

001 Registrador de Eventos

8 de abril de 2021 13:14

Tarjetas internas:

EVR.CV.VA-X1
EVR.CV.VA-X4
EVR.CV.VA-X5
EVR.CV.VA-X6
EVR.CV.VA-X7
EVR.CV.VA-X8

Ubicación proyecto:	C:\Users\Ignacio \OneDrive \Documentos\002 _CAF\CAF\Arduino \simPulsosVelocidad_ v2conErrores_v5
---------------------	---

Actividad	Status	Actividad	Status
Diseño del programa	100%	Simulación funcionamiento Placa	100%
Diseño de la placa	100%	Placa Impresa y perforada	100%
Montaje de Componentes	30%	Funcionamiento de Placa	
Desarrollo conexiones VDR	75%	Montaje de placa en VDR	
Funcionamiento		Modificación_1	Cambio del Diseño de Placa (versión 2)
Modificación_2		Modificación_3	
Revisión			

Prueba en Protoboard con optoacopladores:

El Arduino queda pegado y no carga sketch nuevos, probablemente
Hayan problemas en el microprocesador que ocasionan problemas en la carga
De nuevos sketch. (09/04/2021).

Sol1: Cambiar Microprocesador.

Sol2: Reemplazar placa completa Arduino.

Sol3: Buscar ayudas y soluciones en foros de internet respecto al error.

Sol2 fue la última aplicada, la placa se quemó en la comunicación con el
computador, por ende no permite modificar el integrado o ver los datos
seriales en el computador, la placa solo quedo disponible para arrancar
programas sin la necesidad de conectarlo a un computador, es decir, su
funcionamiento caer en solo ejecutar programas sin comunicación alguna.

002 Notas de avance

23 de abril de 2021 9:38

23/04: los transistores 2N2222 conmutan como corresponde, pero los optoacopladores no realizan dicha acción tan vez se deba al tipo de componente, su arquitectura u otra característica (PRESENCIA DE CORTE EN PISTA DE TIERRA LO QUE CAUSABA EL ERROR EN EL CIRCUITO).

Panel indicador de LEDs

LED	Función	Comportamiento cuando el aparato se enciende
+5 V	Luz verde	La alimentación de +5 voltios esta funcionando correctamente.
+/-12 V	Luz verde	La alimentación de +12 voltios esta funcionando correctamente.
RUN	Verde intermitente a ritmo variable	El programa del TELOC®1500 esta funcionando correctamente
	No luce	Error en el programa – contactar con un centro de servicio o HaslerRail.
REC	Luz verde	La función de registro del TELOC®1500 está funcionando correctamente.
	Verde intermitente	Alarma simple Error en el sistema / en los datos de la jornada que están siendo grabados.
	No luce	Alarma fatal Atención: Los datos no se están registrando!!
INFO	Luz roja	Mensaje de error de nivel 2 (interno) o de nivel 3 (externo).
EXT	Luz roja	Fallo en una unidad periférica – la fuente del fallo no proviene del TELOC®1500.
USER1 USER2	Luces amarillas	Reservada para uso especificada del cliente

003 Análisis Tarjetas del VDR

8 de abril de 2021 15:14

EVR.CV.VA-X8

GND >> tierra
b30(GND) >> GND Sensor
d28(sig. A) >> Señal Sensor 1
d30(+12v) >> EVR.CV.VA-X1-d6(AN2+)
b2(ID1_0) -- z2(ID1_GND)
b32(ID2_9) -- z32(ID2_GND)

EVR.CV.VA-X7

b4 >>
z4 << EVR.CV.VA-X11-a3(POSUB)
b2 >>
z2 << EVR.CV.VA-X11-a3(POSUB)
b32(ID0) -- b30(ID_HI)

EVR.CV.VA-X1

d6(AN2+) << EVR.CV.VA-X8-d30(+12)
z10(AN2-) << EVR.CV.VA-X4-d8(AN1+)
b6 -- z8 [PUENTE]
GND >> tierra
z10(GND) >> GND Sensor 3
b8(sig. A) >> Señal Sensor 3
z8(+12v) >> Vcc Sensor 3
z28 >>
d28 << EVR.CV.VA-X11-a3(POSUB)
z26 >>
d26 >> EVR.CV.VA-X11-a3(POSUB)

EVR.CV.VA-X6

b4 >>
z4 << EVR.CV.VA-X11-a3(POSUB)
b2 >>
z2 << EVR.CV.VA-X11-a3(POSUB)
d32(ID0) -- b30(ID_HI)

EVR.CV.VA-X5

b2(ID1) -- z6(ID_HI)

EVR.CV.VA-X4

GND >> tierra
d8(AN1+) >> EVR.CV.VA-X1-z10(AN2-)
d10(AN1-) >> +12v del Sensor
b8(AINI2) -- d10(AINI2-)
b4(GND) >> GND Sensor
d6(sig. A) >> Señal Sensor 2
d2(ID0) -- z6(ID_HI)

DIJN3--LUZ-DERECHA --CIERRE DE PUERTA
DIJNE--LUZ-IZQUIERDA--CC

004 Programa Arduino

8 de abril de 2021 15:16

Cambiar algunas variables globales a locales no funciona, sigue utilizando más del 80% de memoria para dichas variables

```
//---
DEFINICION_PINES-----
int pulseOut = 3;           //salida pulsos que simulan velocidad, se selecciona el pin11 por sus
cualidades de salida PWM
int luz = 4;               //indicador que representa LUZ indicadora del TREN
int sirena = 5;            //indicador que representa la SIRENA o GRABACIÓN del tren
int puerta = 6;            //indicador que representa las PUERTAS del TREN
int X1_VT_6 = 10;          //lectura de variable provimiente de tarjeta X1 para rele control de velocidad
6
int X1_VT_05 = 7;          //lectura de variable provimiente de tarjeta X1 para rele control de velocidad
05
int X7_VT_6 = 11;          //lectura de variable provimiente de tarjeta X7 para rele control de velocidad
6
int X7_VT_05 = 8;          //lectura de variable provimiente de tarjeta X7 para rele control de velocidad
05
int X6_VT_6 = 12;          //lectura de variable provimiente de tarjeta X6 para rele control de velocidad
6
int X6_VT_05 = 9;          //lectura de variable provimiente de tarjeta X6 para rele control de velocidad
05
int error = 13;            //variable destinada a dar indicación de ERROR a causa de alguna condición
incompleta
//---
DEFINICION_VARIABLES-----
-
int countMax = 1160;        //es 1.25 veces la frecuencia maxima
float tiempo;              //tiempo que estara en funcion de la frecuencia
float i, j;                //contadores
float vel;                 //valor de la velocidad en funcion de la frecuencia
const float vel_1 = 0.5;   //velocidad para apagar el rele 1
const float vel_2 = 6;     //velocidad para apagar el rele 2
const int z = 100;         //ventanas de la rueda
const float var1 = 0.953;  //diametro de la rueda en metros
const float var2 = 3.6;    //constante de velocidad del TREN
const float var3 = 100;    //ventanas de la rueda

void setup() {
  Serial.begin(9600);        //velocidad de lectura y escritura del arduino.
  pinMode(pulseOut, OUTPUT);
  pinMode(puerta, OUTPUT);
  pinMode(luz, OUTPUT);
  pinMode(sirena, OUTPUT);
  pinMode(X1_VT_6, INPUT);   //se define como entrada para lectura de reaccion del REGISTRADOR
  DE EVENTOS
  pinMode(X1_VT_05, INPUT);  //se define como entrada para lectura de reaccion del REGISTRADOR
  DE EVENTOS
  pinMode(X7_VT_6, INPUT);
  pinMode(X7_VT_05, INPUT);
  pinMode(X6_VT_6, INPUT);
  pinMode(X6_VT_05, INPUT);
  pinMode(error, OUTPUT);
}

void loop() {
  digitalWrite(puerta, HIGH);
  Serial.println("TREN sale de ESTACIÓN");
  acelerar();
  Serial.println("VELOCIDAD MAXIMA");
  desacelerar();
  Serial.println("TREN ha llegado a DESTINO");
  enEstacion();
}
//-----FUNCIONES-----
//---SALIDA_SEÑAL_CUADRADA-----
void senalOut() {           //Funcion que se encarga de generar los estados de los semiciclos
  correspondientes.
  tiempo = (1000 / (i / (1.25))); //tiempo de duracion de cada semiCiclo en milisegundos
  digitalWrite(pulseOut, HIGH);
  delay(tiempo / 2);         //semiciclo positivo
  digitalWrite(pulseOut, LOW);
  delay(tiempo / 2);         //semiciclo negativo
}
//---ACELERACION_DEL_TREN-----
void acelerar() {
  for (i = 1; i < countMax; i++) { //se calculo que aproximadamente los 928hz se alcanzan los 100km/h
    senalOut();
    velMax();
    lecturasEnSubida05();
    lecturasEnSubida6();
  }
}
//---DESACELERACION_DEL_TREN-----
void desacelerar() {
  for (i = countMax; i >= 1; i--) {
    senalOut();
  }
}
```

```

    velMax();
    lecturasEnBajada6();
    lecturasEnBajada05();
}
}
//---VISUALIZAR_VELOCIDAD-----
void velMax() {
    vel = ((i / 1.25) * PI * var1 * var2 / z);
    Serial.println(vel);
}
//---ACCIONES_EN_ESTACION-----
void enEstacion() {
    digitalWrite(puerta, LOW);
    Serial.println("APERTURA DE PUERTAS");
    delay(4500);
    digitalWrite(luz, HIGH);
    Serial.println("Se enciende indicador Luminoso");
    delay(2000);
    digitalWrite(sirena, HIGH);
    Serial.println("Suenan la Sirena");
    delay(2500);
    digitalWrite(sirena, LOW);
    Serial.println("Deja de sonar Sirena");
    digitalWrite(luz, LOW);
    Serial.println("Se apaga luz indicadora");
    digitalWrite(puerta, HIGH);
    Serial.println("CIERRE DE PUERTAS");
    delay(1000);
}
//---FUNCIONES_POR_ERRORES-----
void error_001() { //error 1 cuando X1_VT_05 debiera ser HIGH
    Serial.println("");
    for (;;) {
        digitalWrite(error, HIGH);
        delay(500);
        digitalWrite(error, LOW);
        delay(500);
    }
}
void error_002() { //error 2 cuando X1_VT_6 deberia ser HIGH
    for (;;) {
        for (j = 0; j < 2; j++) {
            digitalWrite(error, HIGH);
            delay(500);
            digitalWrite(error, LOW);
            delay(500);
        }
        delay(1000);
    }
}
void error_003() { //error 3 cuando X7_VT_05 deberia ser HIGH
    for (;;) {
        for (j = 0; j < 3; j++) {
            digitalWrite(error, HIGH);
            delay(500);
            digitalWrite(error, LOW);
            delay(500);
        }
        delay(1000);
    }
}
void error_004() { //error 4 cuando X7_VT_6 deberia ser HIGH
    for (;;) {
        for (j = 0; j < 4; j++) {
            digitalWrite(error, HIGH);
            delay(500);
            digitalWrite(error, LOW);
            delay(500);
        }
        delay(1000);
    }
}
void error_005() { //error 5 cuando X6_VT_05 deberia ser HIGH
    for (;;) {
        for (j = 0; j < 5; j++) {
            digitalWrite(error, HIGH);
            delay(500);
            digitalWrite(error, LOW);
            delay(500);
        }
        delay(1000);
    }
}
void error_006() { //error 6 cuando X6_VT_6 deberia ser HIGH
    for (;;) {
        for (j = 0; j < 6; j++) {
            digitalWrite(error, HIGH);
            delay(500);
            digitalWrite(error, LOW);
            delay(500);
        }
        delay(1000);
    }
}

