

Especialidad Tecnólogo en Mecatrónica



CENTRO DE INGENIERIA Y DESARROLLO INDUSTRIAL

Centro de Ingeniería y Desarrollo Industrial

Proyecto Industrial Terminal

Sistema de Monitoreo Portátil

PARA OBTENER LA ESPECIALIDAD EN
“TECNÓLOGO EN MECATRÓNICA”

PRESENTA

Alumno: Fernando de Jesús González Aguado

Asesor: M. en C. Leonardo Barriga

QUERETARO, QRO.2010



000032



Especialidad Tecnólogo en Mecatrónica

INDICE:

RESUMEN	4
ANTECEDENTES	5
DEFINICIÓN DEL TEMA	6
JUSTIFICACIÓN	6
OBJETIVOS	7
FUNDAMENTOS	8
Instrumentación.....	8
Muestreo.....	10
Memoria Flash.....	11
SPI.....	13
PROCEDIMIENTO	17
Adquisición.....	18
Conversor ADC.....	18
Almacenamiento de datos.....	22
Envío SPI.....	24
Envío de datos a la computadora.....	24
Procesamiento de datos en Labview.....	29

Especialidad Tecnólogo en Mecatrónica

ANEXOS	29
CONCLUSIONES	28
BIBLIOGRAFIA	29

Especialidad Tecnólogo en Mecatrónica

Resumen

En el informe actual, se describe la metodología y desarrollo del proyecto para obtención de la especialidad de Tecnólogo en Mecatrónica, el objetivo principal fue desarrollar un sistema práctico de monitoreo portátil de temperatura así como carga.

El cual contemplaba dos sensores, respectivamente, los distintos datos eran guardados en una memoria flash, a efecto de poder más tarde llevarlos a una computadora personal y por medio de una comunicación tipo rs232, poder introducir los datos a un programa en labview, y así poder graficar los distintos datos guardados durante cierto tiempo, y así poder observar el comportamiento del sistema de monitoreo, a su vez se monto un display el cual permite ver distintos estados del sistema, como lo son el estado de los sensores, memoria y sistema en general.

Este sistema fue construido en base a un micro controlador de la familia de Texas Instruments.

Especialidad Tecnólogo en Mecatrónica

Antecedentes

Situándonos en el contexto que envuelve hoy en día al centro, debido a que es una de las instituciones de investigación y desarrollo más importantes del país, cuyo objetivo primordial es promover y apoyar la modernización tecnológica del sector productivo, mediante la investigación aplicada, el desarrollo experimental, la impartición de estudios de tipo superior en todos sus niveles y modalidades y la prestación de servicios científicos y tecnológicos, que propicien la innovación y transferencia de tecnología, impulsando la vinculación del sector industrial con el sistema educativo nacional.

Y gracias a este enfoque en particular el centro, hoy en día realiza numerosos e importantes proyectos para el sector industrial.

Estos proyectos al ser de gran precisión y robustez deben de contar con equipo lo suficientemente capaz de responder a las demandas que se les exige, es por eso que el centro se apoya de equipos que provienen de compañías líderes en el sector, las cuales desarrollan varios sistemas, en este caso, sistemas de instrumentación.

Actualmente el centro cuenta con un proyecto, en el cual el objetivo principal es monitorear cables de alta tensión, estos cuentan con un sistema principal de monitoreo el cual con ayuda de varios sensores y transductores, miden a lo largo de un tiempo estimado de 40 días, el cambio en sus diferentes variables, las cuales son llevadas a una computadora principal mediante tarjetas de adquisición de datos, para así luego graficar todos estos cambios y ver el comportamiento de estos cables a lo largo del tiempo.

Estas mediciones se llevan a cabo con sensores de marcas prestigiadas, como lo es interface, la cual permite tener lecturas estables, así como la posibilidad de ajuste de ciertos parámetros. Lo cual deja ver que al estar este tipo de sensores son confiables, al igual que las graficas que describen su comportamiento.

Aunado a todo esto, el ambiente del lugar en donde se están realizando las pruebas está controlado, esto quiere decir que las variables que se están midiendo no deberían de presentar perturbaciones en sus salidas, pues ya que de presentarse denotaría factores externos que están afectando el sistema, y por lo cual se quiere identificar este tipo de fenómenos que afectan las mediciones, para así garantizar que el sistema funciona adecuadamente.

Otro punto vital a señalar es que existen productos similares al sistema que se creó, pero estos carecen de ciertas características, como lo son expansión, guardado y dependiendo el costo es proporcional a las características:



Ilustración. Mini Registrador Instrumentado

Mini registrador de datos para la medición prolongada, pesa sólo 18 g, memoriza más de 2.000.000 valores de medición como humedad, presión, temperatura, aceleración y posición, con acumulador Li-Ion de larga duración, con un costo de 198 EUR.



Ilustración. Datta logger Instrumentado

Measurement Computing Temperature Data Loggers

8-channel Thermocouple Input Module

- 8 thermocouple input channels
- Supports J, K, R, S, T, N, E & B Type thermocouples
- Open detection feature detects broken thermocouples
- A standalone plug-and-play module which draws power from the USB cable
- Costo 329 Dolares.

Especialidad Tecnólogo en Mecatrónica

Definición del tema

Sistema portátil de Monitoreo, para celdas de carga y sensor de temperatura.

Justificación

Se ha expuesto que el centro actualmente tiene un proyecto de monitoreo, para tal proyecto recurre a equipos de alta precisión en su funcionamiento así como en sus mediciones, los cuales cubren los más rigurosos sistemas de control de calidad por lo que garantizan el éxito de sus proyectos.

Ahora bien sabemos que aun existiendo equipos tan sofisticados, el proyecto integrador puesto por el centro, teóricamente debería no presentar perturbaciones de ningún tipo en las pruebas de monitoreo que se realizan, pues aunado al equipo en sí, que se encarga del monitoreo, el ambiente en donde se encuentra tal proyecto está controlado.

Pero el objetivo de este proyecto no es precisamente garantizar precisión, sino más bien certificar que las mediciones y todo el proceso que se están realizando sean fidedignas, y que al observar el comportamiento del sistema referenciado a otro sistema similar sean similares.

Es por eso que se desarrollo un sistema integral de monitoreo, el cual su tarea principal es hacer un monitoreo el igual que el sistema principal, solo que este puesto en un micro controlador de bajo costo y energía, el cual se encarga de realizar tareas similares a las del sistema principal.

Esto para poder comparar los distintos sistemas de monitoreo y así poder certificar que las mediciones están siendo de lo más fiables posibles, ya que de caer en diferencias entre estos dos sistemas denotaría error en el proceso, pudiendo ser este el método de detección de algún sensor dañado o algún tipo de interferencia o ruido que se le este presentado al sistema principal en ese momento.

Todo esto traería grandes beneficios al centro ya que, la tarea a corto plazo sería instalar este segundo sistema de monitoreo propuesto, y poder verificar que el proceso se esté realizando de manera adecuada. Algo que podría plantearse para un plazo medio sería desarrollar la instrumentación para distintos tipos de sensores, en este caso los más utilizados por el centro en sus distintos proyectos, y como ultimo pero no menos importante cabría señalar que para un plazo mayor se pueden realizar sistemas parecidos al propuesto y así poder proponerlos o utilizarlos en proyectos de la misma índole.

Especialidad Tecnólogo en Mecatrónica

Objetivos

Objetivo General

- Desarrollar e implementar sistema integrado que permite el monitoreo a través de distintos tipos de sensores.

Objetivos Particulares

- Investigar sobre sistemas ya existente en el mercado.
- Investigar funcionamiento y operación del micro controlador MSP430 de Texas Instruments.
- Investigar el uso de registros en el micro controlador a utilizar.
- Implementar código para el micro controlador para realizar las distintas funciones a usar.
- Implementar sistema capaz de realizar el Objetivo general.

Fundamentos

Prueba de Creep

Básicamente esta prueba mide la deformación que se produce en un período cuando un material está sometido a un esfuerzo constante y a temperatura constante. Los datos obtenidos en un ensayo de fluencia se suelen presentar en un gráfico de fluencia contra tiempo con esfuerzo y temperatura constantes.

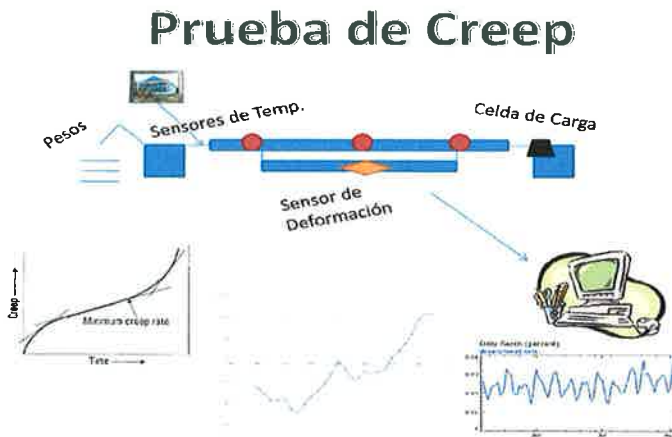


Ilustración. Prueba de Creep

Para el caso, en específico, esta prueba dura alrededor de 1000 hrs, el cual se está constantemente monitoreando tanto temperatura como carga, en cables similares a los de alta tensión, todo esto en un ambiente controlado. Y cualquier cambio abrupto, significaría cambios que en sistema ya sea por ruido o por algún factor externo que es precisamente parte del objetivo de este proyecto.

La instrumentación trata los sistemas integrados cuya finalidad es medir magnitudes físicas de un sistema externo, elaborar la información asociada a ellas y presentarla a un operador.

Las características por las que la tecnología electrónica es la más utilizada por los sistemas de instrumentación, son:

- Las señales eléctricas permiten manejar señales en un rango dinámico de tiempos muy amplio, desde los picosegundos hasta horas.
- Las señales eléctricas pueden ser transmitidas muy fácilmente a través de cables metálicos, sistemas radiados, o fibra óptica.
- Las señales eléctricas pueden ser amplificadas por circuitos electrónicos de forma muy eficientes, y pueden manejarse rangos de señal muy amplios, desde los nanos voltios hasta los kilovoltios.

Especialidad Tecnólogo en Mecatrónica

- Los sistemas electrónicos permite complejas transformaciones funcionales de las señales eléctricas.
- Las señales eléctricas son las más apropiada para ser introducidas en los computadores, los cuales representan el medio más potente de registro, transformación y presentación de la información.
- La tecnología electrónica actual es la que presenta mejor relación prestaciones/costo.

La instrumentación electrónica presenta actualmente ciertas desventajas:

- Presenta un rango de temperaturas limitado desde $-50\text{ }^{\circ}\text{C}$ hasta $175\text{ }^{\circ}\text{C}$.
- Son equipos sensibles a la radiación de alta energía.
- Requiere una fuente de potencia para su operación.
- Los componentes electrónicos activos suelen presentar deriva por envejecimiento.

En la figura 1 se muestra el esquema básico de cualquier sistema de instrumentación.

