

# *Documentación FRC*

*Autores:*

- *Samuel Moreno Vincent*
- *Santiago Rangel Colón*

*09/05/2017*

# Documentación FRC

## Contenido

Introducción .....	3
Manual del Usuario.....	3
F1 .....	4
F2 .....	4
F3 .....	5
F4 .....	6
Maestro Selección.....	6
Esclavo Selección .....	6
Maestro Sondeo.....	7
Esclavo Sondeo .....	7
Maestro Errores .....	8
Esclavo Errores.....	8
Escape .....	9
Manual del Programador .....	9
Jerarquía de clases.....	9
Algoritmos importantes .....	10
Batería de Pruebas.....	13
Caracteres de control utilizados: .....	15
Opciones adicionales .....	16
Problemas encontrados al realizar la práctica .....	16
Conclusiones:.....	16

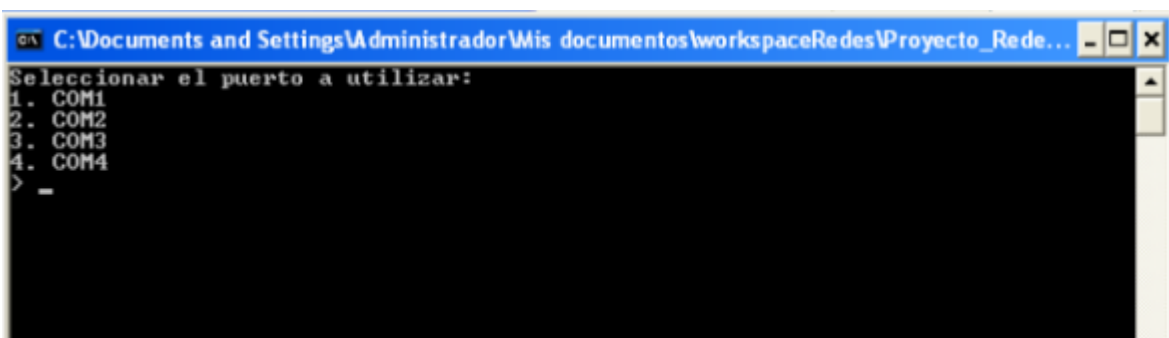
## Introducción

El proyecto de la asignatura consiste en una aplicación que nos permitirá comunicar nuestro equipo con otro que se encuentre conectado al nuestro por un cable serial, esta comunicación se realizará en los dos sentidos, siendo nosotros los que enviaremos a nuestro compañero pero también podremos recibir.

La aplicación cuenta con varias funciones en las que ya entraré en más detalle a continuación en el manual de usuario pero estas pueden dividirse en dos grupos, envío y recepción de mensajes usando tramas de datos, y el envío de ficheros.

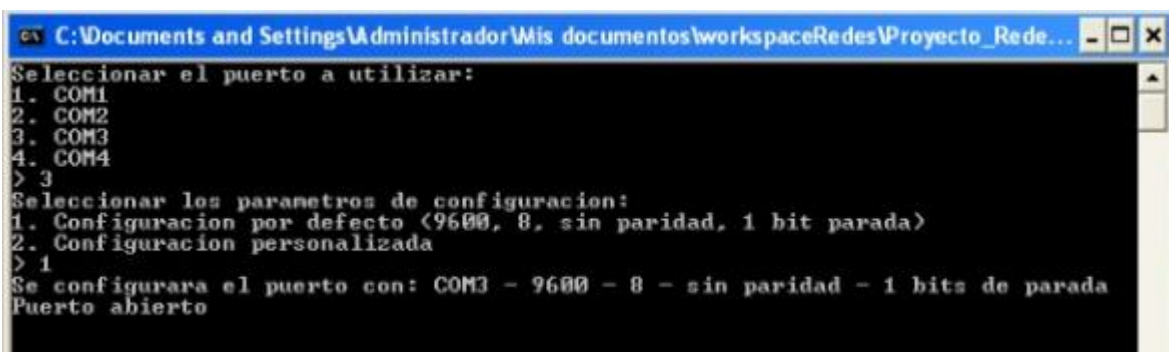
## Manual del Usuario

Interfaz de la aplicación Para ejecutar la aplicación simplemente ejecutaremos el archivo Proyecto\_Red.es.exe que se encuentra dentro de la carpeta reléase dentro de la carpeta del proyecto, una vez ejecutado podremos ver la siguiente ventana.