```

```

    }
}
void error_007() { //error 7 Ninguna lectura cambia
for (;;) {
    for (j = 0; j = 7; j++) {
        digitalWrite(error, HIGH);
        delay(500);
        digitalWrite(error, LOW);
        delay(500);
    }
    delay(1000);
}
}
//---
LECTURAS-----
-----
void lecturasEnSubida05() {
    Serial.println(digitalRead(X1_VT_05));
    Serial.println(digitalRead(X7_VT_05));
    Serial.println(digitalRead(X6_VT_05));
    while (vel > vel_1) { //Cuando la velocidad calculada es MAYOR a 0.5KM/h y...
        if (digitalRead(X1_VT_05) == LOW) { //Si la lectura es 0 digital, se ejecuta el error 1
            Serial.println("ERROR_011: la variable X1_VT_05 debia arrojar un UNO, pero recibio un CERO\nERROR
CONTROL DE VELCIDAD RELE 0.5KM/H");
            error_001();
            break;
        }
        else if (digitalRead(X7_VT_05) == LOW) { //Si la lectura es 0 digital, se ejecuta el error 3
            Serial.println("ERROR_021: la variable X7_VT_05 debia arrojar un UNO, pero recibio un CERO\nERROR
CONTROL DE VELCIDAD RELE 0.5KM/H");
            error_003();
            break;
        }
        else if (digitalRead(X6_VT_05) == LOW) { //Si la lectura es 0 digital, se ejecuta el error 5
            Serial.println("ERROR_031: la variable X6_VT_05 debia arrojar un UNO, pero recibio un CERO\nERROR
CONTROL DE VELCIDAD RELE 0.5KM/H");
            error_005();
            break;
        }
        else{
            break;
        }
    }
    break;
}
}
void lecturasEnSubida6(){
    Serial.println(digitalRead(X1_VT_6));
    Serial.println(digitalRead(X7_VT_6));
    Serial.println(digitalRead(X6_VT_6));
    while (vel > vel_2) { //Cuando la velocidad calculada es MENOR a 6km/h y...
        if (digitalRead(X1_VT_6) == LOW) { //Si la lectura es 1 digital, se ejecuta el error 2
            Serial.println("ERROR_012: la variable X1_VT_6 debia arrojar un UNO, pero recibio un CERO\nERROR
CONTROL DE VELCIDAD RELE 6KM/H");
            error_002();
            break;
        }
        else if (digitalRead(X7_VT_6) == LOW) { //Si la lectura es 1 digital, se ejecuta el error 4
            Serial.println("ERROR_022: la variable X7_VT_6 debia arrojar un UNO, pero recibio un CERO\nERROR
CONTROL DE VELCIDAD RELE 6KM/H");
            error_004();
            break;
        }
        else if (digitalRead(X6_VT_6) == LOW) { //Si la lectura es 1 digital, se ejecuta el error 6
            Serial.println("ERROR_032: la variable X6_VT_6 debia arrojar un UNO, pero recibio un CERO\nERROR
CONTROL DE VELCIDAD RELE 6KM/H");
            error_006();
            break;
        }
        else{
            break;
        }
    }
    break;
}
}
void lecturasEnBajada05() {
    Serial.println(digitalRead(X1_VT_05));
    Serial.println(digitalRead(X7_VT_05));
    Serial.println(digitalRead(X6_VT_05));
    while (vel <= vel_1) { //Si la velocidad calculada es MENOR a 0.5KM/h y...
        if (digitalRead(X1_VT_05) == HIGH) { //Si la lectura es 1 digital, se ejecuta el error 1
            Serial.println("ERROR_013: la variable X1_VT_05 debia arrojar un CERO, pero recibio un UNO\nERROR
CONTROL DE VELCIDAD RELE 0.5KM/H");
            error_001();
            break;
        }
        else if (digitalRead(X7_VT_05) == HIGH) { //Si la lectura es 1 digital, se ejecuta el error 3
            Serial.println("ERROR_023: la variable X7_VT_05 debia arrojar un CERO, pero recibio un UNO\nERROR
CONTROL DE VELCIDAD RELE 0.5KM/H");
            error_003();
            break;
        }
        else if (digitalRead(X6_VT_05) == HIGH) { //Si la lectura es 1 digital, se ejecuta el error 5

```

```

Serial.println("ERROR_033: la variable X6_VT_05 debia arrojar un CERO, pero recibio un UNO\nERROR
CONTROL DE VELCIDAD RELE 0.5KM/H");
error_005();
break;
}
else{
break;
}
break;
}
}
void lecturasEnBajada6(){
Serial.println(digitalRead(X1_VT_6));
Serial.println(digitalRead(X7_VT_6));
Serial.println(digitalRead(X6_VT_6));
while (vel <= vel_2) { //Si la velocidad calculada es MENOR a 6km/h y...
if (digitalRead(X1_VT_6) == HIGH) { //Si la lectura es 1 digital, se ejecuta el error 2
Serial.println("ERROR_014: la variable X1_VT_6 debia arrojar un CERO, pero recibio un UNO\nERROR
CONTROL DE VELCIDAD RELE 6KM/H");
error_002();
break;
}
else if (digitalRead(X7_VT_6) == HIGH) { //Si la lectura es 1 digital, se ejecuta el error 2
Serial.println("ERROR_024: la variable X7_VT_6 debia arrojar un CERO, pero recibio un UNO\nERROR
CONTROL DE VELCIDAD RELE 6KM/H");
error_004();
break;
}
else if (digitalRead(X6_VT_6) == HIGH) { //Si la lectura es 1 digital, se ejecuta el error 2
Serial.println("ERROR_034: la variable X6_VT_6 debia arrojar un CERO, pero recibio un UNO\nERROR
CONTROL DE VELCIDAD RELE 6KM/H");
error_006();
break;
}
else{
break;
}
break;
}
}
}
//---NOTAS-----
//considerar que la grafica de velocidad que se dibuja (o comprende) es lineal, por lo tanto, su aceleración es
constante mientras que a travez
//de una ecuacion de posición en funcion del tiempo es parabolico, esto considerando el M.R.U.A. (movimiento
rectilineo uniforme acelerado).
//El tiempo aproximado desde la salida del tren hasta la llegada a la estacion toma aproximadamente 1:09
minutos.
//Se cambio la proporcion de la frecuencia de 2.5 a 1.25 veces la frecuencia, esto debido a que el programa
toma mas tiempo en las iteraciones.
//los casos de errores evalua una tarjeta a la vez, queda pendiente versionar casos de pruebas para multiples
combinaciones de tarjetas.

