



**UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO**

ACFACULTAD DE INGENIERÍA

*MÁQUINA PARA ENSAYOS DE FLEXIÓN EN MUESTRAS DE
PLANTAS DE DIMENSIONES PEQUEÑAS.*

T E S I S

QUE PARA OBTENER EL TÍTULO DE

INGENIERO MECATRÓNICO

P R E S E N T A N :

RICARDO ROMÁN CUMPLIDO AGUIRRE
ORLANDO IVAN LÓPEZ HERNÁNDEZ
LUIS ALBERTO VÁZQUEZ PÉREZ

DIRECTOR DE TESIS:
M.I. RIGOBERTO NAVA SANDOVAL



MÉXICO, D. F.

2009

A mis padres

Con un gran cariño y admiración a ustedes por el apoyo incondicional que me han brindado a lo largo de mi vida. Por guiarme a través de consejos y, sobre todo, del buen ejemplo.

Este logro y todos los demás son también de ustedes.

Muchas gracias.

A mis hermanos

Por su ayuda, comprensión y compañía. Gracias Mariana y Jorge.

Un agradecimiento especial al M.I. Rigoberto Nava por su apoyo brindado durante la realización de este trabajo.

A todas aquellas personas que, de una u otra forma, han contribuido con mi desarrollo personal y formación profesional.

Ricardo

Junio de 2009

A mis padres, quienes supieron guiar mis pasos a lo largo de mi vida con el único objetivo de verme convertido en un hombre de bien y a quienes espero jamás defraudar, ya que la mejor herencia que pudieron haberme dejado es el darme las herramientas necesarias para terminar mi carrera con éxito, con lo cual tendré más armas para enfrentar los obstáculos que se vayan presentando a lo largo de mi vida.

También quiero agradecer a todas aquellas personas que estuvieron apoyándome en algún momento de mis estudios, quienes nunca perdieron la fe en que podría alcanzar el objetivo planteado al ingresar a la facultad de ingeniería de la UNAM y de quienes sus palabras de aliento siempre fueron un apoyo extra para no claudicar en los momentos difíciles que tuve que enfrentar a lo largo de los 5 años que duró la carrera.

Con cariño y respeto.

Orlando Iván

Junio de 2009

Dedico este trabajo, a todas aquellas personas que contribuyeron a este logro en mi vida; a todos aquellos que confiaron en mí y me apoyaron con un consejo, un abrazo e incluso un regaño; y en especial quiero agradecer a:

Dios,

Por darme la fuerza necesaria para salir adelante, por llenarme de bendiciones y alegría a través de mi familia y seres queridos.

A mi madre,

Quien con su esfuerzo y trabajo me ha dado la oportunidad de alcanzar mis metas; quien ha sido ejemplo y motivación en mi vida; quien soñó y luchó conmigo para alcanzar esta meta; Mamá, este logro es tuyo.

A mis hermanos,

Por brindado su cariño y apoyo; por apoyarme en los momentos más difíciles; y sobre todo por la felicidad y alegrías compartidas. Hermanos, sin ustedes esto no hubiera sido posible.

A todos mis tíos,

Quienes siempre confiaron en mí y me motivaron a seguir adelante.

A mis padrinos,

Quienes me han apoyado de manera especial a lo largo de toda mi vida.

A mis amigos,

Por compartir y vivir momentos especiales conmigo.

A todos ustedes pues este logro no es solo mío, sino un reflejo del trabajo de todos.

¡MUCHAS GRACIAS!

**Luis Alberto
Junio de 2009**

Tabla de contenido

INTRODUCCIÓN.....	4
OBJETIVOS	6
JUSTIFICACIÓN.....	7
METODOLOGÍA.....	9
REQUERIMIENTOS Y CARACTERÍSTICAS DE INGENIERÍA	11
ELEMENTOS DEL SISTEMA.....	13
MECÁNICA.....	14
CONTROL	14
COMPUTACIÓN.....	14
ELECTRÓNICA	15
CAPÍTULO 1. MARCO TEÓRICO	16
1.1 ESFUERZO	16
1.2 DEFORMACIÓN	18
1.3 FLEXIÓN.....	19
1.4 ENSAYO DE FLEXIÓN.....	21
1.5 MÓDULO DE ELASTICIDAD.....	21
1.6 DETERMINACIÓN DEL ESFUERZO FLECTOR	23
1.7 DETERMINACIÓN DEL MÓDULO DE YOUNG EN FLEXIÓN.....	24
CAPÍTULO 2. SISTEMA MECÁNICO	26
2.1 GUÍA	27
2.2 CORREDERA.....	28
2.3 CELDA DE CARGA.....	29
2.4 SOPORTE PARA LA CARCASA	30
2.5 ACTUADOR.....	30
2.6 SENSOR DE DESPLAZAMIENTO LINEAL	31
2.7 PORTA ENCODER	31

2.8 BASE PARA EL SENSOR FOTOELÉCTRICO.....	32
2.9 TUERCA Y TORNILLO SINFIN.....	32
Parámetros de la rosca <i>acme</i>	34
Coeficiente de fricción.....	35
Carga	36
Cálculo del par	36
2.10 SOPORTE DE PROBETAS.....	37
2.11 MOTOR.....	38
2.12 SOPORTE DEL MOTOR.....	38
2.13 SOPORTE PARA LA ELECTRÓNICA.....	39
2.14 BASE DEL EQUIPO.....	40
CAPÍTULO 3. INTERFAZ DE USUARIO.....	41
3.1 PROGRAMACIÓN DE LA INTERFAZ DE USUARIO.....	42
3.2 REQUERIMIENTOS DE LA INTERFAZ DE USUARIO.....	42
3.3 DESCRIPCIÓN DEL PROGRAMA	44
CAPÍTULO 4. HARDWARE Y SOFTWARE DE CONTROL	49
4.1 TARJETA DE COMUNICACIÓN.....	49
4.2 COMUNICACIÓN	51
Antecedentes del puerto USB.....	53
Arquitectura del puerto USB	54
Características eléctricas	55
Tipos de transacción	56
Flujo de datos.....	57
Creación del <i>firmware</i>	58
4.3 CONTROL DE LA VELOCIDAD DEL ACTUADOR	67
Sistemas de control.....	67
Controlador de la velocidad del actuador	75
Sintonización de los parámetros del controlador.....	77
Modulación de ancho de pulso (PWM)	79

Programación del microcontrolador 2.....	80
CAPÍTULO 5. ELECTRÓNICA.....	82
5.1 MICROCONTROLADOR.....	82
5.2 CELDA DE CARGA.....	84
5.3 SENSOR DE DESPLAZAMIENTO LINEAL	89
5.4 ACONDICIONAMIENTO DE LA SEÑAL DEL SENSOR DE DESPLAZAMIENTO.....	91
5.5 ACONDICIONAMIENTO DE LA SEÑAL DE LA CELDA DE CARGA.....	91
5.6 ETAPA DE POTENCIA.....	93
5.7 FUENTE DE ALIMENTACIÓN	94
PRUEBAS Y RESULTADOS.....	95
CONCLUSIONES.....	102
TRABAJO A FUTURO	104
BIBLIOGRAFÍA.....	106
ANEXOS.....	108
ANEXO 1. PLANOS DE FABRICACIÓN DE LAS PIEZAS MECÁNICAS.....	108
ANEXO 2. HOJAS DE ESPECIFICACIONES	123
ANEXO 3. FIRMWARE PARA LOS MICROCONTROLADORES.....	129
ANEXO 4. DIAGRAMAS ELECTRÓNICOS.....	137
ANEXO 5. SOFTWARE.....	138

INTRODUCCIÓN

El Instituto de Biología de la UNAM, en su departamento de botánica lleva a cabo estudios acerca de la evolución morfológica en plantas de zonas secas; Para los estudios que ahí se realizan, se requiere hacer pruebas de flexión a los especímenes en estudio, ya que el conocimiento del *módulo de Young* de dichos elementos, les proporciona información relevante relacionada con la morfología de los organismos. Para realizar dichas pruebas, los investigadores del departamento de botánica de ese Instituto, utilizan actualmente diversos métodos y, cuando la muestra lo permite, emplean una máquina de pruebas mecánicas Instron® de la serie 3340.

Sin embargo, en muchas ocasiones, no es posible utilizar el equipo mencionado, ya que las muestras a estudiar son pequeñas y las cargas que soportan son menores a la mínima que se puede aplicar con una máquina de pruebas mecánicas. Cuando esto ocurre, se hace un estimado de la flexión que soporta el espécimen sometiéndolo al peso de pequeñas masas y midiendo su deformación con un calibrador; sin embargo, con este método los resultados obtenidos son poco confiables e incluso muchas veces no se pueden utilizar esos valores por ser incongruentes.

Por todo lo anterior, es necesario desarrollar equipos capaces de realizar estas pruebas en probetas o muestras de pequeñas dimensiones, de tal forma que se puedan obtener datos confiables, tal y como si se estuviera realizando la prueba con probetas estandarizadas y con un equipo comercial especializado.

El presente trabajo tiene como finalidad dar solución a ese problema que enfrenta el Instituto de Biología de la UNAM, en su departamento de botánica. Consta de 5 capítulos a lo largo de los cuales se explican, de manera breve, los aspectos más importantes del desarrollo del prototipo “*Máquina para ensayos de flexión en muestras de plantas de dimensiones pequeñas*” realizado en las instalaciones del Centro de Ciencias Aplicadas y Desarrollo Tecnológico (CCADET).

En el primer capítulo, *Marco Teórico*, se describen los conceptos que se consideraron indispensables para la comprensión de este trabajo y se mencionan las fórmulas que se emplean, así como su desarrollo, para la obtención del parámetro deseado por el cliente.

En el capítulo dos se mencionan los parámetros más relevantes de diseño y manufactura de las piezas que componen la parte mecánica del equipo, así como el funcionamiento de los elementos que se consideró pertinente.

Durante el capítulo tres se describe la lógica de la interfaz de usuario y sus requerimientos, y se explica el funcionamiento del programa.

En el cuarto capítulo, denominado *Hardware y software de control*, se explica de forma general la parte del hardware y software implementado para lograr el control de la “Máquina para ensayos de flexión en muestras de plantas de dimensiones pequeñas”, así como la interfaz de comunicación con la PC.

La electrónica necesaria para el funcionamiento del prototipo y su implementación es descrita en el capítulo cinco de este trabajo, así como la teoría de funcionamiento de los elementos electrónicos más relevantes.

Al final de este trabajo se describen las pruebas que se hicieron al equipo para poder calibrarlo y caracterizarlo, los resultados obtenidos al realizar pruebas con probetas de ciertos materiales; así como las conclusiones arrojadas, gracias a las pruebas realizadas, y finalmente, las posibles mejoras y aplicaciones futuras de este primer prototipo.

OBJETIVOS

- Diseñar y construir un equipo para realizar pruebas de flexión en especímenes orgánicos pequeños, con un rango de entre 1 y 3 [mm] de diámetro y con una longitud mínima de 30 [mm], el cual funcione de manera similar a la de un equipo profesional, que se encuentre actualmente en el mercado, para realizar este tipo de ensayos; es decir, que muestre la gráfica fuerza-deflexión, indique el *módulo de Young* y sea controlado por el usuario a través de una PC.
- Diseñar el equipo de manera que sea portátil para facilitar su traslado a diferentes laboratorios.
- Solucionar un problema del departamento de botánica del Instituto de Biología referente a una de sus líneas de investigación llamada *Evolución morfológica en plantas de zonas secas*; sin embargo, no se excluye la posibilidad de que el equipo pueda ser empleado en una línea de investigación distinta.
- Construir un prototipo de fácil manufactura, ligero, de bajo costo, de un mantenimiento preventivo y correctivo sencillo, con refacciones que se encuentren fácilmente en el mercado y que cuente con un modo de operación sencillo y amigable con el usuario.

JUSTIFICACIÓN

Actualmente existe un sinnúmero de equipos que permiten conocer el *módulo de Young* a través de un ensayo de flexión; sin embargo, las máquinas o equipos que permiten hacer este tipo de pruebas generalmente están diseñados para realizar el ensayo en probetas de dimensiones preestablecidas, dependiendo del material al cual se le haga la prueba y de la norma que se utilice para realizarla. Por lo tanto, cuando se desea analizar algún material o elemento que no entre en el rango del equipo, es difícil hacer una prueba que arroje resultados confiables puesto que la aplicación y medición de las variables no están diseñadas para elementos con dimensiones menores a las probetas estandarizadas.

Por otra parte, los equipos especializados en probetas de dimensiones reducidas tienen un alto costo por estar enfocadas al desarrollo científico; por lo que, cuando se desea analizar algún elemento del cual no se pueden realizar probetas, por ser únicos o especiales, tales como los huesos, madera y materia orgánica en general, se tienen que realizar estas pruebas utilizando algún método rudimentario (figura 1) y poco confiable, con lo cual generalmente la prueba no puede ser controlada y los resultados obtenidos pierden credibilidad.



Figura 1. Ensayo de flexión rudimentario. [University of Cambridge s.f.]

Existe una amplia gama de equipos que permiten realizar pruebas mecánicas a diversos tipos de materiales, y prácticamente todos los equipos comerciales cuentan con una PC, en la cual se pueden observar los resultados de la prueba y graficarlos; sin embargo, el costo de este tipo de equipo es sumamente alto y en ocasiones es necesario comprar un software adicional.

Por otro lado, la mayoría de las máquinas existentes para realizar ensayos mecánicos son denominadas *máquinas universales de ensayo de materiales* y suelen contar con funciones para realizar ensayos de tracción, compresión, flexión, doblado, pelado, cizallamiento y fricción; sin embargo, el usuario no siempre requiere de todas estas pruebas, por lo que en la mayoría de los casos, el equipo se encuentra subutilizado.

Por todo lo anterior, surge entonces la necesidad de diseñar y fabricar un sistema capaz de realizar pruebas de flexión a probetas de plantas con dimensiones reducidas; que pueda conectarse a una computadora para poder visualizar y graficar los datos que arroje la prueba; y que sea compacta para su fácil transporte.

El equipo debe ser diseñado para que su fabricación sea posible tomando en cuenta la infraestructura de los talleres del CCADET y contemplando piezas que sean comerciales y de fácil adquisición.

El sistema debe ser diseñado, considerando que su mantenimiento, tanto preventivo como correctivo, pueda ser realizado por el personal del CCADET o incluso por el mismo usuario proporcionándole las instrucciones pertinentes.

Además debe tomarse en consideración que se desea que el costo del equipo sea máximo nueve mil pesos, el cual es mucho menor en comparación con equipos comerciales.

METODOLOGÍA

Teniendo como objetivo el desarrollo de un equipo para realizar ensayos de flexión en probetas de materia orgánica de dimensiones pequeñas, se enlistan a continuación los pasos realizados para llevar a cabo, de manera organizada, el proyecto:

1. Planteamiento y análisis del problema a resolver.
2. Búsqueda de artículos e información relacionada con el tema.
3. Definición de requerimientos para construir el prototipo
4. Identificación de las herramientas con las que se cuenta para resolver el problema y aquellas que son necesarias, pero que no se tengan a la mano (manejo de software, dominio de un tema específico, maquinaria con la que se cuenta en los talleres del CCADET, entre otros.)
5. Diseño conceptual y de configuración¹, incluyendo la planeación de la parte mecánica y la elaboración de esquemas.
6. Análisis de los esquemas elaborados tomando en cuenta costos, diseño, material, facilidad de manufactura, ensamble y manejo de equipo.
7. Selección del diseño más viable para la solución del problema.
8. Análisis dinámico del mecanismo encargado de la transmisión de movimiento.
9. Determinación de las características para el motor que genera el movimiento en el mecanismo, tomando en cuenta el análisis dinámico realizado previamente.

¹ Los diseños que no cumplieron cabalmente los requerimientos para el desarrollo del sistema y el funcionamiento del mismo no son mostrados en este trabajo.

10. Diseño de detalle de la parte mecánica, incluyendo la elaboración de planos del diseño seleccionado, en un software CAD (Pro-Engineer®.)
11. Verificación de la fabricación de las piezas mecánicas fabricadas en el taller.
12. Ensamble de las piezas mecánicas.
13. Desarrollo de la interfaz hombre-máquina, utilizando LabVIEW® como software de programación.
14. Planeación y desarrollo de la etapa de control para el prototipo, incluyendo la programación del microcontrolador.
15. Diseño y fabricación de la tarjeta electrónica necesaria para manipular y comunicar los distintos dispositivos con el microcontrolador y éste, a su vez, con la PC.
16. Ensamble de los subsistemas que componen el equipo, parte mecánica, electrónica y de control.
17. Pruebas del equipo, ya ensamblado, para identificación y solución de fallas.
18. Puesta en marcha del prototipo.

REQUERIMIENTOS Y CARACTERÍSTICAS DE INGENIERÍA

Una vez analizado el problema que presenta el Instituto de Biología de la UNAM, en su Departamento de Botánica; y después de una entrevista con el futuro usuario del equipo se determinaron los requerimientos para este prototipo; los cuales se mencionan a lo largo de este trabajo según se fueron resolviendo o planteando para el correcto desarrollo del equipo y se enlistan a continuación.

Requerimientos del cliente:

- Dimensiones de la probeta o espécimen: Longitud mínima: 30 [mm];
Diámetro: 1-3 [mm]
- Capacidad máxima de la fuerza a aplicar: La necesaria para romper un mondadientes comercial de madera.
- Dimensiones del equipo: Tales que el equipo se transporte fácilmente entre laboratorios.
- Funcionamiento del equipo: Que calcule y muestre el módulo de Young.
- Modo de operación: Por medio de un equipo de cómputo portátil (Laptop) con una interfaz gráfica amigable.
- Costo: No mayor a \$9,000.

Características de ingeniería:

- Resolución en la medición de la fuerza con el equipo: 10 [gr]
- Resolución en la medición de la deflexión con el equipo: 0.1 [mm]
- Capacidad mínima del sensor de fuerza: 4 [kg]
- Protocolo de comunicación: USB
- Uso de piezas comerciales
- Fácil ensamble de las piezas y componentes del prototipo.
- Velocidad constante del actuador.

ELEMENTOS DEL SISTEMA

El desarrollo del prototipo “*Máquina para ensayos de flexión en muestras de plantas de dimensiones pequeñas*”, es un claro ejemplo de lo que se conoce como un sistema mecatrónico; ya que la mecatrónica es la integración sinérgica de la ingeniería mecánica con electrónica y control computarizado inteligente, en el diseño y la manufactura de productos y procesos; características con las que cuenta dicho prototipo.

Los subsistemas que integran el equipo son los siguientes:

- Mecánica
- Control
- Computación
- Electrónica

De modo que abarca todas las áreas de la ingeniería que conforman la mecatrónica, tal y como lo muestra la siguiente figura.

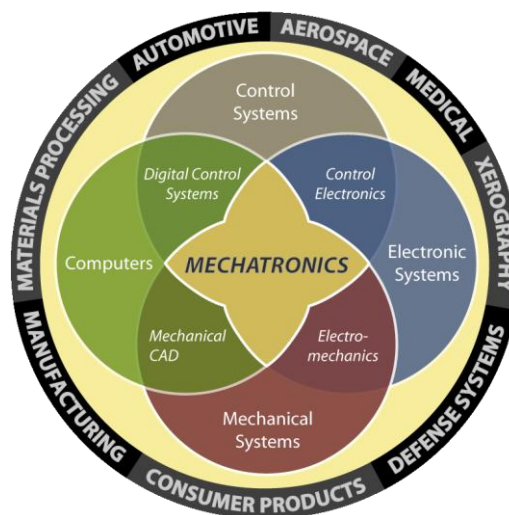


Figura 2. Mecatrónica. [California State University s.f.]

MECÁNICA

En esta área se diseñan las piezas necesarias para el funcionamiento mecánico del equipo, se hacen análisis dinámicos del sistema, se realizan los planos de fabricación de las piezas diseñadas, se generan los códigos de control numérico para la fabricación de las piezas que así lo requieran, se manufacturan y ensamblan.

CONTROL

Para que una prueba de flexión entregue resultados fidedignos, es necesario que la velocidad de la prueba sea constante por lo que se debe tener un control de velocidad para el motor, y así asegurar que el actuador (elemento que ejerce la fuerza) siempre se desplace a una velocidad constante, independientemente de las variaciones de carga. Esto pertenece al área de la ingeniería conocida como control.

COMPUTACIÓN

➤ Software

En esta área se encuentra el uso de un lenguaje de programación para el desarrollo de la interfaz del usuario, mediante la cual el operario del equipo puede ver graficada la prueba e ingresar los valores necesarios (medidas de la probeta) para calcular el *módulo de Young*.

➤ Firmware

En este apartado se considera el desarrollo del programa necesario para el microcontrolador, ya que si bien es un dispositivo electrónico, requiere de un programa

(el cual se puede realizar en lenguaje ensamblador o en un lenguaje de alto nivel) para su correcto funcionamiento.

ELECTRÓNICA

En el campo de la electrónica se incluye el diseño y aplicación de dispositivos electrónicos para la generación, transmisión, recepción, y almacenamiento de información.

El microcontrolador, como ya se había mencionado, es un dispositivo electrónico, por lo que se considera su uso dentro de esta área de la ingeniería, al igual que todos los dispositivos y elementos necesarios para la implementación de la etapa de control y el acondicionamiento de las señales, tales como amplificadores operacionales, resistencias y capacitores, por mencionar algunos.

Los dos microcontroladores utilizados, junto con los demás circuitos integrados y elementos electrónicos, permiten la interacción de la interfaz con el motor y los sensores del equipo.

CAPÍTULO 1. MARCO TEÓRICO

El conocimiento de las propiedades mecánicas de un material es de gran importancia para diversas aplicaciones, tanto en la industria como en la investigación y desarrollo de nuevas tecnologías; por lo que se han desarrollado diversas pruebas que permiten conocer tales propiedades.

Una de las propiedades más importantes en los materiales es su *rigidez*, la cual se define como su capacidad para soportar una carga sin deformación permanente o falla. Esta propiedad es inherente al material mismo y debe determinarse por experimentación. Entre las pruebas más importantes están las de tensión, de compresión y de flexión. Aunque con estas pruebas pueden determinarse muchas propiedades mecánicas importantes de un material, se utilizan principalmente para determinar la relación entre el esfuerzo promedio y la deformación unitaria en muchos materiales, ya sean metales, cerámicos, polímeros, orgánicos o compuestos.

Para comprender mejor la propiedad de rigidez de los materiales y sus implicaciones, es necesario establecer los conceptos de **esfuerzo**, **deformación** y **flexión**, para posteriormente explicar en qué consiste un **ensayo de flexión** y los resultados que éste nos arroja, ya que por la naturaleza de este trabajo, estos conceptos fueron empleados en el desarrollo del instrumento.

1.1 ESFUERZO

La fuerza por unidad de área, o la intensidad de las fuerzas distribuidas a través de una sección dada, se llama esfuerzo sobre esa sección y se representa con la letra griega σ (sigma). El esfuerzo en un elemento con área transversal **A** sometido a una carga axial

P (figura 3a) se obtiene, por lo tanto, al dividir la magnitud P de la carga entre el área A (figura 3b):

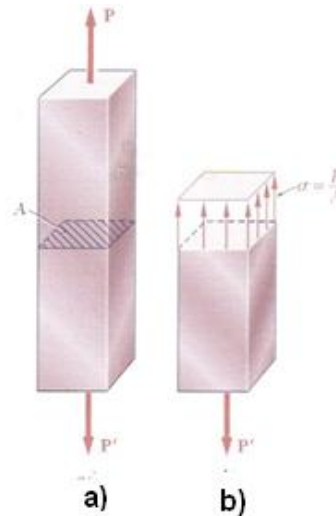


Figura 3. Esfuerzo normal. [Beer y Johnston 2004]

$$\sigma = \frac{P}{A}$$

Ecuación 1

Un tipo muy diferente de esfuerzo se obtiene cuando se aplican fuerzas transversales P y P' a un elemento AB (figura 4a). Al efectuar un corte en C entre los puntos de aplicación de las dos fuerzas (figura 4b), obtenemos el diagrama de la porción AC que se muestra en la figura 4c. Se concluye que deben existir fuerzas internas en el plano de la sección, y que su resultado es igual a P . Estas fuerzas internas elementales se conocen como fuerzas cortantes, y la magnitud P de su resultante es el cortante en la sección. Al dividir el cortante entre el área A de la sección transversal, se obtiene el

esfuerzo cortante promedio en la sección. Representando el esfuerzo cortante con la letra griega τ (tao), se escribe [Beer y Johnston 2004]:

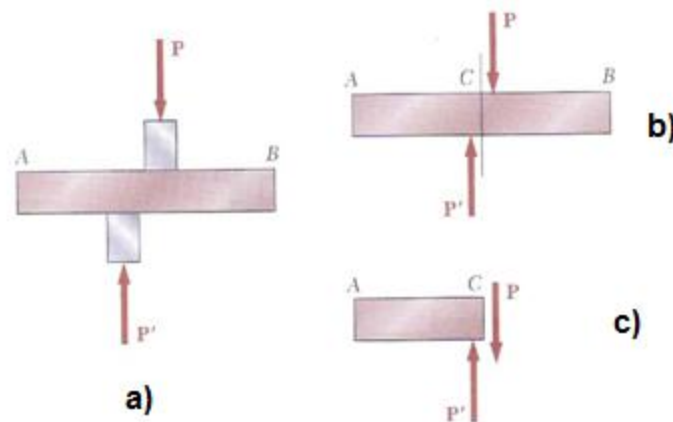


Figura 4. Esfuerzo cortante

$$\tau_{pr} = \frac{P}{A}$$

Ecuación 2.

1.2 DEFORMACIÓN

Cuando se aplica una fuerza a un cuerpo, ésta tiende a cambiar la forma y tamaño del cuerpo. A estos cambios se les llama **deformación** y ésta puede ser visible o prácticamente inadvertida si no se emplea el equipo apropiado para hacer mediciones precisas. Por ejemplo, una banda de hule experimenta una deformación muy grande cuando se estira. En cambio, en un edificio sólo ocurrirán deformaciones microscópicas en sus miembros estructurales debido a la carga de sus ocupantes [Hibbeler 2006].

1.3 FLEXIÓN

Si las fuerzas actúan sobre una pieza de material de tal manera que tiendan a inducir esfuerzos compresivos sobre una parte de una sección transversal de la pieza y esfuerzos de tensión sobre la parte restante, se dice que la pieza está en **flexión**. La ilustración común de la acción flexionante es una viga afectada por cargas transversales (figura 5a); la flexión también puede causarse por momentos o pares, como por ejemplo, los que pueden resultar de cargas excéntricas paralelas al eje longitudinal de una pieza.

La flexión pura se desarrolla bajo ciertas condiciones de carga, en el caso más usual la flexión va acompañada por el esfuerzo cortante. La resultante de los esfuerzos cortantes a través de una sección transversal es igual al corte transversal total V , el cual se calcula con la suma algebraica de todas las fuerzas transversales hacia la izquierda (o la derecha), de una sección.

En una sección transversal de la viga, la línea a lo largo de la cual los esfuerzos flexionantes son cero es llamada eje neutro. A la superficie que contiene los ejes neutros se le denomina superficie neutra (figura 5b). Sobre el lado de la viga en compresión las “fibras” de la viga se acortan, y sobre el lado en tensión se estiran, así la viga se flexiona en una dirección normal a la superficie neutra, tornándose cóncava del lado en compresión.

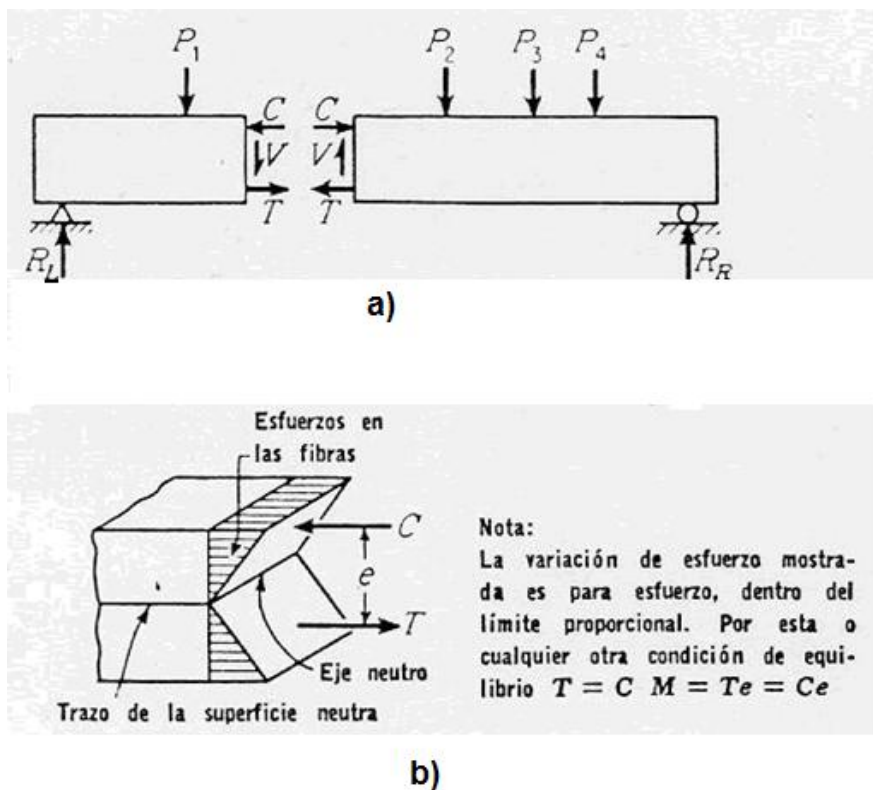


Figura 5. Flexión de una viga. [Universidad de Antioquia s.f.]

Se ha establecido, por muchas observaciones, que en la flexión pura las deformaciones son proporcionales a la distancia desde el eje neutro; y esto parece confirmarse cuando menos con una buena aproximación tanto en el rango de acción inelástico como en el de acción elástica. Ésta es llamada una condición de “**flexión plana**”.

Si los esfuerzos son proporcionales a las deformaciones, la variación del esfuerzo a través de una sección es lineal [Universidad de Antioquia s.f.].

1.4 ENSAYO DE FLEXIÓN

Este ensayo es un método utilizado en la medición del comportamiento mecánico de los materiales. Se realiza sujetando la muestra del material por sus extremos, posteriormente, la probeta apoyada como una viga simple, es sometida a la acción de una carga que se incrementa paulatinamente (figura 6), realizándose mediciones continuas y simultáneas de carga y deformación.

El esfuerzo máximo y la deflexión máxima son calculados para cada incremento de carga; los resultados son graficados en un diagrama de **fuerza-deflexión**. La pendiente de esta curva representa el módulo de elasticidad del material en estudio. Si la curva no presenta una región lineal, una línea secante se ajusta a la curva para determinar la pendiente [Instron s.f.].

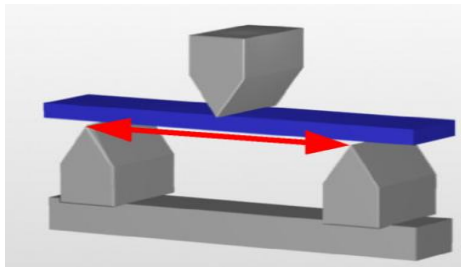


Figura 6. Prueba de flexión. [Zwick and Roell s.f.]

1.5 MÓDULO DE ELASTICIDAD

Se denomina **módulo de elasticidad** a la razón entre el incremento de esfuerzo y el cambio correspondiente a la deformación unitaria. Si el esfuerzo es una tensión o una compresión, el módulo se denomina **módulo de Young** y tiene el mismo valor para una tensión que para una compresión, siendo una constante independiente del esfuerzo, siempre que no exceda de un valor máximo denominado **límite elástico**. Tanto el

módulo de Young como el límite elástico, son naturalmente distintos entre las diversas sustancias.

El hecho de que la variación de deformación unitaria sea directamente proporcional a la variación de esfuerzo, siempre que no se sobrepase el límite elástico, se conoce como **ley de Hooke**.

Una vez que conocemos los términos y definiciones importantes en una prueba de flexión, a continuación se desarrollan las ecuaciones necesarias para poder obtener el *módulo de Young* al realizar un ensayo de este tipo [Meirovitch 1976]. En la figura 7 se muestra una gráfica esfuerzo contra deformación obtenida en un ensayo de tensión. Donde se puede observar que el *módulo de Young* (E) es obtenido en la zona lineal de la gráfica.

