



Estructura de Computadores

Departamento Arquitectura de Computadores

Grado en Ing. Informática, en Ing. de Computadores
e Ing. del Software



Prácticas Tema 4: E/S en Raspberry Pi

PRIMER BLOQUE: E/S CONTROLADA POR POLLING

1. Escribe un programa que encienda los dos leds verdes de la placa de expansión (señal GPIO 22, GPIO 27). Sabemos que, en el arranque van a estar todos apagados. Llámale Ejer1.s. Recuerda que ante la inexistencia de un SO al que retornar, tus programas deben terminar en un bucle infinito.
2. En el arranque, el código de Ejer2.s dejará encendidos los dos LEDs verdes. A continuación, quedará sondeando la pulsación de cualquiera de los dos botones. En función de cuál de los botones ha sido pulsado se quedará encendido sólo el LED verde del mismo lado, apagándose el otro.
3. Ahora, crea un nuevo Ejer3.s en el que se alterne el encendido y apagado del led amarillo conectado al GPIO 17. ¿Qué observas? ¿Funciona como esperabas?
4. Leyendo el contador CLO del timer para controlar el tiempo transcurrido, alterna el encendido y apagado del led rojo conectado a la señal GPIO 17 de manera que se haga visible al ojo humano (mantenlo 1 seg., en cada estado). Llámalo Ejer4.s.
5. Modifica Ejer4.s para que, en vez de actuar sobre el led, genere un tono de 523Hz (nota Do) en el altavoz, generando para ello una onda cuadrada (serie de ceros y unos consecutivos de idéntica duración) sobre el GPIO 4 y llama al programa Ejer5.
6. Escribe un programa, Ejer6.s que sondee continuamente la pulsación del botón 1 (GPIO 2) o del botón 2 (GPIO 3). Cuando se detecte la pulsación del botón 1, se encenderá el led rojo conectado al GPIO 9 y se quedará sonando la nota Do (262Hz). Si, por el contrario, se detecta el botón 2, se encenderá el led amarillo conectado al GPIO 17 y quedará sonando la nota Sol (391Hz).



Estructura de Computadores
Departamento Arquitectura de Computadores

Grado en Ing. Informática, en Ing. de Computadores
e Ing. del Software



SEGUNDO BLOQUE: E/S CONTROLADA POR INTERRUPCIÓN

7. Crea un nuevo programa, Ejer7.s, que configure el comparador C3 del timer para que transcurridos 6 segundos se produzca una IRQ cuya rutina de servicio encienda el LED asociado al GPIO 9.
8. Prepara un código, que llamarás Ejer8.s, que haga parpadear el LED asociado al GPIO 9 usando interrupciones. Para ello, la RTI deberá reprogramar la IRQ del comparador del timer para que vuelva a producirse, además de encender o apagar el led en función de lo que hizo en su última invocación. Tanto el encendido como el apagado durarán medio segundo.
9. El nuevo código Ejer9.s hará parpadear los 6 leds uno detrás de otro en secuencia, de tal manera que cada uno de ellos permanecerá encendido 400 mseg. Cuando se alcance el led del extremo se volverá a empezar por el primero, continuándose con la secuencia de encendidos y apagados de forma indefinida.
10. El código de Ejer10.s en el arranque dejará encendidos los dos LEDs verdes. La pulsación de cualquiera de los dos botones provocará una IRQ, cuyo servicio consistirá en determinar cuál de los botones ha sido pulsado y encender sólo el LED verde del mismo lado, apagándose el otro. Se trata de conseguir la misma funcionalidad que con Ejer2.s, pero utilizando en este caso interrupciones en vez de polling/sondeo.
11. En este ejercicio vamos trabajar simultáneamente con las IRQs de los comparadores C1 y C3 del timer. Con C3 controlaremos el encendido consecutivo de los 6 LEDs con una cadencia de 200msg, de forma similar a lo que hicimos en Ejer9.s: el primer led se enciende durante 200msg, pasados los cuales se apagará para encenderse el siguiente LED, y así sucesivamente (cuando se apague el sexto, volvemos a empezar con el primero). Con C1 controlaremos el altavoz para que se produzca un sonido con una frecuencia de 440Hz. Inicialmente, el programa principal programará la interrupción del timer para dentro de 200msg. La rutina de tratamiento de la IRQ, tras determinar cuál de los dos comparadores ha provocado la interrupción, lo reprogramará para que interrumpa con la periodicidad deseada.



Estructura de Computadores

Departamento Arquitectura de Computadores



Grado en Ing. Informática, en Ing. de Computadores
e Ing. del Software

12. Este ejercicio es similar al anterior pero el encendido de cada LED tendrá asociado un sonido simultáneo distinto. Para ello ahora vamos a utilizar además de la IRQ, una FIQ, de manera que las rutinas de tratamiento serán independientes.

- a. C3, que controla el encendido sucesivo de los 6 LEDs, interrumpirá con una IRQ, mientras que C1, que controla el altavoz, lo hará con una FIQ. Le damos más prioridad a C1 porque es la interrupción que se va a producir con más frecuencia. Los LEDs se mantendrán encendidos durante 500msg. La secuencia de notas que sonará será: Re Re Mi Re Sol Fa# Re Re Mi Re La Sol Re Re Re' Si Sol Fa# Mi Do' Do' Si Sol La Sol. La siguiente tabla indica las frecuencias de cada nota:

Nota	Frecuencia
La	440 Hz
Si	494 Hz
Mi	330 Hz
Sol	392 Hz
Re	293 Hz
Fa#	370Hz
Re'	587 Hz
Do'	523 Hz

- b. Además, en el programa principal se sondearán los dos botones: uno de ellos habilitará la funcionalidad descrita y el otro la deshabilitará.

PRACTICA FINAL:

El estudiante decidirá qué tipo de aplicación desarrollará y las especificaciones de la misma. En cualquier caso, contendrá al menos los siguientes elementos:

- Reproducción de una melodía
- Debe mostrar diversos patrones de encendido de los LEDs
- Uso de los dos pulsadores para modificar su funcionamiento
- Definición de una rutina de tratamiento para la IRQ y otra para la FIQ.

Al entregar los ficheros fuente (incluyendo los .inc), se añadirá un fichero de texto "Readme.txt" con una breve descripción de la aplicación y sus funcionalidades.

Ejemplo:

En cuanto se cargue el programa empezará a sonar una conocida melodía. Las notas que la componen se os proporcionan en el fichero "vader.inc", que a su vez utiliza constantes que se definen en el "notas.inc". Adicionalmente, los 6 leds de la placa deberán cambiar su patrón de encendido al ritmo de la melodía. Cuál es el patrón activo estará controlado por la pulsación de los botones de la placa. Al menos deberás programar dos patrones de encendido, asociados a cada uno de los dos botones. Un patrón consistirá en el encendido y apagado simultáneo de los 6 leds, y el otro en el encendido en secuencia de cada uno de ellos.