```

0041 Programa Arduino (Actual)

26 de abril de 2021 13:17

```
//---
DEFINICION_PINES-----
-----
int error = 2; //variable destinada a dar indicación de ERROR a causa de alguna condición
incumplida
int pulseOut = 3; //salida pulsos que simulan velocidad, se selecciona el pin11 por sus cualidades de
salida PWM
int cc = 4; //Confirmación Cierre de Puertas.
int luz = 5; //indicador que representa LUZ indicadora del TREN
int sirena = 6; //indicador que representa la SIRENA o GRABACIÓN del tren
int puerta = 7; //indicador que representa las PUERTAS del TREN
int X1_VT_6 = 8; //lectura de variable proviniendo de tarjeta X1 para rele control de velocidad 6
int X7_VT_6 = 9; //lectura de variable proviniendo de tarjeta X7 para rele control de velocidad 6
int X6_VT_6 = 10; //lectura de variable proviniendo de tarjeta X6 para rele control de velocidad 6
int X1_VT_05 = 11; //lectura de variable proviniendo de tarjeta X1 para rele control de velocidad 05
int X7_VT_05 = 12; //lectura de variable proviniendo de tarjeta X7 para rele control de velocidad 05
int X6_VT_05 = 13; //lectura de variable proviniendo de tarjeta X6 para rele control de velocidad 05
//---
DEFINICION_VARIABLES-----
-----
int countMax = 1160; //es 1.25 veces la frecuencia maxima
float tiempo; //tiempo que estara en funcion de la frecuencia
float i, j; //contadores
float vel; //valor de la velocidad en funcion de la frecuencia
const float vel_1 = 0.5; //velocidad para apagar el rele 1
const float vel_2 = 6; //velocidad para apagar el rele 2
const float var1 = 0.953; //diametro de la rueda en metros
const float var2 = 3.6; //constante de velocidad del TREN
const float var3 = 100; //ventanas de la rueda
const float var4 = 1.25; //Cantidad de veces que se multiplico la frecuencia maxima

void setup()
{
  Serial.begin(9600); //velocidad de lectura y escritura del arduino.
  pinMode(pulseOut, OUTPUT);
  pinMode(puerta, OUTPUT);
  pinMode(luz, OUTPUT);
  pinMode(sirena, OUTPUT);
  pinMode(X1_VT_6, INPUT); //se define como entrada para lectura de reaccion del REGISTRADOR
DE EVENTOS
  pinMode(X1_VT_05, INPUT); //se define como entrada para lectura de reaccion del REGISTRADOR
DE EVENTOS
  pinMode(X7_VT_6, INPUT);
  pinMode(X7_VT_05, INPUT);
  pinMode(X6_VT_6, INPUT);
  pinMode(X6_VT_05, INPUT);
  pinMode(error, OUTPUT);
  pinMode(A0, OUTPUT); //Se usaran las entradas analogicas como salidas digitales
  pinMode(A1, OUTPUT);
  pinMode(A2, OUTPUT);
  pinMode(A3, OUTPUT);
}
```



```

}

void loop()
{
  digitalWrite(cc, LOW);
  enEstacion();
  Serial.println("TREN sale de ESTACIÓN");
  acelerar();
  Serial.println("VELOCIDAD MAXIMA");
  desacelerar();
  Serial.println("TREN ha llegado a DESTINO");
}
//-----FUNCIONES-----
//---SALIDA_SEÑAL_CUADRADA-----
void senalOut()
{
  //Funcion que se encarga de generar los estados de los semiciclos
  correspondientes.
  tiempo = (1000 / (i / (var4))); //tiempo de duracion de cada semiCiclo en milisegundos
  digitalWrite(pulseOut, HIGH);
  delay(tiempo / 2); //semiciclo positivo
  digitalWrite(pulseOut, LOW);
  delay(tiempo / 2); //semiciclo negativo
}
//---ACELERACION_DEL_TREN-----
void acelerar()
{
  for (i = 1; i < countMax; i++)
  { //se calculo que aproximadamente los 928hz se alcanzan los 100km/h
    senalOut();
    velMax();
    lecturasEnSubida05();
    lecturasEnSubida6();
  }
}
//---DESACELERACION_DEL_TREN-----
void desacelerar()
{
  for (i = countMax; i >= 1; i--)
  {
    senalOut();
    velMax();
    lecturasEnBajada6();
    lecturasEnBajada05();
  }
}
//---VISUALIZAR_VELOCIDAD-----
void velMax()
{
  vel = ((i / var4) * PI * var1 * var2 / var3);
  Serial.println(vel);
}
//---ACCIONES_EN_ESTACION-----
void enEstacion()
{
  digitalWrite(puerta, LOW);
  Serial.println("APERTURA DE PUERTAS");
  delay(4500);
}

```

```

digitalWrite(luz, HIGH);
Serial.println("Se enciende indicador Luminoso");
delay(2000);
digitalWrite(sirena, HIGH);
Serial.println("Suena la Sirena");
delay(2500);
digitalWrite(sirena, LOW);
Serial.println("Deja de sonar Sirena");
digitalWrite(luz, LOW);
Serial.println("Se apaga luz indicadora");
digitalWrite(puerta, HIGH);
Serial.println("CIERRE DE PUERTAS");
digitalWrite(cc, HIGH);
delay(1000);
}
//---FUNCIONES_POR_ERRORES-----
void error_001()
{ //error 1 cuando X1_VT_05 debiera ser HIGH
  Serial.println("");
  for (;;)
  {
    digitalWrite(error, HIGH);
    delay(500);
    digitalWrite(error, LOW);
    delay(500);
  }
}
void error_002()
{ //error 2 cuando X1_VT_6 deberia ser HIGH
  for (;;)
  {
    for (j = 0; j < 2; j++)
    {
      digitalWrite(error, HIGH);
      delay(500);
      digitalWrite(error, LOW);
      delay(500);
    }
    delay(1000);
  }
}
void error_003()
{ //error 3 cuando X7_VT_05 deberia ser HIGH
  for (;;)
  {
    for (j = 0; j < 3; j++)
    {
      digitalWrite(error, HIGH);
      delay(500);
      digitalWrite(error, LOW);
      delay(500);
    }
    delay(1000);
  }
}
void error_004()
{ //error 4 cuando X7_VT_6 deberia ser HIGH

```

```

for (;;)
{
    for (j = 0; j < 4; j++)
    {
        digitalWrite(error, HIGH);
        delay(500);
        digitalWrite(error, LOW);
        delay(500);
    }
    delay(1000);
}
}
void error_005()
{ //error 5 cuando X6_VT_05 deberia ser HIGH
    for (;;)
    {
        for (j = 0; j < 5; j++)
        {
            digitalWrite(error, HIGH);
            delay(500);
            digitalWrite(error, LOW);
            delay(500);
        }
        delay(1000);
    }
}
void error_006()
{ //error 6 cuando X6_VT_6 deberia ser HIGH
    for (;;)
    {
        for (j = 0; j < 6; j++)
        {
            digitalWrite(error, HIGH);
            delay(500);
            digitalWrite(error, LOW);
            delay(500);
        }
        delay(1000);
    }
}
void error_007()
{ //error 7 Ninguna lectura cambia
    for (;;)
    {
        for (j = 0; j < 7; j++)
        {
            digitalWrite(error, HIGH);
            delay(500);
            digitalWrite(error, LOW);
            delay(500);
        }
        delay(1000);
    }
}
}
//---
LECTURAS-----
-----

```

```

void lecturasEnSubida05()
{
  Serial.println(digitalRead(X1_VT_05));
  Serial.println(digitalRead(X7_VT_05));
  Serial.println(digitalRead(X6_VT_05));
  while (vel > vel_1)
  { //Cuando la velocidad calculada es MAYOR a 0.5KM/h y...
    if (digitalRead(X1_VT_05) == HIGH)
    { //Si la lectura es 0 digital, se ejecuta el error 1
      digitalWrite(A0,HIGH);
      digitalWrite(A1,LOW);
      digitalWrite(A2,LOW);
      digitalWrite(A3,LOW);
      Serial.println("ERROR_011: la variable X1_VT_05 debia arrojar un UNO, pero recibio un CERO
\nERROR CONTROL DE VELCIDAD RELE 0.5KM/H");
      error_001();
      break;
    }
    else if (digitalRead(X7_VT_05) == HIGH)
    { //Si la lectura es 0 digital, se ejecuta el error 3
      digitalWrite(A0,LOW);
      digitalWrite(A1,HIGH);
      digitalWrite(A2,LOW);
      digitalWrite(A3,LOW);
      Serial.println("ERROR_021: la variable X7_VT_05 debia arrojar un UNO, pero recibio un CERO
\nERROR CONTROL DE VELCIDAD RELE 0.5KM/H");
      error_003();
      break;
    }
    else if (digitalRead(X6_VT_05) == HIGH)
    { //Si la lectura es 0 digital, se ejecuta el error 5
      digitalWrite(A0,HIGH);
      digitalWrite(A1,HIGH);
      digitalWrite(A2,LOW);
      digitalWrite(A3,LOW);
      Serial.println("ERROR_031: la variable X6_VT_05 debia arrojar un UNO, pero recibio un CERO
\nERROR CONTROL DE VELCIDAD RELE 0.5KM/H");
      error_005();
      break;
    }
    else
    {
      break;
    }
    break;
  }
}

void lecturasEnSubida6()
{
  Serial.println(digitalRead(X1_VT_6));
  Serial.println(digitalRead(X7_VT_6));
  Serial.println(digitalRead(X6_VT_6));
  while (vel > vel_2)
  { //Cuando la velocidad calculada es MENOR a 6km/h y...
    if (digitalRead(X1_VT_6) == HIGH)
    { //Si la lectura es 1 digital, se ejecuta el error 2
      digitalWrite(A0,LOW);

