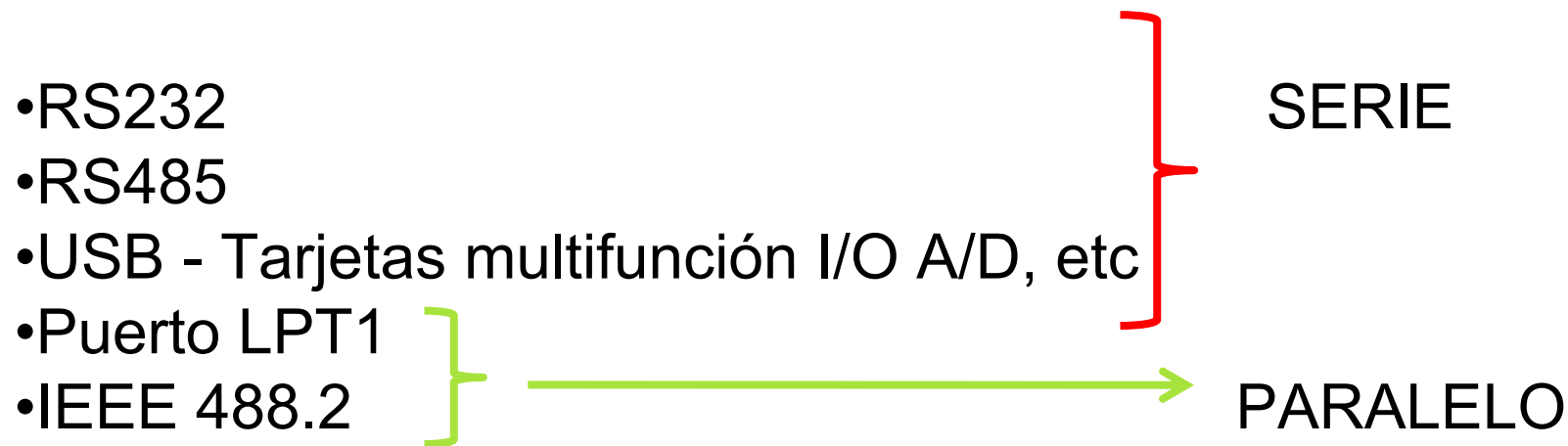


Arquitectura de PC (II)

Hardware para adquirir datos

Tarjetas y puertos

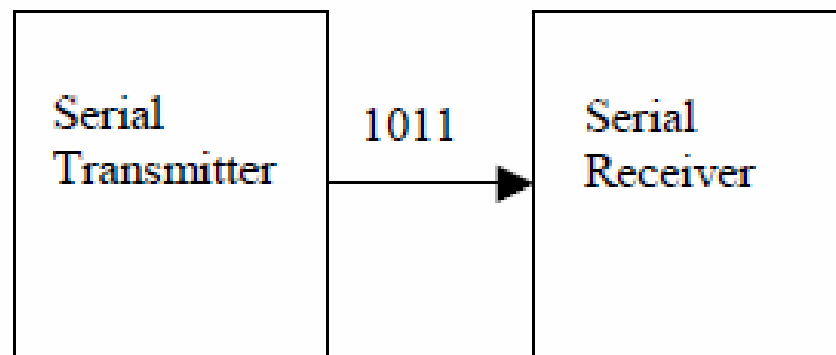


Objetivos: Entender su funcionamiento básico para automatizar experimentos y para digitalizar datos o mediciones.

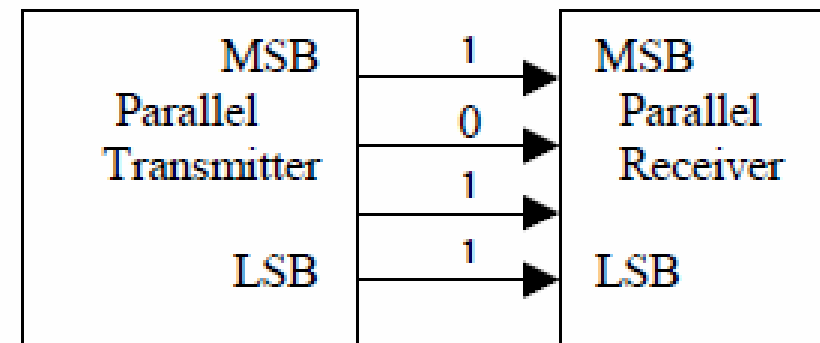


Concepto de comunicación:

El concepto de la comunicación serie es simple ,los ports serie de los dispositivos (PC , microprocesadores ,tarjetas de adquisición) envían y reciben datos (caracteres, bytes, palabra-dato) de información de a un bit a la vez. , en contraste con la comunicación paralelo donde son enviados simultáneamente .



Transmisión Serie



Transmisión paralela



Algunas ventajas y desventajas

- La comunicación en serie es más versátil que la paralela
Transmisión de datos a un periférico a distancia
P ej. IEEE488 (paralelo) no permite más de 20 m de cable
y cada equipo a menos de 2m.
- La com. en serie permite 1200 mts. Cables más baratos
un solo conductor a diferencia de los 8 en el paralelo.
(uso en telefonía).
- La serial debe rastrear cada bit del carácter transmitido
(posición y duración de la cadena-*stream*)
Requiere más tiempo es más lenta.

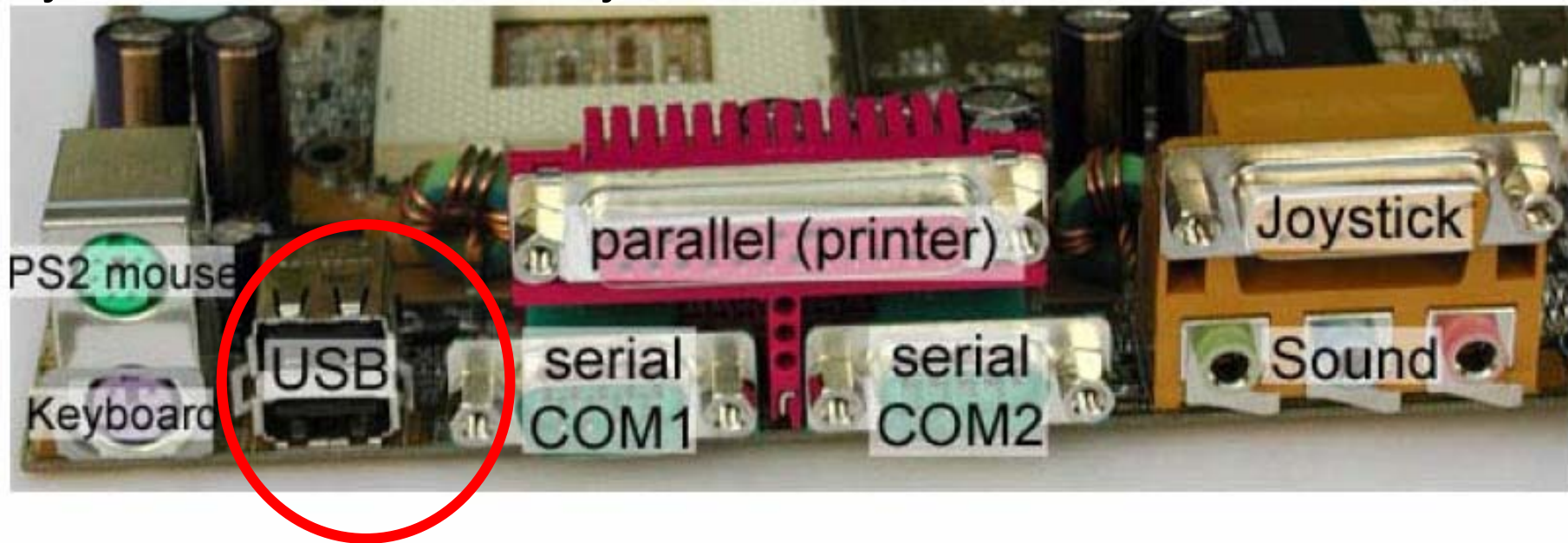


Conectores para salir de la PC

- ⇒ Sirven para conectar al PC los dispositivos externos de E/S.
- ⇒ Puerto serie(RS232C) COM 1,COM 2, etc.
- ⇒ Puerto paralelo(CENTRONICS): LPT1 o PRN.
- ⇒ Puerto USB



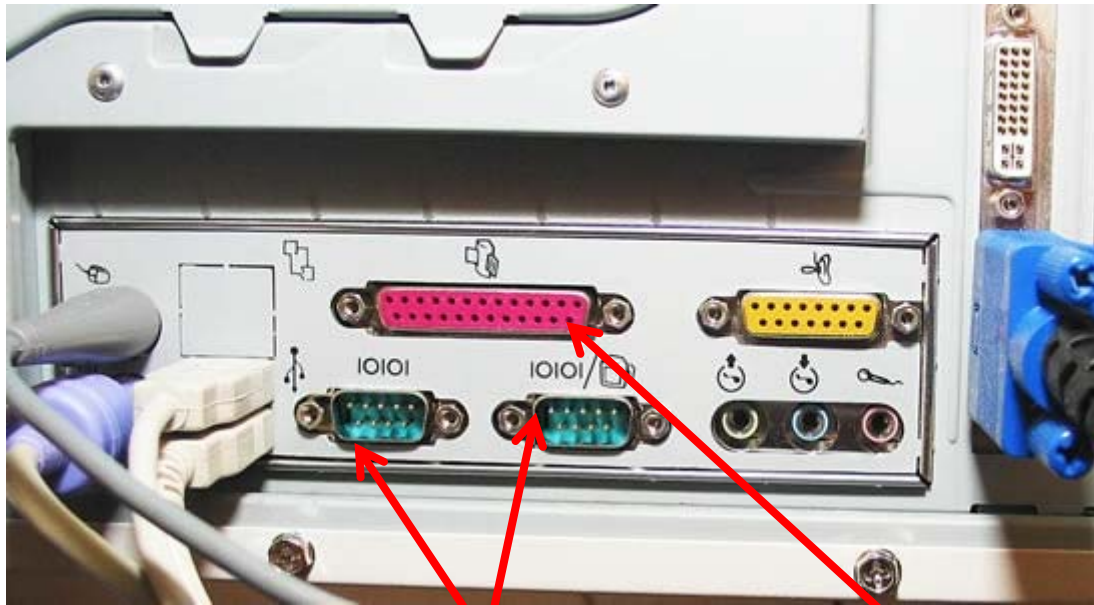
Las PC que generalmente compramos tienen puertos.
Hoy en día el más usado y más común es USB



USB (Universal Serial Bus) hasta 60 MB/sec, pero usualmente a 35 MB/sec .
Capaz de aceptar varios instrumentos (es un bus). La norma define todo, incluso el conector. cables de hasta 5 metros



