Técnicas Digitales II

Transmisor-Receptor Asíncrono Universal - *Universal Asynchronous Receiver/Transmitter* (UART)

Transmisor-Receptor Asíncrono Universal - Universal Asynchronous Receiver/Transmitter (UART)

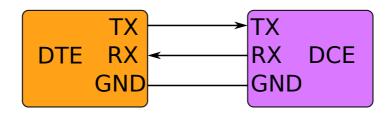
- Es un periférico de comunicación serial.
- Permite comunicar dos dispositivos sin el uso de un reloj.
 - Estos dispositivos deben "acordar" de antemano cual es la velocidad de transmisión.
 - Cada dispositivo deberá tener su reloj.
 - Los relojes no estarán sincronizados → comunicación asíncrona.
- Es utilizada en protocolos RS-232 y RS-485

RS-232

- RS-232 Recommended Standard 232
- Creado en 1960 por Electronic Industries Association (EIA) para teletipos.
- La mas importante revisión se realizo en 1969, RS-232-C
- Historia de Revisiones
 - EIA RS-232 (May 1960)
 - EIA RS-232-A (October 1963)
 - EIA RS-232-B (October 1965)
 - EIA RS-232-C (August 1969) ← Utilizado en las PC
 - EIA EIA-232-D (1986)
 - TIA TIA/EIA-232-E (1991)
 - TIA TIA/EIA-232-F (1997-10-01)
 - ANSI/TIA-232-F-1997 (R2002)
 - TIA TIA-232-F (R2012)

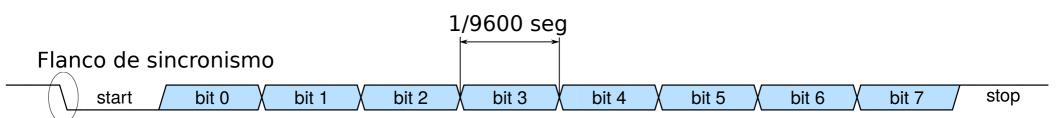
RS-232-C

- En su modo mínimo solo requiere de 3 lineas RX, TX y GND
- Originalmente la norma preveía la conexión entre dos equipos denominados.
 - Equipo Terminal de Dato o Data Terminal Equipment (DTE), este equipo era la computadora.
 - Equipo de Comunicación de Datos o Data Communication Equipment (DCE), este equipo era el MODEM.



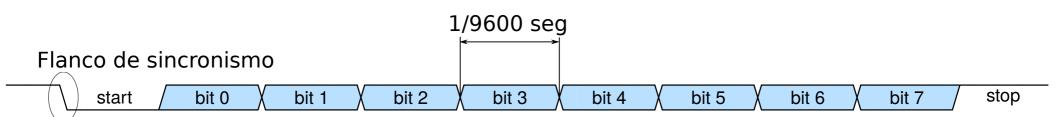
Trama

- Los datos son divididos en palabras de una longitud fija, generalmente octetos (orientada a carácter).
- Al comienzo se agrega un bit en 0 (bit de start), el flanco de bajada entre el 1 del canal desocupado a 0 del bit de start permite sincronizar al receptor.
- Al final se pueden agregar 1 o mas espacios para identificar un nuevo código de sincronismo (bit de stop).



Trama

- Entre el bit de start y el bit de stop, se envían los datos.
- La trama puede tener 7 bit (código ASCII) u 8 bits (byte).
- Adicionalmente y al final de la trama de dato, puede haber un bit de paridad par o impar.
- Finalmente el ejemplo de trama, muestra el envío de un dato de 8 bits sin paridad 1 bit de stop a una velocidad de transmisión de 9600 baudios.



Velocidad de Transmisión

- Baudio mide la cantidad de símbolos enviados por segundo.
- Bit rate o tasa de bit, es la velocidad de bit de datos enviados por segundo.
- En RS-232 cada símbolo puede tener 2 niveles, esto representa a 1 bit.
 - Otros medios de comunicación digital un símbolo puede tener 4 o mas niveles, representando 2 o mas bits
- Además se deben agregar el bit de start y bit de stop que no son datos.
- Se requiere entonces 10 símbolos para enviar 1 byte.

$$\frac{9600 \, simbolos/seg \, x \, 8 \, bits}{10 \, simbolos} = 7680 \, bits/seg = 960 \, bytes$$

Velocidad de Transmisión

La configuración de la velocidad es en baudios.

 Luego y dependiendo de la cantidad de bits de datos, el uso de paridad o no y los bits de stop se puede calcular la tasa

de bit.

La configuración mas común es de 9600 baudios.

- La configuración estándar mas rápida es 115200 baudios.
- Estas velocidades son ampliamente superadas por estándar como SPI o USB.