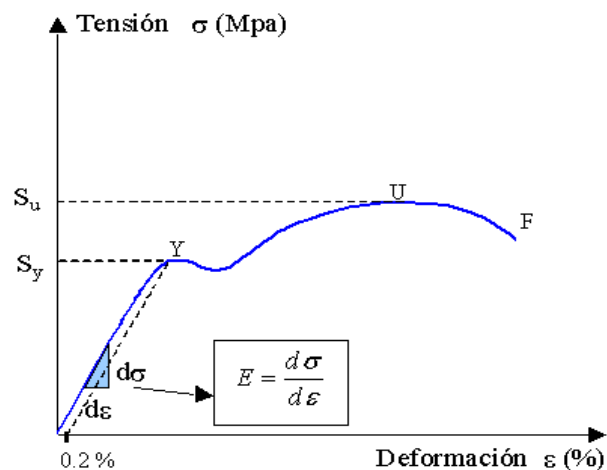


Figura 7. Grafica esfuerzo contra deformación para ensayo de tensión. [González s.f.]

1.6 DETERMINACIÓN DEL ESFUERZO FLECTOR

En el caso de una viga simplemente apoyada, sujeta a una carga puntual en medio del claro, tal como se muestra en la figura 8, el máximo momento flector, M_{max} , ocurre en medio del claro y se calcula así,

$$M_{max} = -\frac{FL}{4}$$

Ecuación 3

El máximo esfuerzo flector, σ_m , se calcula de la manera siguiente:

$$\sigma_m = M \frac{C}{I}$$

Ecuación 4

Donde C es la distancia del eje neutro a la fibra extrema, M es el momento flector e I es el momento de inercia [University of Cambridge s.f.].

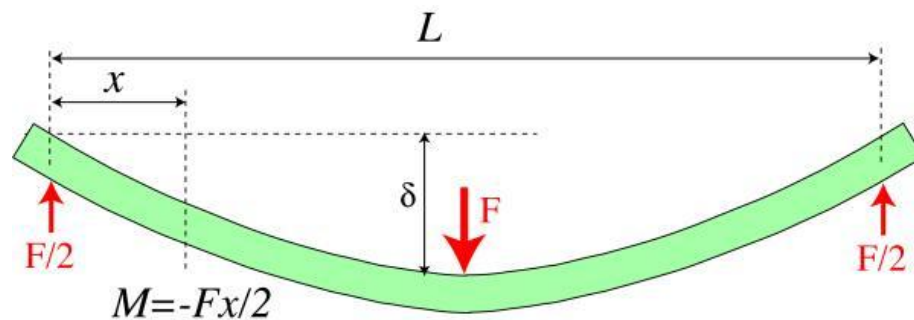


Figura 8. Diagrama de corte y momento de una viga simple [University of Cambridge s.f.].

1.7 DETERMINACIÓN DEL MÓDULO DE YOUNG EN FLEXIÓN

Aplicando el análisis de la ecuación de Euler-Bernoulli para vigas, tenemos:

$$\frac{d^2y}{dx^2} = -\frac{M}{EI}$$

Ecuación 5

si despejamos M

$$EI \frac{d^2y}{dx^2} = -M = \frac{Fx}{2}$$

y si integramos con respecto a x

$$EI \frac{dy}{dx} = \frac{Fx^2}{4} + C_1$$

Con $x = \frac{L}{2}$, $\frac{dy}{dx} = 0$; por lo tanto $C_1 = \frac{FL^2}{16}$

integrando una vez más con respecto a x

$$EIy = -\frac{Fx^3}{12} + \frac{FL^2x}{16} + C_2$$

con $x = 0$, $y = 0$, por lo tanto $C_2 = 0$.

entonces si despejamos y , la ecuación para la deflexión es:

$$y = \frac{Fx}{48EI} (3L^2 - 4x^2)$$

Ecuación 6

Puesto que y es el máximo en $x = \frac{L}{2}$, entonces

$$y_{\text{máx}} = \frac{FL^3}{48EI}$$

Por lo tanto, la deflexión en el centro de una viga está dada por la expresión:

$$\delta = \frac{FL^3}{48EI}$$

Ecuación 7

Y si consideramos F como el peso de la carga y sabiendo que el peso es igual a masa (m) por gravedad (g) y lo sustituimos en la ecuación 7, finalmente al despejar E obtenemos el modelo de elasticidad, en función de la carga aplicada, las dimensiones del claro y la geometría de la probeta: [University of Cambridge s.f.]

$$E = \frac{mg}{\delta} \frac{L^3}{48I}$$

Ecuación 8

CAPÍTULO 2. SISTEMA MECÁNICO

El sistema mecánico de la “*Máquina para ensayos de flexión en muestras de plantas de dimensiones pequeñas*” está compuesto por un conjunto de piezas (figura 9) que no sólo hacen posible el funcionamiento mecánico del equipo, sino que además permiten el correcto posicionamiento y funcionamiento de elementos de electrónica y sensado.

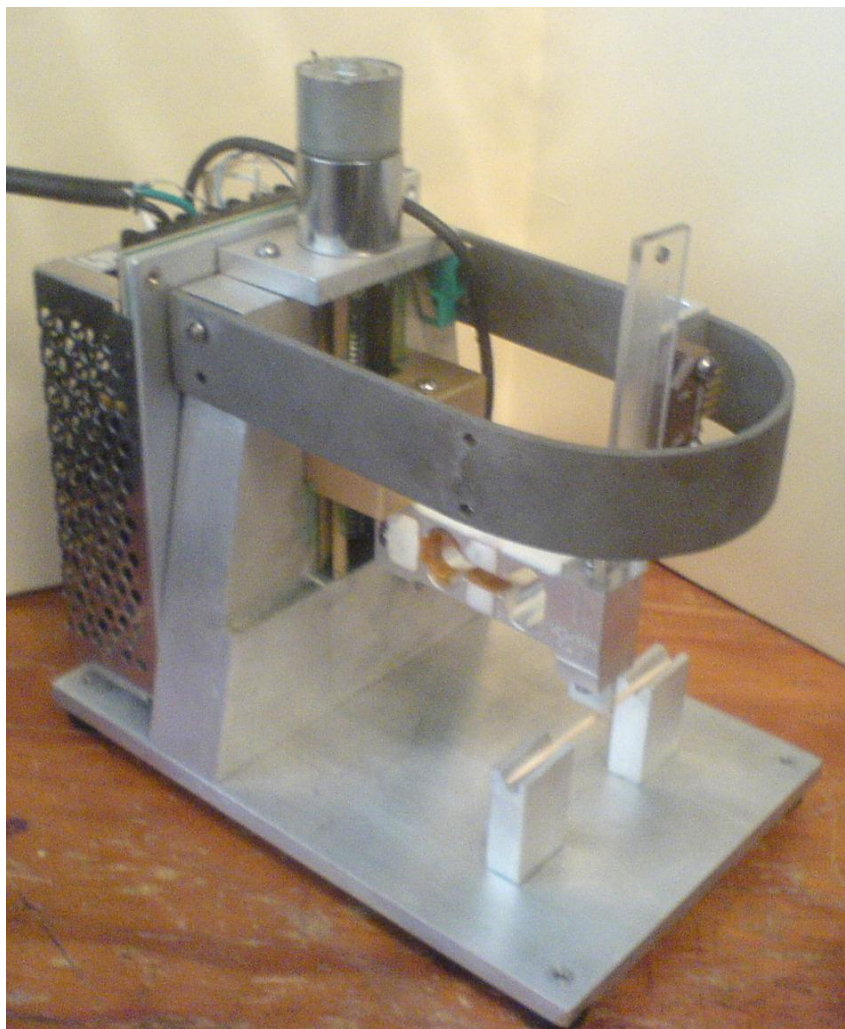


Figura 9. Prototipo ensamblado.

A continuación, se describen los parámetros más relevantes de diseño y manufactura de las piezas² que componen la parte mecánica del equipo, así como la función y operación de los elementos más relevantes.

Si bien la celda de carga y el sensor de desplazamiento lineal son piezas comerciales y su funcionamiento se basa en principios de electrónica, se mencionan en este capítulo ya que fue con base en ellos que se realizó el diseño mecánico del resto del equipo.

2.1 GUÍA

Esta es una pieza fundamental en el arreglo mecánico, ya que pertenece a la estructura del equipo y sirve como soporte para el motor; su función más importante y la que le da nombre es la de guiar la corredera.

La función de guía se logra gracias a un acople especial denominado cola de milano, el cual tiene diversas aplicaciones entre las que destacan la de ensambles entre piezas y la de guía, precisamente. Este tipo de acoplamiento permite un desplazamiento lineal entre las piezas y disminuye la desviación que puede existir en el paralelismo de las superficies de las dos partes que conforman el ensamble. La cola de milano para esta pieza se maquinó con un cortador con este perfil a un ángulo de 45°.

La guía está hecha en aluminio y tiene un inserto de latón (figura 10), se hizo de esta forma ya que esta pieza, por ser la más grande y por su función de guía, debía cumplir con dos requisitos: ser ligera para reducir el peso del equipo, y ser de un material duro para evitar el desgaste ante la fricción a la que estará sometida. Por lo tanto, se eligió hacer la mayor parte del cuerpo de aluminio y la parte correspondiente a la cola de milano hacerla de latón, ya que este material cuenta con excelentes propiedades

² Ver planos de fabricación en el anexo 1

mecánicas tales como buena resistencia a la corrosión, excelentes propiedades de soldadura, es de fácil maquinado por arranque de viruta y buena resistencia al desgaste; todas estas propiedades hacen apropiado al latón para la fabricación de la guía ya que estará en contacto con otra pieza en movimiento, lo cual generara fricción entre ambas piezas y, por tanto, calor y desgaste.

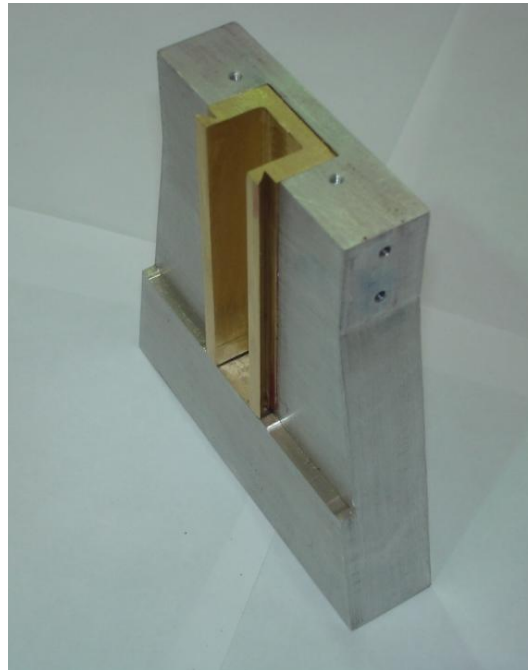


Figura 10. Guía ya maquinada y ensamblada

2.2 CORREDERA

La corredera está fabricada en latón, por los mismos motivos que se mencionaron en el apartado anterior. Esta pieza tiene la función de desplazar la celda de carga, el actuador y todo el conjunto que sirve para aplicar la fuerza y medir el desplazamiento. Esta pieza, al igual que la guía, fue fabricada con cola de milano a 45°.

2.3 CELDA DE CARGA

Para este equipo, se utiliza una celda de carga de la marca Vishay® modelo 1006, de tipo unipunto³ fabricada en aluminio, con una capacidad de carga de 5 [kg]. La elección de esta celda se basó en su capacidad, dimensiones y linealidad. Su finalidad es medir la fuerza aplicada a la probeta, lo cual la convierte en parte fundamental de la máquina.

Se eligió una celda con capacidad de carga de 5 Kg ya que era un requerimiento que el equipo fuera capaz de aplicar la fuerza necesaria para romper un mondadientes de madera. Para conocer su resistencia, se aplicó de forma manual la fuerza necesaria para romperlo y se cuantificó. Se realizaron varias pruebas y los datos obtenidos dieron un promedio de 2 [Kg]. Algunos resultados se muestran a continuación.

Número de la prueba	Fuerza soportada [Kg]
1	1.9
2	2.2
3	2.4
4	1.7
5	1.8
6	2.2
7	1.8
8	2.0
9	1.7
10	1.8

Tabla I. Resultados de la prueba a los mondadientes

Una vez que se obtuvo el valor de la fuerza promedio que soporta una probeta de madera con las características del mondadientes, se optó por elegir una celda que tuviera una capacidad mayor, de manera que se pudiera tener un exceso de la fuerza

³ Ver capítulo 5, apartado 5.2 Celda de carga

sin dañar el equipo, se decidió que el doble de la fuerza necesaria era un rango amplio para la seguridad del equipo, por lo tanto se eligió la celda con capacidad de 5 kg ya que es la capacidad comercial más cercana.

2.4 SOPORTE PARA LA CARCASA

Esta pieza está fabricada en acero, su función principal es servir de base para la cubierta que protege a los elementos más importantes del sistema (celda de carga, sensor de desplazamiento y corredera); ya que, de no protegerse, pueden ser dañados por el usuario o por el medio ambiente.

Por otra parte, el soporte para la cubierta también sirve como base para la carcasa que protege los elementos electrónicos y de control del equipo, de ahí que se fabrique en acero y de dimensiones robustas.

Finalmente, esta pieza cumple con una tercera función: sirve para sujetar el soporte del sensor fotoeléctrico en la posición adecuada, lo cual es fundamental para el correcto funcionamiento del sensor de desplazamiento lineal (sensor fotoeléctrico y *encoder*).

2.5 ACTUADOR

Es una pieza fabricada en aluminio, gracias a la cual se puede realizar la prueba en especímenes de dimensiones pequeñas, ya que su geometría permite transmitir directamente a la celda de carga la fuerza aplicada sobre la probeta; con lo cual se puede trabajar con muestras que cumplan con las dimensiones especificadas por el usuario, ya que no hubiera sido posible aplicar la fuerza directamente con la celda, debido a las dimensiones y geometría, tanto de la probeta como de la celda.

2.6 SENSOR DE DESPLAZAMIENTO LINEAL

El sensor de desplazamiento lineal está compuesto por dos elementos, un *encoder* lineal y un sensor fotoeléctrico, ambos de la marca USDigital®. El acoplamiento de ambos se muestra en la figura 11.

La función de este sensor es indicar la posición en la que se encuentra el actuador y proporcionar la señal de retroalimentación para el control de velocidad.

Por ser parte fundamental de la máquina, las características del sensor se explicarán a detalle en el capítulo 5.

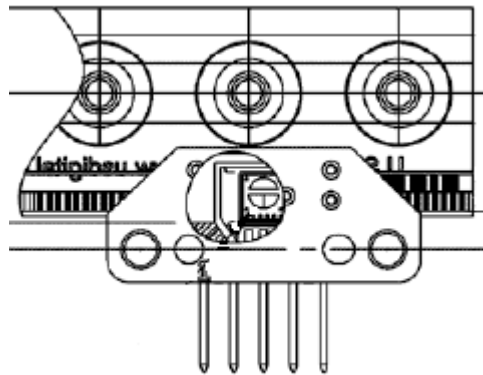


Figura 11. Sensor de desplazamiento lineal. [US-Digital 2008]

2.7 PORTA ENCODER

El *encoder* es una mica muy delgada y sensible, por lo que, para garantizar su perfecto funcionamiento, es necesario protegerla de las inclemencias a las que puede estar expuesta, para ello se diseñó el porta *encoder*.

El porta *encoder* está compuesto de dos piezas de acrílico que permiten que la mica quede alojada entre ellas, protegiéndola de posibles flexiones, dándole además estabilidad y la posibilidad de fijarla sin necesidad de utilizar algún pegamento u otro medio de fijación que pudiera dañarla. Esto es muy importante, ya que las franjas con las que cuenta este elemento son tan pequeñas que cualquier pegamento o agente externo podría modificarlas afectando su lectura.

Por otra parte, la distancia que debe haber entre el sensor fotoeléctrico y el *encoder* es muy reducida (.025") por lo que si no se coloca de manera adecuada se pueden tener errores en la medición o incluso no lograr medición alguna.

2.8 BASE PARA EL SENSOR FOTOELÉCTRICO

Debido a la geometría del sensor fotoeléctrico, se diseñó una pieza para poder sujetar el mismo a la estructura del equipo. Debe señalarse que, en la configuración que se eligió, es el *encoder* el que se desplaza y el sensor fotoeléctrico es el que permanece fijo a la estructura.

Esta base está fabricada en aluminio y garantiza, junto con el soporte del *encoder*, que la separación entre éste y el sensor esté dentro de la tolerancia indicada por el fabricante en las hojas de especificaciones⁴ de estos componentes.

2.9 TUERCA Y TORNILLO SINFIN

Para lograr el movimiento vertical de la corredera y, por consecuencia, del actuador, se emplea un tornillo sinfín con rosca *Acme* acoplado al eje del motor; el tornillo tiene ese

⁴ Ver anexo 2

tipo de cuerda ya que los tornillos de transmisión de potencia son diseñados para ejercer una fuerza con ventaja mecánica.

Debido a que los tornillos de transmisión de potencia están sometidos a combinaciones de esfuerzos, las roscas de estos tornillos son algo diferentes a las que se emplean para los tornillos de sujeción.

La figura siguiente muestra los perfiles de rosca de tres tornillos de transmisión de potencia.

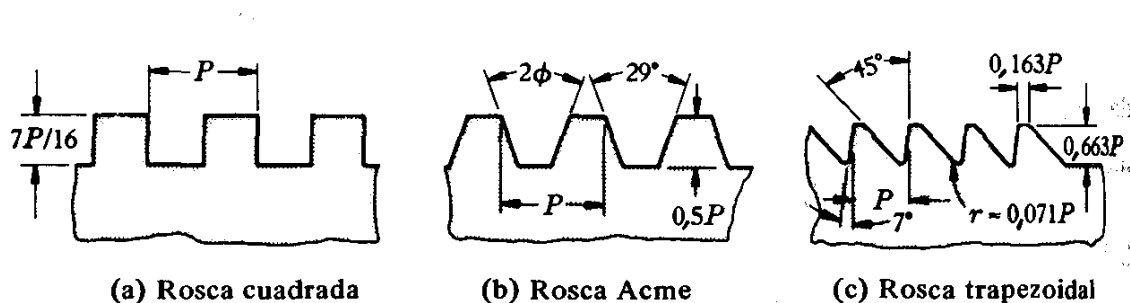


Figura 12. Perfiles de rosca para tornillos de transmisión de potencia. [Faires 2002]

La rosca *Acme*, debido a sus flancos inclinados, no es tan eficiente teóricamente como la rosca cuadrada, pero la práctica ha demostrado que la calidad de la fabricación y el estado de las superficies de la rosca son las que determinan la eficiencia [Faires 2002]

La manufactura de la rosca *Acme* es, además, más sencilla, es por ello que se seleccionó este tipo de rosca para el tornillo sinfín de nuestro equipo.

Parámetros de la rosca acme

Dado que los tornillos de transmisión de potencia suelen estar sometidos a combinaciones de esfuerzos, es importante conocer la información de cálculos de ingeniería referente a ellos.

Para utilizar la ecuación que nos permite calcular el par necesario para desplazar una carga mediante un tornillo sinfín, se necesitan conocer algunos parámetros geométricos de éste. La ecuación que se utiliza para calcular dicho par es la siguiente:

$$T = \frac{W D m}{2} \left[\frac{\cos \phi \tan \lambda + f}{\cos \phi - f \tan \lambda} \right]$$

Ecuación 9

El ángulo 2ϕ (figura 13a) es denominado ángulo de rosca y ϕ es llamado ángulo de presión y es igual a 14.5° para la rosca normalizada Acme. En la figura siguiente se muestran otros parámetros importantes de la cuerda Acme.

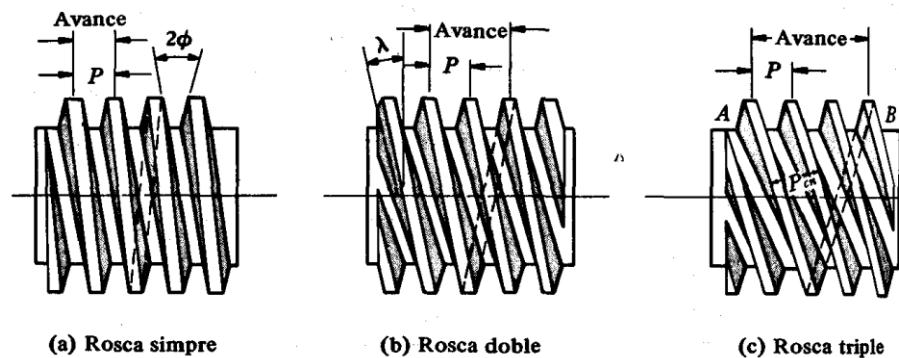


Figura 13. Paso y avance de diferentes tipos de rosca Acme. [(Faires 2002)]

El ángulo de avance es λ , y el *paso axial* o únicamente *paso* P es la distancia, medida axialmente, desde un punto de un hilo o filete al punto correspondiente de un hilo adyacente y se calcula de la siguiente manera:

$$P = \frac{1}{n}$$

Ecuación 10

Donde n es el número de hilos por pulgada en nuestro tornillo sinfín.

La profundidad de engrane o contacto es $0.5P$ para las roscas Acme. Esta profundidad determina el diámetro medio del tornillo.

El **avance** L es la distancia que adelanta la rosca en una vuelta o revolución; dicho de otra manera, es la distancia que se desplaza la tuerca a lo largo del eje en una vuelta.

Un tornillo de rosca simple, tiene un avance igual al paso (figura 13a). Un tornillo de rosca doble tiene dos **arranques** o **entradas** y el avance es igual al doble del paso. Un tornillo de rosca triple tiene tres **arranques** o **entradas**, y el avance es igual al triple del paso, como en la figura 13c, etcétera. En el caso de nuestro tornillo, éste es de rosca simple por lo que el avance es igual al paso.

El **ángulo de avance** λ es el que forma una tangente a la hélice del paso con un plano normal al eje del tornillo (figura 13b). Si D_m es el diámetro medio o diámetro de paso de la rosca, entonces:

$$\lambda = \tan^{-1} \left(\frac{\text{Avance}}{\pi * D_m} \right)$$

Ecuación 11

Coeficiente de fricción

En la ecuación 9, f representa el coeficiente de fricción. Si las superficies de los hilos de la rosca están lisas y bien lubricadas, el coeficiente de fricción puede ser tan bajo como

0.10, pero con materiales y mano obra de calidad promedio, los autores Ham y Ryan⁵ recomiendan $f = 0.125$. En nuestro cálculo se utiliza un coeficiente de fricción igual a 0.3 ya que no sólo se considera la fricción entre tuerca y tornillo, sino además, la fricción que se presenta entre las paredes de la guía y la corredera.

Carga

En cuanto a la carga W , se ha realizado un estimado de la carga que se debe emplear en la ecuación 9, ya que, por un lado, se considera la masa que se debe desplazar verticalmente (peso de la corredera y celda de carga), pero por otro lado se considera la resistencia que presenta la muestra al momento de hacer contacto con el aplicador.

Por lo anterior, la carga W que se utiliza para este cálculo es igual a 5000 gramos.

Cálculo del par

Una vez que tenemos los elementos necesarios, procedemos a realizar el cálculo del par necesario para desplazar la carga para, posteriormente, seleccionar el motor que satisfaga este requerimiento.

El tornillo sinfín fue maquinado en acero en los talleres del CCADET y tiene las siguientes características:

- Diámetro: $D = \frac{5}{8}[\text{pulg}] = 9.525[\text{mm}]$
- Número de hilos por pulgada: $n = 12$

Con estos datos calculamos el paso, el diámetro medio y el ángulo de avance.

⁵ Ham y Ryan. An Experimental Investigation of the friction of screw Threads, Bull. 247, Univ. de Illinois

Paso (ecuación 10):

$$P = \frac{1}{n} = \frac{1}{12} = 0.083 \text{ [pulg]} = 2.12 \text{ [mm]}$$

Diámetro medio:

$$Dm = D - 0.5P = 9.525 - (0.5 \times 2.12) = 8.467 \text{ [mm]}$$

Ángulo de avance (ecuación 11):

$$\lambda = \tan^{-1} \left(\frac{P}{\pi * Dm} \right) = \tan^{-1} \left(\frac{2.12}{\pi * 8.467} \right) = 4.55^\circ$$

Sustituyendo en la ecuación 9 los valores previamente obtenidos, tenemos:

$$T = \frac{WDm}{2} \left[\frac{\cos \varphi \tan \lambda + f}{\cos \varphi - f \tan \lambda} \right] = \frac{5000 \times 0.8467}{2} \left[\frac{\cos 14.5^\circ \tan 4.55^\circ + 0.3}{\cos 14.5^\circ - 0.3 \tan 4.55^\circ} \right]$$
$$T = 845.17 \text{ [gr * cm]}$$

Conociendo todos estos parámetros, la tuerca fue fabricada en latón con las mismas características que el tornillo (tipo de cuerda y paso), para poder ser soldada a la corredera y así poder transmitir el movimiento generado por el motor.

2.10 SOPORTE DE PROBETAS

Los soportes para la probeta están fabricados en aluminio, tienen la geometría necesaria para que una probeta de las dimensiones especificadas por el usuario pueda colocarse sin ningún problema en el espacio que queda entre los dos bloques. Los soportes cuentan con un corte en “v”, el cual fue hecho con la finalidad de que la probeta quede bien apoyada y no se corra el riesgo de perderla durante la prueba.

2.11 MOTOR

El motor que proporciona el movimiento para aplicar la fuerza a la probeta, es un motor de corriente directa con una reducción, también llamado motorreductor. Se eligió este tipo de motor ya que en la aplicación es necesario tener un control preciso de la posición, lo cual es más difícil de obtener con un motor a pasos y, si bien un servomotor pudiera cumplir con esta característica, su costo, en comparación con un motorreductor, lo hacía poco factible.

Características del motorreductor:

- Torque: 2200 [g•cm]
- Alimentación: 12 [V]
- Consumo de corriente: 400 [mA] (en carga máxima)
- Revoluciones por minuto: 50 [RPM].
- Peso: 150 [g]

El motor que se eligió para esta aplicación es un motor que satisface el par calculado al realizar el análisis dinámico del sistema, y no necesita de un arreglo de engranes adicional para aumentar su torque. Además, este motor es de dimensiones pequeñas en comparación con otros motores que entregan el mismo torque, lo anterior es importante para nuestra aplicación pues se desea que el equipo sea lo más compacto y ligero posible.

2.12 SOPORTE DEL MOTOR

El soporte del motor está fabricado en aluminio y se diseñó con la finalidad de poder unir el motor a la guía, ya que por la geometría del motor era imposible unir estos dos componentes de manera directa. Por otra parte, gracias a este soporte se puede

colocar el motor en la posición adecuada para que a su vez sea acoplado al sinfín y pueda transmitir la potencia y movimiento necesario.

El acoplamiento del motor con su soporte se muestra en la siguiente figura, donde también se pueden observar los barrenos pasados con los que cuenta el soporte para poder unirse a la guía del prototipo.



Figura 14. Motor y su base

2.13 SOPORTE PARA LA ELECTRÓNICA

La finalidad de estas cuatro piezas es soportar la electrónica del equipo, por lo que en realidad no cargan demasiado peso, su diseño se realizó en aluminio.

Tienen la forma de una **L** o un ángulo a 90°, para que se pudieran sujetar a la base para la carcasa y que a su vez cumplieran su función de brindar soporte a la electrónica

2.14 BASE DEL EQUIPO

Esta última pieza es fundamental, ya que gracias a ella se puede tener el equipo completo sobre una superficie rígida que permita realizar las pruebas sin problemas de desplazamiento del equipo o vibraciones; está fabricada en una placa de aluminio y cuenta con los barrenos necesarios para poder fijar la guía, la fuente y los soportes de la probeta.

CAPÍTULO 3. INTERFAZ DE USUARIO

En este capítulo se describe la lógica de funcionamiento de la interfaz de usuario.

La “Máquina para ensayos de flexión en muestras de plantas de dimensiones pequeñas” cuenta con una interfaz, la cual permite la comunicación entre el operador y la máquina y, de esta forma, se ejecutan las acciones que el usuario requiere, además de que proporciona información de la prueba realizada.

Las acciones que el usuario puede realizar son las siguientes:

- Posicionamiento manual del actuador.
- Inicio y/o paro de la prueba.
- Ingreso de datos del espécimen al cual se le aplica el ensayo.
- Imprimir gráfica *fuerza-deflexión*.

Los requerimientos mínimos a cubrir por el sistema fueron establecidos por los miembros del Instituto de Biología, quienes serán los futuros usuarios de la máquina.

Estos requerimientos mínimos son:

- Botones para posicionamiento manual del aplicador.
- Mostrar gráfica de *Fuerza vs Deflexión* de la prueba realizada
- Calcular y mostrar el *módulo de Young* del espécimen.
- Botón de inicio de la prueba.
- Permitir el ingreso de los datos del espécimen.

Otra función del programa es brindarle información al usuario de los eventos que suceden en el sistema en cada momento, dichos eventos se enlistan en las siguientes líneas:

- Indicar al usuario que el equipo se encuentra conectado o desconectado de la PC.
- Indicar que una prueba se está realizando.
- Indicar que el actuador está en movimiento.
- Indicar que el sistema está listo para efectuar una prueba.

3.1 PROGRAMACIÓN DE LA INTERFAZ DE USUARIO

LabVIEW[®] es un lenguaje de programación gráfica para instrumentación, adquisición y análisis de datos, automatización, control y comunicación.

LabVIEW[®] permite el desarrollo de programas en un ambiente gráfico de manera rápida por lo que el tiempo de diseño se ve reducido de manera significativa en comparación con la realización del mismo programa en otro lenguaje de programación.

Por todo lo anterior, la interfaz fue desarrollada en LabVIEW[®] y se aprovecharon las siguientes características:

- Facilidad para el análisis de datos.
- Facilidad para generar y mostrar gráficas.
- Facilidad en el llamado a bibliotecas (.DLL).
- Capacidad para mostrar mensajes de error.

3.2 REQUERIMIENTOS DE LA INTERFAZ DE USUARIO

El programa debe ser amigable con el usuario para que pueda hacer uso de la máquina sin necesidad de tener que pasar mucho tiempo leyendo un manual que le indique cómo usarla ni de que deba tener mucha experiencia en el manejo de este tipo de

equipos, por lo que sólo es necesario que la persona que lo utilice esté familiarizada con el ambiente Windows[®]. El programa permite al usuario introducir las dimensiones del espécimen al cual se le realizará la prueba y, una vez que éstos han sido introducidos y se encuentran dentro de los valores permitidos, manda la orden al microcontrolador 1⁶ para que la prueba comience.

Una vez que la prueba ha iniciado, el microcontrolador 1 comienza a adquirir las señales de la celda de carga y del sensor de desplazamiento, y los envía a la PC a través del puerto USB para que dichas señales sean manipuladas en el programa para obtener la gráfica **Fuerza vs. Deflexión**, necesaria para la obtención del *módulo de Young*. El microcontrolador 1 es el medio por el cual la interfaz de usuario se comunica con el resto del equipo.

En esta sección del capítulo se desarrolla y explica el programa⁷, teniendo en cuenta que será utilizado en una computadora que cuente con un sistema operativo Windows[®] 98 o superior y que cuente con un puerto USB libre para ser utilizado en la comunicación con el sistema.

A continuación se representa la lógica de funcionamiento del programa mediante una máquina de estados (figura15).

Dato recibido = *R*

Equipo conectado = *C*

Bandera para reset de gráfica = *G*

Imprimir = *I*

Botón arriba = *X*

Botón abajo = *Y*

Botón inicio = *Z*

Salir = *S*

⁶ Para el desarrollo de este equipo se utilizaron 2 microcontroladores, uno de ellos para adquirir datos(microcontrolador 1) y otro para el control de velocidad(microcontrolador 2)

⁷ En el anexo 5, se muestra los estados y algunas de las condiciones internas de cada uno.