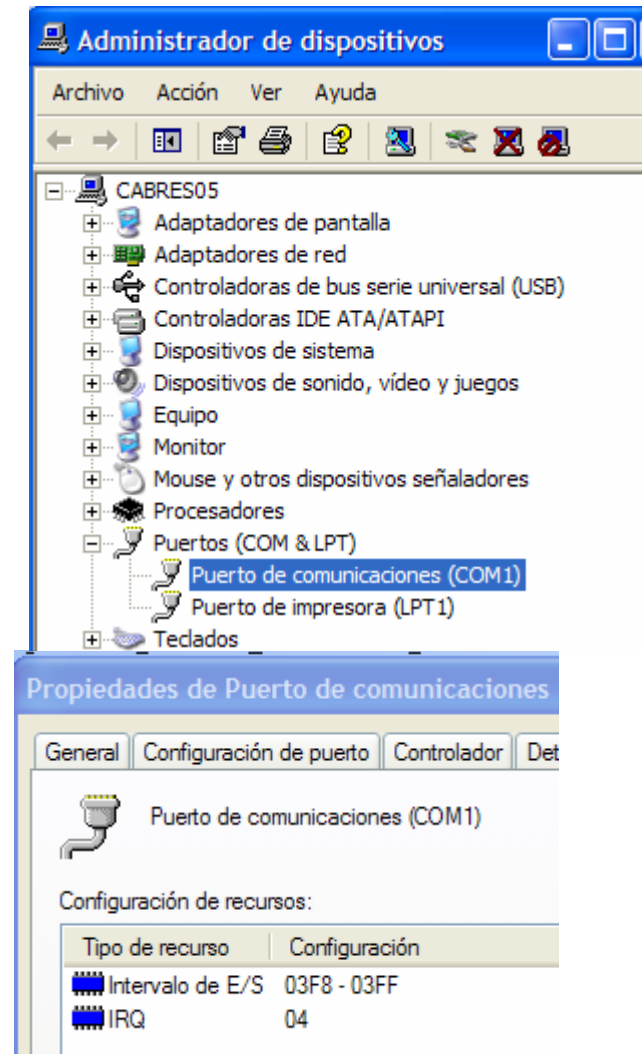
El puerto Serie y el puerto paralelo



En general los COM1 y COM2 son Machos DB9 en el gabinete. Antes también había DB25 macho.



Puerto paralelo, en el gabinete es hembra DB25 (25 pines)



RS232C: estándar de la EIA (Electronic Industries Association)



Table 9-1. DB-9 Connector and Pinouts

Function	Signal	PIN	DTE	DCE
Data	TxD	3	Output	Input
	RxD	2	Input	Output
Handshake	RTS	7	Output	Input
	CTS	8	Input	Output
	DSR	6	Input	Output
	DCD	1	Input	Output
	DTR	4	Output	Input
Common	Com	5	—	—
Other	RI	9	Input	Output

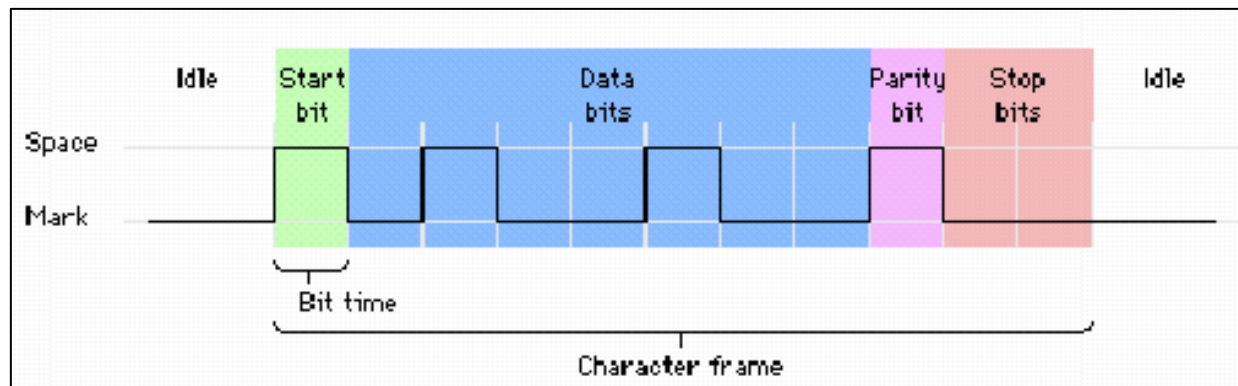
Terminales
asociados

2 líneas de datos

- TxD
- RxD
- GND

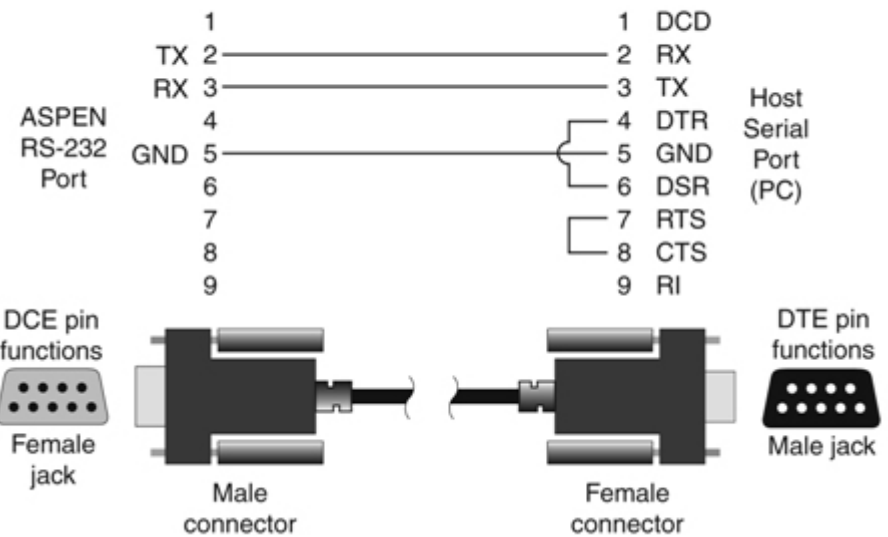
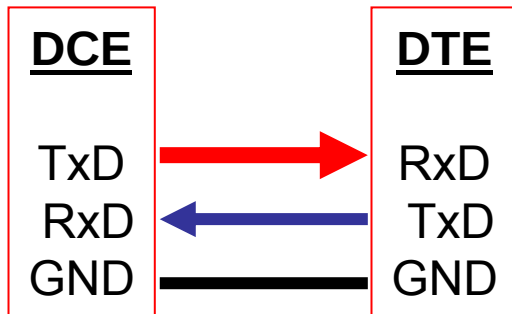
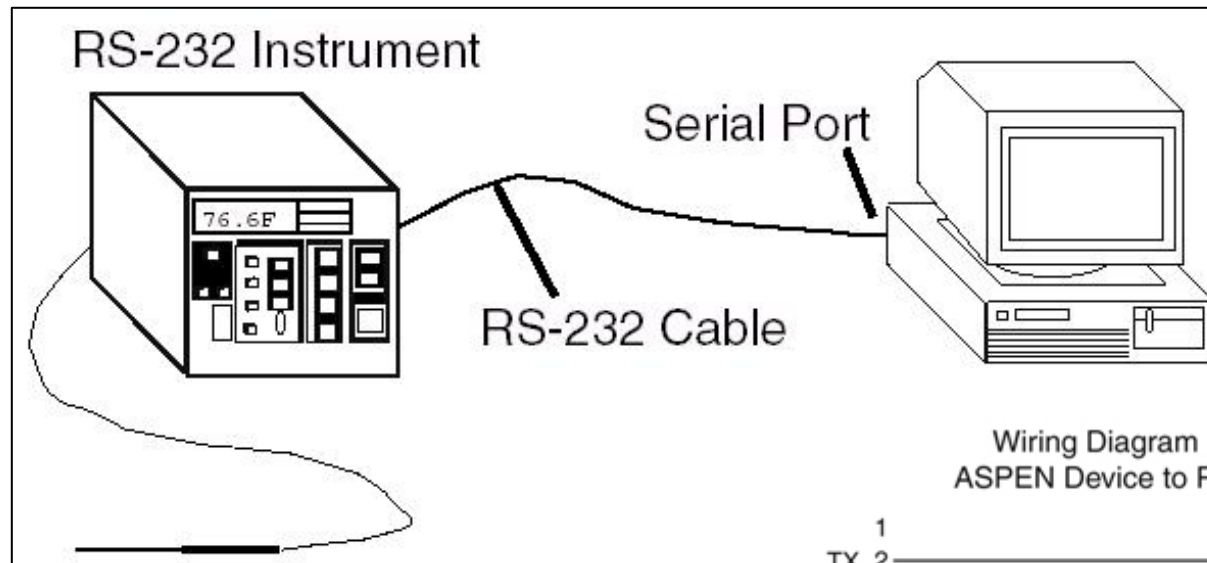
5 líneas de handshaking

- RTS
- CTS
- DSR
- DCD
- DTR



DCE
Data Communications Equipment

DTE
Data Terminal Equipment



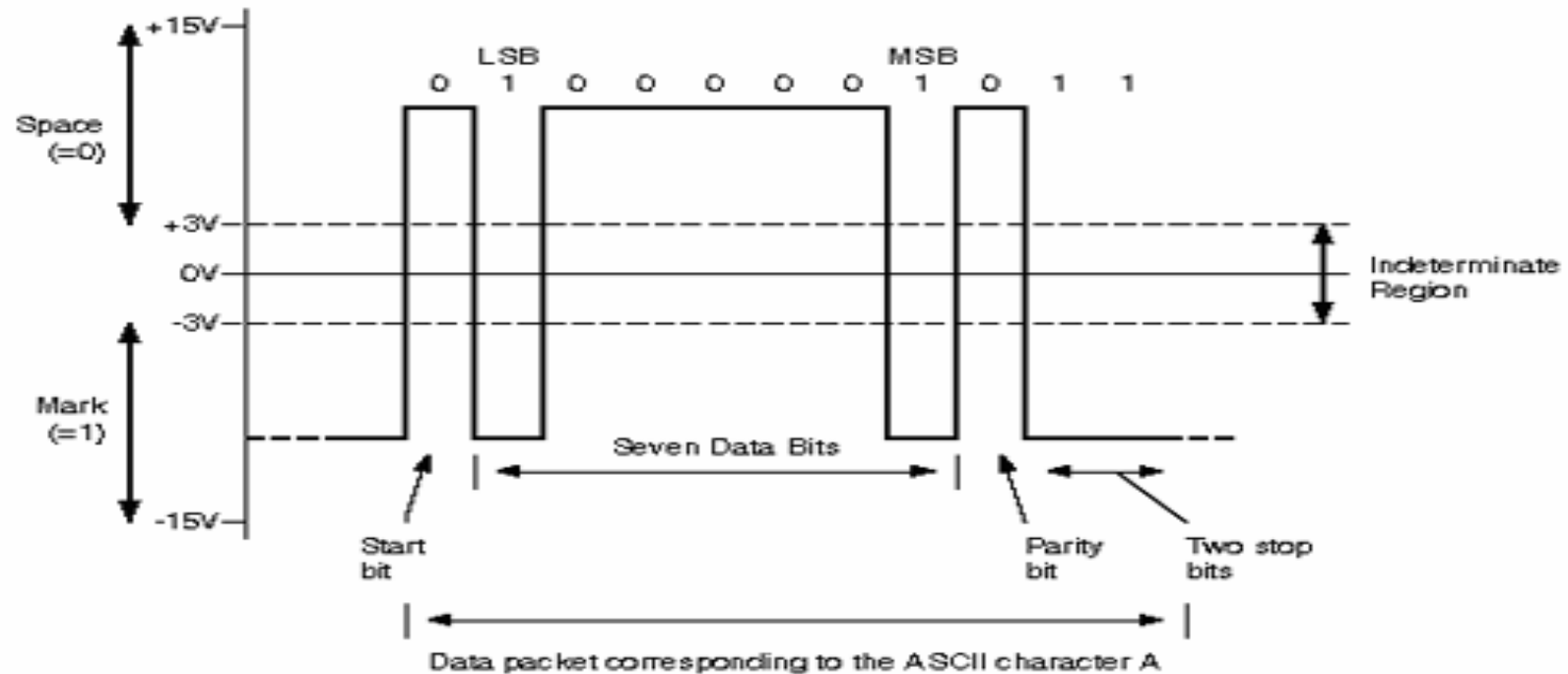
¿Cómo se transmite la información?