Velocidades Estandar RS232
300
1200
2400
9600
14400
19200
38400
57600
115200

Señales

Nombre	Descripción	Dirección	DB-25	DE-9
DTR (Data Terminal Ready)	DTE listo para transmitir, inicializar o continuar una llamada	PC → Modem	20	4
DCD (Data Carry Detect)	DCE está recibiendo una portadora	Modem → PC	8	1
DSR (Data Set Ready)	DCE listo para recibir y enviar datos	Modem → PC	6	6
RI (Ring Inicator)	DCE detectó una llamada entrante en la linea telefónica	Modem → PC	22	9
RTS (Request to Send)	DTE solicita a DCE transmitir datos	PC → Modem	4	7
RTR (Ready to Receive)	DTE listo para recibir datos	PC → Modem	4	7
CTS (Clear to Send)	DCE listo para aceptar datos		5	8
TxD (Transmitted Data)	Canal de envio de datos desde DTE	PC → Modem	2	3
RxD (Received Data)	Canal de envío de datos de DCE	Modem → PC	3	2
GND (Common Ground)	Referencia de masa		7	5
PG (Protective Ground)	Conectado al chasis		1	

Señales RTS y CTS

- Request to Send (RTS)
- Clear to Send (CTS)
- Estas señales utilizadas para *Hardware Handshaking* pueden ser funcionar de dos formas distintas.
- Modo Control de Flujo conexión full-duplex Lineas usadas para Hardware Handshaking o Control de Flujo, ambas lineas son independiente.
 - RTS → 0 (en DTE) indica listo para recibir
 - CTS → 0 (en DCE) indica listo para recibir
- Modo Simple pensado en el uso de antiguos modem, conexión half-duplex, CTS depende de RTS.
 - RTS → 0 (en DTE) indica listo para transmitir entonces CTS → 0 (en DCE) indica listo para recibir

Señales DTR y DSR

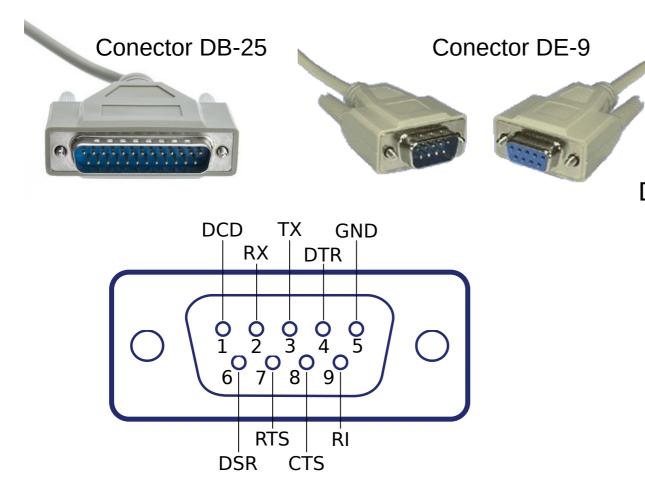
- Lineas utilizadas como Control de Flujo y para conectarse a un modem.
- Data Terminal Ready (DTR) DTE → DCE, terminal listo, el modem puede inicializar la comunicación.
 - Esta señal debe ser mantenida en encendida mientras se está comunicando, cuando se desconecta indica al modem que debe dejar de transmitir.
 - Otro uso es como señal de alimentación.
- Data Set Ready (DSR) DCE→ DTE, el modem está listo para recibir datos.

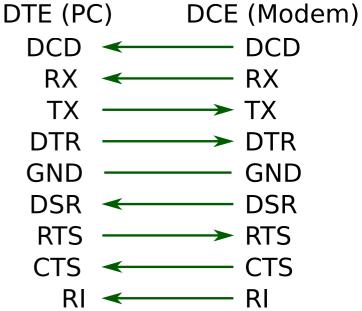
Señales DCD y RI

- Data Carrier Detect (DCD) DCE → DTE, el modem está conectado con otro modem.
- Ring Indicador (RI) DCE → DTE indica que el modem ha detectado la señal de llamada.

