

INSTITUTO TECNOLÓGICO DE TUXTLA GUTIERREZ

REPORTE FINAL DE RESIDENCIA

NOMBRE DEL PROYECTO:

“Sistema de conteo y registro de datos del producto terminado y subproducto en la Harinera de Chiapas S. A. de C. V., implementado con sensores ópticos.”

LUGAR DE RESIDENCIA:

HARINERA DE CHIAPAS S. A. DE C. V.

ASESOR EXTERNO:

ING. PAUL PINEDA DE LOS SANTOS

ASESOR INTERNO:

ING. RIGOBERTO JIMENEZ JONAPA

REVISOR:

DR. JOSÉ LUIS CAMAS ANZUETO

RESIDENTE:

**MOGUEL CAMACHO JUAN ANTONIO
INGENIERIA ELECTRONICA
9º SEMESTRE
04270397**

TUXTLA GUTIERREZ, CHIAPAS; DICIEMBRE DE 2008.

INDICE

1.- INTRODUCCIÓN	1
2.- GENERALES	2
2.1 JUSTIFICACIÓN	2
2.2 OBJETIVO	2
2.2.1 GENERAL	3
2.2.2 ESPECIFICO	3
2.3 DELIMITACION DEL PROBLEMA	3
3.- FUNDAMENTO TEORICO	4
3.1 ETAPA DE SENSADO	4
3.1.1 EMISOR (LASER)	5
3.1.2 RECEPTOR (FOTORRESISTENCIAS)	7
3.2 MICROCONTROLADOR	12
3.2.1 APLICACIONES	14
3.2.2 PIC 16F877	15
3.3. PUERTO SERIAL	22
3.3.1 TIPOS DE COMUNICACIÓN SERIAL	23
3.3.2 LA NORMA RS-232	25
3.3.3 EL CIRCUITO MAX-232	26
3.3.4 ACCESO AL PTO SERIAL A TRAVES DE V. B.	27
3.3.5 APLICACIÓN	30
3.4 VISUAL BASIC	31
4.- DESARROLLO	32
4.1 INVESTIGACION DOCUMENTAL DE LOS SENSORES ...	32
4.2 INVESTIGACION DOCUMENTAL DE OTRO SENSOR ...	33
4.3 IMPLEMENTACION DEL SENSOR LASER	35
4.4 IMPLEMENTACION DEL PROGRAMA EN EL Uc	36
4.5 DISEÑO E IMPLEMENTACION DE LA TARJETA	41
4.6 IMPLEMENTACION DEL PROGRAMA EN V. B.	42
5.- PRUEBAS Y RESULTADOS	52
5.1 PRUEBAS CON LOS CIRCUITOS DE LOS SENSORES ENCONTRADOS.....	52
5.2 IMPLEMENTACION DEL SENSOR INFRARROJO	54
5.3 PRUEBAS CON EL SENSOR INFRARROJO	55
5.4 PRUEBAS CON EL PROGRAMA HECHO EN MIKROC ...	58
5.5 IMPLEMENTACION DE LA TARJETA DE ADQUISCION DE DATOS	59
5.6.- PRUEBAS CON EL PROGRAMA EN VISUAL BASIC	60
6.- CONCLUSIONES	64

7.- BIBLIOGRAFÍA	65
8.- ANEXOS	65
8.1 DATOS GENERALES DE LA EMPRESA.....	65
8.2 BREVE DESCRIPCION DE LA EMPRESA	67
8.3 HOJA DE DATOS DEL CIRCUITO LM555	71
8.4 HOJA DE DATOS DEL CIRCUITO 2N2222	72
8.5 HOJA DE DATOS DEL CIRCUITO BC558	73
8.6 HOJA DE DATOS DEL CIRCUITO MAX 232	74
8.7 VALORES DE LOS CARACTERES	75

1.- INTRODUCCIÓN

En este trabajo se presenta el desarrollo de un sistema de conteo automático del producto terminado (sacos) dentro de una empresa productora y comercializadora de harina y sus derivados que se elaboró de la siguiente manera:

Se diseño, implemento e instalo en la empresa aquí mencionada específicamente en el área de la bodega, un sensor láser en la banda transportadora que al momento en que es interrumpido por el paso del producto envasado en sacos manda un nivel alto de voltaje (5 volts).

Esta señal es recibida por una tarjeta de adquisición de datos conformada principalmente por un microcontrolador que lo recibe y lo almacena en su memoria EEprom interna. Para después enviarlo a través del puerto serial (usando el RS 232) a la computadora ubicada en la oficina del encargado de la bodega donde se cargan y descargan los camiones.

Este dato se envía a un programa hecho en Visual Basic que permite al usuario llevar una cuenta exacta de los sacos que atraviesan por la banda dicho sensor, así como la posibilidad de guardar en bloc de notas y/o documentos de Word dichos datos para su utilización posterior ya sea para tener un historial de las ventas que salen o generar una impresión a manera de comprobante para el cliente o empresa que compra los productos de la empresa.

Aquí se describe detalladamente y por partes como se logró hacer cada una de las partes e implementar todo el sistema, pudiendo ser de gran utilidad

para cualquiera que se interese en aprender acerca de sensores ópticos, microcontroladores, comunicación serial y/o programación en Visual Basic.

2.- GENERALES

2.1 JUSTIFICACIÓN

La implementación de este sistema surge de la necesidad de establecer un conteo en la producción de la harina de trigo y los subproductos (en costales) de manera segura y sin posibilidad de errores; ya que hasta este momento el conteo se realiza de manera manual por los trabajadores de esa área.

Otra de las problemáticas que se presentan es que en algunas ocasiones el cliente no queda conforme con el abastecimiento del producto o subproducto, argumentando que se le proporciono una cantidad menor a la cobrada. Ó por el contrario se ha dado el caso de que los trabajadores den más producto del acordado lo que representa una perdida para la empresa.

De esta manera este sistema reducirá en gran forma esta problemática entre los clientes y la empresa, brindando un mejor servicio a sus compradores. Teniendo también un control en la venta diaria de costales y poder contabilizarlo y guardarlo en una base de datos de manera segura.

