



Unidad Profesional Interdisciplinaria de Ingeniería
Campus Zacatecas IPN

Guiones de prácticas: Implementación de sistemas digitales

Docente:

Ramón Jaramillo Martínez

Zacatecas, Zac.

Fecha de actualización 30 de mayo de 2022

Índice

1. Pruebas del ambiente de programación	3
1.1. Objetivo	3
1.2. Material y equipo	3
1.3. Desarrollo	3
1.4. Conclusiones	3
2. Manejo de entradas y salidas digitales	4
2.1. Objetivo	4
2.2. Material y equipo	4
2.3. Desarrollo	4
2.4. Conclusiones	4
3. Uso de interrupciones	5
3.1. Objetivo	5
3.2. Material y equipo	5
3.3. Desarrollo	5
3.4. Conclusiones	5
4. Modulaci3n por ancho de pulso (PWM)	6
4.1. Objetivo	6
4.2. Material y equipo	6
4.3. Desarrollo	6
4.4. Conclusiones	6
5. Medici3n de per3odo de tiempo	7
5.1. Objetivo	7
5.2. Material y equipo	7
5.3. Desarrollo	7
5.4. Conclusiones	7
6. Convertidor anal3gico digital	8
6.1. Objetivo	8
6.2. Material y equipo	8
6.3. Desarrollo	8
6.4. Conclusiones	8

7. Comunicación UART	9
7.1. Objetivo	9
7.2. Material y equipo	9
7.3. Desarrollo	9
7.4. Conclusiones	9
8. Comunicación I2C y SPI	10
8.1. Objetivo	10
8.2. Material y equipo	10
8.3. Desarrollo	10
8.4. Conclusiones	10

1. Pruebas del ambiente de programación

1.1. Objetivo

Aprender el manejo del entorno de programación y la tarjeta de desarrollo.

1.2. Material y equipo

- Software Code Composer Studio (CCS).
- Tarjeta de desarrollo con microcontrolador MSP432P401R.

1.3. Desarrollo

Genere un código en C/C++ contemplando las siguientes características:

- Contador desde 0 a 1000.
- Utilizar un retardo crudo aproximadamente de 1 segundo.
- Encender el led rojo cuando el contador coincida con un número primo.
- El código deberá estar en un ciclo infinito.
- Demostrar el funcionamiento del código a través de imagenes donde se observe claramente el valor del contador y el valor del registro que corresponde al control de encendido del led.

1.4. Conclusiones

Generar conclusiones haciendo énfasis en los objetivos planteados al inicio de la práctica.

2. Manejo de entradas y salidas digitales

2.1. Objetivo

Configurar los puertos de entrada/salida (GPIOs) del microcontrolador utilizando el lenguaje de programación C/C++.

Analizar, diseñar e implementar una interfaz de entrada/salida, para resolver un problema específico por medio de poleo y de interrupciones.

2.2. Material y equipo

- Software Code Composer Studio (CCS).
- Tarjeta de desarrollo con microcontrolador MSP432P401R.

2.3. Desarrollo

Genere un código en C/C++ contemplando las siguientes características:

- Máquina de Mealy con 4 estados.
- Cada estado corresponde a un color del led RGB.
- La máquina de estado cambia al siguiente estado cuando el pulsador se presiona.
- El estado A, solo realiza transición al estado B.
- El estado B, solo realiza transición al estado C.
- El estado C, solo realiza transición al estado D.
- El estado D, solo realiza transición al estado A.
- El sensado del pulsador se deberá realizar con y sin interrupción con la finalidad de poder analizar el comportamiento de ambos casos.

2.4. Conclusiones

Generar conclusiones haciendo énfasis en los objetivos planteados al inicio de la práctica.

3. Uso de interrupciones

3.1. Objetivo

Analizar, diseñar e implementar una interfaz de entrada/salida, para resolver un problema específico por medio de interrupciones.

3.2. Material y equipo

- Software Code Composer Studio (CCS).
- Tarjeta de desarrollo con microcontrolador MSP432P401R.
- Generador de señales.

3.3. Desarrollo

Genere un código en C/C++ contemplando las siguientes características:

- Medir período y frecuencia.
- Utilizar interrupción por flanco de subida y flanco de bajada.
- No es necesario utilizar el módulo timer, puede utilizar un contador crudo.
- El valor de frecuencia deberá ser identificado claramente a través de un display de 7 segmentos (Considere dos dígitos).
- El código deberá estar en un ciclo infinito y deberá encontrar las limitaciones en cuanto a tiempo de muestreo y rango de operación del medidor.

