

Grado en Ingeniería Electrónica y Automática

Informática Industrial y Comunicaciones

Examen Ordinario –Junio-Julio 2022

- 1) 1.5 puntos Realizar una función que, dada una tabla de valores reales, un límite inferior (valor real *lim_inf*) por debajo del cual los datos no son válidos, y un n° entero (*nc*) con el número de datos contiguos a considerar, calcule:
- Se obtiene el índice *ilst* del 1^{er} elemento de la tabla que es menor a *lim_inf* (si no hay ninguno, usar *ilst* = n° de elementos de la tabla) .
 - Si *ilst* es mayor o igual a *nc*, se calcula y devuelve la media de los valores entre 0 y *ilst*.
 - Si *ilst* es menor a *nc*, se devuelve 0.

Ejemplo:

ind \equiv 0, 1, 4, 5, 6, , 10

tabla \equiv 84, 88, 95, 99, 84, 99, 50, 36, 32, 30, 31

MiFunEj1(tabla,80,2) \rightarrow devuelve: 91.5 (media de los valores entre 0 y 5 incluidos).

MiFunEj1(tabla,85,2) \rightarrow devuelve: 0 (menos de 2 elementos al ppio mayores que 85).

- 2) 1.5 puntos Realizar una función que, dado el nombre de un archivo de texto, un valor de altura (número real), y un puntero a real para devolver el importe, escriba en pantalla el texto correspondiente a la 1^{era} línea que cumple que la altura recibido como argumento es superior al dato de dicha línea, y devuelva un valor true/false indicando si alguna lo ha cumplido.

Ejemplo de archivo: "datos.txt"

Tipo="Camion"	Hmin="2500"	Importe="60.40"
Tipo="Furgoneta"	Hmin="1500"	Importe="32.75"
Tipo="Coche"	Hmin="1000"	Importe="25.50"

Ejemplos de resultado de la función:

MiFnEj2("datos.txt",1200,&imp); \rightarrow Escribe Coche 25.50€, devuelve true, imp=25.50

MiFnEj2("datos.txt", 1800,&imp); \rightarrow Escribe Furgoneta 32.75€, devuelve true, imp=32.75

MiFnEj2("datos.txt",500,&imp); \rightarrow No escribe nada, devuelve false

Algoritmo sugerido:

```
while (línea leída de archivo es válida) {  
    extraer valor Hmin de línea  
    Si Hmin < h_parametro  $\rightarrow$  actualizar importe, cerrar archivo, devolver true  
}
```

Si se llega al fin del while \rightarrow cerrar archivo, devolver false

- 3) 1 puntos Realizar una función que, dado un entero que refleja el estado de un puerto de entrada, y dos enteros adicionales que indican los pesos de 2 bits a chequear, devuelva el valor entero:
- 0: si ambos bits del puerto son iguales
 - 1: si el bit indicado por el 1^{er} peso está a 1, y el segundo a 0
 - 1: si el bit indicado por el 1^{er} peso está a 0, y el segundo a 1

Ejemplos:

valor_entero

B ₇	B ₆	B ₅	B ₄	B ₃	B ₂	B ₁	B ₀
1	0	1	1	0	1	0	1

MiFuncionEj3(valor_entero,2,4) \rightarrow devuelve 0 (B₂ y B₄ son iguales)

MiFuncionEj3(valor_entero,6,4) \rightarrow devuelve -1 (B₆=0 y B₄=1)

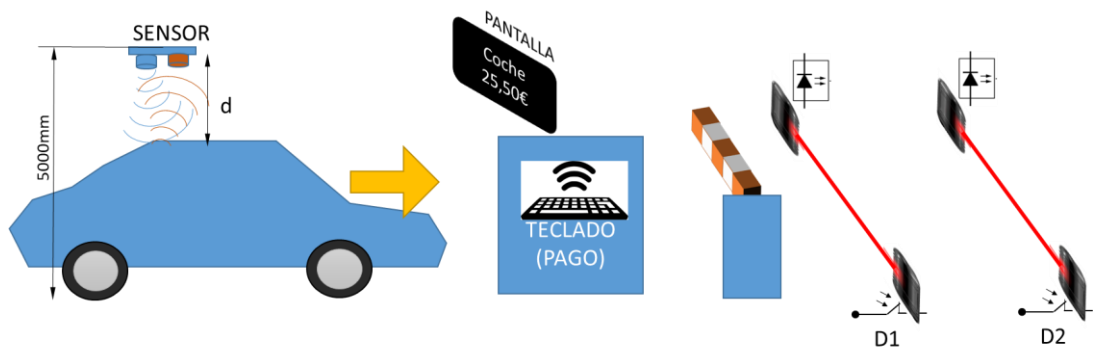
Grado en Ingeniería Electrónica y Automática