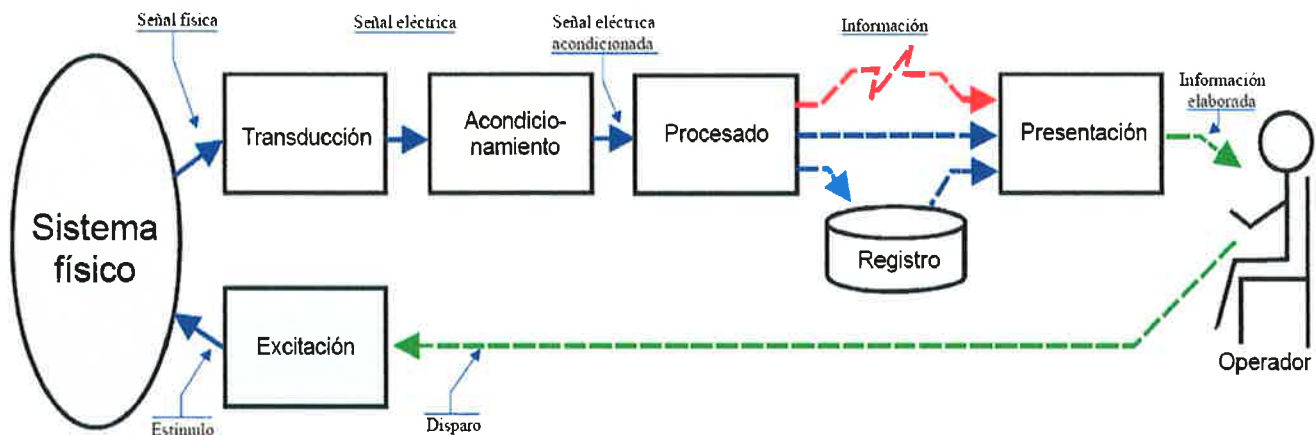


Ilustración 1. Sistema de Instrumentación

a) Transductor

El transductor es el componente que convierte la magnitud física a medir, en una señal eléctrica.

Ahora bien, como la base de este proyecto fue la programación del sistema de monitoreo, hay que mencionar antes varios conceptos:

La digitalización o conversión analógica-digital (conversión A/D)

Especialidad Tecnólogo en Mecatrónica

consiste básicamente en realizar de forma periódica medidas de la amplitud (tensión) de una señal (por ejemplo, la que proviene de un micrófono si se trata de registrar sonidos, de un sismógrafo si se trata de registrar vibraciones o de una sonda de un osciloscopio para cualquier nivel variable de tensión de interés), redondear sus valores a un conjunto finito de niveles preestablecidos de tensión (conocidos como niveles de cuantificación) y registrarlos como números enteros en cualquier tipo de memoria o soporte. La conversión A/D también es conocida por el acrónimo inglés ADC (analogue to digital converter).

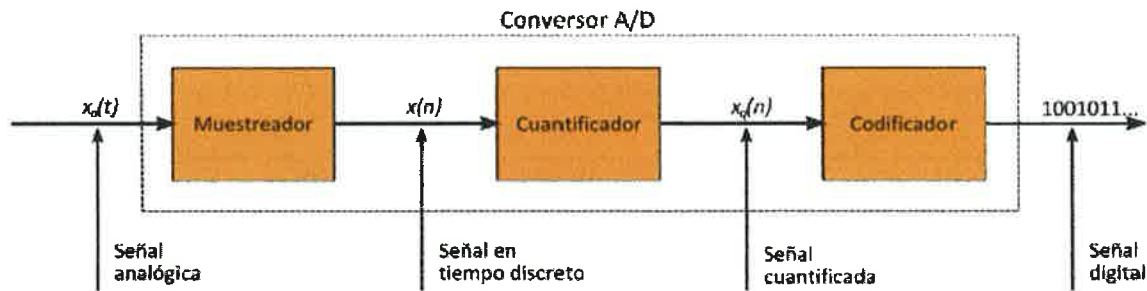


Ilustración 2. Conversor Analógico Digital

En esta definición están patentes los cuatro procesos que intervienen en la conversión analógica-digital:

1. **Muestreo:** el muestreo (en inglés, *sampling*) consiste en tomar muestras periódicas de la amplitud de onda. La velocidad con que se toma esta muestra, es decir, el número de muestras por segundo, es lo que se conoce como frecuencia de muestreo.
2. **Retención** (en inglés, *hold*): las muestras tomadas han de ser retenidas (retención) por un circuito de retención (hold), el tiempo suficiente para permitir evaluar su nivel (cuantificación). Desde el punto de vista matemático este proceso no se contempla, ya que se trata de un recurso técnico debido a limitaciones prácticas, y carece, por tanto, de modelo matemático.
3. **Cuantificación:** en el proceso de cuantificación se mide el nivel de voltaje de cada una de las muestras. Consiste en asignar un margen de valor de una señal analizada a un único nivel de salida. Incluso en su versión ideal, añade, como resultado, una señal indeseada a la señal de entrada: el ruido de cuantificación.
4. **Codificación:** la codificación consiste en traducir los valores obtenidos durante la cuantificación al código binario. Hay que tener presente que el código binario es el más utilizado, pero también existen otros tipos de códigos que también son utilizados.

Durante el muestreo y la retención, la señal aún es analógica, puesto que aún puede tomar cualquier valor. No obstante, a partir de la cuantificación, cuando la señal ya toma valores finitos, la señal ya es digital.

¿Por qué digitalizar?

Especialidad Tecnólogo en Mecatrónica

Ventajas de la señal digital

1. Cuando una señal digital es atenuada o experimenta perturbaciones leves, puede ser reconstruida y amplificada mediante sistemas de regeneración de señales.
2. Cuenta con sistemas de detección y corrección de errores, que se utilizan cuando la señal llega al receptor; entonces comprueban (uso de redundancia) la señal, primero para detectar algún error, y, algunos sistemas, pueden luego corregir alguno o todos los errores detectados previamente.
3. Facilidad para el procesamiento de la señal. Cualquier operación es fácilmente realizable a través de cualquier software de edición o procesamiento de señal.
4. La señal digital permite la multigeneración infinita sin pérdidas de calidad.
5. Es posible aplicar técnicas de compresión de datos sin pérdidas o técnicas de compresión con pérdidas basados en la codificación perceptual mucho más eficientes que con señales analógicas.

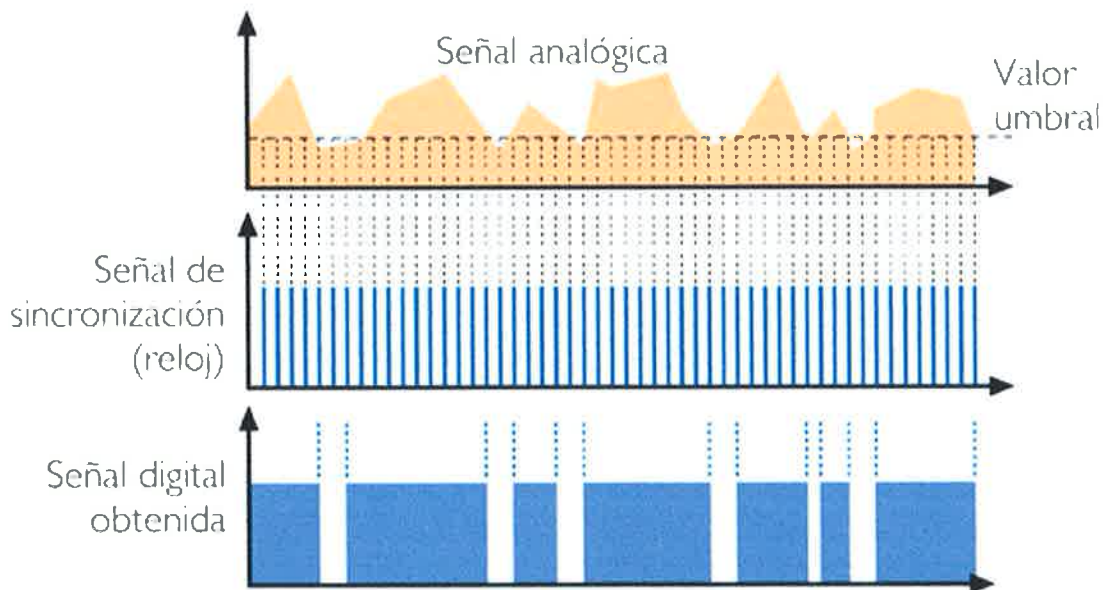


Ilustración 3. Señal de un ADC (Convertidor Analógico Digital).

Por otra parte el proyecto contemplo el uso de una memoria flash, esto para guardar los datos, entonces veamos que es una memoria flash, sus características y comparación.

Memoria Flash

Se trata de una memoria no volátil, de bajo consumo, que se puede escribir y borrar. Funciona como una ROM y una RAM pero consume menos y es más pequeña.

Especialidad Tecnólogo en Mecatrónica

Flash, como tipo de EEPROM que es, contiene una matriz de celdas con un transistor evolucionado con dos puertas en cada intersección. Tradicionalmente sólo almacenan un bit de información. Las nuevas memorias flash, llamadas también dispositivos de celdas multi-nivel, pueden almacenar más de un bit por celda variando el número de electrones que almacenan.

Estas memorias están basadas en el transistor FAMOS (Floating Gate Avalanche-Injection Metal Oxide Semiconductor) que es, esencialmente, un transistor NMOS con un conductor (basado en un óxido metálico) adicional localizado o entre la puerta de control (CG – Control Gate) y los terminales fuente/drenador contenidos en otra puerta (FG – Floating Gate) o alrededor de la FG conteniendo los electrones que almacenan la información.

Memoria flash de tipo NOR

En las memorias flash de tipo NOR, cuando los electrones se encuentran en FG, modifican (prácticamente anulan) el campo eléctrico que generaría CG en caso de estar activo. De esta forma, dependiendo de si la celda está a 1 ó a 0, el campo eléctrico de la celda existe o no. Entonces, cuando se lee la celda poniendo un determinado voltaje en CG, la corriente eléctrica fluye o no en función del voltaje almacenado en la celda. La presencia/ausencia de corriente se detecta e interpreta como un 1 ó un 0, reproduciendo así el dato almacenado. En los dispositivos de celda multi-nivel, se detecta la intensidad de la corriente para controlar el número de electrones almacenados en FG e interpretarlos adecuadamente.

Para programar una celda de tipo NOR (asignar un valor determinado) se permite el paso de la corriente desde el terminal fuente al terminal sumidero, entonces se coloca en CG un voltaje alto para absorber los electrones y retenerlos en el campo eléctrico que genera. Este proceso se llama hot-electrón injection. Para borrar (poner a "1", el estado natural del transistor) el contenido de una celda, expulsar estos electrones, se emplea la técnica de Fowler-Nordheim tunnelling, un proceso de tunelado mecánico – cuántico. Esto es, aplicar un voltaje inverso bastante alto al empleado para atraer a los electrones, convirtiendo al transistor en una pistola de electrones que permite, abriendo el terminal sumidero, que los electrones abandonen el mismo. Este proceso es el que provoca el deterioro de las celdas, al aplicar sobre un conductor tan delgado un voltaje tan alto.

