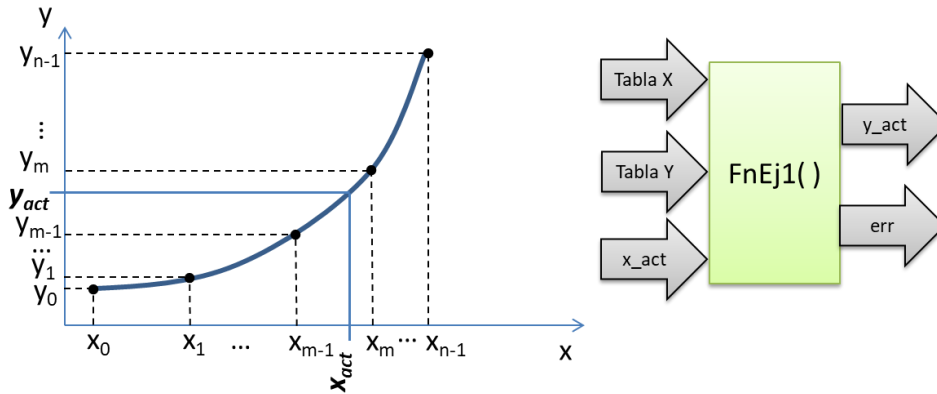


Grado en Ing^a Electrónica Industrial y Automática

Informática Industrial y Comunicaciones

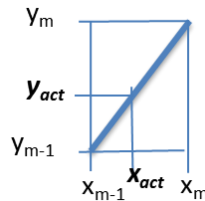
Examen Ordinario – Enero 2023

- 1) Escribir una función que, dadas dos tablas de n valores reales x_i, y_i (ambas con valores crecientes) que indican puntos en un gráfico XY, y un valor real x_{act} , devuelva el valor y_{act} que mejor aproxima la relación XY, esto es, el correspondiente a la interpolación lineal entre los valores más cercanos por debajo y por arriba.
- 2 puntos



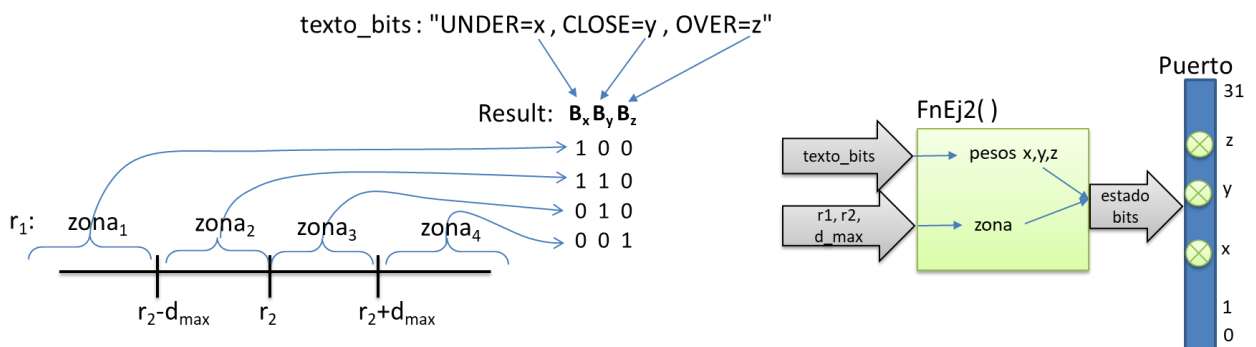
Algoritmo sugerido:

- Buscar el 1^{er} punto de la tabla x que es mayor que x_{act} y obtener su índice (m).
- Realizar la interpolación lineal con x_{act} entre los valores $x_{m-1}, x_m, y_{m-1}, y_m$ para obtener y_{act} .