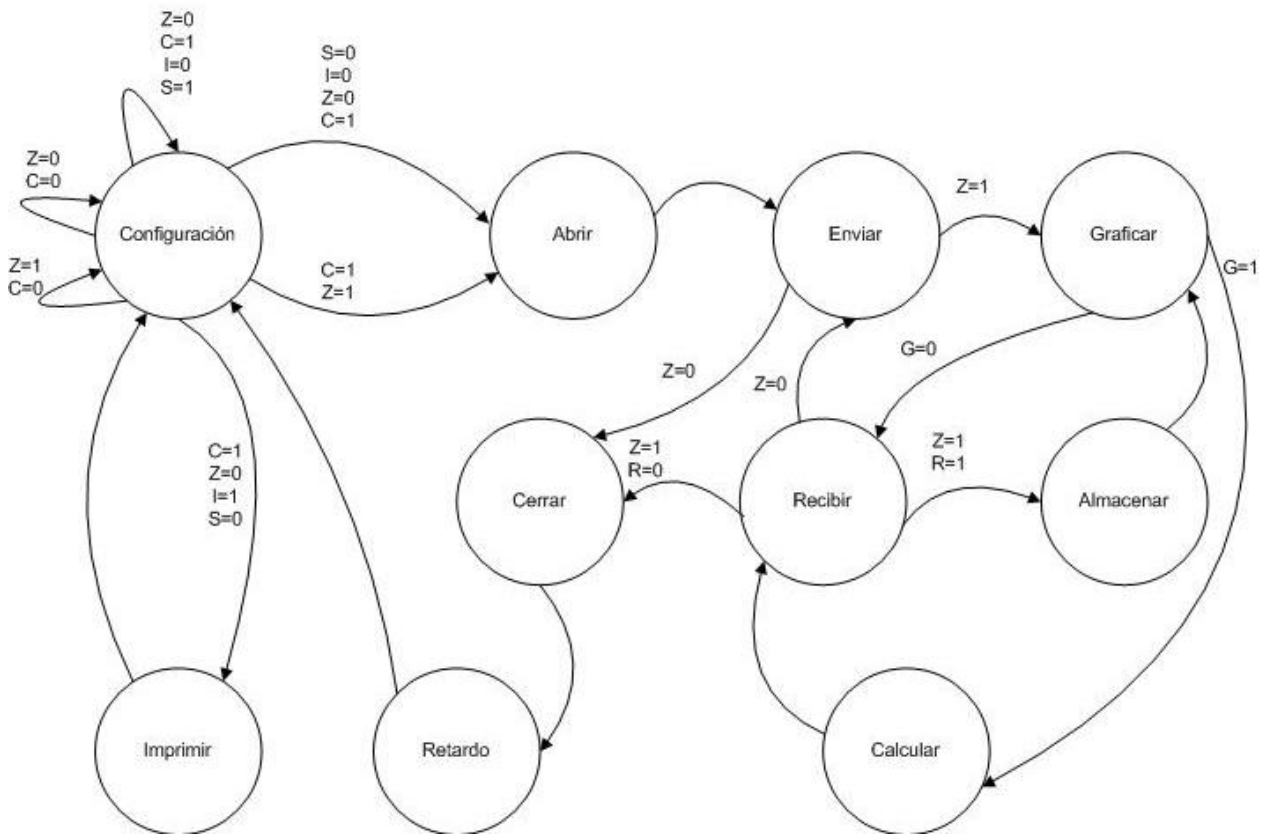


Figura 15. Máquina de estados para el software del equipo.

3.3 DESCRIPCIÓN DEL PROGRAMA

Al inicio del programa se verifica si el dispositivo está conectado, lo cual se hace en el estado de configuración, si lo está se habilitan los botones de movimiento manual del **actuador**, de inicio de prueba y el control donde se indica el tipo de sección transversal de la probeta. Si el equipo se encuentra conectado el indicador de estado del equipo se pone en color verde, además de que se muestra una leyenda que indica que el equipo se encuentra conectado y el indicador de estado de la prueba muestra que se está

esperando alguna acción a ejecutar. En la figura 16 se observa la interfaz cuando el equipo se encuentra conectado.

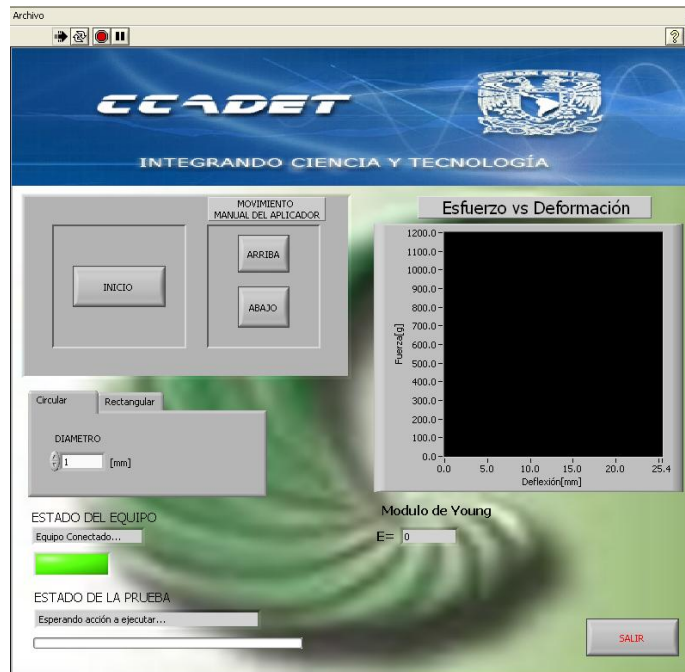


Figura 16. Equipo conectado.

Al estar el equipo conectado también se habilita el menú que está ubicado en la esquina superior izquierda del programa, por medio del cual el usuario podrá salir del programa o imprimir la gráfica que se haya generado al haber realizado una prueba.

En la figura 17 se muestra el despliegue del menú mencionado anteriormente al cual también se puede acceder por medio de los comandos de acceso rápido *Ctrl+P* para imprimir la gráfica y *Ctrl+S* para salir del programa.



Figura 17 Menú del programa.

Cuando el prototipo está desconectado todos los controles están inhabilitados, el indicador de estado del equipo está en color rojo y una leyenda muestra que la máquina está desconectada además de que el indicador de estado de la prueba informa al usuario que conecte el prototipo para poder realizar una prueba. En la figura 18 se muestra la interfaz cuando el equipo está desconectado.

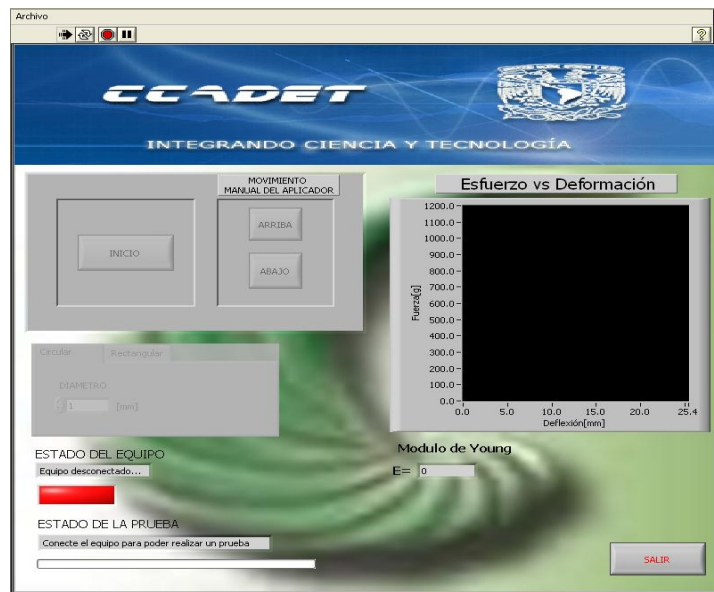


Figura 18. Equipo desconectado.

Cuando el equipo está conectado el usuario puede utilizar los botones de movimiento manual del **actuador** para colocarlo en su posición inicial, en caso de que quiera subirlo o bajarlo a una velocidad mayor a la que se realiza la prueba, ya que la velocidad normal es muy lenta.

Antes de iniciar una prueba, el usuario debe asegurarse de que ha escogido la sección transversal correcta de su probeta e ingresar sus dimensiones ya que por omisión están en 1 [mm], en caso de no especificar las dimensiones y si su probeta tiene dimensiones diferentes a las preestablecidas, los resultados obtenidos en la prueba no tendrán validez.

Las dimensiones no podrán ser mayores a 5 [mm] de diámetro en el caso de una sección transversal circular y 5 [mm] de ancho o de alto en el caso de una probeta de sección transversal rectangular. En el caso de una probeta de sección transversal rectangular en el programa se indica la orientación en que debe colocarse la probeta.

En la figura 19 se muestra el control para selección de la sección transversal, tanto circular como rectangular.

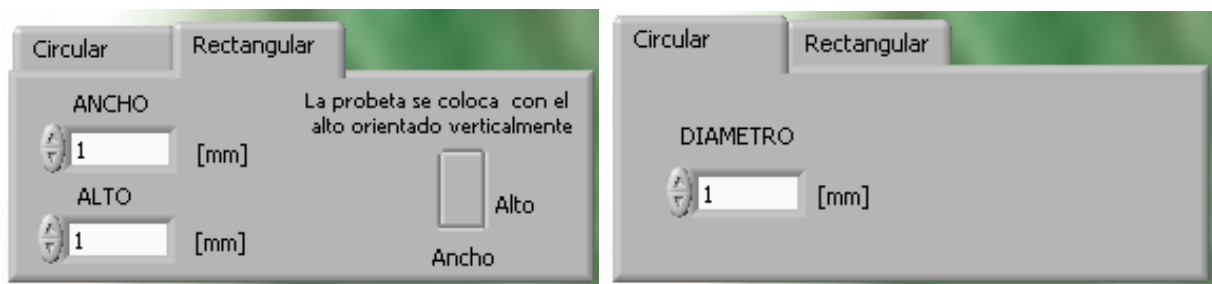


Figura 19 a) sección circular seleccionada, b) sección rectangular seleccionada.

La prueba da comienzo al oprimir el botón de “*inicio*”, que en ese momento cambia su nombre y su función a “*detener*”, para poder parar la prueba en el momento que se desee al volver a oprimirlo. Una vez que se ha iniciado una prueba se inhabilitan los botones de movimiento manual del **actuador** para evitar que el usuario pueda tener acceso a ellos, así como el control para seleccionar la sección transversal de la probeta y sus dimensiones. En la figura 20 se muestra el cambio en el nombre del botón inicio, los botones de movimiento manual y el control para selección de la sección transversal inhabilitados.

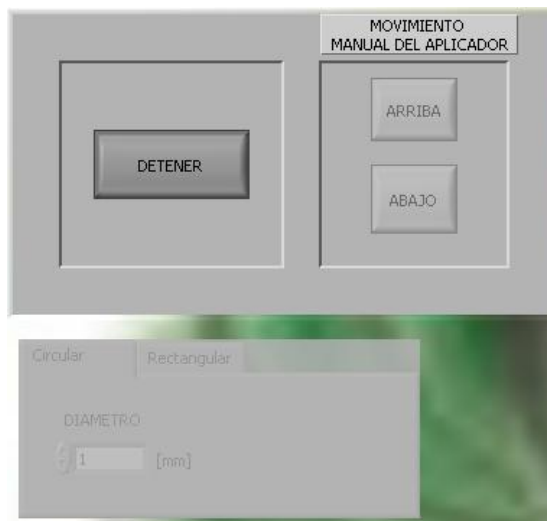


Figura 20. Estado de los botones y el control al estar una prueba en progreso.

Al estar realizándose una prueba, la barra de progreso, que se encuentra en la esquina inferior izquierda del programa, comenzará a llenarse indicando que la prueba está efectuándose de manera correcta, y el indicador de estado mostrará que la prueba está en progreso. En la siguiente figura se muestra el llenado de la barra de progreso al haber iniciado una prueba.

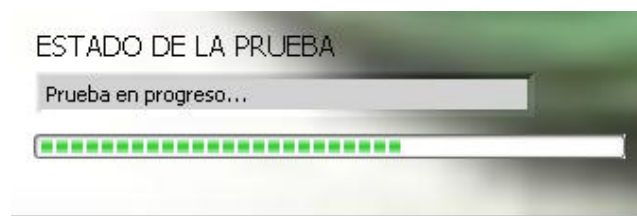


Figura 21. Llenado de la barra de progreso.

Una vez que el usuario ya no quiere realizar más pruebas basta con oprimir el botón “Salir” que se encuentra ubicado en la esquina inferior derecha para cerrar el programa o también podrá salir por medio del menú ubicado en la esquina superior izquierda.

CAPÍTULO 4. HARDWARE Y SOFTWARE DE CONTROL

En este capítulo se explica de forma general la parte del hardware y software implementado para lograr el control de la “*Máquina para ensayos de flexión en muestras de plantas de dimensiones pequeñas*”, así como la comunicación con la PC.

4.1 TARJETA DE COMUNICACIÓN

El control de la máquina se realiza mediante un microcontrolador PIC18F2550 (microcontrolador 1) de la marca *MICROCHIP*[®], de este dispositivo se utilizaron los siguientes periféricos:

- Módulo de comunicación USB 2.0
- 6 Pines de I/O
- 1 canal del convertidor analógico-digital de 10 bits

La elección de este microcontrolador se debió a que fue un requerimiento el uso de una Laptop y, actualmente, éstas sólo cuentan con puertos de comunicación Ethernet y USB. El protocolo de comunicación USB 2.0, puede enviar y recibir datos a una velocidad de hasta 480 [MBit/s], con lo cual el tiempo en el que se realiza la comunicación a la PC se ve reducido de manera significativa en comparación con el puerto serial RS-232, cuya velocidad máxima es de 115200 [bit/s].

Además de lo anterior, este microcontrolador cuenta con un convertidor analógico a digital de 10 bits, el cual es necesario para tener una buena resolución en el sensado de la fuerza aplicada al espécimen.

Un punto importante para la elección de este microcontrolador también es el hecho de que MICROCHIP® desarrollo un archivo DLL (Dynamic Linking Library) en el que proporciona las funciones de acceso al puerto USB con un microcontrolador de la familia PIC18FXX5X y que existen varios ejemplos en internet que facilitan el entendimiento de cómo establecer la comunicación USB, asunto que no es tan fácil de conseguir con otros tipos de microcontroladores. Los archivos DLL son bibliotecas en las cuales se almacena un código ejecutable, el cual puede ser llamado desde una o varias aplicaciones, con lo cual es posible mejorar el rendimiento o solucionar errores simplemente haciendo una actualización de la biblioteca.

Otro punto importante es el hecho de que este microcontrolador cuenta con *PLL* interno.

Un *PLL* es un sistema de control de frecuencia en lazo cerrado, cuyo funcionamiento está basado en la detección de la diferencia entre la fase de la señal de entrada y la fase de la señal de salida. Son utilizados en demodulación de señales de frecuencia modulada (FM), decodificadores de tono, multiplicación y división de frecuencias, por mencionar algunos.

El microcontrolador utiliza el *PLL* como multiplicador de frecuencia, por lo que le permite incrementar la frecuencia que entra por el oscilador externo de 20 [MHz] hasta 96 [MHz]. El USB para poder trabajar correctamente necesita una frecuencia de 48 [MHz], por lo que la frecuencia de 96 [MHz] conseguida con el PLL es dividida entre 2 y de esta forma se obtiene la frecuencia a la cual trabaja el protocolo USB que es de 48 [MHz]. De los 96 [MHz] generados por el PLL también se puede obtener la frecuencia a la cual trabajará el microcontrolador, la cual puede ser de 48 [MHz], 32 [MHz], 24 [MHz] y 16 [MHz] para el caso de un cristal externo de 20 [MHz] dependiendo del divisor que se seleccione para el PLL. En este caso se eligió dividir la frecuencia del PLL entre 2 debido a que de esta forma se obtiene la frecuencia máxima a la cual puede trabajar el

microcontrolador y por lo tanto, el microcontrolador trabaja a una frecuencia de 48 [MHz] con la cual se ejecutan 12 millones de instrucciones por segundo (MIPS).

Por los motivos explicados anteriormente, se optó por la utilización de este microcontrolador ya que de esta forma el tiempo en el que se ejecuta el programa y el tiempo en el que se envían los datos se ve disminuido en comparación con la comunicación serial; además de la facilidad de contar con una computadora con un puerto USB donde se conecte el equipo.

Las funciones que realizan los microcontroladores en la máquina pueden ser divididas en 4 grupos, los cuales son:

- Comunicación con la PC.
- Lectura del convertidor analógico – digital.
- Control de velocidad del motor con el que se desplaza el **actuador**.
- Lectura de la señal enviada por la GAL de la decodificación del *encoder* de cuadratura.

4.2 COMUNICACIÓN

En esta sección del capítulo se describe, de manera general, el funcionamiento del protocolo USB 2.0, que es el utilizado para la comunicación entre la PC (HOST) y el PIC; También se describe cuáles son los datos que son enviados entre la PC y el PIC y la acción que se ejecutará dependiendo del valor del dato que se ha enviado.

Como ya se había mencionado en el capítulo 3, el software está hecho en LabVIEW[®] debido a la facilidad con que se puede programar; el firmware está realizado en lenguaje C en el compilador CCS[®] versión 4.013.

El protocolo de comunicación entre el microcontrolador y la PC se basa en el valor del dato recibido. La tabla 2 muestra cuales son las acciones que se ejecutan dependiendo del valor del dato que se haya recibido en el microcontrolador y también indica si el microcontrolador le envía algún dato a la PC y, en dicho caso, cuál es ese dato.

Dato enviado por la PC	Acción tomada por el microcontrolador 2	Dato enviado a la PC por el microcontrolador 1
Arriba=1 Abajo=0 Inicio=0	Mueve el actuador hacia arriba	Ninguno
Arriba=0 Abajo=1 Inicio=0	Mueve el actuador hacia abajo	Ninguno
Arriba=0 Abajo=0 Inicio=1	Inicia la prueba	Lecturas de la celda de carga y del encoder
Arriba=0 Abajo=0 Inicio=0	Mantiene el actuador sin movimiento	Ninguno

Tabla II. Protocolo de comunicación

El microcontrolador 1 le indica a la PC el estado de la prueba de acuerdo al valor del elemento 31 del arreglo de datos. La tabla 3 muestra cual es el valor enviado en el elemento 31 del arreglo y la acción que ejecutará la PC dependiendo de dicho valor.

Valor del Dato enviado en el elemento 31 del arreglo	Acción tomada por la PC
0	Se detiene la prueba y se indica que se ha excedido la fuerza máxima de 4 kg
1	No hace nada y la prueba continúa
2	Se detiene la prueba y se indica que el aplicador a llegado a su máximo recorrido
3	Se detiene la prueba y se indica que la probeta ha llegado a la ruptura

Tabla III. Protocolo de comunicación.

Antecedentes del puerto USB

El puerto USB es una conexión serial para crear interfaces entre dispositivos y fue desarrollado por un grupo de 7 empresas (Compaq[®], Digital Equipment Corp[®], IBM PC Co[®]., Intel, Microsoft[®], NEC[®] y Northern Telecom[®]) para crear una computadora personal libre de puertos considerados obsoletos (puertos serial RS-232 y paralelo IEEE 1284). El principal objetivo de este bus fue el de conectar dispositivos de distintos tipos en un mismo puerto, además de mejorar las capacidades de *plug-and-play*, para facilitar la detección de nuevos dispositivos y permitir conectar y desconectar dispositivos sin requerir reiniciar la computadora, e incluso crear clases de hardware que no requieran controladores para que el sistema operativo pueda comunicarse con éstos; otro aspecto importante es el de alimentar al dispositivo por el mismo bus, al que se le pueden requerir hasta 500[mA].

Arquitectura del puerto USB

La interconexión física USB usa una topología de estrellas apiladas que permite el funcionamiento simultáneo de 127 dispositivos a la vez. Un **hub** está en el centro de cada estrella, el cual es un elemento *plug and play* que sirve como concentrador al cual se le pueden agregar más dispositivos USB, incluyendo otro **hub**. Sólo hay un host conectado en cualquier dispositivo USB que maneja todos los dispositivos conectados como se indica en la figura siguiente:

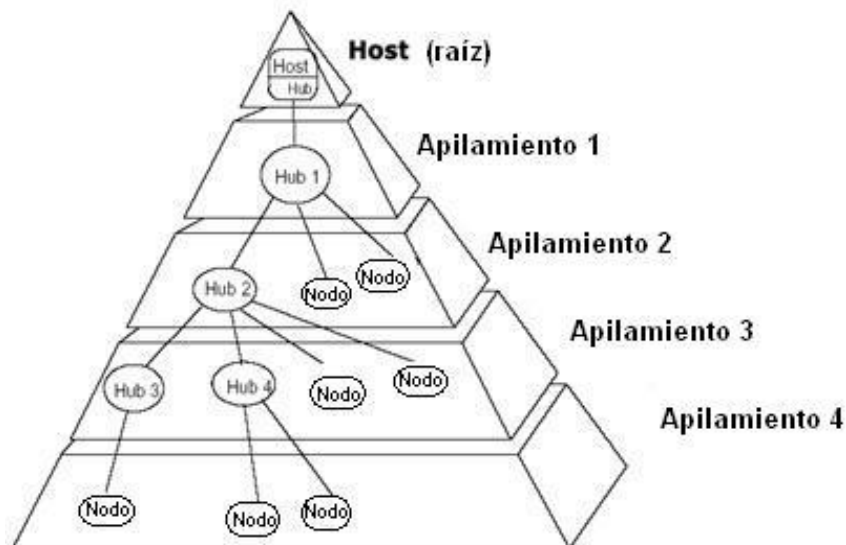


Figura 22. Topología USB. [usb.org s.f.]

La velocidad máxima de transferencia depende de la velocidad del *hub* que se esté utilizando. Hay tres tasas de transferencia que son:

- 480 Mbit/s en High-Speed
- 12 Mbit/s en Full-speed
- 1.5 Mbit/s en Low-speed

La velocidad de transferencia recomendada varía dependiendo del tipo de aplicación. En la siguiente tabla se pueden observar cuáles son las aplicaciones típicas para las diferentes velocidades.

VELOCIDADES	APLICACIONES
LOW - SPEED 10-100 kb/s	Teclado, mouse Periféricos de juegos Periféricos de realidad virtual
FULL - SPEED 500 kb/s – 10 Mb/s	Audio Micrófono Teléfono ordinario
HIGH - SPEED 25 – 400 Mb/s	Video Almacenamiento Imágenes

Tabla IV. Principales aplicaciones para las diferentes velocidades

Características eléctricas

El puerto USB utiliza 2 cables para la transferencia de señales y 2 para la alimentación del dispositivo a conectar, dicha alimentación es controlada por el *Host*, el cual garantiza un voltaje mínimo de 4.75 volts y un máximo de 5.25 volts y se pueden obtener hasta 500 [mA] por puerto. La transferencia de datos se realiza por medio de un par de cables trenzados con impedancia de $90 \pm 15 [\Omega]$ y se denominan como D+ y D-.

Dichos cables típicamente son cortos y utilizan señalización diferencial para combatir los efectos del ruido en enlaces largos. La distancia máxima para los cables es de 5 metros.

En la figura 23 se puede observar la estructura del cable USB, donde se pueden ver los cables para transmisión de datos D+ y D- y los cables para alimentación por USB que son V_{Bus} y GND. La alimentación por USB es un beneficio muy importante ya que, de esta forma, un dispositivo puede funcionar sin la necesidad de una fuente externa siempre y cuando no exceda la corriente máxima permitida por el puerto.

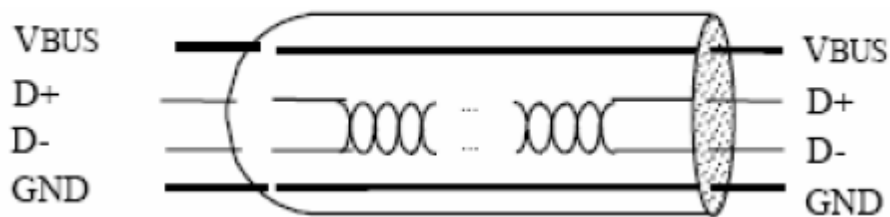


Figura 23. Estructura del cable USB. [usb.org s.f.]

Tipos de transacción

El protocolo USB soporta cuatro tipos básicos de transferencias de datos que son:

- **Control Transfers:** Es usada para configurar un dispositivo cuando se conecta por primera vez notificándole el número que le ha asignado el *host*.
- **Bulk Transfers:** Es usada cuando es necesario enviar un gran volumen de datos de una sola vez como sucede en el caso de escáneres o cámaras. En este tipo de transferencia la calidad en la entrega de datos está garantizada, por lo que no hay pérdida de datos.

- **Interrupt Transfers:** En este tipo de transferencia los datos son entregados en un tiempo no menor al especificado por el dispositivo, por ejemplo en aplicaciones de adquisición de datos o en un teclado o en un mouse. Las características de esta transferencia son: pocos datos, no periódicos y baja frecuencia.
- **Isochronous Data Transfers:** Es usada cuando se necesita enviar datos en tiempo real aunque puede producirse pérdida de datos debido a que, en caso de producirse un problema eléctrico, éste no es corregido y el dato es entregado. Un ejemplo típico de este tipo de datos es la voz ya que necesita ser enviada en tiempo real para su correcto análisis.

Flujo de datos

Un **Endpoint** es una porción direccionable de un dispositivo USB, donde se almacena la información que va a ser enviada o recibida en un flujo de comunicación entre el *host* y el dispositivo.

Un **pipe** es una asociación entre un *endpoint* en un dispositivo y el *software* en el *host*. Representa la habilidad de mover datos entre el *software* en el *host* vía un *buffer* de memoria y un *endpoint* en un dispositivo.

Un dispositivo común puede requerir varios *endpoints* para poder crear un esquema eficiente de transferencia de datos. A la colección de *endpoints* se le denomina **Interfaz** y está directamente relacionada con la conexión física. A una colección de interfaces se le denomina **Configuración**. Una configuración define los atributos y características de un dispositivo específico.

La figura 24 muestra cómo es el flujo de comunicación entre el *host* y el dispositivo y se observan los *pipes* por donde fluye la información entre el *host* y el dispositivo.

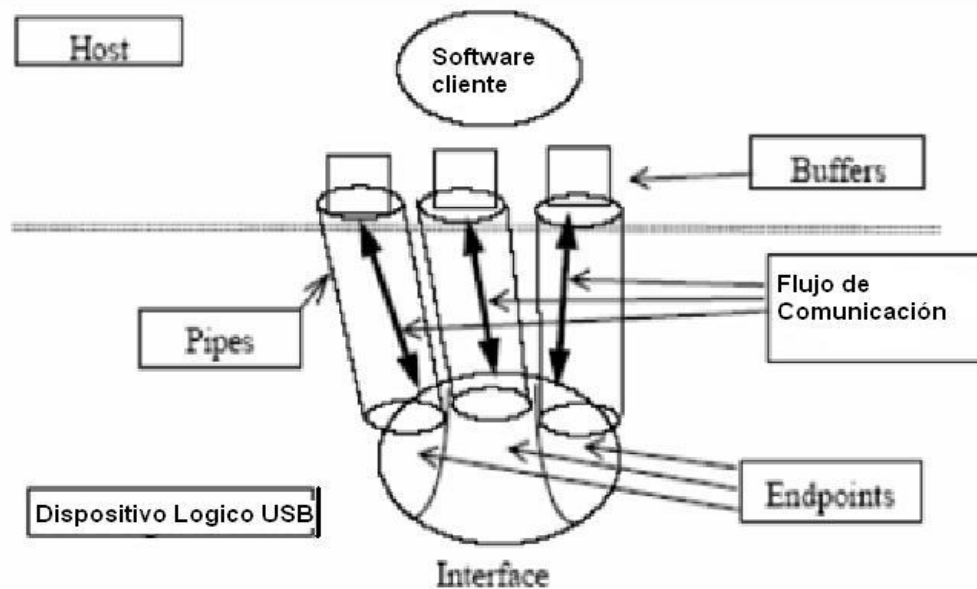


Figura 24. Flujo de comunicación USB. [usb.org s.f.]

Creación del *firmware*

En este apartado se explica cómo se creó el **firmware** en el compilador CCS[®], el cual es el programa que está contenido en el microcontrolador, para poder establecer la comunicación USB con el *host*. Se explica cuáles son las instrucciones que hay que agregar para que la comunicación se realice satisfactoriamente, así como cuáles son las líneas más importantes que hay que configurar en los descriptores, los cuales son el medio por el cual un dispositivo USB reporta sus capacidades al *host*, para que la comunicación se realice satisfactoriamente.

Los descriptores son estructuras de datos con un formato bien definido. Cada descriptor inicia con campos de distintos tamaños que indican el número total de bytes contenidos en el descriptor y el tipo de descriptor. Los descriptores a configurar son 5: descriptor de dispositivo, descriptor de configuración, descriptor de interfaz, descriptor de *endpoint* y descriptor de *cadena*.

➤ **Descriptor de dispositivo**

El descriptor de dispositivo representa las características del dispositivo entero, es decir, información que implica el funcionamiento del dispositivo. Especifica información como la versión USB que soporta, el tamaño máximo de paquete, el VID y PID (*Vendor ID* y *Product ID* por sus siglas en inglés respectivamente) y el número posible de configuraciones que un dispositivo puede tener. Un dispositivo USB puede tener sólo un descriptor de dispositivo.

La tabla 5 muestra la composición del descriptor de dispositivo.

Offset	Campo	Bytes	Valor	Descripción
0	<i>bLength</i>	1	Número	Tamaño del descriptor en bytes (18 bytes)
1	<i>bDescriptorType</i>	1	Constante	Descriptor de dispositivo (0x01)
2	<i>bcdUSB</i>	2	BCD	Versión codificada en BCD, por ejemplo 0x0200 para la versión 2.0
4	<i>bDeviceClass</i>	1	Clase	Código de clase (asignado por USB Org). Si es igual a 0 cada interface especifica su propio código de clase. Si es igual a 0xFF el código de clase es especificado por el

				fabricante
5	<i>bDeviceSubClass</i>	1	SubClase	Código de subclase (Asignado por USB Org.) para cualquier valor distinto de 0xFF, que representa el específico del fabricante
6	<i>bDeviceProtocol</i>	1	Protocolo	Código de protocolo (Asignado por USB Org). Si está en 0xFF implica un protocolo definido por el fabricante
7	<i>bMaxPacketSize</i>	1	Número	Máximo tamaño de paquete para el endpoint cero. Los tamaños válidos son 8, 16, 32, 64
8	<i>IdVendor</i>	2	ID	Identificador del fabricante (Asignado por USB Org.)
10	<i>IdProduct</i>	2	ID	Identificador de producto (Asignado por USB Org.)
12	<i>bcdDevice</i>	2	BCD	Versión del dispositivo en BCD
14	<i>iManufacturer</i>	1	Índice	Índice del descriptor de cadena para el fabricante
15	<i>IProduct</i>	1	Índice	Índice del descriptor de producto
16	<i>iSerialNumber</i>	1	Índice	Índice en el descriptor de cadena para el número de serie
17	<i>bNumConfigurations</i>	1	Número	Número de configuraciones posibles

Tabla V Estructura del descriptor de dispositivo.

➤ **Descriptor de configuración**

Un dispositivo USB puede tener muchas configuraciones diferentes, aunque la mayoría de los dispositivos son simples y sólo tienen una.

El descriptor de configuración contiene información de cómo es alimentado el dispositivo, cuál es la corriente que consume, el número de configuraciones que tiene, etcétera. Una vez que el dispositivo ha sido configurado ya no es posible hacer modificaciones a la configuración. Si una interfaz tiene configuraciones alternativas, puede hacerse un cambio de configuración una vez que el dispositivo ha sido configurado, esto significa que el dispositivo puede cambiar entre una configuración que lo defina como un mouse o una configuración que lo defina como un teclado por ejemplo, pero sin poder hacer cambios a los datos contenidos en dichas configuraciones. La tabla 6 muestra la composición del descriptor de configuración.

Offset	Campo	Bytes	Valor	Descripción
0	<i>bLength</i>	1	Número	Tamaño de descriptor en Bytes
1	<i>bDescriptorType</i>	1	Constante	Descriptor de configuración (0x02)
2	<i>wTotalLength</i>	2	Número	Tamaño total devuelto para esta configuración, incluyendo el tamaño combinado de todos los descriptores de esta configuración (configuración, interfaz, endpoint y específicos de clase o configuración)
4	<i>bNumInterfaces</i>	1	Número	Número de interfaces
5	<i>bConfigurationValue</i>	1	Número	Valor a usar como argumento cuando el host llame a <code>SetConfiguration()</code> en la enumeración para seleccionar esta configuración
6	<i>iConfiguration</i>	1	Índice	Índice en el descriptor de

				cadena para la descripción de esta configuración
7	<i>bmAttributes</i>	1	Mapa de bits	Características de configuración D7 Reservado, siempre 1. (USB 1.0 Bus Powered) D6 Alimentado por USB D5 encendido remoto D[4..0] Reservado, siempre 0. Si D6 está en 1 entonces deberá proporcionarse la cantidad de corriente necesaria en <i>bMaxPower</i>
8	<i>bMaxPower</i>	1	[mA]	Máximo consumo de corriente en el bus USB, cuando el dispositivo está funcionando normalmente, expresado en unidades de 2 [mA]. Por ejemplo 500[mA]=250

Tabla VI Estructura del descriptor de configuración.