Informática Industrial y Comunicaciones

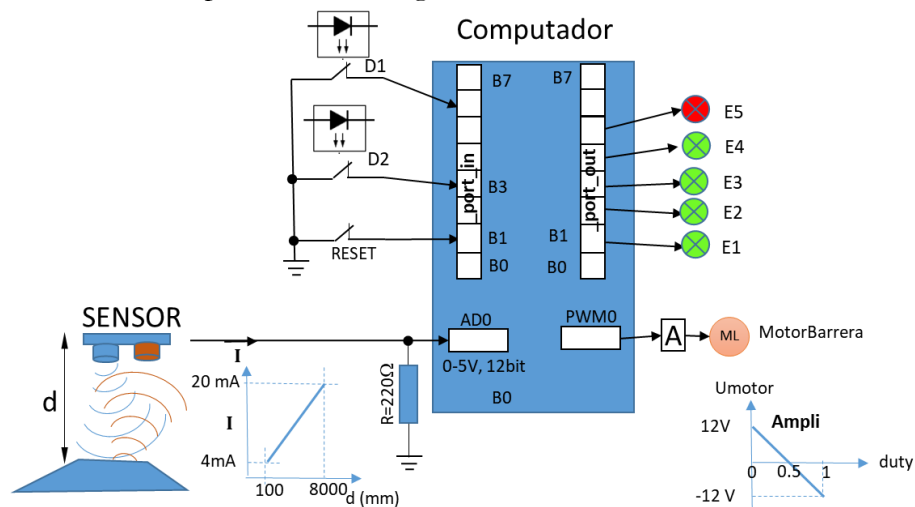
Examen Ordinario –Junio-Julio 2022

4)
5 puntos

Se desea presentar de forma automática a los vehículos que acceden a un peaje la información del importe a pagar, y activar la barrera y el semáforo cuando el usuario ha abonado el importe. Para ello, se dispone del siguiente montaje que permite medir la altura mediante un sensor ultrasónico; con el cual se detecta la presencia de un vehículo y se estima el tipo de vehículo de que se trata. Cuando el usuario efectúa el pago (que equivale a la introducción por teclado del valor del importe), si el importe satisfecho es mayor o igual al estimado, se abre la barrera hasta que el coche haya pasado completamente.



El conexionado con el computador será el siguiente:



Se debe realizar un programa para la gestión como se indica a continuación:

El estado se mostrará activando el LED de visualización correspondiente, y puede ser:

- **E1.** Sin vehículo detectado: la pantalla muestra el texto “STOP”, la barrera bajada
- **E2.** Vehículo detectado, importe no pagado: la pantalla muestra el tipo de vehículo y el importe, la barrera permanece bajada.
- **E3.** Vehículo detectado, importe pagado: la pantalla muestra el tipo de vehículo y el texto “ADELANTE”, la barrera se sube.
- **E4.** Vehículo ha superado la barrera (D2 activo y D1 inactivo) se borra la pantalla y se vuelve al estado **E1**.
- **E5.** ERROR. Tiempo vencido de apertura de barrera sin que pase el vehículo. Se escribe “ERROR” y se regresa a **E1** cuando un operario pulse el botón RESET.

El programa principal solicita el nº de datos total (n) de la tabla de medidas recientes de altura e inicializa memoria para ella, inicializa el resto de datos necesarios, establece el estado **E1**, y

Grado en Ingeniería Electrónica y Automática

Informática Industrial y Comunicaciones

Examen Ordinario –Junio-Julio 2022

arranca una interrupción temporizada (10ms) para la obtención de datos del sensor de altura, y se queda en un bucle a la espera de la entrada por teclado del importe abonado (scanf); si el estado es **E2** y el importe es igual al solicitado, se inicializa un contador de tiempo, se abre la barrera (aplicando 9V al motor) y se pasa al estado **E3**.