La función debe devolver, además del valor real y_{act} , un entero que indique ok/error. Se produce error si: la tabla x no es estrictamente creciente, o bien $x_{act} \leq x_0$ o bien $x_{act} \geq x_{n-1}$

- 2) Realizar una función que, dados dos valores reales $r1, r2$, un valor real de diferencia máxima d_{max} , y una cadena de caracteres que indica los bits a utilizar, devuelva un entero en el cual los bits de peso x, y, z indican la posición de $r1$ respecto a $r2$, según la especificación siguiente:
- 2 punto



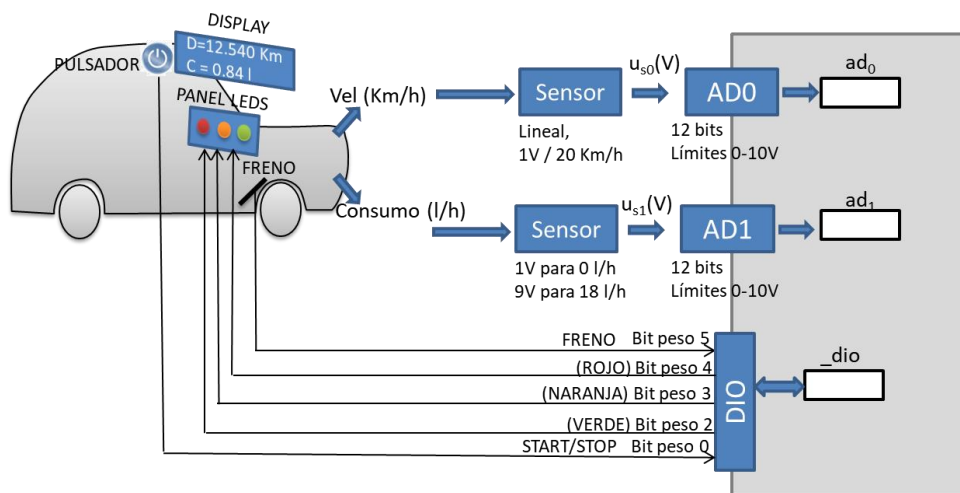
En el texto puede haber espacios antes y después del igual y de la coma, y los diferentes elementos (UNDER, CLOSE, OVER) pueden aparecer en cualquier orden. El máx. de bits es 32.

Grado en Ing^a Electrónica Industrial y Automática

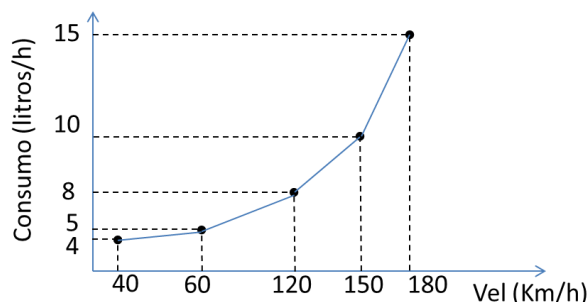
Informática Industrial y Comunicaciones

Examen Ordinario – Enero 2023

- 3) *5 puntos* En un automóvil, se miden la velocidad instantánea (km/h) y el consumo de combustible (l/h), se monitoriza una entrada digital con la pulsación del pedal de freno, y se dispone de 3 LED (verde, naranja, rojo) que deben indicar el estado del consumo instantáneo, y un display LCD que mostrará el consumo total y la distancia total:



Se espera que la relación velocidad / consumo sea como la indicada en la figura siguiente:



Mediante un pulsador con rebote, el usuario activa y desactiva el funcionamiento del sistema. El sistema está por defecto desactivado: la 1ª pulsación lo activa, la siguiente lo desactiva, la siguiente lo vuelve a activar, etc.

Cuando el sistema está activado, se desea comprobar cada 1 seg que el consumo real medido no difiere del esperado (en función de la velocidad medida) en más de 2 l/h, indicando en los LED el estado del consumo medido respecto al esperado (usando la media de los valores más recientes tanto de velocidad como de consumo):

- Ningún LED encendido: sistema desactivado
- Todos los LED encendidos: freno pisado O velocidad fuera del rango (40- 180 Km/h)
- LED Verde sólo (zona 1 ej.2): consumo muy por debajo del esperado para la velocidad actual
- LED Verde+Naranja (zona 2 ej.2): consumo algo por debajo del esperado para la velocidad actual
- LED Naranja sólo (zona 3 ej.2): consumo algo por encima del esperado para la velocidad actual