Un **uno binario** se denomina “marca” (*mark*) y se representa por un voltaje de -3 a -15 voltios.

Un **cero binario** se denomina espacio (*space*) y se representa por un voltaje de +3 a +15 voltios.

Cualquier voltaje entre -3 y +3 voltios se considera inválido.

Una corriente de corto circuito no puede exceder los 500mA



Representación de caracteres

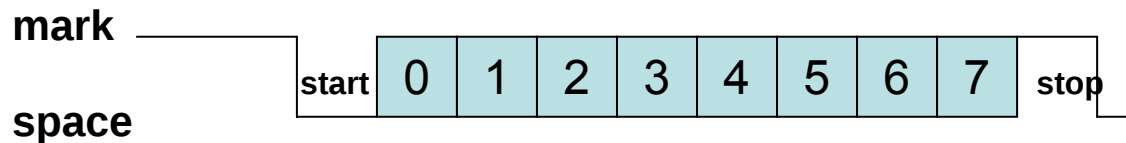
- La paridad puede configurarse de diversas formas:
 - No Parity (sin paridad): No se transmite bit de paridad
 - Even Parity (paridad “par”): el bit de paridad es uno (1) si el carácter lleva un cantidad par de unos.
 - Odd Parity (paridad “impar”): el bit de paridad es uno (1) si el carácter lleva una cantidad impar de unos.
 - Mark Parity (paridad de “marca”): el bit de paridad siempre es uno
 - Space Parity (paridad de “espacio”) : el bit de paridad siempre es cero

DATOS	Cantidad de unos	BIT DE PARIDAD			
		EVEN	ODD	MARK	SPACE
U=01010101	4	1	0	1	0
1111111	7	0	1	1	0
P=01010000	2	1	0	1	0
*=00101010	3	0	1	1	0
1111110	6	1	0	1	0
0011111	5	0	1	1	0
1000000	1	0	1	1	0



Representación de caracteres

- Después del bit de paridad (si lo hay) vienen los bits de parada (stop bits). Estos sirven para decir dónde termina el carácter. Pueden ser uno o dos bits de parada (en esto también deben ponerse de acuerdo el transmisor y el receptor). Algunas implementaciones cortan la transmisión del segundo bit de parada a la mitad, se dice entonces que utiliza uno y medio bits de parada. Los bits de parada se transmiten como unos lógicos (mark).



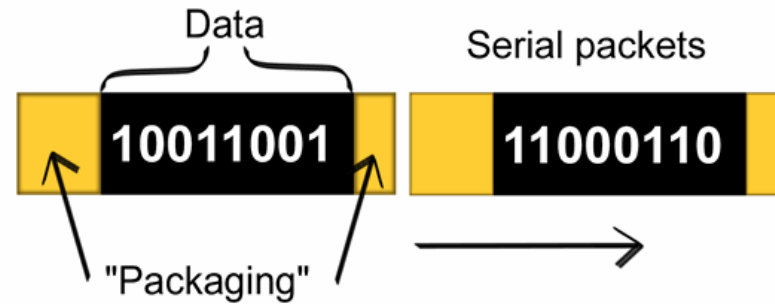
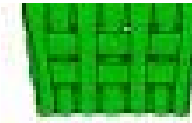
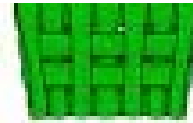
Formulario de configuración de comunicación serial:

- Bits por segundo: 9600
- Bits de datos: 8
- Paridad: Ninguno
- Bits de parada: 1
- Control de flujo: Ninguno

Cuando el bit de parada no se encuentra se produce un *Framing Error*. En estos casos es bueno revisar que el emisor y el receptor esperan la misma cantidad de bits de parada.



Problema: en algún momento cuando la persona que recibe las manzanas completó un canasto, le dice a la que está arriba que espere que lo cambiará.



“handshaking”



Handshaking y Los comandos más usados:



La idea es que exista una señal que diga “tengo un dato para enviarte” o que digan “estoy listo para recibir los datos”.

RTS : Request To Send El receptor hace un “polling” detecta que hay un transmisor e informa

CTS: Clear To Send (esta listo para recibir). El transmisor solo envía datos cuando esta activada la señal de CTS.

Por ejemplo, DTE (PC ó terminal) que envía datos a DCE (modem, equipo, etc).

DTE sube DTR
DCE sube DSR
DTE sube RTS
DCE sube CTS
DCE baja CTS
DCE sube CTS

Estoy en línea y listo
Ok estoy listo también
Estoy listo para enviar
Estoy listo para recibir
Buffer lleno
Estoy listo

DTR (Data-Terminal-Ready)
DSR (Data-Set-Ready)

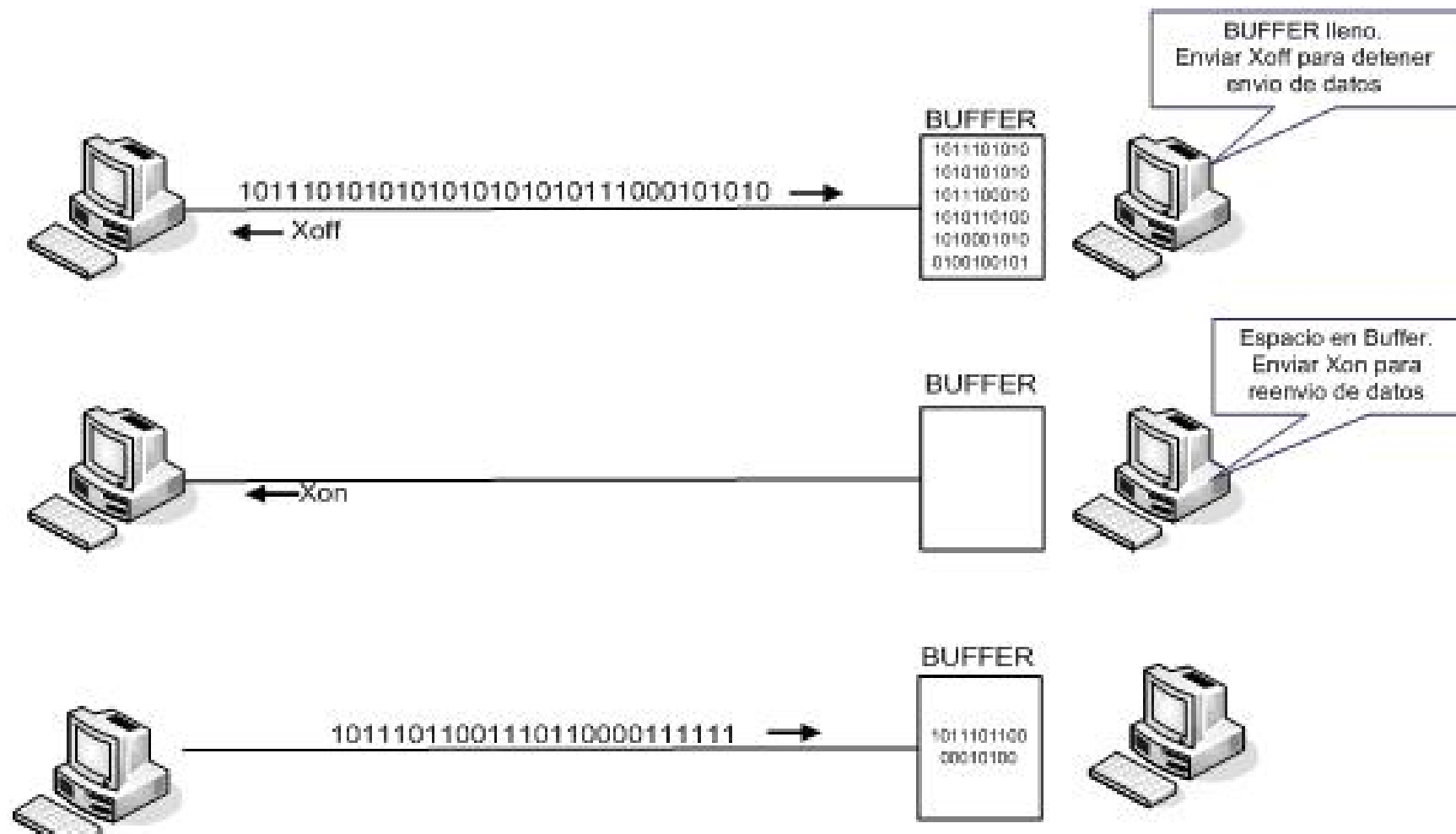
CD (Carrier-Detect)



Control de flujo

- En RS-232 el control de flujo se puede hacer de dos maneras: por hardware (RTS/CTS) o por software (Xon/Xoff).
- **Por Software:** El carácter Xoff (ASCII 19, CTRL-S) es utilizado por el receptor para decir que su buffer está lleno y el emisor debe esperar. Cuando vuelva a tener espacio en el buffer, enviara el carácter Xon (ASCII 17, CTRL-Q), diciéndole que puede volver a transmitir. Esto economiza cables en la interconexión, pero ocupa espacio en el canal (perfecto para un PDA).
- **Por hardware:** a diferencia del anterior, este tipo de control de flujo requiere que entre su PC y su MODEM se conecten dos hilos: RTS y CTS. Cuando el buffer del receptor se llena (supongamos que es un modem “lento”) le dice al PC que espere, desactivando la señal CTS. Cuando vuelva a tener espacio en el buffer, activa nuevamente el CTS para decir que está nuevamente listo (esto puede pasar cuando la UART es más rápida que el MODEM).





Protocolo:

Por otro lado el emisor y el receptor necesitan ponerse de acuerdo en un protocolo para determinar la formalidad de la transmisión.

