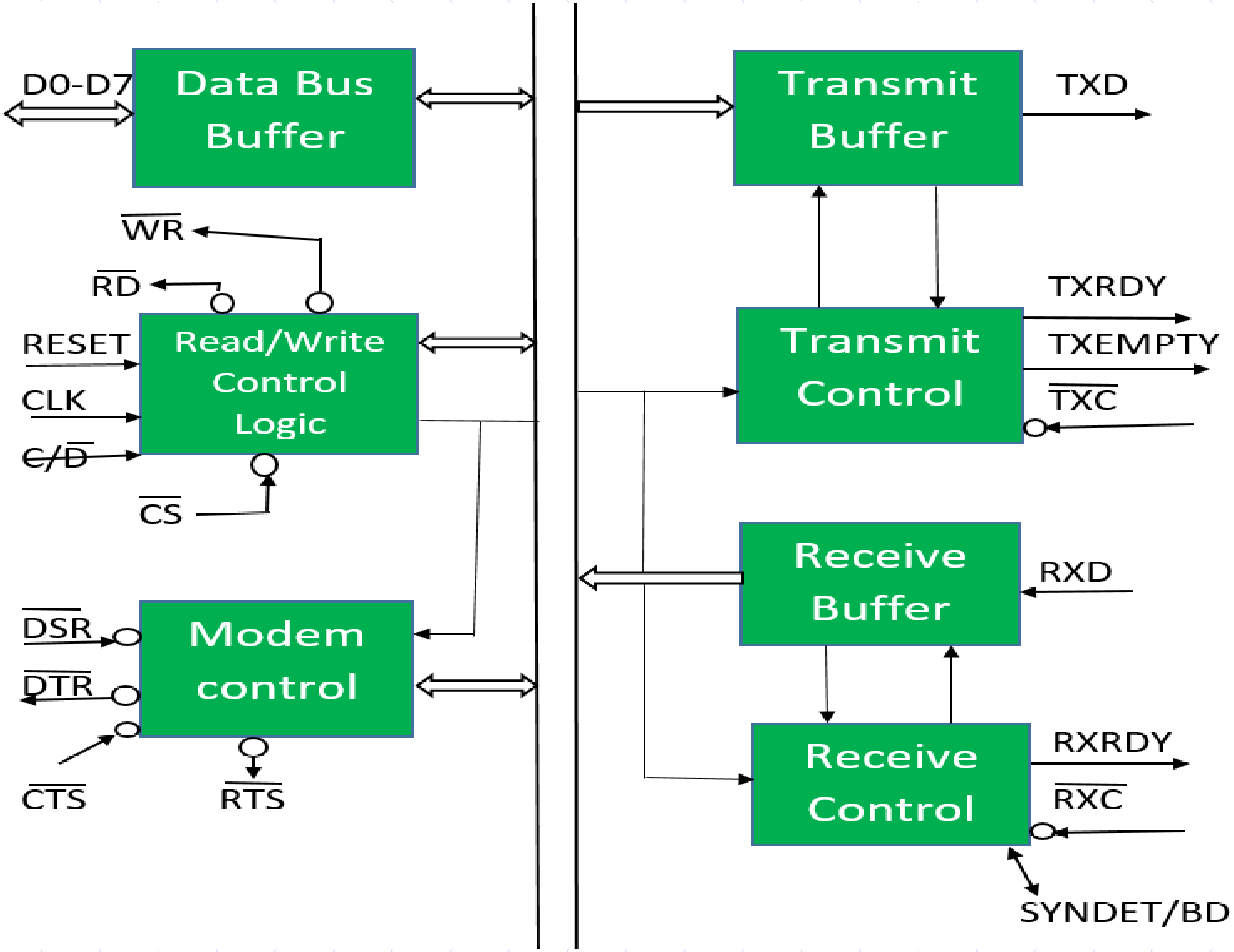
8251 universal synchronous asynchronous receiver transmitter (USART) acts as a mediator between microprocessor and peripheral to transmit serial data into parallel form and vice versa.

- It takes data serially from peripheral (outside devices) and converts into parallel data.

USART(IC 8251A) word format

- After converting the data into parallel form, it transmits it to the CPU.
- Similarly, it receives parallel data from microprocessor and converts it into serial form.
- After converting data into serial form, it transmits it to outside device (peripheral).

