

(C+)

PROGRAM Comunica;

< Comunicación entre PC y CASIO PB-1000, trabajando este último
como un terminal. >

```
<$I Asynchi.inc >      { Level One Global Definitions }
<$I OpenA1.INC >       { Open a COM port }
<$I SetOpA1.INC >       { Set Options of transmission }
<$I WrtChA1.INC >       { Envia un caracter al puerto serie }
<$I RdChA1.INC >       { Lee un caracter del puerto serie }
<$I CloseA1.INC >       { Close a COM port }
<$I QSizeA1.INC >       { Return de input Queue size }
```

TYPE

```
TipoCadena=STRING[250]; { Cadena de caracteres }
```

CONST

```
{ Códigos CCITT nº 5 }
NUL  = #0;   { Null }
STX  = #2;   { Start of text }
ETX  = #3;   { End of text }
EOT  = #4;   { End of transmission }
ENQ  = #5;   { Enquiry }
ACK  = #6;   { Acknowledge }
BEL  = #7;   { Bell }
US   = #31;  { Unit separator }
```

```
{ Códigos touch keys (pantalla tactil del PB-1000) }
TK1  = #240; TK2  = #241; TK3  = #242; TK4  = #243;
TK5  = #244; TK6  = #245; TK7  = #246; TK8  = #247;
TK9  = #248; TK10 = #249; TK11 = #250; TK12 = #251;
TK13 = #252; TK14 = #253; TK15 = #254; TK16 = #255;
```

```
{ Códigos del CASIO PB-1000 }
```

```
BKSP = #8; { BackSpace; borra el último carácter escrito en linea lógica }
HOME = #11; { Coloca cursor al principio de la pantalla interna }
CLS  = #12; { Clear Screen; Limpia la pantalla }
CRLF = #13; { Carriage return & Line feed }
RLDN = #16; { Roll Down; Sube una linea el cursor bajando la pantalla }
```

```
{ Códigos no standard }
```

```
RD   = #200; { Petición de lectura de un fichero }
WR   = #201; { Petición de escritura de un fichero }
```

```
InQsize = 8000; { Tamaño de la cola de entrada }
OutQsize = 250; { Tamaño de la cola de salida }
Qsize   = 8260; { Tamaño de cola: Mayor que InQsize+OutQsize }
Port    = COM1; { Puerto serie utilizado }
Pant   : ARRAY[1..8] OF TipoCadena =
{*** COMUNICACIONES ***,
 [Contab][Estrac],
 '< FIN >,'
```

```
'[ Estudio del mes en pantalla. ]',
 '[ Estudio del mes en impresora ]',
 '< MENU >,'
```

```
'[ Extractos en impresora ]',
 '< MENU >');
```

VAR

```
Err     : Integer;
Queue  : ARRAY[1..Qsize] of CHAR;
Ch     : CHAR;
```

```
EnCola : INTEGER;          { Caracteres que hay en la cola de entrada }
PortStatus: INTEGER;
Fin : BOOLEAN;             { Indicador de fin del programa remoto }
Continuar : BOOLEAN;       { Utilizado para detenciones momentaneas }
Historia : TEXT;           { Fichero de Historia }
NameHist : TipoCadena;    { Nombre del fichero que contiene la Historia }
Dia,Mes,Anio : STRING[2];  { Fecha de la Historia }
Impresora : TEXT;          { Salidas por impresora }
Aux : INTEGER;             { Variable de propósito general }
Resto : INTEGER;            { Variable de propósito general }
```

PROCEDURE Inicializar;

{ Inicializa el puerto serie para su uso }

BEGIN { Inicializar }

```
Err:=OpenA1(Port,InQSize,OutQSize,0,0,Addr(Queue));
Err:=SetOpA1(Port,1,5); { Velocidad de transmisión 2400 Baudios }
Err:=SetOpA1(Port,2,0); { Transmisión sin bit de paridad }
Err:=SetOpA1(Port,3,3); { Transmisión en unidades de 8 bits }
Err:=SetOpA1(Port,4,0); { Un bit de parada }
Err:=SetOpA1(Port,5,1); { Habilitar el control de flujo remoto. }
Err:=SetOpA1(Port,6,1); { Habilitar el control de flujo local. }
Err:=SetOpA1(Port,9,0); { Inhabilitar señal CLEAR TO SEND (CTS) }
```