Conector

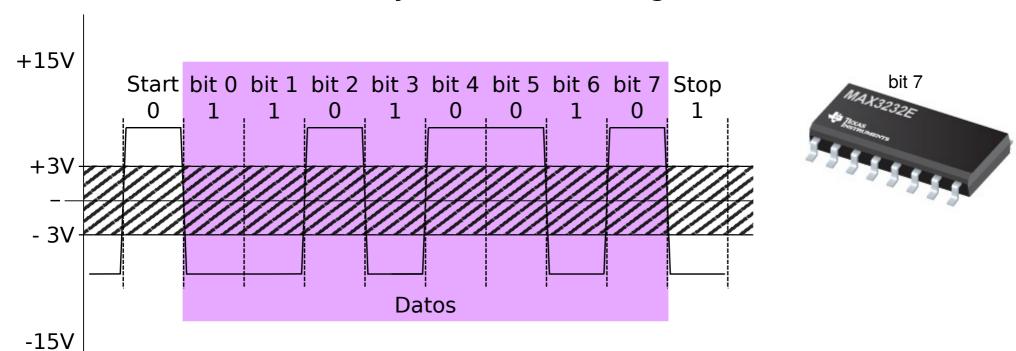
• En principio se utilizó el DB-25, actualmente hay diversos conectores utilizados, el mas común es el DE-9.





Niveles Eléctricos

- Los valores lógicos en RS-232 se representan con señales bipolares 0 para 3 a 15 V y 1 para -3 a -15V.
- Se requiere una interfaz que convierta las señales lógicas de la UART a niveles de RS-232.
- MAX3232E es un integrado que posee dos interfaces lógicas (3,3 o 5 V) a RS-232 y dos RS-232 a lógica.

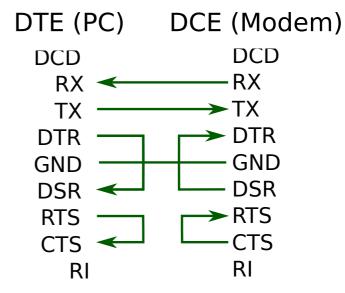


Cable

- No mas de 15 mtrs.
- Se pueden encontrar diferentes configuraciones.

Conexión de 9 cables normal

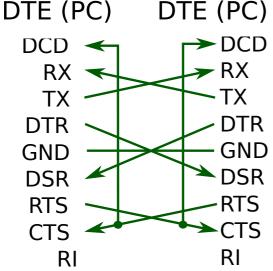
Conexión de 3 cables con control de flujo



Cable NULL-Modem

- Permite conectar dos DTE o PC entre sí.
- Su nombre proviene de la ausencia de un DCE o Modem.

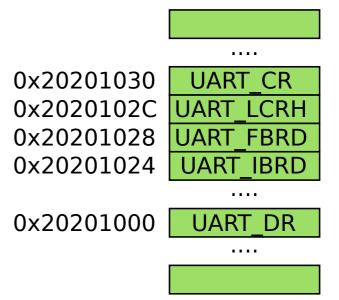
Conexión con control deflujo



Conexión de 3 cables DTE (PC) DCE (Modem) → DCD DCD < **→** RX RX -TX TX DTR DTR **GND GND** → DSR DSR < **RTS** RTS RIRI

UART en BCM2835

- Posee 2 UART denominadas UART0 y UART1.
- Ambas se comunican por los pines 14 y 15 de la GPIO.
- La UARTO posee mayor funcionalidad.
- Para esta UART los pines 14 y 15 deben ser configurado con GPFSEL en el modo ALTO.



Velocidad de Configuración

- La UART posee un reloj interno de 3 MHz.
- El reloj interno es dividido por una constante resultando en un valor 16 veces mas rapido que la velocidad deseada.

 $BDR = IBRD + \frac{FRBD}{64}$

Despejando la constante resulta

$$BDR = \frac{3000000}{16 \times baud\,rate}$$

- BDR está compuesto por
 - Parte entera de 16bits en UART_IBRD
 - La fracción en 6 bits UART_FRBD
- Ejemplo: 9600 baudios

$$BDR = \frac{3000000}{16 \times 9600} = 16,53125$$
 $IBDR = 16$
 $FBDR = 0,53125 \times 64 = 34$

Velocidad de Configuración

- Tabla con los valores de BDR (entero y fracción).
- El error introducido por el divisor en los baudios reales, se corrige por el sincronismo que se realiza al comenzar cada trama (si es los suficientemente pequeño).

Baudios	UART_IBRD	UART_FBRD	Baudios reales	Error (%)
300	625	0	300	0
1200	156	16	1200	0
2400	78	8	2400	0
9600	19	34	9600	0
19200	9	49	19200	0
38400	4	56	38461	0,16
57600	3	16	57692	0,16
115200	1	40	115384	0,16

Registro de Control de Linea UART_LCRH

- Configuración de nro de bits de datos, bits de stop y paridad.
- Se debe configurar el registro para 8 bit de datos, 1 bit de stop y sin paridad.