➤ Descriptor de interfaz

Este descriptor describe una interfaz específica dentro de una configuración. Una configuración provee una o más interfaces, cada una con uno o más *endpoints* describiendo un conjunto único de *endpoints* dentro de una configuración. La tabla 7 muestra la composición del descriptor de interfaz.

Offset	Campo	Bytes	Valor	Descripción
0	<i>bLength</i>	1	Número	Tamaño del descriptor en bytes (9 Bytes)
1	<i>bDescriptorType</i>	1	Constante	Descriptor de interfaz (0x04)
2	<i>bInterfaceNumber</i>	1	Número	Número de interfaz
3	<i>bAlternateSetting</i>	1	Número	Número utilizado para seleccionar una interfaz alternativa usando su

				índice
4	<i>bNumEndpoints</i>	1	Número	Número de <i>endpoints</i> contenidos en la interfaz
5	<i>bInterfaceClass</i>	1	Clase	Código de clase (Asignado por USB Org.) ó 0XFF para usar una clase definida por el constructor ó 0x00 para uso futuro
6	<i>bInterfaceSubClass</i>	1	SubClase	Código de Subclase (Asignado por USB Org.). Asignado de igual manera que <i>bInterfaceClass</i>
7	<i>bInterfaceProtocol</i>	1	Protocolo	Código de protocolo (Asignado por USB Org.) según <i>bInterfaceClass</i> y <i>bInterfaceSubClass</i> . Si es un protocolo definido por el constructor se deberá asignar 0XFF
8	<i>iInterface</i>	1	Índice	Índice en el descriptor de cadena para encontrar la descripción de esa interfaz

Tabla VII. Composición del descriptor de interfaz.

➤ Descriptor de endpoint

Este descriptor es usado para proveer al *host* la información que utiliza para determinar el ancho de banda que requiere cada *endpoint*. El *endpoint* cero siempre es un *endpoint* de control y es configurado antes de que cualquier descriptor sea pedido por el *host* por lo que nunca hay un descriptor para el *endpoint* cero.

En la tabla 8 se muestra la composición del descriptor de *endpoint*.

Offset	Campo	Bytes	Valor	Descripción
0	<i>bLength</i>	1	Número	Tamaño del descriptor en Bytes (7 bytes)
1	<i>bDescriptorType</i>	1	Constante	Descriptor de (0x05)
2	<i>bEndpointAddress</i>	1	<i>Endpoint</i>	Dirección del <i>endpoint</i> Bits [0..3] Número de endpoint. Bits [4..6] Reservados. Siempre a cero Bits 7 Direction 0 = Out, 1 = In (Ignorado por <i>Endpoints</i> de control)
3	<i>bmAttributes</i>	1	Mapa de bits	Bits 0-1 Tipo de transferencia 00 = <i>Control</i> 01 = <i>Isochronous</i> 10 = <i>Bulk</i> 11 = <i>Interrupt</i> Bits 2-7 son reservados. Si es un <i>endpoint</i> isócrono, Bits 3 y 2 = Tipo de sincronización (Iso Mode) 00 = Sin sincronización 01 = Asíncrono 10 = Adaptable 11 = Síncrono Bits 5-4 = Tipo de uso (Iso Mode) 00 = <i>Endpoint</i> de datos 01 = <i>Endpoint</i> de retroalimentación 10 = <i>Endpoint</i> de datos explícito para retroalimentación 11 = Reservado Si la transferencia no es isócrona los bits del 7 al 2 deberán ir establecidos a cero.
4	<i>wMaxPacketSize</i>	2	Número	El tamaño de paquete máximo que este <i>endpoint</i> es capaz de enviar o recibir.
5	<i>bInterval</i>	1	Número	Intervalo para obtener las transferencias de datos, usado sólo para transferencias de tipo interrupción. Es un valor que debe estar entre 1 y 255.

Tabla VIII Estructura del descriptor de endpoint.

➤ **Descriptor de cadena**

Los descriptores de cadena contienen información entendible para el humano y son opcionales. Si los descriptores de cadena no son usados, cualquier referencia a este descriptor debe ser cero indicando que no hay cadena disponible. Las cadenas son codificadas en formato **Unicode** y los productos pueden ser hechos para soportar múltiples lenguajes.

La estructura del descriptor de cadena cero se muestra en la siguiente tabla:

Offset	Campo	Bytes	Valor	Descripción
0	<i>bLength</i>	1	Número	Tamaño del descriptor en Bytes
1	<i>bDescriptorType</i>	1	Constante	Descriptor de cadena (0x03)
2	<i>wLANGID[0]</i>	2	Número	Código de lenguaje cero soportado (ej. 0x0409 Inglés – Estados Unidos)
4	<i>wLANGID[1]</i>	2	Número	Código de lenguaje 1 soportado (ej. 0x0c09 Inglés - Australia)
N	<i>wLANGID[x]</i>	2	Número	Código de lenguaje x soportado (ej. 0x0407 Alemán - Standard)

Tabla IX Estructura del descriptor de cadena cero.

Deberán existir tantos descriptores consecutivos como lenguajes se hayan definido en el descriptor de cadena cero. La tabla 10 muestra la estructura de los descriptores subsecuentes al descriptor cero.

Offset	Campo	Bytes	Valor	Descripción
0	<i>bLength</i>	1	Número	Tamaño del descriptor en Bytes
1	<i>bDescriptorType</i>	1	Constante	Descriptor de cadena (0x03)
2	<i>bString</i>	n	Unicode	Cadena codificada en Unicode

Tabla X Estructura del descriptor de cadena para cada lenguaje.

Los PICs de la serie 18FXX5X tienen 3 modos de funcionamiento con USB que son:

1. **USB Human Interface Device (HID).**- Son Dispositivos de baja velocidad que no requieren driver, ejemplos de esta clase de dispositivos son los ratones, joysticks, teclados, por mencionar algunos
2. **USB Communication Device Class (CDC).**- Son dispositivos de velocidad media que necesitan de un driver para poder funcionar y tienen la característica de que crean un puerto serie virtual.
3. **USB Custom Driver.**- Son dispositivos de alta velocidad que requieren instalar un driver para poder funcionar. El modo de funcionamiento que se utiliza para la comunicación entre el microcontrolador y el *host* es el *USB Custom Driver* ya que de esta forma se puede emplear el modo de transferencia *Bulk* y, como ya se explicó anteriormente, de esta forma el tiempo de transferencia se ve reducido en comparación con la comunicación serial RS-232.

4.3 CONTROL DE LA VELOCIDAD DEL ACTUADOR

En las pruebas de flexión que se realizan con esta máquina, la velocidad a la cual se desplaza el actuador es importante, ya que se debe cumplir con la condición de que dicha velocidad se mantenga constante a lo largo del ensayo.

El objetivo es, por lo tanto, mantener constante la velocidad del actuador. Para ello es necesario conocer algunos conceptos básicos de teoría de control y, posteriormente, aplicarlos a nuestro sistema en particular. En este apartado del capítulo, se describen brevemente algunos de esos conceptos.

Sistemas de control

Un sistema es un conjunto de componentes que relacionados entre sí actúan y realizan un objetivo determinado. Un *sistema de control* se refiere al conjunto de aquellos elementos que cumplen el objetivo de mantener un determinado proceso bajo condiciones deseadas. A continuación se describen aquellos conceptos que son útiles al momento de controlar cualquier proceso; además de los componentes de un sistema de control.

➤ Control en lazo cerrado

Un sistema que mantiene una relación determinada entre la salida y la entrada de referencia, comparándolas y usando la diferencia como medio de control se denomina *sistema de control en lazo cerrado*, también llamado *sistema de control realimentado* (figura 25). En un sistema de control en lazo cerrado, se alimenta al controlador con la señal de *error*, que es la diferencia entre la señal de entrada (*set point*) y la señal de realimentación (determinada mediante el sensor). Esta *señal de error* inicia las acciones

tendientes a acercar la respuesta actual del sistema a la deseada, lo cual conducirá a la señal de error a acercarse a cero.

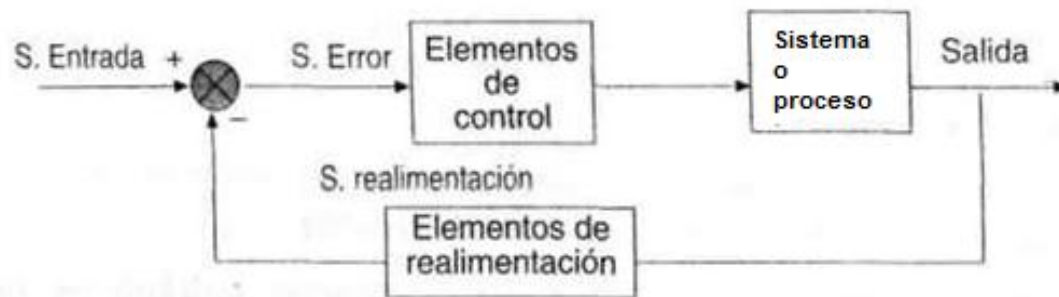


Figura 25. Sistema de control en Lazo cerrado.

➤ Control en lazo abierto

Los sistemas en los cuales la salida no tiene efectos sobre la acción de control se denominan sistemas de control en lazo abierto. En otras palabras, en un sistema de control en lazo abierto no se mide la salida ni se realimenta para compararla con la entrada de referencia.

En cualquier sistema de control de lazo abierto, a cada entrada de referencia le corresponde una condición de operación fija; como resultado de ello, la precisión del sistema depende de la calibración.

Ante la presencia de perturbaciones, un sistema de control en lazo abierto no realiza la tarea deseada.

En la práctica el control en lazo abierto sólo se usa si se conoce la relación entre la entrada y la salida y si no hay perturbaciones internas ni externas. Es evidente que las condiciones anteriores son prácticamente imposibles de tener, por lo tanto, para

controlar la velocidad del actuador, hemos descartado la posibilidad de emplear este tipo de sistema.

➤ Controladores automáticos

Un control automático compara el valor real de la salida de una planta⁸ con la entrada de referencia, determina la desviación y produce una señal de control que reduce la desviación a cero o a un valor pequeño. En nuestro trabajo, estas acciones la realiza el microcontrolador; esta tarea se lleva a cabo cada vez que el *timer 3* genera una interrupción, lo anterior ocurre cada 43.69 milisegundos.

La figura siguiente muestra un diagrama de bloques de un sistema de control que consiste en un controlador automático, un actuador, una planta y un sensor.

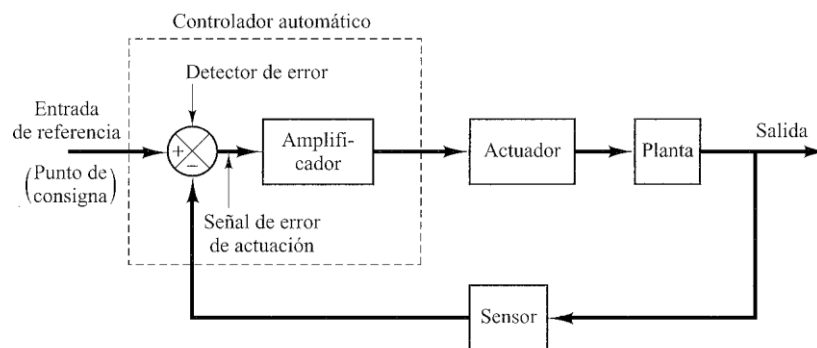


Figura 26. Diagrama de bloques de un sistema de control. [Ogata 2004]

El *sensor* (o elemento de medición) es el que convierte la señal de salida en otra variable manejable, como un desplazamiento, una presión o un voltaje, que puede usarse para comparar la señal de salida con la entrada de referencia. Este elemento

⁸ Se denomina planta a cualquier objeto físico que se va a controlar.

está en la trayectoria de realimentación en lazo cerrado. En nuestro caso, la velocidad de desplazamiento del actuador es leída por un sensor de desplazamiento lineal el cual produce un tren de pulsos cuya frecuencia es proporcional a la magnitud de la velocidad que presenta el actuador en cada instante.

Nótese que, en el caso de nuestro sistema de control, la planta es el actuador, el controlador es el microcontrolador, la entrada de referencia es la velocidad deseada (dada en un valor PWM), el actuador es el motor y el sensor es el *sensor de desplazamiento lineal*.

➤ Clasificación de los controladores

La manera en la cual el controlador automático produce la señal de control se denomina *acción de control*. Los controladores se clasifican, de acuerdo con sus acciones de control, como:

1. De dos posiciones o controladores *on-off*

En un sistema de control de dos posiciones, el elemento de acción sólo tiene dos posiciones fijas que, en muchos casos, son simplemente encendido y apagado. El control de dos posiciones o de encendido y apagado es relativamente simple y barato, razón por la cual su uso es extendido en sistemas de control industriales.

La desventaja de este controlador radica en que sólo es útil para manejar actuadores de dos posiciones y éstos son sometidos a un ciclo de ajuste continuo lo cual puede ocasionar que, en algún momento, fallen.

2. Controladores proporcionales

En un controlador con acción de control proporcional, el actuador no es forzado a tomar una de dos posiciones disponibles. En lugar de esto, tiene un rango continuo de posiciones disponibles. La posición que toma es proporcional a la señal del error. En otras palabras, la salida del bloque controlador (figura 26) es proporcional a su entrada. Este control tiene una ventaja importante sobre el control todo o nada: elimina la constante oscilación alrededor del valor de referencia. Con esto proporciona un control de la planta más preciso y reduce el desgaste de actuadores mecánicos.

La desventaja es que si la planta no posee integradores, siempre habrá un offset o error en estado permanente.

3. Controladores integrales

En un controlador integral, el valor de la salida del controlador se ajusta proporcionalmente a la señal del error. Con un controlador de este tipo se puede llevar el error a cero lo que es, sin duda, deseable en cualquier sistema de control.

Su desventaja radica en que la respuesta del integrador es relativamente lenta, es decir, se alcanza el estado estable después de un tiempo considerable.

4. Controladores proporcionales-integrales (PI)

Este controlador es la suma de una acción proporcional y una integral. Se ha visto que la acción proporcional nos acerca al valor deseado, mientras que la acción integral nos lleva exactamente al valor deseado.

Además de ser lenta su respuesta, con el controlador integral se presentan pequeñas oscilaciones que en algunas plantas serían indeseables. Por otro lado, la respuesta

proporcional, aunque sólo se acerca a la referencia, su respuesta es rápida y no presenta oscilaciones.

Es por eso que se combinan ambas acciones para tener los beneficios de una respuesta rápida sin oscilaciones de una acción proporcional y una respuesta que nos lleve exactamente al valor deseado con la acción integral.

Aún cuando el control proporcional-integral (PI) es adecuado para la mayoría de las situaciones de control, no es adecuado para todas. Hay algunos procesos que presentan problemas de control muy difíciles que no pueden manejarse con un control PI. Por ejemplo, cuando existen cambios muy bruscos en la carga o retardos de tiempo grandes entre la aplicación de la acción correctiva y la aparición sus resultados en la variable medida, es recomendable el empleo de un control PID.

5. Controladores proporcionales-derivativos (PD)

La acción de control derivativa es llamada también *control de velocidad* y tiene la ventaja de ser de previsión, pero amplifica las señales de ruido. Nunca se emplea sola y es útil únicamente en el estado transitorio.

Con este controlador el error al inicio es relativamente grande debido a la acción derivativa. Después el error disminuye y el efecto derivativo disminuye también. Al final sólo queda el efecto proporcional que se aproxima a la referencia.

6. Controladores proporcionales-integrales-derivativos (PID)

Esta acción combinada tiene las ventajas de cada una de las tres acciones de control individuales.

Si el controlador fuese sólo proporcional, es decir, si la variable de control fuese únicamente proporcional al error, el controlador produciría un error en estado estacionario o, también llamado, offset. La función principal de la acción integral es asegurar que la salida del proceso concuerde con la referencia en estado estacionario.

El error en estado estacionario siempre es cero con la acción integral, sin embargo, puede conducir a una respuesta oscilatoria de amplitud decreciente lenta o, incluso, de amplitud creciente, y ambos casos, se consideran indeseables.

El propósito de la acción derivativa es mejorar la estabilidad de lazo cerrado. Una ventaja de usar una acción de control derivativa es que responde a la velocidad del cambio del error y produce una corrección significativa antes de que la magnitud del error se vuelva demasiado grande. Por tanto, el control derivativo prevé el error, inicia una acción correctiva oportuna y tiende a aumentar la estabilidad del sistema.

Debido a que el control derivativo opera sobre la velocidad de cambio del error en estado estacionario, y no sólo sobre el error mismo, este modo nunca se utiliza solo.

En el campo de los sistemas para control de procesos, es un hecho bien conocido que los esquemas de control PID básicos y modificados han demostrado su utilidad para aportar un control satisfactorio, aunque tal vez en muchas situaciones específicas no aporten un control óptimo [Ogata 2004].

➤ **Control analógico y control digital**

Dentro de los sistemas de control, existe una clasificación de acuerdo a la naturaleza de las señales con las que se trabaja. Existen por tanto sistemas de control *analógico* y sistemas de control *digital*.

Un *sistema de control digital* o *sistema en tiempo discreto*, tiene señales que pueden cambiar su valor sólo en instantes discretos de tiempo. Nos referimos a sistemas en tiempo continuo o sistemas *analógicos* a aquellos en los cuales todas las señales pueden cambiar continuamente.

En el control de sistemas, La transformada de *Laplace* es una herramienta matemática útil al momento de modelar sistemas analógicos lineales invariantes en el tiempo; puede ser también utilizada para encontrar la respuesta temporal en este tipo de sistemas.

En contraste con los sistemas en tiempo continuo cuya operación se describe o modela mediante un conjunto de ecuaciones diferenciales, los sistemas en tiempo discreto son aquellos cuya operación se describe mediante un conjunto de *ecuaciones en diferencia*.

El método de transformación empleado en el análisis de sistemas en tiempo continuo lineales e invariantes en el tiempo, es el de la *transformada de Laplace*; de manera similar, la transformada usada en el análisis de sistemas en tiempo discreto invariantes en el tiempo y lineales es la *transformada z*.

La mayoría de los procesos a controlar utilizan señales en tiempo continuo; por lo que si el sistema de control involucra este tipo de señales se tiene que hacer una conversión de señales continuas a señales digitales a través de un *Convertidor Analógico Digital* (ADC).

En los últimos años, los sistemas de control digitales han tenido un gran avance y una aceptación muy importante debido, principalmente, a su bajo costo y a los avances que han surgido en los dispositivos electrónicos en general, y a los microcontroladores en particular.

Controlador de la velocidad del actuador

Muchos sistemas industriales son controlados por el clásico controlador proporcional-integral-derivativo (PID). Esto se hace a pesar de que el sistema sea no-lineal y a pesar del hecho de que el concepto, a menudo, limita el desempeño. La razón por la cual los controladores PID han ganado tal popularidad es que el conocimiento detallado del sistema no es requerido pero el controlador puede ser ajustado por medio de simples reglas de sintonización o por medio de PID “autoajustables” [Nguyen 2003].

El esquema básico del controlador PID se muestra en la figura siguiente.

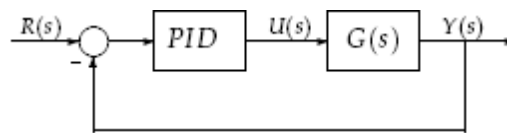


Figura 27. Esquema básico de un controlador PID. [(M. Castillo s.f.)]

Matemáticamente, el algoritmo de control PID se representa de la siguiente manera:

$$u(t) = K \left[e(t) + \frac{1}{T_i} \int_0^t e(t) dt + T_d \frac{de(t)}{dt} \right]$$

Ecuación 12

Donde $u(t)$ es la acción de control y $e(t)$ es el error dado por $e(s) = R(s) - Y(s)$. De esta manera, la variable de control es una suma de tres términos: el término P , que es proporcional al error; el término I , que es proporcional a la integral del error, y el término D , que es proporcional a la derivada del error. Los parámetros del controlador son la ganancia proporcional K , el tiempo integral T_i (o tiempo de reajuste) y el tiempo derivativo T_d (o tiempo de adelanto).

Si transformamos por el método de *Laplace* la ecuación 12, el controlador PID puede ser escrito como:

$$U(s) = K \left[1 + \frac{1}{TiS} + TdS \right] E(S)$$

Ecuación 13

La forma discreta del controlador PID puede obtenerse encontrando la transformada *Z* de la ecuación 13:

$$U(Z) = K \left[1 - \frac{T}{2Ti} + \frac{T}{Ti(1 - Z^{-1})} + Td \frac{(1 - Z^{-1})}{T} \right] E(Z)$$

O bien:

$$\frac{U(Z)}{E(Z)} = Kp + \frac{Ki}{1 - Z^{-1}} + Kd(1 - Z^{-1})$$

Ecuación 14

Donde:

$K_p = K - \frac{KT}{2Ti} = K - \frac{Ki}{2}$	Ganancia Proporcional.
$K_i = \frac{KT}{Ti}$	Ganancia Integral
$K_d = \frac{KTd}{T}$	Ganancia Derivativa

y *T* es el periodo de muestreo.

La ecuación 14 puede ser representada, en forma esquemática, como muestra la figura siguiente.

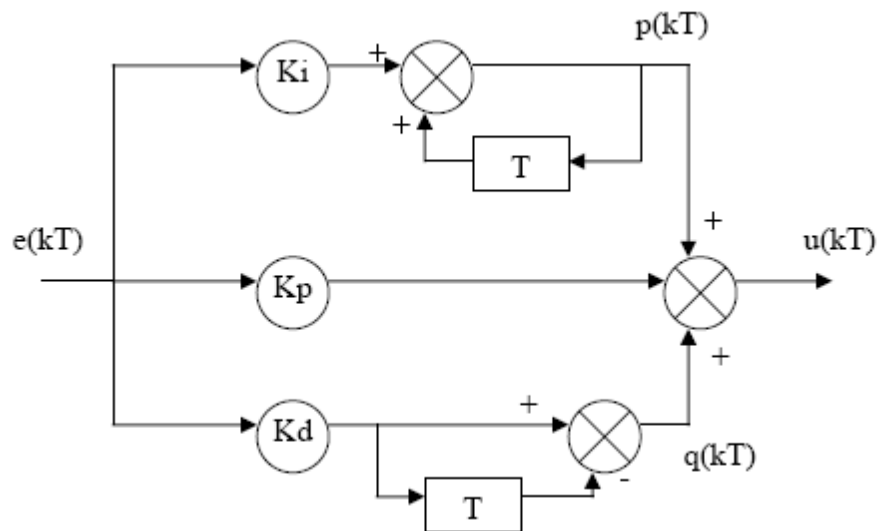


Figura 28. Implementación paralela de un controlador PID. [Narvaez 2008]

El diagrama representado en la figura anterior es de gran utilidad, ya que facilita la generación del código, en cualquier lenguaje de programación, necesario para la implementación del controlador *PID* de una manera digital. En nuestro caso el firmware para el microcontrolador 2 se generó en lenguaje *C* y, como herramienta de programación, se empleó el compilador CCS®.

Sintonización de los parámetros del controlador

La sintonización de los controladores *PID*, consiste en la determinación del ajuste de sus parámetros K , T_i y T_d , para determinar un comportamiento del sistema de control aceptable de conformidad con algún criterio de desempeño establecido.

Conociendo el modelo matemático del sistema de la máquina, es decir su función de transferencia de lazo abierto, podemos aplicar numerosos métodos que incluye la representación de Bode, método de *Laplace*, respuesta en frecuencia, criterio de

Nyquist, lugar geométrico de las raíces, entre otros, para hallar los parámetros del PID adecuados. Existen también, métodos experimentales tales como el ideado en 1942 por John G. Ziegler y Nathaniel B. Nichols que partiendo de la respuesta del proceso a un escalón unitario en lazo abierto, se pueden obtener los parámetros del PID. Ziegler y Nichols también describieron una técnica de ajuste en lazo cerrado.

En casos donde no tenemos el modelo del motor, ni podemos aplicar métodos experimentales como el Ziegler-Nichols, se puede utilizar un método de prueba y error basado en el conocimiento de lo que cada acción de control (K_p , k_i y k_d) aporta a la acción de control general del sistema. Por simplicidad utilizamos esta técnica en el presente trabajo, la cual se describe a continuación:

1. Se requiere de una manera de variar instantáneamente el valor de referencia (set-point) de cero al máximo de la velocidad esperada. Esto es equivalente aplicar un escalón a la entrada del controlador.
2. *Agregar control proporcional para mejorar el tiempo de asentamiento.* Teniendo la ganancia integral y derivativa a cero, se aplica un escalón repetidamente variando cada vez la ganancia proporcional hasta obtener un valor razonable de K_p . Esto es, que la respuesta se aproxime a la entrada y oscile continuamente alrededor de la entrada con una pequeña cantidad de amortiguamiento.
3. *Agregar control proporcional para mejorar el sobrepaso.* Luego que se encuentre un valor aceptable para K_p , se trabaja con el ajuste de K_d . Aplique un escalón repetidamente variando cada vez la ganancia derivativa hasta obtener un valor razonable de K_d . Esto es que la respuesta se amortigüe rápidamente eliminando las oscilaciones alrededor de la entrada y permaneciendo en un valor ligeramente menor que el valor de la entrada. Esto es lo que se denomina error de estado estable y que puede ser corregido con el ajuste de K_i .

4. *Agregar control proporcional para eliminar el error en estado estable.* Luego que se encuentre un valor razonable para K_d , se trabaja con el ajuste de K_i . Se aplica un escalón repetidamente variando cada vez la ganancia integral hasta obtener un valor razonable de K_i . Esto es eliminar el error de estado estable tal que la respuesta sigue a la entrada. Este parámetro debe ajustarse cuidadosamente pues puede introducir inestabilidad en el sistema.

Modulación de ancho de pulso (PWM)

La modulación de ancho de pulso es un mecanismo utilizado para variar la velocidad de un motor de *CD* con escobillas. Consiste en variar el voltaje promedio aplicado al motor, haciendo para ello una variación al ancho del pulso o *duty cycle* en un ciclo de *PWM*, tal como se muestra en la figura siguiente.

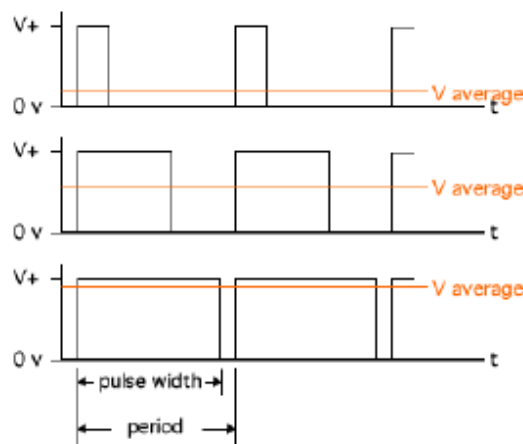


Figura 29. Modulación de ancho de pulso (PWM). [Narvaez 2008]

Tres señales son presentadas en la figura de 29, todas tienen la misma frecuencia, pero el ancho de los pulsos es diferente y, por lo tanto, se obtienen voltajes promedio distintos.

PWM es una manera eficiente, para simular un rango de valores analógicos usando circuitos digitales. Conmutando rápidamente entre cero volts (“0”) y el voltaje nominal del motor (“1”), se obtiene un valor promedio comprendido entre estos dos límites.

Con un microcontrolador es fácil generar una señal PWM, la parte del firmware correspondiente al control de la velocidad del actuador, cumple con la tarea de calcular la anchura de pulso en cada instante para lograr la velocidad constante independientemente de las variaciones de carga, esta señal tiene una frecuencia de 5.0 [KHz] y es enviada al motor a través de un *punte H⁹*.

El control que se implementó soporta únicamente perturbaciones debidas a la variación en la carga, se deja fuera la posibilidad otro tipo de perturbaciones tales como un variación en el voltaje de la línea de alimentación.

Programación del microcontrolador 2

A partir de la figura 28 se generó el siguiente pseudo-código que, a su vez, facilitó la elaboración de una parte del firmware¹⁰ para el microcontrolador 2 (PIC18F2550), correspondiente a la interrupción que genera el *timer 3 la cual ocurre cada 43.69 [ms]*.

⁹ Ver capítulo 5. Apartado: Etapa de potencia.

¹⁰ Ver programa comentado en el anexo 3

BEGIN

DO FOREVER

Obtenga el set point: $r(kT)$

Obtenga la salida del sistema: $y(kT)$

Calcule el error: $e(kT) = r(kT) - y(kT)$

Calcule el término I: $p(kT) = k_i * e(kT) + p(kT - T)$

Calcule el término D: $q(kT) = k_d * e(kT) - k_d * e(kT - T)$

Calcule la salida PID: $u(kT) = p(kT) + k_p * e(kT) + q(kT)$

Envíe control al motor

Guarde variables: $p(kT - T) = p(kT)$

$e(kT - T) = e(kT)$

Espere hasta el próximo periodo de muestreo

ENDDO

END

CAPÍTULO 5. ELECTRÓNICA

En este capítulo se mencionan, los dispositivos de electrónica necesarios para hacer posible el correcto funcionamiento del equipo desarrollado, ya que para la comunicación entre la interfaz, sensores y actuador es necesario de la implementación de diversos dispositivos electrónicos cuyo funcionamiento depende del flujo de electrones para la generación, transmisión, recepción, y almacenamiento de información.

Los circuitos electrónicos ofrecen diferentes funciones para procesar la información, incluyendo la amplificación de señales débiles hasta un nivel que se pueda utilizar; la extracción de información, como por ejemplo la recuperación de la señal de sonido de una onda de radio; y operaciones lógicas, como los procesos electrónicos que tienen lugar en las computadoras.

5.1 MICROCONTROLADOR

Un microcontrolador es un circuito integrado de alta escala de integración capaz de llevar a cabo procesos lógicos. Estos procesos o acciones son programados por el usuario en lenguaje ensamblador o en un lenguaje de alto nivel (como C o Basic), y son introducidos en el microcontrolador a través de un programador. Un microcontrolador incorpora la mayor parte de los elementos que configuran un controlador y contiene todos los componentes fundamentales de una computadora, tales como la CPU, la memoria, puertos de entrada y salida (figura 30), pero a diferencia de una computadora de propósito general, un microcontrolador es diseñado para una tarea específica, generalmente controlar un sistema en particular.

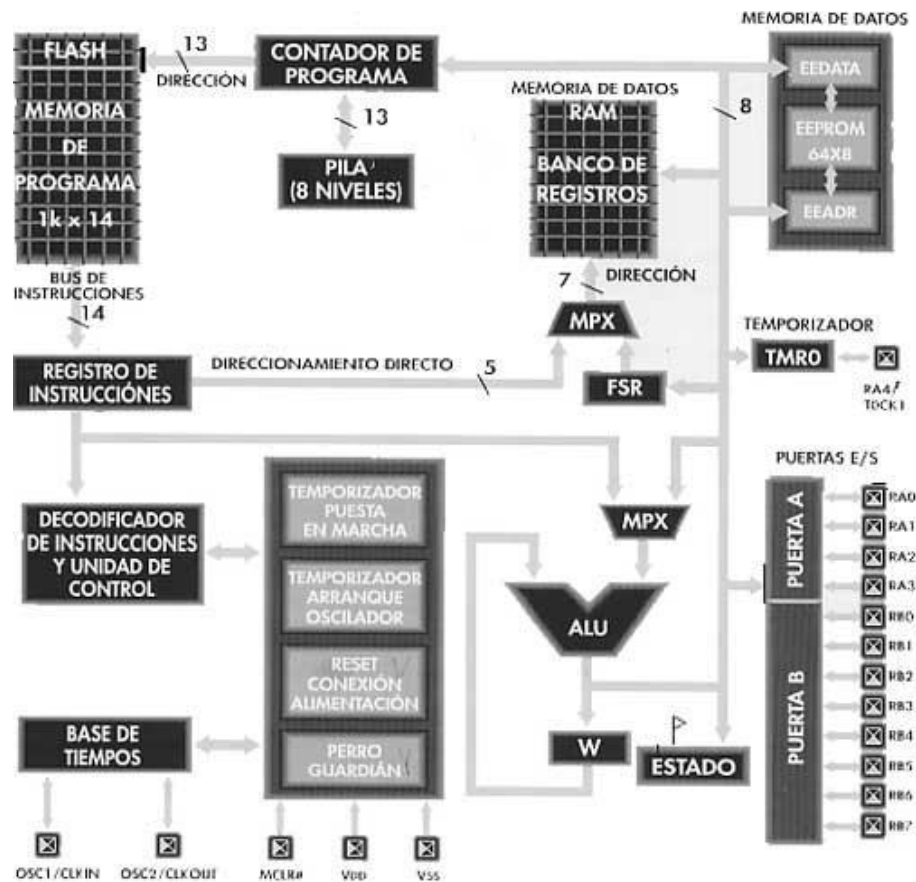


Figura 30. Arquitectura de un microcontrolador. [Pobladores.com s.f.]