Es necesario destacar que las memorias flash están subdivididas en bloques (en ocasiones llamados sectores) y por lo tanto, para el borrado, se limpian bloques enteros para agilizar el proceso, ya que es la parte más lenta del proceso. Por esta razón, las memorias flash son mucho más rápidas que las EEPROM convencionales, ya que borran byte a byte. No obstante, para reescribir un dato es necesario limpiar el bloque primero para después reescribir su contenido.

Memorias flash de tipo NAND

Las memorias flash basadas en puertas lógicas NAND funcionan de forma ligeramente diferente: usan un túnel de inyección para la escritura y para el borrado un túnel de 'soltado'. Las memorias basadas en NAND tienen, además de la evidente base en otro tipo de puertas, un coste bastante inferior, unas diez veces de más resistencia a las operaciones pero sólo permiten acceso secuencial (más orientado a dispositivos de almacenamiento masivo), frente a las memorias flash basadas en NOR que permiten lectura de acceso aleatorio. Sin embargo, han sido las NAND las que han permitido la expansión de este tipo de memoria, ya que el mecanismo de borrado es más sencillo (aunque también se borre por bloques) lo que ha proporcionado una base más rentable

Especialidad Tecnólogo en Mecatrónica

para la creación de dispositivos de tipo tarjeta de memoria. Las populares memorias USB o también llamadas Pendrives, utilizan memorias flash de tipo NAND.

Comparación de memorias flash basadas en NOR y NAND

Para comparar estos tipos de memoria se consideran los diferentes aspectos de las memorias tradicionalmente valorados.

- La densidad de almacenamiento de los chips es actualmente bastante mayor en las memorias NAND.
- El coste de NOR es mucho mayor.
- El acceso NOR es aleatorio para lectura y orientado a bloques para su modificación. Sin embargo, NAND ofrece tan solo acceso directo para los bloques y lectura secuencial dentro de los mismos.
- En la escritura de NOR podemos llegar a modificar un solo bit. Esto destaca con la limitada reprogramación de las NAND que deben modificar bloques o palabras completas.
- La velocidad de lectura es muy superior en NOR (50-100 ns) frente a NAND (10 μ s de la búsqueda de la página + 50 ns por byte).
- La velocidad de escritura para NOR es de 5 μ s por byte frente a 200 μ s por página en NAND.
- La velocidad de borrado para NOR es de 1 s por bloque de 64 KB frente a los 2 ms por bloque de 16 KB en NAND.
- La fiabilidad de los dispositivos basados en NOR es realmente muy alta, es relativamente inmune a la corrupción de datos y tampoco tiene bloques erróneos frente a la escasa fiabilidad de los sistemas NAND que requieren corrección de datos y existe la posibilidad de que queden bloques marcados como erróneos e inservibles.

En resumen, los sistemas basados en NAND son más baratos y rápidos pero carecen de una fiabilidad que los haga eficientes, lo que demuestra la necesidad imperiosa de un buen sistema de archivos. Dependiendo de qué sea lo que se busque, merecerá la pena decantarse por uno u otro tipo.

Ahora viene el protocolo de comunicación por el cual se transfieren los datos micro controlador.

SPI

Spi es un bus de tres líneas, sobre el cual se transmiten paquetes de información de 8 bits. Cada una de estas tres líneas porta la información entre los diferentes dispositivos conectados al bus. Cada dispositivo conectado al bus puede actuar como transmisor y receptor al mismo tiempo, por lo que este tipo de comunicación serial es full duplex. Dos de estas líneas transfieren los datos (una en cada dirección) y la tercer línea es la del reloj.

Algunos dispositivos solo pueden ser transmisores y otros solo receptores, generalmente un dispositivo que tramite datos también puede recibir.

Un ejemplo podría ser un memoria EEPROM, el cual es un dispositivo que puede transmitir y recibir información.

Los dispositivos conectados al bus son definidos como maestros y esclavos. Un maestro es aquel que inicia la transferencia de información sobre el bus y genera las señales de reloj y control.

Un esclavo es un dispositivo controlado por el maestro. Cada esclavo es controlado sobre el bus a través de una línea selectora llamada **Chip Select** o **Select Slave**, por lo tanto es esclavo es

Especialidad Tecnólogo en Mecatrónica

activado solo cuando esta línea es seleccionada. Generalmente una línea de selección es dedicada para cada esclavo.

En un tiempo determinado T1, solo podrá existir un maestro sobre el bus. Cualquier dispositivo esclavo que no este seleccionado, debe deshabilitarse (ponerlo en alta impedancia) a través de la línea selectora (**chip select**).

El bus SPI emplea un simple registro de desplazamiento para transmitir la información.

Especificaciones del Bus

Todas las líneas del bus transmiten la información sobre una sola dirección.

La señal sobre la línea de reloj (SCLK) es generada por el maestro y sincroniza la transferencia de datos.

La línea MOSI (Master Out Slave In) transporta los datos del maestro hacia el esclavo.

La línea MISO (Master In Slave Out) transporta los datos del esclavo hacia el maestro.

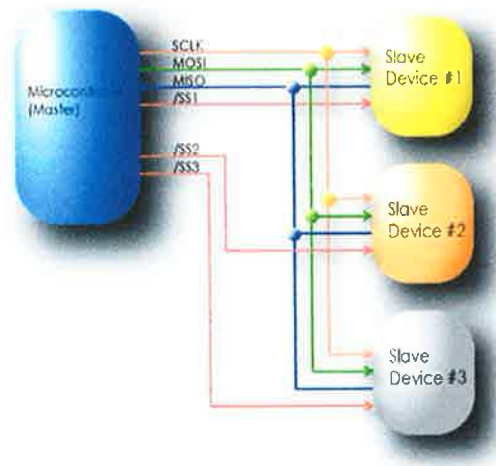


Ilustración 4. Bus SPI

Cada esclavo es seleccionado por un nivel lógico bajo ('0') a través de la línea (CS = Chip Select o SS Slave Select). Los datos sobre este bus pueden ser transmitidos a una razón de casi cero bits/segundo hasta 1 Mbits/segundo. Los datos son transferidos en bloques de 8 bits, en donde el bit más significativo (MSB) se transmite primero.

Modos del Reloj

Todos la transferencia de los datos, son sincronizados por la línea de reloj de este bus. Un BIT es transferido por cada ciclo de reloj.

La mayoría de las interfaces SPI tienen 2 bits de configuración, llamados CPOL (Clock Polarity = Polaridad de Reloj) y CPHA (Clock Phase = Reloj de Fase). CPOL determina si el estado Idle de la línea de reloj esta en bajo (CPOL=0) o si se encuentra en un estado alto (CPOL=1). CPHA determina en que filo de reloj los datos son desplazados hacia dentro o hacia fuera.

(Si CPHA=0 los datos sobre la línea MOSI son detectados cada filo de bajada y los datos sobre la línea MISO son detectados cada filo de subida).

Cada BIT tiene 2 estados, lo cual permite 4 diferentes combinaciones, las cuales son incompatibles una de la otra. Por lo que si dos dispositivos SPI desean comunicarse entre sí, estos deben tener el mismo la misma Polaridad de Reloj (CPOL) y la misma Fase de Reloj (CPHA)

Especialidad Tecnólogo en Mecatrónica

Existen cuatro modos de reloj definidos por el protocolo SPI, estos modos son:

- Modo A
- Modo B
- Modo C
- Modo D

Estos determinan el valor de la polaridad del reloj (CPOL = Clock Polarity) y el bit de fase del reloj (CPHA = Clock Phase). La mayoría de los dispositivos SPI pueden soportar al menos 2 modos de los 4 antes mencionados.

Los diferentes modos son ilustrados a continuación. El BIT de Polaridad del reloj determina el nivel del estado de Idle del reloj y el BIT de Fase de reloj determina que flanco recibe un nuevo dato sobre el bus. El modo requerido para una determinada aplicación, esta dado por el dispositivo esclavo. La capacidad de multi-modo combinada con un simple registro de desplazamiento hace que el bus SPI sea muy versátil. Polaridad del reloj (CPOL=Clock Polarity) Si CPOL está en un 0 lógico y ningún dato está siendo transferido (Estado Idle), el maestro mantiene la línea SCLK en bajo. Si CPOL está en un 1 lógico, el maestro desocupa la línea SCLK alta. Fase Del Reloj (CPHA) CPHA, conjuntamente con CPOL, controlan cuando los nuevos datos son colocados en el bus. Si CPHA es igual a un '1' lógico, los datos son desplazados sobre la línea MOSI según lo determinado por el valor de CPOL.

Si CPOL = 0, los nuevos datos se ponen en la línea cuando el flanco del reloj es ascendente y se leen cuando el reloj tiene un flanco descendente.

Si CPHA = 0, el reloj de cambio es la OR de SCLK con la terminal Chip Select.

Tan pronto como el terminal Chip Select se coloca en un nivel lógico 0, los nuevos datos se ponen en la línea y el primer filo del reloj se leen los datos. Si CPOL se activa a un nivel lógico '1', el primer borde de reloj baja y los bits de datos subsecuentes se leen en cada filo de bajada sobre la línea de reloj.

Cada nuevo bit se pone en la línea cuando el reloj tiene un flanco ascendente de Reloj. Si CPOL es cero, el primer filo de reloj ascendente y los bits de datos subsecuentes se leen en cada filo ascendente de reloj. Cada nuevo bit se coloca en la línea cuando el filo del reloj baja.

En resumen, Si CPHA=1, la transferencia (datos válidos leídos por el receptor) comienza en el segundo filo de reloj.

Si CPHA=0, la transferencia comienza en el primer filo de reloj.

Todas las transferencias subsecuentes dentro del byte ocurren en cada filo de reloj.

Véase las siguientes figuras, en todos los casos, los datos se leen a la mitad del ciclo de reloj después de que se ponen en la línea de datos.

