
Universidad Nacional De Ingeniería
Facultad De Ciencias
Escuela Profesional De Física

**“Desarrollo de un Sistema Automatizado de
Detección de Modos en Guías Ópticas”**

Tesis

**Para Optar el Título Profesional de
Licenciado en Física**

Guido J. Castillo Ocaña

2005

Lima - Perú

Índice

Resumen	1
Introducción	2
Capítulo 1: Fundamento teórico	
1.1 Modos de guía de ondas	3
Capítulo 2: Parte Experimental	
2.1 Descripción General	5
2.2 Descripción detallada de las partes	7
2.2.1 El Motor a pasos	7
2.2.1.1 Principio de funcionamiento	7
2.2.1.2 Tipos de motores a pasos de imán permanente	8
2.2.2 El puerto paralelo de la PC	12
2.2.2.1 El hardware del puerto paralelo	12
2.2.2.2 Direcciones de los Puertos Paralelos	13
2.2.2.3 Registros del Puerto Paralelo Estándar	14
2.2.2.4 Interfaz para el puerto paralelo	15
2.2.2.5 Programación del puerto paralelo	17
2.2.3 El Sensor ..	19
2.2.4 El multímetro digital	20
2.2.4.1 El display de 4½ dígitos	20
2.2.4.2 Precisión	21
2.2.4.3 Formato de datos	21
2.2.4.4 Interfase de conexión con la PC	21
2.2.4.5 Comandos del multímetro	22
2.2.4.6 Especificaciones de comunicación para la transmisión de datos	22

2.2.5 El puerto serial (RS232)	23
2.2.5.1 Características del hardware	24
2.2.5.2 Velocidad de transmisión	25
2.2.5.3 El UART	26
2.2.5.4 Direcciones e Interrupciones del puerto serial	26
2.2.5.5 Registros del UART	26
2.2.5.6 Programación del Puerto Serial	33
2.2.6 Montaje de los componente sobre una plataforma	39
Capitulo 3: Resultados Obtenidos	
3.1 Efecto de Guías Ópticas planas en películas semiconductoras de Sulfuro de Cadmio CdS	41
3.2 Determinación experimental de índices de refracción en películas semiconductoras: CdS, Ti3O5	43
Capitulo 4: Conclusiones y Trabajo Futuro	
4.1 Conclusiones	44
4.2 Trabajo Futuro	44
Referencias	45
Apéndice A: Programa principal	46
Apéndice B: Diseño para circuito impreso	53
Apéndice C: Hojas de especificaciones de los integrados usados	54
Apéndice D: Programa para determinar el número de modos	56
Apéndice E: Programa para calcular el índice y el espesor	61

Resumen

El objetivo del presente trabajo es el desarrollo de un sistema automático de control y adquisición de datos, en el equipo de detección de modos de propagación en guías ópticas planas.

El sistema permite controlar un motor a pasos, mediante una interfase conectada al puerto paralelo de la PC, con el fin de hacer girar una plataforma sobre la que se ubica la guía óptica a ser estudiada.

El sistema permite la adquisición de datos, los cuales toma de un multímetro digital que está conectado al puerto serial de la PC. El multímetro mide los niveles de voltaje que se generan en el sensor al incidir la luz.

El sistema permite obtener los modos de una guía óptica plana, los cuales permiten la caracterización de la guía obteniendo su espesor y su índice de refracción.

Todo esto es controlado mediante un programa desarrollado en Turbo Pascal.

Los resultados obtenidos mediante este sistema, por ejemplo, en la determinación de índices de refracción, están en total concordancia con resultados obtenidos usando otros métodos.

Introducción

En la actualidad el uso de la PC¹, es indispensable en cualquier área de trabajo, especialmente en el área científica, por su gran velocidad y precisión al realizar cálculos muy complejos.

Asimismo los puertos de comunicación de la PC son de particular interés para el diseño de sistemas de adquisición y control, ya que permiten utilizar una computadora personal para controlar todo tipo circuitos electrónicos, utilizados principalmente, en actividades de automatización de procesos, adquisición de datos, tareas repetitivas y otras actividades que demandan precisión.

Por otro lado, las exigencias actuales, requieren sistemas de comunicaciones y ordenadores de alta velocidad, en tal sentido, es necesario modificar la tecnología actual de circuitos integrados hacia tecnologías basadas en interconexiones ópticas que requieren de la utilización de circuitos optoelectrónicos integrados. En tal sentido es importante, el estudio y caracterización de las guías ópticas planas.

En el presente trabajo se ha desarrollado un sistema automático de control y adquisición de datos, para un equipo de detección de modos de propagación en guías ópticas planas; esto se ha logrado básicamente, mediante la programación de los puertos paralelo y serial de la PC; y otros elementos tales como: el multímetro, el motor a pasos y el sensor de luz.

¹ Personal Computer (Computadora Personal)

Fundamento teórico

2.2 Modos de guía de ondas

Para que un rayo de luz se pueda propagar en una guía óptica plana de un extremo al otro (figura 1.1), este debe cumplir la condición de resonancia [1], [2]:

$$2k_0 n_\lambda d \operatorname{sen} \theta_m - 2\phi_V - 2\phi_P = 2\pi m \dots\dots\dots (1.1)$$

En la ecuación 1.1:

$$k_0 = 2\pi / \lambda$$

d es el espesor de la guía óptica.

θ_m es el ángulo de propagación del rayo dentro de la guía óptica.

m es un entero (0, 1, 2...), que identifica los modos de propagación.

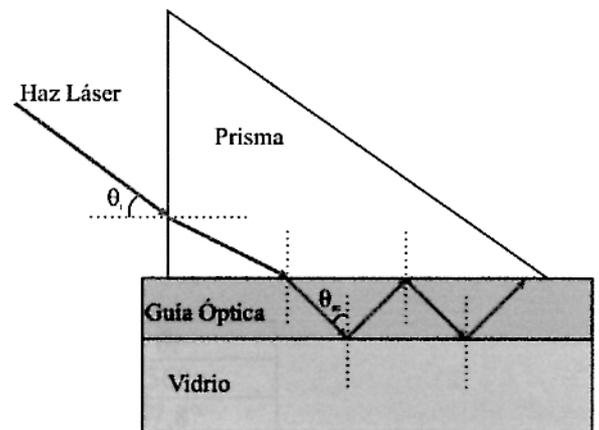


Figura 1.1

ϕ_V y ϕ_P son los desfases que se producen en las respectivas interfaces vidrio, prisma:

$$\phi_V = \operatorname{tg}^{-1} \left[\frac{(n_\lambda^2 \operatorname{sen}^2 \theta_m - n_V^2)^{1/2}}{(n_\lambda \operatorname{cos} \theta_m)} \right] \dots\dots\dots (1.2)$$

$$\phi_A = \operatorname{tg}^{-1} \left[\frac{(n_\lambda^2 \operatorname{sen}^2 \theta_m - n_P^2)^{1/2}}{(n_\lambda \operatorname{cos} \theta_m)} \right] \dots\dots\dots (1.3)$$

La ecuación 1.1 se conoce como ecuación de resonancia y es una ecuación trascendente, que debe ser resuelta por aproximaciones sucesivas y permite obtener de manera directa, el índice de refracción de la guía (n_λ) y el espesor de la guía (d), siempre y cuando se conozcan por lo menos dos valores consecutivos θ_{mi} y m_i .

A continuación se muestra algunos ejemplos de aplicación de las ecuaciones 1.1, 1.2 y 1.3:

1.1.1 Primer caso

Determinación del número de modos y sus respectivos ángulos, para una guía cuyo índice de refracción es $n = 1,6$ y cuyo espesor es $d = 1\mu m$.

El haz láser que se usa tiene una longitud de onda de $\lambda = 633nm$, asimismo el prisma que se usa tiene un índice de refracción de $n_p = 1,758$ y el vidrio tiene un índice de refracción de $n_v = 1,51$. Reemplazando estos datos en las ecuaciones 1.1, 1.2, 1.3 se obtiene:

$$15,88 \text{Sen}^2 \theta_m - \text{tg}^{-1} \left(\frac{\sqrt{2,56 \text{Sen}^2 \theta_m - 2,28}}{1,6 \text{Cos} \theta_m} \right) - \text{tg}^{-1} \left(\frac{\sqrt{2,56 \text{Sen}^2 \theta_m - 3,09}}{1,6 \text{Cos} \theta_m} \right) = m\pi \dots\dots\dots (1.4)$$

La ecuación 1.4 se resuelve por métodos numéricos; en el apéndice D se adjunta un programa desarrollado en Turbo Pascal, usando el método de la bisección, que resuelve esta ecuación. Los resultados que se obtienen son:

Modo (m)	θ_m	θ_i
0	8,9°	56°
1	17,4°	67,8°

Tabla 1.1

1.1.2 Segundo caso

Determinación del índice de refracción de una guía cuyos modos se observan en los ángulos de incidencia $\theta_i = 56^\circ$ y $67,8^\circ$.

Al igual que en el primer caso, la solución de las ecuaciones se hace por métodos numéricos. En el apéndice E se adjunta un programa desarrollado en Turbo Pascal, usando el método de la bisección, que resuelve estas ecuaciones. Los resultados que se obtienen son:

Modos (m)	n_2	$d(\mu m)$
0 y 1	1.6	0.999

Tabla 1.2

Parte Experimental

2.1 Descripción General

El sistema desarrollado en el laboratorio de óptica de la Facultad de Ciencias de la Universidad Nacional de Ingeniería [3], para realizar trabajos en el tema de guías ópticas planas, se muestra en la figura 2.1; el sistema consta de una plataforma giratoria, sobre la cual va montado el conjunto prisma, guía, vidrio y sensor, el giro de la plataforma permite variar el ángulo de incidencia θ_i , el prisma permite introducir el haz láser dentro de la guía en un amplio rango angular, un sensor ubicado en el extremo derecho de la guía genera un voltaje proporcional a la intensidad del haz transmitido, y está conectado a un multímetro digital del cual se toma la lectura de voltaje.

Este sistema en sus inicios era operado de forma manual, es decir el giro de la plataforma y las lecturas del multímetro se hacían manualmente; esta forma de trabajo carecía de precisión y era bastante tedioso.

El presente trabajo experimental, consiste en automatizar el sistema antes descrito, mediante la ayuda de una PC, un motor a pasos y multímetro digital.

En el sistema actual el giro de la plataforma se hace mediante un motor a pasos, el cual es controlado por una PC a través de una interfase conectada al puerto paralelo; la plataforma esta unida al motor mediante un sistema de engranajes, de tal forma que el ángulo que gira el motor en cada paso ($0,9^\circ$) provoca un giro de $0,033^\circ$ de la plataforma; logrando una precisión que es imposible manualmente.

Asimismo, en el sistema actual, la toma o adquisición de datos también es automática, es decir, la PC toma los datos directamente desde un multímetro digital, conectado al puerto serial.

Finalmente, el sistema es capaz de mostrar en pantalla un gráfico de la Intensidad del haz láser transmitido en función del ángulo de incidencia; estos datos a su vez son almacenados en el disco duro, para su posterior tratamiento.