END { Inicializar };

PROCEDURE Recibe(VAR Ch:CHAR);

{ Recibe un carácter del puerto serie }

BEGIN { Recibe }

```
REPEAT Err:=RdChA1(Port,Ch,EnCola,PortStatus) UNTIL Ch<>NUL
```

END { Recibe };

PROCEDURE Envia(Cadena:TipoCadena);

{ Envía una Cadena de caracteres por el puerto serie }

VAR Pos: INTEGER; { Puntero de la cadena }

BEGIN { Envía }

```
FOR Pos:=1 TO LENGTH(Cadena) DO Err:=WrtChA1(Port,COPY(Cadena,Pos,1))
```

END { Envía };

FUNCTION ImpresoraLista : BOOLEAN;

{ Devuelve TRUE si la impresora está preparada }

VAR st : BYTE;

BEGIN { ImpresoraLista }

```
InLine( $B4/$02/          { MOV AH,02h      }
        $BA/$00/$00/         { MOV DX,0000      }
        $CD/$17/            { INT 17        }
        $88/$A6/st           { MOV [BP+st],AH } );
```

```
ImpresoraLista:=(st=$90)
```

END { ImpresoraLista };

PROCEDURE Contabilidad;

{ Se encarga del módulo de contabilidad }

VAR

```
Cadena : TipoCadena; { Cadena de caracteres de uso común }
Previsto,                  { Gastos previstos en campos y/o subcampos }
Gastado,                   { Gastos realizados en campos y/o subcampos }
TotalPrev,                 { Total previsto en el mes }
TotalGast : REAL;          { Total gastado en el mes }
```

PROCEDURE ContaFich;

{ Crea el fichero historia de contabilidad }

VAR

```
Campo : ARRAY[1..100] OF TipoCadena; { Nombres de los campos }
NumCampo : INTEGER; { Usado como puntero de "Campos" }
TotCampos: INTEGER; { Número total de campos contenidos en "Campos" }
BEGIN { ContaFich }
  Dia:=''; Mes:=''; Anio:='';
  Envia(STX+BEL+CLS+CRLF+'Trabajando. Espere por favor.'+RD+'FECHA.CON'+ETX);
```

```

Recibe(Ch); {STX}
{Formato fecha: MM-DD-AAAA}
Recibe(Ch); Mes:=Mes+Ch;
Recibe(Ch); Mes:=Mes+Ch;
Recibe(Ch);
Recibe(Ch); Dia:=Dia+Ch;
Recibe(Ch); Dia:=Dia+Ch;
Recibe(Ch); Recibe(Ch); Recibe(Ch);
Recibe(Ch); Anio:=Anio+Ch;
Recibe(Ch); Anio:=Anio+Ch;
{ La fecha recibida corresponde al dia de la apertura de la contabilidad
  del mes. Si la fecha de apertura es posterior al dia 24 del mes,
  supondré que se debe a que la contabilidad anterior se cerró antes
  de tiempo y que la contabilidad abierta corresponde al mes siguiente
  al del correspondiente a su apertura. }
IF Dia>'24' THEN BEGIN { La Historia corresponde al mes siguiente }
  IF Mes='09' THEN Mes:='10' {El año no cambia}
  ELSE
    IF Mes<>'12' THEN Mes:=COPY(Mes,1,1)+SUCC(COPY(Mes,2,1))
    ELSE BEGIN { Cambia el año y el mes pasa a ser Enero }
      Mes:='01';
      Anio:=COPY(Anio,1,1)+SUCC(COPY(Anio,2,1)); { Valido para decada '90 }
    END;
END;
REPEAT Recibe(Ch); {ETX} UNTIL Ch=ETX;
{Nombre fichero de historia: CONMM-AA.PB donde MM=Mes y AA=Año}
Cadena:='CON'+Mes+'-'+Anio+'.PB';
NameHist:='C:\PB-1000\'+Cadena;
ASSIGN(Historia,NameHist);
REWRITE(Historia);
Envia(STX+RD+'CAMPOS.CON'+ETX);
NumCampo:=1;
Recibe(Ch); {STX}
Recibe(Ch); {Primer caracter del primer campo}
WHILE NOT (Ch=ETX) DO BEGIN
  Cadena:='';
  WHILE NOT (Ch=CRLF) DO BEGIN {Leer un nombre de campo}
    Cadena:=Cadena+Ch;
    Recibe(Ch)
  END;
  Campo[NumCampo]:=Cadena;
  TotCampos:=NumCampo;
  NumCampo:=NumCampo+1;
  Recibe(Ch)
END;
FOR NumCampo:=1 TO TotCampos DO BEGIN
  WRITELN(Historia,'*'+Campo[NumCampo]);
  Envia(STX+RD+COPY(Campo[NumCampo],1,8)+ETX);
  Recibe(Ch); {STX}
  Recibe(Ch); {Primer caracter del fichero del campo}
  WHILE NOT(Ch=ETX) DO BEGIN
    WRITE(Historia,Ch);
    Recibe(Ch)
  END
END;
CLOSE(Historia);
END { ContaFich };

PROCEDURE ContaImpresora;
{ Presenta el estudio contable en la impresora }
BEGIN { ContaImpresora }
  ContaFich;
  WHILE NOT ImpresoraLista DO BEGIN
    Envia(STX+BEL+BEL+BEL+CLS+CRLF+'LA IMPRESORA NO ESTA PREPARADA'+
          CRLF+'Enciendala y ponga papel.'+
          CRLF+'DESPUES PULSE "EXE"'+ETX);
  END;
END;