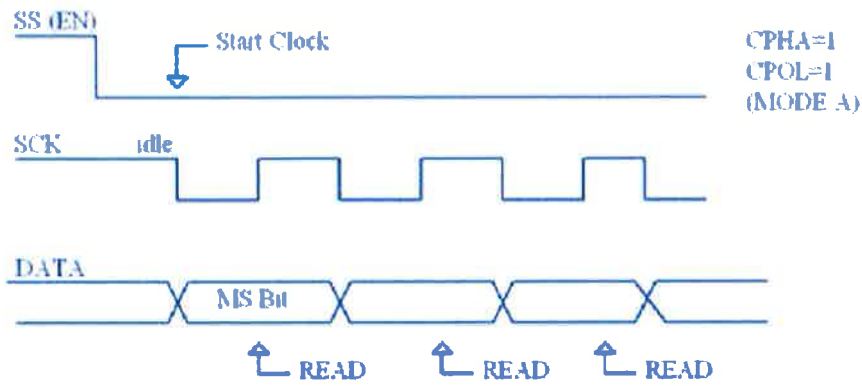


Ilustración 5. Modo A

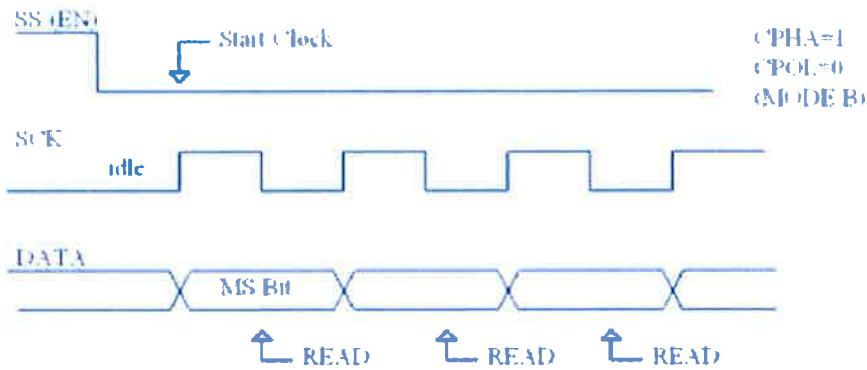


Ilustración 6. Modo B

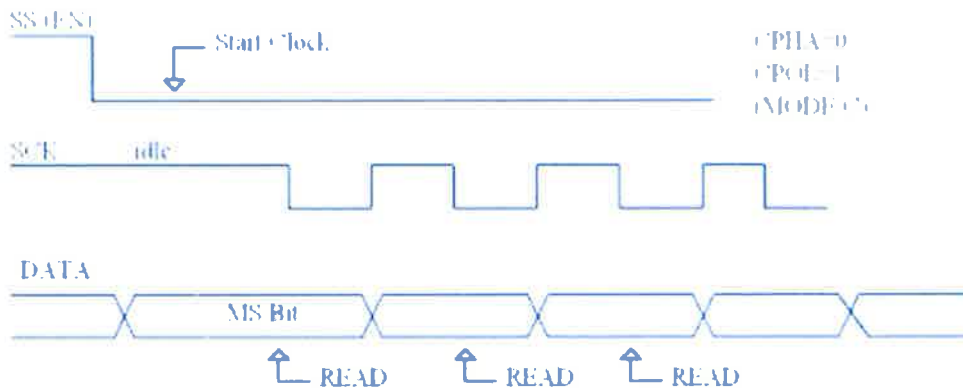
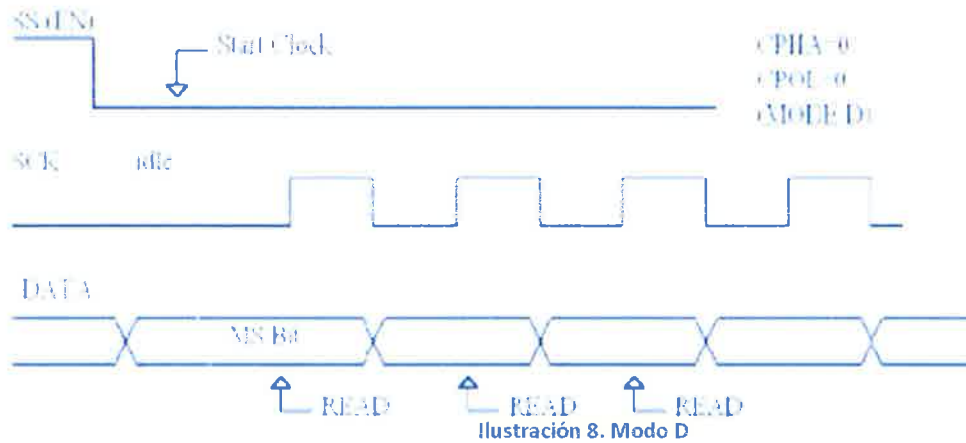


Ilustración 7. Modo C

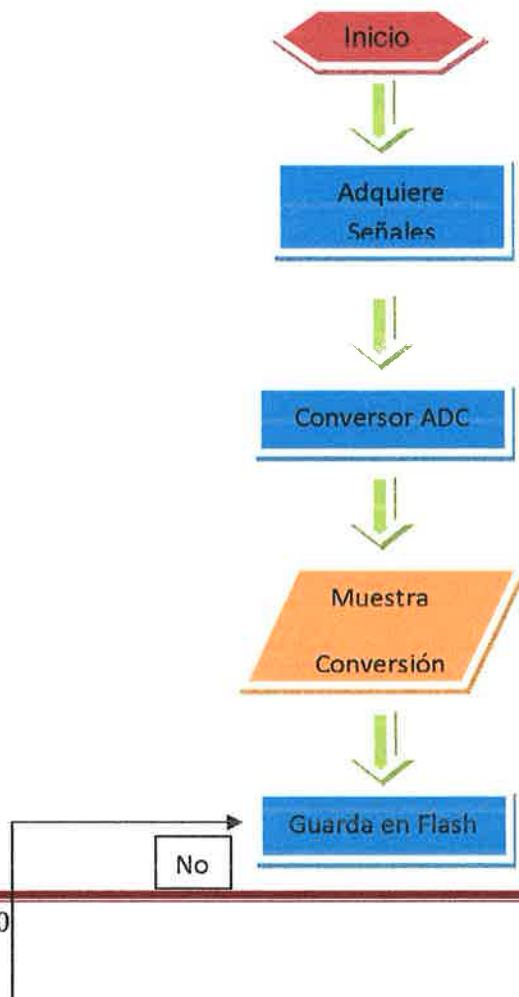
Especialidad Tecnólogo en Mecatrónica

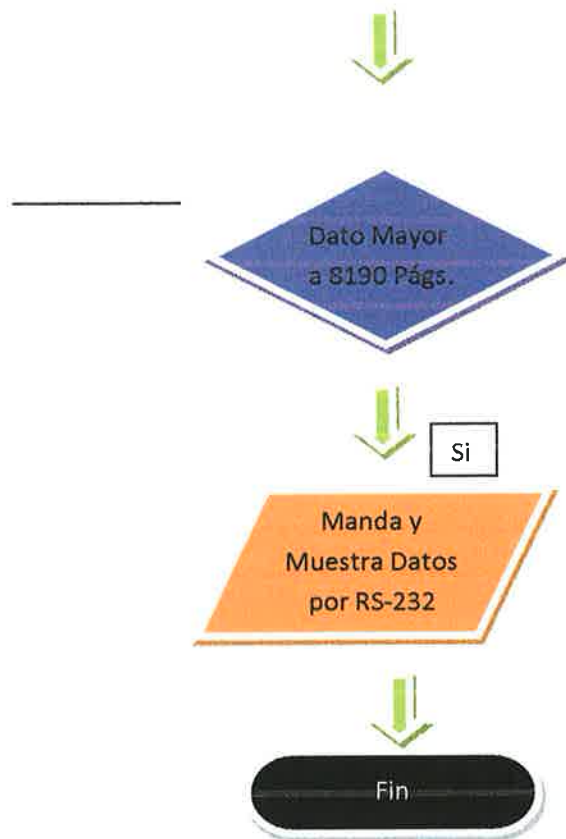


Procedimiento

Primeramente se seleccione el micro controlador con el que se va a trabajar, el cual fue un MSP 430, construido con una CPU de 16 bits, está diseñado para aplicaciones empotradas de bajo costo y bajo consumo de energía, además de sus grandes prestaciones, características y disponibilidad, y en específico se selecciono el MSP4302619.

Ya que el proyecto fue meramente programación, existen diversas partes que conforman el programa principal el cual sigue la siguiente dinámica:





Adquisición de Señales

Básicamente de este proceso el encargado fue el micro controlador, ya que para fines de autonomía y además por sus características de bajo consumo de corriente es ideal para hacer esta parte de adquisición, ya que también cuenta con 8 canales para usarlos ya sea como ADC's o bien DAC's ya que la configuración del micro permite realizar estos cambios por medio de software.

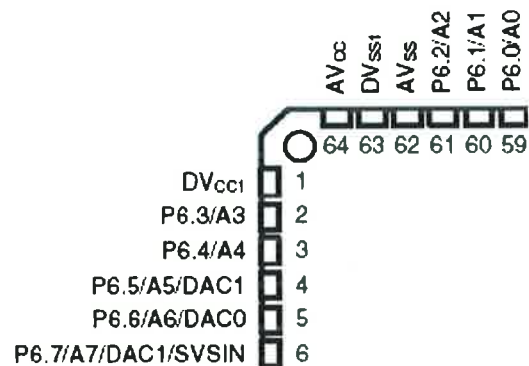


Ilustración 9. Sección de MSP

Especialidad Tecnólogo en Mecatrónica

Convertor de ADC

Esta conversión se hizo con el ADC12 que el micro controlador trae integrado, este a su vez tiene registros para su configuración, y entre sus características más destacables útiles para el proyecto están:

- 12 bits de conversión o resolución
- Velocidad máxima de conversión 200 Ksps (kilo muestras por segundo).
- Muestra y Retención con periodos de muestreo controlados por software (timers).
- Selección por software de referencia interna o externa.
- Conversión de acuerdo al reloj mismo del micro.
- Conversión de modo: simple, simple + repetición, secuencia, secuencia + repetición.
- 16 registros de almacenamiento para los resultados.

Para el ADC definido se utilizaron los siguientes registros:

ADC12ON; Apaga o prende ADC

ADC12CTL0 |= ENC; Habilita las conversiones

MSC; Múltiple muestras y convierte.

SHT0_2; Define el numero de ciclos para el reloj del ADC, útil a la hora de muestrear

SHP; Registro para seleccionar la fuente de muestreo

CONSEQ_3; Modo de conversión de secuencia, el 3 se refiere a que hace

INCH_0; Se selecciona el canal 0, tomando de referencia voltaje positivo de la fuente

INCH_1+EOS; Se selecciona el canal 1, también con referencia a la fuente, y se termina la secuencia de conversión.

Ahora bien la frecuencia a la que se está muestreando, es determinada por la siguiente figura.

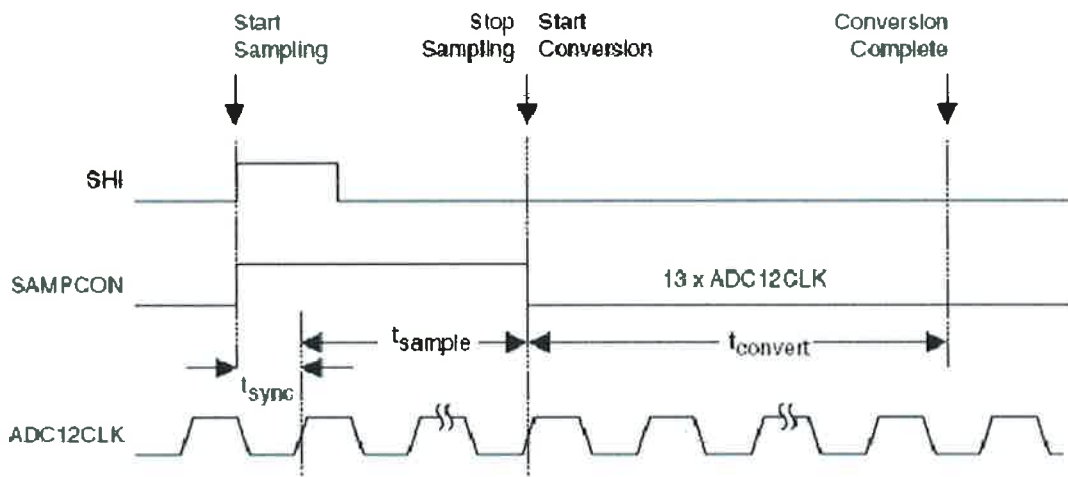


Ilustración 10. Frecuencia de Muestreo

Este tiempo de muestreo, se define como el $t_{sync} + t_{sample}$, en donde el t_{sample} es el que nos dará la información necesaria de cuánto tiempo tarda en tomar una muestra, siempre y cuando la señal de muestreo (SAMPCON) este en alto, ya que se puede simular como un circuito RC,

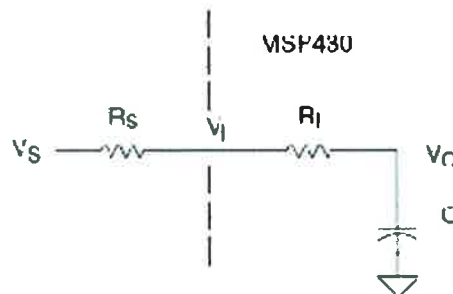


Ilustración 11. Circuito RC

Donde:

V_I = Voltaje del Canal A0,A1 a convertir.

