



**CENTRO DE INGENIERÍA Y DESARROLLO  
INDUSTRIAL**

---

---

**PROYECTO:  
SISTEMA DE ADQUISICIÓN DE DATOS**

Tesis que presenta:  
**ING. SARAHI MORALES PÉREZ**

006952

Para obtener la especialidad de:  
**TECNÓLOGO EN MECATRÓNICA**



Asesor:  
**M. en C. JULIO CÉSAR SOLANO VARGAS**

Tema propuesto por:  
**M. en C. JULIO CÉSAR SOLANO VARGAS**

**SANTIAGO DE QUERÉTARO, QRO., DICIEMBRE 2012**

## DEDICATORIA

*A Dios y a mis padres*

*Luisa Concepción Pérez Soria y  
Carlos Morales Salinas*

*Por haberme financiado la especialidad, les agradezco de todo corazón su amor y apoyo incondicional que siempre me brindan. Muchas Gracias, que Dios los Bendiga hoy y siempre.*

## INDICE DE CONTENIDO

<b>ÍNDICE DE FIGURAS.....</b>	<b>i</b>
Figura 2.1: El nuevo proyecto se visualiza en el panel Projects.....	4
Figura 2.2: Nuevo esquemático agregado al Proyecto.....	5
Figura 2.3: Definiendo la forma de la placa PCB.....	5
Figura 2.4a: Geometría de la hoja de diseño.....	6
Figura 2.4b: Rectángulo trazado.....	6
Figura 2.4c: PCB definida con las dimensiones requeridas.....	6
Figura 2.5: Hoja de diseño.....	7
Figura 2.6: Cuadro de dialogo Board Options.....	8
Figura 2.7: Capas de Trabajo.....	9
Figura 2.8: Administrador de capas de trabajo.....	10
Figura 2.9: Deshabilitando Capas.....	10
Figura 2.10a: Ancho mínima de pistas.....	11
Figura 2.10b: Separación mínima entre redes aisladas.....	11
Figura 2.11: Verificación de error.....	11
Figura 2.12: Reglas de Diseño.....	12
Figura 2.13: Mensaje de error en el PCB.....	12
Figure 2.14. Paquete de Librería después de agregar la librería footprint.....	14
Figure 2.15. Nueva Librería PCB, listo para crea un footprint.....	15
Figure 2.16. Configurar las unidades y grids en el cuadro Board Options.....	16
Figura 2.17: Creación de nuevo componente.....	16
Figura 2.18: Nombrando componente nuevo.....	17
Figure 2.19: Configuración de las propiedades del pad.....	18
Figure 2.20. Etapa 1, colocación de pads.....	19
Figure 2.21: Arco del componente a diseñar.....	20
Figure 2.22: Arquitectura del Microcontrolador MSP430.....	22
Figura 2.23 Designación de patillas del microcontrolador.....	23
Figura 2.24: Módulo de reloj.....	26
Figura 2.25: Consumo de corriente típica del MSP430.....	30
Figura 3.1: Diagrama de Flujo Metodología.....	35
Figura 4.1: Aplicando flux a la tarjeta.....	40
Figura 4.2: Estañando los pines del MSP430.....	41
Figura 4.3: Estañando Pistas.....	41
Figura 4.4: Alineando el MSP430 con las pistas.....	41
Figura 4.5: Estañando cada pin del MSP430.....	41
Figura 4.6: Aplicando calor a los pines del MSP430.....	42
Figura 4.7: MSP430 soldado.....	42
Figura 4.8: Tarjeta MSP-TS430PM64.....	43
Figura 4.9: Programador para MSP430.....	43
Figura 4.10: Diagrama de flujo del código desarrollado para la sincronía de MSP430.....	44

Figura 4.11: Voltaje de alimentación de la tarjeta A.....	45
Figura 4.12: Tarjetas Conectadas.....	45
Figura 5.1: Señal de sincronía del MSP430.....	49

**INDICE DE TABLAS.....ii**

Tabla 2-1: Mapa de memoria del MSP430.....	27
Tabla 2-2: Vector de interrupciones.....	29
Tabla 2-3: Oscilador de Cristal LFXT1, Modo de alta Frecuencia.....	31
Tabla 2-4: Modos del contador.....	33
Tabla 4-1: Componentes creados en Altium.....	38
Tabla 4-2: Material usado para soldar.....	46

**CAPÍTULO 1.....iv**

1.1 Introducción.....	1
1.2 Objetivo General.....	2
1.2.1 Objetivos Especificos.....	2
1.3 Planteamiento del Problema.....	2
1.4 Justificación.....	2

**CAPÍTULO 2: MARCO TEÓRICO.....3**

2.1 Introducción a Altium Designer.....	4
2.2 Como definir el tamaño de la placa PCB en Altium Designer.....	5
2.3 Configurando el PCB.....	7
2.4 Verificando el circuito impreso.....	10
2.5 Librerías.....	13
2.5.1 Creación de componentes PCB (Footprint).....	13
2.6 Microcontrolador MSP430.....	21
2.7 Arquitectura del microcontrolador MSP430.....	21
2.8 Los puertos de entrada y salida Digital.....	23
2.9 El módulo de reloj.....	25
2.10 Temporizadores.....	28
2.10.1 El temporizador Watchdog.....	28
2.10.2 Los Temporizadores A y B.....	28
2.11 Modos de bajo consumo.....	29
2.12 Interrupciones.....	31
2.12.1 Interrupción Externa.....	32

## 1.1 INTRODUCCIÓN

La Adquisición de Datos o adquisición de señales, consiste, en tomar un conjunto de señales físicas, convertirlas en tensiones eléctricas y digitalizarlas de manera que se puedan procesar en una computadora o uno microcontrolador, para ello se requiere una etapa de acondicionamiento, que adecua la señal a niveles compatibles con el elemento que hace la transformación a señal digital, tal elemento, es la tarjeta de Adquisición de Datos.

La corrección de errores en el diseño de la placa de circuito impreso (PCB), así como la identificación de los componentes electrónicos de la tarjeta, lleva a pequeñas modificaciones, por lo cual, es necesario la creación de nuevas librerías para corregir el PCB.

Con el avance tecnológico los circuitos integrados de cierta complejidad tienden a venir en encapsulados de montaje superficial, para ello es necesario el empleo de técnicas para soldar dichos componentes, de manera que no se desuelden a causa de vibraciones ocasionadas en el robot de inspección de ductos.

El presente trabajo de especialidad describe el Sistema de Adquisición de Datos para el robot de inspección de ductos "Diablo Instrumentado". El diseño de la PCB se lleva a cabo en Altium Designer, la adquisición de la información obtenida se realiza mediante la programación de microcontroladores de Texas Instruments MSP430.

## **1.2 OBJETIVO GENERAL**

- ❖ Verificar el PCB de los circuitos eléctricos y programar el MSP430.

### **1.2.1 OBJETIVOS ESPECIFICOS**

- Aprender a manejar Altium.
- Verificar el circuito eléctrico en Altium.
- Crear el footprint de componentes que hagan falta.
- Identificar y elaborar una lista de los componentes de montaje superficial.
- Imprimir las tarjetas una vez verificadas.
- Soldar las tarjetas con los componentes correspondientes.
- Programar el microcontrolador.
- Probar el funcionamiento de las tarjetas.

## **1.3 PLANTEAMIENTO DEL PROBLEMA**

Los sistemas de adquisición de datos diseñados para los robots de inspección de ductos, empleados para la medición, adquisición y procesamientos de parámetros como son espesor, presión, temperatura entre otros, son capaces de soportar condiciones climáticas adversas tales como: altas temperaturas, presión, humedad, etc., estos sistemas se consiguen a precios muy elevados, debido a que, son desarrollados por países de primer mundo, los cuales cuentan con grandes avances tecnológicos en el área de la robótica de inspección, lo anterior nos obliga a depender de su tecnología, puesto que, no se cuenta con el apoyo e inversión necesaria para que nuestro país pueda desarrollar su propia tecnología.

## **1.4 JUSTIFICACIÓN**

Con el desarrollo de tecnología propia para el país hay un ahorro de tipo económico, por lo que, gracias al avance tecnológico en el mundo y a las necesidades de nuestro país, el diseño, control y desarrollo de un sistema de adquisición de datos para un robot de inspección de ductos elaborado por investigadores de nuestro país es posible en estos días, ya que, se cuenta con la tecnología adecuada para el diseño de la PCB, tal como Altium Designer, así como, el uso de microcontroladores MSP430 de la familia de Texas Instruments, los cuales cuentan con modos de bajo consumo de energía, económicos y fácil de programar para el diseño y control del Sistema de Adquisición de Datos.

# CAPITULO 2

# MARCO TEÓRICO

## 2.1 INTRODUCCIÓN A ALTIUM DESIGNER

Altium Designer es software que reúne todas las herramientas necesarias para crear un entorno completo para el desarrollo de productos electrónicos, en una sola aplicación. Altium Designer incluye herramientas para todas las tareas de diseño: desde la captura de esquemáticos y diseño HDL, simulación de circuitos, análisis de integridad de la señal, diseño de PCB, diseño y desarrollo de sistemas embebidos. Además, el entorno de Altium Designer se puede personalizar para satisfacer una amplia variedad de necesidades de los usuarios.

### CREANDO UN NUEVO PROYECTO

Para crear un nuevo Proyecto PCB:

1. Desde el menú principal, selecciona **File » New » Project » PCB Project**.

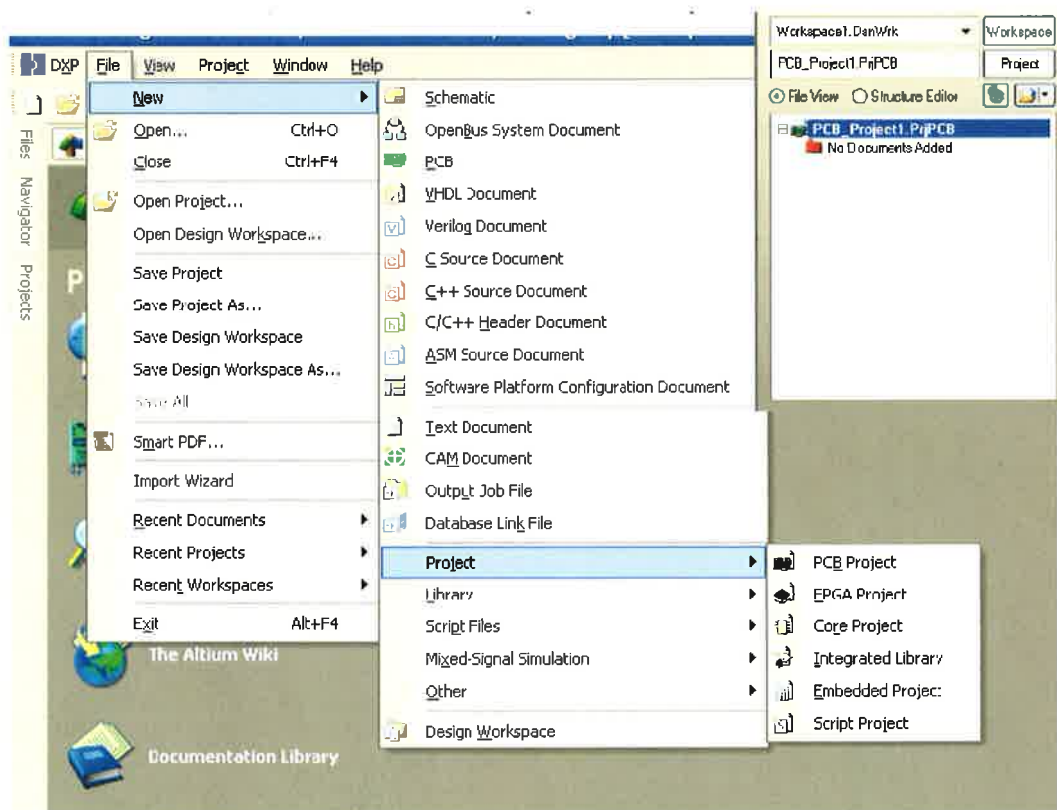


Figura 2.1. El nuevo proyecto se visualiza en el panel Projects

2. Selecciona **Save Project As** desde el menú **File** para nombrar y guardar el documento proyecto o puedes dar click derecho en el proyecto del Panel Projects y seleccionar **Save Project As**.

3. El nuevo proyecto está listo para agregar un documento nuevo o existente



## AGREGANDO UN NUEVO DOCUMENTO AL PROYECTO

Para agregar un nuevo documento al proyecto:

1. Click derecho en el nombre Project del panel Projects y desde sub-menú **Add New to Project**, selecciona el tipo de documento, por ejemplo, **Schematic**.
2. Click derecho en el nuevo documento Esquemático del panel Projects y seleccionamos **Save As** para nombrar y guardar el Esquemático

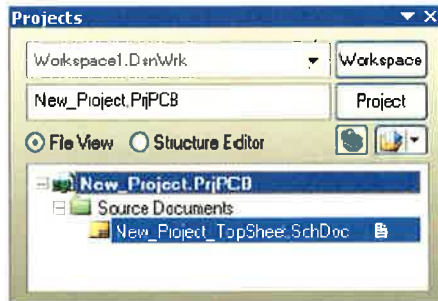


Figura 2.2 Nuevo Esquemático agregado al Proyecto

## 2.2 COMO DEFINIR EL TAMAÑO DE LA PLACA PCB EN ALTIUM DESIGNER

Para definir el tamaño de la placa del circuito a realizar, se realizan los siguientes pasos

Se selecciona **Design>> Board shape>> Redefine Board Shape**, como se muestra en la figura 2.3

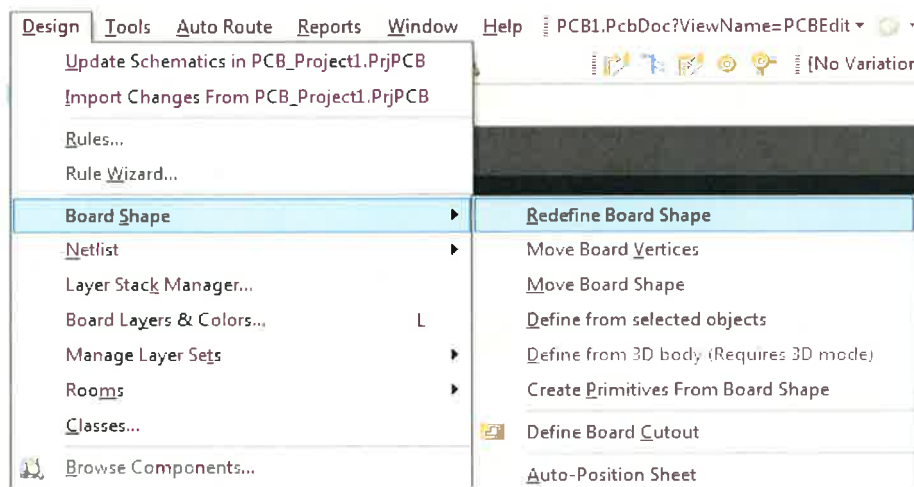


Figura 2.3: Definiendo la forma de la placa PCB

A continuación, se dibuja un rectángulo como se muestra en la figura 2.4a, a partir de ahí se sitúa en la esquina superior izquierda y a partir de ahí se traza el rectángulo dibujado mostrado en la figura 2.4b, después se definen el tamaño que queremos aprovechar del formato seleccionado, tal como se muestra en 2.4c.