```

```

    digitalWrite(A1,LOW);
    digitalWrite(A2,HIGH);
    digitalWrite(A3,LOW);
    Serial.println("ERROR_012: la variable X1_VT_6 debia arrojar un UNO, pero recibio un CERO
\nERROR CONTROL DE VELCIDAD RELE 6KM/H");
    error_002();
    break;
}
else if (digitalRead(X7_VT_6) == HIGH)
{ //Si la lectura es 1 digital, se ejecuta el error 4
    digitalWrite(A0,HIGH);
    digitalWrite(A1,LOW);
    digitalWrite(A2,HIGH);
    digitalWrite(A3,LOW);
    Serial.println("ERROR_022: la variable X7_VT_6 debia arrojar un UNO, pero recibio un CERO
\nERROR CONTROL DE VELCIDAD RELE 6KM/H");
    error_004();
    break;
}
else if (digitalRead(X6_VT_6) == HIGH)
{ //Si la lectura es 1 digital, se ejecuta el error 6
    digitalWrite(A0,LOW);
    digitalWrite(A1,HIGH);
    digitalWrite(A2,HIGH);
    digitalWrite(A3,LOW);
    Serial.println("ERROR_032: la variable X6_VT_6 debia arrojar un UNO, pero recibio un CERO
\nERROR CONTROL DE VELCIDAD RELE 6KM/H");
    error_006();
    break;
}
else
{
    break;
}
break;
}
}
void lecturasEnBajada05()
{
    Serial.println(digitalRead(X1_VT_05));
    Serial.println(digitalRead(X7_VT_05));
    Serial.println(digitalRead(X6_VT_05));
    while (vel <= vel_1)
    { //Si la velocidad calculada es MENOR a 0.5KM/h y...
        if (digitalRead(X1_VT_05) == LOW)
        { //Si la lectura es 1 digital, se ejecuta el error 1
            digitalWrite(A0,HIGH);
            digitalWrite(A1,HIGH);
            digitalWrite(A2,HIGH);
            digitalWrite(A3,LOW);
            Serial.println("ERROR_013: la variable X1_VT_05 debia arrojar un CERO, pero recibio un UNO
\nERROR CONTROL DE VELCIDAD RELE 0.5KM/H");
            error_001();
            break;
        }
        else if (digitalRead(X7_VT_05) == LOW)
        { //Si la lectura es 1 digital, se ejecuta el error 3

```

```

    digitalWrite(A0,LOW);
    digitalWrite(A1,LOW);
    digitalWrite(A2,LOW);
    digitalWrite(A3,HIGH);Serial.println("ERROR_023: la variable X7_VT_05 debia arrojar un CERO,
pero recibio un UNO\nERROR CONTROL DE VELCIDAD RELE 0.5KM/H");
    error_003();
    break;
}
else if (digitalRead(X6_VT_05) == LOW)
{ //Si la lectura es 1 digital, se ejecuta el error 5
    digitalWrite(A0,HIGH);
    digitalWrite(A1,LOW);
    digitalWrite(A2,LOW);
    digitalWrite(A3,HIGH);
    Serial.println("ERROR_033: la variable X6_VT_05 debia arrojar un CERO, pero recibio un UNO
\nERROR CONTROL DE VELCIDAD RELE 0.5KM/H");
    error_005();
    break;
}
else
{
    break;
}
break;
}
}
void lecturasEnBajada6()
{
    Serial.println(digitalRead(X1_VT_6));
    Serial.println(digitalRead(X7_VT_6));
    Serial.println(digitalRead(X6_VT_6));
    while (vel <= vel_2)
    { //Si la velocidad calculada es MENOR a 6km/h y...
        if (digitalRead(X1_VT_6) == LOW)
        { //Si la lectura es 1 digital, se ejecuta el error 2
            digitalWrite(A0,LOW);
            digitalWrite(A1,HIGH);
            digitalWrite(A2,LOW);
            digitalWrite(A3,HIGH);
            Serial.println("ERROR_014: la variable X1_VT_6 debia arrojar un CERO, pero recibio un UNO
\nERROR CONTROL DE VELCIDAD RELE 6KM/H");
            error_002();
            break;
        }
        else if (digitalRead(X7_VT_6) == LOW)
        { //Si la lectura es 1 digital, se ejecuta el error 2
            digitalWrite(A0,HIGH);
            digitalWrite(A1,HIGH);
            digitalWrite(A2,LOW);
            digitalWrite(A3,HIGH);
            Serial.println("ERROR_024: la variable X7_VT_6 debia arrojar un CERO, pero recibio un UNO
\nERROR CONTROL DE VELCIDAD RELE 6KM/H");
            error_004();
            break;
        }
        else if (digitalRead(X6_VT_6) == LOW)
        { //Si la lectura es 1 digital, se ejecuta el error 2

```

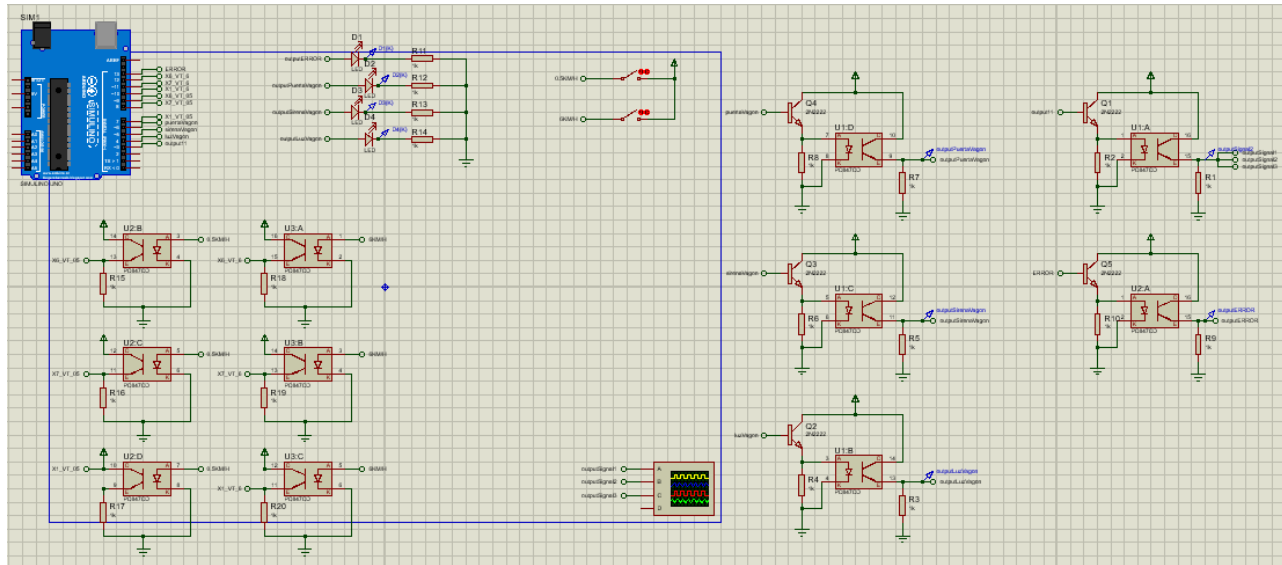
```

    digitalWrite(A0,LOW);
    digitalWrite(A1,LOW);
    digitalWrite(A2,HIGH);
    digitalWrite(A3,HIGH);
    Serial.println("ERROR_034: la variable X6_VT_6 debia arrojar un CERO, pero recibio un UNO
\nERROR CONTROL DE VELCIDAD RELE 6KM/H");
    error_006();
    break;
}
else
{
    break;
}
break;
}
}
}
//---NOTAS-----
//considerar que la grafica de velocidad que se dibuja (o comprende) es lineal, por lo tanto, su
aceleración es constante mientras que a travez
//de una ecuacion de posición en funcion del tiempo es parabolico, esto considerando el M.R.U.A.
(movimiento rectilineo uniforme acelerado).
//El tiempo aproximado desde la salida del tren hasta la llegada a la estacion toma
aproximadamente 1:09 minutos.
//Se cambio la proporcion de la frecuencia de 2.5 a 1.25 veces la frecuencia, esto debido a que el
programa toma mas tiempo en las iteraciones.
//los casos de errores evalua una tarjeta a la vez, queda pendiente versionar casos de pruebas para
multiples combinaciones de tarjetas.

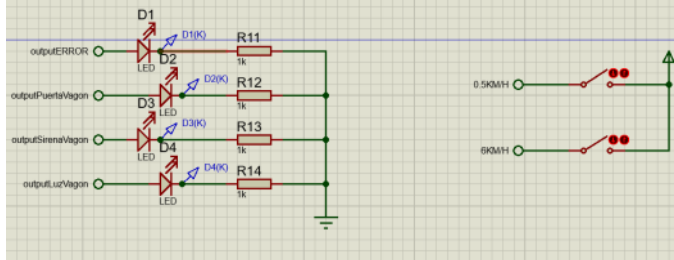
```