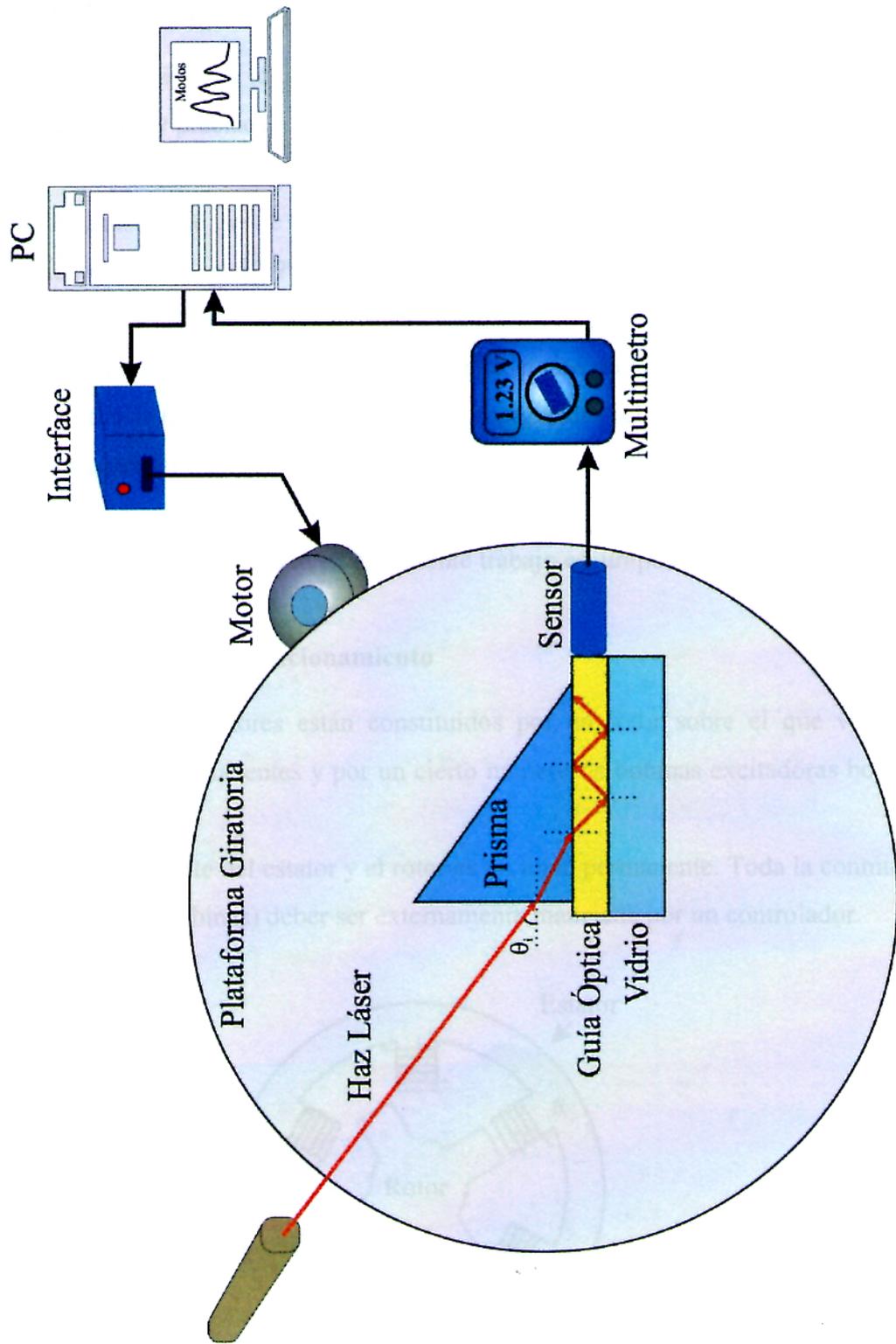


Figura 2.1

2.2 Descripción detallada de las partes

2.2.1 El Motor a pasos [4]

Los motores a pasos son ideales para la construcción de mecanismos en donde se requieren movimientos muy precisos.

La característica principal de estos motores es el hecho de poder moverlos un paso a la vez por cada pulso que se le aplique. El paso puede variar desde 90° hasta pequeños movimientos de tan solo 0.9° , es decir, que se necesitarán 4 pasos en el primer caso (90°) y 400 pasos en el segundo caso (0.9°), para completar un giro completo de 360° .

Estos motores tienen la posibilidad de quedar enclavados en una posición o totalmente libres. Si una o más de sus bobinas están energizadas, el motor estará enclavado en la posición correspondiente y por el contrario quedará completamente libre si no circula corriente por ninguna de sus bobinas.

El tipo de motor a pasos usado en el presente trabajo es, unipolar de imán permanente.

2.2.1.1 Principio de funcionamiento

Básicamente estos motores están constituidos por un rotor sobre el que van aplicados distintos imanes permanentes y por un cierto número de bobinas excitadoras bobinadas en su estator.

Las bobinas son parte del estator y el rotor es un imán permanente. Toda la conmutación (o excitación de las bobinas) deber ser externamente manejada por un controlador.

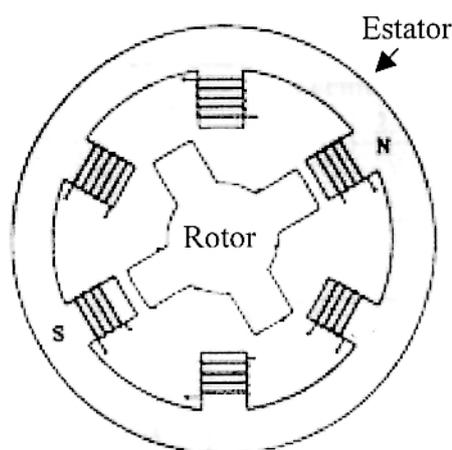


Figura 2.2: Imagen del rotor y del estator

2.2.1.2 Tipos de motores a pasos de imán permanente

a) Bipolar

Estos motores generalmente tienen cuatro cables de salida (ver figura 2.3). Necesitan ciertos trucos para ser controlados, debido a que requieren del cambio de dirección del flujo de corriente a través de las bobinas en la secuencia apropiada para realizar un movimiento.

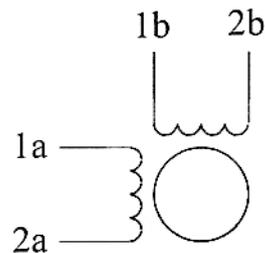


Figura 2.3: Motor Bipolar

a.1) Secuencia para controlar motores a pasos Bipolares

Como se dijo anteriormente, estos motores necesitan la inversión de la corriente que circula en sus bobinas en una secuencia determinada. Cada inversión de la polaridad provoca el movimiento del eje en un paso. El sentido de giro está determinado por la secuencia seguida; por ejemplo si se sigue la secuencia, 1, 2, 3, 4, el motor gira 4 pasos en cierto sentido, pero si seguimos la secuencia 4, 3, 2, 1 el motor gira 4 pasos en sentido contrario.

Paso	Terminal			
	1a	1b	2a	2b
1	+V	-V	-V	-V
2	-V	+V	-V	-V
3	-V	-V	+V	-V
4	-V	-V	-V	+V
1	+V	-V	-V	-V
2	-V	+V	-V	-V
3	-V	-V	+V	-V
4	-V	-V	-V	+V

Tabla 2.1: Secuencia de control de giro de un Motor Bipolar

b) Unipolar

Estos motores suelen tener 6 o 5 cables de salida, dependiendo de su conexionado interno (ver figura 2.4). Este tipo se caracteriza por ser más simple de controlar.

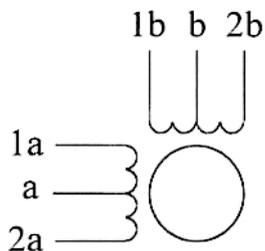


Figura 2.4: Motor Unipolar

b.1) Secuencia para controlar motores a pasos Unipolares

Existen varias secuencias posibles para este tipo de motores, a continuación se detalla tres de las más importantes. Todas las secuencias comienzan por el paso 1 hasta alcanzar el paso final 4 u 8 y repetir la secuencia. Para invertir el sentido de giro, simplemente se deben ejecutar la secuencia en modo inverso.

b.1.1) **Secuencia Normal.-** Esta es la secuencia más usada y la que generalmente recomienda el fabricante. Con esta secuencia el motor avanza un paso por vez y debido a que siempre hay al menos dos bobinas activadas, se obtiene un alto torque de paso y de retención.

Paso	Bobina			
	a-1a	b-1b	a-2a	b-2b
1	1	1	0	0
2	0	1	1	0
3	0	0	1	1
4	1	0	0	1
1	1	1	0	0
2	0	1	1	0
3	0	0	1	1
4	1	0	0	1

Tabla 2.2: Secuencia Normal de control de giro de un Motor Unipolar

b.1.2) **Secuencia del tipo wave drive.-** En esta secuencia se activa solo una bobina a la vez. En algunos motores esto brinda un funcionamiento más suave. La contrapartida es que al estar solo una bobina activada, el torque de paso y retención es menor.

Paso	Bobina			
	a-1a	b-1b	a-2a	b-2b
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1

Tabla 2.3: Secuencia Wave Drive de control de giro de un Motor Unipolar

b.1.3) **Secuencia del tipo medio paso.-** En esta secuencia se activan las bobinas de tal forma de brindar un movimiento igual a la mitad del paso real. Para ello se activan primero 2 bobinas y luego solo 1 y así sucesivamente. Como vemos en la tabla 2.4 la secuencia completa consta de 8 movimientos en lugar de 4.

Paso	Bobina			
	a-1a	b-1b	a-2a	b-2b
1	1	0	0	0
2	1	1	0	0
3	0	1	0	0
4	0	1	1	0
5	0	0	1	0
6	0	0	1	1
7	0	0	0	1
8	1	0	0	1
1	1	0	0	0
2	1	1	0	0
3	0	1	0	0
3	0	1	1	0
5	0	0	1	0
6	0	0	1	1
7	0	0	0	1
8	1	0	0	1

Tabla 2.4: Secuencia Medio Paso de control de giro de un Motor Unipolar

Debido a que los motores a pasos son dispositivos mecánicos y como tal deben vencer ciertas inercias, el tiempo de duración y la frecuencia de los pulsos aplicados es un punto muy importante a tener en cuenta. En tal sentido el motor debe alcanzar el paso antes que la próxima secuencia de pulsos comience. Si la frecuencia de pulsos es muy elevada, el motor puede reaccionar en alguna de las siguientes formas:

- Puede que no realice ningún movimiento en absoluto.
- Puede comenzar a vibrar pero sin llegar a girar.
- Puede girar erráticamente.
- puede llegar a girar en sentido opuesto.

Para obtener un arranque suave y preciso, es recomendable comenzar con una frecuencia de pulso baja y gradualmente ir aumentándola hasta alcanzar la velocidad deseada sin superar la máxima tolerada. El giro en reversa también debe ser realizado previamente bajando la velocidad de giro y luego cambiar el sentido de rotación.

2.2.2 El puerto paralelo de la PC [5], [8]

El puerto paralelo originalmente desarrollado por IBM para manejar la impresora, es el más usado para proyectos de interconexión con la PC; por ejemplo, por este puerto se puede monitorear un conversor análogo digital, controlar un motor a pasos, etc.

La transmisión en paralelo entre un ordenador y un periférico, se basa en la transmisión de datos simultáneamente por varios canales, generalmente 8 bits. Por esto se necesitan 8 cables para la transmisión de cada bit, más otros tantos cables para controles del dispositivo.

El puerto se compone de 4 líneas del control, 5 líneas de estado y 8 líneas de datos.

Un sistema de comunicación en paralelo puede ser de una dirección (unidireccional) o de dos direcciones (bidireccional). El mecanismo más simple utilizado en un puerto paralelo de la PC es de tipo unidireccional y es el que se usará en el presente trabajo. En un sistema de comunicación paralelo se distinguen dos elementos:

1. La parte transmisora, la cual coloca la información en las líneas de datos e informa a la parte receptora que la información (los datos) están disponibles.
2. La parte receptora, es la lee la información en las líneas de datos e informa a la parte transmisora que ha tomado la información (los datos).

Ambas partes sincronizan su respectivo acceso a las líneas de datos, la parte receptora no leerá las líneas de datos hasta que la parte transmisora se lo indique en tanto que la parte transmisora no colocará nueva información en las líneas de datos hasta que la parte receptora remueva la información y le indique a la parte transmisora que ya ha tomado los datos, a ésta coordinación de operaciones se le llama acuerdo ó entendimiento (handshaking).

2.2.2.1 El hardware del puerto paralelo [9]

El puerto paralelo de la PC utiliza un conector hembra, tipo D de 25 pines (DB25), el cual se encuentra ubicado en la parte posterior. La figura 2.5 muestra este conector así como la disposición de pines.