Bits	Nombre	Descripción	Tipo	Reset
7	SPS	0=paridad desactivada 1= depende de EPS	RO	0x0
6:5	WLEN	Bits de datos b11=8 bits b10=7 bits b01=6 bits b00=5 bits	RW	0x0
4	FEN	Habilita FIFO 1=FIFO activa	RW	0x0
3	STP2	1=2 bit de stop 0=1 o 1,5 bit de stop	RW	0x0
2	EPS	Tipo de paridad 0 = paridad impar 1 = paridad par	RW	0x0
1	PEN	Paridad habilitada 0 = sin paridad 1 = con paridad	RW	0x0
0	BRK	BRK = 1 bajo nivel se transmite por RxT luego de completar la transición en curso	RW	0x0

Registro de Control o UART_CR

- Esta configuración se realiza en el Registro de Control de Linea UART CR
- Se debe habilitar la UART (UARTEN = 1)

Bits	Nombre	Descripción	Tipo	Reset
15	CTSEN	Activa el control de flujo por CTS	RW	0x0
14	RTSEN	Activa el control de flujo por RTS	RW	0x0
11	RTS	Permite manejar la linea RTS, su valor es negado	RW	0x0
10	DTR	No soportado	RO	0x0
9	RXE	Habilita la recepción	RW	0x0
8	TXE	Habilita la transmisión	RW	0x0
7	LBE	Cuando está activa permite hacer un loop (une TX con RX)	RW	0x0
0	UARTEN	Habilita la UART	RW	0x0

Registro de Dato o UART_DR

- Principalmente utilizado para la escritura del byte a enviar o leer el dato recibido.
- Posee además banderas que identifican alguna condición de error del último dato recibido.

Bits	Nombre	Descripción	Tipo	Reset
11	OE	OE=1 se sobrescribió un dato recibido	RW	0x0
10	BE	BE=1 ocurrió un break, por mas de una trama completa el canal permaneció en bajo	RW	0x0
9	PE	PE=1 Error de paridad	RW	0x0
8	FE	FE=1 Error de trama, cuando por ejemplo falta el bit de stop.	RO	0x0
7:0	DATA	Lectura del registro = byte recibido Escritura en el registro = byte a enviar	RW	0x0

Registro de Banderas o UART_FR

Bits	Nombre	Descripción	Tipo	Reset
7	TXFE	TXFE = 1 FIFO de transmisión vacio o en caso de tener la FIFO desactivada cuando el registro de transmisión esta lleno (no indica que no esté transmitiendo)	RW	0x0
6	RXFF	RXFF = 1 FIFO de recepción lleno o en caso de tener la FIFO desactivada cuando el registro de envío está lleno	RW	0x0
5	TXFF	TXFF = 1 FIFO de transmisión lleno o en caso de tener la FIFO desactivada cuando el registro de transmisión está lleno	RW	0x0
4	RXFE	RXFE = 1 FIFO de recepción vacio o en caso de tener la FIFO desactivada cuando el registro de envío está vacio	RO	0x0
3	BUSY	La UART está ocupada enviando datos	RW	0x0
0	CTS	Clear to Send este bit es el complemento de la entrada CTS desde el modem	RW	0x0

- La función de inicializar configura los pines de E/S del UARTO como ALTO (modo UART).
- Configura IBRD y FRDB según los baudios enviados por parámetros con baud.
- De LCR solo modifica e tamaño del dato (8 bits).
- Finalmente habilita la UART.

- La función de envío y recepción de datos por la UART son del tipo bloqueantes (no salen hasta haber enviado o recibido un dato).
- Se verifica de FR el bit RXFE → registro de recepción vacío.
- Se verifica de FR el bit TXFE → registro de envío vació.
- En ambos casos para el valor leído o a enviar se recurre a DR

```
#include "EasyPIO.h"
#define MAX STR LEN 80
void getStrSerial(char *str) { // recibe una cadena
  int i = 0;
  do {
                           // recibe toda la cadena hasta un
     str[i] = getCharSerial(); // retorno de carro '\r' o MAX_STR_LEN
  } while ((str[i++]!='\r') \&\& (i<MAX STR LEN));
  str[i-1] = 0; // agrega terminación nula
void putStrSerial(char *str) { // envía una cadena
  int i = 0;
  while (str[i] ! = 0) { // lee desde la cadena y envía hasta
     putCharSerial(str[i++]); // valor nulo
```

```
int main(void) {
   char str[MAX_STR_LEN];
  pioInit();
  UartInit(115200); // inicaliza UART con la velocidad de baudios
   while (1) {
      putStrSerial("Please type something: \r\n");
      getStrSerial(str);
      putStrSerial("You typed: ");
      putStrSerial(str);
      putStrSerial("\r\n");
```

Bibliografía

Harris & Harris. Digital design and computer architecture: ARM edition. Elsevier, 2015. Capítulo 9.

BCM2835-ARM-Peripherals.pdf

¿ Preguntas ?