005 Proteus

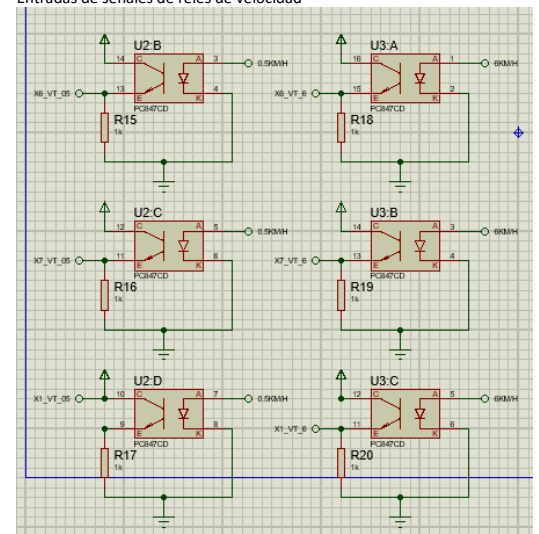
8 de abril de 2021 15:18



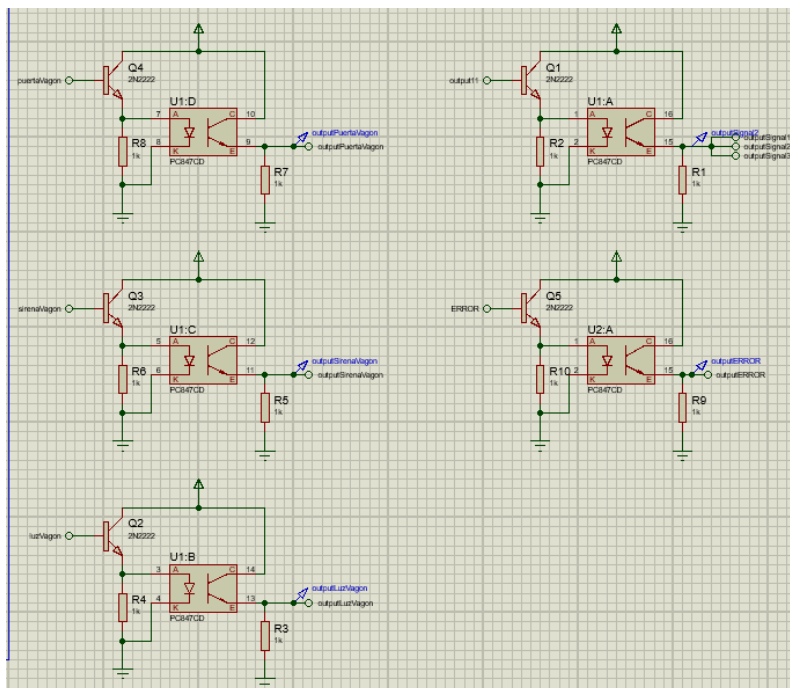
Salidas Hacia el VDR y entrada de pulsos por activación o desactivación de relés de velocidad

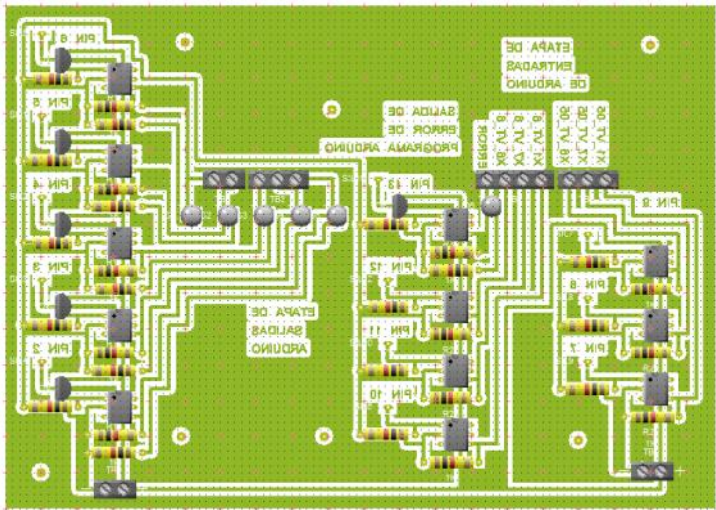
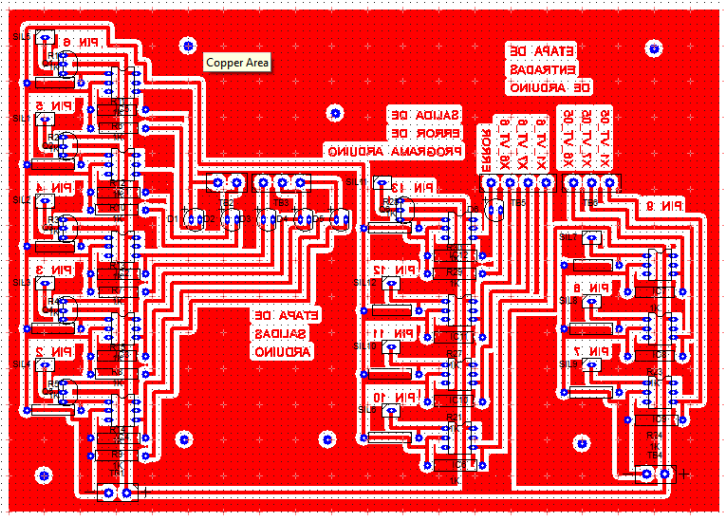


Entradas de señales de relés de velocidad

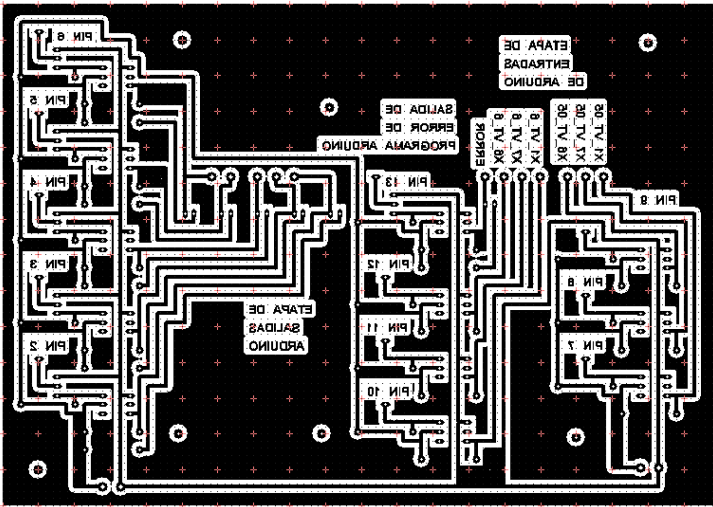
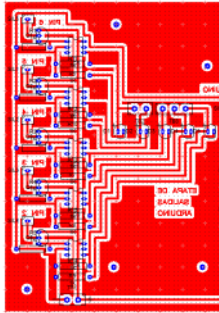