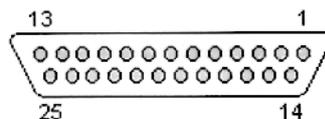


Figura 2.5: Conector DB25 hembra

Pin N°	E/S	Activa en	Señal	Descripción	Registro
1	Entrada/ Salida	0	Strobe	Activa, habilita a la impresora para que reciba datos	Control
2 - 9	Salida	-	Data	bit 0 Pin 2, bit 7 Pin 9	Datos
10	Entrada	0	Acknowledge	Activa cuando el sistema remoto toma datos	Estado
11	Entrada	0	Busy	Si está activa, el sistema remoto no acepta datos	Estado
12	Entrada	1	Paper End	Si está activa, falta papel en la impresora	Estado
13	Entrada	1	Select	Si está activa, la impresora se ha seleccionado	Estado
14	Entrada/ Salida	0	Auto feed	Si está activa, la impresora inserta una nueva línea por cada retorno de carro	Control
15	Entrada	0	Error	Si está activa, hay un error en la impresora	Estado
16	Entrada/ Salida	0	Init	Si se mantiene activa por al menos 50 us, ésta señal autoinicializa la impresora	Control
17	Entrada/ Salida	0	Select input	Cuando está inactiva, obliga a la impresora a salir de línea	Control
18 - 25	-	-	Ground	Tierra eléctrica	

Tabla 2.5: Disposición de Pines del Conector DB25 hembra

En la tabla 2.5, la columna “Activa en” significa que la señal correspondiente se activa en “0” o en “1”, por ejemplo si se tiene un “0” lógico en la línea busy, entonces la impresora está ocupada y no acepta datos.

La salida del puerto paralelo usa niveles lógicos TTL, es decir un voltaje entre 0 y 0.4 voltios representa un “0” lógico y un voltaje entre 2.4 y 5 voltios representa un “1” lógico. La corriente que se puede tomar del puerto es muy baja y varía de puerto a puerto. La mayoría de los puertos paralelos pueden suministrar alrededor de 12mA. Por ello, cuando se diseña un circuito de interfase para el puerto paralelo, lo mejor es utilizar un buffer.

2.2.2.2 Direcciones de los Puertos Paralelos

El puerto paralelo tiene tres direcciones base de uso general. En la tabla 2.6 se enumeran las direcciones de los puertos paralelos. La dirección base 3BCh originalmente fue introducida en algunas tarjetas video antiguas, desaparecieron durante algún tiempo,

cuando los puertos paralelos fueron quitados de las tarjetas video. Ahora han reaparecido como un un puerto paralelo opcional, integrado en la mainboard.

Dirección	Notas
3BCh - 3BFh	Usado en puertos paralelos de tarjetas de Video
378h - 37Fh	Dirección usual para LPT 1
278h - 27Fh	Dirección usual para LPT 2

Tabla 2.6: Direcciones de los puertos Paralelos

Cuando la PC se enciende, el BIOS (Sistema Básico de Entrada – Salida) ejecuta una rutina para determinar el número de puertos presentes en el sistema asignando la etiqueta LPT1 al primer puerto localizado, si existen más puertos entonces se asignarán consecutivamente las etiquetas LPT2 y LPT3. Generalmente LPT1 se asigna a la dirección base 378h, mientras que LPT2 se asigna a la dirección base 278h, la h minúscula denota que está en hexadecimal.

2.2.2.3 Registros del Puerto Paralelo Estándar

Hay tres direcciones o registros de Entrada/Salida asociadas con el puerto paralelo de la PC, estos registros son:

a) El registro de datos (Dirección Base)

Es un puerto de lectura - escritura de ocho bits. Leer el registro de datos (en la modalidad unidireccional) retorna el último valor escrito en el registro de datos.

Dirección	Nombre	Lectura Escritura	bit N°	Propiedades
Base + 0	Puerto de datos	Escritura	bit 7	Dato 7
			bit 6	Dato 6
			bit 5	Dato 5
			bit 4	Dato 4
			bit 3	Dato 3
			bit 2	Dato 2
			bit 1	Dato 1
			bit 0	Dato 0

Tabla 2.7: Registro de Datos

b) El registro de estado (Dirección Base + 1)

Es un puerto solo de lectura. Cualquier dato escrito a este puerto será ignorado. El registro de estado se compone de 5 líneas de entrada (Pines 10, 11, 12, 13 y 15), un bit IRQ y dos bits reservados. La lectura de este registro, da el estado de los cinco pines de entrada al momento de la lectura.

Dirección	Nombre	Lectura Escritura	bit N°	Propiedades
Base + 1	Puerto de Estado	Solo Escritura	bit 7	Busy
			bit 6	Acknowledge
			bit 5	Falta de papel
			bit 4	Select In
			bit 3	Error
			bit 2	IRQ (Not)
			bit 1	Reservado
			bit 0	Reservado

Tabla 2.8: Registro de Estado

c) El registro de control (Dirección base + 2)

Es un puerto de lectura - escritura. Esta compuesto por 4 líneas de salida (Pines 1, 14, 16, 17). La lectura de este registro devuelve lo último que se escribió a dichos bits.

Dirección	Nombre	Lectura Escritura	bit N°	Propiedades
Base + 2	Puerto de Control	Lectura Escritura	bit 7	No usado
			bit 6	No usado
			bit 5	Permite puerto Bidireccional
			bit 4	Permite IRQ a través de la línea Acknowledge
			bit 3	Selecciona Impresora
			bit 2	Inicializa Impresora (Reset)
			bit 1	Auto Linefeed
			bit 0	Strobe

Tabla 2.9: Registro de Control

2.2.2.4 Interfaz para el puerto paralelo

Actualmente, la mayoría de los puertos instalados en las computadoras son de tipo multimodal configurables a través de la BIOS de la PC, en el presente trabajo trataremos el modo normal SPP (standard parallel port).

El modo de operación normal es el más elemental y solamente permite la escritura en las líneas de datos, patitas 2 a la 9 del conector DB-25 del puerto paralelo de la PC.

Eléctricamente, el puerto paralelo entrega señales TTL y como tal, teóricamente, se le puede conectar cualquier dispositivo que cumpla con los niveles de voltaje específicos de la lógica TTL, sin embargo el hardware del puerto paralelo está muy limitado en cuanto a su capacidad de manejo de corriente, por ésta razón se debe ser muy cuidadoso con el manejo de las señales del puerto, un corto circuito puede dañar permanentemente la tarjeta madre de la PC. Para disminuir lo más posible el riesgo de daños al puerto utilizamos el circuito integrado 74LS245 (Bus Transceiver) como etapa separadora y al mismo tiempo mejoramos la capacidad de manejo de corriente.

En la figura 2.6 muestra el circuito que actúa como interfase entre el puerto paralelo de la PC y el motor paso a paso.

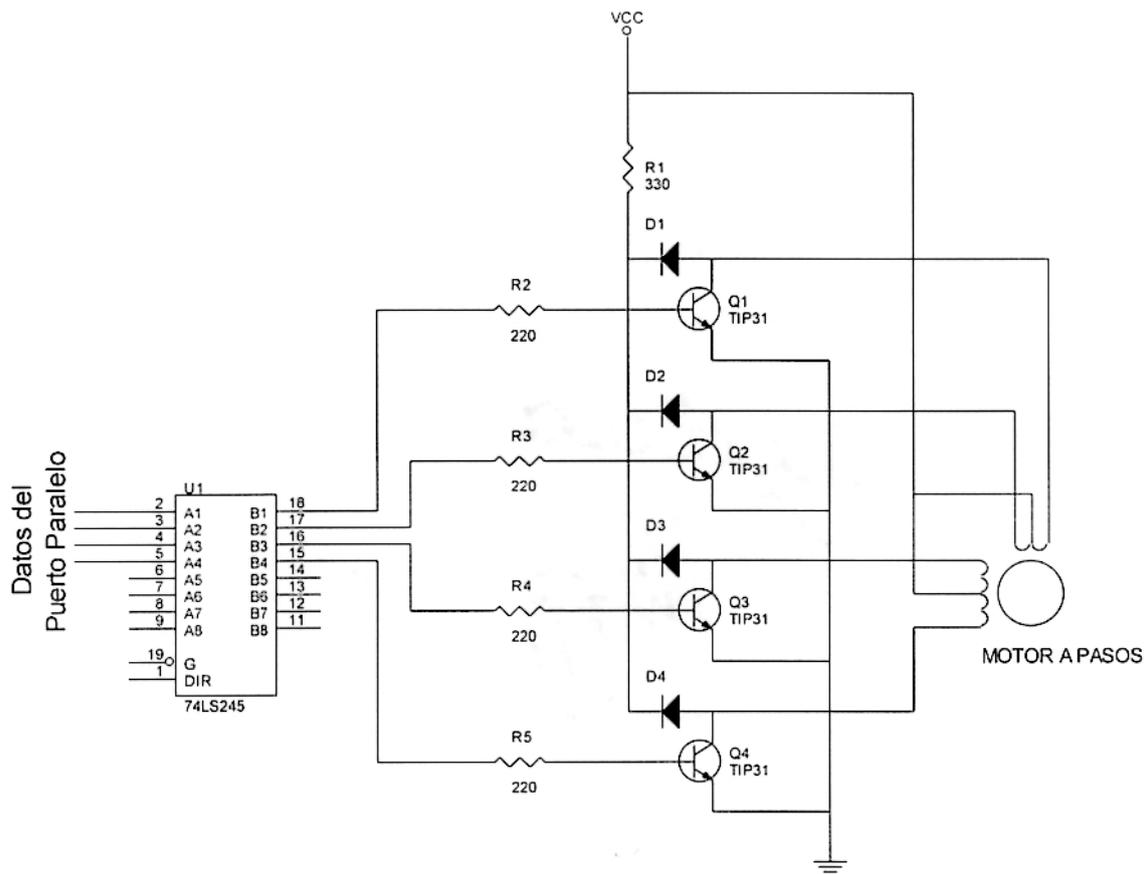


Figura 2.6: Circuito Para Controlar Un Motor a Pasos a través del Puerto Paralelo de la PC

El circuito integrado 74LS245, permite controlar la dirección de la transmisión de datos, del bus A al bus B o viceversa. Para poder controlar el motor paso a paso solo se requieren 4 líneas de datos del puerto paralelo (patitas del 2 al 5 del conector DB-25), los cuales pasan a través del 74LS245, el cual actúa transmitiendo datos en una sola dirección, de la PC hacia el motor; este integrado, también entrega corriente suficiente para activar los transistores (TIP31), que son los que controlan la corriente que circula por las bobinas del motor, los diodos actúan como protección contra las corrientes autoinducidas que se pueden producir en las bobinas del motor. Para operar éste circuito, se requiere de una fuente regulada de 5 Voltios.

El diseño de las pistas, para la implementación en circuito impreso, del circuito esquemático de la figura 2.6, incluyendo una fuente regulada de 5 V, se adjunta en el apéndice B.

La hoja de especificaciones de los integrados usados se adjunta en el apéndice C.

La figura 2.7, muestra la interfase ensamblada.

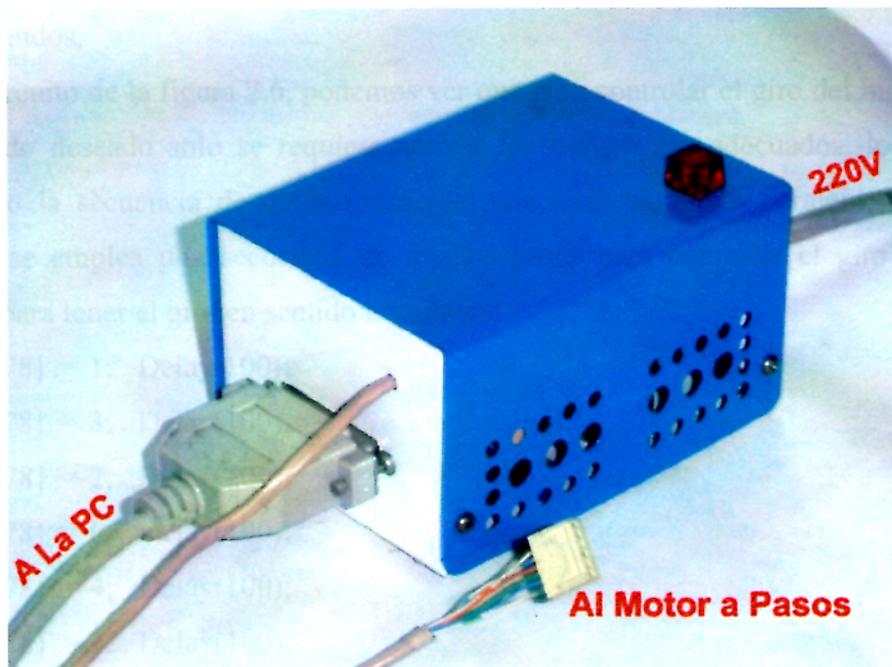


Figura 2.7: Interfase para conectar el motor a pasos al puerto paralelo de la PC.