Grado en Ing^a Electrónica Industrial y Automática

Informática Industrial y Comunicaciones

Examen Ordinario – Enero 2023

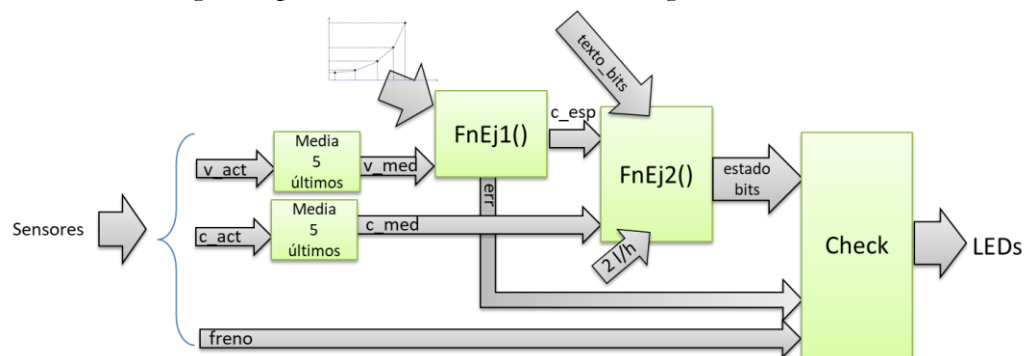
- LED Rojo sólo (zona 4 ej.2): consumo muy por encima del esperado para la velocidad actual

Para ello, se debe realizar un programa que realice las siguientes tareas:

- ❖ Pide por teclado el nº de elementos y los datos de las tablas X,Y de consumo esperado vs velocidad. No se puede presuponer un tamaño máximo para estas tablas.
- ❖ Lee de un archivo “leds.txt” una línea que indica qué LEDs deben activarse según la diferencia entre consumo real y esperado (servirá para fn ejercicio 2).

leds.txt
UNDER=2 , OVER = 4 , CLOSE=3

- ❖ Establece estado a INACTIVO.
- ❖ Arranca rutina de interrupción temporizada cada 1 seg.
- ❖ A continuación, en un bucle while infinito:
 - Si hay una pulsación START/STOP (flanco de subida), cambiar estado INACTIVO ↔ ACTIVO. Si se cambia de estado, reiniciar a 0 los valores de Distancia y Consumo Totales.
- ❖ En una función de interrupción temporizada (cada 1 seg):
 - Si estado actual es INACTIVO, apagar todos los LEDs
 - Si estado actual es ACTIVO :
 - Leer nuevo valor de velocidad (v_{act}) y consumo (c_{act}) actuales, y estado de freno
 - Actualizar valores de distancia total y consumo total, y escribir en display LCD.
 - A continuación, seguir el procesamiento indicado en la figura:



- Calcular la media de los 5 datos más recientes de velocidad (v_{med}) y de consumo (c_{med}).
- Calcular consumo esperado (c_{esp}) y error (velocidad fuera del rango 40-180Km/h) en función de la velocidad media (v_{med}) (usar función ejercicio 1).
- En función de consumo medio (c_{med}) y consumo esperado (c_{esp}): mediante $FnEj2()$ y la cadena caracteres leída del archivo, obtener el valor del nuevo estado de LEDs.
- Si estamos en funcionamiento “normal” (NO se está presionando el freno y no ha habido error en la llamada a $FnEj1()$): actualizar el puerto `_dio` para que refleje el estado de los bits actualizados por la llamada a la función.

Grado en Ing^a Electrónica Industrial y Automática

Informática Industrial y Comunicaciones

Examen Ordinario – Enero 2023

- En caso contrario (freno o error en FnEj1): actualizar el puerto `_dio` para encender todos los LEDs.

Grado en Ing^a Electrónica Industrial y Automática

Informática Industrial y Comunicaciones

Examen Ordinario – Enero 2023

Librería auxiliar para E/S específica del computador destino:

```
int LeerCanalAD(int n_canal); // Obtiene valor de canal A/D, 12 bits 0/5V
void InitTemporizador(int tm_ms,void (*FnISR)()); // Arranca temporizador
int _dio; // Variable global conectada a la E/S digital (32 bits)
void LCD_printf(int col,int row,const char* txt,...); // Escribe en LCD
```