Dentro de esta interfaz podremos ejecutar las funciones de nuestra aplicación pulsando las teclas de las funciones, para poder ejecutar correctamente la aplicación será necesario que el otro equipo también la esté ejecutando para así poder realizar la comunicación correctamente, las funciones que podremos ejecutar están detalladas a continuación.

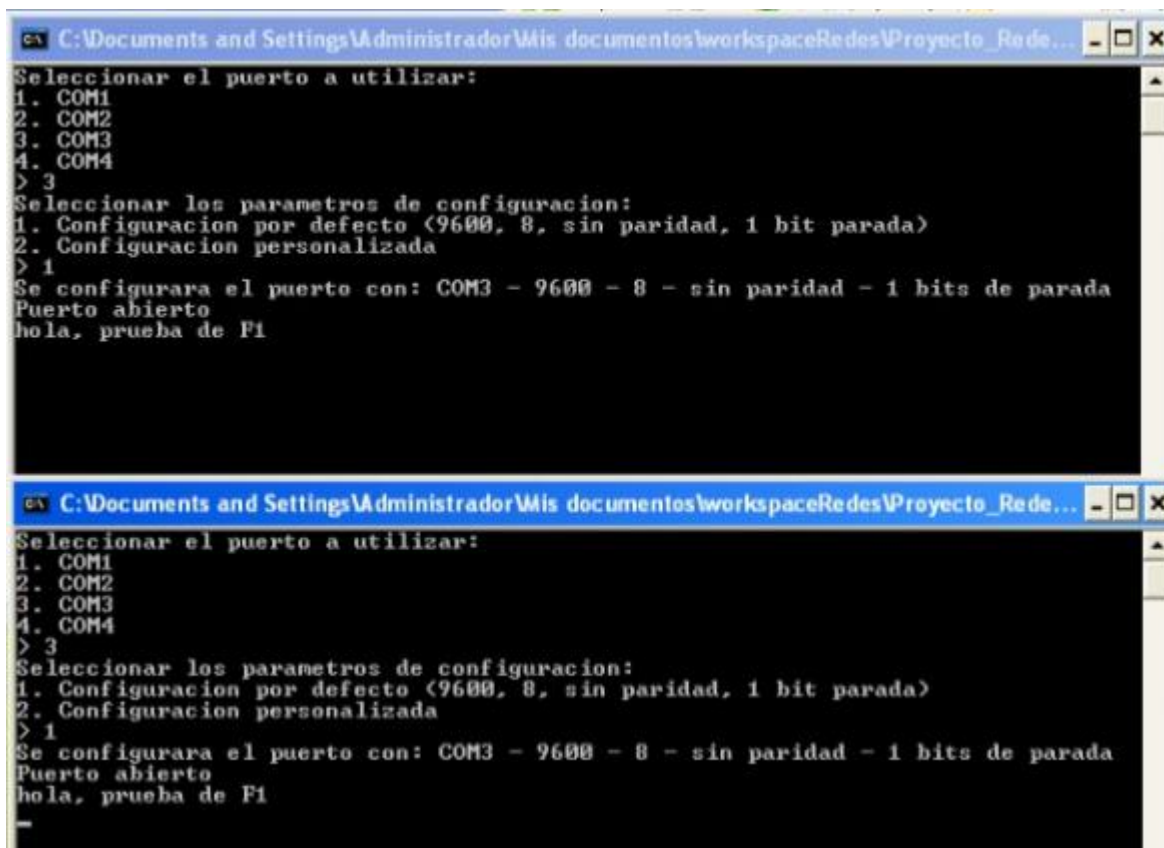
Pero para realizar esto primero tendremos que elegir que puerto queremos utilizar, tras esto nos saldrá a elegir un menú:



La primera opción nos configurará el puerto con la configuración que podemos ver en pantalla por defecto y la segunda opción nos permitirá ir configurando manualmente el puerto opción por opción, tras esto ya podremos empezar a realizar las funcionalidades de nuestra aplicación, F1, F2, F3 y F4.

### F1

Una vez abierta la aplicación podremos escribir de forma normal, incluido podremos borrar lo que hemos escrito o introducir un salto de línea, podremos realizar el envío de todo lo que tenemos escrito en la ventana pulsando la primera tecla de función de nuestra aplicación, “F1”, esta tecla enviará hasta un total de 800 caracteres a nuestro compañero.