```

3050

3052
3052
3052

```

    REPEAT Recibe(Ch) UNTIL Ch=CRLF;
END;
Envia(STX+BEL+BEL+CLS+CRLF+' IMPRESION REALIZANDOSE'+NUL+NUL+NUL+NUL+
      NUL+NUL+NUL+NUL+NUL+NUL+NUL+NUL+BEL+CLS+EOT);
Fin:=TRUE; {Fin del programa remoto}
{Comienza la impresión}
TotalPrev:=0;
TotalGast:=0;
ASSIGN(Historia,NameHist);
RESET(Historia);
ASSIGN(Impresora,'PRN:');
REWRITE(Impresora);
WRITE(Impresora,#27+'!'+#160+'ESTUDIO CONTABLE DE ');
IF Mes='01' THEN WRITE(Impresora,'ENERO');
IF Mes='02' THEN WRITE(Impresora,'FEBRERO');
IF Mes='03' THEN WRITE(Impresora,'MARZO');
IF Mes='04' THEN WRITE(Impresora,'ABRIL');
IF Mes='05' THEN WRITE(Impresora,'MAYO');
IF Mes='06' THEN WRITE(Impresora,'JUNIO');
IF Mes='07' THEN WRITE(Impresora,'JULIO');
IF Mes='08' THEN WRITE(Impresora,'AGOSTO');
IF Mes='09' THEN WRITE(Impresora,'SEPTIEMBRE');
IF Mes='10' THEN WRITE(Impresora,'OCTUBRE');
IF Mes='11' THEN WRITE(Impresora,'NOVIEMBRE');
IF Mes='12' THEN WRITE(Impresora,'DICIEMBRE');
WRITELN(Impresora,' DE 19'+Anio);
WRITELN(Impresora,#27+'!'+#128); {Activa modo impresión normal subrayado}
WRITELN(Impresora,
        'CONCEPTO' PREVISTO GASTADO DIFERENCIA');
WRITELN(Impresora);
WHILE NOT EOF(Historia) DO BEGIN
  READLN(Historia,Cadena);
  IF COPY(Cadena,1,1)='*' THEN BEGIN {Se trata de un nombre de campo}
    Cadena:=COPY(Cadena,2,32);
    Resto:=36-LENGTH(Cadena);
    FOR Aux:=1 TO Resto DO Cadena:=Cadena+' ';
    WRITE(Impresora,#27+'!'+#144+Cadena);
    READLN(Historia,Previsto,Gastado);
    TotalPrev:=TotalPrev+Previsto;
    TotalGast:=TotalGast+Gastado
  END
  ELSE BEGIN { Se trata de un nombre de subcampo }
    Cadena:=' '+COPY(Cadena,1,30);
    Resto:=36-LENGTH(Cadena);
    FOR Aux:=1 TO Resto DO Cadena:=Cadena+' ';
    WRITE(Impresora,#27+'!'+#0+' '+Cadena);
    READLN(Historia,Previsto,Gastado);
  END;
  WRITE(Impresora,#27+'!'+#0); {Modo normal de impresión}
  STR(Previsto:0:0,Cadena);
  Resto:=11-LENGTH(Cadena);
  FOR Aux:=1 TO Resto DO Cadena:=Cadena+' ';
  WRITE(Impresora,Cadena);
  STR(Gastado:0:0,Cadena);
  Resto:=11-LENGTH(Cadena);
  FOR Aux:=1 TO Resto DO Cadena:=Cadena+' ';
  WRITE(Impresora,Cadena);
  WRITELN(Impresora,(Previsto-Gastado):0);
END {WHILE};
CLOSE(Historia);
WRITELN(Impresora,
        ' TOTAL DEL MES: ') ;
WRITE(Impresora,#27+'!'+#140+'TOTAL DEL MES: ') ;
WRITE(Impresora,#27+'!'+#0); {Modo normal de impresión}
STR(TotalPrev:0:0,Cadena);
Resto:=11-LENGTH(Cadena);