The screenshot shows a 'Setup' dialog box with the 'Prodel Comms.' tab selected. The dialog has several tabs: 'Optics', 'Timers', 'ID Matrix', 'External Data', 'Custom Data Code', 'Serial Numbers', and 'Data Definitions'. The 'Prodel Comms.' tab contains the following settings:

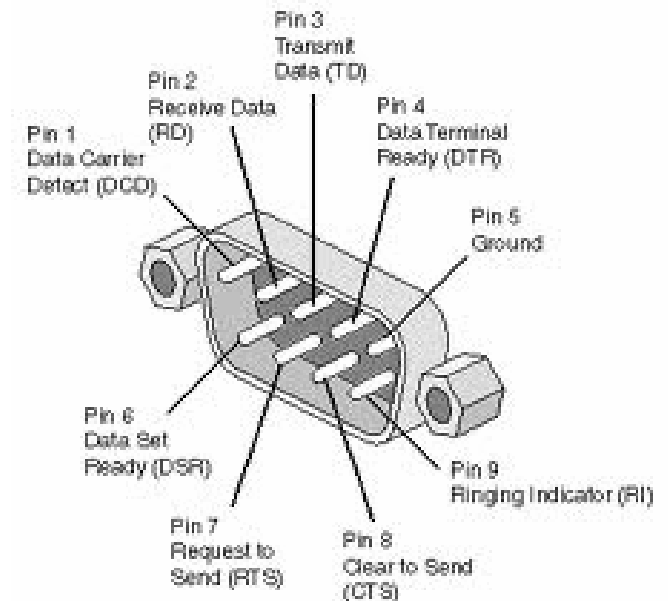
Parameter	Value	Unit
Com Port #:	1	
Baud Rate:	9600	
Data Bits:	8	
Parity:	N	
StopBit:	1	
Port Time Out:	2000	mSec
Msg Time Out:	200	mSec
Data Fault PW:	300	mSec
Write Comp. PW:	300	mSec
Wait Prodel Abort:	150	
STX Char. (Dec):	2	
ETX TChar (Dec):	3	
CSV STX Detection	☐	
CSV ETX Detection	☐	

At the bottom of the dialog are 'OK' and 'Cancel' buttons.

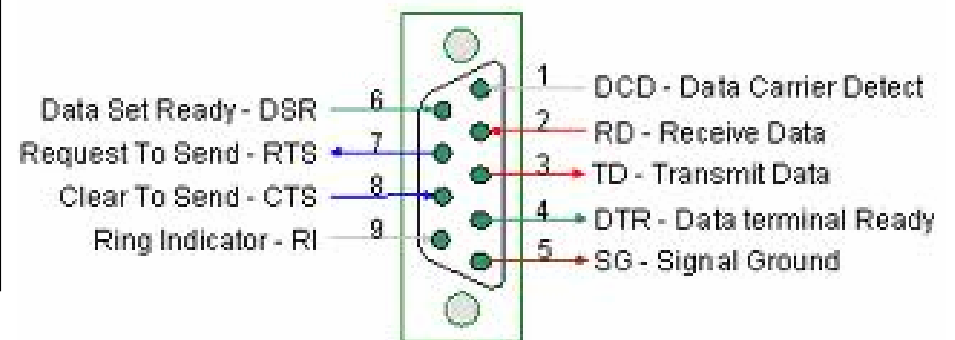


Equivalencias entre DB25 y DB9

DB-9	Descripción	DB-25
1	Carrier Detect	8
2	Receive Data	3
3	Transmitted Data	2
4	Data Terminal Ready	20
5	Signal Ground	7
6	Data Set Ready	6
7	Request To Send	4
8	Clear To Send	5
9	Ring Indicator	22



Esta tabla sirve para construir un conversor de 25 a 9 pines. Por ejemplo, el hilo para *carrier detect* debe ser soldado en el pin 1 del conector DB-9 y en el pin 8 del conector DB-25



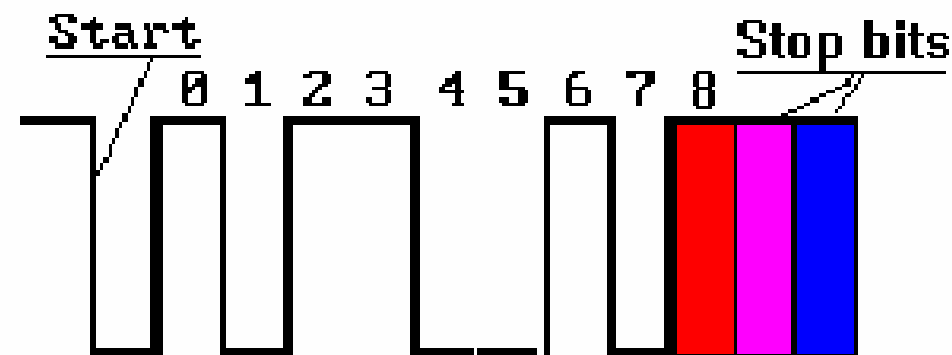
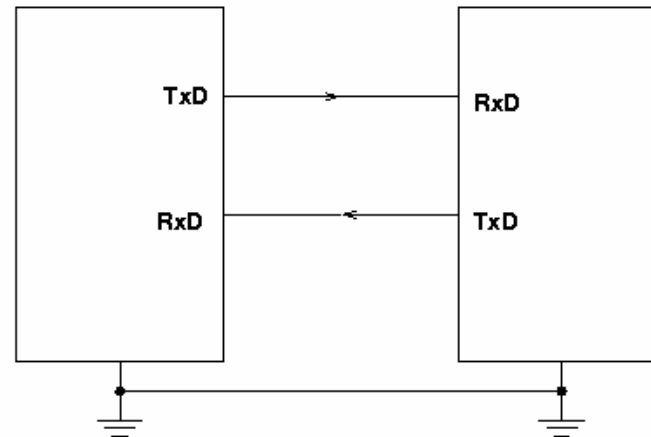
Transmisión de datos sincrónica:

- Reloj interno (*internal timing*): El MODEM (DCE) sincrónico coloca una onda cuadrada sobre el pin 15 con una frecuencia igual a la tasa de transmisión de bits. Así el DTE puede sincronizar la transmisión de datos sobre el pin 2.
-
- Cuando un MODEM (DCE) sincrónico recibe una señal que viene de la línea telefónica, éste coloca una onda cuadrada sobre el pin 17 para contarle al DTE la tasa de transferencia, al mismo tiempo los datos se reciben a través del pin 3. Este es el reloj utilizado para escuchar (recibir).
- reloj del exterior (*external timing*) la terminal (DTE) “coloca el reloj” en el pin 24 para contarle al DCE cuál será la tasa de transferencia.



Transmisión asincrónica en el puerto serial , no se transmiten los datos con el ritmo del reloj hay que marcar donde empieza el byte y donde termina..

Asynchronous Serial Communications



Octavo bit de datos o de paridad
1er Stop Bit 2do Stop Bit



Transmisión de datos Asincrónico vs. sincrónica

- En la transmisión sincrónica se requieren señales que permitan poner de acuerdo a los dos modems en relación con el tiempo (timing signals).
- Diferente a los bits de parada e inicio utilizados en la transmisión asincrónica.
- La transmisión sincrónica es más eficiente, pues se economiza los bits de inicio, paridad y parada.
 - Transmitiendo 1024 bytes de forma asincrónica, con un bit de arranque, uno de parada, sin paridad y ocho de datos se tiene una eficiencia del 80% (10 bits para representar 8: 80% de datos y 20% de overhead)
 - Transmitiendo 1024 bytes sincrónicos y suponiendo que se utilizan 2 bits de arranque, 4 bytes de control al comienzo y, al final, 4 bytes de checksum y 2 bits de parada (total 68 bits de overhead), se tiene una eficiencia del 99%.

