

Comunicación Raspberry Pi con controladora de robot Bioloid CM-530

La forma de comunicarse con la CM-530 es un poco especial, ya que para enviar usa un método muy normal mientras que para recibir datos, el puerto USB está en paralelo con el receptor del mando a distancia, y usa la misma forma de comunicación que el mando a distancia.

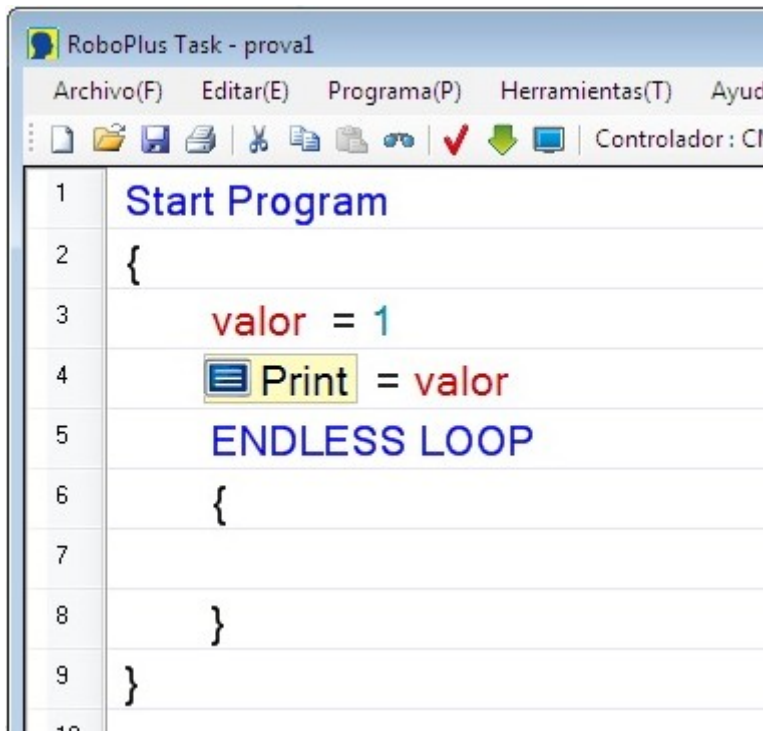
Los parámetros de la comunicación serie son: 57600, 8 bits de datos, sin paridad y 1 bit de stop.

Envío de datos desde el Bioloid a la Raspberry:

Enviar datos de la CM-530 a la Raspberry es muy sencillo, ya que usamos la función de envío al terminal "Print", la que se usa para ver en pantalla variables del estado del robot.

De este modo podemos enviar números, por ejemplo:

Para decirle a la Raspberry que ya puede leer un código Qr enviamos un «1»



Al arrancar el programa del Bioloid este envía por el USB unos mensajes como estos

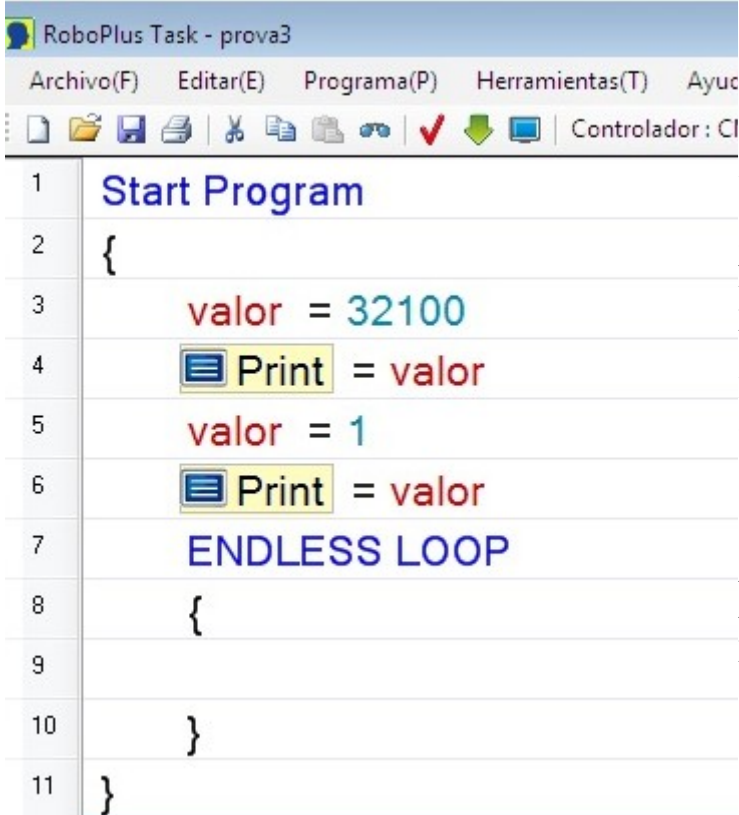
```
[[CM-530:1.1982]]  
[[PC:57142(BPS)][DXL:1000000(BPS)]]  
{  
  
}  
[[DXL:000(PCS)]]
```