2.2.2.5 Programación del puerto paralelo

El puerto paralelo cuenta con 8 bits de datos, cada uno de los cuales puede tomar los valores lógicos “1” ó “0”, por lo tanto se puede escribir un total de 256 valores diferentes,

en el puerto paralelo; cada uno de estos valores representa un Byte de información. Cuando escribimos un número entre 0 y 255 en el puerto paralelo, este número se presenta en formato binario (0V ó 5V en lógica TTL) en las líneas de datos, siendo el pin 2 del conector DB-25 el bit menos significativo y el pin 9 el bit mas significativo.

El comando para escribir un número (dato) en el puerto paralelo (378h), depende del lenguaje de programación que se use, en este caso el lenguaje de programación usado es el Turbo Pascal, en cuyo caso el comando es:

```
Port[$378] := Dato;
```

Donde \$378 representa la dirección base del puerto paralelo en formato hexadecimal y dato representa el número que se quiere escribir en el puerto.

Debido a que la PC puede enviar datos al puerto mucho mas rápido que lo que es capaz de responder el motor, es necesario introducir un tiempo de retraso entre cada dato que se envía al puerto, para ello el Turbo Pascal tiene el siguiente comando:

```
Delay(T);
```

Donde T es el tiempo de espera antes de enviar el siguiente número, expresado en milisegundos.

En el circuito de la figura 2.6, podemos ver que para controlar el giro del motor a pasos en el sentido deseado solo se requiere activar los transistores adecuados, lo cual se logra enviando la secuencia de datos (números) adecuada al puerto paralelo; en el presente trabajo se emplea una secuencia de medios pasos para controlar el giro del motor; el código para tener el giro en sentido horario es:

```
Port[$378] := 1; Delay(100);
```

```
Port[$378] := 3; Delay(100);
```

```
Port[$378] := 2; Delay(100);
```

```
Port[$378] := 6; Delay(100);
```

```
Port[$378] := 4; Delay(100);
```

```
Port[$378] := 12; Delay(100);
```

```
Port[$378] := 8; Delay(100);
```

```
Port[$378] := 9; Delay(100);
```

Para hacer girar el motor, en sentido antihorario, solo se tiene que enviar la secuencia en sentido inverso, es decir, comenzando por la última sentencia y terminando en la primera.

2.2.3 El Sensor

El sensor, es el dispositivo que genera las señales de voltaje proporcionales a la intensidad del haz láser incidente. En este caso, es una celda fotovoltaica de silicio monocristalino, cuyas características más importantes son: tiempo de respuesta muy rápido y eficiencia muy buena para la longitud de onda de trabajo (láser de 633 nm).

La respuesta espectral es una medida de la eficiencia con que en un determinado dispositivo fotovoltaico, produce la conversión de energía luminosa en energía eléctrica para una determinada frecuencia de la luz incidente.

En una célula típica de silicio monocristalino dicha eficiencia de conversión solamente es significativa para longitudes de onda comprendidas entre 350 y 1100 nanómetros. Con un máximo alrededor de los 800 nanómetros. Otros materiales tienen una respuesta espectral diferente.

El sensor que se usa en este sistema se ha construido a partir de un pedazo de panel solar de silicio monocristalino. Este material es muy frágil, lo cual lo hace muy poco práctico a la hora de ser manipulado; debido a esto se ha tenido que construir el diseño que se muestra en la figura 2.8.

En este diseño se puede ver, que el haz láser que sale de la guía óptica, es guiado mediante una fibra óptica hacia la celda fotovoltaica; el conjunto se encuentra encapsulado dentro de una cubierta plástica, de esta forma solo se requiere que el extremo izquierdo de la fibra óptica este en contacto con el extremo derecho de la guía óptica; haciendo mucho más práctico su manipulación.

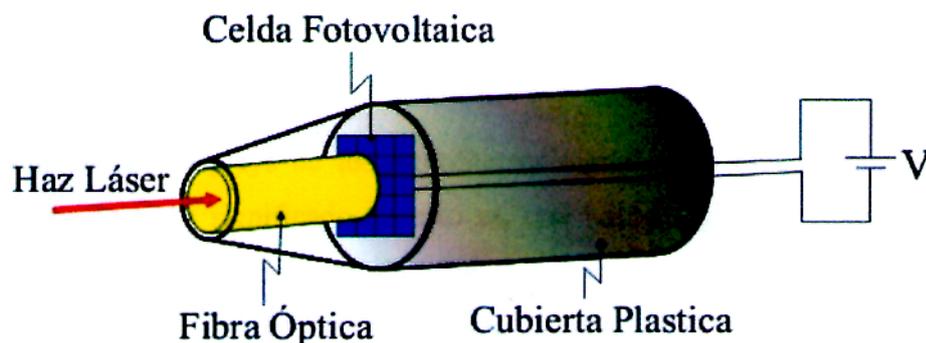


Figura 2.8

2.2.4 El multímetro digital

El multímetro digital es el más común de los instrumentos de medición eléctrica y electrónica. Para los propósitos del presente trabajo, el multímetro se usa exclusivamente para medir voltajes DC, el multímetro con el cual se ha trabajado es el Voltcraft M-4560CR (figura 2.9), cuyas características más importantes son: display de 4½ dígitos, resolución de 10 μV en la escala de 200 mV, precisión $\pm(0,05\% + 3\text{dígitos})$ en la escala de 200 mV y una interfase de comunicación RS232 para interconexión con la PC.



Figura 2.9: Multímetro Voltcraft M-4560CR

2.2.4.1 El display de 4½ dígitos

Tradicionalmente, los displays de los multímetros digitales, se especifican, como: “4½ dígitos” por ejemplo. El significado de esto es, que hay 4 dígitos completos, cada uno capaz de exhibir los números del cero al nueve, y un dígito precedente adicional que puede exhibir solamente el “0” (no se muestra en este caso) o el “1” para una lectura a escala completa de 19999. Esto quiere decir, que en una escala determinada, puede hacer 20000 cuentas, por ejemplo, considerando, la escala de 200 mV, la resolución (mínima lectura) en este caso será $(200 \text{ mV} / 20000) = 10 \text{ uV}$.

El número de cuentas se aplica generalmente a la función de medida de voltajes DC. Para ciertas funciones, solo se puede exhibir pocas cuentas en el mismo instrumento. Por ejemplo, 2000 cuentas al medir capacitancia.

2.2.4.2 Precisión

La precisión de un instrumento digital, en este caso del multímetro se expresa como un porcentaje de la lectura mas un cierto número de dígitos o cuentas, que no es otra cosa que la resolución en determinada escala, por ejemplo, para una lectura de voltaje DC en la escala de 200 mV, la precisión es $\pm(0,05\% + 3\text{digitos})$. Por lo tanto, para una lectura de 140,52 mV la incertidumbre será $\pm[0,05(140,52 \text{ mV})/100 + 3(10^{-3} \text{ mV})] = \pm 0.07 \text{ mV}$.

2.2.4.3 Formato de datos

El formato de datos enviado por el multímetro por cada medición, consta de 14 Bytes, de los cuales los Bytes 1, 2 contienen la función DC/AC, el Byte 3 es la polaridad +/- , los Bytes del 4 al 9 el valor da la medición, los Bytes de la A a la D las unidades A, mA, uA, V, mV, uV, ohm, Kohm, Mohm, etc. y finalmente el Byte E es un retorno de carro CR (enter).

Por ejemplo:

1	2	3	4	5	6	7	8	9	A	B	C	D	E
D	C	-		3	.	9	9	9				V	CR

2.2.4.4 Interfase de conexión con la PC

En la figura 2.10 se muestra la interfase de conexión con la PC.

Si la PC envía una petición del servicio al multímetro, entonces la salida RTS se pone brevemente en +12 V. El LED del opto acoplador activa el transistor asociado (parte inferior de la figura), dejando pasar el mensaje al multímetro.

La respuesta del multímetro son una serie de impulsos al LED del opto acoplador (parte superior de la figura). Cuando el transistor asociado se activa, conecta el pin DTR que se encuentra a +12 V cuando la computadora esta lista para recibir datos, con el pin RXD, que

recibe los datos. Cuando el transistor no está activo, la tensión pasa desde el pin RTS, que normalmente se encuentra a -12 V a través de una resistencia al pin RXD.

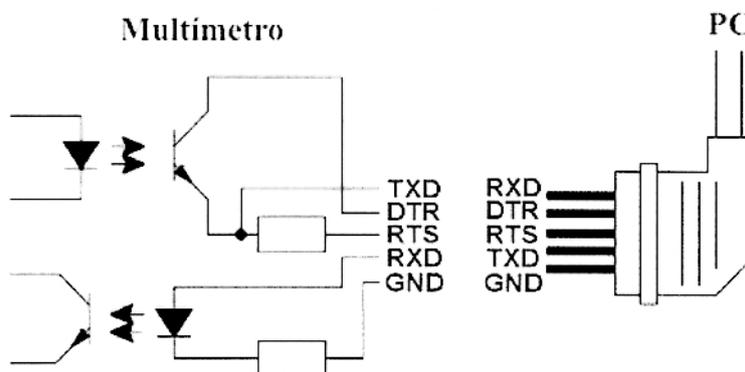


Figura 2.10: Interfase de conexión con la PC

2.2.4.5 Comandos del multímetro

El multímetro puede enviar datos en forma automática, para ello es necesario seleccionar la función COM

El multímetro también puede enviar datos a petición de la PC, no es necesario seleccionar la función COM. En este caso los comandos (caracteres), que puede enviar la PC al multímetro son:

- **M** El multímetro envía los datos almacenados en su memoria.
- **C** El multímetro borra los datos almacenados en su memoria.
- **D** El multímetro envía datos del display.

2.2.4.6 Especificaciones de comunicación para la transmisión de datos

Velocidad de comunicación	:	1200 bps.
Longitud de la palabra	:	7 bits ASCII ² .
Paridad	:	Ninguna.
bits de parada	:	2.

²American Standard Code for Information Interchange (Código Estadounidense Estándar para el Intercambio de Información)

2.2.5 El puerto serial (RS232) [6], [7]

El puerto serial es más difícil de interconectar que el puerto paralelo. Por lo general, cualquier dispositivo que se conecte con el puerto serial necesita convertir la transmisión serial de nuevo a paralelo para poderla utilizar. Esto se logra mediante un chip conocido como UART. En la parte de programación, hay muchos más registros que atender, que en un puerto paralelo estándar (SPP).

Ventajas de usar transferencia de datos serial en lugar de paralelo:

- Los cables seriales pueden ser más largos que los cables paralelos. El puerto serial transmite un “1” como un nivel de voltaje entre -3 y -25 voltios y un “0” como un nivel de voltaje entre +3 y +25 voltios, mientras que un puerto paralelo transmite un “0” como 0V y “1” como 5V. Por lo tanto el puerto serial puede tener una variación máxima de 50V comparado al puerto paralelo que tiene una variación máxima de 5V. De esta forma la pérdida en el cable no va a ser un gran problema para los cables seriales como lo es para los paralelos.
- No se necesita tantos cables como en la transmisión paralela. Si su dispositivo necesita ser montado una distancia lejos de la computadora entonces un cable de 3 hilos (configuración null modem) es suficiente y mucho más barato que un cable de 19 o 25 hilos.
- Muchos dispositivos tales como calculadoras, multímetros, etc. incorporan un puerto RS232.
- Los dispositivos infrarrojos que se han vuelto muy populares en la actualidad, tales como calculadoras, agendas electrónicas, computadoras portátiles, etc. Incorporan un puerto infrarrojo, estos dispositivos utilizan la transmisión serial.
- El microcontrolador se ha vuelto muy popular. Muchos de éstos tienen incorporado un SCI (interfase de comunicación serial) que se puede utilizar para comunicarse con el mundo exterior. La comunicación serial reduce el número de pines de estos microcontroladores. Solo dos pines son necesarios, el transmit data (TXD) y el receive data (RXD), comparado con por lo menos 8 pines si se utiliza un método de transmisión paralelo de 8 bits.