2.2 OBJETIVO

2.2.1 GENERAL

Diseñar e implementar un sistema de conteo y registro de producto en costales que son vendidos diariamente, por medio de un circuito sensor, una tarjeta de adquisición de datos y un programa en la computadora que manipulará el encargado de la bodega.

2.2.2 ESPECIFICO

- Diseñar e implementar un circuito capaz de detectar el paso de los sacos en la banda transportadora por medio de un haz de rayo láser (emisor) y fotorresistencias (receptor).
- Diseñar e implementar una tarjeta de captura de datos, que reciba la señal del sensor y que nos permita proteger la PC de los diferentes dispositivos que integran los procesos del proyecto.
- Implementar la comunicación entre el sistema contador de la bodega y la oficina de la misma a través del puerto serial de la computadora.
- Elaborar un programa capaz de almacenar y procesar los datos recibidos de la tarjeta, que sea sencillo, útil y con un lenguaje amigable y entendible para el usuario.

2.3 DELIMITACIÓN DEL PROBLEMA

El sistema podrá contabilizar todos los sacos que pasen a través de la banda transportadora de salida siempre y cuando este alimentado correctamente (5 volts); sin energía en el sensor y/o la tarjeta de adquisición de datos el conteo no podrá realizarse. .

Debido a que es una comunicación serial entre la tarjeta y la computadora el límite de envío del número de sacos es de 255, resolviéndose esto al enviar

datos menores a este número con un escaneo continuo entre la tarjeta y el programa en la computadora.

Los datos que lleguen a la computadora podrán ser vistos por el usuario solamente que este activo, es decir, si un proceso de carga se lleva a cabo mientras no esta abierto el programa puede llegar a rebasar la capacidad de envío del microcontrolador dando un dato erróneo.

Por lo tanto, es necesario contar con una computadora con el programa instalado y corriendo para poder recibir los datos de manera correcta y poder procesarlo.

3.- FUNDAMENTO TEORICO

3.1.- ETAPA DE SENSADO

SENSORES

Un sensor es un dispositivo capaz de transformar magnitudes físicas o químicas, llamadas variables de instrumentación, en magnitudes eléctricas. Las variables de instrumentación dependen del tipo de sensor y pueden ser por ejemplo temperatura, intensidad luminosa, distancia, aceleración, inclinación, desplazamiento, presión, fuerza, torsión, humedad, etc. Una magnitud eléctrica obtenida puede ser una resistencia eléctrica (como en una RTD), una capacidad eléctrica (como en un sensor de humedad), una tensión eléctrica (como en un termopar), una corriente eléctrica (como un fototransistor), etc.

3.1.1.- EMISOR (LASER)

Un láser es un aparato (o dispositivo) que produce un tipo muy especial de luz. Podemos imaginárnoslo como una superlinterna. Sin embargo, la luz procedente de un láser se diferencia de la de una linterna en cuatro aspectos básicos:

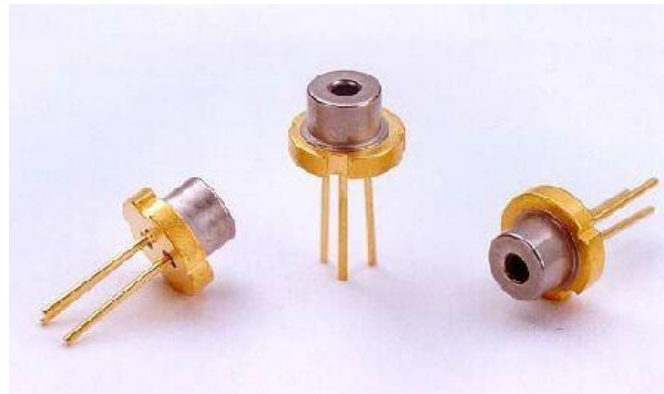


Figura 1. Apuntadores laser

1. La luz láser es intensa. No obstante, sólo ciertos láseres son potentes. Aunque lo parezca, no se trata de una contradicción. La intensidad es una medida de la potencia por unidad de superficie, e incluso los láseres que emiten sólo algunos milivatios son capaces de producir una elevada intensidad en un rayo de un milímetro de diámetro. En realidad, su intensidad puede ser igual a la de la luz del sol. Cualquier lámpara ordinaria emite una cantidad de luz muy superior a la de un pequeño láser, pero esparcida por toda la sala. Algunos láseres pueden producir muchos miles de vatios continuamente; otros son capaces de producir billones de vatios en un impulso cuya duración es tan sólo la mil millonésima parte de un segundo.

2. Los haces láser son estrechos y no se dispersan como los demás haces de luz. Esta cualidad se denomina direccionalidad. Se sabe que ni la luz de un potente foco logra desplazarse muy lejos: si se enfoca hacia el firmamento, su rayo parece desvanecerse de inmediato. El haz de luz comienza a esparcirse en el momento en que sale del foco, hasta alcanzar tal grado de dispersión que llega a perder su utilidad. Sin embargo, se han logrado reflejar haces láser de pocos vatios de potencia sobre la luna y su luz era todavía lo suficientemente brillante para verla desde la tierra. Uno de los primeros haces láser que se disparó contra la luna en 1962 sólo llegó a dispersarse cuatro kilómetros sobre la superficie lunar. ¡No está mal si se considera que se había desplazado cuatrocientos mil kilómetros!

3. La luz láser es coherente. Esto significa que todas las ondas luminosas procedentes de un láser se acoplan ordenadamente entre sí. Una luz corriente, como la procedente de una bombilla, genera ondas luminosas que comienzan en diferentes momentos y se desplazan en direcciones diversas. Algo parecido a lo que ocurre cuando se arroja un puñado de piedrecitas en un lago. Lo único que se crean son pequeñas salpicaduras y algunas ondulaciones. Ahora bien, si se arrojan las mismas piedrecitas una a una con una frecuencia exactamente regular y justo en el mismo sitio, puede generarse una ola en el agua de mayor magnitud. Así actúa un láser, y esta propiedad especial puede tener diversas utilidades. Dicho de otro modo, una bombilla o un foco son como escopetas de cartuchos, mientras que un láser equivale a una ametralladora.