```

```

FOR Aux:=1 TO Resto DO Cadena:=Cadena+' ';
WRITE(Impresora,Cadena);
STR(TotalGast:0:0,Cadena);
Resto:=11-LENGTH(Cadena);
FOR Aux:=1 TO Resto DO Cadena:=Cadena+' ';
WRITE(Impresora,Cadena);
WRITELN(Impresora,(TotalPrev-TotalGast):0);
WRITE(Impresora,#12); {Salto de página}
CLOSE(Impresora);
END { ContaImpresora };

BEGIN { Contabilidad }
Envia(STX+BEL+CLS+CRLF+Pant[4]+Pant[5]+Pant[6]+ETX);
Recibe(Ch);
WHILE NOT (Ch IN [TK5 ,TK6 ,TK7 ,TK8 ,
                  TK9 ,TK10,TK11,TK12,
                  TK13]) DO BEGIN
  Envia(ACK);
  Recibe(Ch);
END;
CASE Ch OF
  TK5 ,TK6 ,TK7 ,TK8 : {ContaPantalla};
  TK9 ,TK10,TK11,TK12 : ContaImpresora;
  TK13                 : {MENU};
END;
END { Contabilidad };

PROCEDURE Estractos;
{ Se encarga del módulo de extractos }
BEGIN { Estractos }
Envia(STX+BEL+CLS+CRLF+Pant[7]+CRLF+Pant[8]+ETX);
Recibe(Ch);
WHILE NOT (Ch IN [TK5 ,TK6 ,TK7 ,TK8 ,
                  TK13]) DO BEGIN
  Envia(ACK);
  Recibe(Ch);
END;
CASE Ch OF
  TK5 ,TK6 ,TK7 ,TK8 : {ListaPantalla};
  TK13                 : {MENU};
END;
END { Estractos };

BEGIN { Comunica }
Fin:=FALSE;
Inicializar;
REPEAT
  ClrScr;
  GotoXY(10,10);WRITELN('PULSE "ESC" SI QUIERE USAR EL ORDENADOR');
  Continuar:=FALSE;
  REPEAT
    IF KEYPRESSED THEN BEGIN
      READ(Kbd,Ch);
      IF Ch=#27 THEN BEGIN
        Err:= CloseA1(Port);
        HALT(1);
      END;
    END;
  END;
  Err:=QSizeA1(Port,EnCola,PortStatus);
  IF EnCola>0 THEN BEGIN
    Recibe(Ch);
    IF Ch=ENQ THEN Continuar:=TRUE;
  END;
UNTIL Continuar;
ClrScr;
GotoXY(10,10);WRITELN('EL ORDENADOR ESTA SIENDO USADO REMOTAMENTE.');

```

REPEAT

Envia(STX+BEL+CLS+Pant[1]+CRLF+CRLF+Pant[2]+CRLF+Pant[3]+ETX);

Recibe(Ch);

WHILE NOT (Ch IN [TK10,TK11,TK13]) DO BEGIN

Envia(ACK);

Recibe(Ch)

END;

CASE Ch OF

TK10 : Contabilidad;

TK11 : Extractos;