Los microcontroladores están siendo empleados en multitud de sistemas presentes en nuestra vida diaria, como pueden ser juguetes, hornos de microondas, refrigeradores, televisores, impresoras, módems, automóviles, por mencionar sólo algunos. Una aplicación típica podría emplear varios microcontroladores para controlar pequeñas partes del sistema.

Cada vez existen más productos que incorporan un microcontrolador con el fin de aumentar sustancialmente sus prestaciones, reducir su tamaño y costo, mejorar su fiabilidad y disminuir el consumo de energía, es por ello que en este proyecto de tesis se optó por utilizar dos microcontroladores, uno como dispositivo de adquisición de

datos y como medio de comunicación entre el CPU y el dispositivo; y un segundo específicamente como controlador. Se optó por utilizar dos microcontroladores en lugar de uno, para evitar pérdida de pulsos del encoder, básicos para el control de velocidad, durante las interrupciones en que el microcontrolador 1 está en comunicación con el CPU.

5.2 CELDA DE CARGA

Una celda de carga es un transductor que convierte una fuerza en una señal eléctrica medible. Este transductor es una configuración especial de cuatro galgas extensométricas.

La galga extensométrica consiste en un cable muy fino, más comúnmente, una hoja metálica organizada en un patrón de rejilla. El patrón de rejilla maximiza la cantidad de cable metálico, o de hoja, sujeto a tensión en la dirección paralela (figura 31). La rejilla se une a un delgado respaldo, denominado el portador, el cual se sujeta directamente al espécimen de prueba. Por tanto, la tensión experimentada por el espécimen de prueba se transfiere directamente a la galga extensométrica, la cual responde mediante un cambio lineal en la resistencia eléctrica.

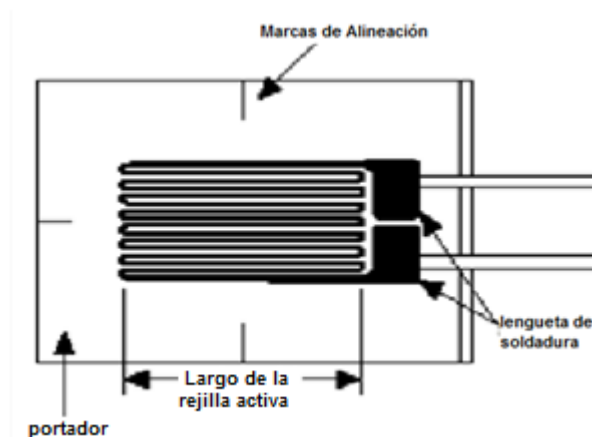


Figura 31. Galga extensiométrica [National Instruments s.f.]

En la práctica, las mediciones de tensión rara vez involucran cantidades mayores a unas pocas milésimas. Por tanto, la medición de tensión requiere de exactitud en la detección de cambios muy pequeños en resistencia. Para medir tales cambios en la resistencia, las galgas extensométricas casi siempre se emplean en configuraciones de puente de Wheatstone con una fuente de excitación de voltaje. El puente general de Wheatstone, que se ilustra en la figura 32, consiste de cuatro brazos resistivos con un voltaje de excitación, V_{EX} , que es aplicado a través del puente. [National Instruments s.f.]

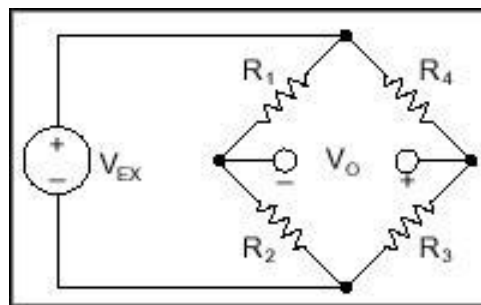


Figura 32. Configuración del puente de Wheatstone. [National Instruments s.f.]

El voltaje de salida del puente, V_O , es igual a:

$$V_O = \left[\frac{R_3}{R_3 + R_4} - \frac{R_2}{R_1 + R_2} \right] V_{EX}$$

Ecuación 15

De esta ecuación, se aprecia que cuando la corriente que fluye en la rama **R1/R2** es igual a la que fluye por **R4/R3**, el voltaje de salida, V_O , es cero. Bajo estas condiciones, se dice que el puente está balanceado. Cualquier cambio en la resistencia de cualquiera de los brazos del puente resultará en un voltaje de salida diferente de cero.

Por tanto, si se reemplaza cualquiera de las 4 resistencias que conforman el puente en la figura 32 por una galga extensométrica activa, cualquier cambio en la resistencia de

esta galga desbalancea el puente y produce un voltaje de salida diferente de cero. Este tipo de configuración es la que se utiliza en las celdas de carga y gracias a la cual deben su sensibilidad y precisión.

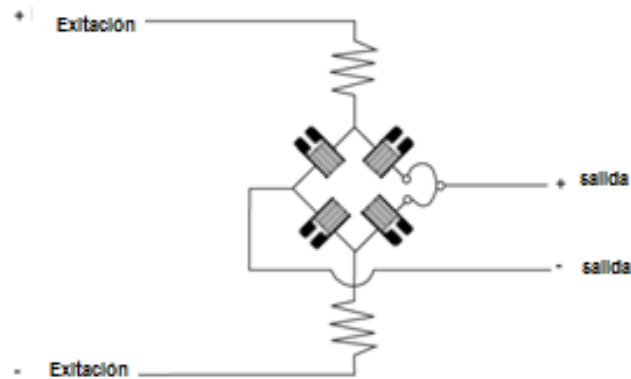


Figura 33. Celda de carga. [Rice LakeWeighing System 2002]

Existen diversos tipos de celdas de carga y su elección depende del tamaño y la capacidad de las mismas así como de su aplicación. Los tipos más comunes son:

➤ **Celda de carga *canister***

La *canister* es la celda de carga más antigua (figura 34). Está sellada herméticamente, lo que le provee excelente protección del medio ambiente. Las celdas *canister* pueden ser adquiridas para medir tensión, compresión o ambas. Su aplicación más común es medir presión dentro de tanques o tolvas.



Figura 34 Celda de carga Canister. [Omega.com s.f.]

➤ **Celda de carga tipo S**

Las celda de carga tipo S obtiene su nombre de su forma (figura 35). Las medidas de tensión son vinculadas al área central conformada por un puente de Wheatstone. Sus usos incluyen medir el nivel de tanques y tolvas.



Figura 35. Celda de carga tipo S. [Omega.com s.f.]

➤ **Celda de carga *bending beam***

En este tipo de celdas, las galgas están ubicadas en las secciones planas superiores e inferiores de la celda en los puntos de tensión máxima. Este tipo de celda de carga es usado para baja capacidad y trabaja con linealidad razonable. Una de sus principales ventajas es que está construida para utilizarse dentro de espacios reducidos.



Figura 36. Celda de carga Bending Beam. [Omega.com s.f.]

➤ **Celda de carga *shear beam***

Las galgas son ubicadas en una parte reducida de la sección cruzada del dispositivo para maximizar el efecto de la presión, tal como se observa en la figura 37. Son usadas para capacidades de medianas a grandes, esta celda de carga tiene buena linealidad y no es tan susceptible a cargas extrañas, en particular a cargas laterales.



Figura 37. Celda de carga Shear Beam. [Omega.com s.f.]

➤ **Celdas de carga de plataforma**

Las celdas de carga de **plataforma** son también llamadas celdas **unipunto**. Son usualmente utilizadas en rangos de escala de baja capacidad. Son comúnmente fabricadas en aluminio (figura 38). En algunas escalas, este tipo de celdas son fabricadas con un tope de sobrecarga. En este tipo de celdas son permitidas sobrecargas del 200% de la capacidad máxima sobre el punto de carga central. Las celdas de plataforma son comúnmente fabricadas con capacidad desde 2 hasta 1000 [kg].



Figura 38. Celda de carga de plataforma. [Omega.com s.f.]

5.3 SENSOR DE DESPLAZAMIENTO LINEAL

El sensor fotoeléctrico HEDS, es un sensor comercial desarrollado por la compañía USDigital®, el cual podría describirse como un *sensor* óptico tradicional, ya que está diseñado para detectar la posición lineal cuando es usado con un *encoder* lineal o la posición angular cuando es usado con un *encoder* rotatorio. El sensor HEDS consta de una fuente emisora de luz (LED) y un detector monolítico (fabricado en un solo cristal, habitualmente de silicio, pero también existen en germanio, arseniuro de galio y silicio-germanio) ambos encapsulados en un pequeño módulo fabricado de polímero (figura 39).

Este sensor de posicionamiento lineal entrega salidas digitales de cuadratura y un tercer canal puede entregar un *index*, dependiendo del modelo. Este sensor se alimenta con una fuente de 5 [V] de corriente directa, es compatible con tecnología TTL, tiene una resolución mayor a 250 cuentas por pulgada (CPI), dependiendo del modelo; en nuestro caso se eligió un sensor con una resolución de 360 CPI, ya que con dicha resolución se tiene una medición cada 0.0705 milímetros.



Figura 39. Sensor HEADS. [US-Digital 2008]

La regleta lineal que forma parte de este sensor, está fabricada en poliéster Mylar[®], el cual es un film flexible, fuerte y duradero con un abanico de propiedades que lo hacen recomendable en múltiples aplicaciones industriales.

Como se puede observar en la figura 40, las dimensiones del *encoder* lineal son muy pequeñas, por lo que su manejo debe realizarse con los cuidados pertinentes para evitar dañarla y lograr que funcione adecuadamente, en conjunto con el HEADS.

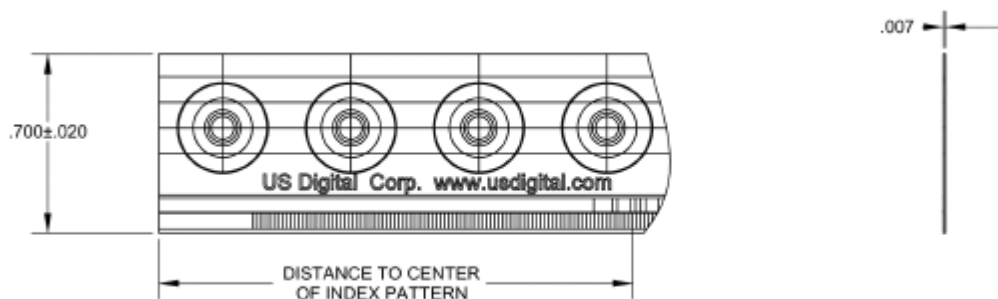


Figura 40. Encoder lineal. [(US-Digital 2008)]

5.4 ACONDICIONAMIENTO DE LA SEÑAL DEL SENSOR DE DESPLAZAMIENTO

Una vez que se acciona el motor, el *encoder* lineal (cuyo desplazamiento depende directamente del movimiento del motor) envía dos señales en cuadratura, **QA** y **QB**, desfasadas una de otra 90°; sin embargo, si se leyeran por el microcontrolador esas dos variables de manera directa, se tendría una resolución de .070[mm.], la cual nos pareció insuficiente, por lo que se decidió aumentar la resolución, y el dispositivo electrónico que se utilizó para este fin, fue una GAL16V8. La GAL es un PLD (Dispositivo Lógico Programable) con tecnología ECMOS reprogramable y reconfigurable con borrado de alta velocidad.

Gracias a la implementación de este circuito, se puede incrementar por software la resolución del *encoder*, esto se logró gracias a un programa en Wincupl[®] desarrollado previamente en el CCADET. Este programa permite captar los pulsos entregados por el sensor de desplazamiento lineal e incrementa la resolución por cuatro, obteniendo una resolución de 0.0175[mm], asimismo permite detectar el sentido en el que se está desplazando el actuador, y entrega a la salida nuevamente dos señales, llamadas **Countup** y **Countdown**.

5.5 ACONDICIONAMIENTO DE LA SEÑAL DE LA CELDA DE CARGA

La celda de carga, como se mencionó con anterioridad, entrega un voltaje diferencial muy pequeño, del orden de milivolts; por lo que para poder tener una medición correcta de este voltaje es necesario acondicionar la señal, de tal forma que cualquier cambio en la salida sea apreciable y no se pierda la resolución que entrega la celda de carga.

La celda de carga que se utilizó para este proyecto entrega un voltaje de 2 [mV/V] y debe alimentarse en un rango de 10 a 15 volts, lo que implica que si se alimenta la celda a 10[V], en su carga máxima entrega 20 [mV]. Esto muestra claramente que los valores de voltaje que entrega este dispositivo son muy pequeños, por lo que es necesario amplificarlos, ya que de no hacerlo el microcontrolador no podría detectar cambios significativos en la fuerza aplicada, debido a que, por la resolución del convertidor analógico digital con el que cuenta, el mínimo cambio que detecta es de 4.8 [mV] lo que daría por resultado que sólo se detectarían cambios cada 1200 [g] lo cual no sería representativo para la prueba.

Para amplificar una señal se utiliza generalmente un amplificador operacional (AO), que es un amplificador diferencial con una ganancia muy alta, con una elevada impedancia de entrada y una impedancia de salida baja.

Sin embargo, al amplificar una señal, se corre el riesgo de amplificar el ruido que la acompaña, por lo que es importante hacer una correcta implementación de la etapa de acondicionamiento.

Para el acondicionamiento de la señal de la celda de carga se utilizó un amplificador operacional de instrumentación, lo cual permite tener una ganancia preestablecida de manera muy precisa y estable, además evita, en gran medida, amplificar el ruido a la entrada. El AO que se utilizó es el AD624, de la marca Analog Device[®], que permite tener diversas ganancias fijas gracias a sus precisas resistencias internas, las cuales al conectarse en serie o en paralelo (simple conexión de pines del mismo amplificador) pueden proporcionar ganancias predeterminadas diversas que van desde 1 hasta 1000; para nuestro desarrollo se eligió una ganancia de 250, con lo cual se logra que el microcontrolador detecte cambios cada 4.2 gramos lo cual para este tipo de pruebas es una buena resolución.

5.6 ETAPA DE POTENCIA

Debido a que para este prototipo se utiliza un motor, el cual al estar en funcionamiento consume una corriente, se requiere del uso de una etapa de potencia, que permita la correcta actividad del motor, y de la electrónica en general, sin problemas de ruido eléctrico y que además permita tener una ganancia de corriente. Como etapa de potencia, el dispositivo más utilizado es el puente H, ya que además de servir como etapa de potencia, permite el cambio de sentido de giro del motor, lo cual para nuestro equipo es indispensable.

El puente H es simplemente un circuito para controlar motores de corriente continua, es capaz de controlar una corriente eléctrica, amplificándola y/o conmutándola. El nombre se refiere a la posición en que quedan los transistores en el diagrama del circuito, tal como se puede observar en la siguiente figura.

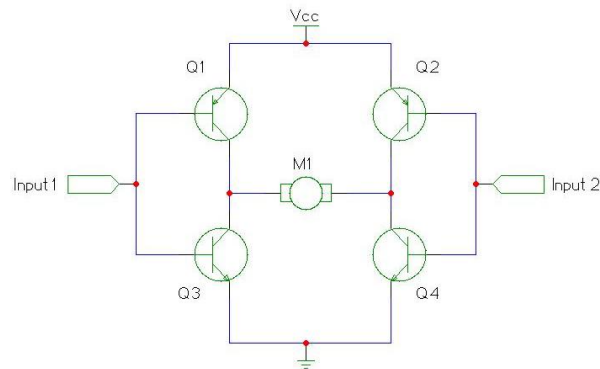


Figura 41. Puente H. [Monigot.com s.f.]

El puente H que se utilizó para este prototipo es el integrado L293E, por sus características eléctricas, ya que nos permite un consumo máximo de 1[A] y sólo se requieren 400 [mA].

5.7 FUENTE DE ALIMENTACIÓN

Para el funcionamiento de la “Máquina para ensayos de flexión en muestras de plantas de dimensiones pequeñas” se requiere de una fuente de alimentación, que sea capaz de proporcionar el voltaje y la corriente necesaria para el correcto funcionamiento del actuador, los sensores y la electrónica en general; ya que si bien el equipo se conectará a una PC a través del puerto USB, y este puerto puede entregar un voltaje de 5 [V] y una corriente máxima de 500 [mA], el consumo del equipo es mayor y requiere como mínimo una fuente de alimentación de 12[V] que suministre mínimo 1[A]; además de que se requiere de voltajes negativos para la alimentación del amplificador operacional.

Por todo lo anterior, se optó por comprar una fuente que cumpliera con dichos requisitos, ya que si bien era factible su fabricación, el ahorro económico que representaba era mínimo e implicaba invertir un tiempo innecesario.

La fuente que se eligió para este desarrollo, fue el modelo HF30W-TF-A¹¹ de la marca HF[®] ya que esta fuente es de dimensiones pequeñas, que si bien no era un requerimiento esencial es importante por las dimensiones del equipo, y cubre con los requerimientos del equipo al contar con las siguientes salidas:

- a) 5[V] 0.5 - 3.0[A]
- b) 12[V] 0.1 - 1.0 [A]
- c) -12[V] 0.1 - 0.3[A]

¹¹ Para mayor información ver hoja de especificaciones en el anexo 2

PRUEBAS Y RESULTADOS

Para analizar el comportamiento de la “Máquina para ensayos de flexión en muestras de plantas de dimensiones pequeñas” se decidió hacer pruebas de flexión con dos materiales: acero y madera, el primero por ser un material con un *modulo de Young* conocido y el segundo por tratarse del material con el que se utilizará el equipo.

Una vez que se puso en funcionamiento el equipo, se observaron dos problemas:

El primero fue mecánico, se observó que una vez que la celda entraba en contacto con la probeta, había un desplazamiento hacia arriba debido al juego existente entre la guía y la corredera, este problema se trató de solucionar maquinando nuevamente las piezas, sin embargo el problema se presentó nuevamente, por lo que se concluyó que se debía a las características del maquinado, es decir, el error era debido a las altas tolerancias entre ambas piezas, y no se pudo corregir debido a que los técnicos del taller de la *Sección de prototipos* del CCADET carecen de la capacitación y experiencia necesaria para realizar este tipo de piezas.

Si bien este problema nos pareció importante no nos trajo errores en la medición de la deflexión, ya que en la etapa de diseño se previó que la medición se hiciera lo más cercana al aplicador para evitar un posible error *abbe*, por lo que el encoder se colocó exactamente arriba del actuador, gracias a esto quedó resuelto este problema. Por otra parte, aunque existe un desplazamiento negativo, no se mide debido a las características del software; ya que las mediciones de desplazamiento sólo se hacen en la dirección de la aplicación de la fuerza, y se inician sólo cuando la celda detecta carga.

El segundo problema que se observó fue más serio, ya que afectaba directamente la medición de la fuerza aplicada; se detectó que durante la prueba a la entrada del convertidor analógico-digital existía mucho ruido eléctrico, reflejándose en lecturas erróneas de la celda de carga. Después de varias pruebas y análisis notamos que el ruido era, en su mayoría, generado por el contacto directo de la fuente con el equipo, y puesto que la fuente está aterrizada a su carcasa, al estar en contacto con la base del equipo se presentaba dicho problema; por lo tanto, se decidió colocar un filtro paso bajas al circuito y aislar física y eléctricamente la fuente del resto del equipo. Una vez implementadas ambas soluciones se vio resuelto el problema, no completamente, ya que el filtro aún deja pasar ruido eléctrico a ciertas frecuencias, pero ya no es tan representativo.

Una vez que se resolvieron todos los problemas se procedió a realizar las pruebas pertinentes para observar las características y especificaciones del prototipo, para ello se utilizó un alambre de acero inoxidable remanium®, estirado al diamante, con un diámetro de 0.89 [mm]; el cual, según las especificaciones del fabricante, cuenta con un módulo de elasticidad de 170000 [MPa]. [Dentaurum s.f.]

Para tener una segunda referencia sobre el valor del *módulo de Young* del acero con el que se caracterizó el equipo, y conocer el rango de error que manejan los equipos comerciales, se realizaron ensayos de flexión en una máquina Instron® modelo 5567 del *Laboratorio de materiales* de la División de Estudios de Posgrado de la Facultad de Odontología. Los resultados obtenidos en ese equipo se muestran en la siguiente imagen.

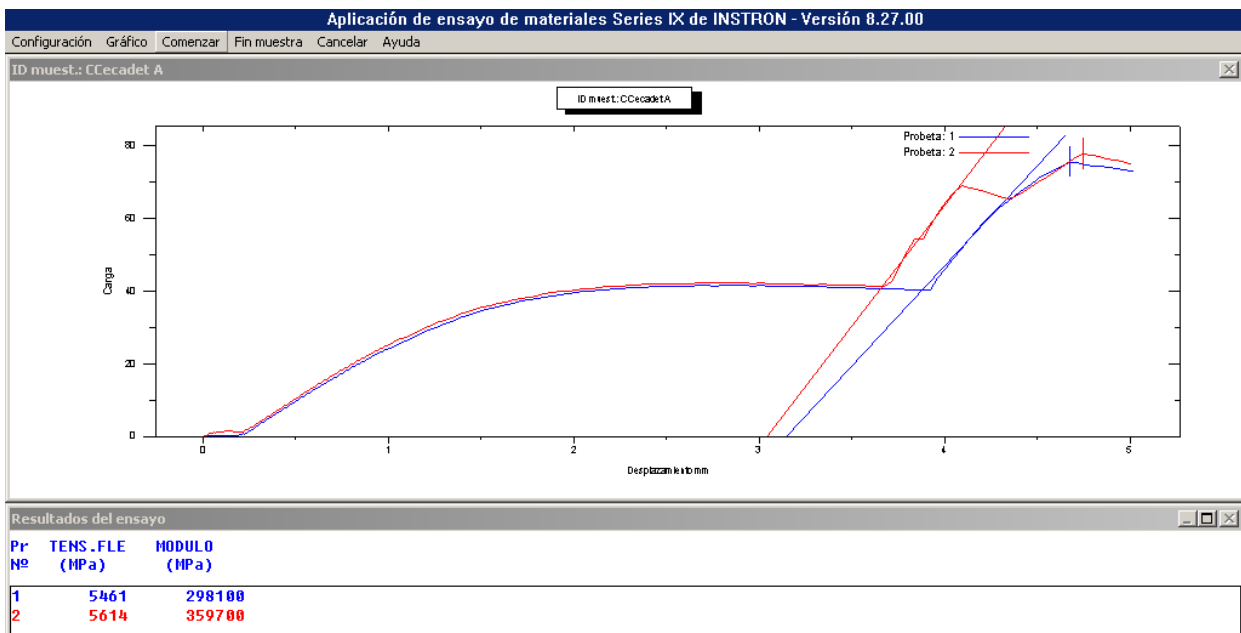


Figura 42. Ensayos de flexión en la máquina Intron®

Con nuestro equipo se realizaron varias pruebas con este mismo material obteniendo un valor promedio de 166,483.3 [MPa], algunos resultados obtenidos se muestran en la siguiente tabla.

Prueba	Módulo de Young [MPa]
1	188320
2	147632
3	170369
4	166308
5	189861
6	148434
7	187630
8	178611
9	145422

10	191767
11	171228
12	163744
13	143584
14	150782
15	153557

Tabla XI. Algunos valores obtenidos durante los ensayos con el acero.

A continuación se muestra una imagen que muestra una de las pruebas realizadas.

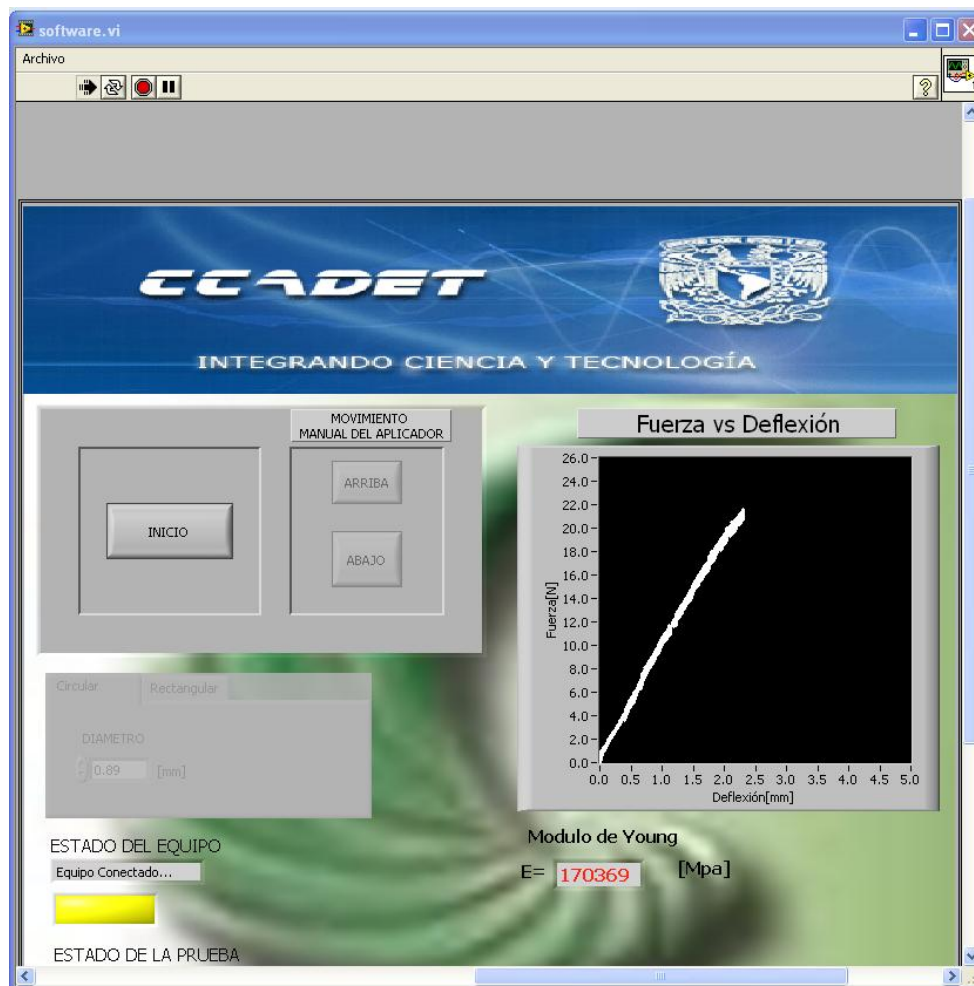


Figura 43. Ensayo de Flexión para el acero con el equipo.

Finalmente, se realizaron pruebas con un mondadientes, por ser éste el material más duro con el que se pretende utilizar nuestro equipo, y si bien no se tenía un valor previo del *módulo de Young* de la madera con la que están hechos estos palillos, comparamos nuestros resultados con los datos que se manejan para el pino, por ser ésta la madera con la que están hechos, este valor puede variar entre 8534-10791 [MPa] [Colombia s.f.], dependiendo del tipo de pino, el porcentaje de humedad y otros factores.

Los resultados obtenidos para los mondadientes se muestran a continuación:

Prueba	Modulo de Young en [MPa]
1	11934.3
2	8494.3
3	7620.7
4	10802.1
5	10881.8
6	8116.3
7	10071.8
8	9375.0
9	9538.8
10	11543.2
11	9968.8
12	10465.5
13	11570.9
14	8119.3
15	10828.1

Tabla XII. Algunos valores obtenidos durante las pruebas con los mondadientes.

Los resultados obtenidos dan un promedio de 9955.3 [MPa]. La siguiente imagen muestra una de las pruebas realizadas.

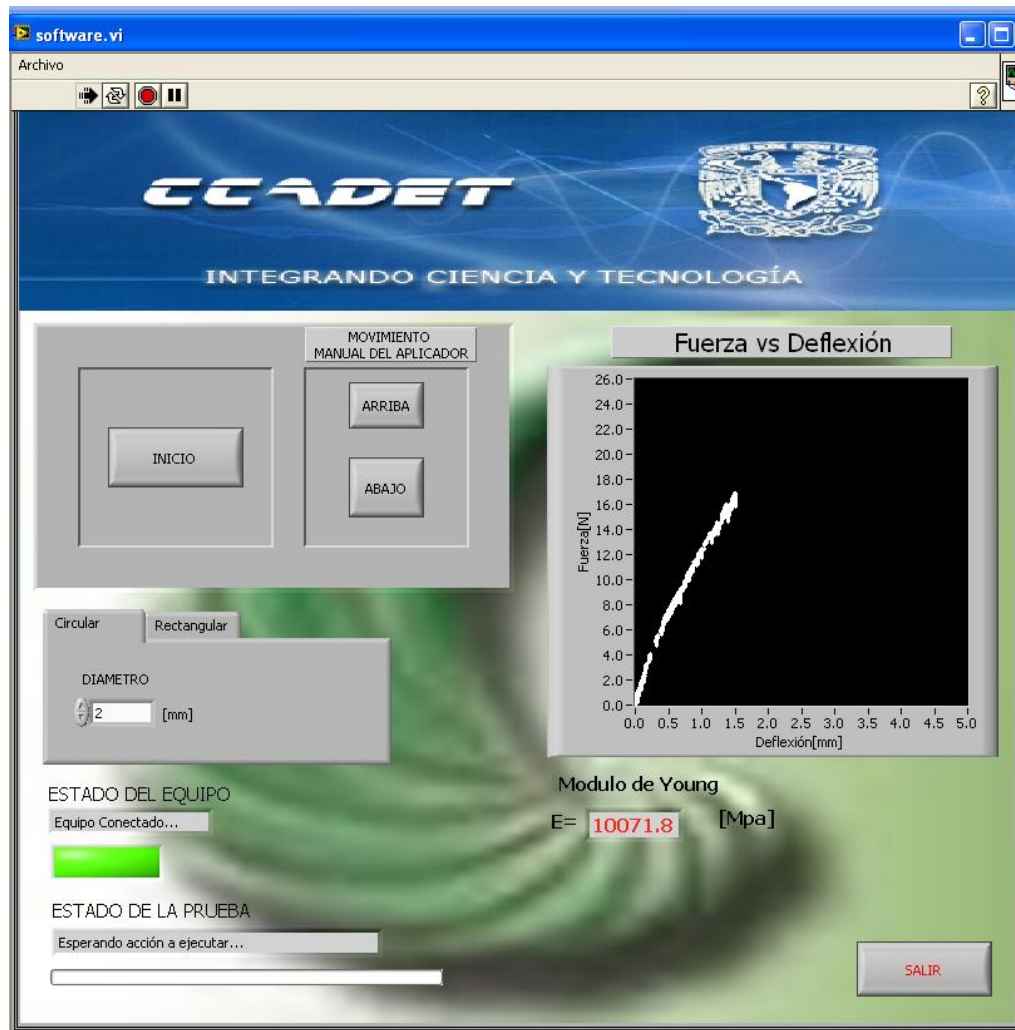


Figura 44. Ensayo de flexión de los mondadientes.

Como se puede observar en las pruebas que se realizaron con "La máquina para ensayos de flexión en muestras de plantas de dimensiones pequeñas" los resultados que se obtuvieron fueron muy cercanos a los valores establecidos, tanto por el fabricante, como por la bibliografía respectivamente; y en el caso del acero, los

resultados fueron aún mejores que los arrojados por la máquina Instron®, esto se pudo deber a diversos factores, sin embargo, es claro que nuestro equipo tiene un mejor desempeño por ser un equipo diseñado para este tipo de probetas en particular. El error que se tiene con respecto al valor manejado por el fabricante es de 2.068% para este material.