4. Los láseres producen luz de un solo color, o para decirlo técnicamente, su luz es monocromática. La luz común contiene todos los colores de la luz visible (es decir, el espectro), que combinados se convierten en blanco. Los haces de luz láser han sido producidos en todos los colores del arco iris (si bien el más común es el rojo), y también en muchos tipos de luz invisible; pero un láser determinado sólo puede emitir única y exclusivamente un solo color. Existen láseres sintonizables que pueden ser ajustados para producir diversos colores, pero incluso éstos no pueden emitir más que un color único en un momento dado. Determinados láseres, pueden emitir varias frecuencias monocromáticas al mismo tiempo, pero no un espectro continuo que contenga todos los colores de la luz visible como pueda hacerlo una bombilla. Además, existen numerosos láseres que proyectan luz invisible, como la infrarroja y la ultravioleta

3.1.2.- RECEPTOR (FOTORRESISTENCIAS)

Las LDR (Light Dependent Resistor, o Resistor Dependiente de la Luz) son, como su nombre lo indica, resistencias cuyo valor varia de acuerdo al nivel de luz al que están expuestas.

Si bien los valores que puede tomar una LDR en total oscuridad y a plena luz puede variar un poco de un modelo a otro, en general oscilan entre unos 50 a 1000 ohmios (1K) cuando están iluminadas (por ejemplo, con luz solar) y valores comprendidos entre 50K (50,000 Ohms) y varios megohmios (millones de ohms) cuando está a oscuras. Desde el punto de vista constructivo, las LDR están fabricadas con materiales

de estructura cristalina, siendo los mas utilizados el sulfuro de cadmio y el seleniuro de cadmio, aprovechando sus propiedades fotoconductoras

Una cuestión a tener en cuenta cuando diseñamos circuitos que usan LDR es que su valor (en Ohmios) no variara de forma instantánea cuando se pase de estar expuesta a la luz a oscuridad, o viceversa, y el tiempo que se dura este proceso no siempre es igual si se pasa de oscuro a iluminado o si se pasa de iluminado a oscuro (se dice que muestra inercia a las variaciones de la intensidad luminosa). Igualmente, estos tiempos son cortos, generalmente del orden de una décima de segundo

Esto hace que el LDR no se pueda utilizar en algunas aplicaciones, concretamente en aquellas que necesitan de mucha exactitud en cuanto a tiempo para cambiar de y a exactitud de los valores de la fotorresistencia al estar en los mismos estados anteriores. Pero hay muchas aplicaciones en las que una fotorresistencia es muy útil. En casos en que la exactitud de los cambios no es importante

La primera figura es una imagen de uno de los tipos de LDR que existen en el mercado.

La figura siguiente nos muestra el símbolo utilizado para representar las LDR en los esquemas electrónicos, aunque a veces pueden ser ligeramente diferentes pero siempre tomando como base el símbolo de una resistencia común con alguna(s) flecha(s) que simbolizan la incidencia de la luz.



Figura 2. Un formato posible para las LDR



Figura 3. Símbolo de una LDR

Resulta un montaje muy conveniente para que al llegar a nuestra casa por la noche encontremos alguna luz encendida o también para quien no puede estar en determinado lugar para encender o apagar las luces al anochecer o amanecer. Además de evitar el gasto excesivo de electricidad, ya mantiene las luces encendidas sólo mientras falta luz natural. Otra aplicación para este circuito tiene que ver con la señalización, por ejemplo, de edificios altos o antenas, cuya baliza en la parte superior debe permanecer encendida todo el tiempo que dura la noche

En el circuito, cuyo esquema eléctrico vemos en la figura 5, se utiliza una

configuración poco frecuente en el TRIGGER (disparo) del integrado NE555. En los contactos del rele podremos conectar cualquier lámpara que no supere los 200W si estamos trabajando con una tensión de red de 110V o los 400W si estamos trabajando con 220V. Siempre es conveniente tener a mano la hoja de datos del rele en cuestión para ver cual es el máximo de corriente que soporta en sus contactos. Los valores que mencionábamos corresponden a un rele típico, pero nunca esta de más asegurarse.

Si necesitáramos conectar lámparas o dispositivos (por ejemplo, un cartel luminoso) con un consumo mayor, podemos colocar entre el rele y el dispositivo un contactor o rele de la potencia necesaria

Como podemos ver en la figura 5, el circuito integrado NE555 se utiliza tal lo visto en la nota específica sobre el, para configurar un disparador. Una característica importante de este circuito es el hecho de no sufrir esas desagradables oscilaciones de los circuitos convencionales cuando la iluminación llega al umbral de disparo. La transición del punto de espera al disparo es inmediata y única. El relé cierra y abre de inmediato los contactos, sin oscilación

Se aplica al pin 5 del NE555 una tensión de referencia, aproximadamente igual a la mitad de la tensión de alimentación. La tensión aplicada en el pin 2 depende del valor de la LDR, y cuando sea menor que la mitad de la tensión de alimentación (la del pin 5) se activara la salida (pin 3). En nuestro circuito, fijamos la tensión de referencia por un divisor formado por R2 y R3 y ajustamos el disparo en función de la luz que incide en el LDR a través del potenciómetro VR1. El ajuste se realiza en función de la luz ambiente,

ajustando el potenciómetro hasta que se dispare el circuito con la luz que nosotros queramos.

Por supuesto, se debe evitar a toda costa que las lámparas que estén activadas por este circuito iluminen el LDR, dado que se produciría un efecto de realimentación, produciendo oscilaciones o bien impidiendo el buen funcionamiento del mismo. Se puede poner la LDR dentro de un tubo opaco, apuntando al cielo, con una tapa transparente, de esta forma estará protegido de la luz de las lámparas y de la intemperie.

Los 12 voltios necesarios para alimentar el circuito se pueden obtener mediante un transformador, puente de diodos y un regulador de voltaje, o bien recurrir a alguna fuente de alimentación de 12v.

