

	<p>Universidad de Costa Rica Escuela de ingeniería Eléctrica Experimento 7</p>	<p>EIE Escuela de Ingeniería Eléctrica</p>
<p>IE0424: Laboratorio de Circuitos Digitales II-2020</p>		

Objetivo General

Utilizar los periféricos de la tarjeta de desarrollo.

Objetivos específicos

- Investigar cómo utilizar periféricos en la tarjeta de desarrollo.
- Entender el mecanismo de interrupciones de un procesador.
- Realizar interfaz entre un periférico y un procesador sencillo.

Preguntas a responder en el anteproyecto.

- Documente cómo se habilitan las interrupciones en picorv32.
- Documente cómo se procesan las interrupciones en pivorv32.
- Investigue y documente el protocolo serial SPI.
 - ¿Qué señales forman parte de la interfaz?
 - ¿Cuál es el significado y la funcionalidad de cada una de las señales que conforman la interfaz?
 - ¿Cómo se da la comunicación entre un dos dispositivos (maestro/esclavo).
- Documente e investigue cómo leer los datos del acelerómetro presente en la tarjeta Nexys 4 DDR usando la interfaz SPI. ¿Qué registros y direcciones hay que leer para obtener la aceleración en los ejes x, y y z?
- Documente e investigue cómo configurar una interrupción del acelerómetro presente en la tarjeta Nexys 4 DDR para configurar una interrupción que se dispare solamente cuando haya 'actividad'.

Propuesta del problema.

Proceda ahora a obtener el código inicial del proyecto. Para ello, siga el siguiente enlace:

<https://classroom.github.com/g/UN-qW7J0>

De la misma forma que en los Laboratorios anteriores, autorice la aplicación, busque su nombre (si su nombre no aparece, contacte al profesor) y seleccione un equipo (si su equipo no aparece listado, proceda a crear uno). Una vez finalizado un repositorio debió haber sido creado. Este repositorio va a ser el repositorio con el que va a trabajar y va a ser uno de los entregables de este laboratorio.

Proceda a clonar el repositorio que ha sido creado:

```
$ git clone https://github.com/ucr-ie-0424/...
```

La dirección completa de su repositorio debe aparecer seleccionando el botón verde que dice *Code*.

El código que se provee es el mismo código que se usó como punto de inicio de los Laboratorios anteriores.

El objetivo de este experimento es practicar el uso de dispositivos de entrada y salida además de entender la implementación y manejo de interrupciones de un procesador sencillo basado de la arquitectura RISC-V.

Ejercicio 1:

Realice las **modificaciones necesarias** a la instancia de **picorv32** para que pueda procesar **correctamente interrupciones**.

Seguidamente, realice **cambios** en el **firmware** de forma tal que se envíe a la salida **out_byte** el **número de veces que se ha recibido una interrupción**. Llame a este **firmware** **firmware_lab7_part1.c**.

Realice una simulación donde el **testbench** inyecte **varias interrupciones en el tiempo al procesador** y muestre con **diagramas de señales** que el procesador está **manejaando correctamente las interrupciones inyectadas**.

Ejercicio 2:

Realice los cambios necesarios para **desplegar en los LEDs de 7 segmentos** los **valores de aceleración de los ejes y y z**.

Para ello, instancie en **system.v** un módulo con el nombre **accelerometer_reader**. Este módulo debe de tener 3 entradas:

- **La señal de MISO.**
- **La señal de clock.**
- **La señal de reset.**

Adicionalmente debe de tener las siguientes salidas:

- **La señal de MOSI.**

- La señal de SCLK.
- La señal de CS.
- El valor en 16 bits de la aceleración en y.
- El valor en 16 bits de la aceleración en x.

Seguidamente, realice cambios en el firmware de forma tal que el procesador esté **constantemente** leyendo los datos del **acelerómetro** y **luego enviando a desplegar** **estos datos en los LEDs de 7 segmentos**. Use los primeros 4 **dígitos en hexadecimal** para representar la aceleración en **z** y los siguientes 4 **dígitos** para representar la aceleración en **y**.

Llame a este firmware `firmware_lab7_part2.c` y use las siguientes direcciones como interfaz:

- **Escritura** a la dirección `0x10000000`: Se despliega un valor en los LEDs de **7 segmentos**.
- **Lectura** a la dirección `0x20000000`: Se lee el valor de la **aceleración** en **y** del **acelerómetro**.
- **Lectura** a la dirección `0x30000000`: Se lee el valor de la **aceleración** en **z** del **acelerómetro**.

Escriba un **testbench** de tal forma que se **emule** la respuesta del **acelerómetro** una vez que el se reciban los **estímulos** adecuados en la **interfaz de SPI**. Demuestre con **diagramas de señales** el correcto funcionamiento de la lectura de los valores del **acelerómetro**.

Luego, **implemente el diseño en la tarjeta** y suba un video a Youtube con el **resultado**. Coloque el video como **'Unlisted'** y para efectos de la **Bitácora**, coloque un **enlace al video**.

Ejercicio 3:

Configure el **acelerómetro** para que se mapee el estado de **activity** del **acelerómetro** a la interrupción **INT1**. Configúrelo de tal forma que se **dispare** el estado de **actividad** cuando se ha dado un **movimiento pronunciado** en alguno de los **ejes**.

Conecte esta interrupción a picorv32 y modifique el **firmware** del **ejercicio anterior** de forma tal que **sólo se lean los datos del acelerómetro** y se envíen a desplegar **una vez que se haya recibido la interrupción de actividad del acelerómetro**. Llame a este **firmware** `firmware_lab7_part3.c`.

Realice cualquier cambio necesario para que el módulo **accelerometer_reader** no haga transacciones de SPI hasta que se procese la interrupción por parte del **procesador** y se pida leer los datos de la aceleración. Puede llamar a este nuevo módulo: **accelerometer_reader_int**.

Escriba un **testbench** de tal forma que se **emule el disparo** de la **interrupción** por parte del **acelerómetro** y posteriormente se **emule** la **respuesta** del mismo cuando el **procesador** **lea** los datos de la **aceleración**. Demuestre con diagramas de

señales el correcto tratamiento de la interrupción y el correcto funcionamiento de la lectura de los valores del acelerómetro.

De igual forma que en el ejercicio anterior, implemente el diseño en la tarjeta y suba un video a Youtube con el resultado. Coloque el video como 'Unlisted' y para efectos de la Bitácora, coloque un enlace al video.