V_S = Voltaje de fuente externa

R_S = Resistencia Externa

R_I = Resistencia interna en MUX o multiplexor.

C_I = Capacitancia interna

V_C = Voltaje de capacitancia

Especialidad Tecnólogo en Mecatrónica

He aquí que se pueden sacar las siguientes formulas en donde R_s y R_l afectan directamente a t_{sample} . Y se presenta una ecuación para determinar el tiempo de muestreo mínimo para 12 bits:

$$t_{sample} > (R_s + R_l) * \ln(2^{13}) * C1 + 800ns$$

Sustituyendo los valores de R_l y $C1$ tenemos que:

$$t_{sample} > (R_s + 2k\Omega) * 9.010 * 40pF + 800ns$$

Ahora bien falta por sustituir el R_s , pero este dependerá de la resistencia que tengan los diferentes sensores, un ejemplo es sustituir el $R_s = 10k\Omega$, valor con el cual se estuvieron haciendo pruebas, por tanto quedaría una ecuación y tiempo de muestreo de:

$$t_{sample} > (10k\Omega + 2k\Omega) * 9.010 * 40pF + 800ns$$

$$t_{sample} = 5.125 \mu s$$

Lo que quiere decir que cada 5.125 microsegundos toma una muestra el ADC.

Es así que si sustituimos en ecuación para obtener la frecuencia de muestreo, obtenemos algo como:

$$F = \frac{1}{T}; \text{ en donde } T = 5.125\mu s \quad F = 195.121.95 \text{ Hz}$$

Por tanto decimos que la frecuencia de muestreo es de aproximadamente 200Mhz.

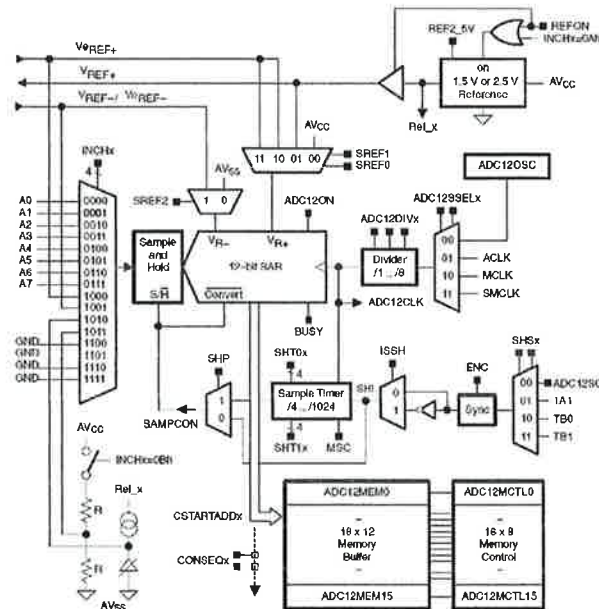


Ilustración 12. Diagrama a bloques de ADC12

Ahora bien como sabemos este núcleo convierte una entrada analógica para su representación de 12 bits y almacena el resultado en los registros de memoria, además de usar dos voltajes de referencia programables (VR+ y VR-) para definir los límites de conversión.

Se dice que la salida digital (NADC) toda su escala está en alto, cuando la señal es igual o superior a VR+, y bajo cuando es igual o menor a VR-, la señal de entrada y los voltajes de referencia son configurables mediante registros, en donde una simple fórmula de conversión nos muestra cómo es que está definida:

$$N_{ADC} = 4095 \times \frac{V_{in} - V_{R-}}{V_{R+} - V_{R-}}$$

En este caso, ya que los voltajes de referencia son de 2.5, cada uno, al hacer la operación nos queda la unidad multiplicada por la escala total que puede ser representada, que es 4095.

Almacenamiento de datos

Una vez que se realizó la conversión parte del proceso es que estas muestras que se están adquiriendo sean guardadas en un dispositivo de almacenamiento a modo de poder acceder a los datos para su descarga e interpretación.

Es por eso que a continuación se describe la forma en la que estos datos están siendo guardados, para esto se utilizó una memoria flash de Atmel AT45DB321D de 2.7 Volts, con interfaz serial y acceso secuencial, ideal para sonido digital, imágenes, código de programa y de almacenamiento

Especialidad Tecnólogo en Mecatrónica

de datos, además de contar con interfaz SPI para aplicaciones que requieren de alta velocidad de hasta 66 MHz

Por otra parte cuenta con 34.603.008 bits de memoria que se organizan en 8192 páginas de 512 bytes o 528 bytes cada una. Otro dato interesante de esta memoria es que además de su memoria principal cuenta con dos buffers de 512/528 bytes cada uno. Estos permiten la recepción de datos mientras que una página en la memoria principal se está escribiendo, así como escritura secuencial de datos.

Para utilizarla solo hace falta realizar 3 sencillos pasos, lectura-modificación-escritura.

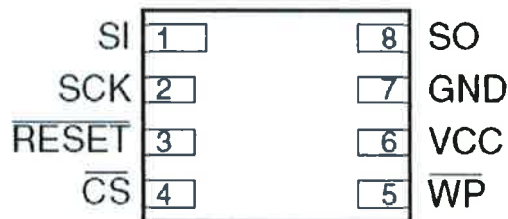


Ilustración 13. Vista de Memoria Flash, vista desde arriba.

Símbolo	Nombre y Función
CS	Chip Select: Cuando se activa este pin, se selecciona el dispositivo, y cuando es desactivado el dispositivo se deselecta y se pone en modo de espera, y el pin SO, estará en estado de alta impedancia. Cuando el dispositivo no está seleccionado los datos no se aceptaran el pin SI. Se necesita una transición alto a bajo para iniciar una operación y viceversa para terminarla
SCK	Este pin se utiliza para proporcionar un reloj para el dispositivo, que sirve para controlar el flujo de datos. Comandos, dirección y datos de entrada son controlados por SI siempre atados al flanco de subida del Reloj, mientras que el la salida esta del SO se bloquea en la caída del flanco.
SI	Serial Input: Este pin es utilizado para mover datos hacia el dispositivo. Es utilizado además para incluir comandos y secuencias de direcciones. Los datos relacionados a este pin siempre van junto con el flanco de subida del reloj.
SO	Serial Output: Es utilizado para mover datos fuera del dispositivo, los datos van junto con el flanco de bajada del reejo.
RESET	Reset: Un estado bajo en este pin hará que se termine la operación, en curso e iniciara la maquina interna de estados a un estado de inactividad.

Especialidad Tecnólogo en Mecatrónica

Vcc	Device Power Supply: Este pin es usado, para alimentar el dispositivo. Las operaciones con voltajes erróneos pueden producir errores.
GND	Ground: La referencia de tierra para la fuente de alimentación, esta debe estar conectada a una tierra física.

Tabla 1 . Descripción de pines, Memoria Flash.

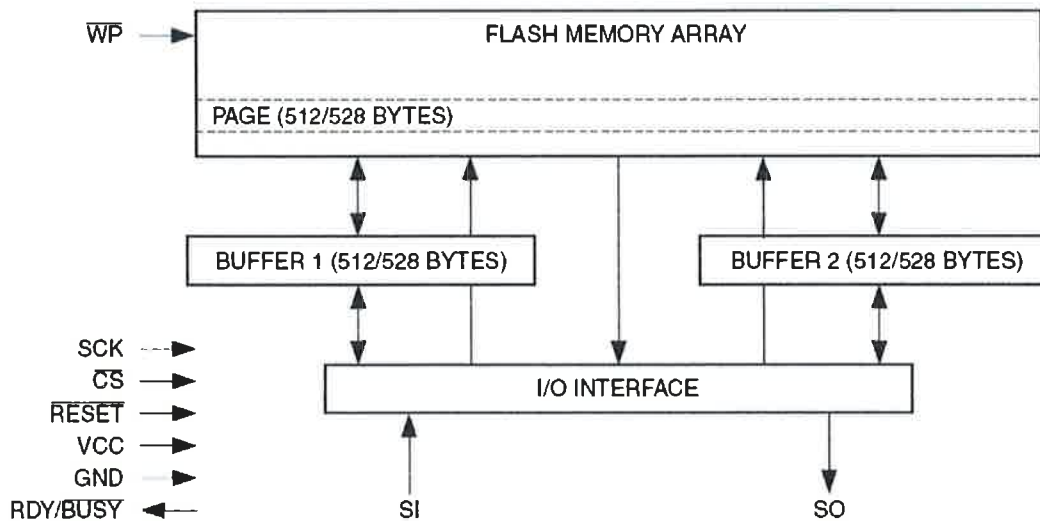


Ilustración 14. Diagrama a bloques de Memoria Flash

El proceso que se utiliza para guardar un dato, viene adjunto en la parte de código de los anexos, ya que para esto se utilizó una librería, creada por el M.C. Jorge Alberto Soto Cajíga, la cual consiste en ir escribiendo dato por dato y almacenarlo en las diferentes locaciones que tiene cada página de la memoria para su almacenamiento, y así sucesivamente hasta completar las 8190 páginas con las que cuenta la memoria

Entonces por cada dato a guardar estamos ocupando 12 bits, en total, y como tenemos dos diferentes datos tenemos en total 24 bits que son los que estamos guardando, entonces si:

1 byte = 8 bits

3 bytes = 24 bits, bits totales que queremos guardar.

Y la memoria que estamos utilizando tiene 512 bytes por página, y la memoria a su vez 8190 páginas. Por tanto tenemos 4194304 bytes, en los cuales podemos guardar 1398101.33 datos, entonces, si guardamos cada segundo, podemos guardar hasta 388.36 horas lo que es equivalente a 16.18 días.

Envío SPI

Especialidad Tecnólogo en Mecatrónica

Como sabemos para el envío de datos a la memoria desde el MSP, es necesario utilizar el protocolo SPI de comunicación, este tipo de comunicación es síncrona, y es un estándar establecido por Motorola, el cual se utilizó un bus de 4 líneas para interconectarla memoria flash al MSP, de baja y media velocidad.

Ya que tal estándar maneja un bus full dúplex, es decir, que se puede enviar y recibir información de manera simultánea, lo cual eleva la tasa de transferencia de datos, se estableció la velocidad igual a la utilizada en la parte serial, la cual es aproximadamente de 1Mhz.