2.2.5.1 Características del hardware

Los dispositivos que utilizan el puerto serial para su comunicación se dividen en dos categorías, estas son:

- DCE (Data Communications Equipment), equipo de comunicación de datos, generalmente es el módem.
- DTE (Data Terminal Equipment), equipo terminal de datos, que generalmente es la PC.

Las especificaciones eléctricas del puerto serial están dadas en el estándar RS232C que significa literalmente "Recomended Standard 232 revisión C" (también conocida como EIA 232). Es un estándar publicado en 1969 por la EIA ("Electronic Industries Alliance") que define las características eléctricas que deben presentar los elementos de conexión para la comunicación serie entre ordenadores y equipos periféricos. De los muchos parámetros que se indican a continuación se menciona los más importantes:

1. Un "espacio" (0 lógico) estará entre +3 y +25 voltios.
2. Una "marca" (1 lógico) estará entre -3 y -25 voltios.
3. La región entre -3 y +3 voltios no esta definida.
4. Un voltaje en circuito abierto no debe exceder los 25 voltios. (En referencia a tierra).
5. Una corriente en cortocircuito no debe exceder los 500mA. El conductor debe poder conducir esto sin sufrir daño.

Los puertos seriales vienen en dos tamaños, el conector tipo "D" de 25 pines (DB25) figura 2.11 y el conector tipo "D" de 9 pines (DB9) figura 2.12, ambos machos, ubicados en la parte posterior de la PC. En la tabla 2.10, se muestra la disposición de pines de ambos conectores [9].

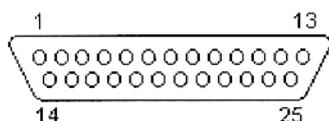


Figura 2.11: Conector D25

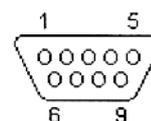


Figura 2.12: Conector D9

Disposición de Pines del puerto serial (conectores DB25 y DB9)

Pin N° (DB25)	Pin N° (DB9)	Abreviatura	Nombre completo
Pin 2	Pin 3	TD	Transmit Data
Pin 3	Pin 2	RD	Receive Data
Pin 4	Pin 7	RTS	Request To Send
Pin 5	Pin 8	CTS	Clear To Send
Pin 6	Pin 6	DSR	Data Set Ready
Pin 7	Pin 5	SG	Signal Ground
Pin 8	Pin 1	CD	Carrier Detect
Pin 20	Pin 4	DTR	Data Terminal Ready
Pin 22	Pin 9	RI	Ring Indicator

Tabla 2.10: Disposición de Pines de los DB25 y DB9

Descripción de los Pines

Abreviatura	Nombre completo	Función
TD	Transmit Data	Salida serial de datos (TXD)
RD	Receive Data	Entrada serial de datos (RXD)
CTS	Clear To Send	Esta línea indica que el módem esta listo para intercambiar datos.
DCD	Data Carrier Detect	Cuando el módem detecta un “mensaje” del módem en el otro extremo de la línea telefónica, esta línea se activa.
DSR	Data Set Ready	Esto dice al UART que el módem esta listo para establecer un enlace.
DTR	Data Terminal Ready	Este es el opuesto a DSR. Esto dice al módem que el UART esta listo para el enlace.
RTS	Request To Send	Esta línea informa al módem que el UART esta listo para intercambiar datos.
RI	Ring Indicator	Se activa cuando el módem detecta una señal de sonido del PSTN.

Tabla 2.11: Descripción de pines del puerto serial

2.2.5.2 Velocidad de transmisión

La velocidad de transmisión es la cantidad de datos transmitidos en la unidad de tiempo. Se expresa en bits por segundo (bps). Por ejemplo, la velocidad máxima del UART 16550A es de 115.200 bps.

2.2.5.3 El UART

El corazón del sistema de comunicaciones seriales es el UART, acrónimo de Universal Asynchronous Receiver Transmitter, el cual es un chip cuya misión principal es convertir los datos recibidos del bus del PC en formato paralelo, a un formato serial que será utilizado en la transmisión hacia el exterior; también realiza el proceso contrario, es decir transforma los datos seriales recibidos del exterior en un formato paralelo entendible por el bus de la PC. El UART es un dispositivo programable en el que pueden establecerse las condiciones que se utilizarán para la comunicación serial (velocidad, paridad, longitud y bits de parada). Los UART mas significativos son: 8250A, 8250B, 16450, 16550, 16550A.

2.2.5.4 Direcciones e Interrupciones del puerto serial

Las direcciones de los puertos de comunicación serial (COM) y sus respectivas interrupciones.

Nombre	Dirección	IRQ
COM 1	3F8	4
COM 2	2F8	3
COM 3	3E8	4
COM 4	2E8	3

Tabla 2.12: Direcciones del puerto serial

2.2.5.5 Registros del UART

La tabla 2.13 muestra los registros o direcciones del UART.

Dirección Base	DLAB	Lectura/Escritura	Abreviatura	Nombre de registro
+ 0	= 0	Escritura	-	Buffer de Transmisión
	= 0	Lectura	-	Buffer de Recepción
	= 1	Lectura/Escritura	-	Byte Inferior del Divisor
+ 1	= 0	Lectura/Escritura	IER	Registro de habilitación de Interrupción
	= 1	Lectura/Escritura	-	Byte Superior del Divisor
+ 2	-	Lectura	IIR	Registro de Identificación de Interrupción
	-	Escritura	FCR	Registro de control de FIFO
+ 3	-	Lectura/Escritura	LCR	Registro de Control de Línea
+ 4	-	Lectura/Escritura	MCR	Registro de Control de Modem
+ 5	-	Lectura	LSR	Registro de estado de Línea
+ 6	-	Lectura	MSR	Registro de estado de Modem
+ 7	-	Lectura/Escritura	-	Registro de Garabato

Tabla 2.13: Tabla de registros del UART

a) DLAB

En la tabla de registros hay una columna llamada DLAB. Cuando DLAB se fija a “0” ó “1” algunos de los registros cambian. De esta forma el UART puede tener 12 registros con solo 8 direcciones de puerto. DLAB significa Divisor Latch Access bit. Cuando DLAB se fija a “1” vía el registro de control de línea (LCR), se tiene, dos registros más disponibles, mediante los cuales se puede fijar la velocidad de las comunicaciones medida en bits por segundo.

El UART es capaz de transmitir y recibir a la velocidad de 115.200 bps. Pero no todos los dispositivos pueden trabajar a esta velocidad por lo tanto el UART tiene un generador de velocidad programable que es controlado por dos registros.

Por ejemplo para fijar una velocidad de comunicación de 1200 bps, se tendrá que dividir 115.200 entre 96 para conseguir 1200 bps. El “divisor”, en este caso 96, este se almacena en los dos registros controlados por el DLAB. Este divisor puede ser cualquier número de 16 bits (0 a 65535). El UART solo tiene un bus de datos de 8 bits, por eso que se utilizan dos registros. El primer registro (Base + 0 cuando DLAB = 1) almacena el Byte Inferior del Divisor y el segundo registro (Base + 1 cuando DLAB = 1) almacena Byte Superior del divisor.

En la tabla 2.14 se muestra algunas velocidades comúnmente usadas. Todos los divisores están dados en formato Hexadecimal.

Velocidad (BPS)	Divisor (DEC)	Byte Superior del Divisor	Byte Inferior del Divisor
50	2304	09h	00h
300	384	01h	80h
600	192	00h	C0h
1200	96	00h	60h
2400	48	00h	30h
4800	24	00h	18h
9600	12	00h	0Ch
19200	6	00h	06h
38400	3	00h	03h
57600	2	00h	02h
115200	1	00h	01h

Tabla 2.14: Tabla de divisores de uso general

b) Registro de Habilitación de Interrupción (IER)

bit	Notas
bit 7	Reservado
bit 6	Reservado
bit 5	Habilita el modo de baja Potencia (16750)
bit 4	Habilita el modo de Dormir (16750)
bit 3	Habilita la Interrupción de estado de Modem
bit 2	Habilita la Interrupción de Estado de la Línea de Recepción
bit 1	Habilita la Interrupción de Buffer de Transmisión Vacío
bit 0	Habilita la Interrupción de Dato Recibido Disponible

Tabla 2.15: El registro de Interrupción

Poniendo el bit 0 en “1” se habilita la interrupción de dato recibido disponible el cual genera una interrupción cuando el registro de recepción (FIFO) contiene los datos que se leerán por la CPU.

El bit 1 habilita la interrupción de buffer de transmisión vacío, esto interrumpe la CPU cuando el buffer del transmisor esta vacío.

El bit 2 habilita la interrupción estado de la línea de recepción, el UART interrumpirá cuando el estado de línea de recepción cambie.

El bit 3 habilita la interrupción de estado del modem, el UART interrumpirá cuando el registro de estado del modem (MSR) cambie.

Los bits 4 y 5 solo están habilitados en el UART 16750.

c) Registro de identificación de interrupción (IIR)

bit	Notas		
bits 6 y 7	bit 6	bit 7	
	0	0	No FIFO
	0	1	FIFO habilitado pero no usable
	1	1	FIFO habilitado
bit 5	FIFO de 64 Bytes habilitado (solo en el 16750)		
bit 4	Reservado		
bit 3	0	Reservado en el 8250, 16450	
	1	Estado de Interrupción de Sobre tiempo en el 16550	
bits 1 y 2	bit 2	bit 1	
	0	0	Interrupción de estado del Modem
	0	1	Interrupción de Buffer de Transmisión vacío
	1	0	Interrupción de Dato recibido disponible
	1	1	Interrupción de Estado de la Línea de Recepción
bit 0	0	Interrupción pendiente	
	1	Ninguna Interrupción pendiente	

Tabla 2.16: Registro de identificación de la interrupción

El registro de identificación de interrupción es solo de lectura. Los bits 6 y 7 indican el estado del buffer FIFO, Cuando ambos bits son “0” no hay buffer FIFO activos. Éste será el único resultado que se obtenga del 8250 o del 16450. Si el bit 7 es “1” y el bit 6 es “0” entonces el UART tiene el buffer FIFO activo pero no usable. Esto ocurre en los UART 16550 donde hay un error en los buffer FIFO que los hace no usables. Si ambos bits son “1” entonces los buffer FIFO están habilitados y completamente operativos.

El bit 3 muestra el estado de la interrupción de sobre tiempo en el UART 16550 o posteriores.

El bit 0 muestra si ha ocurrido una interrupción. Si ha ocurrido una interrupción los bits 1 y 2 mostraran el estado. Estas interrupciones trabajan en un estado de prioridad. La interrupción de estado de la línea de recepción tiene la prioridad más alta, seguida por la interrupción de dato recibido disponible, luego la interrupción de buffer de transmisión vacío y finalmente la interrupción de estado del módem que tiene la prioridad más baja.

d) Registro de Control de FIFO (FCR)

bit	Notas		
bits 6 y 7	bit 7	bit 6	Nivel del disparador de la interrupción
	0	0	1 Byte
	0	1	4 Bytes
	1	0	8 Bytes
	1	1	14 Bytes
bit 5	Habilita el FIFO de 64 Bytes (solo en el 16750)		
bit 4	Reservado		
bit 3	Selecciona el Modo DMA (acceso directo a memoria)		
bit 2	Limpia el FIFO de Transmisión		
bit 1	Limpia el FIFO de Recepción		
bit 0	Habilita los FIFO		

Tabla 2.17: Registro de control del FIFO

El registro FIFO es solo de escritura. Este registro se utiliza para controlar los buffer FIFO que se encuentran en el UART 16550 y posteriores.

El bit 0 habilita los FIFO de transmisión y recepción. Un “0” en este bit inhabilitará la operación de los FIFO y se perderían los datos allí almacenados.

Los bits 1 y 2 puestos a “1” limpian los contenidos de los FIFO de recepción y transmisión, esto no afectará los registros de intercambio.

El bit 3 habilita el modo de acceso directo a memoria que se encuentra en los UART 16550 y posteriores.

El bit 5 habilita el FIFO de 64 Bytes y esta habilitado solo en el UART 16750.