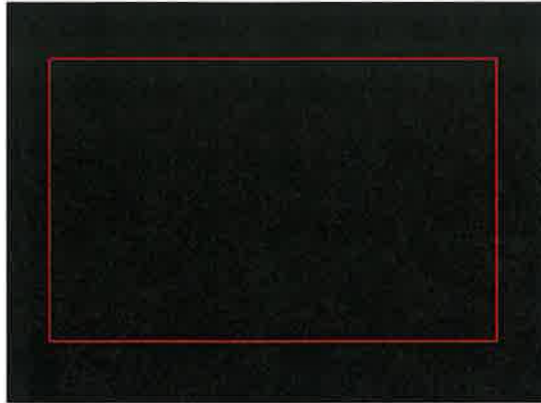


Figura 2.4a: Geometría de la hoja de diseño

006952

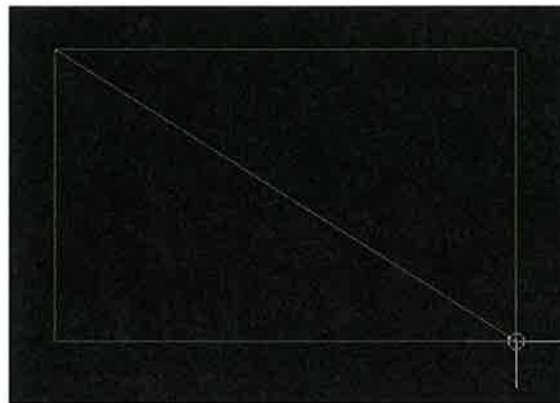


Figura 2.4b: Rectángulo trazado

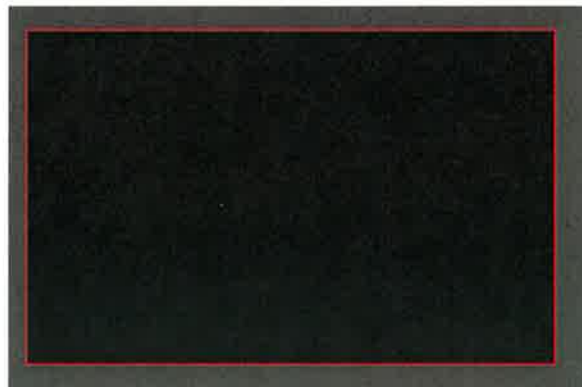


Figura 2.4c: PCB definida con las dimensiones requeridas

Una vez que se han definido los nuevos límites de la placa aprovechando la gran mayoría del espacio de la hoja tamaño carta, la hoja debe quedar semejante a la de la siguiente figura 2.5.

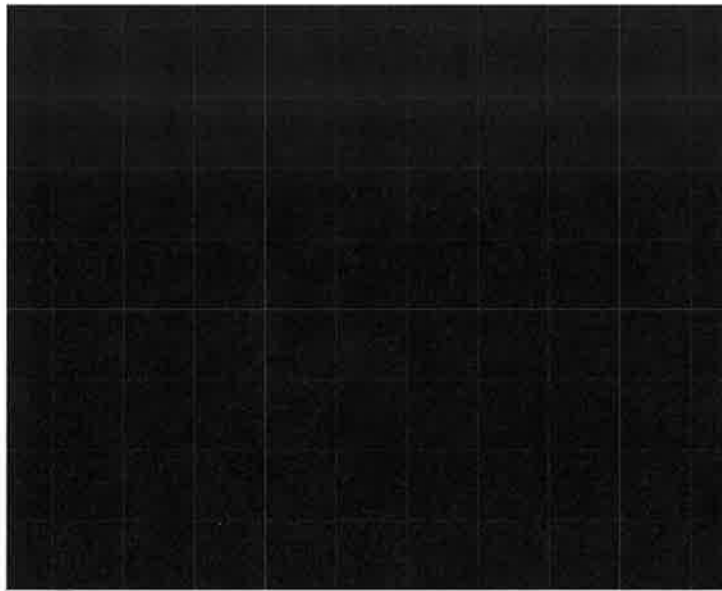


Figura 2.5: Hoja de diseño

Se puede observar que existen cuadrículas en la hoja de diseño, pero en realidad existe otra más que no se ve a simple vista.

Estas cuadrículas llamadas Grid sirven para orientarnos al momento de movernos dentro de la placa, su tamaño puede variar y a la vez existe la opción de verla en líneas o punteada.

Para modificar su tamaño, se selecciona **Design>> Board Options**. En el cuadro de diálogo emergente dentro de la opción visible Grid se puede modificar el valor existente para el tamaño del Grid ya sea a un valor menor o mayor.

### 2.3 CONFIGURANDO EL PCB

El cuadro de diálogo Board options permite configurar los parámetros relativos a documentos individuales de PCB. Para ello se selecciona **Design» Board options (BO)** de los menús para abrir el cuadro de diálogo. Las opciones de este cuadro de diálogo se guardan con el archivo PCB.

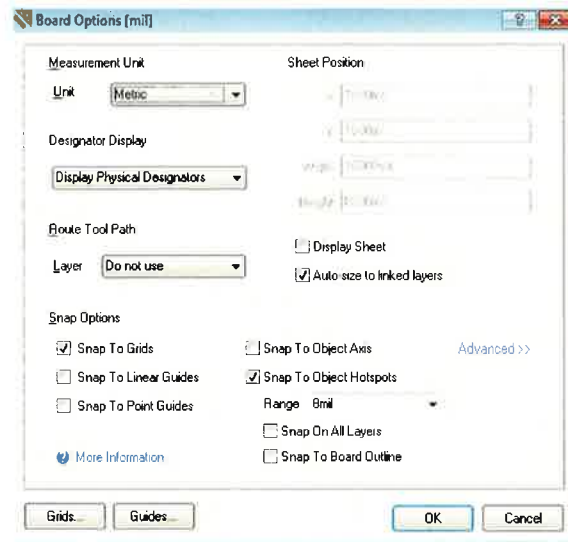


Figura 2.6: Cuadro de diálogo Board options.

## VER CONFIGURACIONES

- Este diálogo se utiliza para establecer el estado y el color de cada capa en el PCB en el modo de vista en 2D, y los colores y la transparencia en el modo de visualización 3D (tecla L para abrir el cuadro de diálogo).
- También se utiliza para configurar otra información relacionada, como la visualización de cada tipo de objeto, y la presentación de nombres de red en los pads.
- Vista de Configuraciones se pueden guardar y volver a cargar, con la última vista de configuraciones usada siendo aplicada automáticamente cuando la tarjeta se vuelve a abrir.

PCB puede trabajar con un amplio tipo de capas (Layers), que pueden ser: Capas de pistas (Signal Layers), Capas internas de Alimentación (Internal Planes), Capas de Serigrafía de Componentes (Silkscreen) y capas de dibujo Mecánico (Mechanical Layers)

Para controlar la visibilidad de estas capas es necesario configurarlas en la opción Board Layers & Colors, donde podremos agregar, cambiar de color o removerlas de nuestro diseño. Seleccionamos **Design>>Board Layers & Colors** para ver el cuadro de diálogo de esta opción.

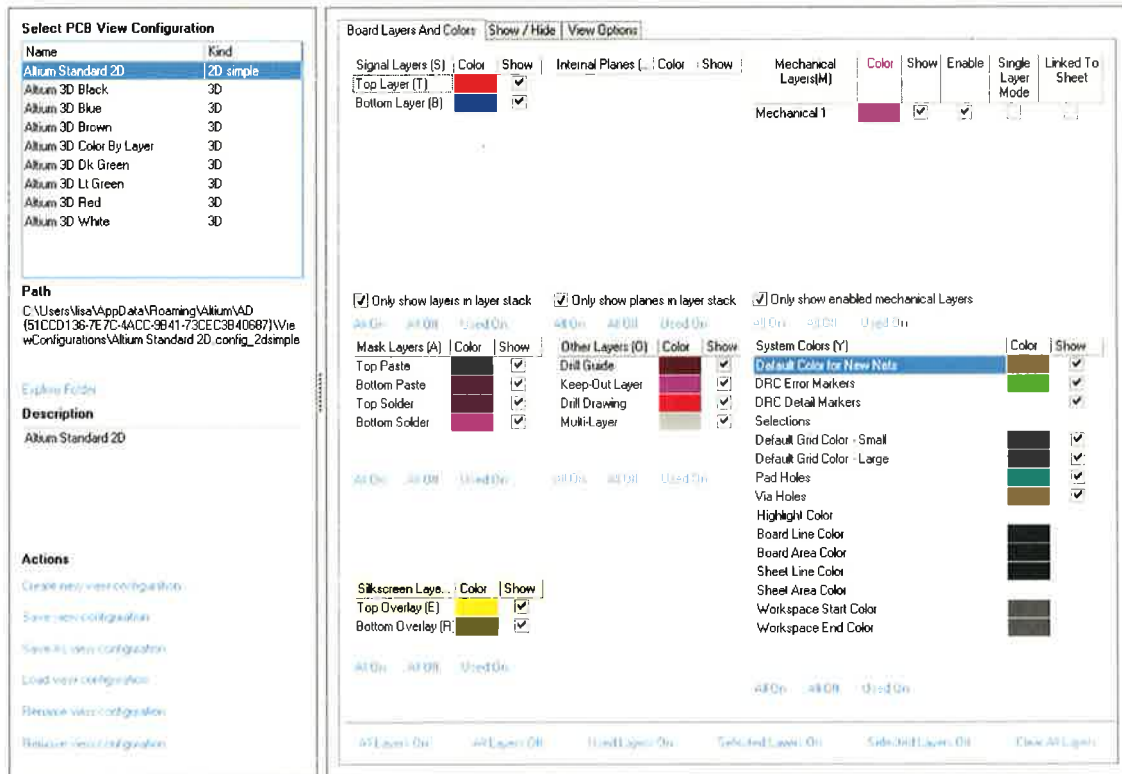


Figura 2.7: Capas de trabajo.

La funcionalidad de las capas de trabajo más importantes se detalla a continuación:

- a. **Signal Layers**, manejan las señales del circuito y cada una tiene un color en específico.
- b. **Internal Planes**, a esta capa se conecta cualquier señal de alimentación, así como, también las terminales de componentes de la placa que lo requieran.
- c. **Silkscreen Layer**, se utilizan para dibujar mediante proceso xerográfico el contorno de los componentes, y su identificación general así como también cualquier texto que se desee agregar al circuito.
- d. **Mechanical Layer**, se utiliza para proporcionar información de la fabricación de la placa como dimensiones, alineaciones etc.

### ADMINISTRADOR DE CAPAS

El siguiente paso consiste en el Administrador de Capas de Trabajo (**Layer Stack Manager**)

El circuito que se maneja hasta este punto puede ser trazado en una sola capa o en dos capas, pero si se requieren más capas estas pueden agregarse a través del asistente Layer stack Manager, para ello se hace lo siguiente.

1. Seleccionamos **Design>>Layer Manager**.

2. A partir de este cuadro de diálogo se agregan las capas necesarias. Presione OK para cerrar el cuadro de diálogo.

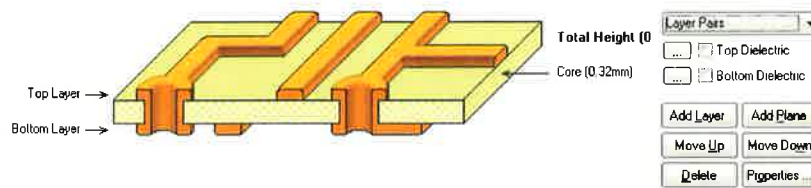


Figura 2.8: Administrador de capas de trabajo

Como el circuito impreso se abre con más capas de las necesarias dentro de la opción **Board Layers And Colors**, se pueden deshabilitar todas las capas innecesarias presionando el botón inferior **Used On**.

Hay que asegurarse que las cuatro capas de la opción **Mask y Drill Drawing** tengan deshabilitadas la opción show. Presione OK para cerrar el cuadro de diálogo.

Mask Layers (A)	Color	Show	Other Layers (O)	Color	Show
Top Paste	[Color]	<input type="checkbox"/>	Drill Guide	[Color]	<input type="checkbox"/>
Bottom Paste	[Color]	<input type="checkbox"/>	Keep-Out Layer	[Color]	<input type="checkbox"/>
Top Solder	[Color]	<input type="checkbox"/>	Drill Drawing	[Color]	<input type="checkbox"/>
Bottom Solder	[Color]	<input type="checkbox"/>	Multi-Layer	[Color]	<input checked="" type="checkbox"/>

Figura 2.9: Deshabilitando capas

## 2.4 VERIFICANDO EL CIRCUITO IMPRESO

Esta sección tiene como objetivo informar al diseñador de aquellos aspectos básicos a tener en cuenta para realizar un diseño cuyo resultado sea entendible y fácil de manejar para el fabricante.