Algunas funciones de C:

```
int atoi(const char* cad); // Devuelve entero equivalente a cadena
double atof(const char* cad); // Devuelve real equivalente a cadena
double strtod(const char* cad,char** next); // Id. a atof() y guarda en next puntero
// a final de conversión
int strlen(const char* cadena); // Devuelve longitud de cadena
char* strcpy(char* dst,const char* src); // Copia cadena fuente en destino
char* strncpy(char* dst,const char* src,int n); // Id. Máximo n caracteres
char* strcat(char* dst,const char* src); // Concatena cadena Fuente a destino
char* strncat(char* dst,const char* src,int n); // Id. Máximo n caracteres
char* strchr(const char* cad,char c); // Busca caracter en cadena, devuelve puntero
// a la primera ocurrencia o NULL si no está
char* strstr(const char* cad,const char* busca); // Id. buscando cadena
int strcmp(const char* c1,const char* c2); // Compara cadenas, devuelve 0 si iguales
int strncmp(const char* c1,const char* c2,int n); // Compara cadenas hasta max n
// caract, devuelve 0 si iguales
char* gets(char* destino); // Lee cadena de consola, almacena en destino
void* malloc(int n_bytes); // Asigna memoria para n bytes
void free(void* ptr); // Libera memoria asignada
FILE* fopen(const char* nombre,const char* modo); // Abre stream
char* fgets(char* dest,int n_max,FILE* fid); // Lee línea de stream de texto
int fscanf(FILE* fid,...); // Lee datos de stream de texto con formato
int fprintf(FILE* fid,...); // Escribe datos en stream de texto con formato
void fclose(FILE* fid); // Cierra stream
```

Grado en Ing^a Electrónica Industrial y Automática

Informática Industrial y Comunicaciones

Examen Ordinario – Enero 2023

APELLIDOS Y NOMBRE: _____

Cuestiones (responder aquí y entregar esta hoja con nombre y apellido):

(0.25 ptos Acertada, -0.15 ptos Fallada, 0 ptos No Contestada)

a) ¿Cuál de estas afirmaciones es **falsa** respecto a una salida PWM ?

- El valor instantáneo varía automáticamente entre 0 y 1 a impulsos de un reloj
- Nunca puede utilizarse para LED
- El valor promedio depende de la relación entre T_{on} y T_{total} del ciclo
- Habitualmente se debe amplificar externamente para conseguir la tensión y corriente necesaria para el accionamiento al que se conecte

b) En una comunicación RS-232:

- El **emisor** activa las señales TxD y CLK, el **receptor** comprueba las señales RxD y CLK para leer los bits (sincronizados con los pulsos del reloj).
- El **maestro** activa la señal CLK, el **emisor** activa la señal TxD y, el **receptor** comprueba las señales RxD y CLK para leer los bits (sincronizados con los pulsos del reloj).
- El **emisor** activa la señal TxD, el **receptor** comprueba la señal RxD; ambos mantienen su propio reloj (que debe ser de la misma frecuencia) y la sincronización se hace mediante el bit de **start** en la propia señal TxD.
- El **emisor** activa la señal TxD, el **receptor** comprueba la señal RxD; ambos mantienen su propio reloj (que debe ser de la misma frecuencia) y la sincronización se hace mediante las señales auxiliares **XON** y **XOFF**.

c) Una dirección IP es:

- Un valor de 32 bits que identifica a cada **equipo** conectado en una red
- Un valor de 32 bits que identifica a cada **socket** de programa conectado en una red
- Una cadena de caracteres con 4 valores enteros separados por punto que identifica a cada **equipo** conectado en una red
- Una cadena de caracteres con 4 valores enteros separados por punto que identifica a cada **socket** de programa conectado en una red

d) ¿Qué hace el código siguiente?

```
int main()
{
    char txt[40];
    printf("Escriba texto: ");
    gets(txt);
    if (txt>="HOLA")
        printf("Saludo\n");
    return 0;
}
```

- Escribe en pantalla **Saludo** si el usuario introduce por teclado un texto que **comience** por la palabra **HOLA**
- Escribe en pantalla **Saludo** si el usuario introduce por teclado cualquier texto que **contenga** la palabra **HOLA**
- Escribe en pantalla **Saludo** si el usuario introduce por teclado cualquier texto que **no contenga** la palabra **HOLA**
- Ninguna de las anteriores