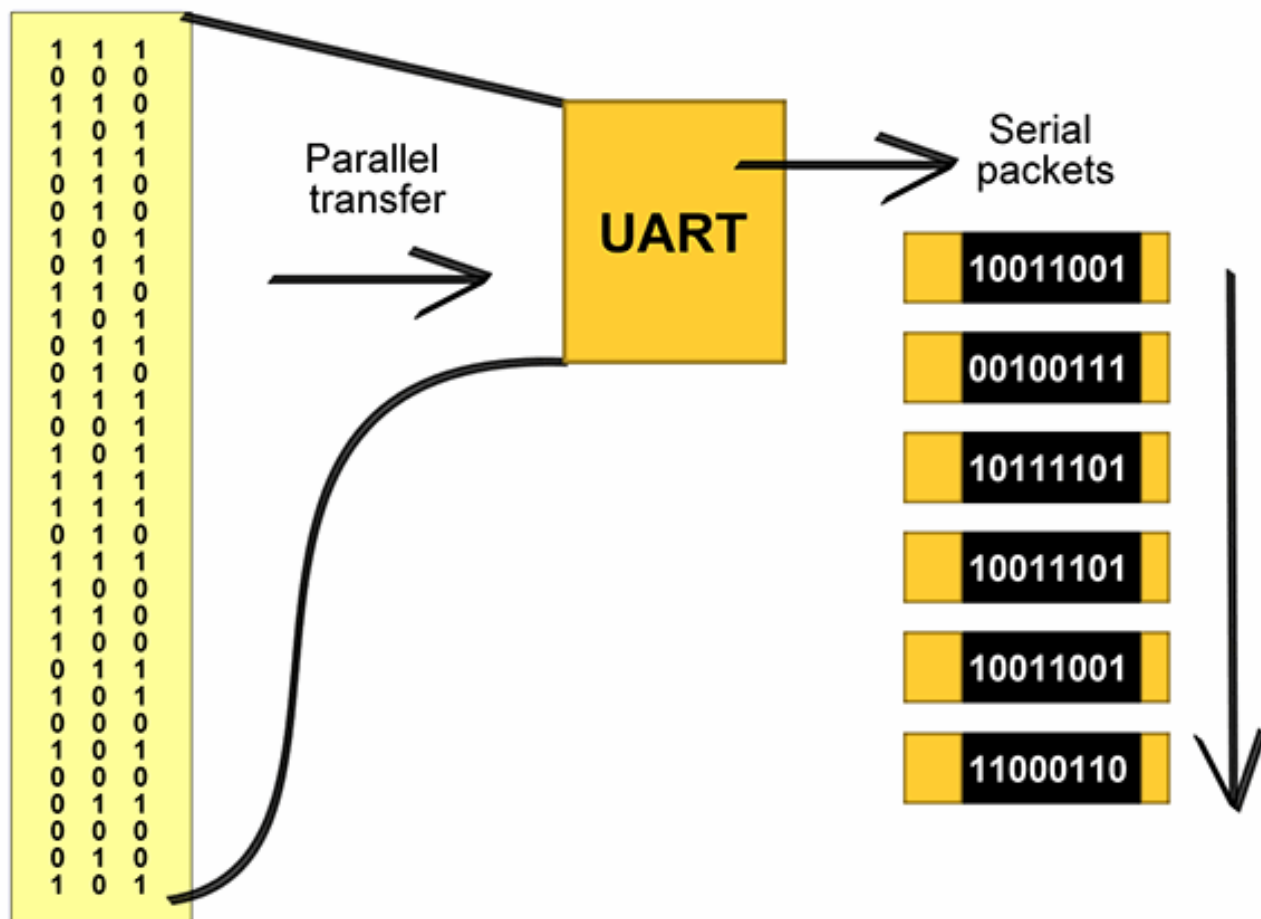


UART Universal Asynchronous Receiver/Transmitter

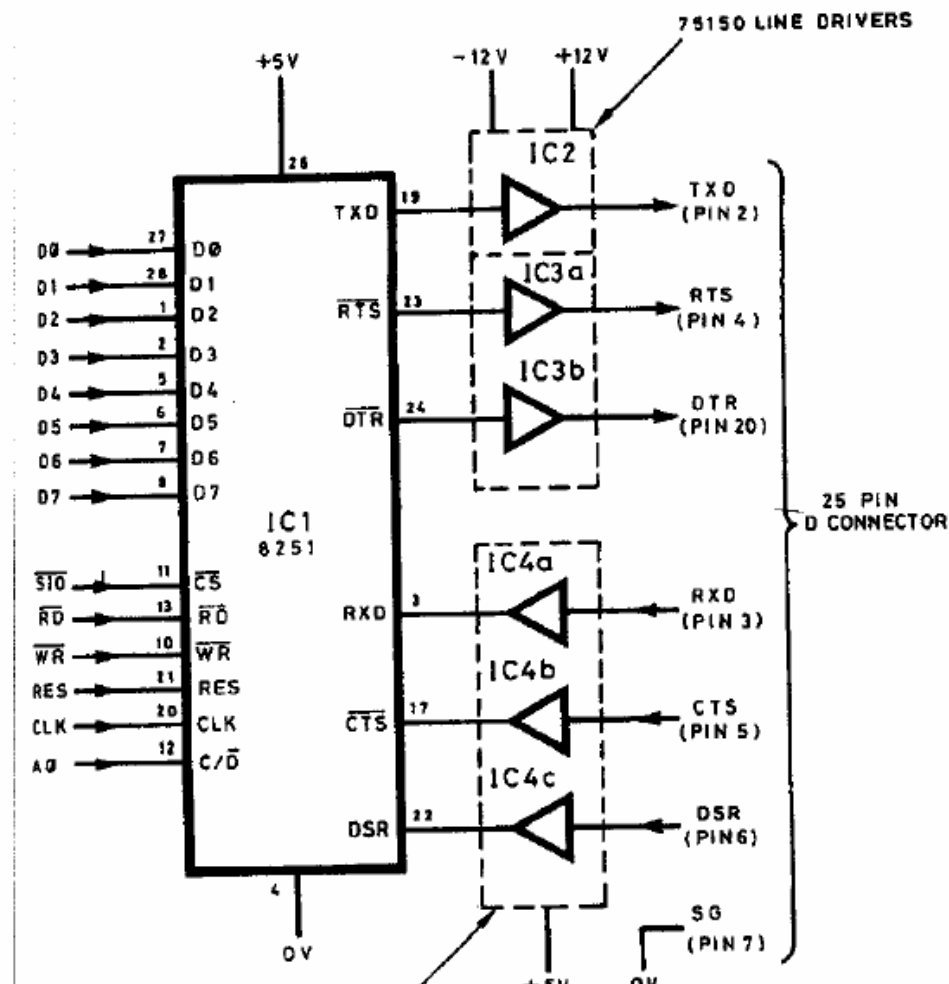
Palabra-Dato Byte (BUS PC)

INTERFACE

Cadena de 7 ú 8 bits (ASCII)



Disposición simplificada de una interfase RS-232 de computadora :



Salidas
TxD ,RTS, y DTR

Tanto IC2 como IC3 requieren alimentaciones de +/- 12V.

El "1" y "0" lógicos serán representados por -12 y +12V respectivamente.

Entradas

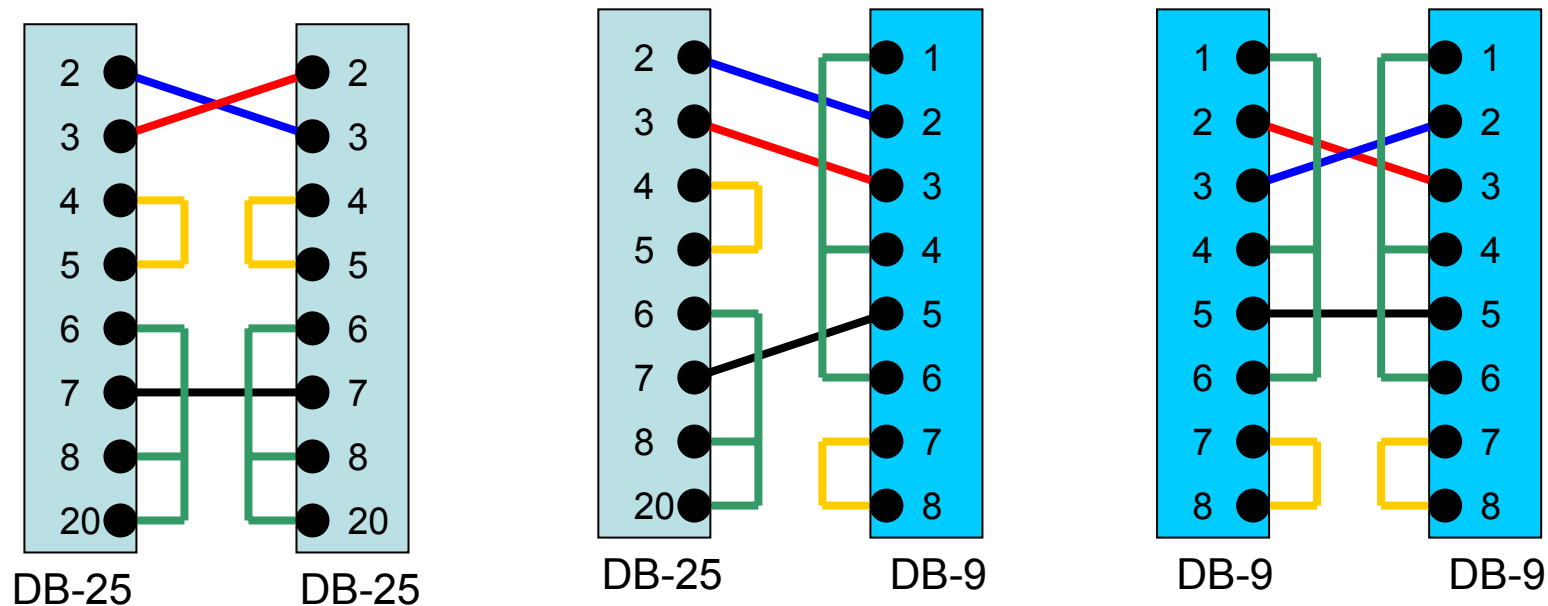
RxD ,CTS, y DSR

IC1 (8251 ó 8250) es un chip serie programable que convierte las señales del bus (paralelo) de la computadora (DTE) en señales serie de salida hacia el DCE y viceversa con las señales serie que se dirigen hacia la computadora .Este chip es denominado UART *Universal Asynchronous Receiver Transmitter* .



RS-232 Null MODEM

Un cable null modem se utiliza para conectar dos DTEs directamente a través de interfaces RS-232. Los siguientes diagramas muestran null modems elaborados con solo tres hilos. La idea es hacer pensar al DTE que está conectado a un DCE. El indicador de ring (pin 22 en DB-25 y pin 9 en DB-9) no se necesitan pues no hay línea telefónica.



¿Qué cable serie utilizar entre el DTE y DCE?

Básicamente tenemos 2 tipos de equipos RS232.

DTE (*Data Terminal Equipment*, o *Equipo Terminal de Datos*) y

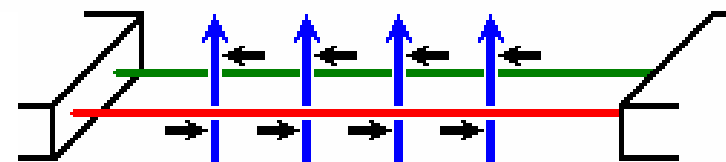
DCE (*Data Communication Equipment*, o *Equipo de Comunicación de Datos*).

Un PC es un DTE. Un módem es un DCE. Una estación metereológica suele ser un DCE, un sensor de temperatura suele ser un DCE, ...

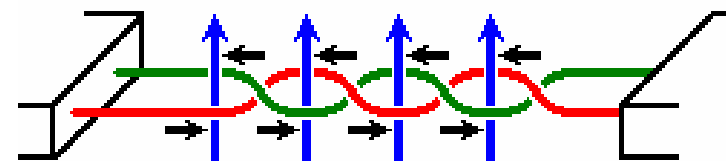
DTE Device (Computer)			DB9	DTE to DCE Connections	DCE Device (Modem)	DB9	
Pin#	DB9	RS-232 Signal Names		Signal Direction	Pin#	DB9	RS-232 Signal Names
#1	Carrier Detector (DCD)	CD			#1	Carrier Detector (DCD)	CD
#2	Receive Data (Rx)	RD			#2	Receive Data (Rx)	RD
#3	Transmit Data (Tx)	TD			#3	Transmit Data (Tx)	TD
#4	Data Terminal Ready	DTR			#4	Data Terminal Ready	DTR
#5	Signal Ground/Common (SG)	GND			#5	Signal Ground/Common (SG)	GND
#6	Data Set Ready	DSR			#6	Data Set Ready	DSR
#7	Request to Send	RTS			#7	Request to Send	RTS
#8	Clear to Send	CTS			#8	Clear to Send	CTS
#9	Ring Indicator	RI			#9	Ring Indicator	RI
Soldered to DB9 Metal - Shield		FGND			Soldered to DB9 Metal - Shield		FGND



DCE Device (Modem)			DB9		DCE to DCE Connections		DCE Device (Modem)		DB9	
Pin#	DB9	RS-232 Signal Names			Signal Direction		Pin#	DB9	RS-232 Signal Names	
#1		Carrier Detector (DCD)		CD			#1		Carrier Detector (DCD)	
#2		Receive Data (Rx)		RD			#2		Receive Data (Rx)	
#3		Transmit Data (Tx)		TD			#3		Transmit Data (Tx)	
#4		Data Terminal Ready		DTR			#4		Data Terminal Ready	
#5		Signal Ground/Common (SG)		GND			#5		Signal Ground/Common (SG)	
#6		Data Set Ready		DSR			#6		Data Set Ready	
#7		Request to Send		RTS			#7		Request to Send	
#8		Clear to Send		CTS			#8		Clear to Send	
#9		Ring Indicator		RI			#9		Ring Indicator	
Soldered to DB9 Metal - Shield				FGND			Soldered to DB9 Metal - Shield			