“Defina las reglas de diseño en función de la clase de fabricación de su circuito”.

Antes de comenzar es esencial configurar los parámetros de diseño de acuerdo con los límites de fabricación del fabricante. Para ello es conveniente revisar las limitaciones de cada fabricante, siendo las más relevantes las que hacen referencia:

- Grosos mínimos de trazos de cobre
- Distancias mínimas entre puntos aislados.

Las reglas de diseño se pueden configurar en el editor de reglas del programa a través de menú

**Design>>Rules** accediendo a él podremos modificar los diferentes parámetros de diseño entre los que encontramos los que hacen referencia a los límites de fabricación, y que pasamos a enumerar en la figura 2.10:

**ANCHO MÍNIMO DE PISTAS:**

Design rules>>Routing>>with

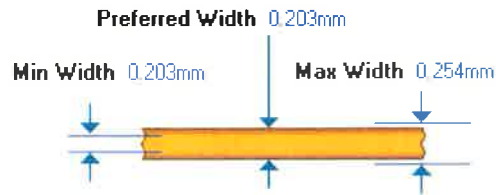


Figura 2.10a: Ancho mínimo de pistas

**SEPARACIÓN MÍNIMA ENTRE REDES AISLADAS:**

Design rules>>electrical>>Clearance



Figura 2.10b: Separación mínima entre redes aisladas

Para verificar el ruteo realizado en la tarjeta conforme a las reglas de diseño se utiliza la opción Design Rules Check como se muestra a continuación.

a. Seleccionar de los menús principales **Design>>Board Layer & Colors**, del cuadro de diálogo dirigirse a la opción systems Colors y verificar que la casilla SHOW de la opción DRC Error Markers este activada.

System Colors [Y]	Color	Show
Default Color for New Nets		<input checked="" type="checkbox"/>
<b>DRC Error Markers</b>		<input checked="" type="checkbox"/>
DRC Detail Markers		<input checked="" type="checkbox"/>

Figura 2.11: Verificación de error

b. Se selecciona el menú **Tools>>Design Rules Check** ó utilizar las teclas de acceso directo **T>>D**. Del cuadro de dialogo emergente presionar el botón Run Design Rule Check.

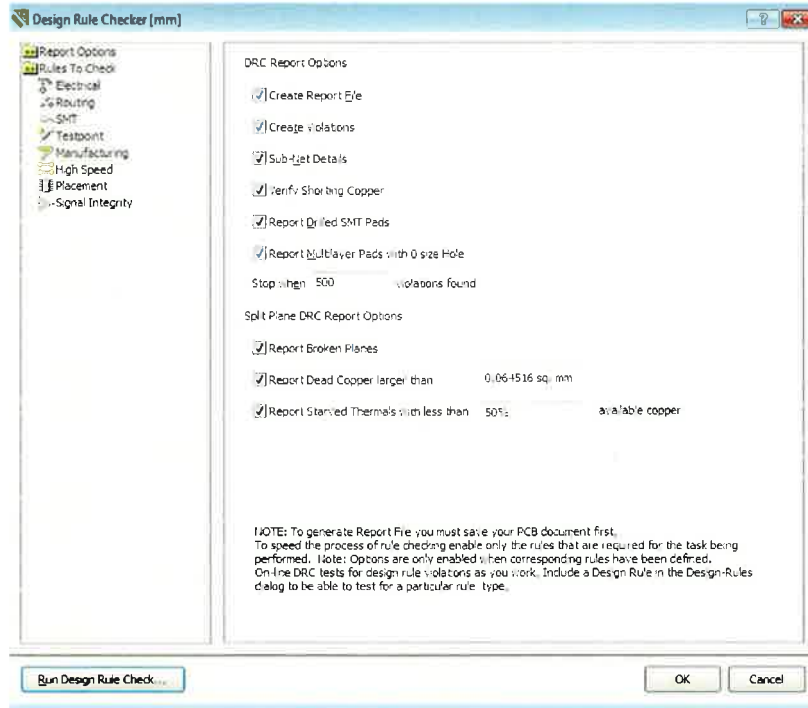


Figura 2.12: Reglas de diseño

c. El DRC se activará y se efectuará enseguida el proceso de verificación de errores. Si existe algún error, este aparecerá dentro del panel de mensajes. Para ello abra de nuevo el documento PCB, los errores en el trazado de las pistas o en la colocación de algún componente aparecen indicando la pieza o pista que realiza la violación en color verde.

d. Si el circuito carece de errores, se procede a colocar los componentes restantes y se realiza de nueva cuenta el procedimiento anterior.

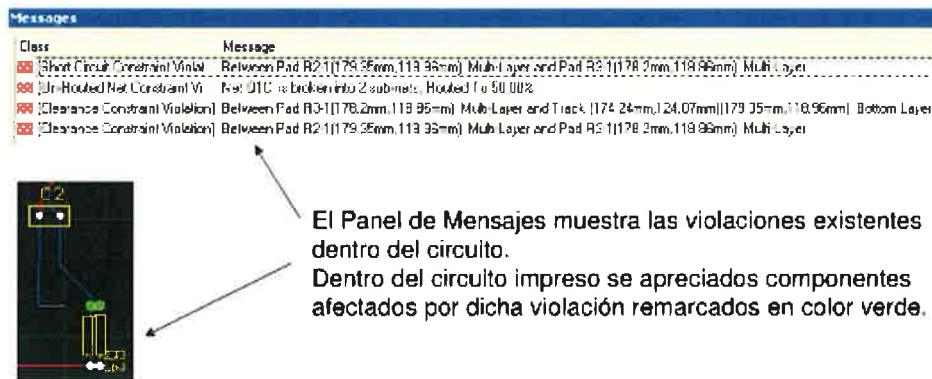


Figura 2.13: Mensaje de errores en el PCB



## 2.5 LIBRERIAS

Las librerías pueden existir como documentos individuales, por ejemplo, librerías esquemáticas que contienen símbolos esquemáticos, las librerías que contienen modelos de componentes PCB, modelos discretos SPICE (MDL y CKT), y así sucesivamente.

Altium Designer también ayuda a la creación de librerías integradas. Una librería integrada es el resultado compilado de un paquete de librería. Incluye todas las librerías de esquemas en el paquete de librería original, además de los modelos de referencia, incluyendo el footprint, la simulación y modelos de integridad de señal.

La mayoría de las librerías adjuntas se proporcionan como librerías integradas y se almacenan en la carpeta `\Program Files\Altium Designer Summer 09\Library` folder. librerías integradas pueden volver a convertirse en sus librerías constituyentes; simplemente abrirlas en Altium Designer para hacer esto. Librerías PCB también se proporcionan en el directorio `\Program Files\Altium Designer Summer 09\Library\Pcb` folder.

El Editor de Librería esquemática y Editor de Librería PCB son cubiertos durante la captura del esquemático y sesiones de entrenamiento de diseño PCB. Los fundamentos de la creación de una librería integrada también se cubren.

### 2.5.1 CREACION DE COMPONENTES PCB (FOOTPRINT)

Componentes PCB se construyen siempre en la parte superior. Capas con Atributos específicos, tales como pads de montaje superficial y soldadura, estas definiciones se transfieren automáticamente a las capas inferiores cuando se voltea el footprint al otro lado de la tarjeta durante la colocación de los componentes.

Un footprint se puede copiar desde el Editor de PCB en una librería de PCB, copiando entre librerías PCB, o creado desde cero, utilizando el Asistente para PCB, PCB Component Wizard del editor de componentes o herramientas de dibujo. Si se tuvo un diseño de PCB con todos los footprints ya colocados, se puede utilizar el Diseño usando el comando » Make PCB library en el Editor de PCB para generar una librería de PCB que incluye estos footprint únicamente.

Altium Designer también incluye completas bibliotecas de predefinido para componentes through-hole y SMD para su uso en el diseño de PCB. Las librerías de componentes proporcionados en (.PcbLib files) se almacenan en la carpeta `\Altium Designer 10\Library\Pcb` del directorio de instalación Altium Designer.

Los footprint que se crean manualmente en esta parte de la guía son sólo para ilustrar los procedimientos necesarios, no son dimensionalmente precisas. Siempre verifique las especificaciones de una nuevo footprint.

### Para crear una nueva librería PCB:

Seleccione **File >> New >> Library>> PCB library**. Un nuevo documento PCB library, se llama PcbLib1.PcbLib, se crea y una hoja sin componentes llamado PCBComponent\_1 que se despliega.

1. Renombre el nuevo documento PCB library a PCB Footprint.PcbLib, por ejemplo, al seleccionar **File >> Save As**. El nuevo PCB footprint library debe ser parte de la librería de paquete, como se muestra en la figura 2.14.
2. Abra el panel de PCB Library al dar click en la pestaña PCB Library.
3. Click una vez en el área gris del workspace PCB Library Editor y presiona la tecla PAGE UP unas cuantas veces, hasta que se pueda ver las rejillas como se muestra en la figura 2.15.

Ahora está listo para agregar, eliminar o editar los componentes en la nueva biblioteca de PCB utilizando los comandos PCB Library Editor.

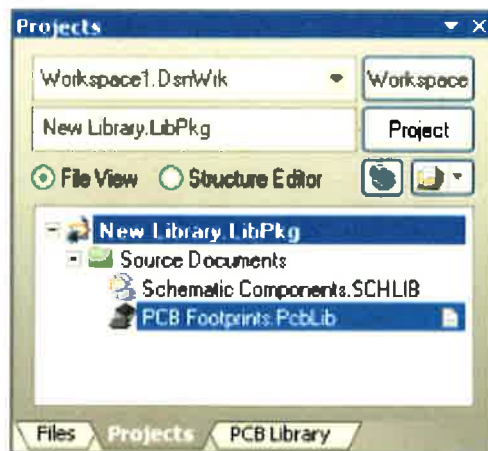


Figure 2.14. Paquete de Librería después de agregar la librería footprint.

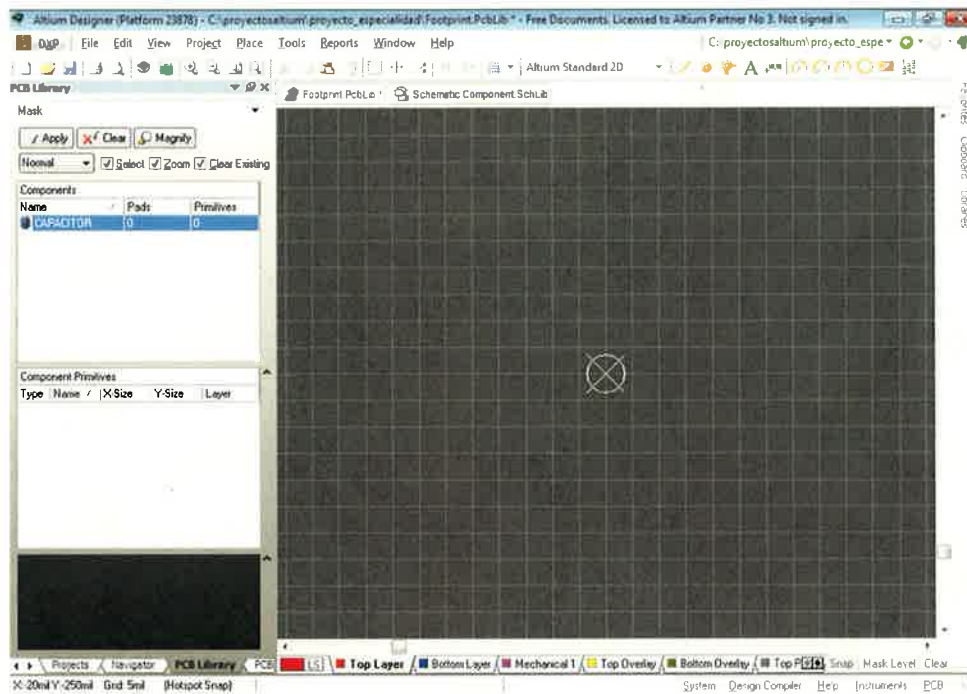


Figure 2.15. Nueva Librería PCB, listo para crea un footprint.

**Para crear un footprint manualmente se hace lo siguiente:**

Los Footprint se crean y modifican en el Editor de listas de PCB utilizando el mismo conjunto de herramientas y objetos de diseño disponibles en el Editor de PCB. Cualquier cosa puede ser guardada como un footprint de PCB.

Para crear el componente, vamos a colocar los pads para formar los pines del componente, y entonces colocar las pistas y arcos para el contorno. Los objetos de diseño se puede colocar en cualquier capa, sin embargo, el esquema se crea normalmente en el Top Layer (serigrafía) de la capa y los pads sobre la multicapa (para componentes de through hole) o la señal de la capa superior (para unos componentes de montaje superficial). Cuando se coloca el footprint en una PCB, todos los objetos que forman el footprint serán asignados a sus capas definidas.

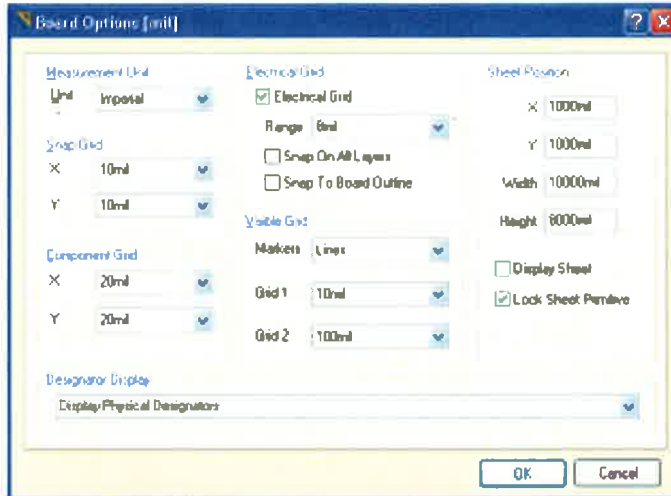


Figure 2.16. Configurar las unidades y grids en el cuadro Board Options.