En cuanto a las pruebas realizadas con los mondadientes, si bien no estamos seguros de qué tipo de pino es el utilizado y cuál es su *módulo de Young*, los resultados que se obtuvieron se encuentran dentro del rango que se maneja para este tipo de madera, por lo que se asume que los valores son correctos y si consideramos que otros autores manejan 10,000[MPa] como valor promedio para el pino, nuestro error es de 0.47% para este material.

CONCLUSIONES

Finalizado este trabajo se puede observar que se alcanzaron los objetivos planteados al inicio, ya que “La máquina para ensayos de flexión en probetas de plantas de dimensiones pequeñas” cumple con las características especificadas por el cliente, es decir, se logró hacer un prototipo que realice ensayos de flexión en probetas de material orgánico con un diámetro de entre 1 y 3 [mm] y una longitud mínima de 30[mm], y su funcionamiento es similar a un equipo profesional de este tipo; ya que muestra la gráfica *fuerza-deflexión* y entrega el *módulo de Young* de la probeta.

Este primer prototipo además es ligero y compacto, por lo que su traslado de un lugar a otro es sencillo; su diseño permitió que su manufactura fuera fácil y de bajo costo; además permite un mantenimiento preventivo y correctivo asequible. Todas las piezas que se compraron, son de sencilla adquisición y el costo total aproximado del equipo, sin contar el desarrollo, fue de \$7,500, por lo que no se rebasó el presupuesto con el que se contaba.

El software desarrollado permite un modo de operación sencillo y amigable, ya que muestra en pantalla botones de operación, y los resultados del ensayo, el estado de la prueba además de mensajes en caso de error o sobrecarga del equipo, evitando al máximo posibles errores y daños al mismo.

Una vez concluido este trabajo, estamos en condiciones de mostrar en una tabla las especificaciones del producto para que éste pueda ser evaluado.

Especificación	Dato
Velocidad del actuador	24.2 [mm/min]
Frecuencia de muestreo	8 [KHz]
Deflexión mínima medible	17.63 [μ m]
Máxima carga aplicable	2.25 [kg]
Mínima carga medible	4.2 [gr]
Dimensiones	11.5 [cm] x 18 [cm] x 22.5 [cm]

Tabla XIII. Especificaciones del producto

TRABAJO A FUTURO

Una vez concluido este trabajo se observó que este prototipo puede ser optimizado con algunas pequeñas modificaciones e implementaciones, las cuales no se realizaron en este primer diseño, dados los recursos tan limitados con los que se contó para la realización del mismo, y debido a que con las características actuales de la máquina se cubren los requerimientos del cliente.

A continuación se citan las modificaciones que se le podrían hacer a la “Máquina para ensayos de flexión en muestras de plantas de dimensiones pequeñas”, así como el beneficio que se obtendría de las mismas.

- Modificar el mecanismo de desplazamiento, es decir cambiar la guía y corredera para evitar el desplazamiento negativo y el cabezo que existe. Podría ser sustituido con una guía de desplazamiento que ya exista en el mercado, brindando un mejor desempeño de la parte mecánica.
- Una vez cambiado el mecanismo de guía, medir la velocidad de desplazamiento directamente de un encoder rotatorio acoplado al motor para lograr así un mejor sistema de control en lazo cerrado. De esta manera podríamos obtener un mayor número de pulsos por periodo de muestreo en el contador del *timer 1*, con lo cual la velocidad podría ser menor a la actual y el controlador tendría un mejor desempeño.
- Agregar un convertidor analógico digital externo de mayor capacidad para mejorar la resolución en la medición de la carga, ya que debido a que el convertidor del microcontrolador es de solo 10 bits sólo podemos detectar carga cada 4.2 [gr].

- Diseñar un mejor filtro paso bajas que permita eliminar completamente el ruido eléctrico y así evitar posibles lecturas erróneas en el convertidor analógico digital.

Todas estas mejoras pueden resultar en un producto completo y competitivo, y no se descarta la posibilidad de que en un futuro este proyecto pueda venderse a otros laboratorios y escuelas, donde se requiera hacer pruebas mecánicas y en específico ensayos de flexión bajo estas características, ya que se ha visto que incluso universidades como la de Cambridge realiza este tipo de pruebas de manera rudimentaria, y un equipo como este podría resultarles atractivo, por sus prestaciones y bajo costo.

BIBLIOGRAFÍA

- Axelson, Jan. *USB Complete*. LakeView Research, 2005.
- BALAM, ING. JULIO ALEJANDRO LUGO JIMÉNEZ ING. GABRIEL JESÚS POOL. *Integración de PIC18F2455 de Microchip™ con LabVIEW™ 8.2 de National*. Mérid, Yucatán: Universidad Modelo, 2008.
- Beer, Ferd, y Russ Johnston. *Mécanica de Materiales*. México, D.F.: McGraw-Hill Interamerican, 2004.
- California State University. *Mechatronic Engineering*. <http://www.ecst.csuchico.edu/~rameshv/> (último acceso: 30 de noviembre de 2008).
- Colombia, Universidad Nacional de. *Laboratorio de Productos Forestales*.
<http://www.unalmed.edu.co/~lpforest/html/fichastecnicas.html>.
- Dentaurum. *Dentaurum-Productos de Ortodoncia*.
http://www.dentaurum.com.mx/productos_de_ortodoncia.htm (último acceso: 20 de abril de 2009).
- Faires, Moring Virgil. *Diseño de Elementos de Máquinas*. Limusa, 2002.
- González, Antonio Pérez. *Enciclopedia Virtual de Ingeniería Mecánica*.
http://www.emc.uji.es/d/mecapedia/modulo_de_elasticidad.htm (último acceso: 23 de octubre de 2008).
- Hibbeler, R. C. *Mecánica de Materiales*. México, D.F.: Pearson educación , 2006.
- Instron. http://www.instron.us/wa/applications/test_types/flexure/default.aspx (último acceso: 15 de diciembre de 2008).
- M. Castillo, M. C. I. Campos Cantón, Ing. J. A. Barbosa. «Universidad Autónoma de Zacates .»
<http://www.reduaz.mx/eninvie/CD2k6/CyRpds/34.pdf> (último acceso: 30 de Enero de 2009).
- Meirovitch, Leonard. *Analytical Methods in Vibrations*. New York: McMillan, 1976.
- Monigot.com. *Motores de corriente continua*.
<http://www.monigot.com/tutoriales/robotica/motores.php> (último acceso: 2 de febrero de 2009).
- Narvaez, Carlos A. «Universidad de Oriente Bolivar.» 2008.
<http://www.bolivar.udo.edu.ve/microinternet/articulos/Control%20Digital%20de%20velocidad%20de%20Un%20Motor%20DC.pdf> (último acceso: 15 de Noviembre de 2008).
- National Instruments. *Cómo realizar una medición con una Galga Extensiométrica*.
<http://zone.ni.com/devzone/cda/tut/p/id/7372> (último acceso: 20 de enero de 2009).

Nguyen, Hung T. *A first course in fuzzy and neural control*. Chapman & Hall/CRC, 2003.

Ogata, Katsuhiko. *Ingeniería de Control Moderna*. Pearson Educación, 2004.

Omega.com. *Load Cells, Force Sensors and Torque Transducers*.

http://www.omega.com/toc_asp/sectionSC.asp?section=F&book=pressure (último acceso: 25 de Enero de 2009).

Pobladores.com. *Microrobotica*.

http://www.pobladores.com/data/pobladores.com/uc/fo/ucfort/channels/microrobotica/images/2159690_estructurapic.jpg (último acceso: 15 de Enero de 2008).

Rice Lake Weighing System. *Load Cell Handbook*. Rice Lake, WI, 2002.

Universidad de Antioquia. *Ensayo de Flexión*.

http://materiales.eia.edu.co/laboratorios/flexion/teoria_del_ensayo.htm (último acceso: 10 de octubre de 2008).

University of Cambridge. *Dissemination of IT for the Promotion of Materials Science (DoITPoMS)*.

http://www.doitpoms.ac.uk/tlplib/wood/wood_stiffness.php (último acceso: 28 de noviembre de 2008).
—. *DoITPoMS Teaching and Learning Packages*.

http://www.doitpoms.ac.uk/tlplib/beam_bending/3_point_bending.php (último acceso: 20 de diciembre de 2008).

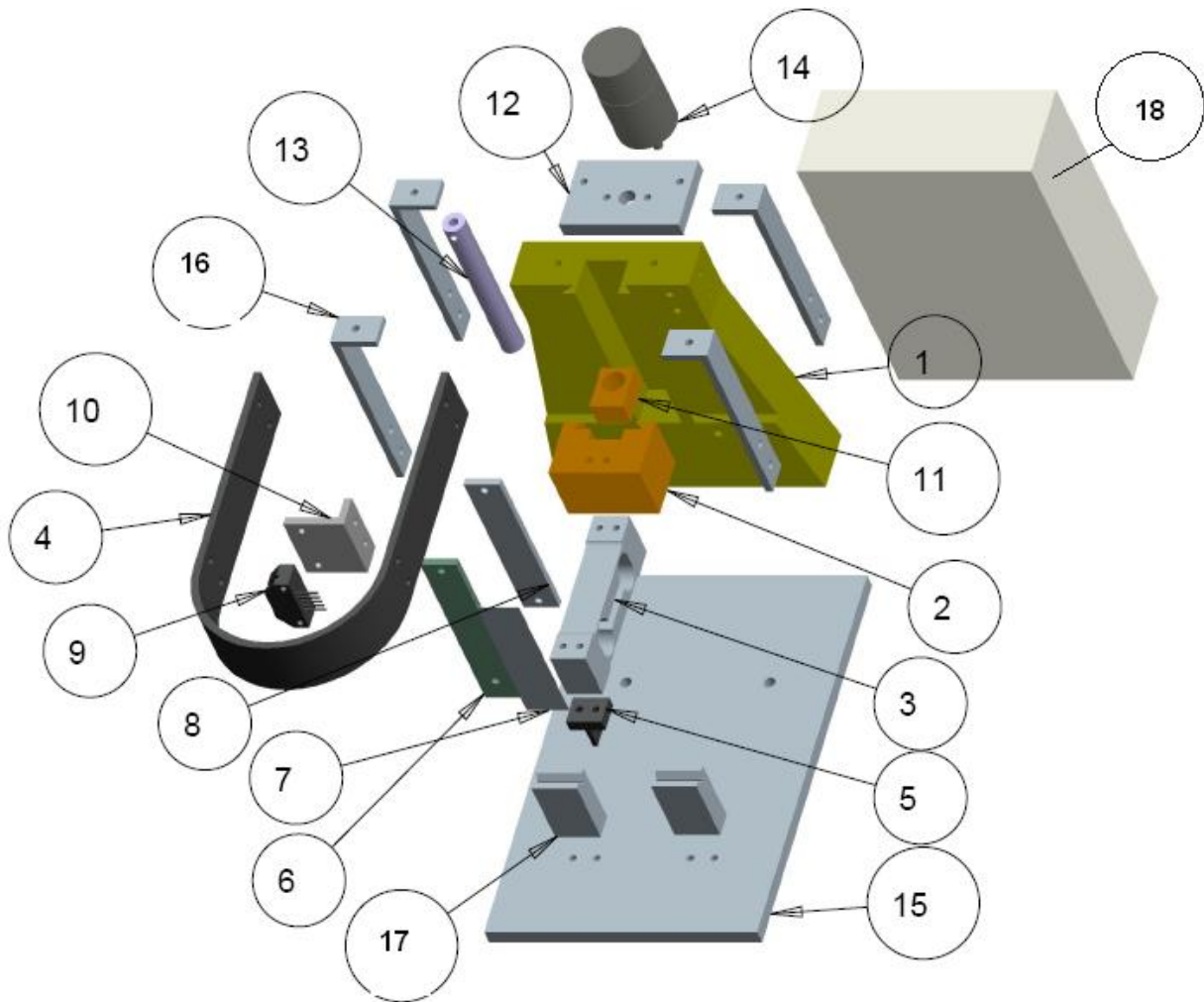
usb.org. *Universal Serial Bus*. <http://www.usb.org/developers/docs/> (último acceso: noviembre-diciembre de 2008).

US-Digital. *Data Sheet*. Vancouver, Washington, 2008.



Zwick and Roell. *Zwick Material Testing*. <http://www.zwick.co.uk/glossary.php> (último acceso: 28 de Octubre de 2008).

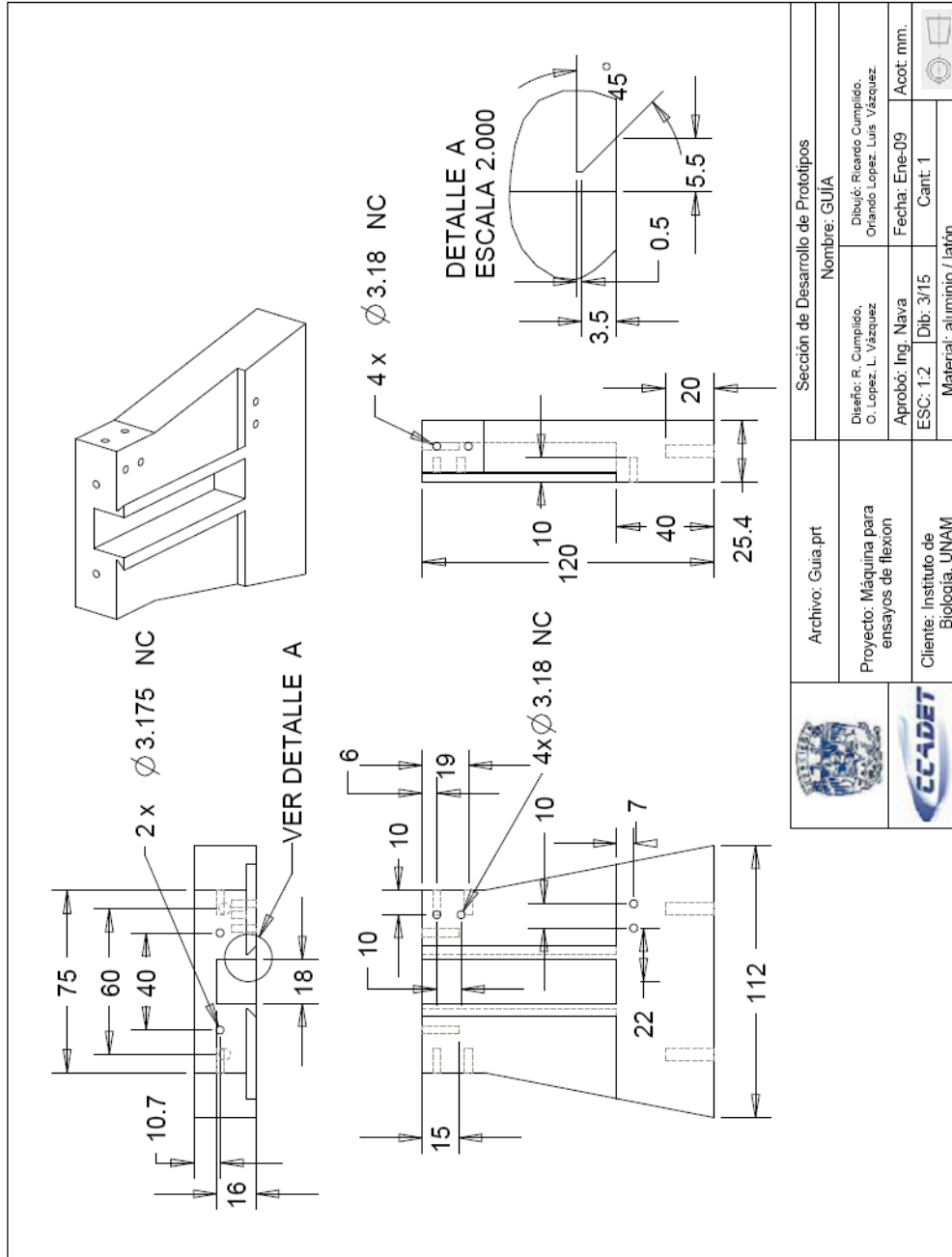
ANEXOS

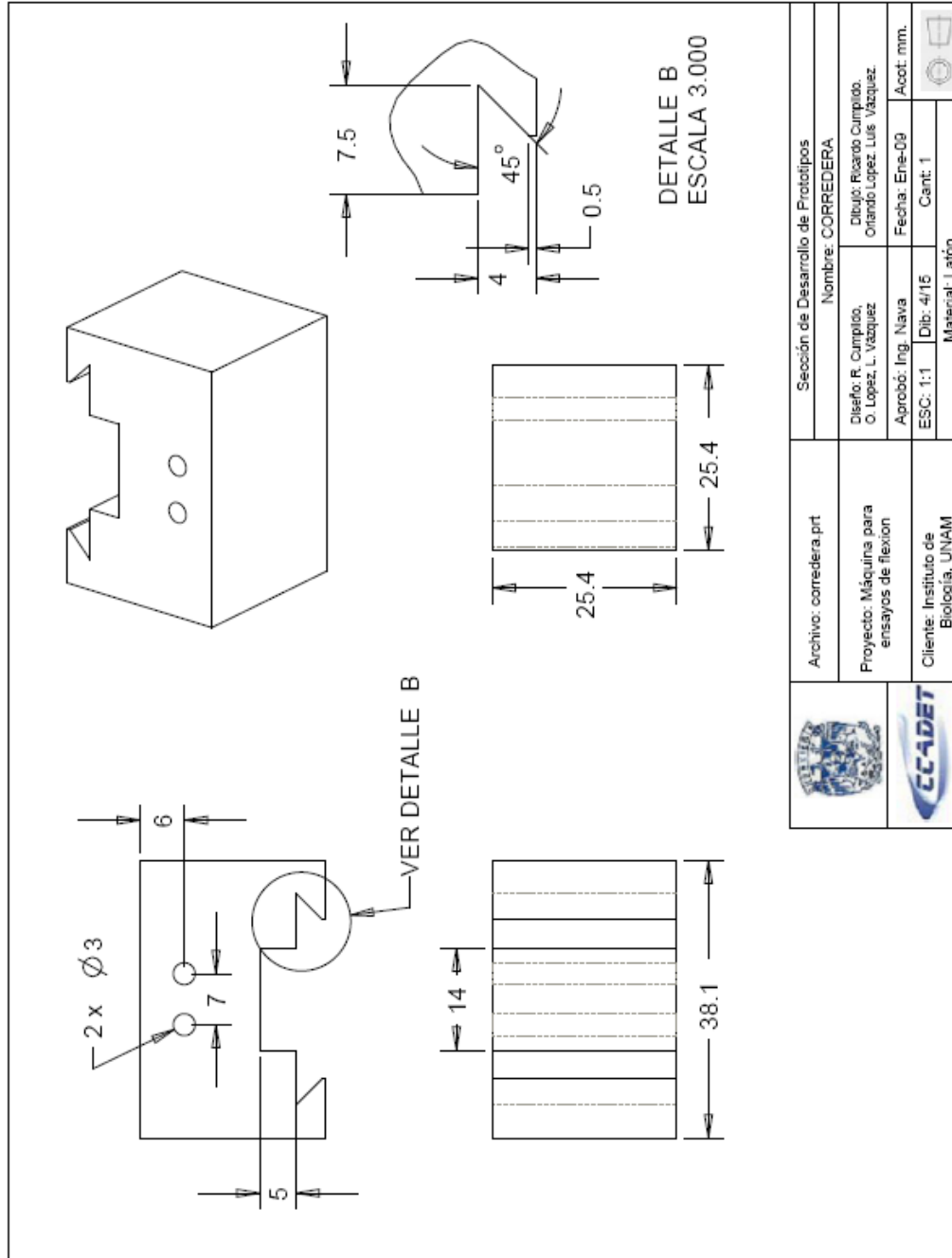
ANEXO 1. PLANOS DE FABRICACIÓN DE LAS PIEZAS MECÁNICAS





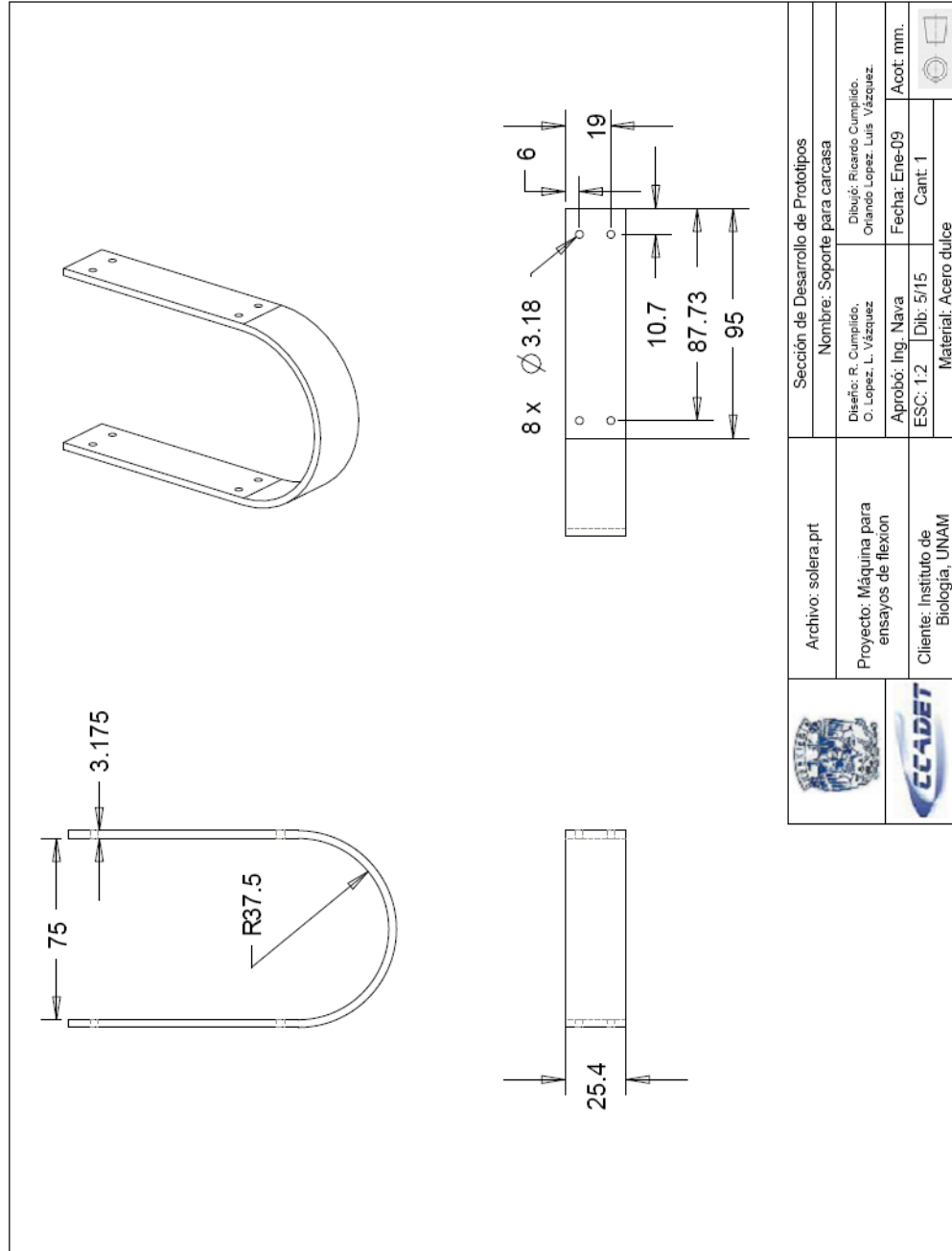
Parte	Nombre	Parte	Nombre	Parte	Nombre
1	Guía	7	Encoder	13	Tornillo sinfin
2	Corredera	8	Porta encoder (2)	14	Motor
3	Celda de carga	9	Sensor fotoeléctrico	15	Base del equipo
4	Soporte para la carcasa	10	Base para el sensor	16	Soporte para electrónica
5	Actuador	11	Tuerca	17	Soporte de probetas
6	Porta encoder (1)	12	Base para el motor	18	Fuente

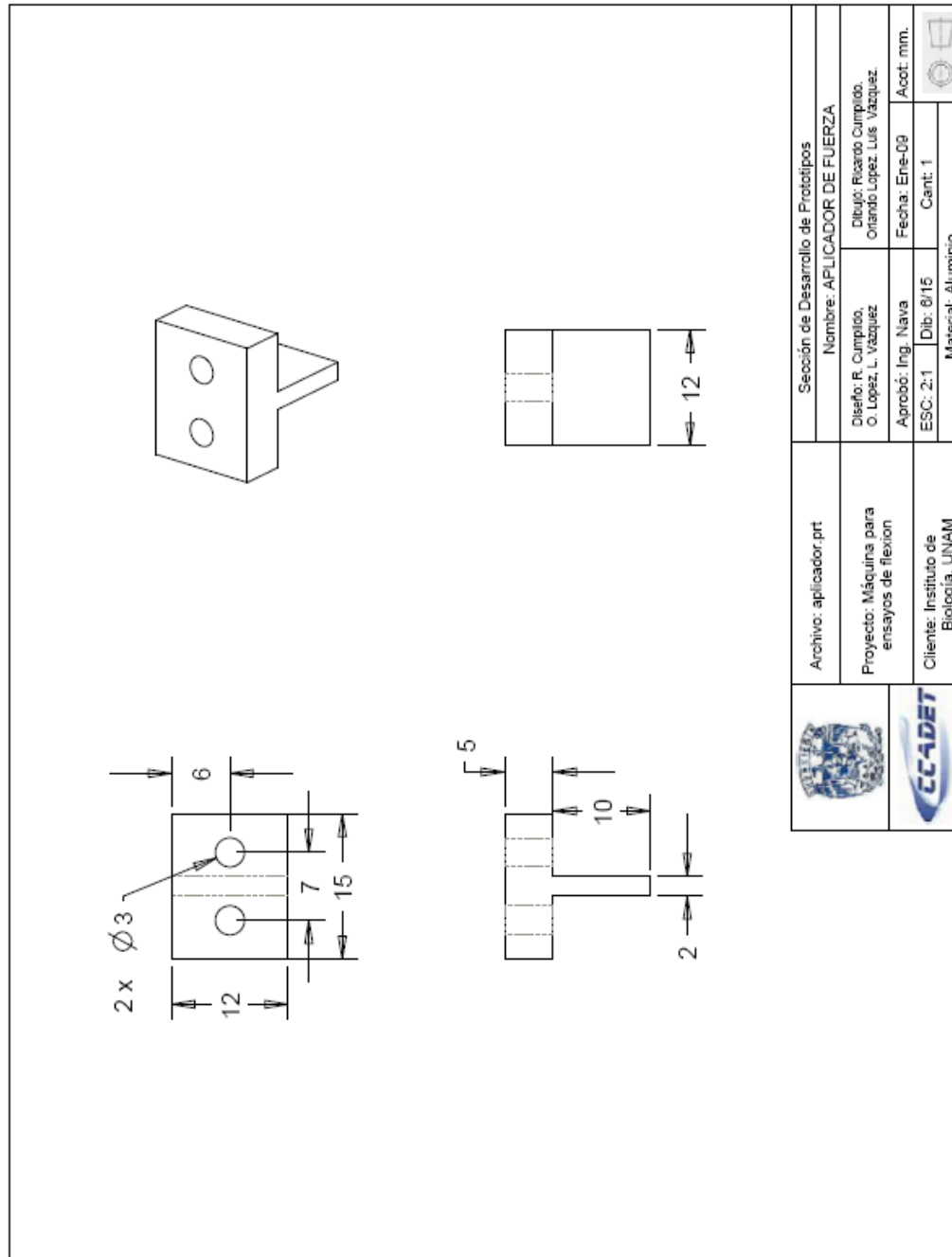
		<p>Sección de Desarrollo de Prototipos</p> <p>Nombre: ENSAMBLE</p>
<p>Archivo: Ensamble.prt</p> <p>Proyecto: Máquina para ensayos de flexion</p> <p>Cliente: Instituto de Biología, UNAM</p>	<p>Diseño: R. Cumpido, O. Lopez, L. Vázquez</p> <p>Aprobó: Ing. Nava</p> <p>ESC: 12 Dib: 2/15</p>	<p>Dibujó: Ricardo Cumpido, Orlando Lopez, Luis Vázquez.</p> <p>Fecha: Ene-09</p> <p>Acot: mm.</p> <p>Cant: --</p> <p>Material: Varios</p>