Al pulsar la tecla de función “F1” el mensaje que teníamos por pantalla se enviará a nuestro compañero y este mensaje le aparecerá en la pantalla.

### F2

Al pulsar la tecla de función “F2” nos aparecerá un menú, en el cual podremos elegir un tipo de trama de control para enviársela automáticamente al receptor. Esta elección la realizaremos pulsando las teclas que nos indique para elegir que trama de control queremos enviar.

Al enviar la trama, en el equipo de nuestro compañero le aparecerá en la pantalla que ha recibido una trama, y nos aparecerá indicado con un mensaje que tipo de trama nos ha llegado.

A continuación se puede observar esto en la captura de pantalla.

```

C:\Documents and Settings\Administrador\Mis documentos\workspaceRedes\Proyecto_Rede...
Seleccionar el puerto a utilizar:
1. COM1
2. COM2
3. COM3
4. COM4
> 3
Seleccionar los parametros de configuracion:
1. Configuracion por defecto (9600, 8, sin paridad, 1 bit parada)
2. Configuracion personalizada
> 1
Se configurara el puerto con: COM3 - 9600 - 8 - sin paridad - 1 bits de parada
Puerto abierto
Trama de control a enviar:
1: Trama ENQ
2: Trama EOI
3: Trama ACK
4: Trama NACK
Tecla 1 Pulsada
Creada trama control tipo ENQ
->Trama de control enviada

```

### F3

Al pulsar la tecla de función comenzará a enviar el fichero llamado “Frc-E.txt” que se encuentre en la misma carpeta donde se encuentra el ejecutable. Si no existe el fichero nos dará un mensaje de error, es posible enviar el fichero aunque este este vacío, pero debe de existir para evitar el mensaje de error. En la recepción nos aparecerá un mensaje de recepción de archivo y se guardara en el archivo llamado “Frc-R.txt” que se encontrara en la carpeta donde se encentra el ejecutable. A continuación se puede observar esto en las imágenes de la ejecución:

```

C:\Documents and Settings\Administrador\Mis documentos\workspaceRedes\Proyecto_Rede...
Seleccionar el puerto a utilizar:
1. COM1
2. COM2
3. COM3
4. COM4
> 3
Seleccionar los parametros de configuracion:
1. Configuracion por defecto (9600, 8, sin paridad, 1 bit parada)
2. Configuracion personalizada
> 1
Se configurara el puerto con: COM3 - 9600 - 8 - sin paridad - 1 bits de parada
Puerto abierto
Enviando fichero...
Enviado fichero correctamente

```

```

C:\Documents and Settings\Administrador\Mis documentos\workspaceRedes\Proyecto_Rede...
Seleccionar el puerto a utilizar:
1. COM1
2. COM2
3. COM3
4. COM4
> 3
Seleccionar los parametros de configuracion:
1. Configuracion por defecto (9600, 8, sin paridad, 1 bit parada)
2. Configuracion personalizada
> 1
Se configurara el puerto con: COM3 - 9600 - 8 - sin paridad - 1 bits de parada
Puerto abierto
Recibiendo fichero...
Fichero recibido.

```

Es posible realizar envíos de tramas de hasta 800 de tamaño mientras que se está recibiendo un fichero.

#### F4

Al pulsar la tecla de función “F4” nos aparecerá un menú que nos podría seleccionar si se quiere ser maestro o esclavo. Si la elección es “maestro” esto nos dará otra nueva selección que nos permitirá decidir si el envío se realizara por sondeo o por selección.

#### Maestro Selección

```
Elija una de las posibles opciones:
1: Estacion Maestra
2: Estacion esclava
Elegida estacion como maestra
Elija una de las posibles opciones:
1: Seleccion
2: Sondeo
Elegido envio por seleccion

E: R ENQ 0
R: R ACK 0

E: R STX 0 39
R: R ACK 0
E: R STX 1 90
R: R ACK 1
E: R STX 0 61
R: R ACK 0
E: R STX 1 6
R: R ACK 1
E: R STX 0 94
R: R ACK 0
E: R STX 1 40
R: R ACK 1
E: R STX 0 7
R: R ACK 0
E: R STX 1 25
R: R ACK 1
E: R STX 0 93
R: R ACK 0

E: R EOT 0
R: R ACK 0
```