Para crear manualmente un footprint adecuado para el transistor NPN:

1. Antes de crear el componente, verifique que las unidades y las cuadrículas son adecuados. Seleccione **Tools»Library Options** [atajo: D, O] para que aparezca el cuadro de diálogo de Opciones, y confirmar que las unidades sean Imperial, y el Snap Grid está ajustado a 10mil en las direcciones X e Y. Usted tendrá que configurar la red para adaptarse al espacio requerido por los pads en el footprint que se está creando. Establecer Cuadrícula visible 1 a 10mil y Cuadrícula visible 2 a 100mil.

2. Un espacio de trabajo para un componente vacío se crea al seleccionar **Tools>> New Blank Component** [atajo: T, W] figura 2.17, sin embargo, la nueva biblioteca ya cuenta con un footprint en blanco, por lo que vamos a utilizar ese.

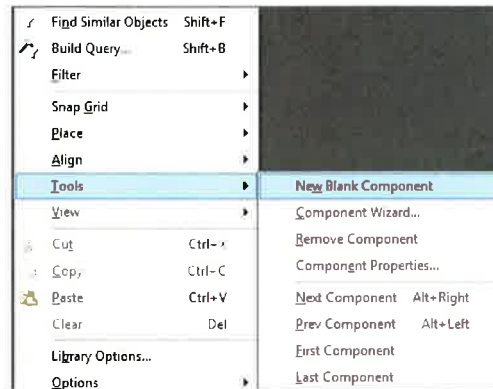


Figura 2.17: creación de nuevo componente

3. Para cambiar el nombre de este componente en blanco por defecto, haga doble clic sobre su nombre en la lista en el PCB del panel Biblioteca (que se llama algo así como PCBComponent\_1). Vamos a nombrar el componente como BCY-W3, escriba el nombre del componente nuevo en el cuadro de diálogo de componentes PCB Library.

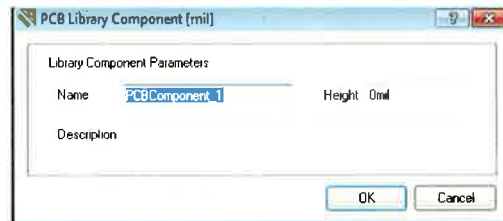


Figura 2.18: nombrando componente nuevo

4. Se recomienda que se construye el footprint en todo el espacio de trabajo 0, 0 punto de referencia, indicado por el marcador de origen. Utilice el método abreviado J, R para saltar el cursor hasta el origen en cualquier momento mientras está trabajando.

El punto de referencia es el punto que tendrá el componente cuando usted lo coloque. Típicamente, el punto de referencia es el centro del pad 1 o el centro geométrico del componente. El punto de referencia puede ajustarse a cualquiera de estos términos en cualquier momento utilizando la Edición »Configuración de opciones Referencia submenú.

Para cambiar la rejilla de referencia mientras trabaja presione CTRL + G. Para mostrar u ocultar las rejillas visibles pulse la tecla L para mostrar el cuadro de diálogo Vista Configuraciones. Si el marcador de origen no aparece, abra el cuadro de diálogo Configuraciones Ver y active la opción Marcador de Origen en la página Opciones de vista.

El diálogo de propiedades Pad tiene un visor que le permite inspeccionar las formas de pad en las capas definidas. Usted será capaz de definir circular normal, ovalados (ranurado) o agujeros cuadrados en pads y cambiar sus propiedades plateado (plateado o no plateado) y todo el trabajo necesario para soportar la generación de nodos térmicos, cálculo de distancias, salida a Gerber, ODB ++ y Drill NC por ejemplo, se maneja automáticamente. La salida NC Drill va a generar hasta seis diferentes archivos NC para tres tipos diferentes de hoyos y si son o no plateados o no. Uno de los procedimientos más importantes para crear un nuevo componente es colocar los pads que se utilizan para soldar el componente en la PCB. Estos deben ser colocados en las posiciones exactas para corresponder a los pines del dispositivo físico.

#### **Como se Colocan los pads:**

i. Seleccione Colocar>> [atajo: P,P] o click al botón en la barra de herramientas. Un pad aparecerá flotando en el cursor. Antes de colocar el primer pad, presione la tecla TAB para

definir las propiedades del pad. El cuadro de diálogo que se despliega del pad, se muestra en la figura 2.17.

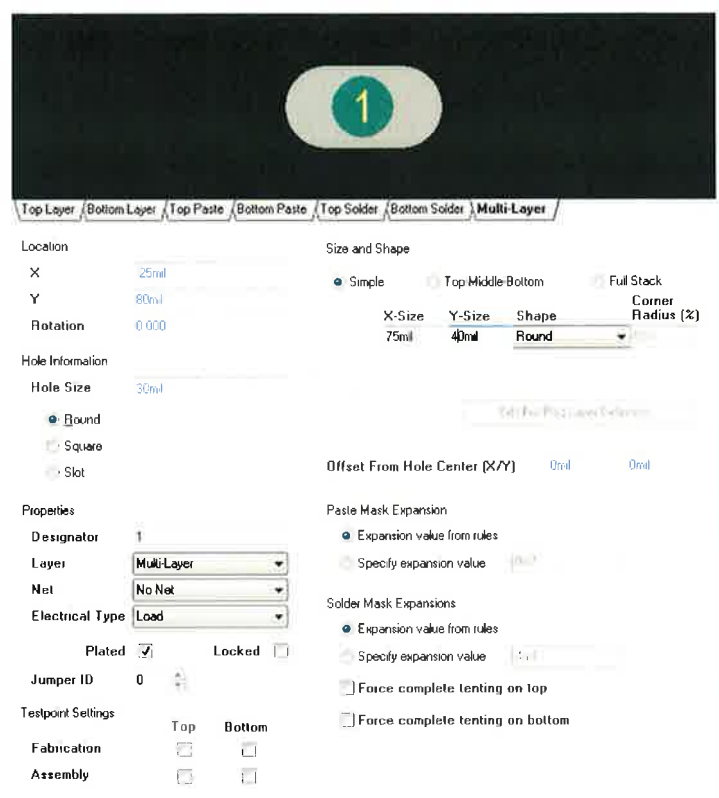


Figure 2.19. Configuración de las propiedades del pad.

ii. Editar las diversas regiones de la ventana de diálogo, como se muestra en la Figura 2.19. Esto crea un pad redondo estirado.

iii. Usando las coordenadas mostradas en el lado izquierdo de la barra de estado, la posición de la primer pad en X: 0, Y: -50, y haga clic (o pulse ENTER) para colocar el pad 1.

iv. Después de colocar el primer pad, otro aparecerá en el cursor. Coloque el cursor en X: 0, Y: 0, luego haga clic para colocar el segundo pad. Nota: el designador de pad se incrementa automáticamente

v. Coloque el cursor en X: 0, Y: 50, haga clic para colocar el tercera pad.

vi. Haga clic derecho, o presione ESC para salir del modo colocación de pad. Los tres pad deben ser similares a la Figura 2.20.

vii. Guarde el footprint al seleccionar **File » Save** [atajo: CTRL + S].

Los Pads pueden ser etiquetados con un designador (por lo general que representa el número de componente del Pad) de hasta 20 caracteres alfanuméricos. El designador se puede dejar en blanco si lo desea.

Si el designador comienza o termina con un número, el número será incrementado automáticamente cuando se coloca una serie de pads de forma secuencial. Para lograr incrementos alfa, por ejemplo,



Figure 2.20. Etapa 1, colocación de pads.

El esquema que aparece en la serigrafía PCB se define en la capa de revestimiento superior. Si el componente se da la vuelta a la parte inferior de la placa durante la colocación, la plantilla se transfiere automáticamente a la capa de superposición inferior.

Presione Q para cambiar las coordenadas de imperial (mil) a metric (mm).

1. Haga clic en la ficha Top Overlay de la parte inferior de la ventana de edición principal antes de colocar objetos superpuestos, tales como arcos o líneas (las pistas).
2. En primer lugar, vamos a poner el arco, como se muestra en la Figura 2.19. Para colocar el arco, seleccione Place »Arc (Centro) en los menús. Coloque el cursor en X: 0, Y: 0 y haga clic para definir el centro del arco. Si conoce el radio del arco y ángulos de inicio y fin, es realmente más fácil de completar la colocación del arco sin tratar de definir interactivamente estos ajustes, a continuación, editar el arco colocado a través de los ajustes en el cuadro de diálogo Arco.
3. Haga clic en cualquier lugar para aproximadamente definir el radio del arco y haga clic para definir el ángulo inicial del arco. Si es necesario, puede pulsar la barra espaciadora para cambiar la dirección del arco antes de definir el ángulo final, establezca la dirección como se muestra en la Figura 2.21 a continuación, haga clic de nuevo para definir el ángulo final del arco. Haga clic para salir del modo de arco de colocación.

Ahora haga doble clic sobre el arco colocado para mostrar el cuadro de diálogo Arco, y establezca las propiedades de la siguiente manera:

Width=6mil, Radius=105mil, Start Angle=55, End Angle=305.

4. Lo siguiente es la línea. Seleccione Place »Line [atajo: P, L], o haga clic en el botón. Coloque el cursor cerca del final del arco y pulse PAGE UP para acercar la imagen, como se muestra en la Figura 2.21. Al mover el cursor cerca del extremo del arco se encajará a la misma, esto es la red eléctrica tirando el cursor hasta el final del objeto existente. Haga clic para iniciar el segmento de línea.

5. Presione la tecla TAB para definir el ancho de la línea (6mil) y compruebe la capa en el diálogo Línea de restricciones.

6. Mueva el ratón hasta que está en el otro extremo del arco, a continuación, haga clic de nuevo para definir el otro extremo de la línea.

Nota: Durante la colocación de una vía que puede alternar entre los modos de esquina diferente de la línea pulsando la tecla SHIFT + SPACEBAR combinación clave.

7. Para salir del modo de línea de colocación, haga clic derecho o presione ESC

Si se comete algún error al colocar la línea, presiona BACKSPACE para remover el último segmento de línea.



Figure 2.21: Arco del componente a diseñar



## 2.6 MICROCONTROLADOR MSP430

La familia de microcontroladores MSP430 Texas Instruments de potencia ultra baja consta de varios dispositivos que ofrecen diferentes conjuntos de periféricos específicos para diversas aplicaciones. La arquitectura, junto con cinco modos de bajo consumo se ha optimizado para lograr una vida prolongada de la batería en aplicaciones de medición portátiles. El dispositivo cuenta con una poderosa CPU RISC de 16-bit, los registros de 16-bits, y generadores constantes que contribuyen a la eficiencia máxima del código. El oscilador calibrado controlado digitalmente (DCO) permite despertador de modos de bajo consumo al modo activo en menos de 1  $\mu$ s.

La principal fuente de información acerca de este dispositivo son: la guía del usuario del MSP430 y las hojas de datos del dispositivo en particular, en este caso del MSP430x2xx.

La serie MSP430x2xx son configuraciones de microcontroladores con dos temporizadores de 16-bits, un rápido convertidor de 12-bit A/D, un comparador, convertidor dual de 12-bit D/A, cuatro módulos de comunicación universal serie (USCI), DMA, y un máximo de 64 pines I/O.

Las aplicaciones típicas incluyen sistemas de sensores, aplicaciones industriales de control, etc. 12x12 mm paquete LQFP-64 también está disponible como un paquete no magnético para aplicaciones de imágenes médicas.

## 2.7 ARQUITECTURA DEL MICROCONTROLADOR MSP430

El microcontrolador MSP430 incorpora un CPU de 16-bits con arquitectura RISC, diferentes periféricos y un sistema de reloj flexible que se interconectan usando memorias comunes de direccionamiento y datos tipo Von-Neumann. Además incorpora la combinación de periféricos digitales y analógicos que ofrecen solución a diferentes aplicaciones de señales mixtas (analógicas y digitales).

Las principales características que incluye la familia MSP430x2xx son:

- Ultra bajo consumo de energía, que extiende la vida de baterías
- 0.1  $\mu$ A de retención de memoria RAM
- 0.8  $\mu$ A operando el reloj de tiempo real
- 250  $\mu$ A por cada millón de instrucciones por segundo (MIPS) en modo activo
- Módulos analógicos de alto desempeño para aplicaciones de medición de precisión.
- Temporizadores comparadores para diferentes aplicaciones.

- CPU de 16 bits que desarrolla aplicaciones en fracciones de código
- Archivos de registro de gran tamaño
- Diseño compacto del núcleo que reduce energía y costo
- Optimizado para programación moderna de alto nivel
- Solo 27 instrucciones de núcleo y 7 modos de direccionamiento
- Vector de interrupciones
- Programación de flash en sistema, para reprogramación sencilla.

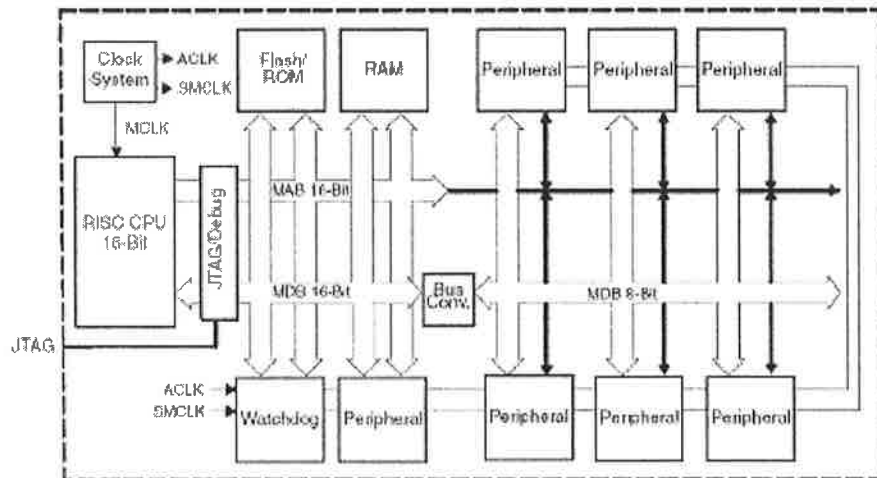


Figura 1. Arquitectura del Microcontrolador MSP430.