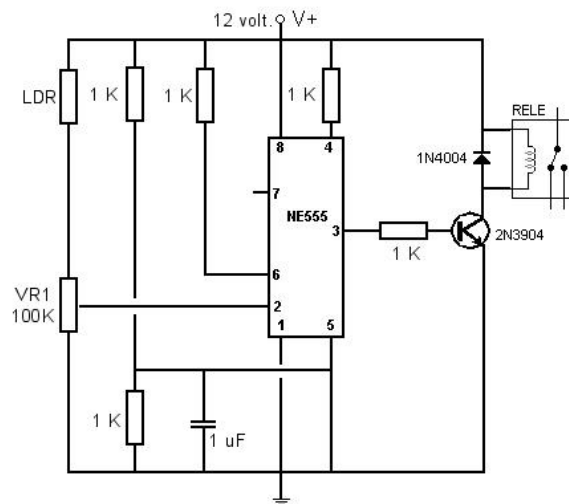


Figura 4. Circuito del interruptor crepuscular.

Los cables que conectan la LDR (que deberá estar en el exterior, para poder recibir la luz) a la placa pueden ser de 4 o 5 metros sin que haya ningún

problema.

En el momento del montaje, deberemos observar que la posición del NE555 sea la correcta, utilizar resistencias de $1/8$ o $1/4$ de watt (no es necesario mas), y que la(s) lámpara(s) utilizadas estén conectadas correctamente al rele. Este rele cumple la función del interruptor de las lámparas, así que deberá estar en serie con ellas y la fuente de alimentación. Las lámparas, entre si, se conectaran en paralelo.

Para probar el aparato conecte cualquier lámpara de acuerdo con su red local, un velador u otro aparato electrodoméstico cuyo funcionamiento se pueda verificar. Luego, coloque el LDR de modo que reciba directamente la luz ambiente y ajuste el VR1 para que el relé abra sus contactos. Ajuste el VR1 de modo que quede en el umbral de accionamiento. A continuación, cubriendo el LDR con la mano, se va moviendo gradualmente el cursor del potenciómetro hasta obtener el punto de mayor sensibilidad, o sea, en el que el circuito es conectado al faltar la luz y desconectado con la presencia de luz. Hay que mover lentamente el cursor del potenciómetro ya que la LDR, como dijimos antes, tiene una cierta inercia.

3.2.- MICROCONTROLADOR

Los microcontroladores están conquistando el mundo. Están presentes en nuestro trabajo, en nuestra casa y en nuestra vida, en general. Se pueden encontrar controlando el funcionamiento de los ratones y teclados de los computadores, en los teléfonos, en los hornos microondas y los televisores de nuestro hogar. Pero la invasión acaba de comenzar y el nacimiento del

siglo XXI será testigo de la conquista masiva de estos diminutos computadores, que gobernarán la mayor parte de los aparatos que fabricaremos y usamos los humanos.

Un microcontrolador es un circuito integrado de alta escala de integración que incorpora la mayor parte de los elementos que configuran un controlador.

Un microcontrolador dispone normalmente de los siguientes componentes:

- Procesador o UCP (Unidad Central de Proceso).
- Memoria RAM para Contener los datos.
- Memoria para el programa tipo ROM/PROM/EPROM.
- Líneas de E/S para comunicarse con el exterior.
- Diversos módulos para el control de periféricos (temporizadores, Puertas Serie y Paralelo, CAD: Conversores Analógico/Digital, CDA: Conversores Digital/Analógico, etc.).
- Generador de impulsos de reloj que sincronizan el funcionamiento de todo el sistema.

Los productos que para su regulación incorporan un microcontrolador disponen de las siguientes ventajas:

- Aumento de prestaciones: un mayor control sobre un determinado elemento representa una mejora considerable en el mismo.
- Aumento de la fiabilidad: al reemplazar el microcontrolador por un elevado número de elementos disminuye el riesgo de averías y se precisan menos ajustes.

- Reducción del tamaño en el producto acabado: La integración del microcontrolador en un chip disminuye el volumen, la mano de obra y los stocks.
- Mayor flexibilidad: las características de control están programadas por lo que su modificación sólo necesita cambios en el programa de instrucciones.

El microcontrolador es en definitiva un circuito integrado que incluye todos los componentes de un computador. Debido a su reducido tamaño es posible montar el controlador en el propio dispositivo al que gobierna. En este caso el controlador recibe el nombre de controlador empotrado (embedded controller).

3.2.1.- APLICACIONES

Cada vez existen más productos que incorporan un microcontrolador con el fin de aumentar sustancialmente sus prestaciones, reducir su tamaño y coste, mejorar su fiabilidad y disminuir el consumo.

Algunos fabricantes de microcontroladores superan el millón de unidades de un modelo determinado producidas en una semana. Este dato puede dar una idea de la masiva utilización de estos componentes.

Los microcontroladores están siendo empleados en multitud de sistemas presentes en nuestra vida diaria, como pueden ser juguetes, horno microondas, frigoríficos, televisores, computadoras, impresoras, módems, el sistema de arranque de nuestro coche, etc. Y otras aplicaciones con las que

seguramente no estaremos tan familiarizados como instrumentación electrónica, control de sistemas en una nave espacial, etc. Una aplicación típica podría emplear varios microcontroladores para controlar pequeñas partes del sistema. Estos pequeños controladores podrían comunicarse entre ellos y con un procesador central, probablemente más potente, para compartir la información y coordinar sus acciones, como, de hecho, ocurre ya habitualmente en cualquier PC.

3.2.2.- PIC 16F877

Características:

- Velocidad de operación: hasta 20 MHz de reloj
- 8K x 14 bits por palabra de memoria de programa FLASH
- 368 x 8 bytes de memoria de datos (RAM)
- 256 x 8 bytes de memoria de datos EEPROM 14 fuentes de interrupciones
- Memoria de pila (stack) de 8 niveles de profundidad
- Protecciones:
 - Power-on Reset (POR)
 - Power-up Timer (PWRT)
 - Oscillator Start-up Timer (OST)
 - Watchdog Timer (WDT) independiente del cristal.

PUERTOS DE SALIDA Y ENTRADA

- PORTA (RA5, RA4, RA3, RA2, RA1, RA0)
- PORTB (RB7, RB6, RB5, RB4, RB3, RB2, RB1, RB0)
- PORTC (RC7, RC6, RC5, RC4, RC3, RC2, RC1, RC0)
- PORTD (RD7, RD6, RD5, RD4, RD3, RD2, RD1, RD0)
- PORTE (RE2, RE1, RE0) Timer 0: timer/counter de 8 bits con un pre-escalador de 8 valores.