Straight cable



Twisted pair cable

 **Magnetic field**
 **Induced noise current**



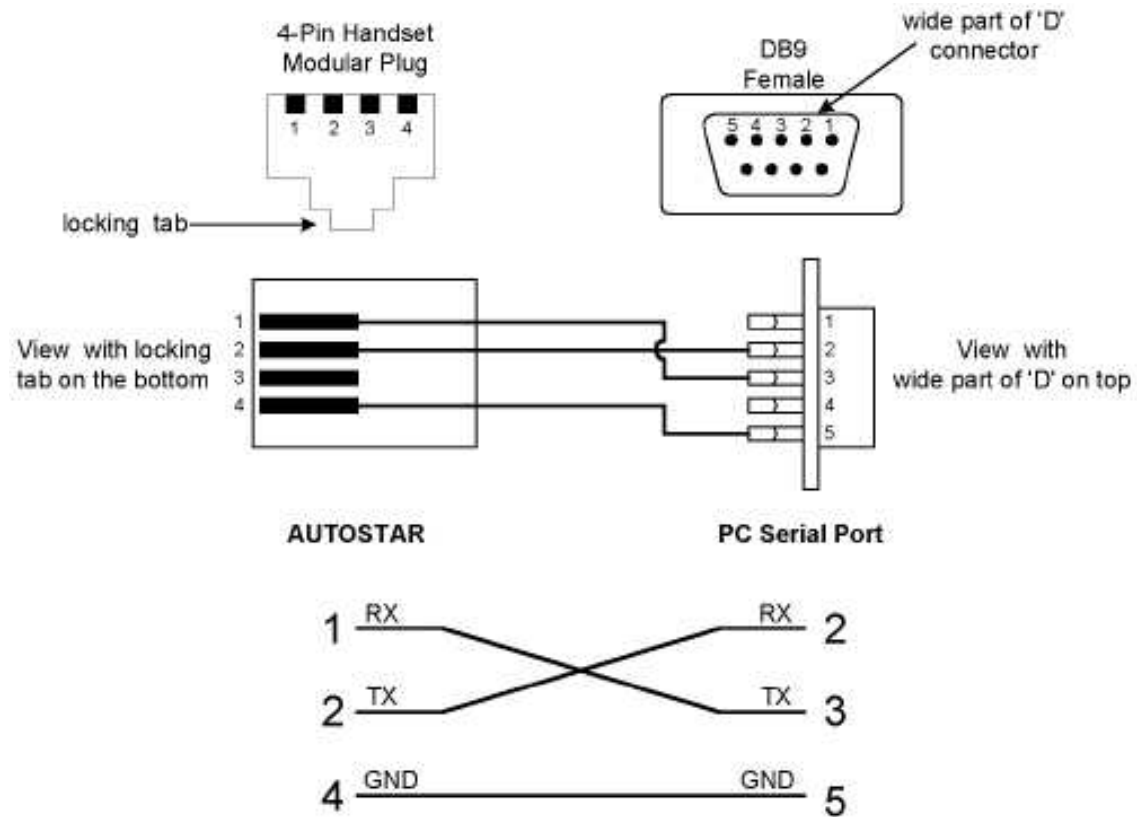
	RS232	RS485
Differential	no	yes
Max number of drivers	1	32
Max number of receivers	1	32
Network topology	point-to-point	multipoint
Max distance (acc. standard)	15 m	1200 m
Max speed at 12 m	20 kbs	35 Mbs
Max speed at 1200 m	(1 kbs)	100 kbs
Max slew rate	30 V/ μ s	n/a
Receiver input sensitivity	± 3 V	± 200 mV
Receiver input range	± 15 V	$-7..12$ V
Max driver output voltage	± 25 V	$-7..12$ V
Min driver output voltage (with load)	± 5 V	± 1.5 V



COMPARACIÓN RS232 y RS485

- El uso de tensiones elevadas de hasta 15V en RS-232 y de circuitos no balanceados hace que sea más susceptible al ruido.
- En cambio en RS-485 se utilizan voltajes de cómo máximo 6V y circuitos balanceados por lo que se reduce el factor de ruido.
- Con RS-485 se permiten conectar hasta 64 dispositivos.





USB-1408FS

USB-based Analog and Digital I/O Module



Figure 4. USB-1408FS Screw terminal pin numbers



Figure 1. USB-1408FS



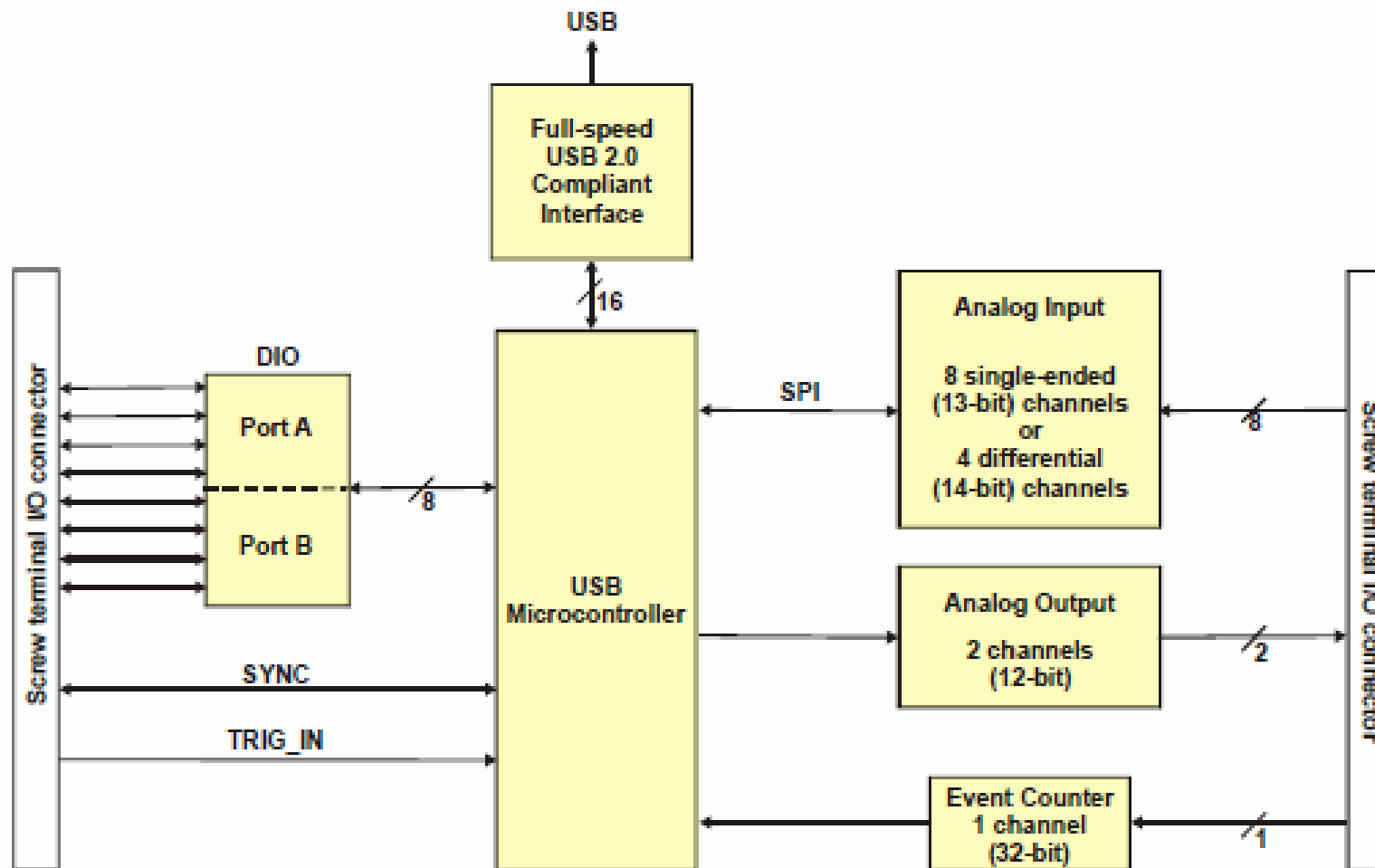
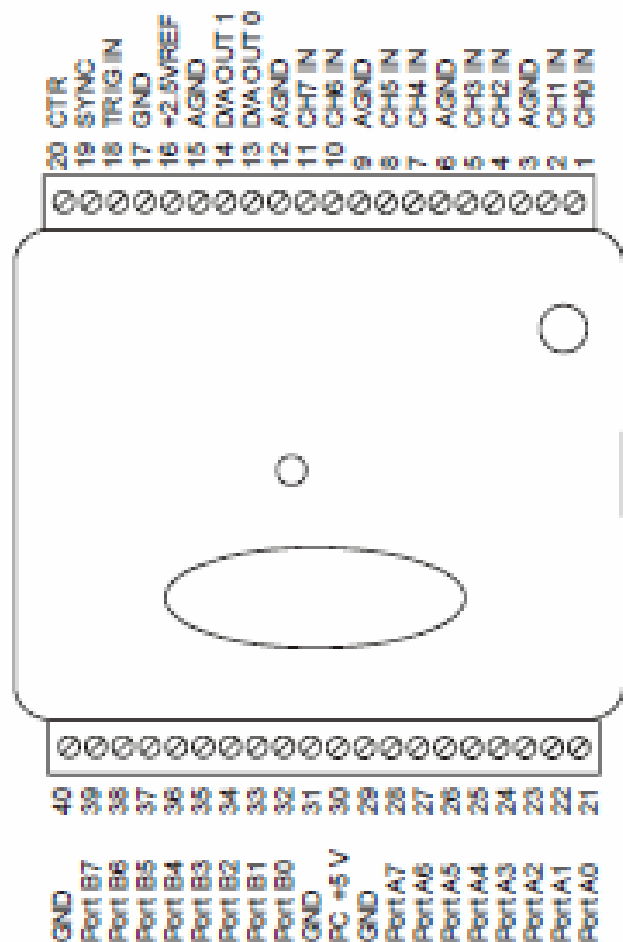


Figure 2. USB-1408FS functional block diagram





8-channel single-ended mode pin out

- Eight analog input connections (CH0 IN to CH7 IN)
- Two analog output connections (D/A OUT 0 to D/A OUT 1)
- One external trigger source (TRIG_IN)
- One SYNC terminal for external clocking and multi-unit synchronization (SYNC)
- One voltage output source (2.5VREF)
- Five analog ground connections (AGND)
- One ground connection (GND)
- One external event counter connection (CTR)

16 digital I/O connections (Port A0 to Port A7, and Port B0 to Port B7)

One power connection (PC+5 V)

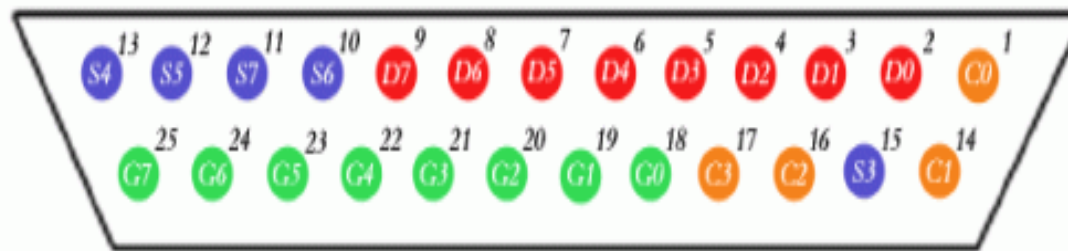
Three ground connections (GND)



Puerto Paralelo – Printer PC LPT1 - CENTRONIC

IEEE 1284 Esta es la interface paralelo usual. 300 kB/sec.