Figura 2.22 Arquitectura del Microcontrolador MSP430

## 2.8 LOS PUERTOS DE ENTRADA Y SALIDA DIGITAL

Trabajando con el **MSP430** no se tiene posibilidad de disponer de los buses de datos y direcciones en el exterior.

Para su comunicación con el resto de componentes y periféricos que formen parte del circuito en que se aloje el **MSP430**, éste dispone de 6 puertos de entrada y salida de 8 bits cada uno, que se pueden habilitar o no individualmente como se muestra en la figura 2.23.

El microcontrolador **MSP430x2xx** dispone de 6 puertos de entrada y salida P1-P6, de 8 bits cada uno; de ellos, los puertos uno y dos tienen capacidad de producir interrupciones. Todos los registros tienen sus bits programables individualmente como entrada o salida y como pin de puerto o como periférico. Para lo cual, cada puerto dispone de cuatro registros de control para su configuración; salvo los puertos uno y dos, que disponen de seis permitiendo así trabajar con interrupciones **P1IN**: Registro de solo lectura, su valor es el de la entrada del puerto.

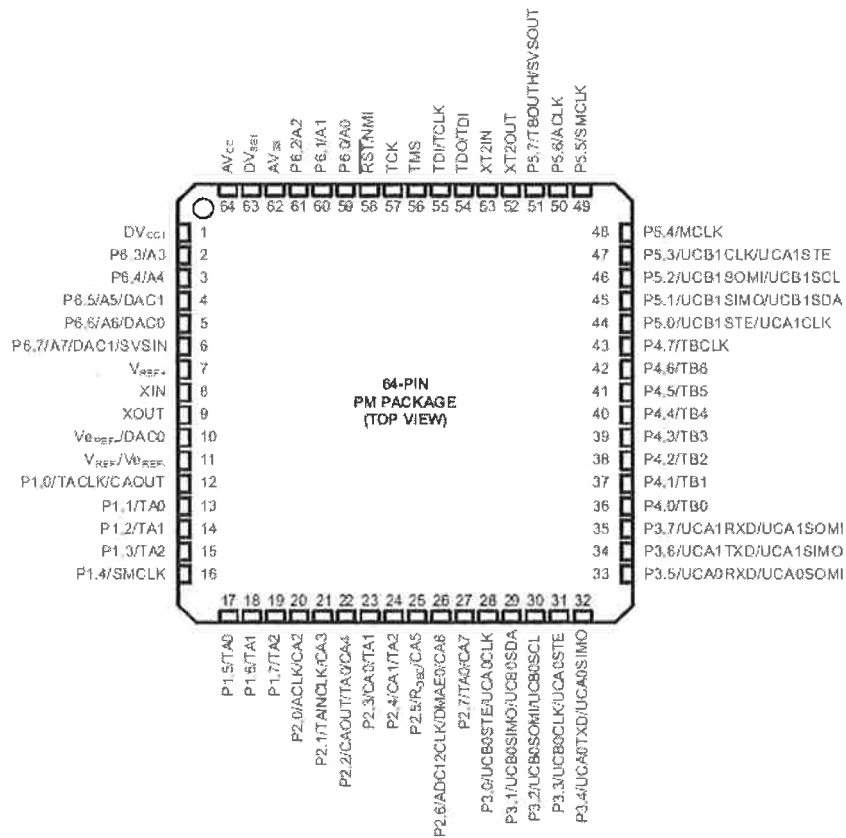


Figura 2.23 Designación de patillas del microcontrolador

Como ejemplo se muestran los registros del puerto uno, siendo este caso directamente extrapolable al resto de puertos.

- Registros de control de P1:

**Puerto P1 de entrada, P1IN:** de lectura devuelve los valores lógicos de las entradas si están configurados como entrada/salida digital. Este registro es de sólo lectura y volátil. No necesita ser inicializado porque sus contenidos son determinados por las señales externas.

**Puerto P1 de salida P1, P1OUT:** de escritura, envía el valor a ser manejado a cada pin si está configurado como una salida digital. Si el pin no es actualmente una salida, el valor se almacena en una memoria intermedia y aparece en el pin si más tarde se cambió para ser una salida. Este registro no se inicializa y por lo tanto se debe escribir P1OUT antes de configurar el pin como salida.

**Puerto P1 de Dirección, P1DIR:** si se escribe un 0 configura un pin como una entrada, que es la de defecto en la mayoría de los casos. Se escribe un 1 el pin se configura como una salida. Esto es para la entrada y salida digital, el registro funciona de forma diferente si otras funciones se seleccionan con P1SEL.

**Puerto P1 resistencia de habilitación, P1REN:** activación de un bit a 1 se activa un pull-up o pull-down resistencia en un pin. Pull-ups se utilizan a menudo para conectar un interruptor a una entrada como en la sección "Lectura de entrada del switch". Las resistencias están inactivas por defecto (0). Cuando el resistor se activa (1), el bit correspondiente del registro P1OUT selecciona si la resistencia pull de entrada es up para VCC (1) o down para VSS (0).

**Puerto P1 Selección, P1SEL:** selecciona cualquiera de las entradas/salidas digitales (0, por defecto) o una función alternativa (1). Registros adicionales pueden ser necesarios para elegir la función particular.

**Puerto P1 habilitación de interrupción, P1IE:** permite interrupciones cuando el valor de un pin de entrada cambia. Esta función se activa mediante el establecimiento de los bits apropiados de P1IE a 1. Las interrupciones son apagado (0) de forma predeterminada. El puerto en su conjunto comparte un único vector de interrupción aunque los pines pueden ser habilitados de forma individual.

**Puerto P1 interrupción selección de borde, P1IES:** puede generar interrupciones, ya sea en un flanco ascendente (0), cuando la entrada pasa de bajo a alto, o en un flanco descendente de alto a bajo (1). No es posible seleccionar las interrupciones en ambos bordes simultáneamente, pero esto no es un problema debido a que la dirección puede invertirse después de cada transición. Es necesario tener cuidado si se cambia la dirección mientras que las interrupciones están habilitadas porque una interrupción espuria puede

ser generada. Este registro no se inicializa y por lo tanto debe ser establecido antes de que las interrupciones estén habilitadas.

**Puerto P1 bandera de interrupción, P1IFG:** un bit se pone a 1 cuando la transición ha sido detectada en la entrada. Además, una interrupción se solicita si se ha activado. Estos bits también se puede configurar mediante software, que proporciona un mecanismo para la generación de una interrumpir por software.

## 2.9 EL MÓDULO DE RELOJ

El módulo de reloj del microcontrolador tiene tres fuentes de señal de reloj para alimentar otras tres señales que sincronizan la CPU y los periféricos

**Las fuentes de reloj son las siguientes:**

- **LFXT1CLK:** reloj de baja frecuencia / alta frecuencia, que puede usarse con cristales de baja frecuencia o con una fuente de reloj externa de 32,768 Hz. O con cristales estándar, resonadores o fuentes externas de reloj en el rango de 400KHz a 16MHz.
- **XT2CLK:** Oscilador de alta frecuencia opcional que puede utilizarse con cristales estándar, resonadores, o fuentes externas de reloj en un rango de 400KHz a 6MHz.
- **DCOCLK:** Oscilador interno digitalmente controlado (DCO).
- **VLOCLK:** Oscilador interno de muy baja potencia y de baja frecuencia típicamente de 12KHz.

**Con estas fuentes el módulo de reloj genera las siguientes tres señales:**

- **Reloj maestro (MCLK):** que alimenta la CPU y el sistema y se puede generar a partir de la señal de baja frecuencia, la de alta frecuencia o el oscilador digital; que pueden dividirse por un factor de 1, 2, 4 u 8.
- **Reloj de sub-sistema (SMCLK):** que alimenta a los periféricos seleccionados por software. Se puede generar a partir de la señal de baja frecuencia, la de alta frecuencia o el oscilador digital; que pueden dividirse por un factor de 1, 2, 4 u 8.
- **Reloj auxiliar (ACLK):** que alimenta a los periféricos seleccionados por software. Se genera a partir de la señal de baja frecuencia dividida por un factor de 1, 2, 4 u 8.

Esta flexibilidad en el sistema de reloj está especialmente indicada para ahorrar energía, ya que se puede ocupar un cristal de baja frecuencia, por ejemplo 32 KHz, para los periféricos, mientras la CPU se opera mediante el oscilador interno a una frecuencia que puede ir de los 800 KHz a los 8 MHz, dependiendo de la configuración y características del dispositivo en particular. Además para contrarrestar el efecto de corrimiento de frecuencia del oscilador interno se puede corregir periódicamente con la señal más estable proveniente del oscilador de baja frecuencia controlado por cristal.

El siguiente es un esquema del módulo generador de reloj:

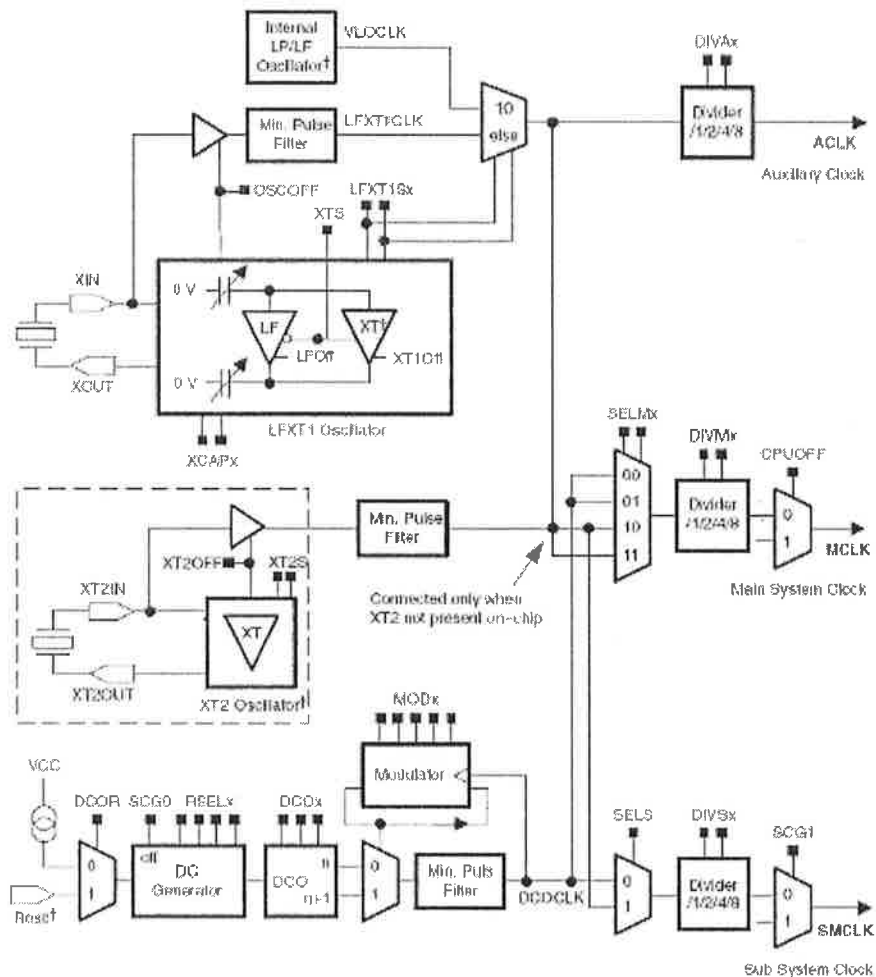


Figura 2.24: Módulo de reloj

Después de energizar el microcontrolador la fuente de reloj del MCLK y el SMCLK es proporcionada por el DCOCLK (~1.1MHz) y la fuente del ACLK es proporcionada por el LFXT1CLK en baja frecuencia con un capacitor interno de 6pF.

Los registros que configuran los módulos de reloj básico son el DCOCTL, BCSCTL1, BCSCTL2, y BCSCTL3. Estos registros pueden ser configurados o reconfigurados en cualquier momento.

Tabla 2-1 Oscilador de Cristal Oscilador LFXT1, Modo de alta Frecuencia <sup>(1)</sup>

PARAMETER	TEST CONDITIONS	V <sub>OC</sub>	MIN	TYP	MAX	UNIT	
f <sub>LFXT1,HF0</sub>	LFXT1 oscillator crystal frequency, HF mode 0	XTS = 1, LFXT1Sx = 0, XCAPx = 0	1.8 V to 3.6 V	0.4	1	MHz	
f <sub>LFXT1,HF1</sub>	LFXT1 oscillator crystal frequency, HF mode 1	XTS = 1, LFXT1Sx = 1, XCAPx = 0	1.8 V to 3.6 V	1	4	MHz	
f <sub>LFXT1,HF2</sub>	LFXT1 oscillator crystal frequency, HF mode 2	XTS = 1, LFXT1Sx = 2, XCAPx = 0	1.8 V to 3.6 V	2	10	MHz	
			2.2 V to 3.6 V	2	12		
			3 V to 3.6 V	2	16		
f <sub>LFXT1,HF,logk</sub>	LFXT1 oscillator logic-level square-wave input frequency, HF mode	XTS = 1, LFXT1Sx = 3, XCAPx = 0	1.8 V to 3.6 V	0.4	10	MHz	
			2.2 V to 3.6 V	0.4	12		
			3 V to 3.6 V	0.4	16		
OAHF	Oscillation allowance for HF crystals (see Figure 23 and Figure 24)	XTS = 1, XCAPx = 0, LFXT1Sx = 0, f <sub>LFXT1,HF</sub> = 1 MHz, C <sub>L,eff</sub> = 15 pF		2700		Ω	
				800			
				300			
C <sub>L,eff</sub>	Integrated effective load capacitance, HF mode <sup>(2)</sup>	XTS = 1, XCAPx = 0 <sup>(3)</sup>		1		pF	
	Duty cycle, HF mode	XTS = 1, XCAPx = 0, Measured at P2.0/ACLK, f <sub>LFXT1,HF</sub> = 10 MHz	2.2 V/3 V	40	50	60	%
				40	50	60	
f <sub>FaultHF</sub>	Oscillator fault frequency <sup>(4)</sup>	XTS = 1, LFXT1Sx = 3, XCAPx = 0 <sup>(5)</sup>	2.2 V/3 V	30	300	kHz	

(1) Para mejorar la EMI en el oscilador XT2 las pautas siguientes deben ser observados.