#### Esclavo Selección

```
Elija una de las posibles opciones:
1: Estacion Maestra
2: Estacion esclava
Elegida estacion como esclava

R: R ENQ 0
E: R ACK 0

R: R STX 0 39 39
E: R ACK 0
R: R STX 1 90 90
E: R ACK 1
R: R STX 0 61 61
E: R ACK 0
R: R STX 1 6 6
E: R ACK 1
R: R STX 0 94 94
E: R ACK 0
R: R STX 1 40 40
E: R ACK 1
R: R STX 0 7 7
E: R ACK 0
R: R STX 1 25 25
E: R ACK 1
R: R STX 0 93 93
E: R ACK 0

R: R EOT 0
E: R ACK 0
```

### Maestro Sondeo

```

Elegida estacion como maestra
Elija una de las posibles opciones:
1: Selecccion
2: Sondeo
Elegido envio por sondeo

E: I ENQ 0
R: I ACK 0

R: I STX 0 39 39
E: I ACK 0
R: I STX 1 90 90
E: I ACK 1
R: I STX 0 61 61
E: I ACK 0
R: I STX 1 6 6
E: I ACK 1
R: I STX 0 94 94
E: I ACK 0
R: I STX 1 48 48
E: I ACK 1
R: I STX 0 7 7
E: I ACK 0
R: I STX 1 25 25
E: I ACK 1
R: I STX 0 93 93
E: I ACK 0

R: I EOT 0
Desea terminar la transmision? 1: Si, 2: No
E: I NACK 0
R: I EOT 1
Desea terminar la transmision? 1: Si, 2: No
E: I NACK 1
R: I EOT 0
Desea terminar la transmision? 1: Si, 2: No
E: I ACK 0

```

### Esclavo Sondeo

```

Elija una de las posibles opciones:
1: Estacion Maestra
2: Estacion esclava
Elegida estacion como esclava

R: I ENQ 0
E: I ACK 0

E: I STX 0 39
R: I ACK 0
E: I STX 1 90
R: I ACK 1
E: I STX 0 61
R: I ACK 0
E: I STX 1 6
R: I ACK 1
E: I STX 0 94
R: I ACK 0
E: I STX 1 48
R: I ACK 1
E: I STX 0 7
R: I ACK 0
E: I STX 1 25
R: I ACK 1
E: I STX 0 93
R: I ACK 0

E: I EOT 0
R: I NACK 0
E: I EOT 1
R: I NACK 1
E: I EOT 0
R: I ACK 0

```

Durante el envío del fichero podremos pulsar la tecla de función “F5” introduciendo un error en la próxima trama que se enviará. A continuación se puede observar esto en las imágenes de la ejecución.

### Maestro Errores

```

E: R STX 0 61
R: R ACK 0
Error introducido
E: R STX 1 6
R: R NACK 1
E: R STX 1 6
R: R ACK 1
Error introducido
E: R STX 0 94
R: R NACK 0
E: R STX 0 94
R: R ACK 0
Error introducido
E: R STX 1 48
R: R NACK 1
E: R STX 1 48
FS Pulsado
R: R ACK 1
Error introducido
E: R STX 0 7
R: R NACK 0
E: R STX 0 7
R: R ACK 0
Error introducido
E: R STX 1 25
R: R NACK 1
E: R STX 1 25
R: R ACK 1
E: R STX 0 93
R: R ACK 0

E: R EOT 0
R: R ACK 0

```

### Esclavo Errores

```

R: R STX 0 39 39
E: R ACK 0
R: R STX 1 90 210
BCE recibido: 90, BCE calculado: 210
E: R NACK 1
R: R STX 1 90 90
E: R ACK 1
R: R STX 0 61 180
BCE recibido: 61, BCE calculado: 180
E: R NACK 0
R: R STX 0 61 61
E: R ACK 0
R: R STX 1 6 140
BCE recibido: 6, BCE calculado: 140
E: R NACK 1
R: R STX 1 6 6
E: R ACK 1
R: R STX 0 94 216
BCE recibido: 94, BCE calculado: 216
E: R NACK 0
R: R STX 0 94 94
E: R ACK 0
R: R STX 1 48 176
BCE recibido: 48, BCE calculado: 176
E: R NACK 1
R: R STX 1 48 48
E: R ACK 1
R: R STX 0 7 140
BCE recibido: 7, BCE calculado: 140
E: R NACK 0
R: R STX 0 7 7
E: R ACK 0
R: R STX 1 25 171
BCE recibido: 25, BCE calculado: 171
E: R NACK 1
R: R STX 1 25 25
E: R ACK 1
R: R STX 0 93 93
E: R ACK 0

R: R EOT 0
E: R ACK 0