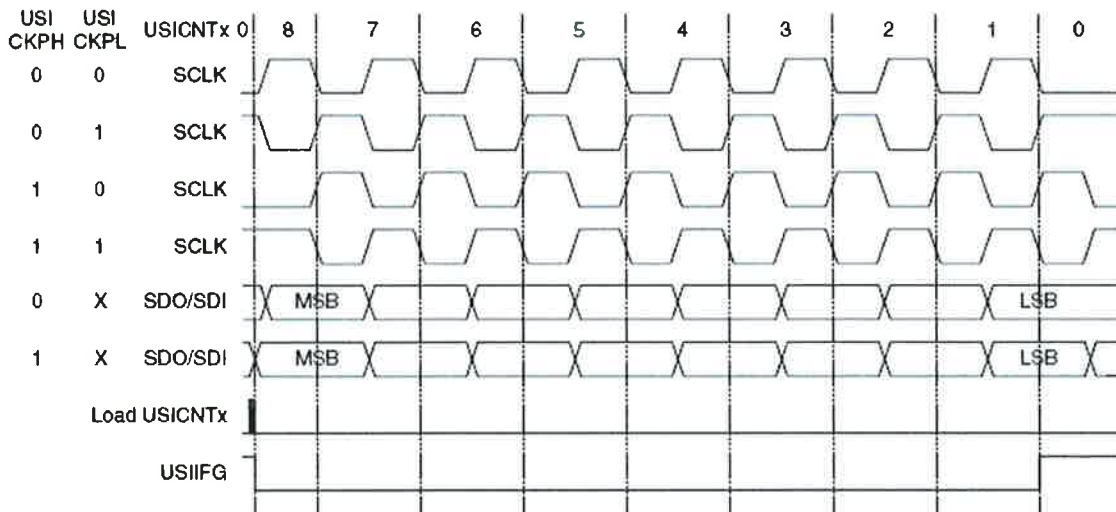


Ilustración 15. Sincronización SPI

Envío de Datos a la computadora

Para esta parte se utilizó la parte de comunicación serial, en modo asíncrono la parte de esta comunicación dentro del micro se utiliza por medio de dos terminales UCAXRXD y UCAXTXD.

Algunas de las ventajas al utilizar esta comunicación para el proyecto fueron:

- Tiene 7-8 bits de paridad, impar o sin paridad.
- Transmisión independiente y transmisión de registros.
- Transmisión y recepción separada de registros del buffer.
- Capacidad de interrupción de interrupción para recibir y transmitir.

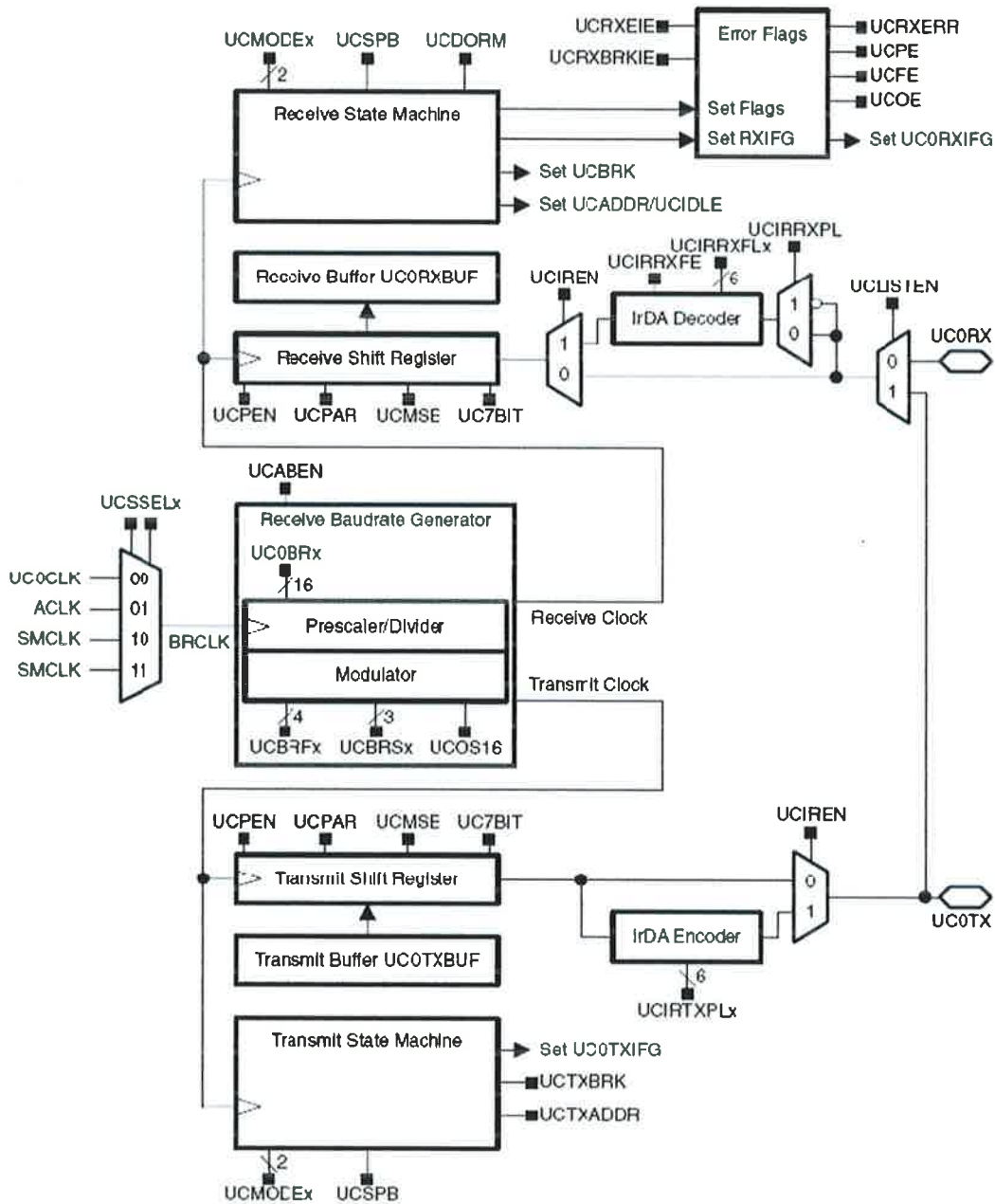
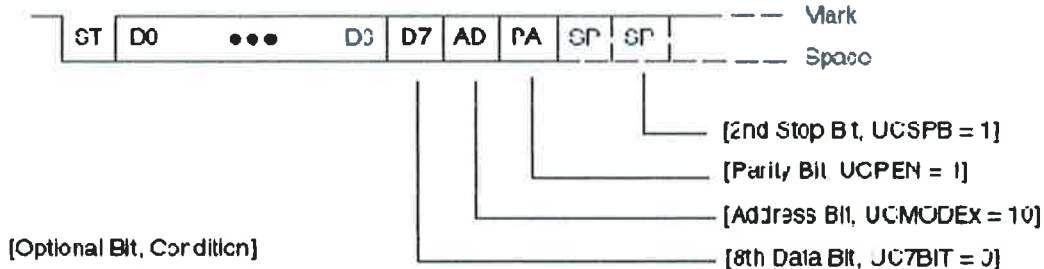


Ilustración 16. Diagrama a bloques de funcionamiento de UART

En este modo el micro envía caracteres a una velocidad de bits asíncrona a otro dispositivo. El tiempo para cada carácter se basa en la velocidad de transmisión del serial. La transmisión y la recepción utilizan en mismo baud rate.

Especialidad Tecnólogo en Mecatrónica

El formato de UART, consiste en un bit de inicio 7 u 8 bits, un par/ impar / sin paridad, un bit de dirección y 1 o 2 bits de parad. El bit UCMSB controla la dirección de la transferencia.



Es así que cada dato se manda por las terminales mencionadas anteriormente del micro, y estas son recibidas en la computadora por medio de estándar RS-232 hacia la computadora por medio de un adaptador db9 a USB. Y ya con estos datos poderlos mandar a alguna terminal para procesamiento futuro.

Estos datos son enviados en cadenas de 4 caracteres por vez, ya que la resolución significativa que podemos ver es de 4 dígitos.

Interfaz de graficador en Labview

Para esta parte se hizo uso del software bien conocido Labview, el cual nos permitió una vez teniendo los datos por RS-232, poder graficarlos y así ver el comportamiento del sistema, para así poder rectificar el propósito del sistema de monitoreo.

He aquí cómo llegan los datos a la computadora aproximadamente cada 500 ms, Nota, que esto es diferente a la velocidad q se está muestreando.

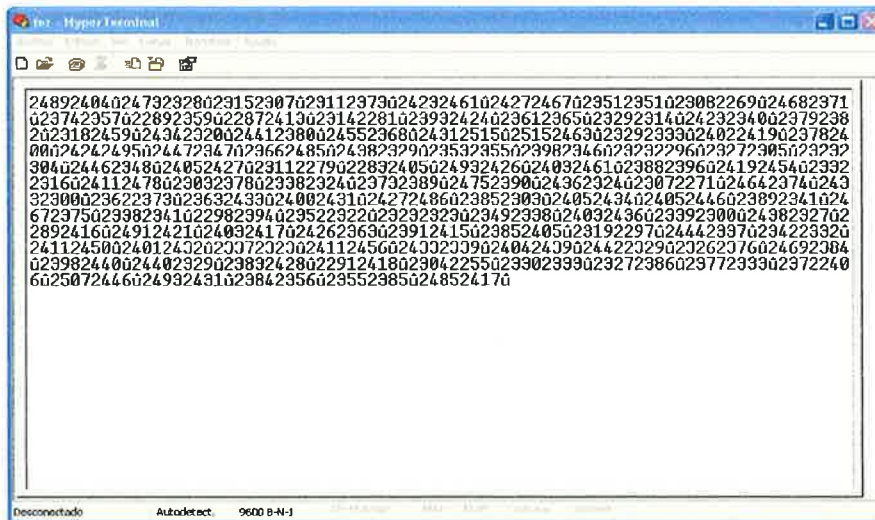


Ilustración 17. Cadena de información de sensores en medición.

Especialidad Tecnólogo en Mecatrónica

Después se creó un Vi, en Labview el cual nos permite ver el compartimiento tanto como el sensor de carga como el de temperatura.