TK13 : Fin:=TRUE

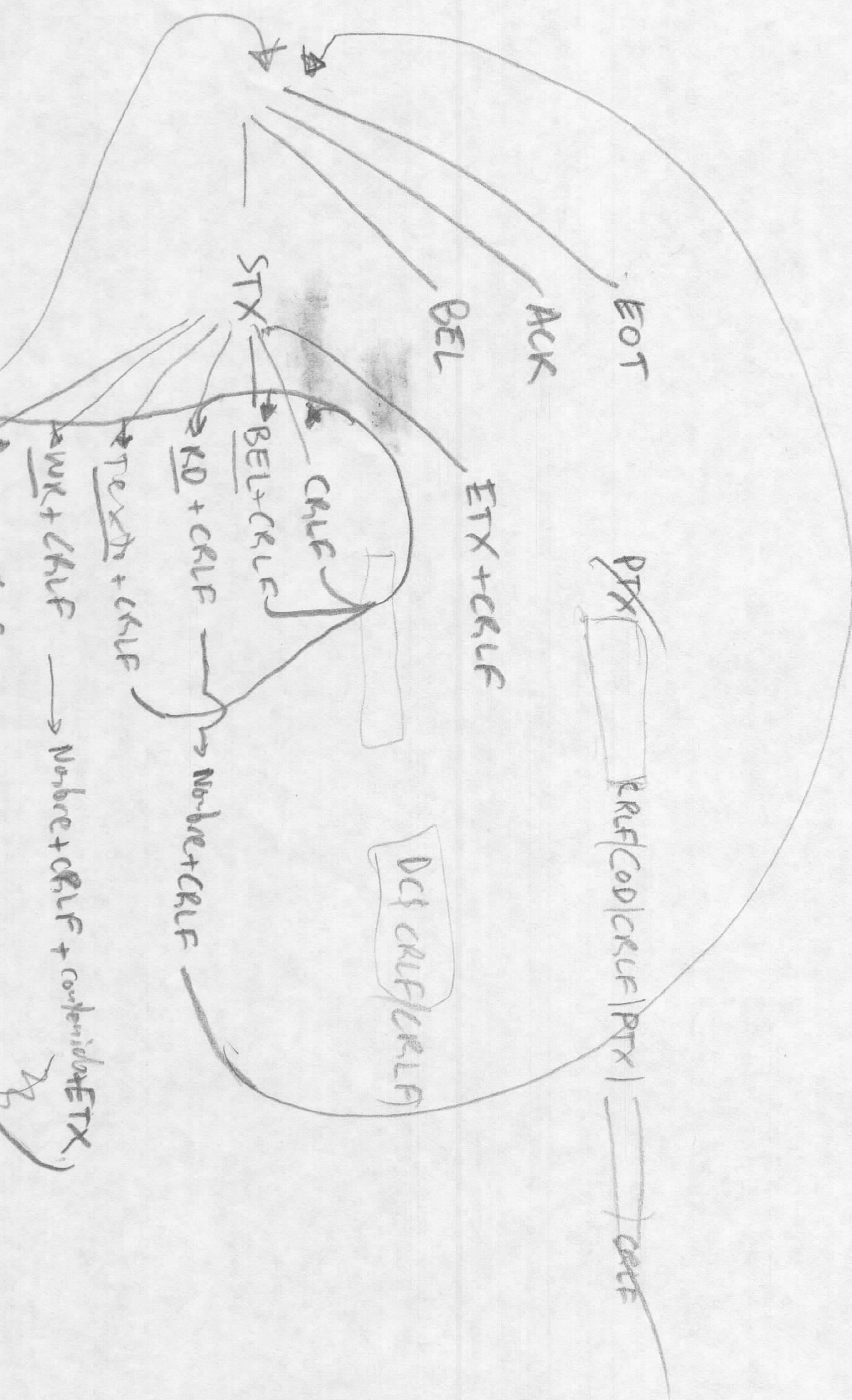
END;

UNTIL Fin;

Envia(STX+BEL+CLS+EOT);