- (a) Mantener la traza entre el dispositivo y el cristal lo más corto posible.
  - (b) Diseñar una toma de tierra alrededor de los pines del oscilador.
  - (c) Prevenir la interferencia de otro reloj o líneas de datos en los pines del oscilador XIN y XOUT.
  - (d) Evite correr trazos de PCB por debajo o al lado de los pines XIN y XOUT.
  - (e) Utilizar materiales de montaje y praxis de evitar cualquier carga parasita en los pines del oscilador XIN y XOUT.
  - (f) Si el revestimiento de conformación se utiliza, se procurará que no induzca fuga capacitiva/resistiva entre los pines del oscilador.
  - (g) No pase la línea XOUT al JTAG para mantener el adaptador de programación en serie como se muestra en otra documentación. Esta señal ya no es necesario para el adaptador de programación en serie.
- (2) Incluye enlaces parásitos y paquete capacitivo (aproximadamente 2 pF por pin). Debido a que el PCB añade capacitancia adicional, se recomienda comprobar la carga correcta mediante la medición de la frecuencia ACLK. Para una instalación correcta, la capacidad de carga efectiva debe coincidir siempre con la especificación del cristal utilizado.
- (3) Requiere condensadores externos en ambas terminales. Los valores son especificados por los fabricantes de cristal.
- (4) Las frecuencias por debajo de la especificación MIN establecer la bandera de falla, las frecuencias por encima de la especificación MAX no la establece la bandera, y las frecuencias intermedias podrían establecer la bandera.
- (5) Medida con frecuencia de entrada de nivel lógico, pero también se aplica a la operación con los cristales.

## 2.10 TEMPORIZADORES

Existen tres temporizadores disponibles en el microcontrolador MSP430x2x, uno que realiza la función de *perro guardián* (watchdog). Un temporizador A que cuenta con 3 registros de captura o comparación y otro que cuenta con 7 registros de captura o comparación.

### 2.10.1 EL TEMPORIZADOR WATCHDOG

La principal función del temporizador de watchdog es reiniciar el procesador después de una falla en el software o al no producirse un evento esperado, al completarse un período de tiempo determinado (timeout).

Es decir, el programador selecciona un tiempo adecuado y dentro del programa reinicia periódicamente el temporizador, cuando el software se bloquee por alguna razón el procesador será reiniciado por el temporizador de watchdog al término del ciclo actual del temporizador.

Todo programa para este microcontrolador debe comenzar configurando o desactivando el temporizador de watchdog, ya que el microcontrolador se reinicia con esta función activada de forma predeterminada.

Si no se utiliza como temporizador de watchdog el temporizador se puede utilizar para generar intervalos de tiempo, por ejemplo para generar interrupciones.

El período del temporizador se fija seleccionando la fuente de reloj que puede ser el reloj auxiliar (ACLK) o el reloj de sub sistemas (SMCLK) y el factor de división que puede ser 62, 512, 8192 ó 32768. Los valores son configurados mediante el registro de control del temporizador de watchdog (WDTCTL).

### 2.10.2 LOS TEMPORIZADORES A Y B

Estos son temporizadores o contadores con registros de captura o comparación con la capacidad de generar múltiples salidas o interrupciones.

La principal diferencia entre los temporizadores A y B es que el segundo tiene la posibilidad de seleccionar la longitud del registro temporizador entre 8, 10, 12 ó 16 bits, en tanto el primero tiene un registro de longitud fija de 16 bits. Además el temporizador B tiene 7 registros de captura o comparación, en tanto el A tiene sólo 3.

Por cada uno de los registros de captura o comparación se tiene una etapa de salida. Además los registros de captura o comparación del temporizador B pasan por un buffer adicional y se pueden agrupar.

En modo de temporizador las fuentes del reloj pueden ser las señales ACLK, SMCLK o una señal externa a través de los pines TACLK (TBCLK) o INCLK. A su vez la fuente de reloj elegida puede ser dividida por un factor de 2, 4 u 8 mediante los bits IDx del registro de control TACTL (TBCTL). Puede funcionar en uno de cuatro modos, seleccionado con los bits MCx.



Tabla 2-2 Modos del contador

MCx	Modo	Descripción
00	Detenido	El temporizador está detenido
01	Arriba	El temporizador cuenta desde cero hasta TACCRO(TBCLO)
10	Continuo	Cuenta desde cero hasta 0xFFFF para el temporizador A o los bits indicados por TBCNTLx y parte desde 0 nuevamente
11	Arriba y abajo	Cuenta desde cero hasta 0xFFFF para el temporizador A o los bits indicados por TBCNTLx y hacia cero nuevamente

## 2.11 MODOS DE BAJO CONSUMO

La familia MSP430 está diseñado para aplicaciones de ultra-baja potencia y ofrece cinco modos de bajo consumo, de las cuales dos son rara vez utilizados. Los modos más importantes son: El modo activo, LPM0, LPM3 y LPM4. El cuadro comparativo muestra el consumo de corriente típica de cada modo para  $V_{cc} = 3\text{ V}$  y  $V_{cc} = 2.2\text{ V}$ , mientras que el DCO está funcionando a 1MHz y el LFXT1\_2 a 32 KHz de un cristal.

Los cinco modos de bajo consumo son seleccionables por software. Un evento de interrupción puede despertar el chip de cualquiera de ellos, el servicio de solicitud de interrupción y restaurar de nuevo al modo de bajo consumo al regreso de la rutina de interrupción. El MSP430 se inicia en modo activo con la CPU, los relojes y todos los módulos habilitados están activas. Se consume  $\approx 300\mu\text{A}$  a  $V_{cc} = 3\text{ V}$ , pero la corriente se puede reducir a  $\approx 200\mu\text{A}$  mediante la ejecución de la MCU en 1,8 V, que es su menor tensión de alimentación. Es racional para hacerlo, ya que disminuye la corriente y alarga la capacidad de la batería. Las características esenciales de los modos de funcionamiento se indican en la tabla 2-4, junto con el consumo de corriente a 1 MHz y la frecuencia de 32 KHz DCO LFXT1 a  $V_{cc} = 2.2\text{ V}$ .

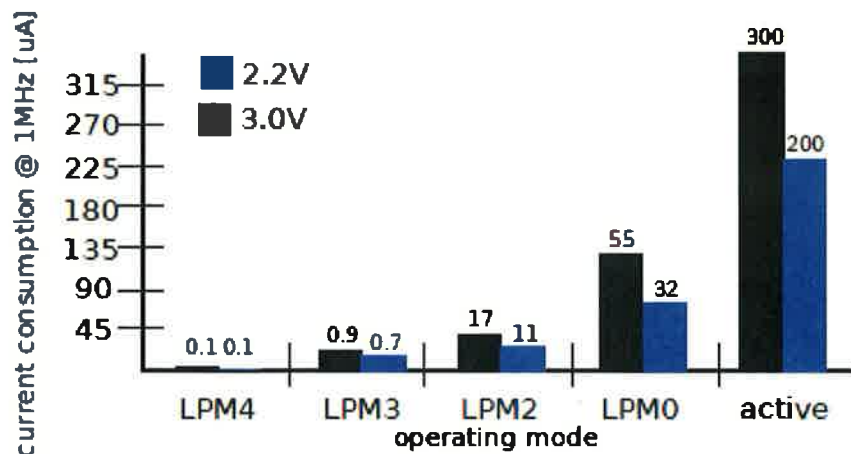


Figura 2.25: Consumo de corriente típica del MSP430

Cuando la CPU se requiere, el modo activo debe ser utilizado. Una interrupción cambia automáticamente el microcontrolador de cualquier modo de bajo consumo al modo activo, y vuelve de nuevo después que la interrupción es atendida. Sin embargo, a veces puede ser deseable permanecer en el modo activo después de despertar del modo de bajo consumo. Probablemente, la forma más importante es la LPM3, cuando solo el ACLK permanece activa. Se utiliza cuando el dispositivo tiene que despertar con regularidad para ejecutar rápidamente una serie de comandos antes de volver al modo de bajo consumo. El DCO deberá estar iniciado, esto toma más tiempo, que a partir de LPM0. Está a unos 1.1 $\mu$ s, que es un intervalo muy pequeño.

El MSP430 tiene un modo activo y cinco modos de operación de bajo consumo seleccionables por software. Una interrupción puede despertar el dispositivo desde cualquiera de los cinco modos de bajo consumo.

Los siguientes cinco modos de operación se pueden configurar por software:

Tabla 2-3: Modos de operación y estado del reloj

Modo	CPU & Estado del Reloj
Activo	CPU está activo Todos los relojes están activos
LPM0	CPU está deshabilitado ACLK y SMLK permanecen activos MCLK está deshabilitado
LPM2	CPU está deshabilitado MCLK y SMCLK permanecen activos DCO generador –dc permanece habilitado ACLK permanece activo
LPM3	CPU está deshabilitado MCLK y SMCLK están deshabilitados DCO generador –dc está deshabilitado ACLK permanece activo
LPM4	CPU está deshabilitado MCLK y SMCLK están deshabilitados ACLK está deshabilitado DCO generador –dc está deshabilitado El Oscilador del cristal se detiene

## 2.12 INTERRUPCIONES

**Interrupciones:** Por lo general generados por hardware (aunque pueden ser iniciadas por software) y a menudo indican que ha ocurrido un evento que requiere una respuesta urgente. Un paquete de datos podría haber sido recibido, por ejemplo, y necesita ser procesada antes de que el próximo paquete llegue. El procesador deja de hacer lo que estaba haciendo, almacena información suficiente (el contenido del contador de programa y registro de estado) para reanudar más adelante y ejecuta una rutina de servicio de interrupción (ISR). Se vuelve a su actividad anterior cuando el ISR ha concluido. Así, un ISR es algo así como una subrutina llamada por hardware (en un momento impredecible) en lugar de software. Un segundo uso de las interrupciones, que es particularmente importante en el MSP430, es despertar el procesador de un estado de bajo consumo.

La mayoría de las interrupciones son enmascarable, lo que significa que sólo son eficaces si la interrupción general (GIE) se habilita, el bit se pone a 1 en el registro de estado (SR). Se ignora si GIE se limpia.

Por lo tanto el bit de habilitación en el módulo y GIE se debe escribir a 1 para generar las interrupciones. Las interrupciones (no) enmascarables no puede ser suprimida al limpiar GIE. La razón de los paréntesis (no) es que estas interrupciones también requieren los bits que se escriban a 1 en función especial o registros periféricos para que puedan ser

habilitados. Así, incluso las interrupciones (no) enmascarable se puede desactivar y de hecho todos están desactivados por defecto.

El MSP430 usa vectores de interrupción, lo cual significa que la dirección de cada ISR- su vector se almacena en una tabla vector en una dirección de memoria definida. En la mayoría de los casos cada vector es asociado con una única interrupción pero algunas fuentes comparten un vector.

El propio ISR debe localizar la fuente de interrupciones que comparte el vector. Por ejemplo, un vector TAIFG comparte un vector con las interrupciones captura/comparación para todos los canales del Timer\_A distintos de 0. El canal 0 tiene su propia bandera de interrupción TACCR0 CCIFG y vector separado.

Cada vector de interrupción tiene una prioridad distinta, que se utiliza para seleccionar qué vector se toma si más de una interrupción se activa cuando el vector es buscado. Las prioridades se fijan en hardware y no puede ser cambiado por el usuario. Están dados simplemente por la dirección de la Vector: Una dirección más alta significa una mayor prioridad. El vector de reset tiene la dirección 0xFFFFE, lo que le da la máxima prioridad, seguida por 0xFFFFC para el único de interrupción no enmascarable. Los vectores de las interrupciones enmascarables dependen de los periféricos en un dispositivo en particular y se muestran en una tabla de direcciones vector de interrupción en la ficha técnica.

### **2.12.1 INTERRUPCIÓN EXTERNA**

Puertos P1 y P2 pueden solicitar una interrupción cuando el valor de una entrada cambia. Esta es una de las pocas interrupciones que permanece activa en LPM4 y es por lo tanto útil para despertar a la CPU en equipos portátiles que se encuentra inactivo durante un largo tiempo.

Las Interrupciones para el puerto P1 son controlados por los registros P1IE y P1IES, mencionado anteriormente, y de manera similar para puerto P2. Hay un único vector para cada puerto, por lo que el usuario debe comprobar P1IFG para determinar el bit que causó la interrupción. Este bit debe ser limpiado explícitamente, no se produce automáticamente como con las interrupciones que tienen una sola fuente.

La dirección de la transición que causa la interrupción se puede cambiar en P1IES en cualquier momento por el programa. Esto es útil si ambos bordes de un pulso deben ser detectados, por ejemplo, cuando se pulsa un botón y se suelta. Existe el peligro de que interrupciones falsas se puedan generar, así que es una buena idea desactivar esta interrupción, cambia P1IES, y borrar las banderas falsas en P1IFG antes de rehabilitar la interrupción. De hecho, en la práctica esto no debería ser un problema si se cambia la dirección de la manera más obvia. Por ejemplo, el puerto puede esperar para una transición de bajo-a-alto, mientras que la entrada es baja. Una interrupción es solicitada cuando la entrada es alta. La sensibilidad se cambia entonces de alto-a-bajo para detectar el borde próximo.

En la Tabla 2-4 se puede ver el vector de interrupciones del microcontrolador MSP430x2xx.