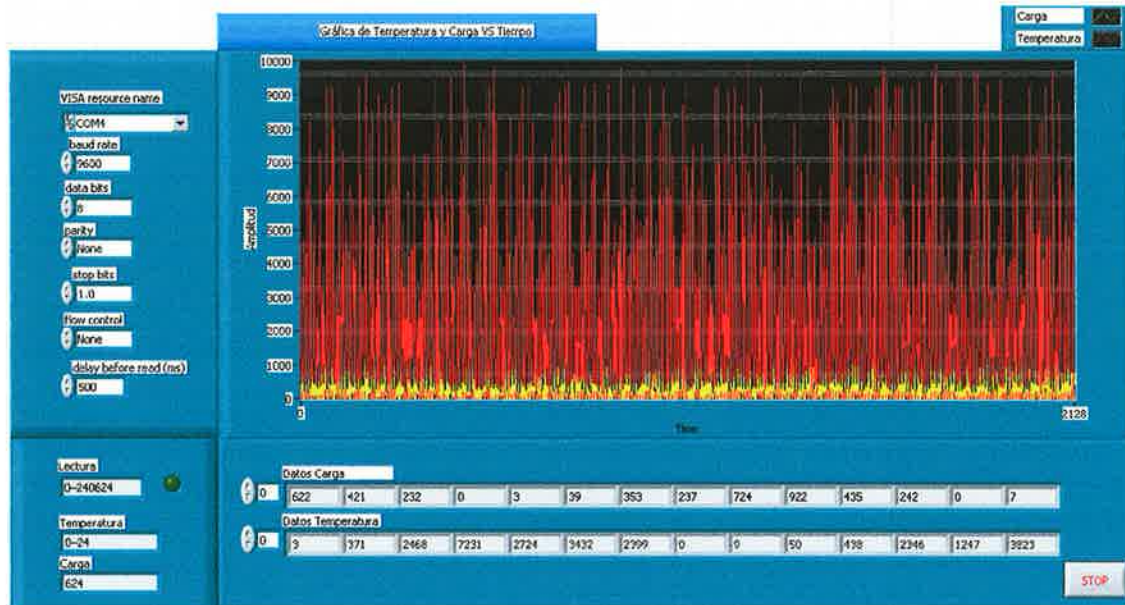


Ilustración 18 Interfaz en Labview para graficación de datos.

Anexos

Código de programación

```
//Interfaz de Monitoreo por SPI
//MSP4302619, flash atmel AT45DB321D

#include <msp430x26x.h>
#include <spi_func.h>
#include <icd.h>
#include <stdio.h>
#include <stdlib.h>
//#include <serialpru.h>

//*****Subrutinas*****

void configuracion ();
void convert();
void uartTxString();
void retardo();

//*****Variables*****

volatile unsigned int results[2];    //Registros de memoria
volatile unsigned int DAT, DAT1;    //Sensores T y C
char temp[4], carga[4];            //Sensores T y C
volatile unsigned int s;            //Variable Inicio - int
volatile unsigned int g;            //Variable Gaurdar - int
unsigned int j=0;                    //Dato
unsigned int k=0;                    //Página
char m=0;                            //Variable para memoria llena
char is=0, js=0;
unsigned int ks;
char stringI[4];
char enviar[8];

//*****Interrupciones*****
```

Especialidad Tecnólogo en Mecatrónica

```
/*
##pragma vector=PORT2_VECTOR          //Interrupción del puerto 2
__interrupt void Port_2(void)
//{
s = (P2IN & 0x00);
g = (P2IN & 0x01);

P2IFG &=~ 0x01;
}
*/
//-----
// Interrupcion TX
#pragma vector=USCIAB0TX_VECTOR
__interrupt void USCI0TX_ISR(void)
{
while (!(IFG2&UCA0TXIFG));
retardo(1000);
UCA0TXBUF = enviar[is++];          // TX siguiente caracter
retardo(1000);
if (is == sizeof enviar)          // TX over?
IE2 &= ~UCA0TXIE;                // Desabilita interrupcion USCI_A0 TX
}
// Interupcion en RX
#pragma vector=USCIAB0RX_VECTOR
__interrupt void USCI0RX_ISR(void)
{
stringl[ks++] = UCA0RXBUF;
}

//*****Main*****
```

Especialidad Tecnólogo en Mecatrónica

```
void main(void)
{

configuracion();           //Configuramos el micro
_BIS_SR(GIE);             //Habilita todas las interrupciones
lcdInitialise();          //Inicializa LCD
lcdDelay();
lcdGoto(0x80); lcdPutString("Monitoreo por SPI");
lcdDelay();
lcdGoto(0xC0); lcdPutString("Especialidad CIDESI");
lcdDelay();
lcdGoto(0xD4); lcdPutString("");
lcdClear();

while(1)                   //Convierte y Muestra en LCD
{
    s = (P2IN & 0x0F);
    if (s == 0x01)
    {
        convert();
        convertir(DAT, temp);
        lcdGoto(0x80); lcdPutString("Muestreando...");
        lcdGoto(0xC0); lcdPutString("Temp =");
        lcdGoto(0xC9); lcdPutString(temp);
        convertir(DAT1, carga);
        lcdGoto(0xD4); lcdPutString("Load = ");
        lcdGoto(0xDD); lcdPutString(carga);
    }

    else if (s == 0x02)     //Opción de Guardar
    {
```



```
while (s==0x02 && m==0)
{
  lcdClear();
  lcdGoto(0x80); lcdPutString("Saving...");
  convert();

  //---Temp
  convertir(DAT, temp);
  lcdGoto(0xC0); lcdPutString("Temp =");
  lcdGoto(0xC9); lcdPutString(temp);

  //----Carga
  convertir(DAT1, carga);
  lcdGoto(0xD4); lcdPutString("Load = ");
  lcdGoto(0xDD); lcdPutString(carga);

  j++;
  spi_wbyte_buff(j,DAT);
  spi_write_pag(k);
  if (j==65500)          //Checar dato
  {
    j=0; k++;
  }
  if (k==8190)
  {
    lcdClear();
    lcdGoto(0xC4); lcdPutString("Full Memory");
    m=1;
  }
}

else if (s == 0x04)          //Opción de Enviar Datos
{
```

```
convert();
convertir(DAT, temp);
lcdGoto(0x80); lcdPutString("Sendig Data...");
lcdGoto(0xC0); lcdPutString("Temp =");
lcdGoto(0xC9); lcdPutString(temp);
convertir(DAT1, carga);
lcdGoto(0xD4); lcdPutString("Load =");
lcdGoto(0xDD); lcdPutString(carga);
uartTxString(temp,carga);
}
else if (string!="1234")
{
    /***-----Descargar-----

    /***-----

    string={" "};
}
else if (s==0x00)
{
    lcdClear();
    lcdGoto(0xC4); lcdPutString("Select Option");
    lcdDelay();
    retardo(5000);
    lcdClear();
}
else
{
    {
        ADC12CTL0 &= ~INCH_1;      //Desactiva el canal 1
        ADC12CTL0 &= ~INCH_0;      //Deshabilita canal 0
        lcdClear();
    }
}
```

Especialidad Tecnólogo en Mecatrónica

```
    lcdGoto(0xC0); lcdPutString("Invalid Select");
    lcdGoto(0xD4); lcdPutString("Select Option");
    lcdClear();
}
}
}
}
//*****RS232*****
void uartTxString(const char *p)
{
    while(*p != 0){           //write data
        enviar[js++]=(*p++);

    }
    if (js > sizeof enviar - 1)
    {
        is = 0;
        js = 0;
        retardo(500);
        IE2 |= UCA0TXIE;           // Habilita interrupcion USCI_A0 TX
        UCA0TXBUF = enviar[js++];
    }
}

//*****Subrutina de Conversión (ADC)*****
void convert()
{
    ADC12CTL0 |= ADC12SC;           // Start convn - software trigger

    results[0] = ADC12MEM0;         // Move results, IFG is cleared
    DAT = ADC12MEM0;
```

Especialidad Tecnólogo en Mecatrónica

```
results[1] = ADC12MEM1;           // Move results, IFG is cleared
DAT1 = ADC12MEM1;
}

//*****Subrutina de Configuracion*****

void configuracion()
{
//-----Reloj-----
WDTCTL = WDTPW+WDTHOLD;           // Detiene Watchdog
if (CALBC1_1MHZ == 0xFF || CALDCO_1MHZ == 0xFF) // Se calibra a 1 Mhz.
BCSCTL1 = CALBC1_1MHZ;           // DCO a 1MHz
DCOCTL = CALDCO_1MHZ;

//-----SPI-----

if (CALBC1_1MHZ == 0xFF || CALDCO_1MHZ == 0xFF)
{

while(1);                         // If calibration constants erased do not load, trap CPU!!
}

UCB0CTL0 |= UCMST+UCSYNC+UCCKPL+UCMSB; //3-pin, 8-bit SPI master
UCB0CTL1 |= UCSSEL_2;               // SMCLK
UCB0BR0 = 0x02;                     // /2
UCB0BR1 = 0;                         //
UCB0CTL1 &= ~UCSWRST;               // **Initialize USCI state machine**

//-----Puertos-----

//P1Puerto reservado para display
P2DIR |= 0x00;                       // Configura el puerto 2 como entrada
```

Especialidad Tecnólogo en Mecatrónica

```
P3DIR |= 0X41;           //Chip Select & Any option
P3SEL |= 0x0E;          // P3.1,2,3 option select
P6SEL |= 0x0F;          // Enable A/D channel inputs

//-----ADC-----
ADC12CTL0 = ADC12ON+MSC+SHT0_2;    // Turn on ADC12, set sampling time
ADC12CTL1 = SHP+CONSEQ_3;          // Use sampling timer, single sequence
ADC12MCTL0 = INCH_0;               // ref+=AVcc, channel = A0
ADC12MCTL1 = INCH_1+EOS;;          // ref+=AVcc, channel = A1
ADC12CTL0 |= ENC;                  // Enable conversions

//-----interrupcion-----
//P2IE = 0x03;                    // Habilitamos la interrupcion de 2.0
//P2IES = 0x00;                    // Flanco 2.0 Low to High
//P2IFG &=~ 0x01;                  // Limpia la bandera
//__bis_SR_register(GIE);          // Habilitamos todas las interrupciones

//-----Serial-----
WDTCTL = WDTPW + WDTHOLD; // Detiene el temporizador del guardián del programa
if (CALBC1_1MHZ == 0xFF || CALDCO_1MHZ == 0xFF)
{
    while(1);                      // If calibration constants erased, do not load, trap CPU!!
}
BCSCTL1 = CALBC1_1MHZ;            // Set DCO
DCOCTL = CALDCO_1MHZ;
P3SEL = 0x30;                      // P3.4,5 = USCI_A0 TXD/RXD

UCA0CTL1 |= UCSSEL_2;             // SMCLK
UCA0BR0 = 6;                       // 1MHz 9600
UCA0BR1 = 0;                       // 1MHz 9600
UCA0MCTL = UCBRF3 + UCOS16;        // ModIn UCBSRx=1, over sampling
UCA0CTL1 &= ~UCSWRST;              // **Initialize USCI state machine**
```

Especialidad Tecnólogo en Mecatrónica

```
IE2 |= UCA0RXIE;           // Enable USCI_A0 RX interrupt

__bis_SR_register(GIE);    // Habilitamos todas las interrupciones
}

//*****Subrutina de Retardo*****
void retardo(unsigned int ret)
{
for(ret=(ret*1); ret > 0; ret--); // Ciclo para retardo.

}
```