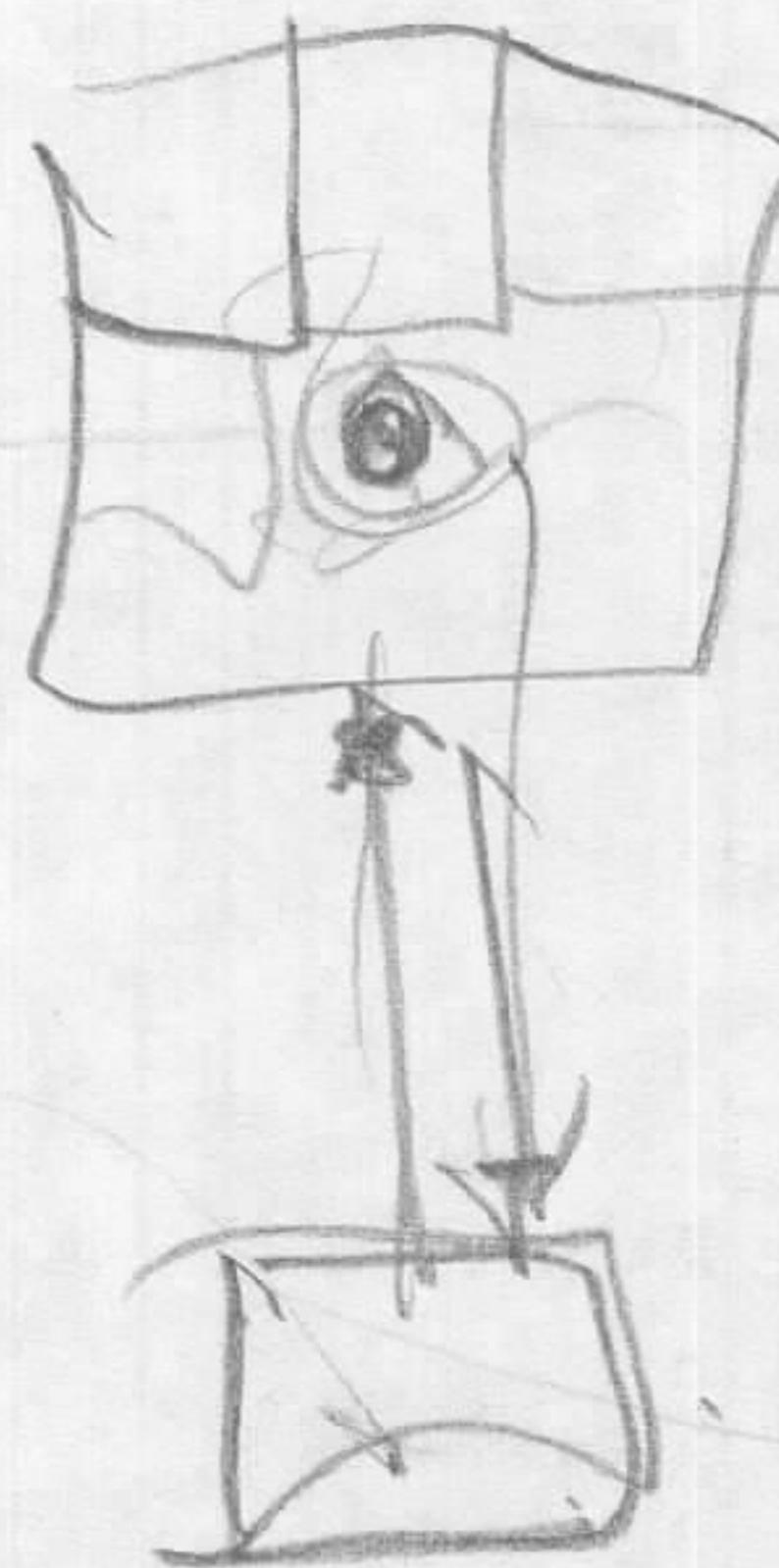
UNTIL FALSE; { Ciclo permanentemente cerrado }

END(Comunica).



Prev Gast
 Sobre cargo
 Prev Gast
 Sobre cargo
 prev carb
 !

Corp
 Corp 1
 Corp 2
 Corp 3
 !