Tabla 2-4 Vector de interrupciones

FUENTE DE INTERRUPCIÓN	FLAGS AFECTADOS	TIPO DE INTERRUPCIÓN	DIRECCIÓN	PRIORIDAD
Power-up External reset Watchdog Timer+ Flash key violation Pc out-of-range <sup>(1)</sup>	PORIFG RSTIFG WDTIFG KEYV See <sup>(2)</sup>	Reset	0FFFEh	31,highest
NMI Oscillator fault Flash Memory access violation	NMIIFG OFIFG ACCVIFG <sup>(2)(3)</sup>	(Non)maskable (Non)maskable (Non)maskable	0FFFCh	30
Timer_B7	TBCCR0 CCIFG <sup>(4)</sup>	Maskable	0FFFAh	29
Timer_B7	TBCCR1 to TBCCR6 CCIFGs TBIFG <sup>(2)(4)</sup>	Maskable	0FFF8h	28
Comparator_A+	CAIFG	Maskable	0FFF6h	27
Watchdog Timer+	WDTIFG	Maskable	0FFF4h	26
Timer_A3	TACCR0 CCIFG <sup>(4)</sup>	Maskable	0FFF2h	25
Timer_A3	TACCR1 CCIFG TACCR2 CCIFG <sup>(2)(4)</sup>	Maskable	0FFF0h	24
USCI_A0/USCI_B0 receive USCI_B0 I2C status	UCA0RXIFG, UCB0RXIFG <sup>(2)(5)</sup>	Maskable	0FFEeh	23
USCI_A0/USCI_B0 transmit USCI_B0 I2C receive/transmit	UCA0RXIFG, UCB0RXIFG <sup>(2)(6)</sup>	MAskable	0FFECh	22
ADC12	ADCIFG <sup>(2)(4)</sup>	Maskable	0FFEAh	21
		Maskable	0FFE8h	20
I/O port P2 (eight flags)	P2IFG.0 to P2IFG.7 <sup>(2)(4)</sup>	Maskable	0FFE6h	19
I/O port P1 (eight flags)	P1IFG.0 to P1IFG.7 <sup>(2)(4)</sup>	Maskable	0FFE4h	18
USCI_A1/USCI_B1 receive USCI_B1 I2C status	UCA1RXIFG, UCB1RXIFG <sup>(2)(5)</sup>	Maskable	0FFE2h	17
USCI_A1/USCI_B1 transmit USCI_B1 I2C receive/transmit	UCA1RXIFG, UCB1RXIFG <sup>(2)(6)</sup>	Maskable	0FFE0h	16
DMA	DMA0IFG, DMA1IFG, DMA2IFG <sup>(2)(4)</sup>	Maskable	0FFDEh	15
DAC12	DAC12_0IFG, DAC12_1IFG <sup>(2)(4)</sup>	Maskable	0FFDCh	14
See <sup>(7)(8)</sup>			0FFDAh to 0FFC0h	15 to 0,lowest

(1) Un reset se genera si la CPU intenta obtener instrucciones dentro del rango de registro de dirección de memoria del módulo (0h a 01FFh) o desde dentro de los rangos de direcciones no usadas.

(2) Banderas de múltiples fuentes

(3) (no) enmascarable: el bit individuo interrumpir-habilitación puede desactivar un evento de interrupción, pero las interrupciones generales de habilitación de interrupción no se puede.

(4) Banderas de interrupción están localizadas en el módulo.

(5) En modo SPI: UCB0RXIFG. En modo I<sup>2</sup>C mode: UCALIFG, UCNACKIFG, ICSTTIFG, UCSTPIFG.

(6) In UART/SPI mode: UCB0TXIFG. In I2C: UCB0RXIFG, UCB0TXIFG.

(7) La dirección 0FFBEh se utiliza como clave de seguridad bootstrap (BSLSKEY).

A 0AA55h en este lugar deshabilita el BSL completamente.

A cero deshabilita el borrado de la flash si una contraseña no válida se suministra

(8) Los vectores de interrupción en las direcciones 0FFDAh a 0FFC0h no se usan en este dispositivo y se pueden usar para regular el código del programa si es necesario.

### 3.1 DIAGRAMA DE FLUJO DE LA METODOLOGÍA

En figura 3.1 se presenta el diagrama de la metodología realizada para el proyecto Sistema de Adquisición de Datos.

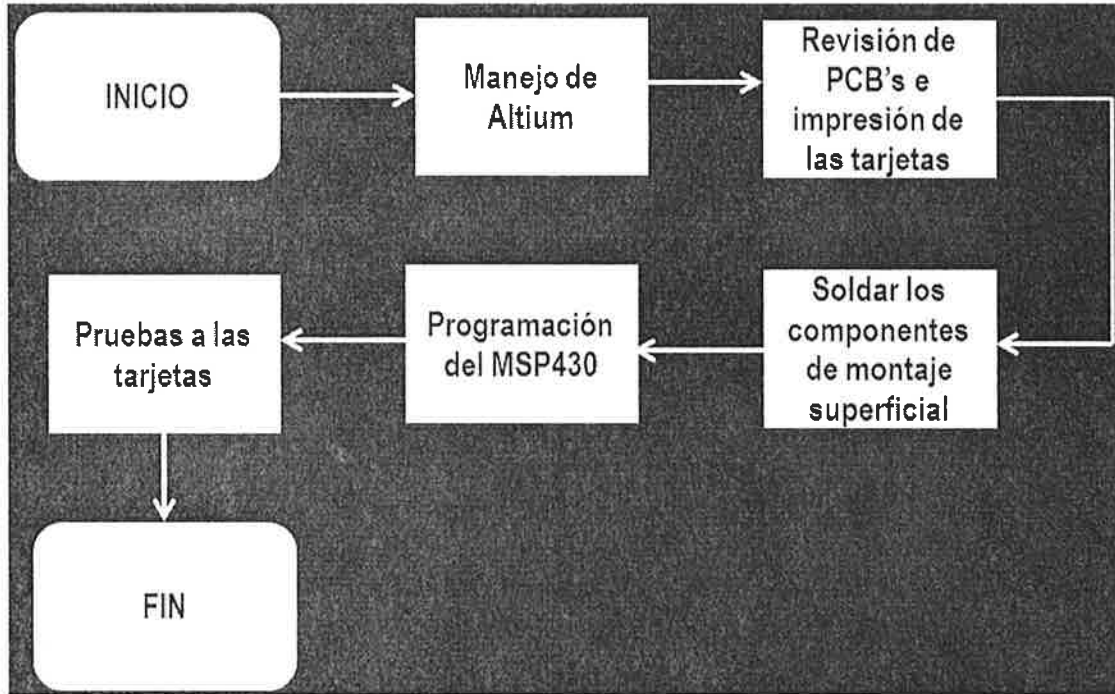


Figura 3.1: Diagrama de flujo

A continuación se especifican cada uno de los puntos correspondientes al diagrama de flujo anterior.

a) El primer punto consiste en conocer Altium Designer principalmente el cómo crear proyectos nuevos o existentes, librerías, manipular esquemáticos y PCB.

b) Posteriormente se lleva a cabo la revisión detallada de los PCB's, corrección de posibles errores presentes en las mismas, para posteriormente enviar a imprimir las tarjetas.

c) El siguiente paso consiste en soldar los componentes de montaje superficial en las tarjetas, además, elaborar los cables para hacer las pruebas correspondientes, tales como cables de alimentación, cables para conectar los sensores, el cable para la programación de los MSP430 y el cable que conecta a los sensores.

d) Una vez realizado todo lo referente a la corrección y soldado de componentes de montaje superficial en las tarjetas, lo siguiente es aprender a manejar los timer e interrupciones externas para enviar los pulsos de sincronía mediante el MSP430.

e) Finalmente se prueban las tarjetas, alimentándolas con el voltaje y corriente correspondientes, así como, programar el código elaborado para enviar los pulsos de sincronía.

# **CAPÍTULO 4**

## **DESARROLLO DEL PROYECTO**

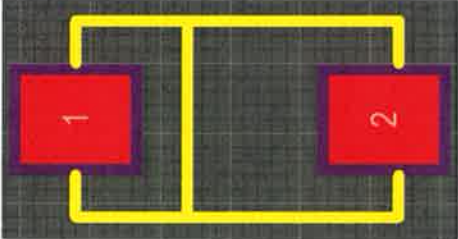
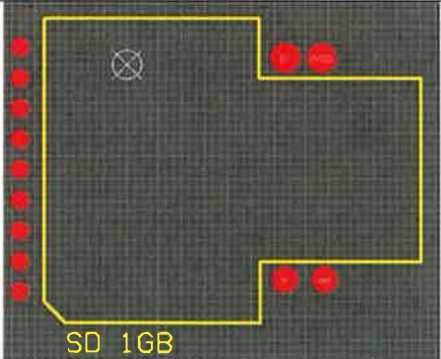


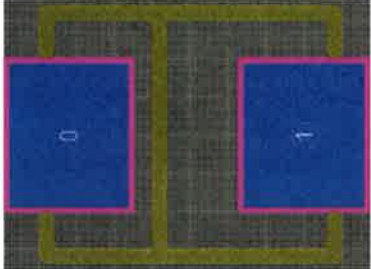
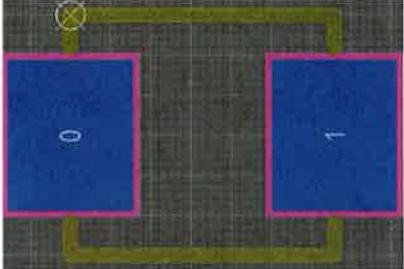
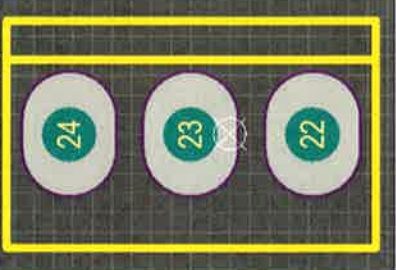
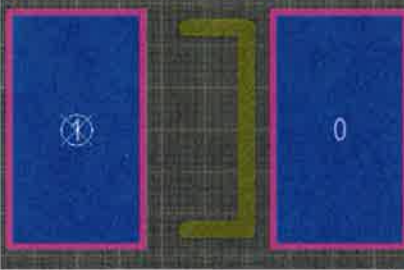
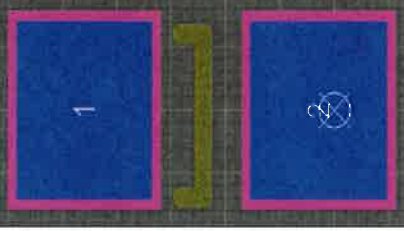
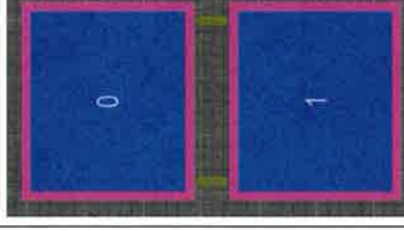
## 4.1 REVISIÓN Y CORRECCIÓN DE LOS PCB.

En esta etapa se realiza la comparación de los PCB's con el esquemático, tomando en cuenta los siguientes puntos:

- Conexiones:** en este punto se realiza la corrección de las conexiones entre los dispositivos usando el datasheet de los dispositivos electrónicos.
- Identificación de los dispositivos utilizados:** consiste en recopilar la información correspondiente al nombre y valor de cada dispositivo utilizado en los PCB's, para posteriormente etiquetarlos y hacer la lista de materiales para su compra, los parámetros que se toman en cuenta al hacer la compra de los materiales son voltaje máximo, corriente máxima, coeficiente de temperatura máxima, precio y cantidad en existencia.
- Ángulos de 90 grados o menores a 45 grados:** se deben evitar estos ángulos para se tenga un buen funcionamiento de las tarjetas.
- Separación entre pistas:** Las pistas deben estar lo más separado posible, de tal manera que se utilice al máximo el espacio disponible en las tarjetas, ya que, esto evita posibles saltos de voltaje entre las pistas.
- Reguladores de voltaje:** estos dispositivos deben estar alimentados correctamente, ya que, ellos serán los encargados de alimentar todos los dispositivos electrónicos, además se debe revisar la correcta conexión de la alimentación y tierra en cada dispositivo.
- Footprint de componentes creados:** una vez identificado los errores en las tarjetas, estos se corrigen, existen componentes que no se conectaron, para ello se deben crear el footprint de cada componente correspondiente, en la tabla 4.1 se muestran todos los footprint de los componentes creados para las tarjetas.

TABLA 4.1: FOOTPRINT DE COMPONENTES CREADOS EN ALTIUM

COMPONENTE	COMPONENTE
	
i) DIODO 1N4148:	ii) MEMORIA SD 1GB

	
ii) DIODO DE 1 AMPERE	iii) CRYSTAL
	
iv) JTAG DE 3 TERMINALES	v) CAPACITOR ELECTROLÍTICO.
	
vi) CAPACITOR CERÁMICO	vii) CAPACITOR CERÁMICO 0805

Una vez corregidos los PCB's se enviaron los archivos correspondientes para la impresión de las tarjetas de cuatro capas cada una, esto debido a que el centro no cuenta con el equipo necesario para poder imprimirlas.

## 4.2 PROCESO PARA SOLDAR LOS MSP430

A continuación se explica el proceso que se realiza para soldar componentes de montaje superficial.

El proceso que se utilizó para soldar los microcontroladores de montaje superficial, es el que a continuación se explica:

**A) Estañado del área donde se va a colocar el MSP430:** Puesto que se tienen que soldar componentes sensibles a cargas electrostáticas se debe utilizar una pulsera antiestática. Una vez puesta la pulsera, se le aplica flux al área de la tarjeta donde se va a estañar el microcontrolador, tal como se muestra en la figura 4.1, la temperatura del cautín debe ser de 65°C, para comenzar a estañar el área en donde se va a colocar el microcontrolador se toma una cantidad muy pequeña de soldadura, a esta se le aplica un poco de pasta y se comienza a pasar el cautín de tal manera como si se peinaran las pistas, se debe tener en cuenta de no dejar la punta del cautín por más de 10 segundos en cada pista, de lo contrario se puede levantar, además, evitar la formación de montículos, puesto que, dificulta el alineamiento y facilita los cortos, por lo tanto, la soldadura debe ser completamente lisa. Una vez estañada el área del MSP430 se aplica alcohol isopropílico para limpiar la placa y comenzar a colocar el microcontrolador.

