Control de Procesos en Tiempo Real

Examen Ordinario - Septiembre 2006

Se desea desarrollar un sistema de Tiempo Real para la lectura y comprobación de valores de un conjunto de 8 canales digitales de entrada a través del puerto paralelo, mientras se genera una onda periódica en un canal analógico de salida (D/A); esta onda periódica estará formada por la suma de las senoidales compuestas por la frecuencia fundamental (armónico orden 0) y los armónicos hasta orden 7, donde la presencia o no de cada armónico es indicada en cada uno de los bits de la entrada.

Ej: Entrada = 01100101 → sumar armónicos de orden 0, 2, 5 y 6 → Genera valor = sin(2.(1+0).π.ffund.t) + sin(2.(1+2).π.ffund.t) + sin(2.(1+5).π.ffund.t) + sin(2.(1+6).π.ffund.t)

Este sistema de TR constará de dos hilos (principal y H2) y una función de activación asíncrona F1. El hilo principal hará las inicializaciones y se queda a la espera de entradas del usuario por teclado. El hilo H2 se activará periódicamente para generar la onda periódica. La función asíncrona se activará cuando haya aviso de cambio en el puerto paralelo, y procederá a su procesamiento.

1) *2.5 puntos*

Escribir el programa principal, que recibe 3 parámetros en línea de comandos: el nombre del archivo de salida, el periodo Tm de activación periódica del hilo 2 (en milisegundos) y la frecuencia base para la generación de la onda periódica. El programa principal, en este orden:

- Vacía el archivo de salida, abriéndolo en modo escritura/texto y cerrándolo inmediatamente.
- Arranca el hilo H2, pasándole como parámetro el periodo Tm de activación periódica.
- Llama a la función de inicialización de puerto paralelo, con aviso por función callback:

```
InitParPort(void (*FnAsincrona)() );
```

- Se queda en un bucle de espera a que el usuario pulse una tecla; si la tecla es 'X' termina la ejecución del programa; si es 'A', pide el valor de la amplitud de la onda; si es 'F', pide nuevo valor de la frecuencia fundamental de la onda (las variables amplitud y frecuencia deben ser accesibles desde el hilo H2); si la tecla es '?' pide un valor entero entre 0 y 7, y llama a una función que lee las líneas del archivo de salida y escribe en pantalla el nº de líneas en que está activa la frecuencia de ese orden.

2) *2 puntos*

Escribir el código de la función de activación asíncrona F1, la cual se activa cuando han cambiado los valores del puerto paralelo. Dicha función debe:

- Llamar a la función de librería `LeeParPort(unsigned char* dato);` la cual lee en la dirección indicada el valor de los bits del puerto paralelo.
- Obtiene los valores de los bits.
- Añadir al archivo de salida una nueva línea con los valores de los bits (usar el operador & para dejar el bit deseado, los operadores >> ó << para rotar los bits en la palabra).

Control de Procesos en Tiempo Real

Examen Ordinario - Septiembre 2006

---

3) *1.5 puntos*

Escribir el código correspondiente al hilo H2, que debe generar una onda periódica en un canal de salida analógico con las siguientes características:

- Recibe como parámetro en la inicialización del hilo el periodo de generación de la onda Tm.
- Es activada periódicamente mediante un mecanismo síncrono cada Tm milisegundos.
- En cada activación calcula el nuevo valor de la onda periódica, obteniendo los valores de amplitud A y frecuencia fundamental F introducidos desde el hilo principal, y el valor de los armónicos a sumar obtenido del puerto paralelo, y llama a la función de librería `EscribeDA(float valor);`

4) *2 puntos*

Describir y/o implementar cómo se corregiría el código de los hilos principal y H2 para que puedan acceder sincronizadamente a las variables **y otros recursos compartidos** mediante un mecanismo de semáforos.

5) *2 puntos*

Contestar **razonadamente** a las siguientes preguntas:

a) ¿Qué mecanismo hardware debe activar la función asíncrona F1? Explicar su funcionamiento.

b) Si se desea implantar el sistema de control anterior en un equipo empotrado con gran capacidad de procesamiento de datos, ¿qué tipo de hardware se utilizaría?

c) ¿Qué alternativas a un RTOS se podrían usar para implementar este sistema de tiempo real?

d) ¿Por qué no se pueden utilizar las funciones fopen(), fscanf(), fprintf() etc. para la lectura y escritura de los canales A/D y D/A?

Control de Procesos en Tiempo Real

Examen Ordinario - Septiembre 2006

Pase de parámetros a main() desde línea de comandos:

```
int main(int argc,char* argv[])  // argv[1] es una cadena de caracteres para el
                                 // 1er parámetro, argv[2] para el 2°, etc.
```

Documentación de funciones y estructuras POSIX a utilizar:

```
int pthread_create(pthread_t* id,pthread_attr* attr,void *(*fn)(void*),void* arg);
                   // arg es un puntero al parámetro que se quiere pasar al hilo
                   // usar attr=NULL

sem_t* sem_open(const char* name,int openflag,mode_t mode,unsigned value);
                                 // Si openflag==O_CREAT

sem_t* sem_open(const char* name,int openflag); // Si openflag==0

int sem_wait(sem_t* sem);

int sem_post(sem_t* sem);

int timer_create(clockid_t clockid,struct sigevent* sig,timer_t* id);
                                 // Usar clockid= CLOCK_MONOTONIC y sig=NULL

int timer_settime(timer_t id,int flags,const struct itimerspec* ival,
                  struct itimerspec* oval);   // Usar flags=0, oval=NULL

int sigwait(sigset_t* set,int* sig);    // usar *set=SIGRTALL para el timer

int timer_delete(timer_t id);

struct timespec
{
       time_t tv_sec;      // Segundos
       long tv_nsec;// Nanosegundos
} ;

struct itimerspec
{
       struct timespec it_interval;     // Periodo de temporización
       struct timespec it_value;        // Primera activación
} ;
```