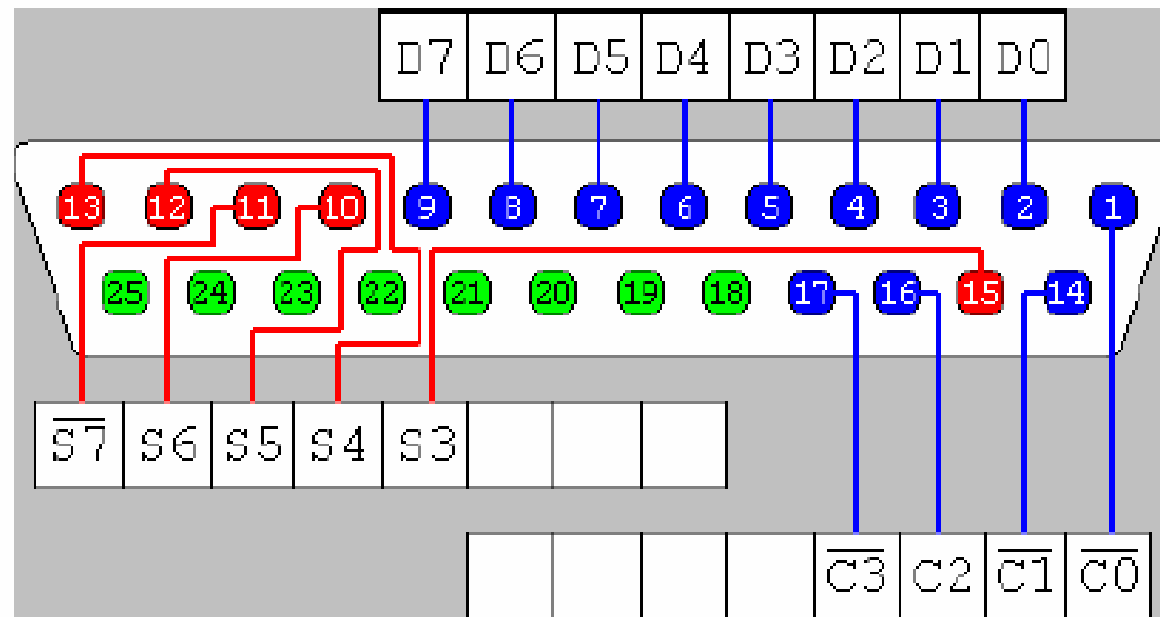
Cables de hasta 2 metros. 1 equipo



Aplicaciones del Standard Printer Port

Existen muchos tipos de dispositivos ademas de los printers que se pueden conectar a este puerto: voltímetros electronicos con salida paralelo, tarjetas de adquisición de datos, discos magnéticos-opticos (ZIP drivers), control de motores paso a paso, comunicación entre computadoras PC-PC, manejo de display de cuarzo líquido LCD, conversores DAC y ADC , DATA LOGIN SYSTEM (capturador de datos) , control de experimentos , sistemas de control y printers en general.





- 8 pines de salida de datos DATA Port D0-D7
- 5 pines de entrada de señales (uno invertido) STATUS Port S3-S7
- 4 pines de salida de señales (tres invertidos) CONTROL Port C0-C3
- 8 pines restantes estan conectados a masa



En el DATA Port: D0 es el bit menos significativo (LSD least significant data) y D7 corresponde al bit más significativo (MSD most significant data).

Para las salidas tiene 12 buffers TTL que estan "latched" los cuales pueden ser "escritos" o "leídos" con cualquier programa que los controle con una simple instruccion de "in" o "out". Estos pines tienen una corriente de "source" (de salida hacia una carga referida a masa) de 2,6 mA ; y una corriente de "sink" (de entrada desde una carga referida a 5 V) de 24 mA.

Cada uno de estos registros DATA Port, STATUS Port y CONTROL Port tiene una dirección específica en el mapa I/O de la PC para un puerto paralelo simple (puede tener hasta tres puertos paralelo).

Base I/O	378 h	DATA
Base +1	379 h	STATUS
Base +2	37A h	CONTROL

Una señal "high" (en un pin o una linea) es un valor TTL alto, +2.4 a +5V.

Una señal "low" es un valor TTL bajo, 0 a +0,8V.

Un dato High es un "1" lógico y un un low es "0" lógico.



PRINTER

At Standard TTL Levels			
	Signal Name	Adapter Pin Number	
	← - Strobe	1	
E	← +Data Bit 0	2	P
X	← +Data Bit 1	3	A
T	← +Data Bit 2	4	R
E	← +Data Bit 3	5	A
R	← +Data Bit 4	6	L
N	← +Data Bit 5	7	L
A	← +Data Bit 6	8	E
L	← +Data Bit 7	9	L
	→ - Acknowledge	10	
D	→ +Busy	11	A
E	→ +Paper End	12	D
V	→ +Select	13	A
I	← -Auto Feed	14	P
C	→ -Error	15	T
E	← -Initialize	16	E
	← -Select Input	17	R
	→ Ground	18-25	

IBM-PC

Las señales con "-" son activas en bajo; y las señales con "+" son activas en alto.

Los principales tipos y nombres de canales que son utilizados como control son:

1. **STROBE** - a través de el, el ordenador comunica al periférico que esta preparado para transmitir.
2. **BUSY** - el periférico comunica a través de el, que no esta preparado para recibir datos.
3. **ACK** - el periférico comunica a través de el, que esta preparado para recibir datos.
4. **SELECT Y SELECTIN** - indican el tipo de error producido en el periférico.
5. **ERROR** - indica que se ha producido un error en el periférico.
6. **PE** - depende del tipo del periférico, en el caso de la impresora indica que no tiene papel.



$\Leftarrow$ in	DB25	Centronics	Name of Reg	Signal Bit	Function Notes
$\Rightarrow$ out	pin	pin			
$\Rightarrow$	1	1	-Strobe	C0-	Set Low pulse >0.5 us to send
$\Rightarrow$	2	2	Data 0	D0	Set to least significant data
$\Rightarrow$	3	3	Data 1	D1	...
$\Rightarrow$	4	4	Data 2	D2	...
$\Rightarrow$	5	5	Data 3	D3	...
$\Rightarrow$	6	6	Data 4	D4	...
$\Rightarrow$	7	7	Data 5	D5	...
$\Rightarrow$	8	8	Data 6	D6	...
$\Rightarrow$	9	9	Data 7	D7	Set to most significant data
$\Leftarrow$	10	10	-Ack	S6+ IRQ	Low Pulse ~ 5 uS, after accept
$\Leftarrow$	11	11	+Busy	S7-	High for Busy/Offline/Error
$\Leftarrow$	12	12	+PaperEnd	S5+	High for out of paper
$\Leftarrow$	13	13	+SelectIn	S4+	High for printer selected
$\Rightarrow$	14	14	-AutoFd	C1-	Set Low to autofeed one line
$\Leftarrow$	15	32	-Error	S3+	Low for Error/Offline/PaperEnd
$\Rightarrow$	16	31	-Init	C2+	Set Low pulse $> 50\mu\text{S}$ to init
$\Rightarrow$	17	36	-Select	C3-	Set Low to select printer
$=$	18-25,	19-30,33, 17, 16	Ground		



Tarjeta comunicación en paralelo PC-equipos d medición HP-IB, GPIB ó IEEE-488

Equipo para medir
con interface GPIB



IEEE488 - USB para PC



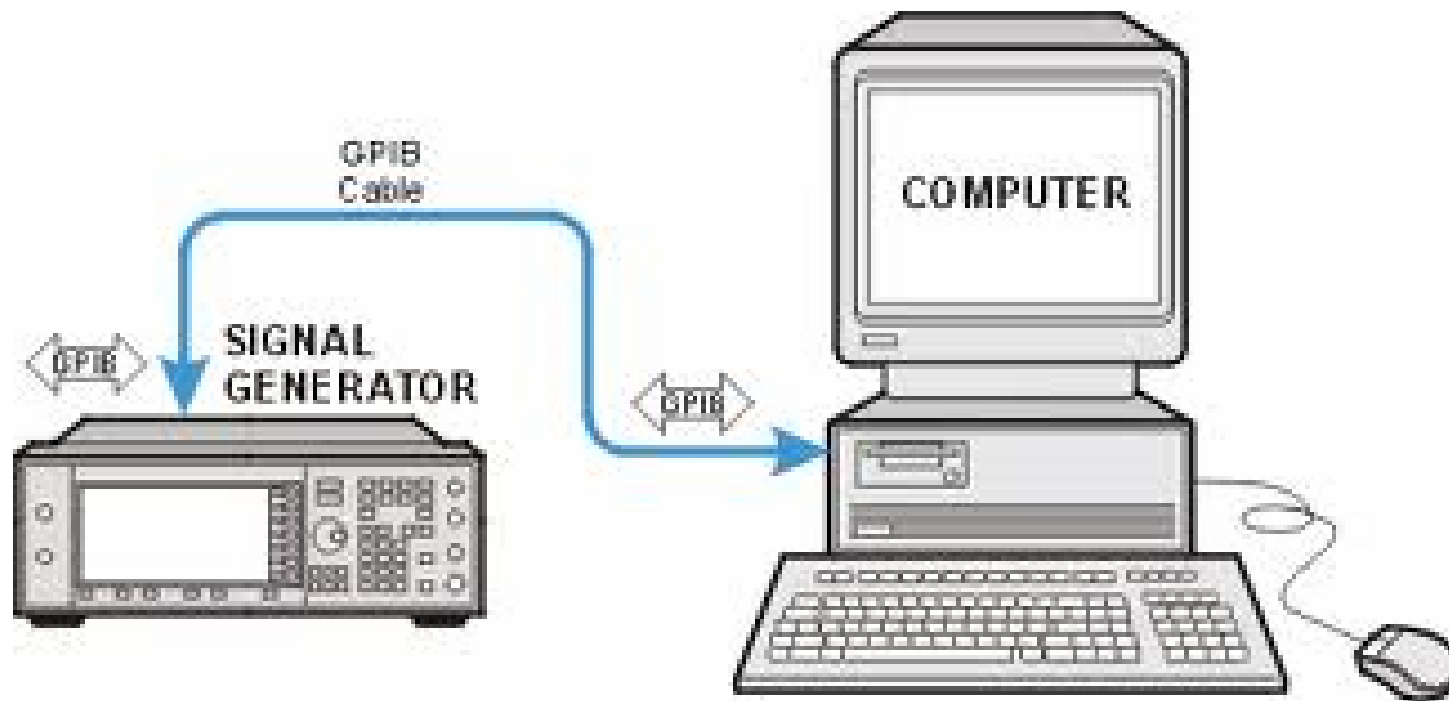
Placa IEEE488 - PCI para PC



Cable paralelo norma IEEE-488.2



Es una norma (standar) de comunicación digital entre instrumentos programables, utilizada para interconexión de los mismos. Como tal, determina claramente todas las especificaciones: mecánicas (por ejemplo tipo y tamaño de conectores, cables, etc.), eléctricas (niveles de tensión, timing de señales, etc.) y protocolos de hardware y software que deben cumplir los instrumentos que respondan a ella.



EVOLUCION HISTORICA

1965 - La empresa Hewlett-Packard crea la norma interna HP-IB (Hewlett-Packard Interface Bus) para la interconexión de su propia línea de instrumentos programables y computadoras personales. Debido a su alta performance, gana rápida popularidad.

1975 - Con ligeras modificaciones, esa norma es aceptada por el Institute of Electrical and Electronic Engineers de USA como el standar I-EEE 488 .

1978 – Se hace una revisión con la denominación I-EEE 488.1.