Comunicación Raspberry Pi con controladora de robot Bioloid CM-530

Estos mensajes se recibirán en la Raspberry Pi y no nos interesan.

La forma más sencilla de evitar este problema es ejecutar el programa de la Raspberry después de arrancar el Bioloid.

Otra forma más elaborada, pero mucho mejor, es no hacer caso a lo que llegue por el puerto hasta que aparezca un mensaje. Por ejemplo podemos hacer que la Raspberry no haga caso de los caracteres recibidos hasta que no reciba el número 32100.



```

1 Start Program
2 {
3     valor = 32100
4     Print = valor
5     valor = 1
6     Print = valor
7     ENDLESS LOOP
8     {
9
10    }
11 }
    
```

La Raspberry, tras recibir esos cinco caracteres seguidos se pone en modo de aceptar ya lo que le llegue. Esto no lo hace la Raspberry porque si, sino que el programa que nosotros realizamos se encarga de detectar esa secuencia. De este modo, el siguiente 1 ya sería aceptado por el software de la Raspberry como una orden.

Aquí vemos lo que se ha recibido en ASCII en la Raspberry al ejecutar este programa en el Bioloid.

```

21 21 08 08 21 21 08 08 0D 0A 0A 20 7B
5B 43 4D 2D 35 33 30 3A 31 2E 31 39 38
32 5D 7D 0D 0A 20 7B 5B 50 43 3A 35
37 31 34 32 28 42 50 53 29 5D 5B 44 58
4C 3A 31 30 30 30 30 30 28 42 50 53 29 5D 7D 0D 0A 20 7B 0D 0A 0D 0A 20 7D 0D 0A 20 7B
5B 44 58 4C 3A 30 30 30 28 50 43 53 29 5D 7D 0D 0A 33 32 31 30 30 20 20 20 20 31
    
```

La parte en negrita es lo que recibimos al enviar el 32100.

La parte en rojo es lo que recibimos al enviar un 1.

El caracter 20 es el código ASCII del espacio en blanco, el 33 es el código ASCII del 3 , el 32 del 2, el 31 del 1 y el 30 del 0

La cadena que deberíamos comprobar, escrita ya con caracteres sería: «32100»

Y al recibir el 1 recibiremos 4 espacios en blanco y el 1.

(Los números se reciben como si tuviesen 5 cifras).

Comunicación Raspberry Pi con controladora de robot Bioloid CM-530

Envío de datos desde la Raspberry al Bioloid

Para enviar datos al Bioloid es más extraño.

Si queremos enviar al robot el byte 07, el mensaje que hay que enviar desde la Raspberry es:

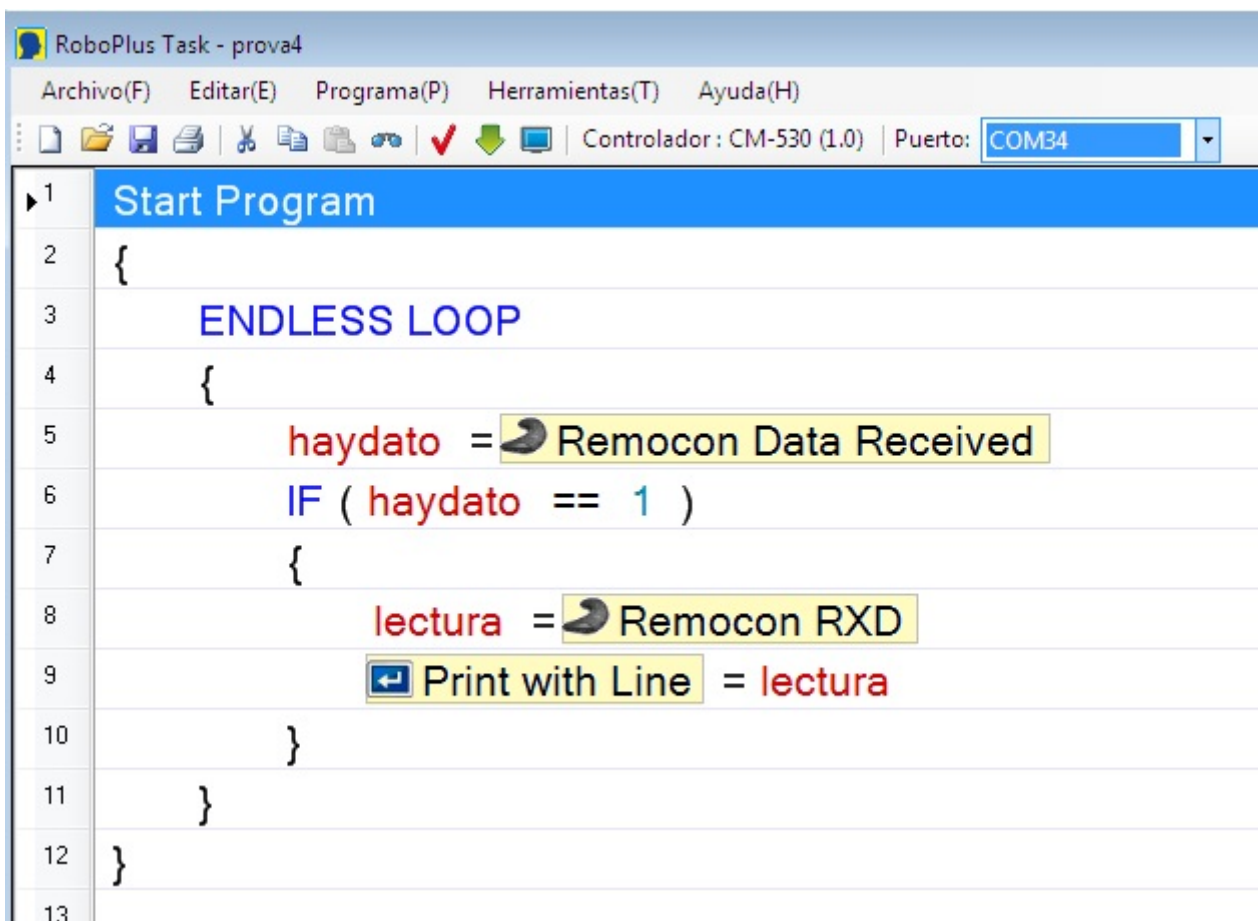
FF 55 07 F8 00 FF

Los primeros 2 bytes y los dos últimos son fijos.

El tercer byte es el que queremos enviar

El cuarto byte es el complemento a 1 del byte que queremos enviar.

Para recibir este byte en el Bioloid debemos mirar si hay algún byte recibido por el mando a distancia y si es así, leerlo. Al leerlo nos devolverá solo el byte 07.



```
1 Start Program
2 {
3     ENDLESS LOOP
4     {
5         haydato = Remocon Data Received
6         IF ( haydato == 1 )
7         {
8             lectura = Remocon RXD
9             Print with Line = lectura
10        }
11    }
12 }
13
```

Este programa está esperando que llegue algún dato al Bioloid. Aunque parezca que recibe órdenes del mando a distancia, también recibe lo que le llega por el USB.

Comunicación Raspberry Pi con controladora de robot Bioloid CM-530

Software para realizar pruebas.