```



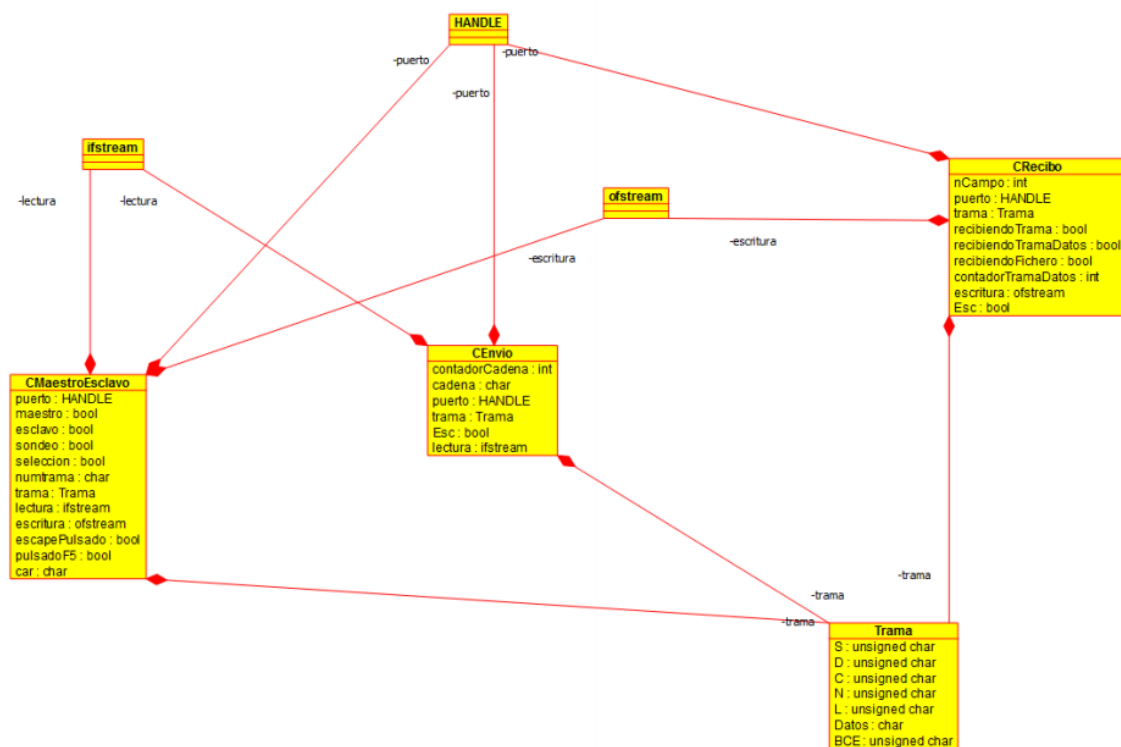
En el momento que se produce un error al pulsar la tecla “F5” aparecerá en pantalla el mensaje y cuando esta trama que tiene un error llega al receptor este mostrará que se ha producido y los BCE calculado y recibido y mandará una trama NACK para que el emisor del mensaje lo vuelva a mandar.

## Escape

Esta tecla terminará con la ejecución de la aplicación o se saldrá de un método en específico, mediante su pulsación repetida se asegura la salida de la ejecución.

## Manual del Programador

### Jerarquía de clases



La práctica se divide en el main más 4 clases y la librería del puerto serie. En el main únicamente se realiza para inicializar las clases y abrir los puertos, luego se realiza una llamada en un buque while a recibir y enviar.

La clase recibir se encarga de toda la parte de recibir datos tanto de F1, F2 y F3.

La clase envío se encarga de toda la parte de enviar datos en tramas, tanto de F1, F2 y F3. Además se encarga de guardar la información tecleada por teclado y de llamar a la clase Maestro/Esclavo si se pulsa F5. La clase trama contiene toda la información necesaria de las tramas, así como todos los métodos necesarios para editarlas, mostrarlas, enviarlas, etc.

La clase Maestro/Esclavo se encarga de toda la gestión de la funcionalidad F4, se llama desde la clase envío y se borra al terminar la cada comunicación.