USART(IC 8251A) word format



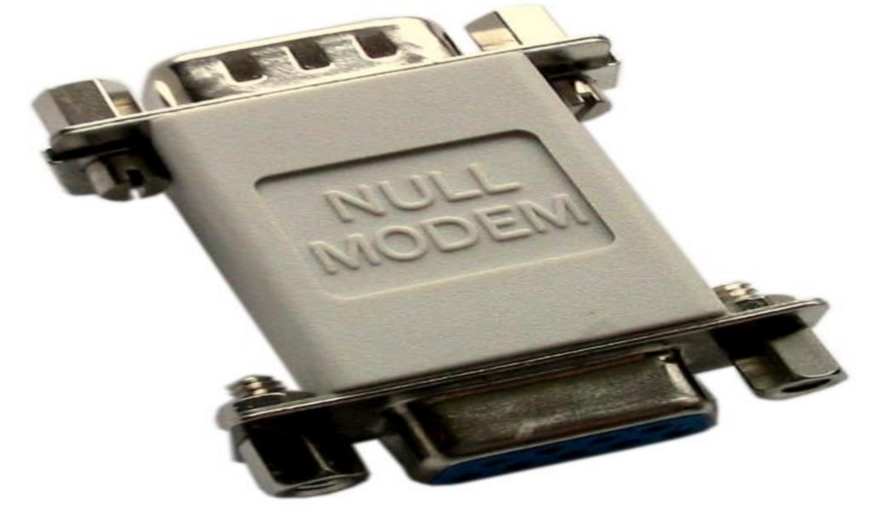
Null Modem Configuration

Null modem is a communication method to directly connect two DTEs (computer, terminal, printer, etc.) using an RS-232 serial cable. The name stems from the historical use of RS-232 cables to connect two teleprinter devices or two modems in order to communicate with one another; null modem communication refers to using a crossed-over RS-232 cable to connect the teleprinters directly to one another without the modems. It is also used to serially connect a computer to a printer, since both are DTE, and is known as a Printer Cable.

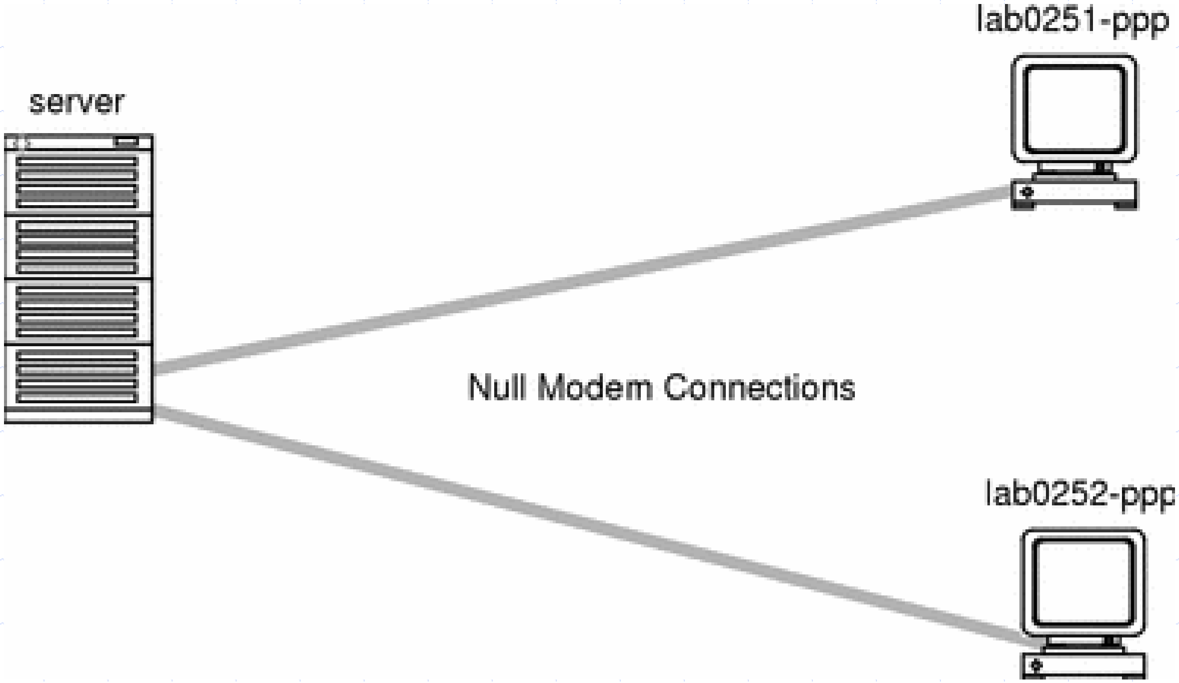
Null Modem Configuration

The RS-232 standard is asymmetric as to the definitions of the two ends of the communications link, assuming that one end is a DTE and the other is a DCE, e.g. a modem. With a null modem connection the transmit and receive lines are crosslinked. Depending on the purpose, sometimes also one or more handshake lines are crosslinked. Several wiring layouts are in use because the null modem connection is not covered by the RS-232 standard.

Null Modem Configuration



Null Modem Configuration



In this example, the server has two asynchronous null modem connections to the clients lab0251-ppp and lab0252-ppp. The modem keyword in the link.conf file is always set to none, and a dummy telephone number is assigned to the server.