Diseño de Placas

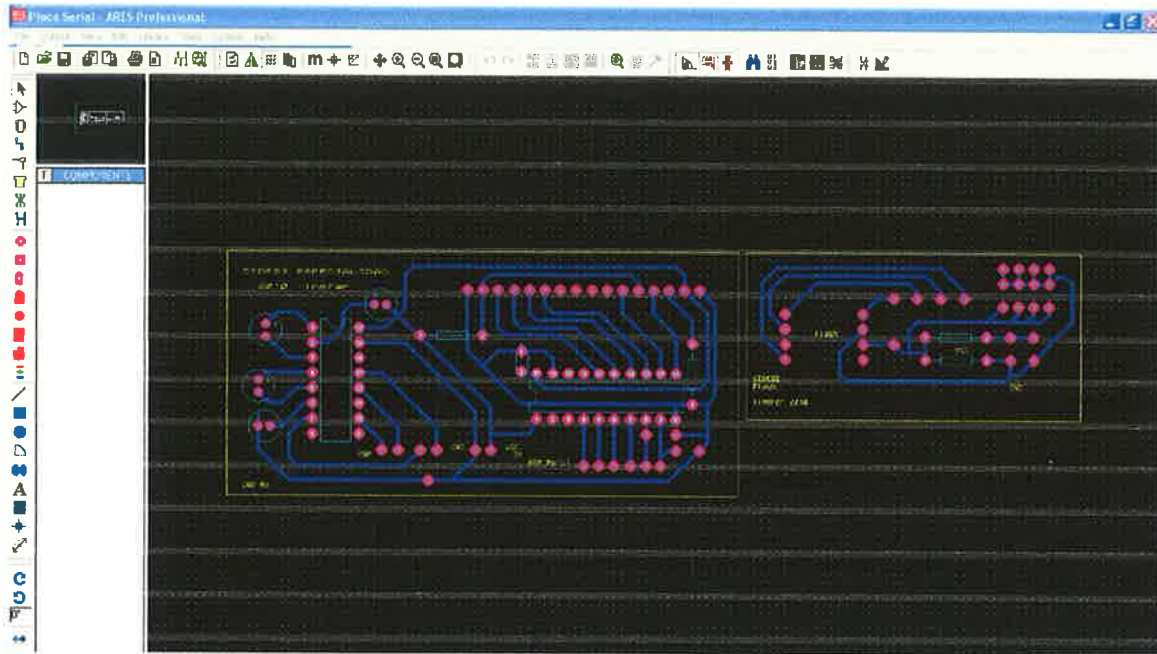


Ilustración 19. Diseño de placas en Proteus.

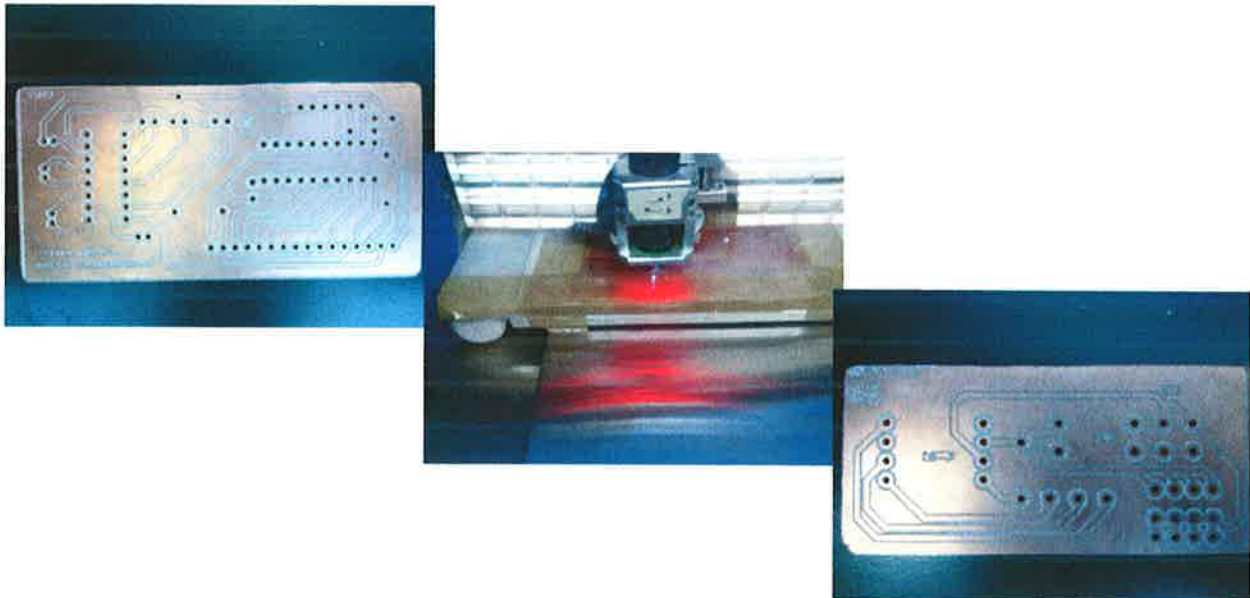


Ilustración 20. Fabricación de Placas.

Interfaz en Labview

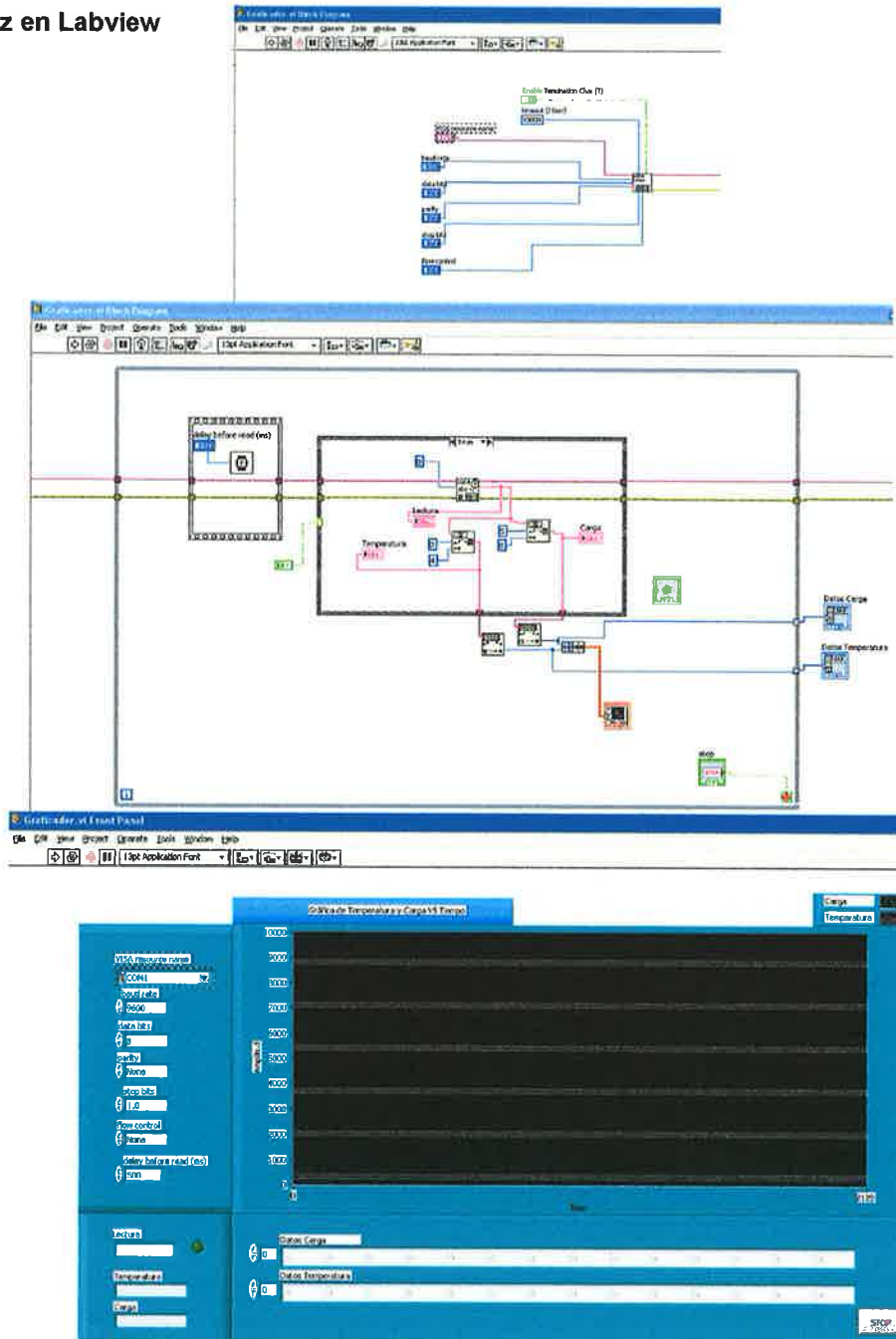


Ilustración 21. Vi's en Labview

Especialidad Tecnólogo en Mecatrónica

Conclusiones

Como ya se ha expuesto este tipo de sistemas de monitoreo existen, sin embargo el desarrollo de este prototipo es parte de objetivos más particulares y objetivos que van mas allá de un simple desarrollo, como lo fueron el aprender el uso muy particular de este micro controlador, ya que si se mira desde el un enfoque meramente escueto, solo vemos otro micro controlador, pero al indagar más en vemos cuenta con extensas funciones que al mirar rápidamente la hoja de datos no podemos denotar el alcance de sus capacidades.

Entonces es hasta cuando comenzamos a trabajar con él que vemos que algunas opciones para desarrollar son casi infinitas y que depende de la aplicación el uso final que le daremos a tantas opciones.

Ahora bien por la parte de comparación entre sistemas que ya existen en el mercado, este prototipo puede ser una opción a implementar cuando de sistemas de monitoreo estamos hablando ya que al ser un sistema abierto, podemos manipular las diferentes funciones e incluso aumentar las opciones para el sistema.

Ahora bien en la parte del código a implementar hay que recalcar que al ser un código basado en C, es entendible los comandos para poder realizar las distintas rutinas e incluso poder generar nuestras propias librerías, ya que para algunas funciones se tuvo que agregar pequeñas partes de código que pudieran resolver algunas operaciones y funciones.

Es así que al cumplir con los objetivos anteriores, solo quedaba realizar paralelamente el sistema físico capaz de realizar lo ya mencionado, lo cual quedo implementado en un pequeña caja para futuras modificaciones y aplicaciones de funciones, todo esto queda como parte de trabajo futuro para este proyecto.

Y así podemos decir que se desarrolló un sistema mínimo para el monitoreo de distintos sensores, con un micro controlador de bajo coste ideal para aplicaciones sin mucho capital o sin mucha complejidad.

Especialidad Tecnólogo en Mecatrónica

Bibliografía

PDF MSP430x2xxxx Autor: Texas Instruments, Fecha: 2008.

PDF MSP430x1xxxx Autor: Texas Instruments, Fecha: 2008.

PDF MSP430F16xx Autor: Texas Instruments, Fecha: 2008.

PDF Memoria Flash Atmel AT45DB321D. Autor: ATMEL, Fecha: 2008

PDF REporte de RS232 Autor: Celso Hilario Caralmpio, Rafael Montes, Fernando González, Fecha: Agosto 2010-12-21

PDF Sistemas de Intrumentación. Autor: José María Draka Moyano, Autor: 2005

Ejemplos contenidos en el software del MSP430.



M. en C. Leonardo Barriga

Firma Asesor



Fernando de Jesús González Aguado

Firma Alumno