En la rutina de interrupción temporizada, se actúa en función del estado:

- Si el estado es **E1**, se obtiene una medida nueva de distancia del sensor, y se añade la altura correspondiente a la tabla de medidas de altura. Se considera que hay un vehículo si la altura media retornada por la función del ejercicio 1, usando n° de datos contiguos igual a $\frac{3}{4}$ de **n**, es mayor que 1000mm. En tal caso, se escribe en pantalla el importe correspondiente a esta altura media mediante la función del ejercicio 2 (archivo “importes.txt” y, si esta función devuelve true, se pasa al estado **E2** y se borran las medidas más recientes.
- Si el estado es **E2** no se hace nada (main pasará a **E3** cuando se introduzca el importe).
- Si el estado es **E3**, se incrementa el contador de tiempo y se comprueban los sensores D1 y D2, usando la función del ejercicio 3. Si D1 está inactivo y D2 está activo, se pasa al estado **E4**. Si, por el contrario, el contador de tiempo indica 20 seg, se pasa al estado **E5** (error).
- Si el estado es **E4**, se cierra la barrera (aplicando -9V al motor), se actualiza la pantalla y se vuelve al estado **E1**.
- Si el estado es **E5**, se actualiza la pantalla y se comprueba el pulsador RESET. Si el pulsador está activo, se regresa al estado **E1**.

- Se dispone de las siguientes declaraciones de funciones y variables de E/S en “io.h”:

```
void InitTemporizador(int T_ms,void (*FnCallback)() );
// Lanza una temporización por callback.
// La función callback debe declararse como: void MiFn();

int ValorAD(int n_canal,int n_bitsAD);
// Obtiene el valor de la conversión A/D del canal deseado con el n° de bits
// de digitalización indicado.

void EscribirPWM(float duty_0_a_1);
// Escribe en el PWM una señal con el duty deseado

int _port_in; // Valor del puerto de entradas digitales
int _port_out; // Valor del puerto de salidas digitales
```

Grado en Ingeniería Electrónica y Automática Informática Industrial y Comunicaciones

Examen Ordinario –Junio-Julio 2022

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 Ingeniería Electrónica y Automática

Informática Industrial y Comunicaciones

Examen Ordinario –Junio-Julio 2022

APELLIDOS Y NOMBRE: _____

5) Cuestiones:(responder aquí):

1.0 puntos (+0.25 cuestión acertada, -0.1 cuestión fallada, 0 cuestión no contestada)

a) ¿Cuál de las siguientes afirmaciones sobre una comunicación serie RS-232 es **falsa**? Marcar la respuesta apropiada.

- Se puede realizar una comunicación RS-232 únicamente con 3 hilos: emisión, recepción y masa común.
- Los dos extremos de una comunicación RS-232 deben usar la misma velocidad de reloj (baudrate).
- Se programa una comunicación RS-232 siguiendo un esquema similar a la escritura/lectura de un archivo.
- RS-232 necesita un archivo de configuración llamado “/dev/ttyS0” (en sistemas Linux).

b) Nuestro navegador de páginas web:

- Es un programa cliente TCP que se conecta con un programa servidor TCP que se ejecuta en el ordenador que indicamos en la dirección.
- Es un programa cliente DHCP que se conecta con un programa servidor DHCP en el router para obtener el puerto IP del destino.
- Ambas son ciertas: el navegador primero pide el puerto IP del destinatario al servidor DHCP del router, y después la página a dicho puerto del servidor de páginas web.
- Ninguna es cierta: el servidor de páginas web no ejecuta ningún programa, sólo las sirve.

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

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

d) ¿Cómo se puede saber el número de bits a 1 de un valor entero ? Ej: $int\ x = 01011011b;$ $\rightarrow n=5$

- Con la función `printf()` y el formato “%b”
- Usando la función `atoi()`
- Usando la función `dec2bin()`
- Ninguna de las anteriores, hay que hacer un código específico