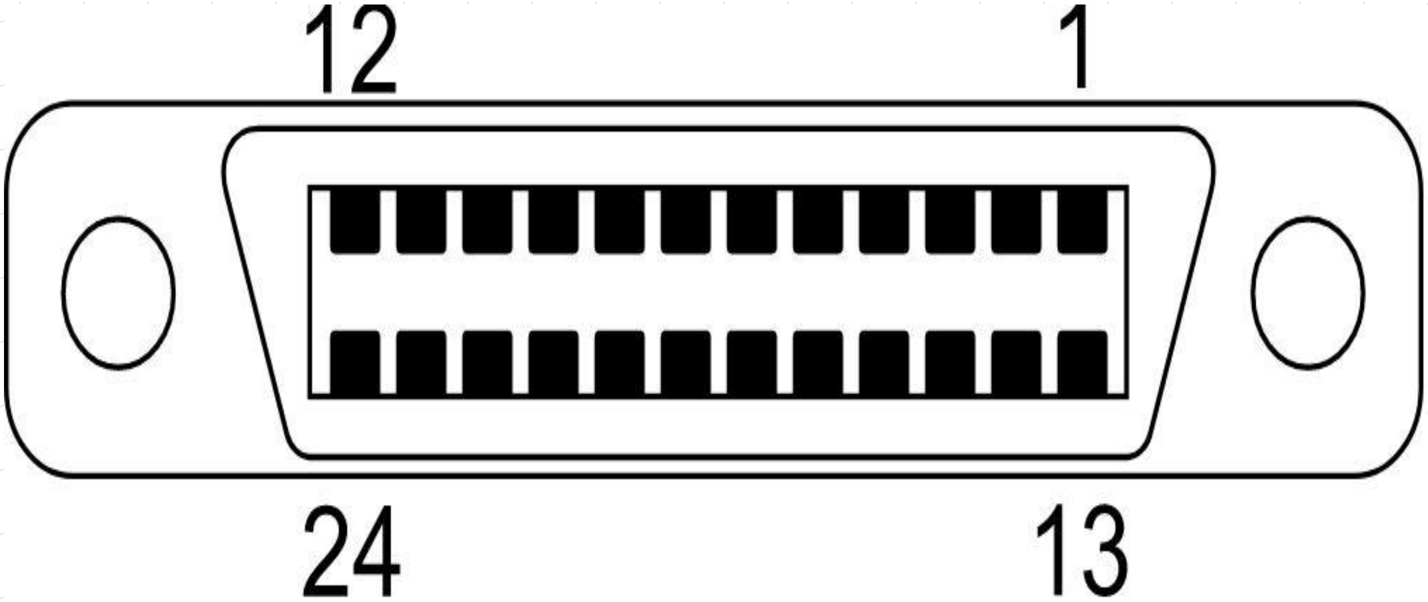
GPIB Bus/IEEE 488

- IEEE 488 is a short-range digital communications 8-bit parallel multi-master interface bus specification developed by Hewlett-Packard as HP-IB (Hewlett-Packard Interface Bus). It subsequently became the subject of several standards, and is generically known as GPIB (General Purpose Interface Bus).
- Although the bus was created in the late 1960s to connect together automated test equipment, it also had some success during the 1970s and 1980s as a peripheral bus for early microcomputers, notably the Commodore PET.

GPIB Bus/IEEE 488

Newer standards have largely replaced IEEE 488 for computer use, but it is still used by some test equipment.

GPIB Bus/IEEE 488



GPIB Bus/IEEE 488

Pin 1	DIO1	Data input/output bit.
Pin 2	DIO2	Data input/output bit.
Pin 3	DIO3	Data input/output bit.
Pin 4	DIO4	Data input/output bit.
Pin 5	EOI	End-or-identify.
Pin 6	DAV	Data valid.
Pin 7	NRFD	Not ready for data.
Pin 8	NDAC	Not data accepted.
Pin 9	IFC	Interface clear.
Pin 10	SRQ	Service request.
Pin 11	ATN	Attention.
Pin 12	SHIELD	

GPIB Bus/IEEE 488

Pin 13	DIO5	Data input/output bit.
Pin 14	DIO6	Data input/output bit.
Pin 15	DIO7	Data input/output bit.
Pin 16	DIO8	Data input/output bit.
Pin 17	REN	Remote enable.
Pin 18	GND	(wire twisted with DAV)
Pin 19	GND	(wire twisted with NRFD)
Pin 20	GND	(wire twisted with NDAC)
Pin 21	GND	(wire twisted with IFC)
Pin 22	GND	(wire twisted with SRQ)
Pin 23	GND	(wire twisted with ATN)
Pin 24	Logic ground	