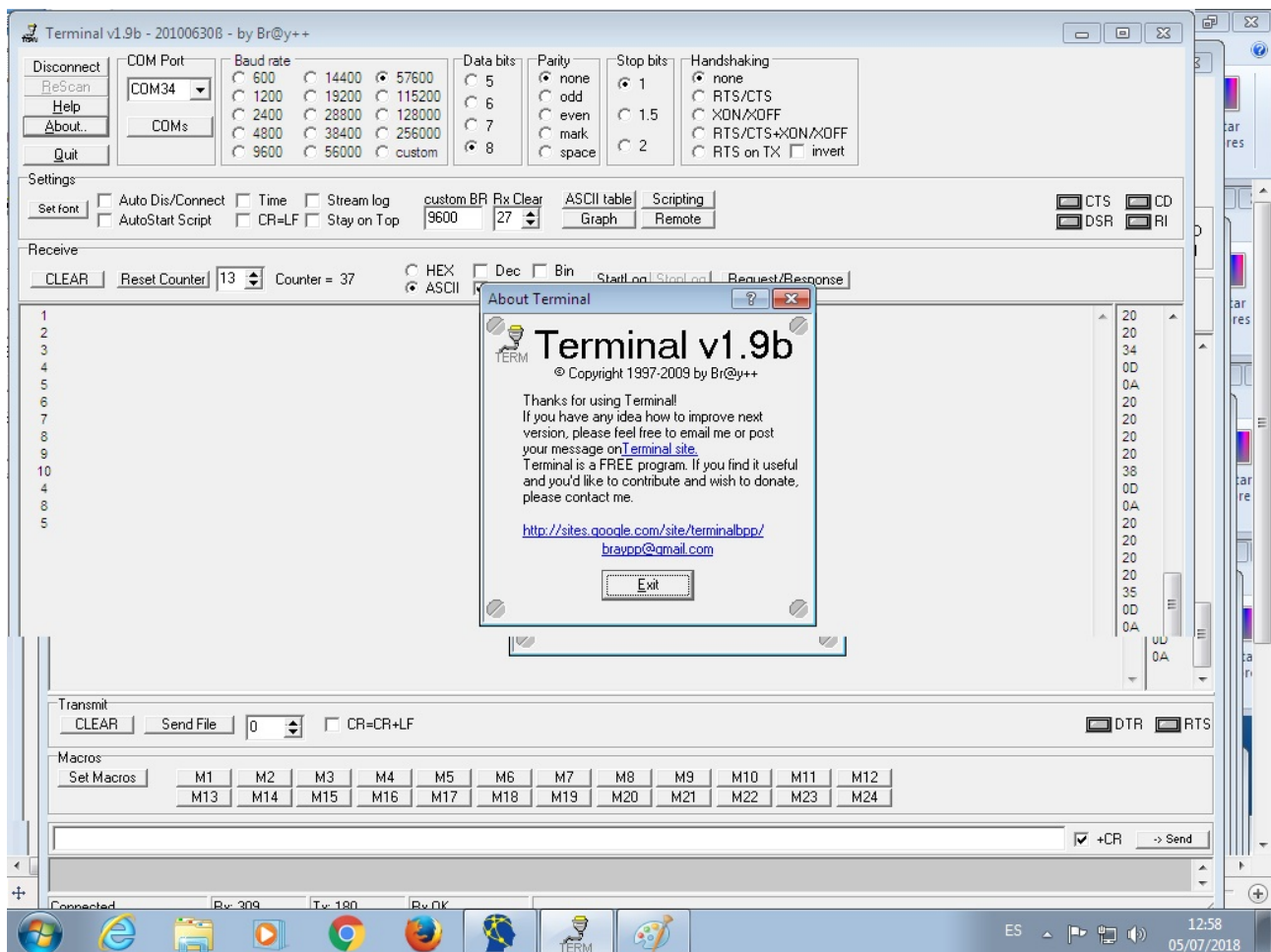
Aunque la comunicación se va a hacer desde la Raspberry, también podemos utilizar para hacer pruebas un PC en Linux o en Windows.