3.4. Conclusiones

Generar conclusiones haciendo énfasis en los objetivos planteados al inicio de la práctica.

4. Modulación por ancho de pulso (PWM)

4.1. Objetivo

Analizar, diseñar e implementar una interfaz que permita resolver la cinemática directa de un robot manipulador de 3 grados de libertad.

4.2. Material y equipo

- Software Code Composer Studio (CCS).
- Tarjeta de desarrollo con microcontrolador MSP432P401R.
- Osciloscopio.

4.3. Desarrollo

Genere un código en C/C++ contemplando las siguientes características:

- Controlar 3 servomotores, los cuales corresponde a cada grado de libertad del robot.
- Implementar una interfaz de entrada que permita introducir el valor de los parámetros para cada servomotor (Puede ser potenciómetro, pulsadores, comunicación UART, etc).

4.4. Conclusiones

Generar conclusiones haciendo énfasis en los objetivos planteados al inicio de la práctica.

5. Medición de período de tiempo

5.1. Objetivo

Analizar, diseñar e implementar una interfaz que permita medir el período de tiempo de una señal digital utilizando el módulo timer.

5.2. Material y equipo

- Software Code Composer Studio (CCS).
- Tarjeta de desarrollo con microcontrolador MSP432P401R.
- Generador de señales.

5.3. Desarrollo

Genere un código en C/C++ contemplando las siguientes características:

- Medir período y frecuencia.
- Utilizar el módulo timer para realizar la medición del período.
- El valor de frecuencia deberá ser enviado por el puerto UART.
- Considerar la frecuencia máxima de reloj con la finalidad de obtener el rango máximo de medición.

5.4. Conclusiones

Generar conclusiones haciendo énfasis en los objetivos planteados al inicio de la práctica.

6. Convertidor analógico digital

6.1. Objetivo

Analizar, diseñar e implementar una interfaz de entrada/salida, para resolver un problema específico por medio del módulo ADC y UART.

6.2. Material y equipo

- Software Code Composer Studio (CCS).
- Tarjeta de desarrollo con microcontrolador MSP432P401R.
- Generador de señales.

6.3. Desarrollo

Genere un código en C/C++ contemplando las siguientes características:

- Adquirir una señal analógica.
- Aplicar filtro EMA, SMA y FIR.
- Filtro FIR debe ser diseñado con base a los siguientes parámetros: filtro pasa bajas, $F_c = 1200k$, método de ventanas, ventana rectangular.
- Enviar todas las señales obtenidas a la salida de los filtros por UART así como la señal original.

6.4. Conclusiones

Generar conclusiones haciendo énfasis en los objetivos planteados al inicio de la práctica.

7. Comunicación UART

7.1. Objetivo

Analizar, diseñar e implementar una interfaz de entrada/salida, para resolver un problema específico por medio del modulo UART.

7.2. Material y equipo

- Software Code Composer Studio (CCS).
- Tarjeta de desarrollo con microcontrolador MSP432P401R.

7.3. Desarrollo

Genere un código en C/C++ contemplando las siguientes características:

- Recepción de información por un puerto UART a velocidad de 9600 baudios. (Puede considerar UART0)
- Envío de dato recibido por otro puerto UART a una velocidad mayor con la finalidad de poder enviar el eco sin pérdida de información. (Puede considerar UART2)
- Justificar si es posible o no, realizar el eco utilizando el módulo AES, para recibir los datos crudos y encriptarlos antes de enviarlos por el otro módulo UART.

7.4. Conclusiones

Generar conclusiones haciendo énfasis en los objetivos planteados al inicio de la práctica.

8. Comunicación I2C y SPI

8.1. Objetivo

Analizar, diseñar e implementar una interfaz de entrada/salida, para resolver un problema específico por medio del modulo I2C y SPI.

8.2. Material y equipo

- Software Code Composer Studio (CCS).
- Tarjeta de desarrollo con microcontrolador MSP432P401R.

8.3. Desarrollo

Genere código en C/C++ contemplando las siguientes características:

- Eco a través del protocolo SPI, utilizando 3 cables.
- Los datos enviados a través del buffer deberán estar encriptados.

Genere código en C/C++ contemplando las siguientes características:

- Implementar una comunicación entre un maestro y al menos 4 esclavos.
- Los datos enviados a través del buffer deberán estar encriptados.

8.4. Conclusiones

Generar conclusiones haciendo énfasis en los objetivos planteados al inicio de la práctica.