Señales de salida del Arduino

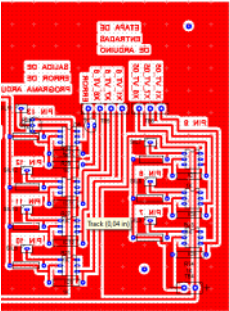




Etapa de Salida de Señales

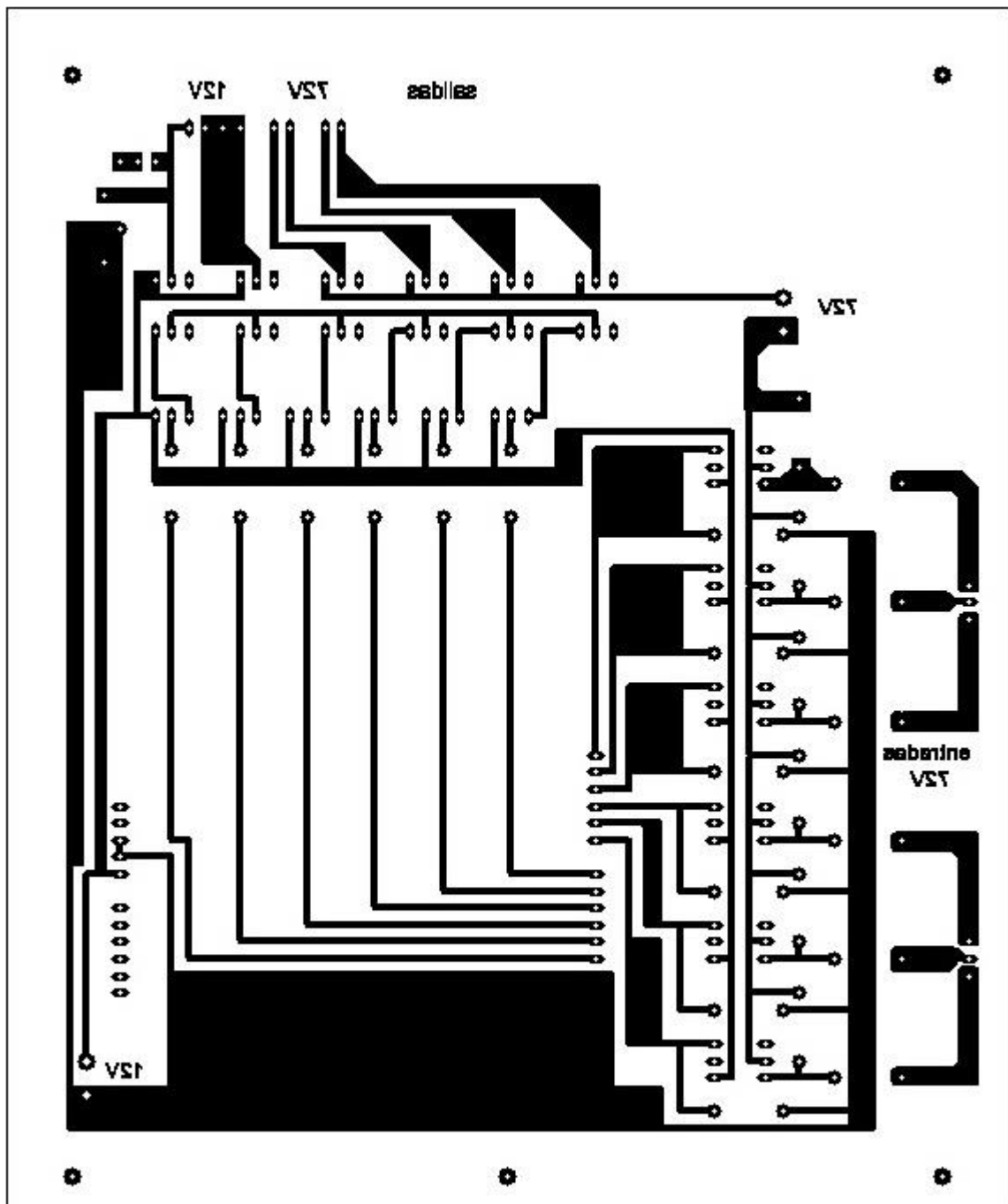


Etapa de salida y Luz Piloto de error



0061 Diseño PCB (Actual)

26 de abril de 2021 13:08



Detalle: Falta relacionar las tierras, de los optoacopladores en la etapa de salida y la tierra de la etapa de control (+12V).

007 ¿Qué hacer con el pic 16f628A?

14 de abril de 2021 11:01

Proyectar mensaje de error dependiendo de qué y en qué momento...

Estará condicionado por la respuesta del Arduino...

¿Qué respuesta necesito por parte del Arduino para saber que error proyectar?

¿Qué necesito que reciba el pic para proyectar el error?

Actividad	Estatus	Solución/problema
Programa para PIC 16F628A	ok	
Prueba Simulación Display	ok	
Carga de programa en PIC FISICO	Ok	Utilización de PICKIT3 programer para subir el programa
Prueba del sistema	Ok	El programa aunque aparentemente está cargado, no se ejecuta al energizar. Sol: problemas de alimentación, el programa se ejecuta sin problemas y responde a los estímulos establecidos
Diseño de PCB/ Impresión/ Montaje		
Prueba en conjunto Arduino		Funciona correctamente según las combinaciones provenientes del ARDUINO. En primera fase.

008 Programa PIC

16 de abril de 2021 9:36

```
#include <16F628A.h>
#use delay(clock = 4000000) /* OSCILADOR INTERNO */ //No aplica, ya que se activo el cristal interno
#FUSES NOWDT /* No Watch Dog Timer*/
//#FUSES HS /* High speed Osc (> 4mhz)*/ //Se usa con un cristal mayor al de 4Mhz, se
recomienda usar el cristal interno
#FUSES PUT /* Power Up Timer*/
#FUSES NOPROTECT /* Code not protected from reading*/
#FUSES NOBROWNOUT /* No brownout reset*/
#FUSES NOMCLR /* Master Clear pin disabled*/
#FUSES NOLVP /* No low voltage prgming, B3(PIC16) or B5(PIC18) used for I/O*/
#FUSES NOCPD
#FUSES INTRC //Activacion uso cristal interno
```

```
#include <lcd.c> /*libreria para usar el lcd*/
#define LCD_ENABLE_PIN PIN_B0 /* ASIGNAMOS EL PIN "B0" COMO SALIDA PARA E */
#define LCD_RS_PIN PIN_B1 /* ASIGNAMOS EL PIN "B1" COMO SALIDA PARA RS*/
#define LCD_RW_PIN PIN_B2 /* ASIGNAMOS EL PIN "B2" COMO SALIDA PARA RW*/
#define LCD_DATA4_PIN_B4 /* ASIGNAMOS EL PIN "B4" COMO SALIDA PARA D4*/
#define LCD_DATA5_PIN_B5 /* ASIGNAMOS EL PIN "B5" COMO SALIDA PARA D5*/
#define LCD_DATA6_PIN_B6 /* ASIGNAMOS EL PIN "B6" COMO SALIDA PARA D6*/
#define LCD_DATA7_PIN_B7 /* ASIGNAMOS EL PIN "B7" COMO SALIDA PARA D7*/
```

```
void main()
{
    output_high(pin_a4);
    delay_ms(500);
    output_low(pin_a4);
    lcd_init();
    lcd_putc("\f");
    do{
        lcd_gotoxy(1,1);
        printf(lcd_putc, "Esperando ERROR");
        lcd_gotoxy(1,2);
        printf(lcd_putc, "Registrador de Eventos");
        //funcion de arduino lecturasEnSubida05
        if (input(PIN_A0) == 1 && input(PIN_A1)==0 && input(PIN_A2)==0 && input(PIN_A3)==0)
        {
            lcd_putc("\f");
            lcd_gotoxy(1,1);
            printf(lcd_putc, "Error en:");
            lcd_gotoxy(1,2);
            printf(lcd_putc, "ERROR_011: la variable X1_VT_05");
            break;
        }
        else if (input(PIN_A0) == 0 && input(PIN_A1)==1 && input(PIN_A2)==0 && input(PIN_A3)==0)
        {
            lcd_putc("\f");
            lcd_gotoxy(1,1);
            printf(lcd_putc, "Error en:");
            lcd_gotoxy(1,2);
            printf(lcd_putc, "ERROR_021: la variable X7_VT_05");
            break;
        }
        else if (input(PIN_A0) == 1 && input(PIN_A1)==1 && input(PIN_A2)==0 && input(PIN_A3)==0)
        {
            lcd_putc("\f");
            lcd_gotoxy(1,1);
            printf(lcd_putc, "Error en:");
            lcd_gotoxy(1,2);
            printf(lcd_putc, "ERROR_031: la variable X6_VT_05");
            break;
        }
        //funcion de arduino lecturasEnSubida6
        else if (input(PIN_A0) == 0 && input(PIN_A1)==0 && input(PIN_A2)==1 && input(PIN_A3)==0)
        {
            lcd_putc("\f");
            lcd_gotoxy(1,1);
            printf(lcd_putc, "Error en:");
            lcd_gotoxy(1,2);
            printf(lcd_putc, "ERROR_012: la variable X1_VT_6");
            break;
        }
        else if (input(PIN_A0) == 1 && input(PIN_A1)==0 && input(PIN_A2)==1 && input(PIN_A3)==0)
        {
            lcd_putc("\f");
            lcd_gotoxy(1,1);
            printf(lcd_putc, "Error en:");
            lcd_gotoxy(1,2);
            printf(lcd_putc, "ERROR_022: la variable X7_VT_6");
            break;
        }
        else if (input(PIN_A0) == 0 && input(PIN_A1)==1 && input(PIN_A2)==1 && input(PIN_A3)==0)
        {
            lcd_putc("\f");
            lcd_gotoxy(1,1);
            printf(lcd_putc, "Error en:");
            lcd_gotoxy(1,2);
            printf(lcd_putc, "ERROR_032: la variable X6_VT_6");
            break;
        }
        //funcion de arduino lecturasEnBajada05
        else if (input(PIN_A0) == 1 && input(PIN_A1)==1 && input(PIN_A2)==1 && input(PIN_A3)==0)
        {
            lcd_putc("\f");
            lcd_gotoxy(1,1);
            printf(lcd_putc, "Error en:");
            lcd_gotoxy(1,2);
        }
    }
}
```

```
:1000000002308A00D22A0000
0A108A100A118207E0
:1000100028340C3401340634
0A108A100A1182077D
:10002000C3400340A108A10
0A11820745347334E4
:100030007034653472346134
6E3464346F34203417
:100040004534523452344F34
523400340A108A103A
:100050000A11820752346534
673469347334743456
:100060007234613464346F34
7234203464346534EF
:100070002034453476346534
6E3474346F347334DC
:1000800000340A108A100A11
82074534723472341F
:100090006F34723420346534
6E343A3400340A10CC
:1000A0008A100A1182074534
523452344F34523484
:1000B0005F34303431343134
3A3420346C34613488
:1000C0002034763461347234
6934613462346C348F
:1000D0006534203458343134
5F34563454345F340A
:1000E0003034353400340A10
8A100A11820745343E
:1000F000523452344F345234
5F3430343234313429
:100100003A3420346C346134
20347634613472348F
:100110006934613462346C34
653420345834373493
:100120005F34563454345F34
3034353400340A107C
:100130008A100A1182074534
523452344F345234F3
:100140005F34303433343134
3A3420346C346134F5
:100150002034763461347234
6934613462346C34FE
:100160006534203458343634
5F34563454345F3474
:100170003034353400340A10
8A100A1182074534AD
:10018000523452344F345234
5F3430343134323498
:100190003A3420346C346134
20347634613472342F
:1001A0006934613462346C34
653420345834313409
:1001B0005F34563454345F34
363400340A108A1085
:1001C0000A11E53E03188A0A
82004534523452343B
:1001D0004F3452345F343034
323432343A34203491
:1001E0006C34613420347634
61347234693461346F
:1001F00062346C3465342034
583437345F345634C8
:1002000054345F3436340034
0A148A100A118207D9
:100210004534523452344F34
52345F3430343334F2
:1002200032343A3420346C34
6134203476346134DE
:100230007234693461346234
6C3465342034583437
:1002400036345F3456345434
5F34363400340A1450
:100250008A100A1182074534
523452344F345234D2
:100260005F34303431343334
3A3420346C346134D4
:100270002034763461347234
6934613462346C34DD
:100280006534203458343734
5F34563454345F3452
:100290003034353400340A14
8A100A118207453488
:1002A000523452344F345234
5F3430343234333475
:1002B0003A3420346C346134
20347634613472340E
:1002C0006934613462346C34
6534203458343734E2
:1002D0005F34563454345F34
3034353400340A14C7
:1002E0008A100A1182074534
523452344F34523442
:1002F0005F34303433343334
3A3420346C34613442
:100300002034763461347234
6934613462346C344C
:100310006534203458343634
5F34563454345F34C2
:1003200020345F3460340A14
```

```

{
  lcd_putc("\f");
  lcd_gotoxy(1,1);
  printf(lcd_putc, "Error en:");
  lcd_gotoxy(1,2);
  printf(lcd_putc, "ERROR_013: la variable X7_VT_05");
  break;
}
else if (input(PIN_A0) == 0 && input(PIN_A1)==0 && input(PIN_A2)==0 && input(PIN_A3)==1)
{
  lcd_putc("\f");
  lcd_gotoxy(1,1);
  printf(lcd_putc, "Error en:");
  lcd_gotoxy(1,2);
  printf(lcd_putc, "ERROR_023: la variable X7_VT_05");
  break;
}
else if (input(PIN_A0) == 1 && input(PIN_A1)==0 && input(PIN_A2)==0 && input(PIN_A3)==1)
{
  lcd_putc("\f");
  lcd_gotoxy(1,1);
  printf(lcd_putc, "Error en:");
  lcd_gotoxy(1,2);
  printf(lcd_putc, "ERROR_033: la variable X6_VT_05");
  break;
}
//funcion de arduino lecturasEnBajadas6
else if (input(PIN_A0) == 0 && input(PIN_A1)==1 && input(PIN_A2)==0 && input(PIN_A3)==1)
{
  lcd_putc("\f");
  lcd_gotoxy(1,1);
  printf(lcd_putc, "Error en:");
  lcd_gotoxy(1,2);
  printf(lcd_putc, "ERROR_014: la variable X7_VT_6");
  break;
}
else if (input(PIN_A0) == 1 && input(PIN_A1)==1 && input(PIN_A2)==0 && input(PIN_A3)==1)
{
  lcd_putc("\f");
  lcd_gotoxy(1,1);
  printf(lcd_putc, "Error en:");
  lcd_gotoxy(1,2);
  printf(lcd_putc, "ERROR_024: la variable X7_VT_6");
  break;
}
else if (input(PIN_A0) == 0 && input(PIN_A1)==0 && input(PIN_A2)==1 && input(PIN_A3)==1)
{
  lcd_putc("\f");
  lcd_gotoxy(1,1);
  printf(lcd_putc, "Error en:");
  lcd_gotoxy(1,2);
  printf(lcd_putc, "ERROR_034: la variable X6_VT_6");
  break;
}
}
}while(true);
}