PERIFÉRICOS

- Timer 1: 16-bit timer/counter con pre-escalador
- Timer 2: 8-bit timer/counter con registro de estado de 8-bit, pre-escalador y post-escalador
- Dos módulos de Capture, Compare, PWM
- Capture es de 16-bit, max. resolución es 12.5 ns
- Compare es de 16-bit, max. resolución es 200 ns

- PWM max. resolución de 10-bit Convertidor analógico a digital de 10-bit multi-canal
- Puerto serial síncrono (SSP) con SPI. (modo maestro) e I2C (maestro/esclavo)
- Transmisor-Receptor síncrono-asíncrono universal (USART/SCI) con 9-bit
- Puerto paralelo esclavo (PSP) con 8-bits de ancho, con terminales de control RD, WR y CS

ARQUITECTURA INTERNA

- Arquitectura HARVARD.
- Buses separados (datos e instrucciones)
- Memoria de programa : 14 bits
- Memoria de datos: 8 bits.
- Recursos mapeados en memoria de datos.

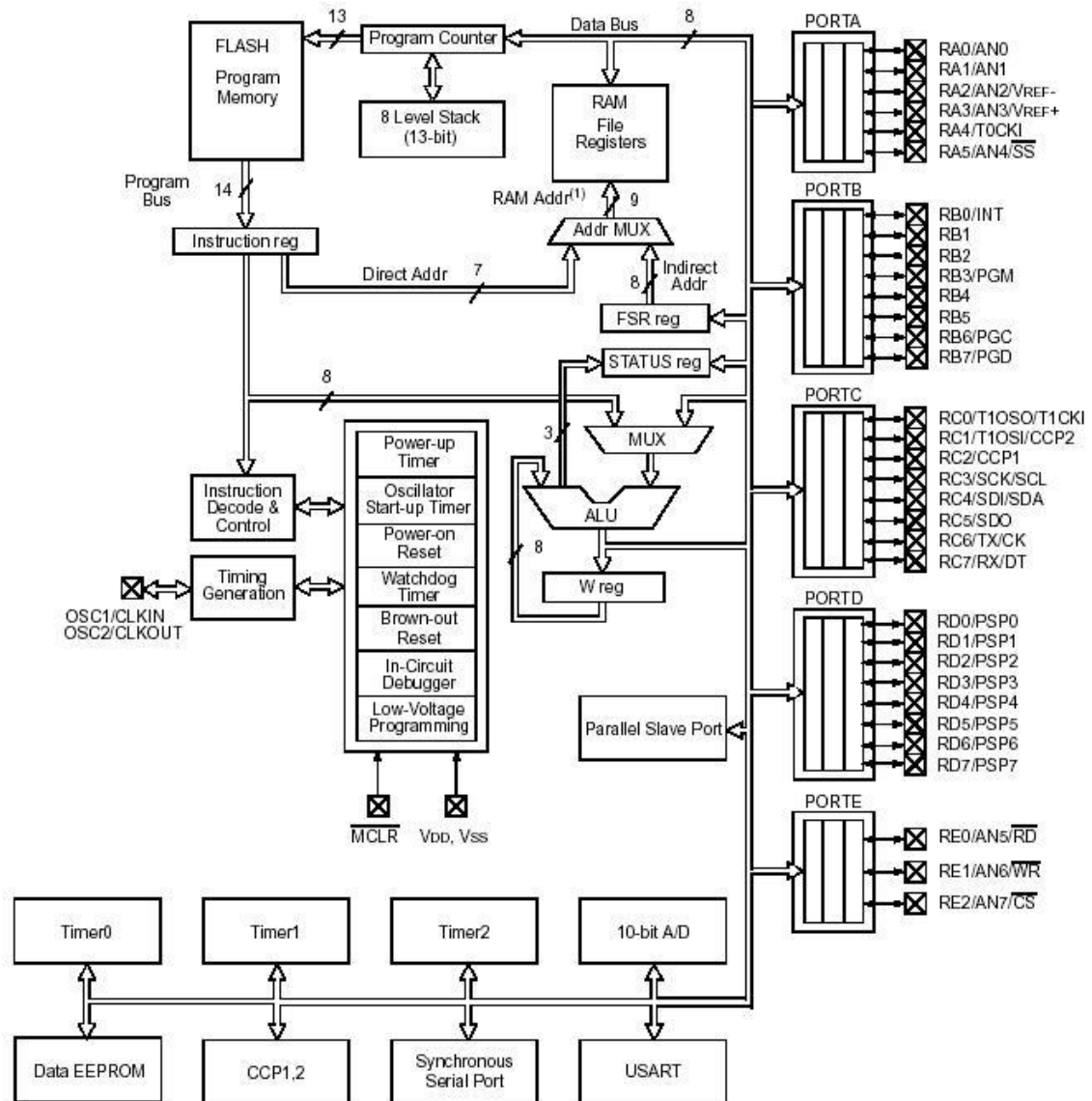


Figura 5. Arquitectura interna del PIC

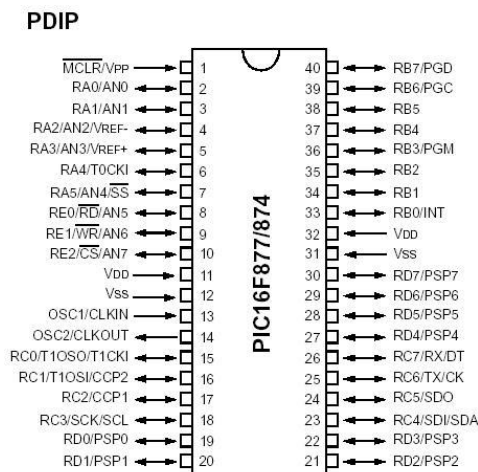


Figura 6. Terminales físicas

PUERTOS

Puerto	# funciones	Funciones
PORTA	3	Entradas digital Salidas digital Entradas analógicas
PORTB	2	Entradas digital Salidas digital
PORTC	3	Entradas digital Salidas digital Medios de comunicación
PORTD	3	Entradas digital Salidas digital Puerto paralelo esclavo
PORTE	4	Entradas digital Salidas digital Entradas analógicas Control del puerto paralelo esclavo