Los bits 6 y 7 se utilizan para fijar el nivel de disparo de la interrupción de los FIFO. Por ejemplo si el bit 7 se fija a “1” y el bit 6 se fija a “0” entonces el nivel de disparo se fija a 8 Bytes. Cuando haya 8 Bytes de datos en el FIFO de recepción, la interrupción de datos recibidos disponibles se activará, ver (IIR).

e) Registro de control de línea (LCR)

bit	Notas			
bit 7	1	bit de acceso al Divisor (DLAB)		
	0	Acceso al buffer receptor, buffer transmisor y a al registro de habilitar interrupción		
bit 6	Habilitar La Terminación			
bits 3, 4 y 5	bit 5	bit 4	bit 3	Selección de Paridad
	X	X	0	Ninguna paridad
	0	0	1	Paridad impar
	0	1	1	Paridad par
	1	0	1	Paridad de Marca
	1	1	1	Paridad de Espacio
bit 2	Longitud del bit de parada			
	0	Un bit de parada		
	1	2 bits de parada para palabras de 6, 7 o 8 bits. 1.5 bits de parada para palabras de 5 bits.		
bits 0 y 1	bit 1	bit 0	Longitud de la palabra	
	0	0	5 bits	
	0	1	6 bits	
	1	0	7 bits	
	1	1	8 bits	

Tabla 2.18: Registro de control de línea

El registro de control de línea fija los parámetros básicos para la comunicación.

El bit 7 puesto a “1” permite el acceso al divisor (DLAB).

El bit 6 puesto a “1” habilita la terminación, esto hace que la línea de TD entre en estado de “espaciamiento” (“0” lógico), lo cual causa la terminación del UART de recepción.

Los bits 3, 4 y 5 permiten seleccionar la paridad. El bit 3 controla la paridad. Es decir, si se fija a “0” entonces no se utiliza ninguna paridad; si se fija a “1” entonces se utiliza la paridad. El bit 5, controla la paridad de Marca (“1” lógico) y de Espacio (“0” lógico), en este caso el bit de paridad siempre es enviado como un

“1” o un “0” según corresponda, este no es un buen método para comprobar si hay errores de paridad. Fijando el bit 5 a “0” y manteniendo el bit 3 en “1” tendremos la paridad normal, el bit de paridad puede ser “1” o “0”; por ejemplo en el caso de la paridad impar si el número de 1s incluyendo el bit de paridad es impar entonces el bit de paridad es “1” y si el número de 1s incluyendo el bit de paridad es par entonces el bit de paridad es “0”; lo mismo sucede para la paridad par.

El bit 2 fija la longitud de los bits de parada; cuando este bit se fija a “0” producirá un bit de parada; si se fija a “1” producirá 1.5 o 2 bits de parada dependiendo de la longitud de la palabra.

Los bits 0 y 1 fijan la longitud de la palabra según se observa en la Tabla 2.18.

f) Registro de control del módem (MCR)

bit	Notas
bit 7	Reservado
bit 6	Reservado
bit 5	Habilita el control de flujo automático (solamente en el 16750)
bit 4	Modo de Realimentación
bit 3	Salida Aux. 2
bit 2	Salida Aux. 1
bit 1	Obliga Petición para enviar
bit 0	Obliga Terminal de datos listo

Tabla 2.19: Registro de control del módem

El registro de control del módem es de lectura/escritura.

El bit 4 activa el modo de realimentación. En este modo la salida serial del transmisor se pone en el estado de marca (“1” lógico), la entrada serial del receptor se desconecta y la salida del transmisor se reconecta a la entrada del receptor. DSR, CTS, RI y DCD están desconectados DTR, RTS, OUT1 y OUT2 están conectados a las entradas de control del módem. Los pines de salida de control del módem se ponen en un estado inactivo. En este modo cualquier dato que se ponga en el buffer transmisor para salida es recibido por el circuito del receptor en el mismo chip, quedando disponible en el buffer del receptor. Esto es útil para probar la operación de los UARTs.

La salida Aux. 2 puede ser conectada con un circuito externo que controla el proceso de interrupción del UART.

La salida Aux. 1 normalmente está desconectada.

Los bits 0 y 1 simplemente controlan sus líneas respectivas. Por ejemplo fijando el bit 1 a “1” se activa la línea de petición para enviar.

g) Registro de estado de línea (LSR)

bit	Notas
bit 7	Error en el buffer de Recepción
bit 6	Registros de datos vacíos
bit 5	Registro de transmisión vacío
bit 4	Terminar Interrupción
bit 3	Error de estructura
bit 2	Error de paridad
bit 1	Error de sobrecarga
bit 0	Datos listos

Tabla 2.20: Registro de estado de Línea

El registro de estado de línea es solo de lectura.

El bit 7 es el bit de error en el FIFO de recepción. Este bit es “1” cuando por lo menos ocurra uno de los errores dados por los bits 3, 2, 1, y 0.

Cuando el bit 6 es “1”, el registro transmisión y el registro desplazamiento están vacíos. El registro de desplazamiento se utiliza para convertir el Byte (paralelo) a serial.

Cuando el bit 5 es “1”, sólo el registro de transmisión está vacío.

La terminación de interrupción bit 4 en “1”, ocurre cuando la línea de recepción de datos permanece en el estado lógico “0” por más tiempo del que toma enviar una palabra completa, eso incluye el tiempo para el bit de inicio, los bits de datos, los bits de paridad y los bits de parada.

El error de estructura bit 3 en “1”, ocurre cuando el último bit no es un bit de parada. Esto puede ocurrir debido a un error en la sincronización.

Un error de paridad pone el bit 2 en “1”.

El error de sobrecarga bit 1 en “1”, ocurre normalmente cuando el programa no puede leer el puerto lo suficientemente rápido. Si un Byte en el registro entrante no se saca lo suficientemente rápido antes que otro Byte sea recibido el último Byte se perderá y ocurrirá un error de sobrecarga.

El bit 0 en “1”, indica datos listos, significa que un Byte ha sido recibido por el UART y está en el buffer de recepción, listo para ser leído.

h) Registro de estado del módem (MSR)

bit	Notas
bit 7	Detección del portador
bit 6	Indicador de tono
bit 5	Datos listos
bit 4	Libre para enviar
bit 3	Detección de cambio del portador de datos
bit 2	Indicador de tono final
bit 1	Cambio en dato listo
bit 0	Cambio en libre para enviar

Tabla 2.21: Registro de estado del Modem

El bit 0 muestra un cambio en libre para enviar, significa que hubo un cambio en la línea libre para enviar, lo mismo ocurre con los bits 1 y 3.

El bit 2 es el indicador de tono final, el cual muestra que ha habido un cambio del estado "0" al estado "1" en la línea de indicador de tono.

Los bits del 4 al 7 muestran el estado actual de las líneas de datos; el bit 7 muestra la detección del portador, el bit 6 muestra el indicador de tono, el bit 5 muestra el dato listo y el bit 4 muestra el estado de la línea libre para enviar.

i) Registro de garabato

El registro de garabato no se utiliza para las comunicaciones solo se utiliza para dejar un Byte de datos, con el fin determinar si el UART es un 8250/8250B o un 8250A/16450; actualmente no es muy práctico.

2.2.5.6 Programación del Puerto Serial

Hay dos métodos para hacer un programa de comunicaciones con el UART:

- Interrogar al UART, para ver si hay datos disponibles. Este método es muy lento y hace uso intensivo del CPU, no se recomienda.
- Instalar un controlador de interrupciones para tomar los datos del UART cuando estos generen una interrupción. Este es el método que se ha usado en el presente trabajo.

El primer paso para usar el método de interrupciones es conocer cuales son los servicios de interrupción del puerto serial. La tabla 13 muestra las direcciones base y los IRQ (Interrupt ReQuest), peticiones de interrupción de algunos de los puertos más comunes. Los IRQ 3 y 4 son los más comúnmente usados, los IRQ 5 y 7 se utilizan a veces.

a) Vectores de interrupción

Los procesadores de la familia 8086 tienen un conjunto de 256 vectores de interrupción. Cada uno de estos vectores contiene un código de 4 Bytes el cual es una dirección del ISR (rutina del servicio de interrupción). Afortunadamente, lenguajes de programación como C o Turbo Pascal, se encargan del direccionamiento, solo se necesita conocer el vector real de interrupción. La tabla 2.22 muestra solo las interrupciones que se asocian a los IRQ.

INT(hex.)	IRQ	Aplicaciones comunes
08	0	Reloj del sistema
09	1	Teclado
0A	2	Redirigir
0B	3	Puerto Serial. COM2/COM4
0C	4	Puerto Serial. COM1/COM3
0D	5	Reservado tarjeta de sonido
0E	6	Controlador de disquetera
0F	7	Puerto Paralelo.
70	8	Reloj de tiempo real
71	9	Reservado
72	10	Reservado
73	11	Reservado
74	12	Mouse PS/2
75	13	Co-Procesador matemático
76	14	Disco duro
77	15	Reservado

Tabla 2.22: Vectores de interrupción.

En el presente trabajo se empleará el puerto COM1 que tiene un IRQ de 4, cuyo vector de interrupción es 0C en formato hexadecimal. Usando el Turbo Pascal como lenguaje de programación el vector de interrupción se instala usando el comando `SetIntVect($0C, @DatoListo)`, donde `DatoListo` es un procedimiento que contiene las instrucciones que se ejecutaran al ocurrir la interrupción. Pero, primero se debe salvar el vector de interrupción existente, para poder ser restaurado al finalizar el programa. Para salvar el vector de interrupción se usa el comando `GetIntVec($0C, IntCom1)`, donde `IntCom1` es una variable de tipo puntero.

b) Rutina de servicio de interrupciones (ISR)

El procedimiento `DatoListo` es el manejador de interrupciones, llamado también rutina de servicio de interrupciones (ISR). En este procedimiento se pondrá las

instrucciones que se ejecutaran cuando ocurre la interrupción. El procedimiento implementado para leer los datos que envía el multímetro es:

```

Procedure DatoListo; Interrupt;
Begin
  Dato[NumCar] := Char(Port[Base]);
  Inc(NumCar,1);
  If (NumCar = 15) or (Dato[NumCar-1] = #13) Then
    NumCar := 1;
    Port[$20] := $20;

```

End:

El procedimiento anterior se encarga de tomar los caracteres que llegan al puerto y los almacena en la variable Dato[NumCar], el cual es un array de caracteres; NumCar es un contador que verifica el número de caracteres enviado por el multímetro, este envía los caracteres en un formato de 14 Bytes, siendo el Byte 14 un CR (retorno de carro o enter); por lo tanto cuando NumCar = 15 o el Dato[14] = #13 (CR); NumCar es reiniciado a “1” para poder tomar una nueva lectura.

La penúltima línea del procedimiento Port[\$20] := \$20, le dice al controlador programable de interrupciones que la interrupción ha terminado.

c) El Controlador Programable de Interrupciones (PIC).

El controlador programable de interrupciones maneja las interrupciones del hardware. La mayoría de las PC tiene dos de ellos ubicados en diferentes direcciones. Uno controla los IRQ del 0 al 7 y el otro los IRQ del 8 al 15. Los puertos de comunicación serial residen en los IRQ 3 y 4, por lo tanto se usa el PIC1, que está situado en la dirección 20h.

bit	Deshabilitar IRQ	Función
7	IRQ7	Puerto paralelo
6	IRQ6	Controlador de Disquetera
5	IRQ5	Tarjeta de sonido
4	IRQ4	Puerto serial COM1/COM3
3	IRQ3	Puerto serial COM2/COM4
2	IRQ2	PIC2
1	IRQ1	Teclado
0	IRQ0	Reloj del sistema

Tabla 2.23: Palabra de control para el PIC1 (21h)

La mayor parte de la inicialización del PIC es hecha por la BIOS. Solo se necesita saber dos instrucciones; como habilitar y como enmascarar (deshabilitar) una IRQ en particular. La tabla 15 muestra los bits correspondientes para deshabilitar una IRQ, cuando un bit se pone en “1” significa que la IRQ correspondiente esta enmascarada (deshabilitada) y cuando este bit se pone en “0” significa que la IRQ esta habilitada. El puerto 21h es la dirección de la palabra de control del PIC1 es decir a este puerto es donde se envía el número (palabra de control) para habilitar o enmascarar una IRQ.