ESTUDIO CONTABLE MAYO 1989

| <u>CONCEPTO</u> | <u>PREVISTO</u> | <u>GASTADO</u> | <u>DIFERENCIA</u> |
|---------------------|-----------------|----------------|-------------------|
| <u>ALIMENTACIÓN</u> | 37 | 47 | ↑ 56 |
| FRUTAS Y VERDURAS | | | |
| DULCES | | | |
| LACTEOS | | | |
| CARNE | ! | ! | ! |
| <u>CASA</u> | | | |
| LIMPIEZA | | | |
| BANOS | | | |
| REPARACIONES | | | |
| MENAJE | | | |
| TOTAL | | | |



Caro ~

CON
CON 07-89.PB

MES, AÑO

ALIMENTACIÓN Prev., costos

FRUTAS Prev., costos

DULCES Prev., costos

*CASA Pre costos



Recibir fecha; abrir fichero fecha

listo ligado. -modif.

Recibir & [CAMPO] → guardarlo en un punto ~~en~~ ~~entre~~ un punto de strings.

Añadir guardar a fichero fecha con *

Tomar cada campo → pedirlo al terminal.

recibir el campo.

almacenar sus datos en fichero fecha.

Hasta que no queden más campos.

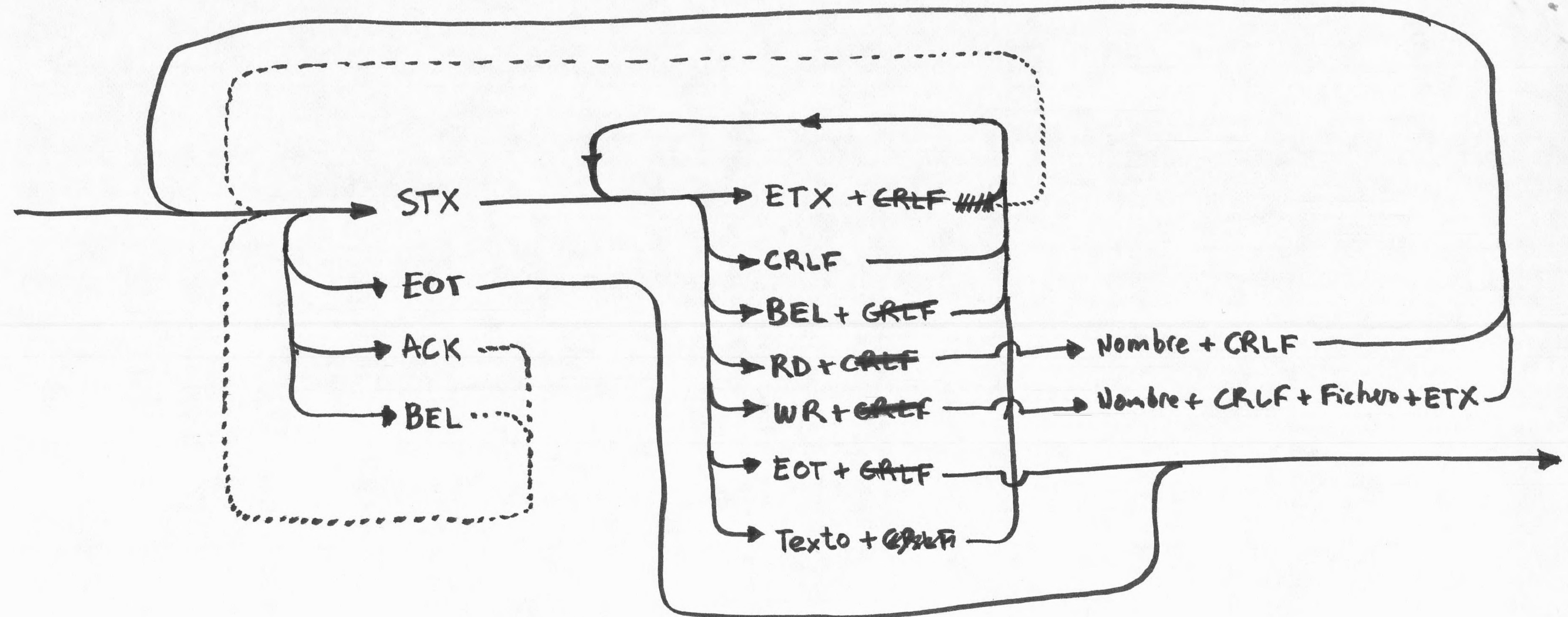
Cerrar fichero fecha.

Dar mensaje de recibido y mostrar con formato el fichero fecha en el

terminal o en papel.

Regresar al menu principal.





— : paso directo.

---- : paso tras intercambiar sentido o puesta.