Tabla 1. Puertos

Funciones PORTA

Terminal	Funciones			
RA0	Ent. Digital	Sal. Digital	Ent. Analógica	
RA1	Ent. Digital	Sal. Digital	Ent. Analógica	
RA2	Ent. Digital	Sal. Digital	Ent. Analógica	VREF -
RA3	Ent. Digital	Sal. Digital	Ent. Analógica	VREF +
RA4	Ent. Digital	Sal. Digital	Ent. contador 1	
RA5	Ent. Digital	Sal. Digital	Ent. Analógica	

Tabla 2. Funciones PORTA

Funciones PORTB

Terminal	Funciones		
RB0	Ent. Digital	Sal. Digital	Ent. Interrupción 0
RB1	Ent. Digital	Sal. Digital	
RB2	Ent. Digital	Sal. Digital	
RB3	Ent. Digital	Sal. Digital	PGM (función LVP)
RB4	Ent. Digital	Sal. Digital	
RB5	Ent. Digital	Sal. Digital	
RB6	Ent.	Sal. Digital	PGC (función LVP)

	Digital		
RB7	Ent. Digital	Sal. Digital	PGD (función LVP)

Tabla 3. Funciones PORTB

Funciones PORTC

Terminal	Funciones			
RC0	Ent. Digital	Sal. Digital	Sal. Osc timer 1	Ent. Contador 1
RC1	Ent. Digital	Sal. Digital	Ent. Osc Timer 1	Captura/Comp/PWM 1
RC2	Ent. Digital	Sal. Digital		Captura/Comp/PWM 2
RC3	Ent. Digital	Sal. Digital	Reloj sincrono SPI	Reloj síncrono I2C
RC4	Ent. Digital	Sal. Digital	Datos entrada SPI	Datos I2C
RC5	Ent. Digital	Sal. Digital	Datos salida SPI	
RC6	Ent. Digital	Sal. Digital	Transmisión USART	
RC7	Ent. Digital	Sal. Digital	Recepción USART	

Tabla 4. Funciones PORTC

ESTRUCTURA DE UN PROGRAMA EN C (CICLO WHILE)

```
// Definición de variables globales

// Definición de funciones

void main(void)
{
    // Definición de variables locales

    // Configuración de registros (recursos y puertos)

    // ciclo infinito
    while ( 1 )
    {

        // Programa de usuario

    }
}
```

3.3.- PUERTO SERIAL

El puerto serial de las computadoras es conocido como puerto RS-232, la ventaja de este puerto es que todas las computadoras traen al menos un puerto serial, este permite la comunicación entre otros dispositivos tales como otra computadora, el mouse, impresora y para nuestro caso con los microcontroladores.

Existen dos formas de intercambiar información binaria: la paralela y la serial.

La comunicación paralela transmite todos los bits de un dato de manera simultánea, por lo tanto la velocidad de transferencia es rápida, sin embargo tiene la desventaja de utilizar una gran cantidad de líneas, por lo tanto se vuelve mas costoso y tiene la desventaja de atenuarse a grandes distancias, por la capacitancia entre conductores así como sus parámetros distribuidos.

3.3.1.- TIPOS DE COMUNICACION SERIAL:

Existen dos tipos de comunicaciones seriales: la síncrona y asíncrona. En la comunicación serial síncrona además de una línea sobre la cual se transmitirán los datos se necesita de una línea la cual contendrá los pulsos de reloj que indicaran cuando un dato es valido.

Ejemplos de este tipo de comunicación son:

- I2C
- ONE WIRE
- SPI

En la comunicación serial asíncrona, no son necesarios los pulsos de reloj. La duración de cada bit esta determinada por la velocidad con la cual se realiza la transferencia de datos.

La siguiente figura muestra la estructura de un carácter que se transmite en forma serial asíncrona.

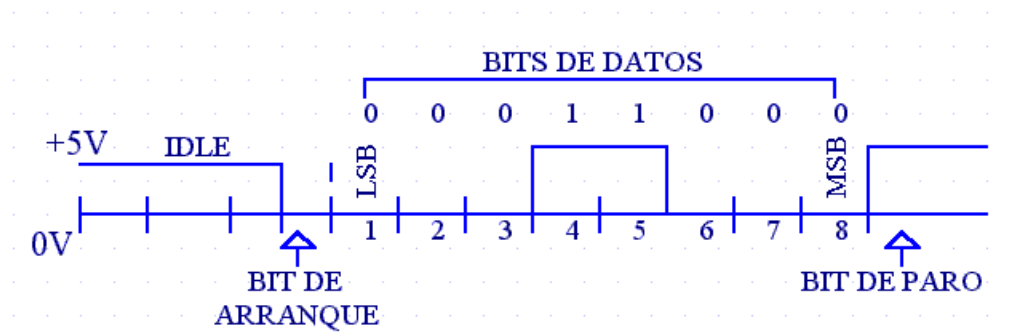


Figura 7. Transmisión asíncrona.

Normalmente cuando no se realiza ninguna transferencia de datos, la línea del transmisor se encuentra en estado de (idle) esto quiere decir en estado alto.

Para iniciar la transmisión de datos, el transmisor coloca esta línea en bajo durante determinado tiempo, lo cual se le conoce como bit de arranque (start bit) y a continuación empieza a transmitir con un intervalo de tiempo los bits correspondientes al dato, empezando siempre por el BIT menos significativo (LSB), y terminando con el BIT mas significativo.

Si el receptor no esta sincronizado con el transmisor, este desconoce cuando se van a recibir los datos.

Por lo tanto el transmisor y el receptor deberán tener los mismos parámetros de velocidad, paridad, número de bits del dato transmitido y de BIT de parada.

En los circuitos digitales, cuyas distancias son relativamente cortas, se pueden manejar transmisiones en niveles lógicos TTL (0-5V), pero cuando las distancias aumentan, estas señales tienden a distorsionarse debido al

efecto capacitivo de los conductores y su resistencia eléctrica. El efecto se incrementa a medida que se incrementa la velocidad de la transmisión.