### Algoritmos importantes

#### CEnvio::F1()

```

    inc=0;

    Mientras (Cadena > 255) {
        for (i= 0 hasta 255)
            cadenaAux[i] <- Cadena[i+inc];

        crearTramaDatos (cadenaAux);

        enviarTramaDatos for(255, L);

        Cadena = Cadena - 255;

        inc = inc + 255;
    }

    for (i= 0 hasta Cadena)
        cadenaAux[i] <- Cadena[i+inc];

    crearTramaDatos (cadena);

    enviarTramaDatos;

    Cadena = 0;

```

#### Void CMaestroEsclavo::esclavoSelecccon(){

//Establecimiento de la conexión

```

    numTrama = ObtenerNumeroTrama();

    trama.crearACK('R', numTrama); //Cambia los valores de la trama para que sean una ACK de
    selección y el número de trama obtenido

    trama.enviarTramaC(puerto); //Manda la trama de control al puerto

    trama.ImprimirEnvio();

```

//Transferencia

```

    enviarArchivo('T'); // Manda el fichero usando Tramas de Selección

```

//Liberacion

```

numTrama= '0';

recibirTrama();

if(trama.esEOT){

    numTrama = ObtenerNumeroTrama();

    trama.crearACK('R', numTrama);

    trama.enviarTramaC(puerto); //Manda la trama de control al puerto

    trama.ImprimirEnvio();

}

else

    cout <<"Error, la trama recibida no es EOT" << endl;

}

Void cMaestroEsclavo::enviarArchivo (char ToR){ //ToR carácter que indica si es sondeo o selección

    numTrama='0';

    lectura.open("frcE.txt");

    Si (!lectura.estaAbierto()){

        cout << "ERROR AL ABRIR ARCHIVO" << endl;

        enviarCar('#');

        return;

    }

    enviarCar('#');

    bool finEnvio = false;

    Mientras (!finEnvio && !escape()){

//Envio

        Si(!lectura.eof()){

            Char Linea[255];

            Lectura.read(línea,255);

            Aux = lectura.gcount();

```

```

Si(aux > 0){
    Int aux = lectura.gcount();

    Línea[aux] = '\0';

    Trama.crearTramaDCompleta(ToR, numtrama, strlen(Línea), Línea); //Forma
    la trama a enviar siendo de selección o sondeo según el parámetro ToR

    Si (pulsarF5()){ // Detecta si se ha pulsado F5 para introducir el error

        Trama.setDato('ç', 0);

        cout<< "Error introducido" <<endl;

        pulsadoF5 =false;

    }

    Trama.enviarTramaD(puerto);

    Trama.imprimirEnvio();

//Confirmacion

    recibirTrama();

    Si (trama.esNACK()){ //Si le llega una NACK vuelve a formar la trama y la
    envía

        trama.crearTramaDCompleta(ToR, numtrama, strlen(Línea), Línea);

        trama.enviarTramaD(puerto);

        trama.imprimirEnvio();

        recibirTrama();

        if (!trama.esACK){

            Cout << "la trama recibida no es ACK << endl;

            Return;

        }

    }

    Else if (!trama.esACK){

        Cout << "la trama recibida no es ACK << endl;

        finEnvio = true;;