Para habilitar la IRQ4 correspondiente al puerto serial COM1 se debe poner el bit 4 en “0” y si todos los demás bits se ponen en “1” lo cual significa deshabilitar las IRQ 0,1,2,3,5,6,7; el número resultante que se envía a Port[\$21] será EFh. Pero, si alguna de las IRQ ya estaba habilitada, entonces primero se tiene que conocer el estado del PIC1, para ello primero se debe leer el valor que hay en el Port[\$21] y luego realizar la operación lógica and con EFh para no alterar el estado inicial del PIC1. El código que hace posible esto es:

```
Port[$21] := Port[$21] and $EF;
```

Para enmascarar (deshabilitar) la IRQ4 se debe poner el bit 4 en “1” y todos los demás bits se ponen en “0”; el número resultante que se envía a Port[\$21] será 10h; al cual, en este caso hay que realizar la operación lógica or con el estado actual del puerto. El código que hace posible esto es:

```
Port[$21] := Port[$21] or $10;
```

d) Configuración del UART

Finalmente la parte más importante para la comunicación del multímetro con la PC es la configuración del UART. Para ello se debe enviar ciertos números (palabras de control) a los diferentes registros del UART, la dirección base del puerto serial COM1 es 3F8h, la cual se ha declarado como una constante llamada Base en el programa.

La primera instrucción debe ser para deshabilitar la generación de interrupción en el UART, de tal forma que su inicialización no sea interrumpida por el UART. Esto se logra con la siguiente sentencia:

```
Port[Base+1] := 0.
```

Luego se salva el vector de interrupción existente y se instala el nuevo vector de interrupción con las siguientes sentencias:

```
GetIntVec($0C, IntCom1);
```

```
SetIntVect($0C, @DatoListo);
```

El paso siguiente es fijar la velocidad a la cual se va establecer la comunicación entre la PC y el multímetro que en este caso envía datos a una velocidad de 1200 BPS; para ello tenemos que fijar en “1” el bit 7 (el DLAB) del registro de control de línea (LCR) en Port[Base+3], para tener acceso a los Bytes superior e inferior del divisor de cierre; como podemos observar en la tabla 6 para una velocidad de 1200 BPS corresponde un divisor de 96, por lo tanto el Byte superior del divisor de cierre será 00h en Port[Base+1] y el Byte inferior del divisor de cierre será 60h en Port[Base+0]. El código que hace posible esto es:

```
Port[Base+3] := $80;
```

```
Port[Base+0] := $60;
```

```
Port[Base+1] := $00;
```

El paso siguiente es deshabilitar el bit de acceso al divisor de cierre (bit 7 de LCR en “0”) de esta forma se tendrá acceso al registro de habilitación de interrupción y a los buffers de recepción y transmisión; a la vez podemos fijar los siguientes parámetros de comunicación: los bits de paridad, el bit de parada y la longitud de la palabra; puesto que el multímetro en este caso transmite los datos con una longitud de palabra de 7 bits, 2 bits de parada y ninguna paridad, por lo tanto el número que se envía al registro de control de línea (LCR) en Port[Base+3] será 06h (binario 00000110); que a la vez deshabilita el DLAB. El código que hace posible esto es:

```
Port[Base+3] := $06;
```

Lo siguiente será habilitar los buffers recepción y transmisión poniendo en “1” el bit 0 del registro de control de FIFO (FCR), la PC que usamos es del tipo 80386 cuyo UART solo puede recibir o transmitir un Byte a la vez, por lo tanto fijamos el nivel de disparo de la interrupción en 1 Byte, entonces ponemos en “1” los bits 6 y 7 de este registro; también se debe limpiar los buffers de recepción y transmisión, para ello fijamos en “1” los bits 1 y 2; Esto elimina cualquier desperdicio que algún otro programa haya dejado; entonces el número que debemos enviar al registro de control de FIFO (FCR) en Port[Base+2] será 07h (binario 00000111). El código que hace posible esto es:

```
Port[Base+2] := $07;
```

Luego, como en este caso solo se reciben datos del multímetro entonces se debe fijar en “1” el bit 0 del registro de control de modem (MCR), algunas tarjetas requieren que la salida Aux 2 esté activa para atender las peticiones de interrupción por lo tanto se debe fijar en “1” el bit 3 del registro de control de modem (MCR), entonces el número que debemos enviar al registro de control de modem (MCR) en Port[Base+4] será 09h (binario 00001001). El código que hace posible esto es:

```
Port[Base+4] := $09;
```

Finalmente se tiene que instalar la interrupción del PIC1 y habilitar la generación de interrupción en el UART, cuando un nuevo dato este disponible, Es conveniente hacer esto al final de tal forma que la inicialización del UART no sea interrumpida.

El código que hace posible esto es:

```
Port[$21] := Port[$21] and $EF;
```

```
Port[Base+1] := $01;
```

e) Procedimiento Principal

Este procedimiento, se encarga de leer permanentemente, los datos que llegan al puerto, y los almacena en la variable lectura, hasta completar la cadena completa de caracteres enviados por el multímetro para cierta medición (en nuestro caso, voltaje). El código que hace posible esto es:

```
Procedure Leer_Dato;
```

```
Begin
```

```
Repeat Lectura := Dato[1]+Dato[2]+ ..... +Dato[13];
```

```
Until (NumCar = 14);
```

```
End;
```

En el apéndice A, se adjunta, el listado completo del programa, el cual se encarga de lo siguiente:

- Controlar el giro del motor a pasos.
- Leer los datos enviados por el multímetro (voltaje).
- Graficar dichos datos en función del ángulo de incidencia del haz láser.
- Almacenar dichos datos en un archivo, para su posterior tratamiento.

2.2.6 Montaje de los componentes sobre una plataforma

La guía óptica, el prisma de acople y el sensor están montados sobre una plataforma giratoria, la plataforma es una pieza circular de acrílico, montado sobre un eje vertical (se ha usado un goniómetro disponible). El acople de la plataforma con el motor a pasos se hace por medio de unos engranajes. Para lograr los engranajes de la plataforma, se ha pegado sobre su borde una banda dentada, como la que se usa en las impresoras. El motor a pasos tiene sobre su eje un pequeño engranaje que encaja sobre la banda dentada, El ángulo que gira la plataforma, por cada paso que da el motor, es de $0,033^\circ \pm 0,001^\circ$. Para determinar este valor, se midió varias veces el ángulo girado por la plataforma para determinado número de pasos.

En la figura 2.13 se muestra el montaje de los componentes sobre la plataforma: en el centro de la plataforma se ubica el prisma de acople, el cual se ha montado sobre un soporte de acrílico; para fijar la guía óptica, la cual es fijada sobre un sustrato de vidrio, se dispone de otro soporte de acrílico que tiene incrustado un perno móvil hecho de teflón; el sensor está ubicado dentro de un tubo plástico el cual se encuentra dentro de una caja de acrílico que tiene tres pernos que permiten regular la posición del tubo.

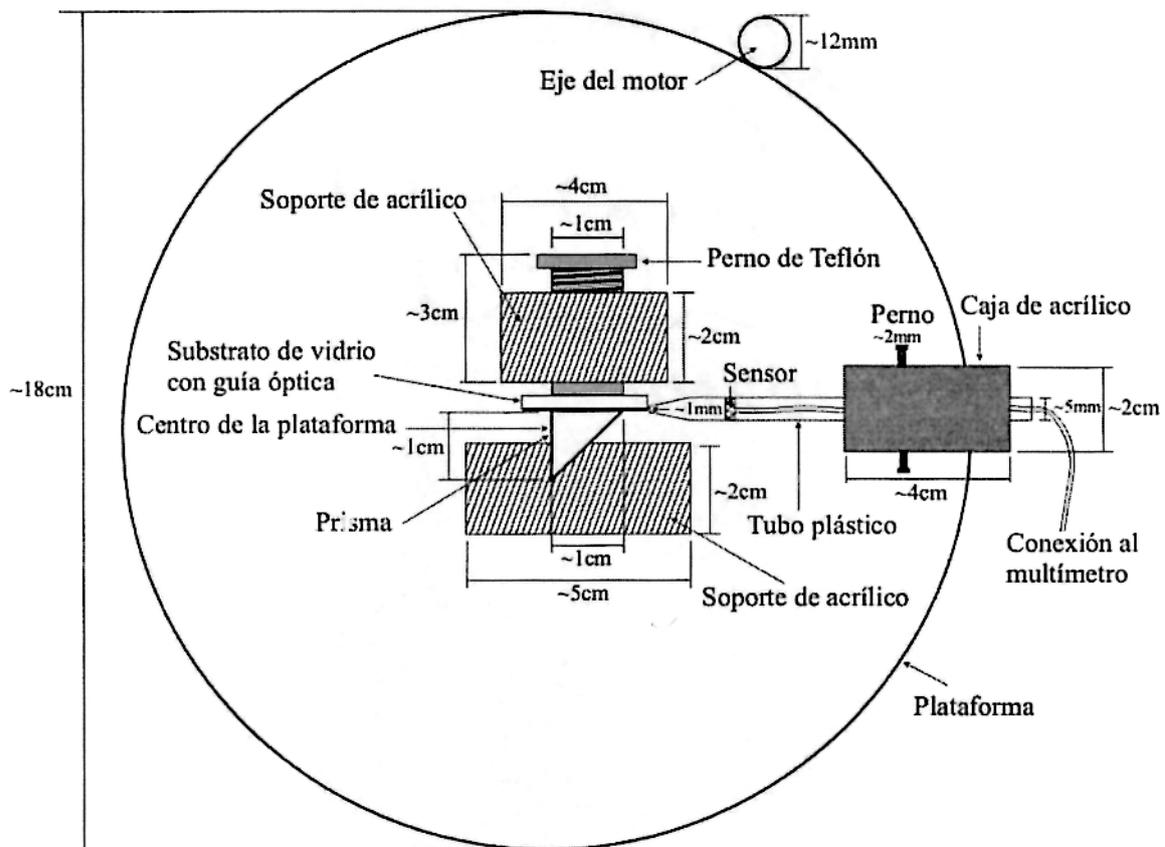


Figura 2.13: Montaje de los componentes sobre una plataforma

A continuación, en la figura 2.14 se muestra una fotografía del equipo completo, en plena operación.

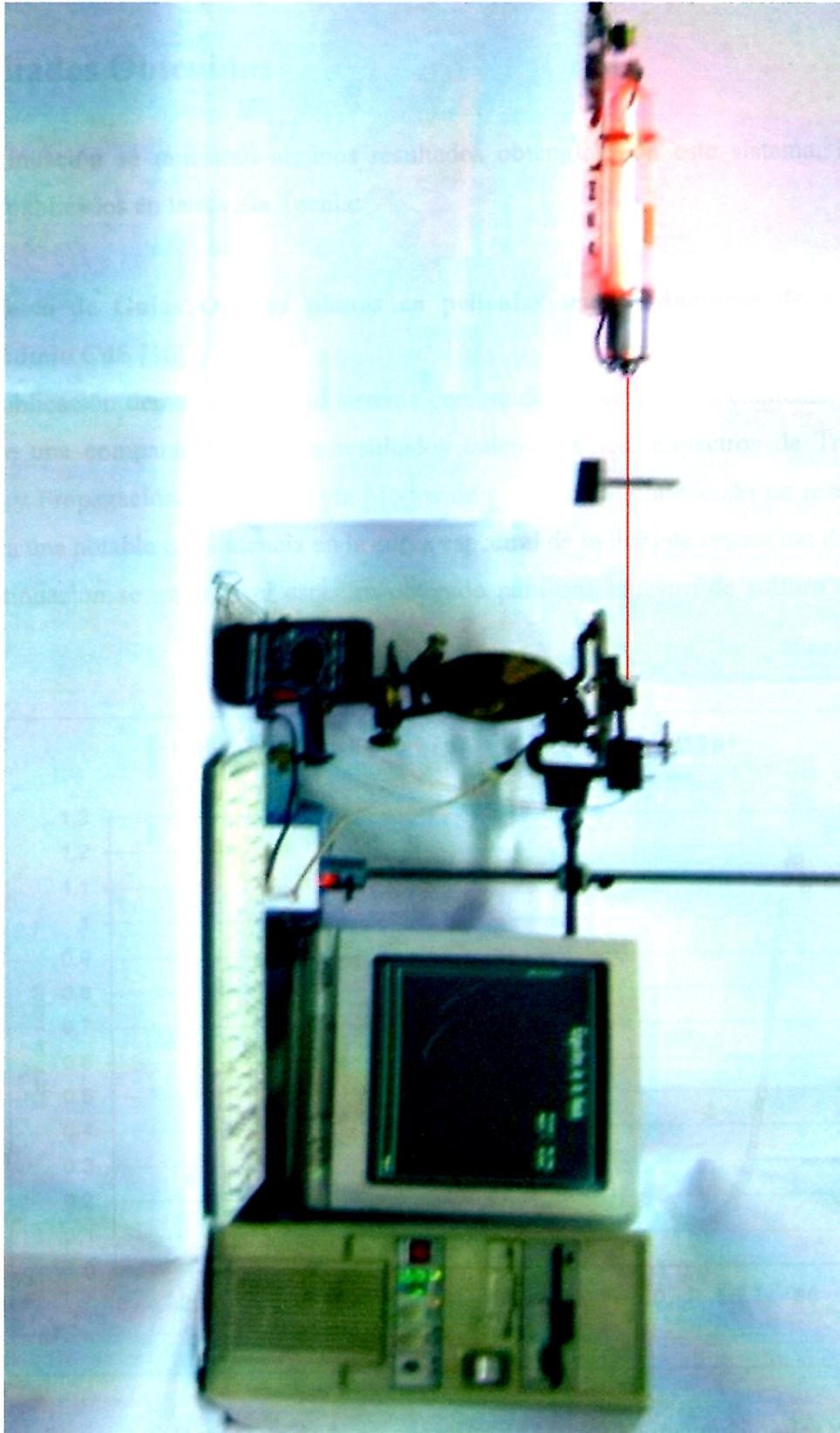


Figura 2.14: Fotografía del equipo en plena operación

Resultados Obtenidos

A continuación se muestran algunos resultados obtenidos con este sistema, los cuales fueron publicados en la revista Tecnia:

3.1 Efecto de Guías Ópticas planas en películas semiconductoras de Sulfuro de Cadmio CdS [10]

Esta publicación demuestra, que el sistema construido, funciona perfectamente, pues aquí se hace una comparación de dos resultados independientes: Espectros de Transmisión frontal y Propagación transversal vía Modos de guía óptica, obteniendo un resultado que muestra una notable coincidencia en la curva espectral de índices de refracción de CdS.

A continuación se muestra el espectro obtenido para una muestra de sulfuro de cadmio (CdS).

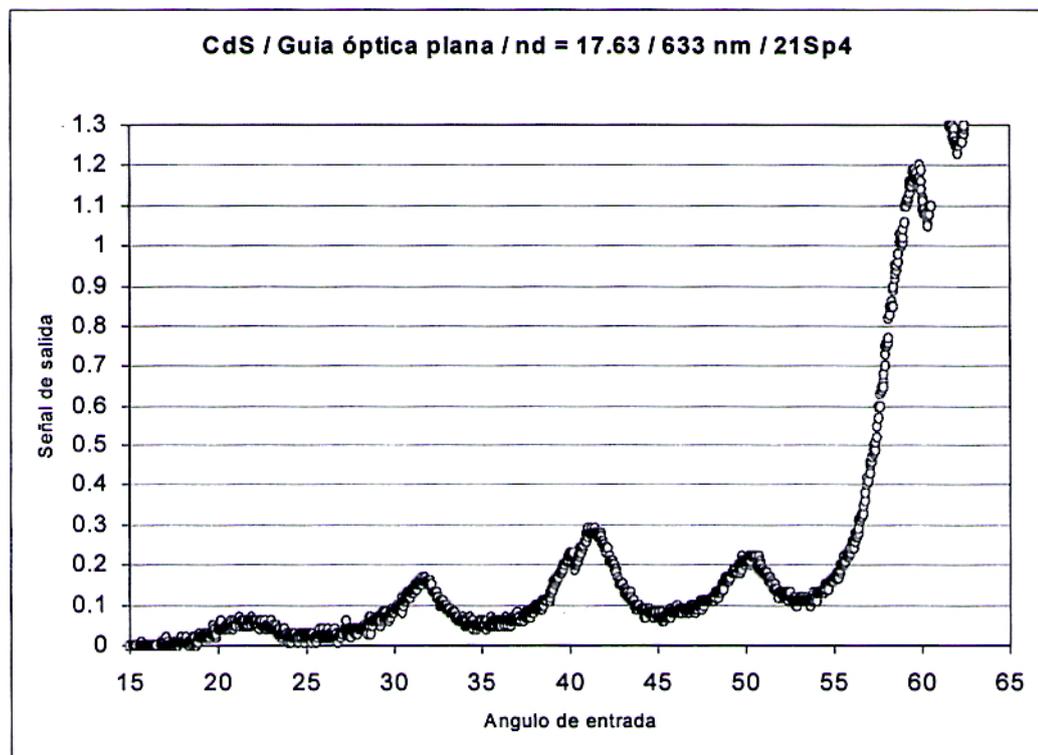


Figura 3.1: Respuesta de la propagación de Modos en una guía óptica plana semiconductoras de CdS (Espesor $7.33 \mu\text{m}$) para $\lambda = 633 \text{ nm}$.

En el espectro de modos mostrado en la figura 3.1, se observa 5 máximos relativos, correspondientes a los ángulos de incidencia (α) que se indican en la tabla 3.1.

Modo	θ_{modo}	$\alpha_{\text{teo.}}$	$\alpha_{\text{exp.}}$
37	42.17	6.13	No Obs.
38	43.54	22.71	21.2
39	44.94	32.59	32.5
40	46.37	41.22	41.9
41	47.83	49.69	49.8
42	49.32	58.77	59.4

Tabla 3.1: Ángulos y modos correspondientes al espectro de la figura 1.1

Los 5 ángulos de incidencia experimentales son introducidos en la ecuación de resonancia (ecuación 1.1), obteniéndose como resultado, que los modos observados corresponden a los Modos absolutos 38, 39, 40, 41 y 42 de una película de espesor $d = 7.333 \mu\text{m}$ y de índice de refracción $n_{633} = 2.404$. En base a lo cual se recalcula todos los Modos posibles de ser observados con la geometría disponible y los correspondientes ángulos de ingreso teóricos de esta guía de onda. Del conjunto de ángulos teóricos factibles de ser observados se comprueba una buena concordancia con los valores experimentales, a excepción del modo 37 que no se observa en el espectro, probablemente por una débil respuesta.

3.2 Determinación experimental de índices de refracción en películas semiconductoras: CdS, Ti₃O₅ [11]

El espectro que se muestra a continuación corresponde a una muestra de Ti₃O₅.

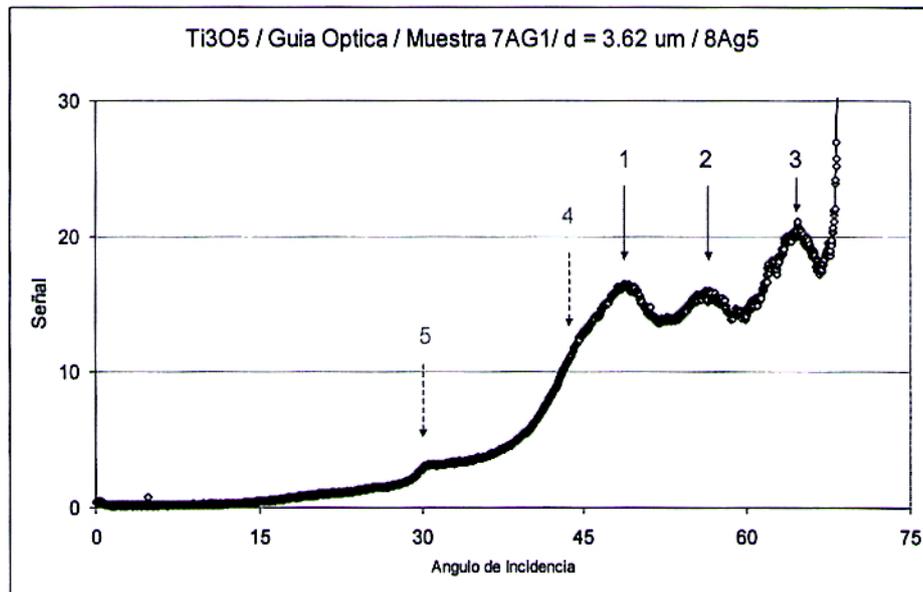


Figura 3.2: Respuesta de la propagación de Modos en una guía óptica plana semiconductoras de Ti₃O₅ (Espesor 3.62 μm) para $\lambda = 633$ nm.

En el espectro de modos mostrado en la figura 3.2, se observa 5 máximos relativos, correspondientes a los ángulos de incidencia (α) que se indican en la tabla 3.3

Modo	$\alpha_{teo.}$	$\alpha_{exp.}$
0	22.36	No Obs.
1	24.01	No Obs.
2	26.59	No Obs.
3	29.93	30
4	33.91	No Obs.
5	38.49	No Obs.
6	43.66	44
7	49.56	49.5
8	56.42	56.6
9	64.88	64.8

Tabla 3.2: Ángulos y modos correspondientes al espectro de la figura 3.2

Los ángulos de incidencia experimentales correspondientes a los 3 últimos modos, se introducen en la ecuación de resonancia (ecuación 1.1), como resultado, se obtiene, que estos tres modos corresponden a los Modos absolutos 7, 8 y 9 de una película de espesor $d = 3.606$ μm y de índice de refracción $n_{633} = 1.746$. En base a esta última determinación se recalcula todos los Modos posibles de esta guía de onda.

Conclusiones y Trabajo Futuro

4.1 Conclusiones

Podemos mencionar los siguientes logros:

- a) El equipo de detección de modos en guías ópticas ha sido automatizado.
- b) La sensibilidad se ha incrementado notablemente pues el operador no interfiere con las señales.
- c) La resolución angular que se ha logrado en el giro de la plataforma es de $0,033^\circ$, esto nos da una gran precisión, comparada con los 5° que se lograba manualmente.
- d) Actualmente, se realiza trabajos de investigación en guías ópticas semiconductoras; los resultados de algunos de estos trabajos (efecto de guías ópticas planas en películas semiconductoras de sulfuro de cadmio CdS [10] y determinación experimental de índices de refracción en películas semiconductoras: CdS, Ti_3O_5 [11]), que fueron publicados en la revista TECNIA, se han incorporado en el capítulo 3.

4.2 Trabajo Futuro

El equipo puede ser mejorado, algunas de estas mejoras incluyen:

- a) El sistema mecánico; el acople del motor con la plataforma es muy delicado, debido a que son unidades separadas. Para mejorar esto podría construirse una sola unidad para la plataforma y el motor.
- b) El software puede mejorarse haciendo la programación en un lenguaje visual, por ejemplo: Visual C++ o LabView; para ello será necesario una computadora con la suficiente capacidad para trabajar con estos lenguajes.

Referencias

- [1] Tamir, T. "Integrated Optics"
Springer Verlag, Berlin 1982.
- [2] L. Mosquera, R. Ramos, E. Fernandez, G. Castillo "Optica Integrada: Elaboración y Caracterización de Guías de Onda Planas"
TECNIA, Vol. 9 N°1, págs. 25 - 29, 1999.
- [3] G. Castillo, R. Ramos, E. Fernandez, A. Valera "Sistema Automatizado de Detección de Modos de Propagación en Guías Ópticas"
TECNIA, Vol. 9 N°2, págs. 47 - 51, 1999.
- [4] <http://www.todorobot.com.ar/informacion/informacion.htm>
- [5] <http://www.modelo.edu.mx/univ/virtech/circuito/paralelo.htm>
- [6] <http://www.beyondlogic.org/serial/serial.htm>
- [7] http://www.lammertbies.nl/comm/info/RS-232_uart.html
- [8] <http://www.zator.com/Hardware/index.htm>
- [9] <http://www.hardwarebook.net/index.html>
- [10] D. Cachi, A. Valera "Efecto de Guías Ópticas planas en películas semiconductoras de Sulfuro de Cadmio: CdS"
TECNIA, Vol. 12 N°1, págs. 95 - 105, 2002.
- [11] S. Rodríguez, O. Miculicich, C. Oliva, A. Fiorentini, A. Valera "Determinación experimental de índices de refracción en películas semiconductoras: CdS, Ti₃O₅"
TECNIA, Vol. 11 N°2, págs. 3 - 12, 2001.