Todo esto origina que los datos recibidos nos sean igual a los datos transmitidos, por lo que nos e puede permitir la transferencia de datos.

Una de las soluciones mas lógica es aumentar los márgenes de voltaje con que se transmiten los datos, de tal manera que las perturbaciones a causa de la línea se pueda corregir.

3.3.2.- LA NORMA RS-232

Ante la gran variedad de equipos, sistemas y protocolos que existen surgió la necesidad de un acuerdo que permitiera a los equipos de varios fabricantes comunicarse entre si. La EIA (Electronics Industry Association) elaboro la norma RS-232, la cual define la interfase mecánica, los pines, las señales y los protocolos que debe cumplir la comunicación serial.

Todas las normas RS-232 cumplen con los siguientes niveles de voltaje:

- Un "1" lógico es un voltaje comprendido entre $-5v$ y $-15v$ en el transmisor y entre $-3v$ y $-25v$ en el receptor.
- Un "0" lógico es un voltaje comprendido entre $+5v$ y $+15 v$ en el trasmisor y entre $+3v$ y $+25 v$ en el receptor.

El envío de niveles lógicos (bits) a través de cables o líneas de transmisión necesita la conversión a voltajes apropiados. En los microcontroladores para representar un 0 lógico se trabaja con voltajes inferiores a 0.8v, y para un 1 lógico con voltajes mayores a 2.0V. En general cuando se trabaja con familias TTL y CMOS se asume que un "0" lógico es igual a cero Volts y un "1" lógico es igual a cinco Volts.

La importancia de conocer esta norma, radica en los niveles de voltaje que maneja el puerto serial del ordenador, ya que son diferentes a los que utilizan los microcontroladores y los demás circuitos integrados. Por lo tanto se necesita de una interfase que haga posible la conversión de niveles de voltaje a los estándares manejados por los CI TTL.

3.3.3.- EL CIRCUITO MAX-232

Este circuito soluciona los problemas de niveles de voltaje cuando se requiere enviar unas señales digitales sobre una línea RS-232.

Este chip se utiliza en aquellas aplicaciones donde no se dispone de fuentes dobles de +12 y -12 Volts. El MAX 232 necesita solamente una fuente de +5V para su operación, internamente tiene un elevador de voltaje que convierte el voltaje de +5V al de doble polaridad de +12V y -12V. Cabe mencionar que existen una gran variedad de CI que cumplen con la norma RS-232 como lo son:

MAX220, DS14C232, MAX233, LT1180A.

3.3.4.- ACCESO AL PUERTO SERIAL A TRAVÉS DE VISUAL BASIC



Para poder acceder al puerto serial y así poder enviar datos utilizado una aplicación creada en Visual Basic, se hace uso del control MS COMM, el cual trae incorporadas todas las funciones para configurar el puerto. Es gracias a este control que el manejo del puerto serial se facilita enormemente. Las propiedades más importantes de este control son las siguientes:

- ComPort: Activa y regresa el número del puerto serial (Comm1, Comm2)
- PortOpen: Activa y regresa el acceso al puerto.
- Input: Regresa los caracteres del buffer receptor.
- Output: Escribe una cadena sobre el buffer Transmisor.
- Settings: Activa y regresa la razón de Baudios, paridad, número de bits, bits de paro.

Para poder tener acceso a cualquier propiedad del puerto serial se utiliza la siguiente sintaxis:

Nombre del Control. Propiedad = Valor

En este caso el objeto es MS Comm1, por lo tanto si quisiera abrir el puerto, la instrucción sería:

MS Comm1.PortOpen = True

Sin embargo, para poder utilizar el puerto serial, primeramente, se debe colocar el control MS Comm1 en la forma y hacer clic con el botón derecho del mouse, para que puedan aparecer sus propiedades, tal y como lo muestra la siguiente figura:

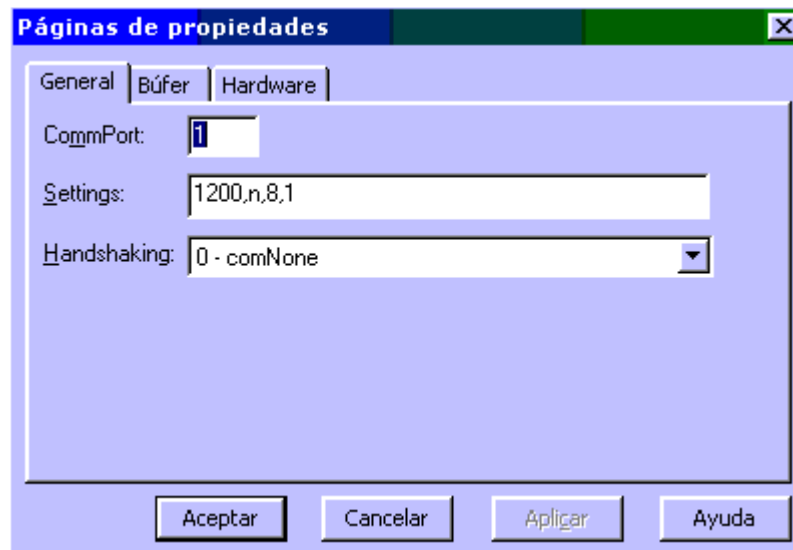


Figura 8. Configuración de los Parámetros del Puerto Serial en V Basic

Como la computadora sobre la cual se está trabajando solamente tiene un puerto serial y este es el comm1, en la propiedad CommPort debe tener el número 1, como los Microcontroladores envían y reciben la información a una velocidad de 1200 bps, 8 bits, sin paridad y 1 BIT de paro, en la propiedad Settings se debe configurar con la siguiente cadena: 1200, n ,8, 1, y como no se va a realizar ningún control sobre el flujo de la información la propiedad handshaking debe ser igual a 0.

El objeto MS Comm1 responde al siguiente evento On Comm, el cual genera una interrupción, indicando cuando hay comunicación o si algún error ha ocurrido en la transferencia de la información.

Para poder enviar una cadena de caracteres a través del puerto serial, lo único que se tiene que hacer es utilizar la propiedad output del objeto MS Comm:

Ejemplo:

```
MS Comm1.Output = "Esto es una prueba"
```

Como se observa, una vez configurado el puerto serial, con esta instrucción se envía a través del puerto la cadena de caracteres, "Esto es una prueba".