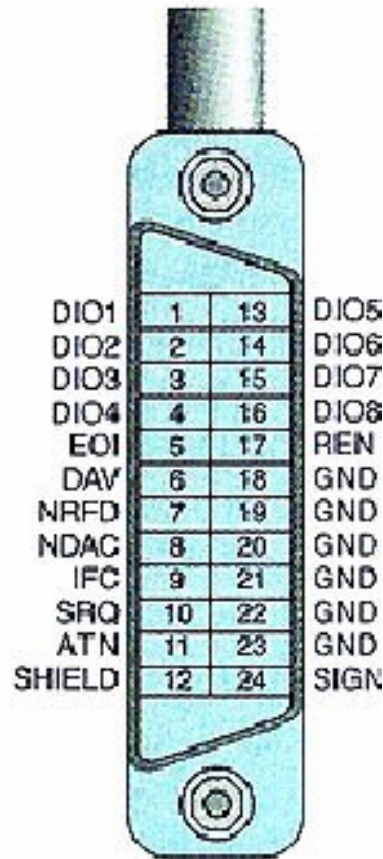
1987 - Se introduce una nueva revisión, evolucionando hacia la I-EEE 448.2. Aparece la denominación GP-IB (General Purpose Interface Bus).

1990 - Se simplifica la estructura de software definida en la versión anterior , creando un set de comandos único y más simple. Se llega así a la SCPI (Standar Comands for Programable Instruments).

1993 - Empresas como National Instrumens generan versiones de alta velocidad que denominan HS488 .

2008- Convertidor GPIB → USB





cables
~ US \$ 100



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	
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	

Características

Alta velocidad de transferencia de datos.

Compatibilidad total IEEE488, IEEE488.1, IEEE488.2, SCPI.

Compatible con LabWindows, LabVIEW, VIEWDAC, LABTECH NOTEBOOK, TESTPOINT, ASYST, ..., etc.

Pueden ser conectados hasta 14 dispositivos con una sólo tarjeta.
Soporta hasta 20 metros de cable.

La velocidad de transferencia de datos es de 1Mbyte/seg. (La PCI-488 5 Mbytes/s).

Autoconfiguración Plug & play. (PCI-488).

Soporta todos los comandos y funciones IEEE488.2 como por ejemplo: listen, talk, device clear, group execute trigger, go to local, interface clear, local lockout, parallel poll configure or disable, remote enable, selected device clear, serial poll, pass control, unlisten, untalk.



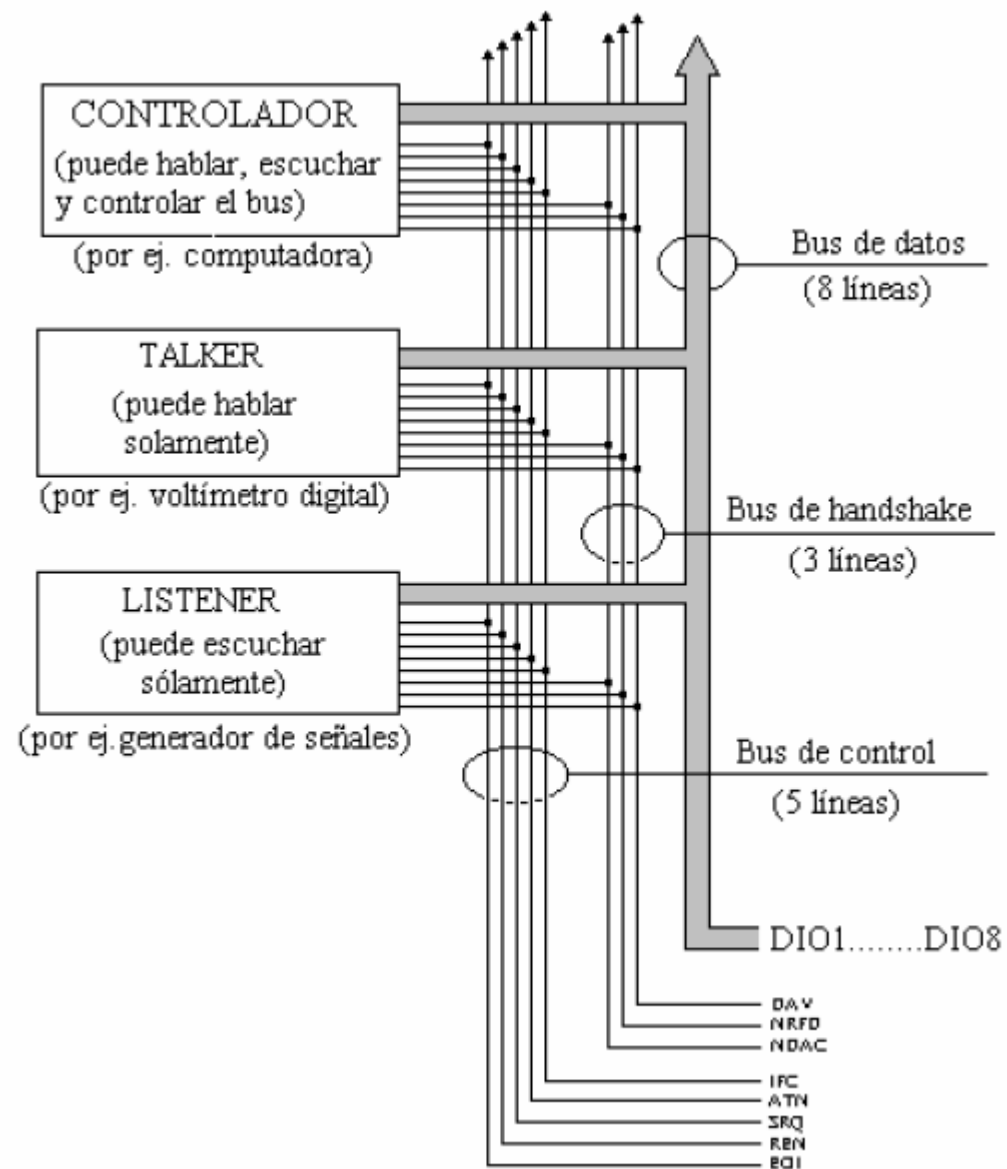
ESTRUCTURA

La GP-IB es una norma de comunicación paralelo (todos los dispositivos estan conectados al mismo grupo de cables simultáneamente) y es capaz de interconectar hasta 15 instrumentos a la vez.

Es de alta velocidad (1Megabyte/seg. o más), razón por la cual la norma establece claramente limitaciones en el largo de los cables y la separación física entre instrumentos (generalmente no más de 15 metros de distancia entre el primer y el último instrumento con una separación promedio entre ellos no mayor a un metro).

Emplea un total de 16 líneas distribuidas en tres buses (grupos de cables) cuyas funciones se distribuyen de la siguiente manera:





Los datos son transmitidos y recibidos en paquetes de 8 bits (bytes) por el bus del mismo nombre, generalmente como códigos ASCII (American Standar Code for Information Interchange).

El bus de líneas de control se utiliza para dirigir del “tránsito” de la información por el bus de datos, y el bus de líneas de handshake para la sincronización de ese tránsito.

Los dispositivos (devices) interconectados por esta norma pueden desempeñar tres tipos de funciones:



-LISTENER. Un dispositivo capaz de recibir datos desde otros instrumentos.
Ej. printers, fuentes de señal programables, etc.

-TALKER . Un dispositivo capaz de transmitir datos hacia otros instrumentos.
Ej. Multímetros programables, contadores, frecuencímetros, etc.

-CONTROLLER. Un dispositivo capaz de dirigir el tránsito de la información a través de la interface, mediante el manejo de las líneas del bus de control y de comandos (por ej. direccionamientos, etc.) mandados por el bus de datos.

Generalmente cada dispositivo conectado a la interfase asume un único rol, pero puede suceder que alguno de ellos cumpla más de uno, como por ejemplo ser listener en un determinado momento y talker en otro.



DESCRIPCION DE LAS LINEAS.

Ocho líneas de datos.

Son ocho en total :DIO1 a DIO8, siendo DIO1 la menos significativa (Least Significant Bit) y DIO8 la más significativa (Most Significant Bit). Ellas transportan comandos o datos dependiendo del estado de la línea ATN; si ATN está activa las líneas de DIO transportan comandos y si ATN está inactiva, datos. Como se mencionó anteriormente tanto los datos como los comandos son transmitidos como códigos ASCII los cuales necesitan sólo 7 bits, utilizándose el bit 8 generalmente para chequeo de paridad.



Cinco líneas del bus de control.

IFC (Interface Clear). línea manejada sólo por el controlador de la interface e indica a todos los instrumentos que se pongan inactivos.

REN (Remote Enable). línea manejada por el controlador . Indica a los instrumentos que se pongan en modo de programación remota (dejan de reponder del panel frontal).

SRQ (Service Request). línea manejada por los instrumentos. Se utiliza cuando necesitan atención del controlador (por ejemplo, cuando tienen un dato que pasarle). El controlador deberá hacer un polling de los instrumentos para saber cuál de ellos solicitó la atención.

EOI (End or Identify) a) el controlador la utiliza (con ATN) para identificar al dispositivo que solicitó el SRQ y b) un dado talker la utiliza para marcar el final de un mensaje que requiera más de un byte por el bus de datos (cadena o “string”).

ATN (Attention) el controlador determina la forma en que deben interpretarse los datos presentes en ese bus. Si ATN está activa la GP-IB está en modo comando y los instrumentos reciben comandos por el bus de datos. Si ATN no está activa el bus de datos es usados por los instrumentos para el tránsito de información.



Tres líneas de handshake.

controlan el “timing” del tránsito de la información por el bus de datos para evitar colisiones que se traducirían en errores de transmisión y/o recepción.

NRFD (Not Ready For Data). simultáneamente por todos los dispositivos (listeners y talkers) cuando se está en modo comando (el controlador está mandando comandos por el bus) y por todos los listeners cuando se está en modo data. Cuando está activa, indica que al menos uno de los que la manejan no está listo para recibir mensajes (not ready). El dispositivo que pretenda poner un nuevo comando o dato en el bus deberá esperar a que esta línea esté inactiva.

NDAC (No Data Accepted). línea manejada simultáneamente por todos listeners y talkers en el modo comando y por los listeners en el modo data. Cuando está activa indica que al menos uno de los estos dispositivos mencionados no terminó de aceptar el comando o el dato presente en el bus. El dispositivo que puso el comando o el dato en el bus deberá ahora esperar a que esta línea esté inactiva para removerlo.

DAV (Data Valid). Esta línea es manejada por el controlador cuando está mandando comandos o por un talker cuando está mandando datos por el bus. Cuando está activa indica a todos los dispositivos que lo que está presente en el bus es un comando o un dato válido y pueden leerlo.



Function	Abbreviation	Description / Example
Source Handshake	SH	SH1 - complete
Acceptor Handshake	AH	AH1 - complete
		T5 - responds to serial poll; untalks when listen address received; talk only capability
Basic Talker	T	T6 - untalks when listen address received; no talk only T7 - no serial poll; untalks when listen address received; talk only capability
Extended Talker	TE	TE0 - no extended talker
Basic Listener	L	L3 - Listen only mode; unlistens if talk address received L4 - Unlistens if talk address received
Extended Listener	LE	LE0 - no extended listener
Service Request	SR	SR0 - no service request capability SR1 - complete
Remote-Local	RL	RL0 - no local lockout RL1 - complete
Parallel Poll	PP	PP0 - does not respond to Parallel Poll
Device Clear	DC	DC1 - complete
Device Trigger	DT	DT0 - no device trigger capability DT1 - complete
Controller	C	C0 - no controller function
	E	E1 - open collector drive electronics E2 - three state drivers