GPIB Bus/IEEE 488

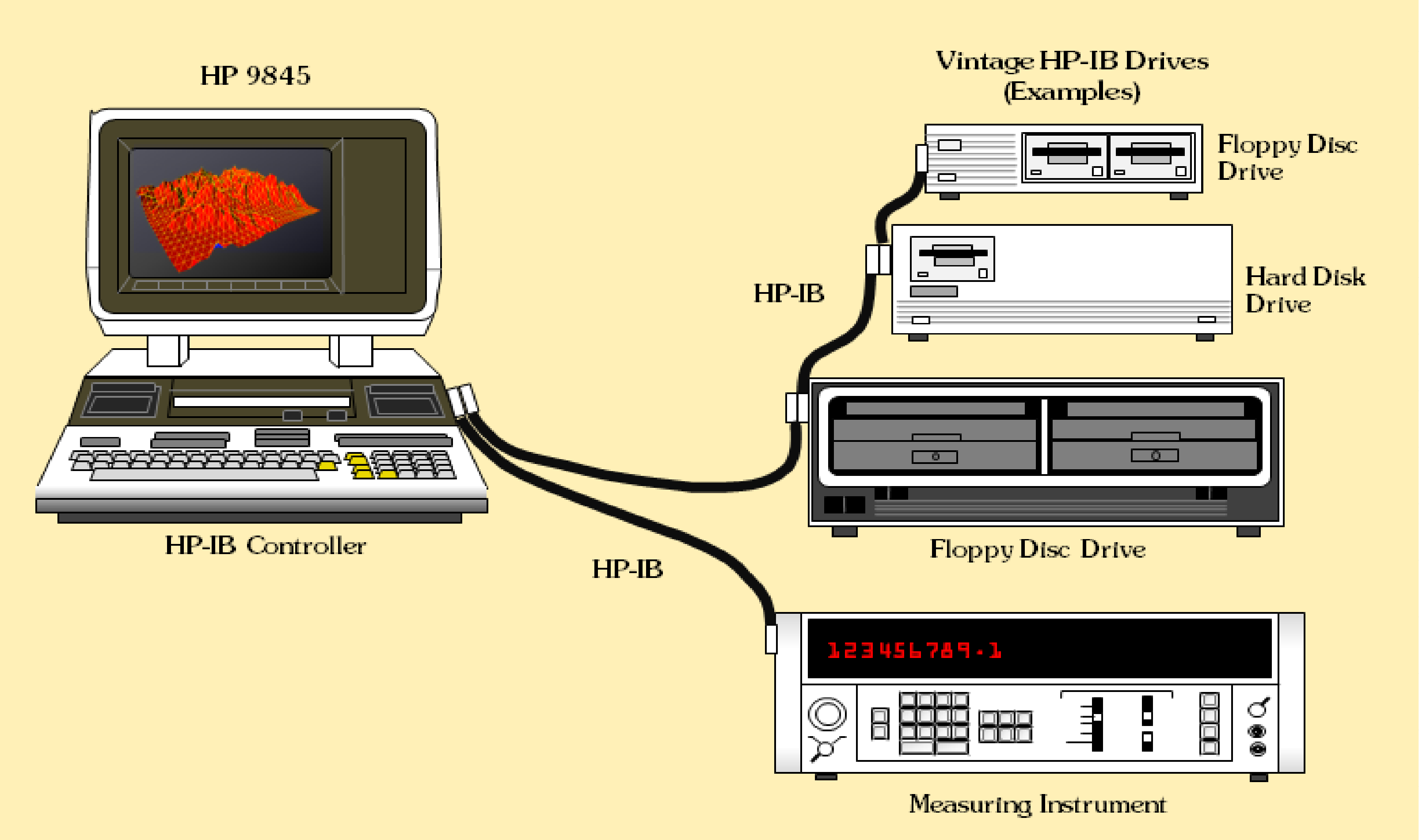
IEEE 488 is an 8-bit, electrically parallel bus which employs sixteen signal lines — eight used for bi-directional data transfer, three for handshake, and five for bus management — plus eight ground return lines.

HPIB

The Hewlett-Packard Interface Bus (HP-IB)

The HP 9845 is closely related to the Hewlett-Packard Interface Bus or HP-IB, also known as IEEE 488 or General Purpose Interface Bus (GPIB). Most of the HP 9845 periperals such as mass storage, graphics devices or printers can be connected via the HP 98034 HP-IB interface. Even today the HP-IB is the entry to the HP 9845, since it is the easiest and most versatile connection between the HP 9845 and a modern PC. And the required 98034A/B HP-IB interfaces are still relatively easy to acquire.

HPIB


HP 9845
Vintage HP-IB Drives
(Examples)
Floppy Disc
Drive
Hard Disk
Drive
HP-IB
HP-IB Controller
Floppy Disc Drive
HP-IB
123456789.1
Measuring Instrument

THANK YOU!!!