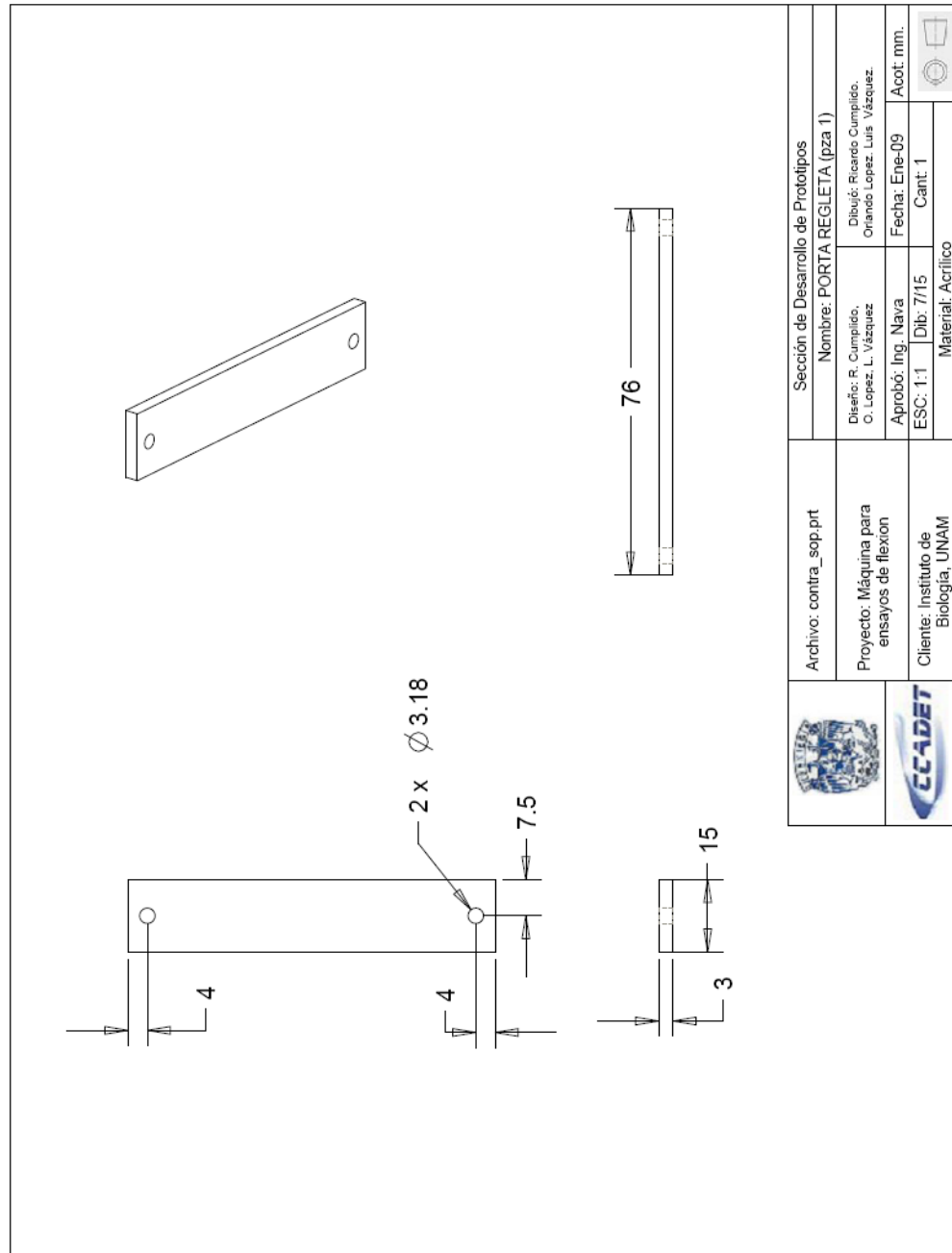


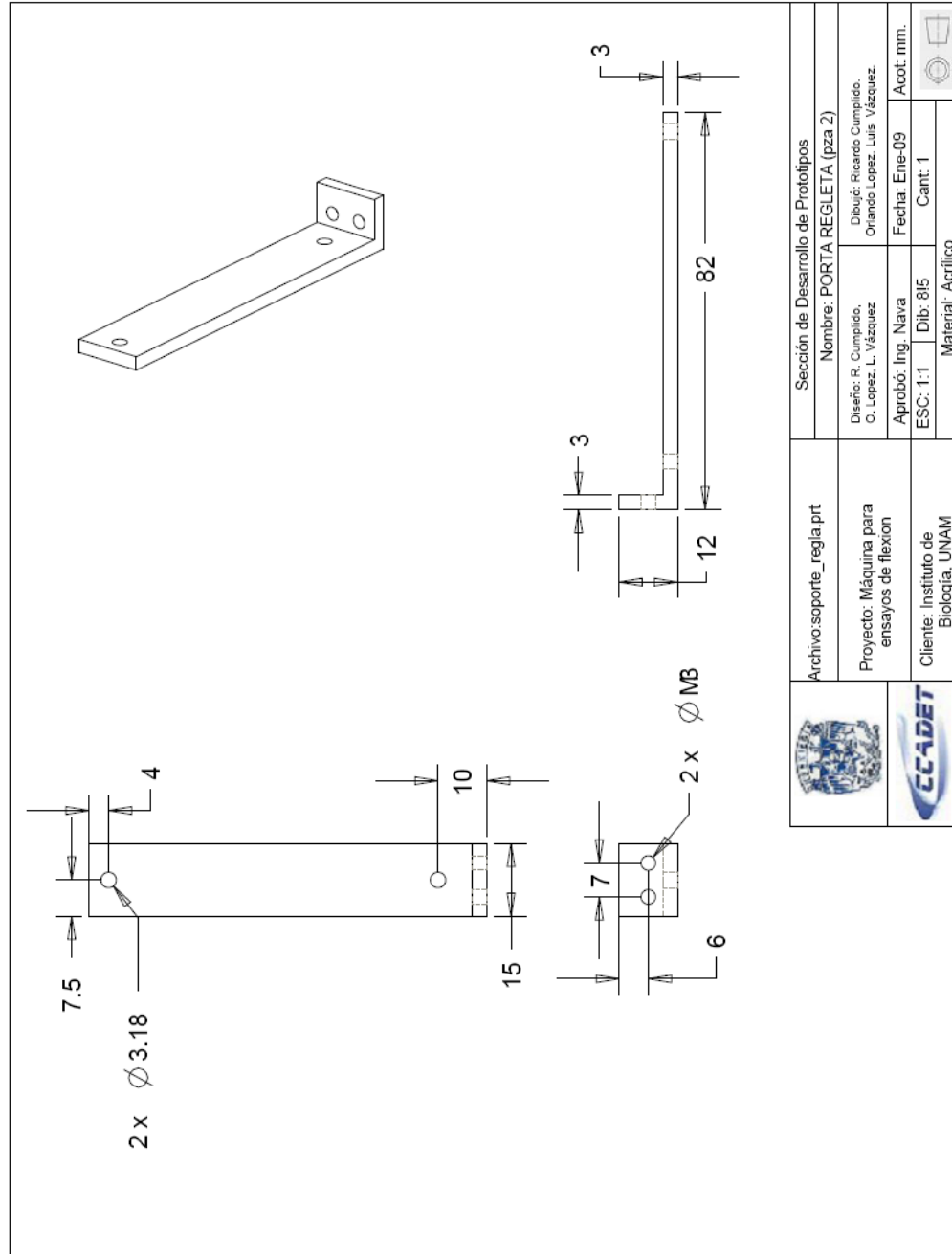


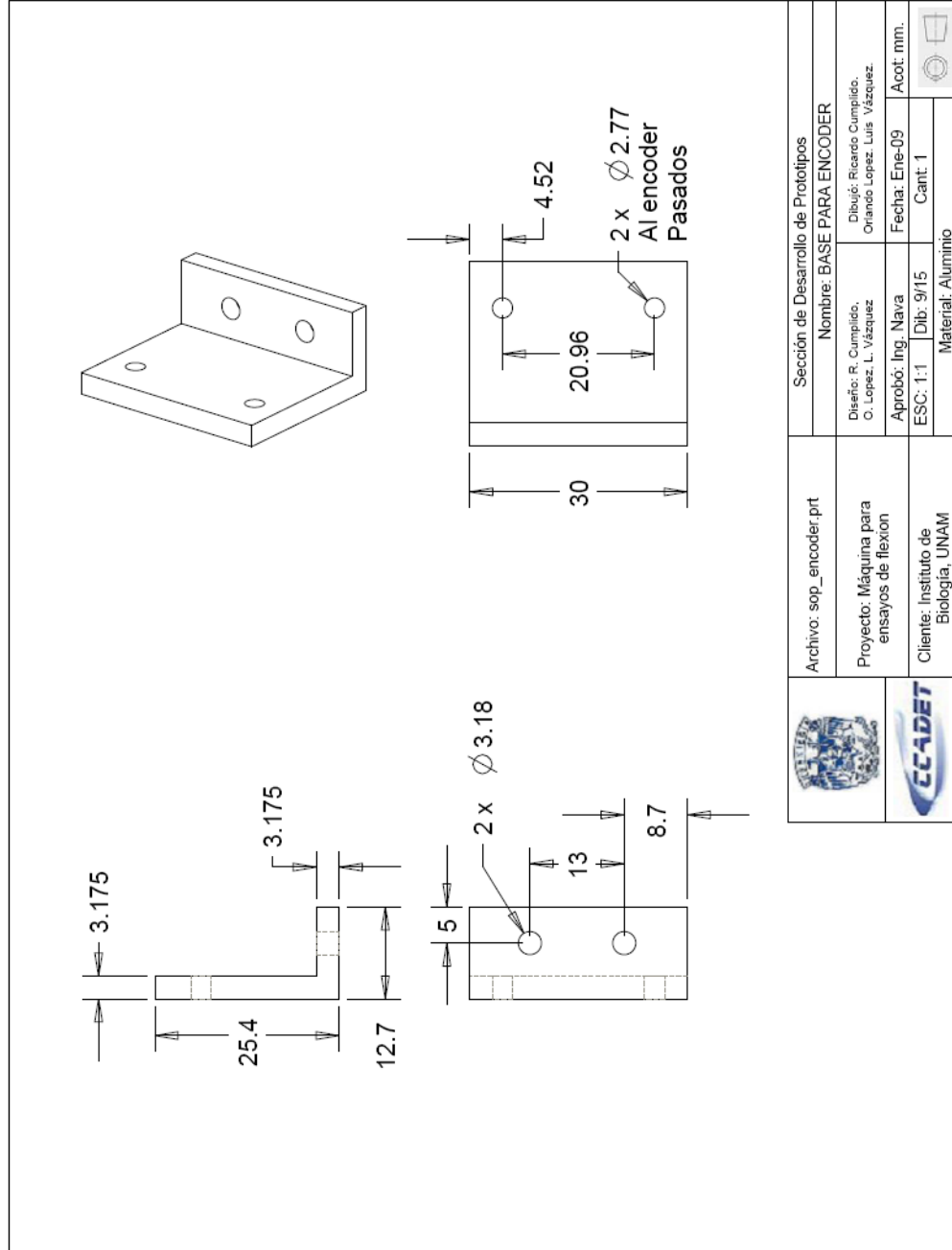
 	Archivo: corredera.prt Proyecto: Máquina para ensayos de flexion Cliente: Instituto de Biología, UNAM	Sección de Desarrollo de Prototipos Nombre: CORREDERA
	Diseñó: R. Cumpio, O. Lopez, L. Vazquez Aprobó: Ing. Nava ESC: 1:1	Dijo: Ricardo Cumpio, Oriando Lopez, Luis Vazquez. Fecha: Ene-09 Acoot. mm.






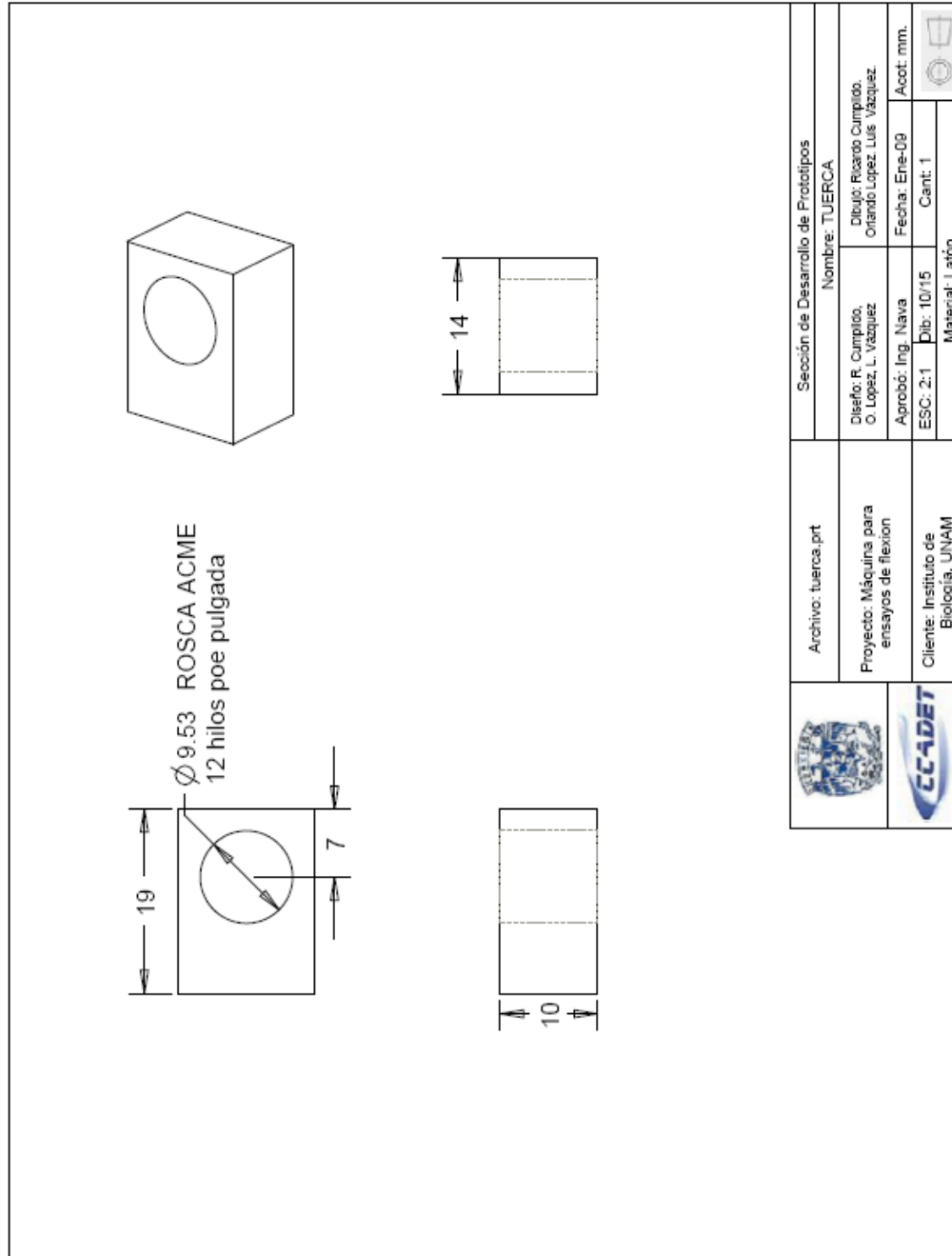


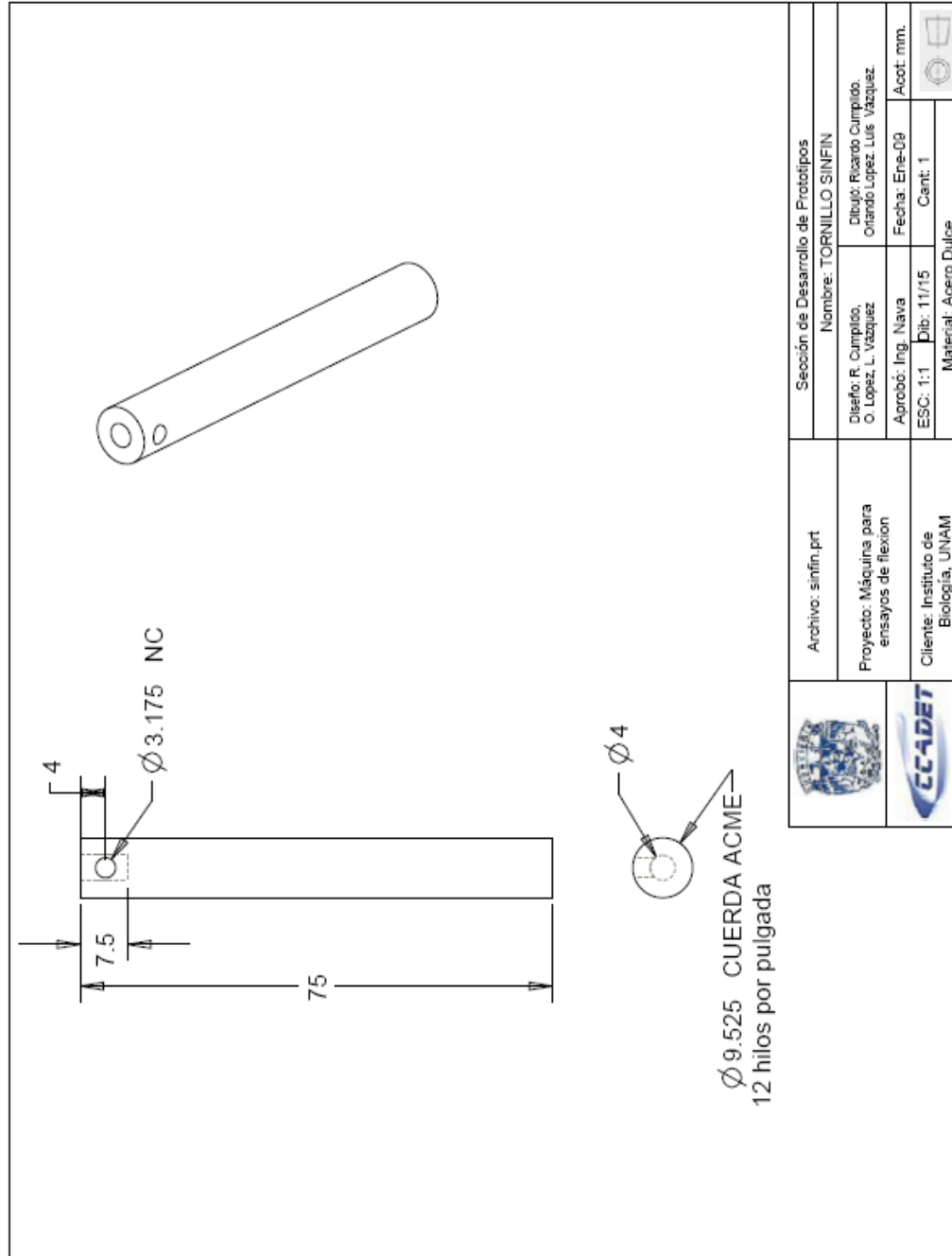


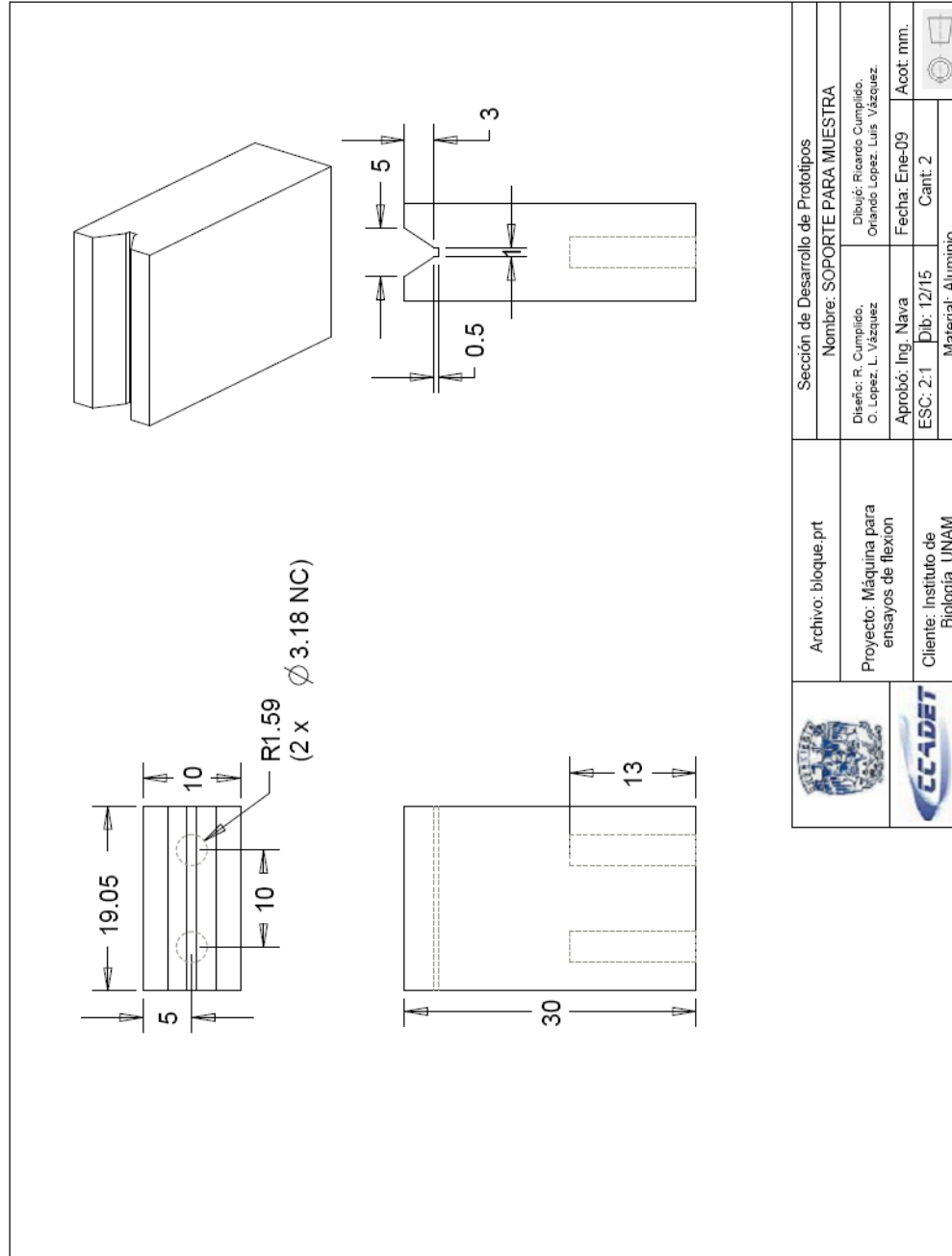







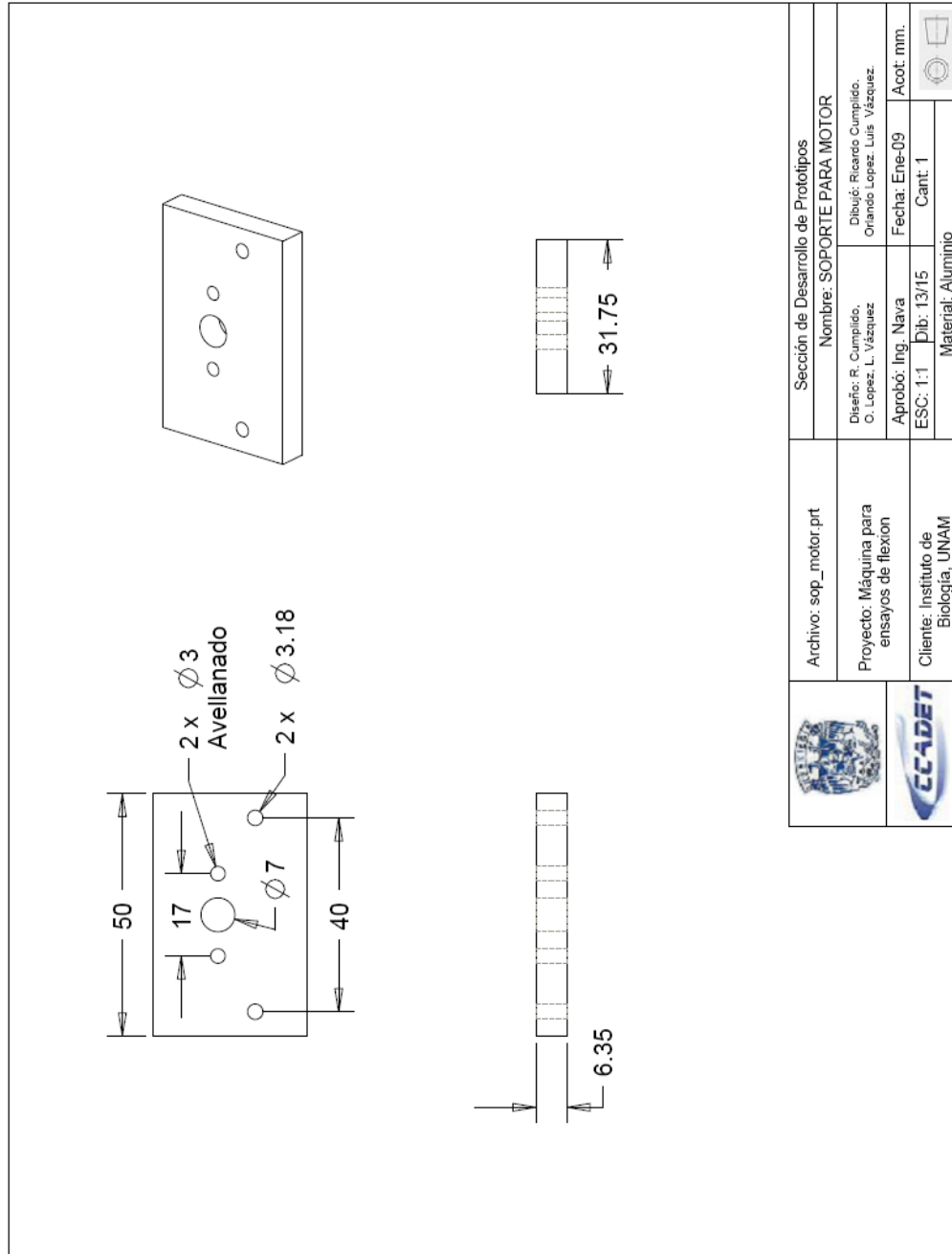
	Sección de Desarrollo de Prototipos	Nombre: BASE PARA ENCODER
	Archivo: sop_encoder.prt Proyecto: Máquina para ensayos de flexion Cliente: Instituto de Biología, UNAM	Diseño: R. Cumpido, O. Lopez, L. Vázquez Aprobó: Ing. Nava ESC: 1:1 Dib: 9/15 Material: Aluminio
	Fecha: Ene-09 Cant: 1	Acot: mm. 






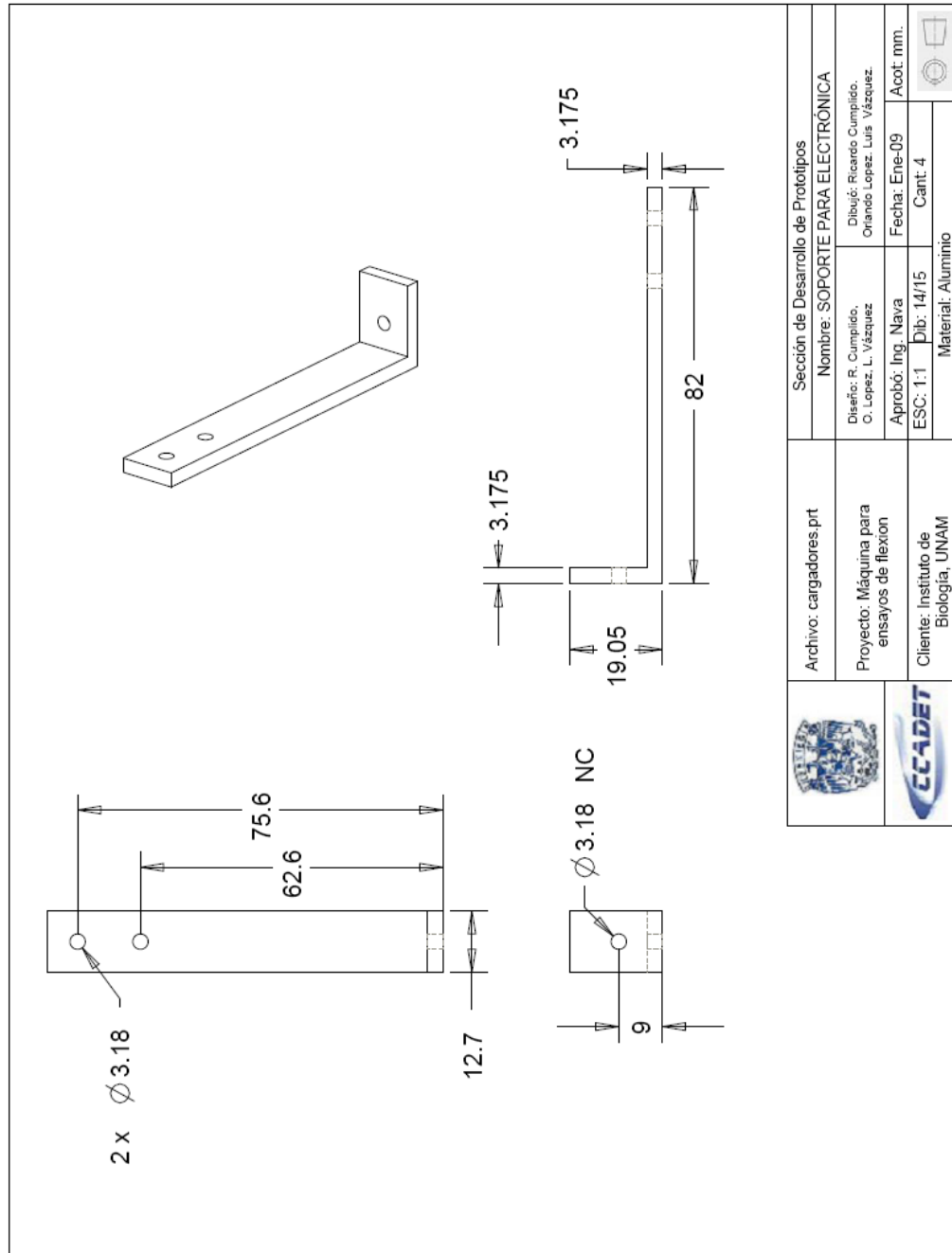


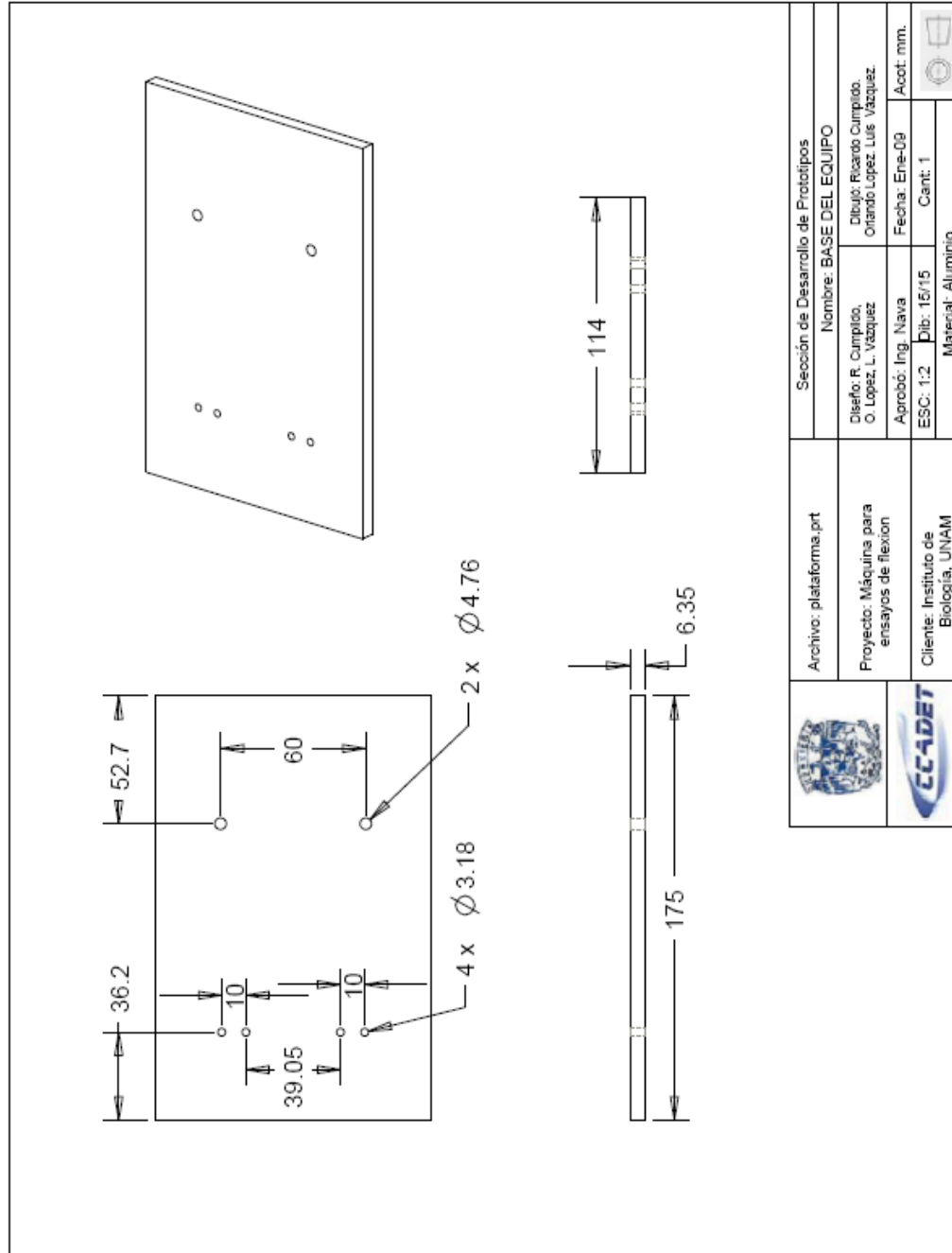


 	Archivo: bloque.prt	Sección de Desarrollo de Prototipos		
	Proyecto: Máquina para ensayos de flexión	Nombre: SOPORTE PARA MUESTRA	Diseño: R. Cumpido, O. Lopez, L. Vázquez	Dibujó: Ricardo Cumpido, Orlando Lopez, Luis Vázquez.
Cliente: Instituto de Biología, UNAM	Aprobó: Ing. Nava ESC: 2:1	Fecha: Ene-09	Dib: 12/15	Acot: mm.
		Cant: 2	Material: Aluminio	



 	Archivo: sop_motor.prt	Sección de Desarrollo de Prototipos Nombre: SOPORTE PARA MOTOR
	Proyecto: Máquina para ensayos de flexion Cliente: Instituto de Biología, UNAM	Diseño: R. Cumpido, O. Lopez, L. Vázquez Dibujo: Ricardo Cumpido, Orlando Lopez, Luis Vázquez. Aprobó: Ing. Nava Fecha: Ene-09 ESC: 1:1 Dib: 13/15 Cant: 1 Material: Aluminio
		Aoot: mm. 





ANEXO 2. HOJAS DE ESPECIFICACIONES

a) Sensor de desplazamiento lineal



HEDS

Transmissive Optical Encoder Module
Page 1 of 9



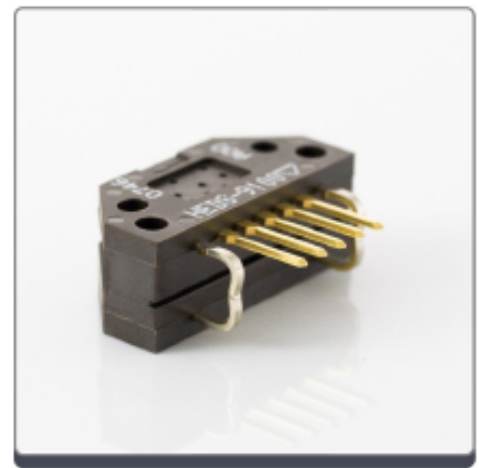
Description

HEDS is a transmissive optical encoder module. This module is designed to detect the rotary or linear position when used together with a codewheel or linear strip. The HEDS module consists of a lensed LED source and a monolithic detector IC enclosed in a small polymer package. The HEDS module uses phased array detector technology to provide superior performance and greater tolerances over traditional aperture mask type encoders.