```

```

3A3420340C34013442
:100300002034763461347234
6934613462346C344C
:100310006534203458343634
5F34563454345F34C2
:100320003034353400340A14
8A100A1182074534F7
:10033000523452344F345234
5F34303431343434E4
:100340003A3420346C346134
20347634613472347D
:100350006934613462346C34
653420345834373451
:100360005F34563454345F34
363400340A148A10FF
:100370000A11820745345234
52344F3452345F3488
:100380003034323434343A34
20346C3461342034F0
:100390007634613472346934
613462346C34653477
:1003A0002034583437345F34
563454345F34363460
:1003B00000340A148A100A11
820745345234523428
:1003C0004F3452345F343034
333434343A3420349C
:1003D0006C34613420347634
61347234693461347D
:1003E00062346C3465342034
583436345F345634D7
:1003F00054345F3436340034
2330840083130008CF
:100400000319102A0130F800
F701F70B052AF80B41
:10041000042A4A30F700F70B
0B2A0E2A800B022A17
:1004200008002A0EF039F700
0F3006057704860021
:10043000000006141B2A0610
08000608F700060E26
:100440000F39F80008008316
061086100611831273
:1004500086100F3083160605
F0388600083120615C5
:100460000000061400001D22
7808AA0006100000F3
:10047000061400001D227808
A90006100F3083160C
:100480000605860083122A0E
F700F030F70577087C
:100490002904F8007808A900
A91B292A271C86101E
:1004A0002718861400000611
00000610280EA90067
:1004B0000F30A9052908AA00
112228080F39A90020
:1004C000AA00112208000F30
8316060586000610C8
:1004D0008610061183128610
061106100F30A30035
:1004E000FC210130A1002108
033C031C7F2A0330BA
:1004F000AA0011220530A300
FC21A10A732A0230B0
:10050000AA0011220530A300
FC21A1012108033C0F
:10051000031C932A21080420
A200A7012208A80096
:100520002322A10A862AE82A
240B972A9A2A4030F5
:10053000A5009B2AA5010130
2302A50725088038C4
:10054000A600A7012608A800
232208002208073ACF
:100550000319B42A0B3A0319
B92A063A0319C12A16
:10056000023A0319C72ACC2A
0130A300A40094221E
:10057000D12AA7010130A800
23220230A300FC21C8
:10058000D12A0130A3000230
A4009422D12AA7016D
:100590001030A8002322D12A
0130A7002208A80089
:1005A000222080003081F39
830007309F008313AC
:1005B0008316051283120516
0230A100FA30A3003B
:1005C000FC21A10BDE2A8316
051283120512632A71
:1005D000A10121080C200038
0319F22AA10AA20067
:1005E000A622E92A0130A300
A4009422A101210837
:1005F0001220A10AF700A200
A6220F302102031D3B
:10060000F72A0130A3000230
A4009422A10121089E
:100610002620A10AF700A200
A62216302102031DFF
:10062000072B831605148312
051C502B8316851483
:1006300083128518502B8316
051583120519502B2C

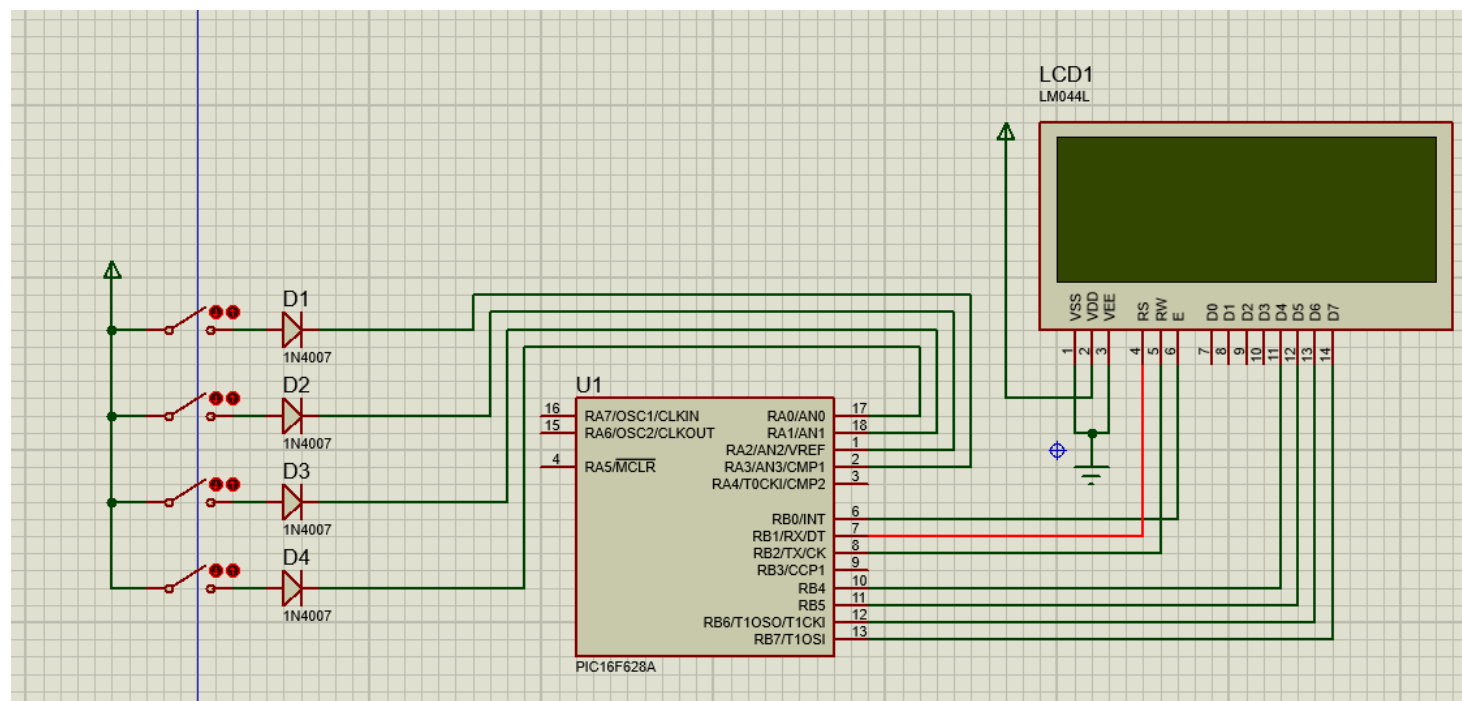
```