```

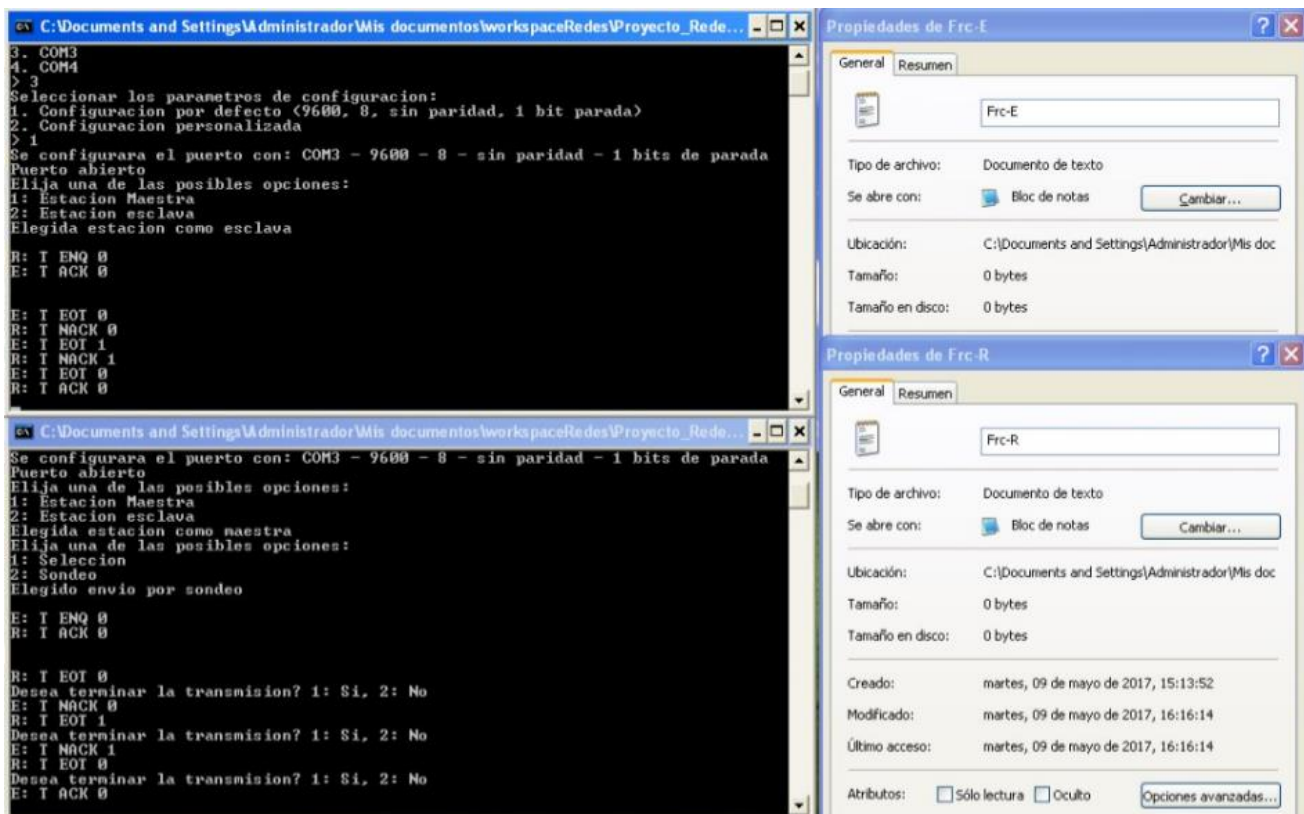
```

        }
    }
    Si (lectura.eof()){
        Lectura.close();
        finEnvio = true;
        enviarCaracter(puerto, '@');
    }
    Si no {
        cambiarNumTrama();
    }
}
}
}