The HEDS module provides digital quadrature outputs. It is available with a third index channel output on some resolutions. Power is supplied from a single +5Vdc source.

The resolution of the modules and encoder disks or linear strips must match. Two mounting holes are provided to accept 4-40 machine screws.

For open collector and higher voltage applications, add the PC3 cable driver, or for differential cable driver outputs, add the PC4 cable driver. Encoder disks, linear strips, quadrature decoder chips, counter chips, computer interface boards, mating connectors and cables are also available.



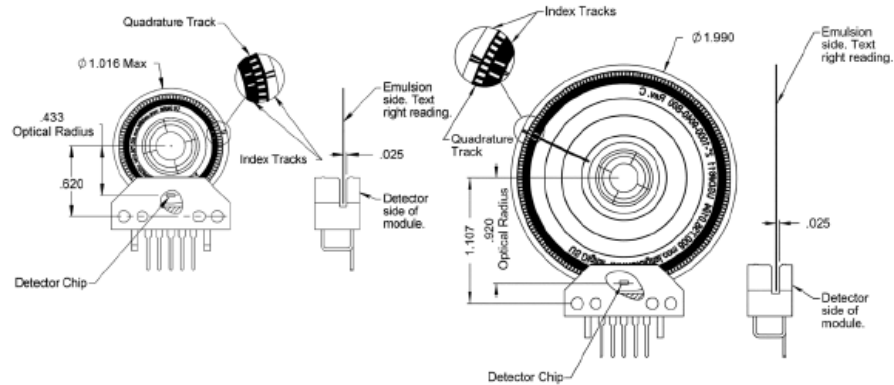
Features

- † Two channel quadrature output with optional index pulse
- † TTL Compatible
- † Single +5V supply
- † Resolutions up to 2048 CPR (8192 PPR)
- † -40C to 100C operating temperature

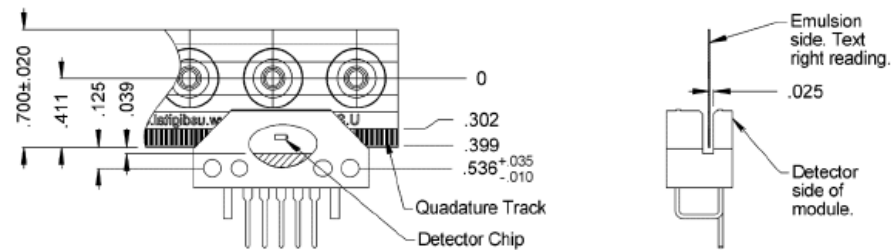
US DIGITAL **HEDS** Transmissive Optical Encoder Module Page 2 of 9



Module & Disk Mechanical Alignment



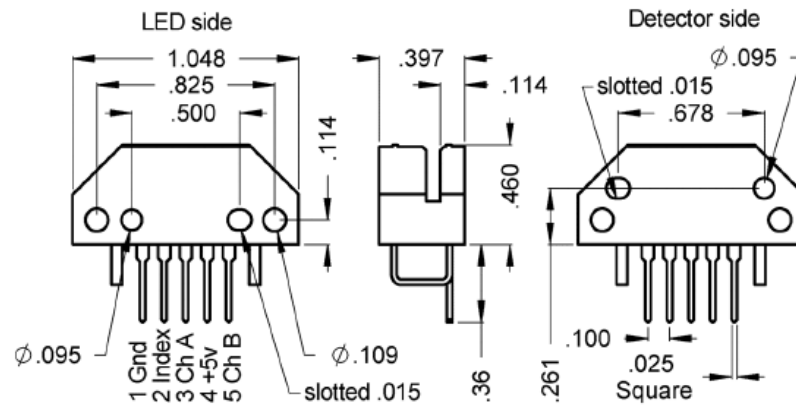
Module & Linear Strip Mechanical Alignment



US DIGITAL **HEDS** Transmissive Optical Encoder Module Page 3 of 9

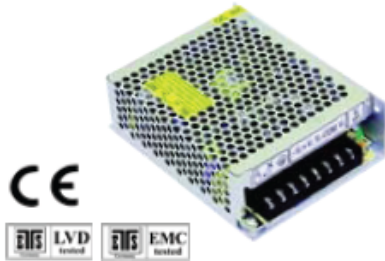


Mechanical Drawing



b) Fuente de alimentación

30W Triple Output Series - with approval

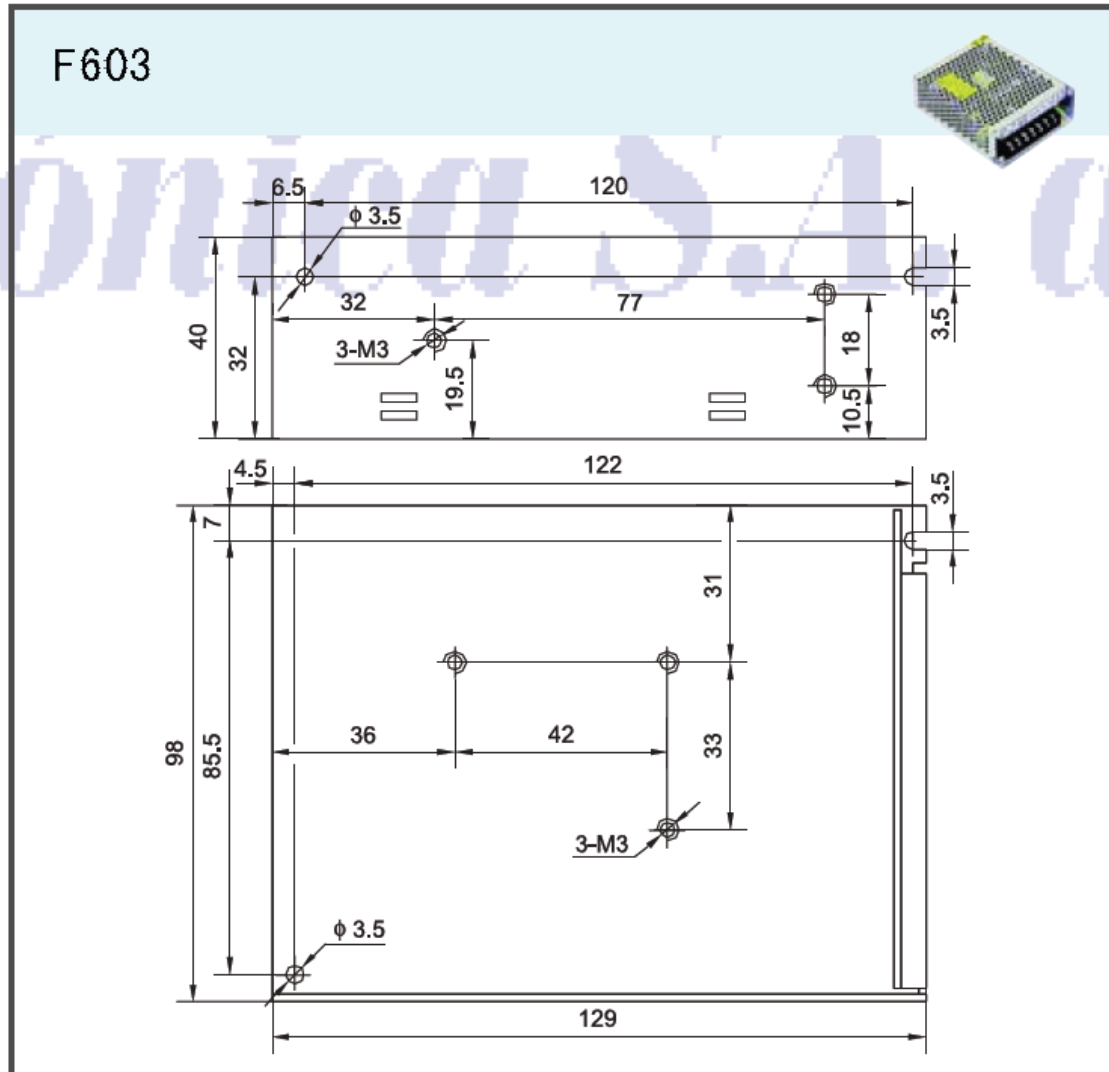


- AC input range selected by switch
- High reliability
- Approvals: ETS/LVD/EMC/ CE
- 100% full load burn-in test
- Protections: overload/ over voltage/ short circuit
- V1 is isolated from the other outputs
- 2 years warranty
- F603 129 x 98 x 40(mm)

AC input voltage range	85~132/ 170~264VAC selected by switch
Inrush current	cold start, 20A/115V, 40A/230V
Input leakage current	< 0.5mA/ 230VAC
Line regulation (full load)	≤ ± 0.5%
Load regulation	V ₁ : ≤ ± 0.5%, V ₂ ,V ₃ : ≤ ± 1% (with regulators) ≤ ± 6% (without)
Output voltage adjust range	V ₁ : ± 5% of rated output voltage
Output overload protection	110~150%
Output over voltage protection	115~150%
Withstand voltage	I/P-O/P: 3.0KVAC/1min; I/P-F/G: 1.5KVAC/1min O/P-F/G: 0.5KVAC/1min
Rise, Hold up time	50ms, 20ms@full load (typical)
Operating temp. & humidity	-10°C~+50°C, 20%~93%RH (non condensing)
Storage temp. & humidity	-20°C~+85°C, 20%~95%RH (non condensing)
Safety standards	GB4943, UL60950, EN60950
EMC standards	GB9254, EN55022 classB, EN55024 EN61000-3-2,3, EN61000-4-2,3,4,5,6,8,11
Cooling method	convection

Model	DC Output		R&N	Efficiency
HF30W-TF-A	5V	0.5~3.0A	50mV	69%
	12V	0.1~1.0A	100mV	
	-12V	0.1~0.3A	100mV	
HF30W-TF-B	5V	0.5~3.5A	50mV	70%
	15V	0.1~0.8A	120mV	
HF30W-TF-C	-15V	0.0~0.5A	120mV	68%
	5V	0.5~3.5A	50mV	
	12V	0.1~1.2A	100mV	
	-5V	0.0~1.0A	80mV	

Drawing



c) Celda de carga



Model 1006

Vishay Tedea-Huntleigh

Aluminum Single Point Load Cell



FEATURES

- Capacities 2 - 5kg
- Aluminum construction
- Single point 200 x 200mm platform
- IP66 protection
- Total error better than 0.0067% of R.O.

DESCRIPTION

Model 1006 is a very low capacity, high precision single point load cell designed for direct mounting in low capacity scales.

This load cell is suitable for applications including postal scales, counting scales, general purpose weighing scales and is also suitable for a wide variety of force measurement applications, such as industrial process control or specialist medical devices.

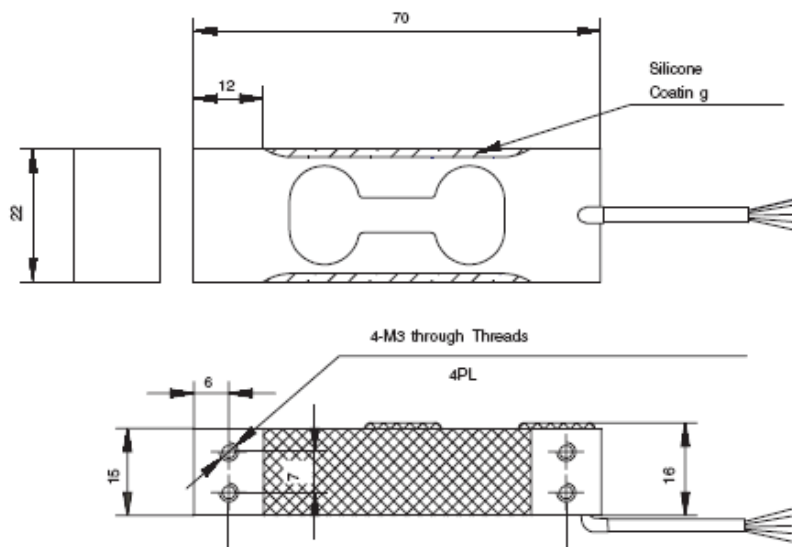
Model 1006 offers very high performance from a very small size. It is very easy to

use, and easy to apply in a wide variety of applications, where the acting center of force application is within 100mm of the load cell vertical axis.

APPLICATIONS

- Bench scales
- Counting scales
- Grocery scales

OUTLINE DIMENSIONS in millimeters



Model 1006

Vishay Tedea-Huntleigh

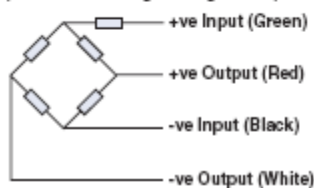
Aluminum Single Point Load Cell



SPECIFICATIONS

PARAMETER	VALUE		UNIT
	Non-Approved	G	
Accuracy class			
Maximum no. of Intervals (n)	1000	3000	
Rated capacity-R.C. (E _{max})	2, 3, 5		kg
Rated output-R.O.	2.0		mV/V
Rated output tolerance	0.2		±mV/V
Zero balance	0.2		+mV/V
Zero Return, 30 min.	0.050	0.0170	±% of applied load
Total Error	0.0300	0.0200	±% of rated output
Temperature effect on zero	0.0100	0.0040	±% of rated output/°C
Temperature effect on output	0.0030	0.0010	±% of load/°C
Eccentric loading error	0.0074	0.0057	±% of rated load/cm
Temp. range, compensated	-10 to +40		°C
Temp. range, safe	-20 to +70		°C
Maximum safe central overload	150		% of R.C.
Ultimate central overload	300		% of R.C.
Excitation, recommended	10		Vdc or Vac rms
Excitation, maximum	15		Vdc or Vac rms
Input impedance	415±20		Ohms
Output impedance	350±8		Ohms
Insulation resistance	>2000		Mega-Ohms
Cable length	0.4		m
Cable type	4 wire, PVC, single floating screen		Standard
Construction	Aluminum		
Environmental protection	IP66		
Platform size (max)	200 x 200		mm
Recommended torque	2 & 3kg - 4.0 5kg - 6.0		N*m

Wiring Schematic Diagram
(Unbalanced bridge configuration)



ANEXO 3. FIRMWARE PARA LOS MICROCONTROLADORES

a) Microcontrolador 1: Adquisición de datos y comunicación con la PC.

Este microcontrolador es el encargado de adquirir la señal proveniente del amplificador operacional, convierte la misma en datos digitales y los envía a la PC. Así mismo, recibe las órdenes de la PC para subir, bajar o detener el actuador e iniciar o detener la prueba.

Finalmente, mantiene una comunicación con el microcontrolador 2, indicándole el movimiento o detención del actuador y, en su caso, la dirección del mismo.

```
#include <18F2550.h>
#device ADC=10
#fuses HSPLL,NOWDT,NOPROTECT,NOLVP,NODEBUG,USBDIV,PLL5,CPUDIV1,VREGEN
#define LOADER_END 0x7FF
#define LOADER_SIZE 0x6FF
#build(reset=LOADER_END+1, interrupt=LOADER_END+9)
#org 0, LOADER_END {} // nothing will replace the bootloader memory space
//END OF bootloader definition

#use delay(clock=4800000)
#use rs232(baud=115200,xmit=PIN_C6,rcv=PIN_C7)

/**** USB *****/
#define USB_HID_DEVICE FALSE //deshabilitamos el uso de las directivas HID
#define USB_EP1_TX_ENABLE USB_ENABLE_BULK //turn on EP1(EndPoint1) for IN bulk/interrupt
transfers
#define USB_EP1_RX_ENABLE USB_ENABLE_BULK //turn on EP1(EndPoint1) for OUT bulk/interrupt
transfers
#define USB_EP1_TX_SIZE 64 //size to allocate for the tx endpoint 1 buffer
#define USB_EP1_RX_SIZE 3 //size to allocate for the rx endpoint 1 buffer

#include <pic18_usb.h> //Microchip PIC18Fxx5x Hardware layer for CCS's PIC USB driver
#include <Descriptor.h> //Configuración del USB y los descriptores para este dispositivo
#include <usb.c> //handles usb setup tokens and get descriptor reports
#include <string.h>

/****Definicion de los datos recibidos por USB*****/
#define abajo recepcion[0]
#define arriba recepcion[1]
#define inicio recepcion[2]
#define prueba datos[31] //Indica a la PC el estado de la prueba.
#zero_ram
```

```

/***** Definiciones RAM para USB *****/
int recepcion [3];
long datos[32]; // Arreglo donde se almacenaran los valores de la celda y de la posicion para enviarse
int i=0;
int envbytes=15; // Numero de lecturas a enviar
int1 ready=0;
long datosant=0;
int1 flag=0;
long lectura; // Lectura de la celda de carga
int1 carga; // Indica que la celda ha comenzado a hacer lecturas
int bajada=0; // Bandera que indica que el aplicador puede seguir bajando
int subida=0; //Bandera que indica que el aplicador puede seguir subiendo
/*****
 I N I C I O (CONFIGURACIÓN DEL MICROCONTROLADOR)
 *****/
void configuracion(void)
{
 /***** PUERTOS *****/
 set_tris_A(0b111111);
 set_tris_c(0b00000001);
 set_tris_b(0xFF);
 set_tris_d(0);

 /***** TIMER_1: CONTADOR DE EVENTOS EXTERNOS *****/
 setup_timer_1(T1_EXTERNAL|T1_DIV_BY_1); // Contador eventos Externos (Pulsos de la GAL)
 // Entrada: PIN_C0

 /***** Configuracion del ADC *****/
 setup_adc(ADC_CLOCK_INTERNAL); // Enables the a/d module and sets the clock to internal adc clock
 setup_adc_ports(AN0); // Configura solo el pin A0 como analogico
 set_adc_channel(0); // Indica que se utilizara el canal 0 del convertidor analogico

 /***** Interupciones externas *****/
 ext_int_edge(1,H_TO_L); //Configura el flanco de interrupcion de alto a bajo
 ext_int_edge(2,H_TO_L);

 /***** INTERRUPTOS *****/
 enable_interrupts(GLOBAL); // Habilita Interrupciones globales
 enable_interrupts(INT_EXT1); // Interrupcion externa 0 (PIN B1)
 enable_interrupts(INT_EXT2); // Interrupcion externa 2 (PIN B2)
 enable_interrupts(INT_AD); // Interrupcion ADC

 usb_init(); // Inicializamos el USB
 usb_task(); // Habilita periferico usb e interrupciones
 usb_wait_for_enumeration(); //Esperamos hasta que el PicUSB sea configurado por el host
}
/*****Funcion para subir el actuador*****/
void subir(void)
{
 output_high(pin_d7);

```

```
output_low(pin_c1);
}
/****Funcion para bajar el actuador*****/
void bajar(void)
{
output_low(pin_d7);
output_high(pin_c1);
}
/*****Funcion para detener el actuador*****/
void detener(void)
{
output_low(pin_d7);
output_low(pin_c1);
}
/**Función donde se atiende la interrupción del ADC*****/
#INT_AD
void AD_isr()
{
delay_us(10);
lectura=read_adc(ADC_READ_ONLY); // Almacena la lectura del ADC en el arreglo
if(lectura<1) set_timer1(0); // Inicia lectura de pulsos a cero
if(lectura>3) // Pregunta si la celda ya ha comenzado a detectar carga
{
carga=1; // Indica que se empezaran a almacenar los valores leídos
datos[i]=lectura;
datos[envbytes+i]=get_timer1();
}
if(lectura>20 && carga==1) flag=1; //Indica que se ha pasado el valor de 10 en la lectura del ADC
//por lo que la fuerza va en aumento

if(datos[i]>532) // Entra cuando se ha aplicado una fuerza mayor de 2.25 kg en el aplicador
{ // y detiene la prueba.
datos[i]=0;
prueba=0; // Indica que se ha aplicado una fuerza mayor a 4 kg
ready=0;
detener();
usb_put_packet(1,datos,64,USB_DTS_TOGGLE);
}
if(carga==1)
{
i++;
datosant=datos[i]; //Almacena el valor de la lectura actual
}
if(i==envbytes && ready==1) //Entra cuando se han tomado 15 lecturas de la celda de carga y el encoder
{ //y envia los datos a la PC.
i=0;
prueba=1;
usb_put_packet(1,datos,64,USB_DTS_TOGGLE);
}
}
```

```
if(ready==1) read_adc(ADC_START_ONLY); // Inicia una nueva conversión del ADC si la bandera ready
// esta en "1"
}
/*****
Interrupción externa1 para detectar el tope físico superior del aplicador
*****/
#INT_EXT1
void isr_ext1()
{
    subida=0;
    detener();
}
/*****
Interrupción externa2 para detectar el tope físico inferior del aplicador
*****/
#INT_EXT2
void isr_ext2()
{
    prueba=2;
    ready=0;
    bajada=0;
    usb_put_packet(1,datos,64,USB_DTS_TOGGLE);
    detener();
}
/*****
* Programa Principal
*****/
void main()
{
    int j=0;
    configuracion(); // Configuramos el microcontrolador
    portd=0;
    while(TRUE)
    {
        if(usb_enumerated()) // Entra si el dispositivo esta enumerado
        {
            if (input(pin_b1)) subida=1; //Lee el estado inicial del aplicador para verificar si esta en un tope
            fisico
            if (input(pin_b2)) bajada=1;
            if(usb_kbhit(1)) //Entra si se ha recibido un dato por el USB
            {
                usb_get_packet(1,recepcion,3); //Recibe los datos enviados por la PC
            }
            if(inicio==1 && ready==0) //Entra cuando la PC indica que se inicie una prueba
            {
                ready=1; //Enciende bandera para indicar que una prueba se esta realizando
                prueba=1; //Indica a LabVIEW que la prueba esta en progreso
                read_adc(ADC_START_ONLY); //Inicia conversion del ADC
            }
        }
    }
}
```



```
    output_high(pin_d7);
    output_high(pin_c1);                //Llama a funcion para bajar el aplicador
  }
  if(arriba==1 && inicio==0 && abajo==0 && subida==1 ) //Entra cuando la PC indica que el aplicador se
mueva hacia arriba
  {
    subir();
  }
  if(arriba==0 && abajo==0 && inicio==0) //Entra cuando la PC indica que no debe realizarse ninguna
accion
  {
    //por lo que el aplicador permanece detenido
    if(ready==1) ready=0;
    detener();
    for(j=0;j<32;j++) //Pone a 0 el arreglo de datos.
    {
      datos[j]=0;
    }
    i=0;
  }
  if(abajo==1 && inicio==0 && arriba==0 && bajada==1) //Entra cuando la PC indica que el aplicador se
mueva hacia abajo
  {
    bajar();
  }
}
}
```

b) Microcontrolador 2: Control de velocidad del aplicador.

Este microcontrolador se encarga de enviarle las señales al *punte H*, necesarias para mover el actuador. Se comunica con el microcontrolador 1 a través de 2 pines, con los cuales se conoce la acción a realizar: subir, bajar o detener el actuador.

Si el actuador se mueve hacia abajo, el *timer* 3 genera una interrupción cada 43.69 [ms]. En esta interrupción se realizan los cálculos del lazo PID.

```
#include <18F2550.h>
#device ADC=10
#fuses HSPLL,NOWDT,NOPROTECT,NOLVP,NODEBUG,USBDIV,PLL5,CPUDIV1,VREGEN
#define LOADER_END 0x7FF
#define LOADER_SIZE 0x6FF
#build(reset=LOADER_END+1, interrupt=LOADER_END+9)
#org 0, LOADER_END {} // nothing will replace the bootloader memory space
```

```

//END OF bootloader definition

#use delay(clock=4800000)
#define MAX_PULSOS 10 // Máximo pulsos por periodo de muestreo
#define MAX_PWM 1010 // Maximo PWM (100%) del controlador
#define MIN_PWM 5 // Minimo PWM del controlador
#define SPEED_UP 100 // Velocidad de subida
#define SPEED_DOWN 200 // Velocidad de bajada
#define SPEED_TEST 101 // Velocidad de prueba
#define Input_Scale 1024
#define I1 input(PIN_A2) // Pines de entrada y salida
#define I2 input(PIN_A3)
#define O1 PIN_C1
#define O2 PIN_D7
##zero_ram

/***** Valor de constantes P I D (Determinadas de manera experimental) *****/
unsigned int16 set_point;
unsigned int16 P_GAIN=494;
unsigned int16 I_GAIN=2;
unsigned int16 D_GAIN=197;
/***** Variables utilizadas en el lazo PID *****/
signed int16 P_Term, I_Term, D_Term;
unsigned int16 delta_pulsos;
signed int16 pwmSet;
signed int16 vel_pwm,error, ePrev, iPrev;
unsigned int16 pwm_PID;
unsigned int16 t2,n=2500;
int1 led;
/*****
CONFIGURACIÓN DEL MICROCONTROLADOR
*****/
void configuracion(void)
{
    /***** PUERTOS *****/
    set_tris_a(0b001100);
    set_tris_b(0b00000000);
    set_tris_c(0b00000001);
    set_tris_d(0b00000000);
    /***** TIMER_1: CONTADOR DE EVENTOS EXTERNOS *****/
    setup_timer_1(T1_EXTERNAL|T1_DIV_BY_1); // Contador eventos Externos (Pulsos de la GAL)
                                           // Entrada: PIN_C0

    /***** PWM *****/
    Setup_ccp1(CCP_PWM); // Configura módulo CCP1 como PWM
                        // Salida: PIN_C2

    /***** TIMER 2 *****/
    setup_timer_2(T2_DIV_BY_16, 149, 16); // PWM 5.0 KHz
                                           // (4*16*150)/48E6=200 microseg (PWM 5.0 KHz )
                                           // Interrupción del timer_2 cada 400 microseg

    /***** TIMER 3 *****/
    Setup_timer_3(T3_INTERNAL|T3_DIV_BY_8); // (4*8*2^16)/48E6=46.69 miliseg

```

```

// Timer_3: Contador de 16 bits.

enable_interrupts(GLOBAL); // Habilita Interrupciones globales
disable_interrupts(INT_TIMER3); // Deshabilita interrupción del timer 3
}
/****Función para subir el actuador*****/
void subir(void)
{
    disable_interrupts(INT_TIMER3);
    set_pwm1_duty(SPEED_UP);
    output_high(O1);
    output_low(O2);
}
/****Funcion para bajar el actuador*****/
void bajar(void)
{
    enable_interrupts(INT_TIMER3);
    output_low(O1);
    output_high(O2);
}
/****Función para detener el actuador*****/
void detener(void)
{
    disable_interrupts(INT_TIMER3);
    set_pwm1_duty(0);
    output_low(O1);
    output_low(O2);
}
/*****
Interrupción del timer 3: Lazo PID. Periodo de muestreo : 43.69 miliseg
*****/
#int_timer3
void lazo_PID(void)
{
    delta_pulsos=get_timer1(); // Velocidad del actuador (pulsos por periodo de muestreo )
    set_timer1(0); // Ponemos a cero el timer1
    vel_pwm= ((delta_pulsos*(MAX_PWM))/MAX_PULSOS);
    error = (set_point - vel_pwm); // Escale delta_pulsos a valores PWM (0-1023)
    // Para usar la misma escala y calcular el Error
/***** Ecuación Estándar PID *****/
    P_Term = (signed int32)P_GAIN * error/input_scale;
    I_Term = (signed int32)I_GAIN * error/Input_Scale;
    I_Term = I_Term + iPrev;
    D_Term = (signed int32)(D_GAIN * error - D_GAIN * ePrev)/Input_Scale;
    pwmSet = pwmSet + (P_Term + I_Term + D_Term) ; //Acción de control escala 0-1023

    ePrev = error; //Guarde el error actual y el término integral actual para ser
    iPrev = I_Term; //usado como valores previos en el próximo periodo de muestreo

    if(pwmSet >= MAX_PWM) pwmSet = MAX_PWM; // Limite la salida al Máximo o mínimo del Controlador

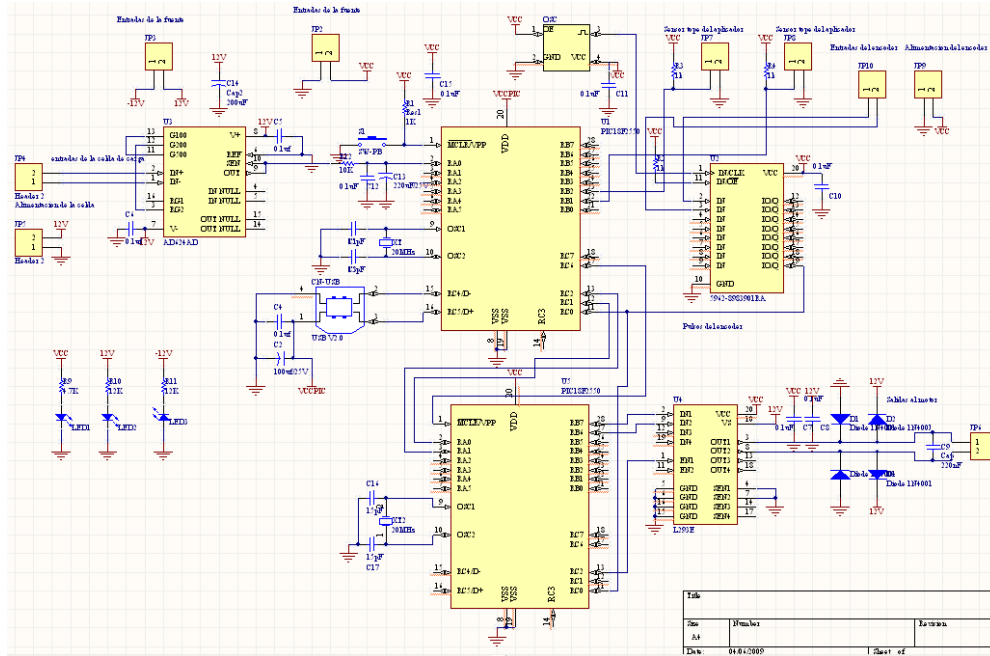
```

```
else if(pwmset <= MIN_PWM) pwmSet = MIN_PWM;

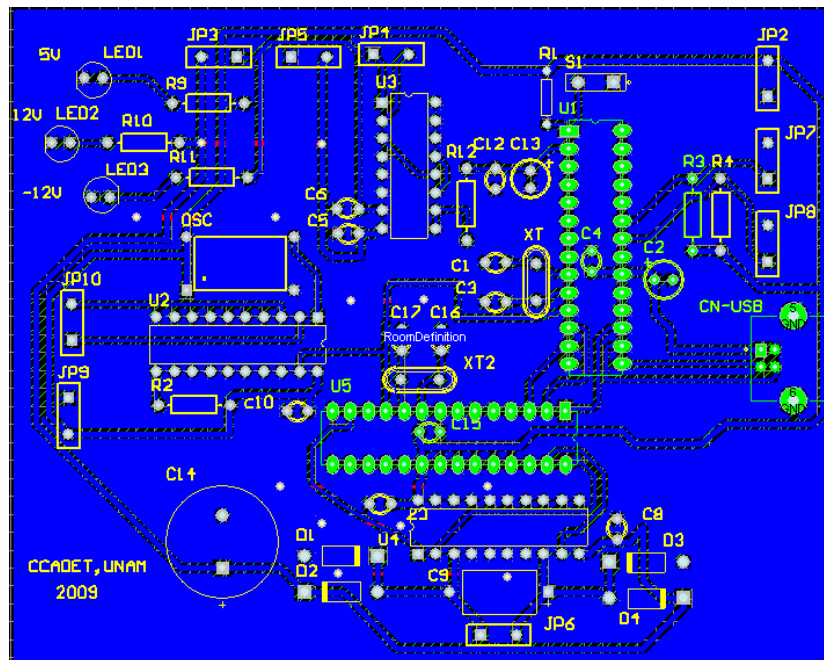
pwm_PID = (int16) pwmSet;    // Salida del controlador
set_pwm1_duty(pwm_PID);    // Envía PWM calculado al puente H
}
/*****
* Programa Principal
*****/
void main()
{
    configuracion();        //Configuramos el microcontrolador
    detener();
    while(TRUE)
    {
        if(I1==0 && I2==0)
            { detener();}
        else if(I1==0 && I2==1)
            { subir();}
        else if(I1==1 && I2==0)
            {
                set_point=SPEED_DOWN;
                bajar();
            }
        else if(I1==1 && I2==1)
            {
                set_point=SPEED_TEST;
                bajar();
            }
    }
}
```

ANEXO 4. DIAGRAMAS ELECTRÓNICOS

a) Diagrama esquemático



b) PCB



ANEXO 5. SOFTWARE