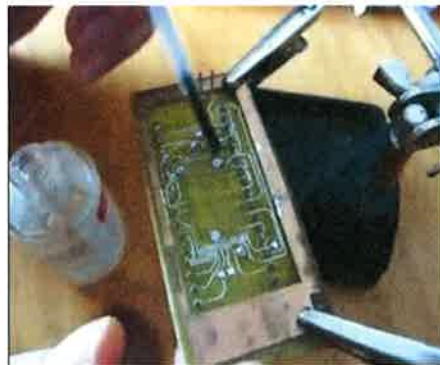


Figura 4.1: Aplicando flux a la tarjeta

**B) Estañado de los pines del MSP430:** se procede estañar las patitas del microcontrolador, se le aplica pasta a todos los pines del MSP430 y se toma una pequeña cantidad de soldadura con el cautín y se procede a aplicar la soldadura en cada uno de los pines de forma como si se estuviesen peinando los pines, como se muestra en la figura 4.2. Se debe tener cuidado de no dejar mucho tiempo el cautín en la los pines, de lo contrario se pueden doblar, así como aplicar demasiada soldadura al cautín lo cual podría juntar más de dos pines y esto ocasionaría corto circuito al microcontrolador.



Figura 4.2: Estañando los pines del MSP430

**C) Soldando el MSP430 en la tarjeta:** una vez estañada las pistas, así como los pines del microcontrolador, se aplica una pequeña cantidad de Resistol en pasta a la parte de abajo del microcontrolador para que al colocarlo en la tarjeta quede firme temporalmente, a continuación se alinean los pines del microcontrolador con las pistas, una vez alineadas como en la figura 4.3 y 4.4, se procede a calentar cada una de las esquinas del microcontrolador, ya que el microcontrolador no se mueva, se coloca el cautín a cada uno de los pines del dispositivo teniendo precaución de no dejar la punta del cautín por más de 3 segundos, esto se muestra en la figura 4.5, en caso que falte estaño en alguno de los pines, se aplica de manera cuidadosa un poco más de estaño en los pines que faltan.

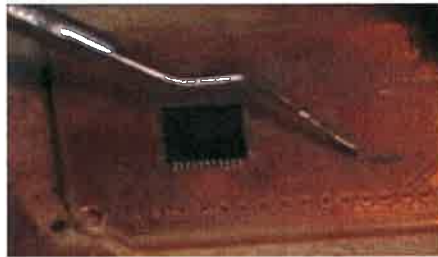


Figura 4.3: Estañando pistas



Figura 4.4: Alineando el MSP430 con las pistas



Figura 4.5: Estañado cada pin del MSP430

**D) Aplicación de aire caliente:** se pasa un caudín de aire caliente por todos los pines del microcontrolador, esto se muestra en la figura 4.6. La temperatura adecuada es de 30°C a una presión de aire de 10.



Figura 4.6: Aplicando calor a los pines del MSP430

**E) Limpieza de la tarjeta:** Una vez terminado de aplicar aire caliente al dispositivo, se le aplica nuevamente alcohol isopropílico en la tarjeta y se limpia cuidadosamente con un poco de papel para quitar los residuos de pasta, el microcontrolador debe quedar como se muestra en la figura 4.7.

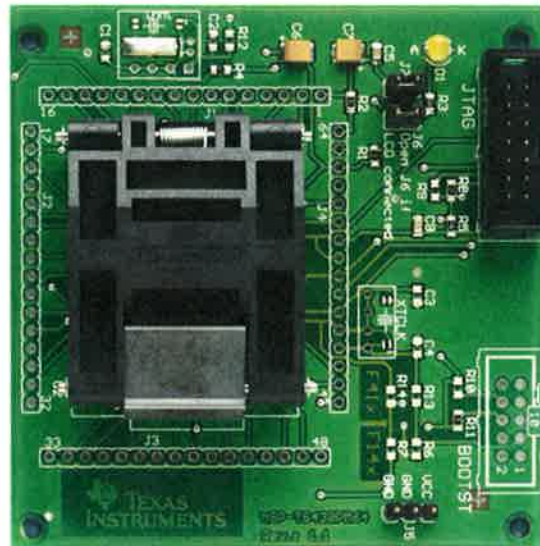


Figura 4.7 MSP430 soldado

Una vez soldado los microcontroladores, se procede a soldar los demás componentes comenzando por los más pequeños tales como resistencias, capacitores cerámicos, y demás circuitos integrados, se continúa con los capacitores electrolíticos y finalmente se colocan los pines hembra y macho.

### 4.3 PROGRAMACIÓN DEL MSP430

Para programar el MSP430 se utilizó el módulo MSP-TS430PM64 de Texas Instruments



MSP430 64-pin Target Board  
MSP-TS430PM64



Figura 4.8 Tarjeta MSP-TS430PM64



Figura 4.9 Programador para MSP430



El diagrama de flujo de la figura 4.10, explica la forma de cómo se envían los pulsos de sincronía del MSP430, el modo de envío de estos pulsos se selecciona por un jumper de la tarjeta, en caso de que el jumper este en alto, los datos de sincronía que se van a enviar va a ser mediante el modo por interrupción TMRO, en caso contrario, si el jumper está en bajo, los pulsos de sincronía que se van a enviar del MSP430, será mediante el modo de interrupción externa.

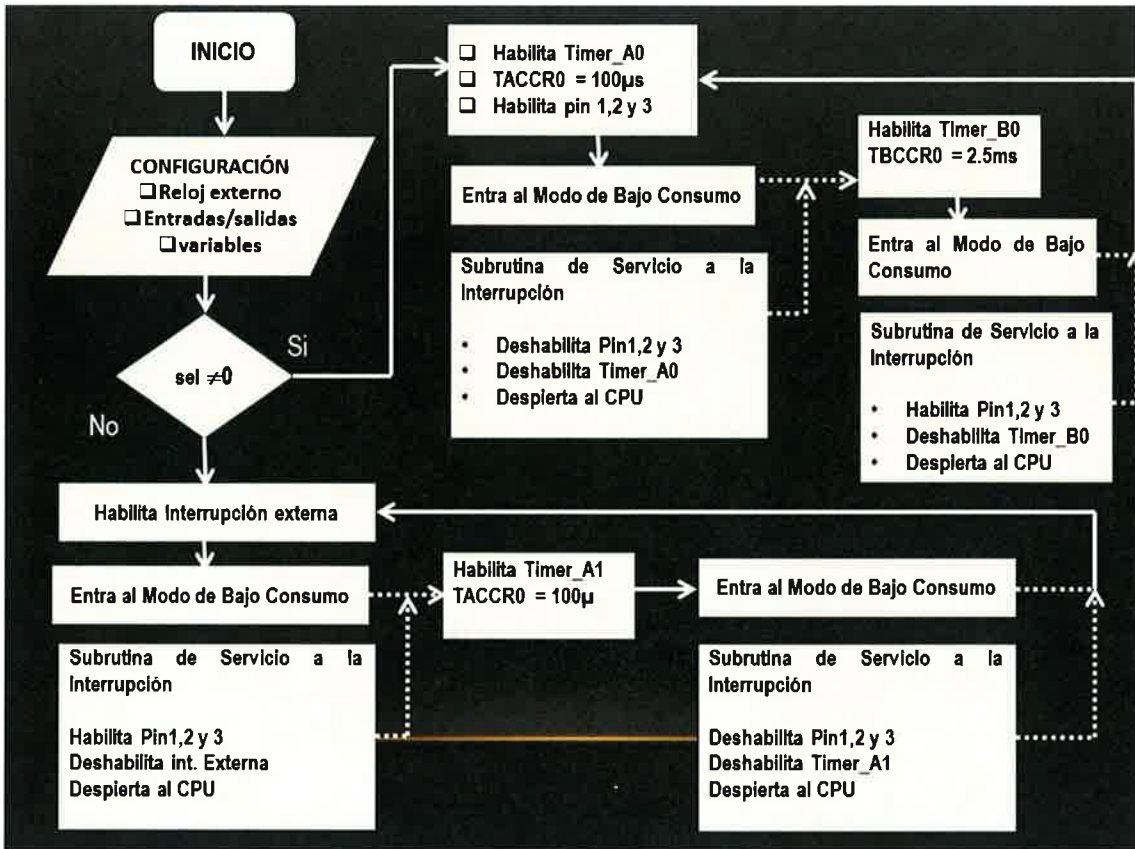


Figura 4.10 Diagrama de flujo del código desarrollado para la sincronía del MSP430

#### 4.4 PRUEBAS REALIZADAS A LAS TARJETAS

**Prueba a)** Esta prueba consistió en alimentar la tarjeta A con un voltaje de 12V a 40mA, en este punto se realiza la verificación de los voltajes de salida de los reguladores, así como de posibles corto circuito que se puedan presentar en la tarjeta para corregirlos, se programa el MSP430 con el código elaborado.



Figura 4.11: Voltaje de alimentación de la tarjeta A

**Prueba b)** Se alimentó la tarjeta B con dos voltajes, uno de 9V y otro de 5V con una corriente de 40mA, así como la verificación de posibles cortos circuitos en la tarjeta.

**Prueba c)** Una vez verificadas las tarjetas A y B, se procede a ensamblar ambas tarjetas tal como se muestra en la figura 4.12, por lo que, esta prueba consistió en conectar los sensores inductivos en la tarjeta A, y proceder a alimentar ambas tarjetas con un voltaje de 12V a 90mA, además, se observaron las señales de salida correspondientes a los sensores en el osciloscopio, así como la señal de sincronía que proporciona el MSP430 programado.



Figura 4.12: Tarjetas conectadas


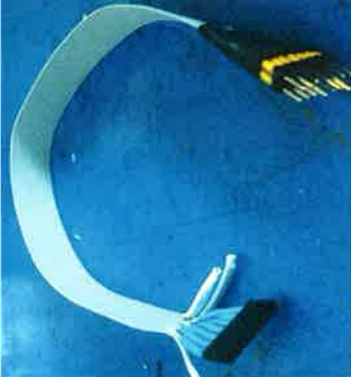




## 4.5 EQUIPOS Y MATERIALES UTILIZADOS

Para soldar los componentes en las tarjetas, así como, para la elaboración de los cables de prueba se utilizaron los siguientes materiales que se muestran en la tabla 4.2:

TABLA 4.2: MATERIAL USADO PARA SOLDAR

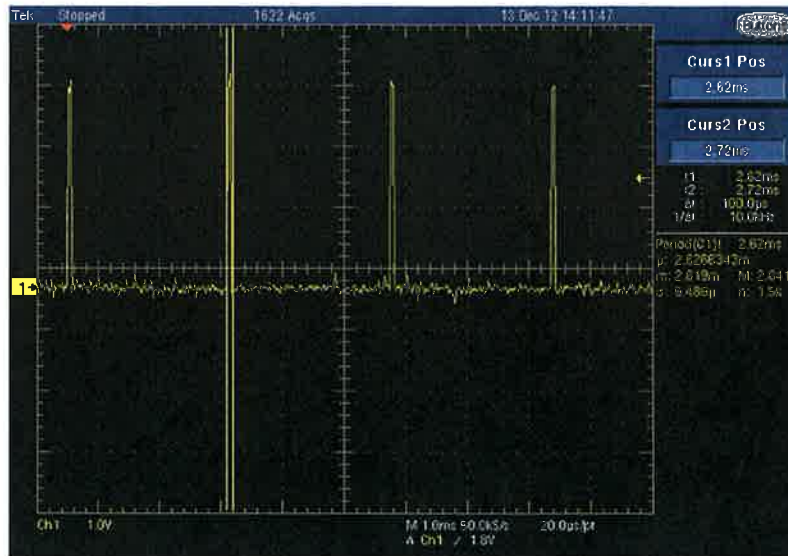
		
<p>a) Cautín con regulador de temperatura</p>	<p>b) Soldadura</p>	<p>c) Pasta</p>
		
<p>d) Alcohol isopropílico</p>	<p>e) Pinzas de punta</p>	<p>f) Pistola de aire caliente</p>
		
<p>g) Pinzas para componentes SMD</p>	<p>h) Multímetro</p>	<p>i) Osciloscopio fluke</p>

		
<p>a) Pulsera antiestática</p>	<p>b) Cable para programar</p>	<p>c) Cable de alimentación</p>
		
<p>m) Fuente de Voltaje y corriente</p>		

# **CAPÍTULO 5**

## **RESULTADOS Y CONCLUSIONES**

Los resultados obtenidos al probar las tarjetas con los voltajes y corrientes, fueron los diseñados, puesto que, los reguladores de voltaje de las tarjetas proporcionan los voltajes y corriente necesaria para alimentar todos los dispositivos y sensores conectados. La señal obtenida del código programado en el MSP430 correspondiente a los pulsos de sincronía se pueden observar en la figura 5.1.



5.1 Señal de sincronía de los MSP430

Se puede concluir que este trabajo aporta una parte del Sistema de Adquisición de Datos para el Robot de Inspección de ductos, el código desarrollado para enviar los pulsos de sincronía mediante el MSP430 puede funcionar en dos modos: mediante interrupción por TMRO o interrupción externa, este se puede elegir por medio de un jumper que tiene una de las tarjetas, además, los MSP430 de las tarjetas desarrolladas durante este proyecto están listas para ser programados con los códigos correspondientes para comenzar con la adquisición de datos, por otro lado, en el presente trabajo no se muestran los diagramas eléctricos, ni el PCB del mismo, debido a que forma parte de un proyecto interno de este centro de investigación.

## **BIBLIOGRAFÍA**

- [1]MSP430 MICROCONTROLLER BASIC JOHN DAVIES (2008)
- [2]USER'S GUIDE MSP430X2XX FAMILY (DECEMBER 2004)
- [3]MSP430x241x, MSP430x261x MIXED SIGNAL MICROCONTROLLER (JUNE 2007)
- [4]TÉCNICAS DE DISEÑO CON MICROCONTROLADOR MSP430 JAIME A. ZÚÑIGA CAMIRUAGA (DICIEMBRE 2003) TESIS.

Tutoriales de Altium Designer obtenidos de la página [www.altium.com](http://www.altium.com)

- [5]TUTORIAL SETTING UP THE PCB.
- [6]TUTORIAL PCB LIBRARY EDITOR.
- [7]TUTORIAL DESIGN RULES