```

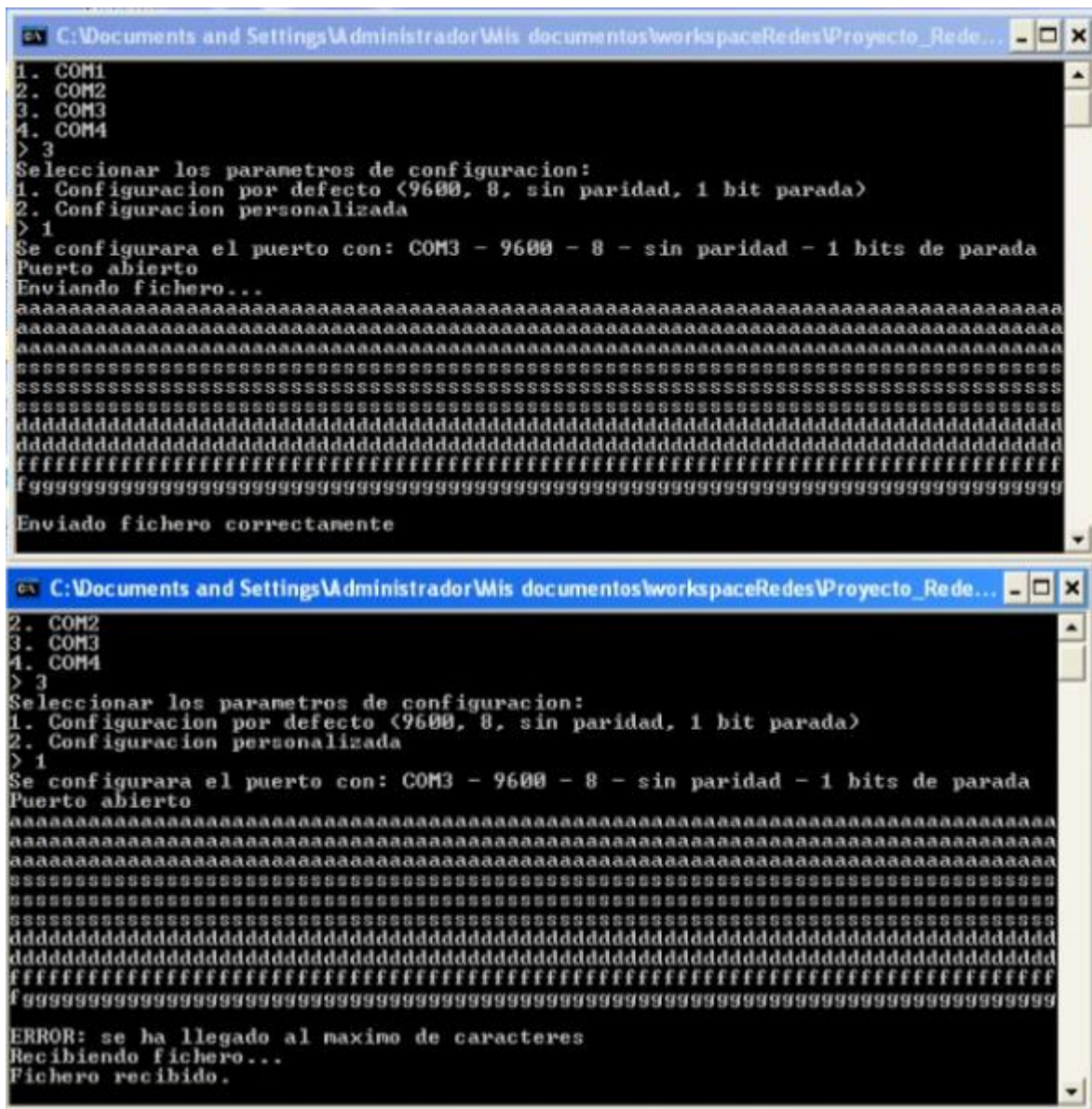
### Batería de Pruebas

Se han probado con muchos y distintos ficheros, aquí comentaremos cuatro de las pruebas que se han realizado, la primera consistió en el envío de un fichero de 0 kB con la tecla F4 que se envió sin problemas teniendo el mismo peso, usando selección.



Tras esto realicé una prueba de todas y cada una de las funcionalidades por separado probando a realizar las funciones tanto antes como después de realizar el protocolo maestro esclavo para comprobar que tras realizarlo no impedía que tras el la aplicación pudiera seguir realizando el resto de funciones y todo funcionó de forma correcta y pudimos ejecutar todas las funciones que se nos pedía.

La siguiente prueba fue para comprobar que efectivamente podemos enviar una trama de tamaño máximo mientras enviamos un fichero de gran tamaño 524Kb mediante F3, y como se puede ver a continuación se pudo realizar durante el envío.



Otra prueba fue comprobar que efectivamente se podía salir de todas las funciones mediante la tecla `esc`.

La última prueba consistió en el envío de un fichero grande, 435kB también con el protocolo de maestro esclavo de la tecla f4, el envío y la recepción se completó exitosamente aun con los fallos que se introdujeron durante su envío, tras completar el envío se probó a enviar mensajes con el F1 y a enviar tramas de control con el F2 y funcionaban correctamente.

### Caracteres de control utilizados:

Para la realización de esta práctica solamente utilizamos los caracteres de control que usamos para abrir el fichero y cerrarlo:

- #: Para abrir el fichero.
- @: Para cerrar el fichero.

### Opciones adicionales

Ninguna.

### Problemas encontrados al realizar la práctica

El mayor problema a la hora de realizar las entregas ha sido que no hacíamos la confirmación de que la trama que nos llegaba era o no la que debía de ser y al haber arrastrado este problema a la hora de solucionarlo nos llevó bastante tiempo, además de ese lo único que nos ha causado trabajo era el error que nos daba cuando el fichero estaba vacío pero ya está todo solucionado.

### Conclusiones:

---

La experiencia al realizar el proyecto ha sido muy buena, al haberlo realizado otro año ha resultado más fácil era más fácil de comprender que se pedía y prever futuras entregas. Este año ha sido bastante llevadera la práctica y pienso que para nuestros compañeros también ha sido así, o eso esperamos.