En Linux podemos usar el programa **gtkterm**, y desde Windows podemos usar el programa **Terminal**. Este programa se puede descargar desde esta página:

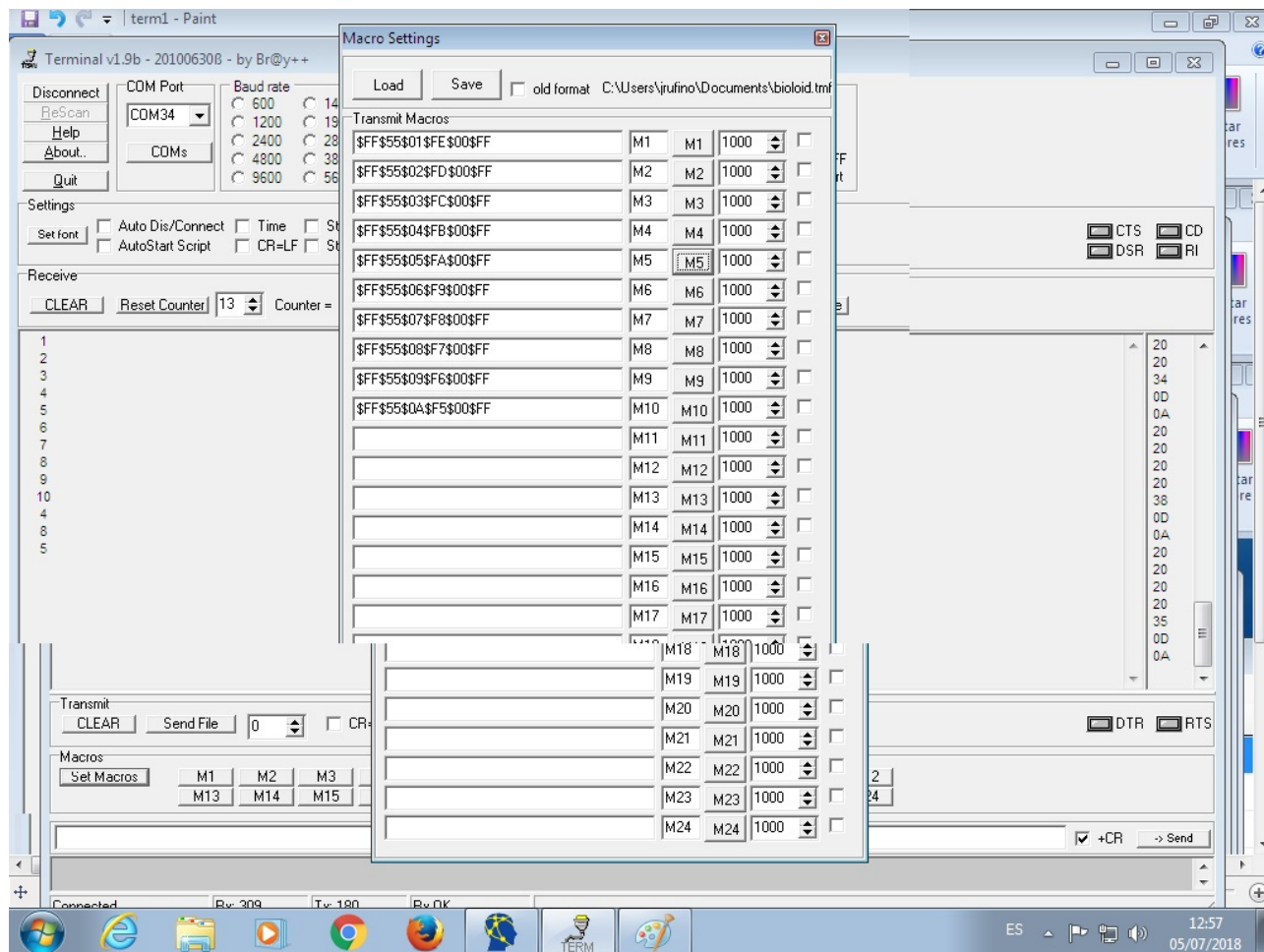
<https://sites.google.com/site/terminalbpp/>

Este programa es gratuito y es muy cómodo para realizar pruebas.

A continuación podemos ver unas capturas de pantalla del programa terminal en el que podemos ver como se configuran las macros para poder enviar los 6 bytes de cada mensaje.



Comunicación Raspberry Pi con controladora de robot Bioloid CM-530



Cualquier comentario o duda podéis comunicarla a jrufino@eln.upv.es