:100640008316851583128519
502BA10121080C20D2
:10065000003803192F2BA10A
A200A622262B013055
:10066000A300A4009422A101
21084120A10AF700BF
:10067000A200A62209302102
031D342B0130A30061
:100680000230A4009422A101
21084F20A10AF70002
:10069000A200A6221F302102
031D442B052E042E8A
:1006A0008316051483120518
8F2B83168514831265
:1006B000851C8F2B83160515
831205198F2B831626
:1006C0008515831285198F2B
A10121080C20003874
:1006D00003196E2BA10AA200
A622652B0130A300EC
:1006E000A4009422A1012108
4120A10AF700A20040
:1006F000A62209302102031D
732B0130A300023012
:10070000A4009422A1012108
7320A10AF700A200ED
:10071000A6221F302102031D
832B052E042E8316D3
:1007200005148312051CCE2B
831685148312851C99
:10073000CE2B831605158312
0519CE2B831685152E
:1007400083128519CE2BA101
21080C200038031932
:10075000AD2BA10AA200A62
2A42B0130A300A40065
:100760009422A10121084120
A10AF700A200A6229B
:1007700009302102031DB22B
0130A3000230A40076
:100780009422A10121089720
A10AF700A200A62225
:100790001F302102031DC22B
052E042E83160514C3
:1007A000831205180D2C8316
8514831285180D2CC1
:1007B000831605158312051D
0D2C831685158312CE
:1007C00085190D2CA1012108
0C2000380319EC2BF0
:1007D000A10AA200A622E32
B0130A300A4009422C8
:1007E000A10121084120A10A
F700A200A622093098
:1007F0002102031DF12B0130
A3000230A40094223A
:10080000A1012108BB20A10A
F700A200A6221E30E8
:100810002102031D012C052E
042E831605148312BC
:10082000051C4C2C83168514
831285184C2C8316BA
:1008300005158312051D4C2C
831685158312851909
:100840004C2CA10121080C20
003803192B2CA10AE3
:10085000A200A622222C0130
A300A4009422A10110
:1008600021084120A10AF700
A200A6220930210296
:10087000031D302C0130A300
0230A4009422A101FA
:100880002108DE20A10AF700
A200A6221E302102C4
:10089000031D402C052E042E
831605148312051803
:1008A0008B2C831685148312
851C8B2C83160515BF
:1008B0008312051D8B2C8316
8515831285198B2CAD
:1008C000A10121080C200038
03196A2CA10AA200FA
:1008D000A622612C0130A300
A4009422A1012108CA
:1008E0004120A10AF700A200
A62209302102031D1F
:1008F0006F2C0130A3000230
A4009422A101210832
:100900000421A10AF700A200
A6221E302102031D25
:100910007F2C052E042E8316
05148312051CCA2C69
:10092000831685148312851C
CA2C83160515831221
:10093000051DCA2C83168515
83128519CA2CA101A1
:1009400021080C2000380319
A92CA10AA200A62214
:10095000A02C0130A300A400
9422A1012108412071
:10096000A10AF700A200A622
09302102031DAE2C25
:100970000130A3000230A400
9422A1012108272104

:10098000A10AF700A200A622
1F302102031DBE2CDF
:10099000052E042E83160514
83120518092D8316BF
:1009A000851483128518092D
8316051583120519E0
:10098000092D831685158312
851D092DA101210896
:1009C0000C2000380319E82C
A10AA200A622DF2C73
:1009D0000130A300A4009422
A10121084120A10A12
:1009E000F700A200A6220930
2102031DED2C0130E0
:1009F000A3000230A4009422
A10121084B21A10AE6
:100A0000F700A200A6221F30
2102031DFD2C052E97
:100A1000042E831605148312
051C482D8316851495
:100A200083128518482D8316
051583120519482D44
:100A3000831685158312851D
482DA10121080C20E0
:100A400000380319272DA10A
A200A6221E2D01306D
:100A5000A300A4009422A101
21084120A10AF700CB
:100A6000A200A62209302102
031D2C2D0130A30073
:100A70000230A4009422A101
21086F21A10AF700ED
:100A8000A200A6221F302102
031D3C2D052E042E9C
:100A90008316051483120518
872D83168514831277
:100AA000851C872D83160515
83120519872D83163E
:100AB00085158312851D872D
A10121080C20003882
:100AC0000319662DA10AA20
0A6225D2D0130A30004
:100AD000A4009422A1012108
4120A10AF700A2004C
:100AE000A62209302102031D
6B2D0130A300023024
:100AF000A4009422A1012108
9321A10AF700A200D9
:100B0000A6221E302102031D
7B2D052E042E8316E6
:100B100005148312051CC62D
831685148312851CAB
:100B2000C62D831605158312
0519C62D8316851546
:100B30008312851DC62DA101
21080C200038031940
:100B4000A52DA10AA200A62
29C2D0130A300A4007D
:100B50009422A10121084120
A10AF700A200A622A7
:100B600009302102031DAA2
D0130A3000230A40088
:100B70009422A1012108B621
A10AF700A200A62211
:100B80001E302102031DBA2D
052E042E83160514D6
:100B900083120518042E8316
851483128518042EDB
:100BA000831605158312051D
042E831685158312E1
:100BB000851D042EA1012108
0C2000380319E42D05
:100BC000A10AA200A622DB2
D0130A300A4009422DA
:100BD000A10121084120A10A
F700A200A6220930A4
:100BE0002102031DE92D0130
A3000230A40094224C
:100BF000A1012108D921A10A
F700A200A6221E30D6
:0C0C00002102031DF92D052E
F22A6300CD
:02400E00113F60
:00000001FF
;PIC16F628A
;CRC=EC14 CREATED="16-
abr.-21 08:52"

009 Proteus PIC

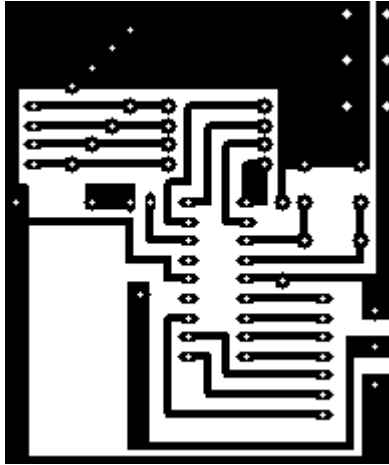
16 de abril de 2021 9:38



010 Diseño placa Display

16 de abril de 2021

11:15



Detalle: Las conexiones de b0,b1,b2 y b3 están erróneas, corregirlas si se piensa utilizar este diseño

011 Posibles errores

26 de abril de 2021 10:55

ERRORES CON LOS QUE NOS PODRIAMOS ENCONTRAR:

140	Fallo en la grabación La grabación ha fallado. <i>Remedio: re-inicializar el TELOC®1500, o descargar la configuración original con la Unidad de Servicio SG, o contactar HaslerRail o nuestro representante en su país.</i>	Aviso interno
150	Datos erróneos en la copia de seguridad en RAM La RTC Ram contiene datos erróneos debido a una batería defectuosa o a un mal arranque del TELOC®1500. <i>Causa: El condensador Gold Cap de la placa principal (MAINBB) esta descargado, o el voltaje de la batería auxiliar es demasiado bajo.</i> <i>Solución: Encender el TELOC®1500 para cargar el condensador, o cambiar la batería auxiliar.</i>	Aviso interno
301	Velocidad no es valida. Velocidad no es valida. Ninguna velocidad ha sido grabada. <i>Solución: comprobar la conexión al TELOC®1500 o la señal del generador de pulsos incluso cambia la placa principal MAINBB.</i>	externo fatal
401	Fallo en el sensor de velocidad 1 La velocidad dada por el sensor 1 es errónea. <i>Solución: comprobar el generador de pulso y cambiar el sensor dañado, número 1.</i>	Aviso externo
402	Fallo en el sensor de velocidad 2 La velocidad dada por el sensor 2 es errónea. <i>Solución: comprobar el generador de pulso y cambiar el sensor dañado, número 2.</i>	Aviso externo
403	Fallo en el sensor de velocidad 3 La velocidad dada por el sensor 3 es errónea. <i>Solución: comprobar el generador de pulso y cambiar el sensor dañado, número 3.</i>	Aviso externo
404	Fallo en el sensor de velocidad 4 La velocidad dada por el sensor 4 es errónea. <i>Solución: comprobar el generador de pulso y cambiar el sensor dañado, número 4.</i>	Aviso externo

AVISOS DE SOLUCION DE ERRORES

701	Sensor de velocidad 1 esta OK Entrada de velocidad 1 está OK <i>Acción: se borra el código 401</i>	-
702	Sensor de velocidad 2 esta OK Entrada de velocidad 2 está OK <i>Acción: se borra el código 402</i>	-
703	Sensor de velocidad 3 esta OK Entrada de velocidad 3 está OK <i>Acción: se borra el código 403</i>	-
704	Sensor de velocidad 4 esta OK Entrada de velocidad 4 está OK <i>Acción: se borra el código 404</i>	-
960	Evento desconocido, no grabado. Este evento está fuera del rango permitido. Grabación ignorada. Códigos a grabar --. <i>Acción: borrar código ---.</i>	-
961	Id del mensaje desconocida La Id del mensaje es desconocida No hacer nada. Códigos a borrar--. <i>Acción: borrar código ---.</i>	-