Pasos para poder enviar datos a través del Puerto Serial:

- Insertar el control MS Comm sobre la forma:
- Establecer las siguientes propiedades :

ComPort:

Settings:

Handshaking:

- Abrir el puerto, si este ya está abierto por otra aplicación, entonces se debe cerrar esa aplicación, para después volverlo a abrir el puerto con una aplicación en Visual Basic, esto se hace utilizando la siguiente instrucción:

```
MSComm1.Portopen = true
```

- Definir el tamaño del buffer receptor, esto se hace con la propiedad InputLen

```
MSComm1.InputLen = 1024
```

- Enviar los datos que se desean
- Cuando la aplicación se termine se debe cerrar el puerto.

3.3.5.- APLICACIÓN

1. Cree un Nuevo proyecto Form1 (es creado por default).
 2. Seleccione del menú Project ,verifique que el control Microsoft Comm, este en la barra de herramientas
 3. Agregue el control MSCOMM a la forma.
 4. Agregue 2 controles Command Buttons a la forma.
- Agregue el siguiente código a sus respectivos controles:

Option Explicit

Const Xon = &H11

Const Xoff = &H13

Private Sub Form_Load()

Form1.Caption = "Primera aplicación con el Puerto Serial"

With MSComm1

.Handshaking = 2 - comRTS

.RThreshold = 1

.RTSEnable = True

.Settings = "9600,n,8,1"

.SThreshold = 1

.PortOpen = True

End With

Command1.Caption = "&Send Xoff"

Command2.Caption = "Send &Xon"

End Sub

Private Sub Command1_Click()

MSComm1.Output = "123456789" & Chr\$(Xoff)

End Sub

Private Sub Command2_Click()

```
MSComm1.Output = "987654321" & Chr$(Xon)
```

```
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)
```

```
MSComm1.PortOpen = False
```

```
End Sub
```

3.4.- VISUAL BASIC

Visual Basic es la versión del lenguaje de programación BASIC, orientado al sistema operativo Windows con todas sus características (manejo de ventanas y de controles, iconos, gráficos, funciones de la API, etc.). El lenguaje de programación BASIC es uno de los más sencillos de aprender y por tanto, Visual Basic, es la mejor herramienta para aquellos que quieran iniciarse en la programación, ya que con unas pocas líneas de código pueden observarse ya los resultados.

Visual Basic es una forma de programación *basada en objetos* u *orientada al evento*. ¿Por qué esto? Al ser Windows un entorno multitarea, varias aplicaciones pueden ejecutarse a la vez (esta es una de las diferencias entre este sistema operativo y otros como MS-DOS); entonces las aplicaciones permanecen a la espera de acciones de parte del usuario, de parte de sí mismas o del sistema operativo para ejecutar una parte de su código y volver al estado de espera. Estas acciones se denominan eventos y el código que se ejecuta son funciones y métodos, todos estos se encuentran dentro de unas entidades llamadas Objetos.

Con Visual Basic se pueden crear aplicaciones (*.exe), librerías dinámicas (*.dll), controles ActiveX (*.ocx) entre otras cosas. Contiene algunos

complementos para utilizar dentro del entorno de trabajo para gestionar bases de datos, crear archivos de recursos (*.res), utilizar la API de Windows, crear clases, etc. Posee muy buenas herramientas de depuración para encontrar errores (bugs) y corregirlos, para ejecutar el programa de forma controlada y verificar los valores de variables, detener el flujo en un momento dado y más.

Si aparte se instala la ayuda en línea, denominada MSDN (no viene en el mismo CD que Visual Basic), proporciona mucha ayuda junto con ejemplos de código, explicaciones, etc.

Luego de creada toda la estructura de la aplicación se compila ésta para crear el ejecutable, librería o control ActiveX que se haya creado, lo cual podrá funcionar en otros ordenadores, aunque en ellos no se encuentre instalado Visual Basic. Para ello se necesitará que se encuentre en el directorio System del directorio Windows (o donde se encuentre éste) la librería *msvbvm60.dll* para las aplicaciones sencillas; si se utilizan otros componentes y referencias se necesitarán además de otros archivos en el ordenador.

4.- DESARROLLO

4.1.- INVESTIGACIÓN DOCUMENTAL DE LOS SENSORES:

Durante este periodo se investigo todo lo referente a los dispositivos que se tenían contemplados usar desde un principio.

Sensores. Los sensores ópticos que se investigaron principalmente fueron los infrarrojos, encontrando varias configuraciones de circuitos que constaban básicamente de un diodo emisor de infrarrojo (Tx) y un fototransistor (Rx). Un circuito integrado que hacía la función de generador de pulsos (LM 555 o LM567) que crea la rafaga de pulsos que son transmitidos por el led de chorro infrarrojo, luego es recibido por el fototransistor colocado de tal manera que solo los reciba cuando un objeto refleje los pulsos. Luego esa señal es procesada para poder utilizarla y enviarla a la tarjeta de adquisición de datos.

Al parecer por lo investigado, con simples reflectores de LED's se pueden obtener alcances del orden del metro. Con lentes convexas se pueden cubrir distancias de cinco metros. Es conveniente sacrificar algo de rango pero colocar filtros UV y SUNLIGHT los cuales no dejan entrar al fototransistor (elemento receptor) los rayos del sol.

Ya que la distancia que necesitábamos era menor a un metro cualquiera de estos circuitos supliría las necesidades del proyecto.

Después de los resultados obtenidos de la pruebas con estos sensores, se procedió a continuar con la búsqueda de uno que cumpliera con las necesidades del lugar y la distancia.

4.2.- INVESTIGACIÓN DOCUMENTAL DE OTRO SENSOR OPTICO

Se opto por buscar otros sensores que cumplieran con todos los requisitos necesarios de uso y que fueran robustos en su uso.

La opción que cumplía con estos requisitos además de un costo no muy elevado fue usar como emisor un láser comercial que va desde 1mW de potencia.

En esta búsqueda de un circuito detector por láser el que mas usos y recomendaciones tenía fue el siguiente:

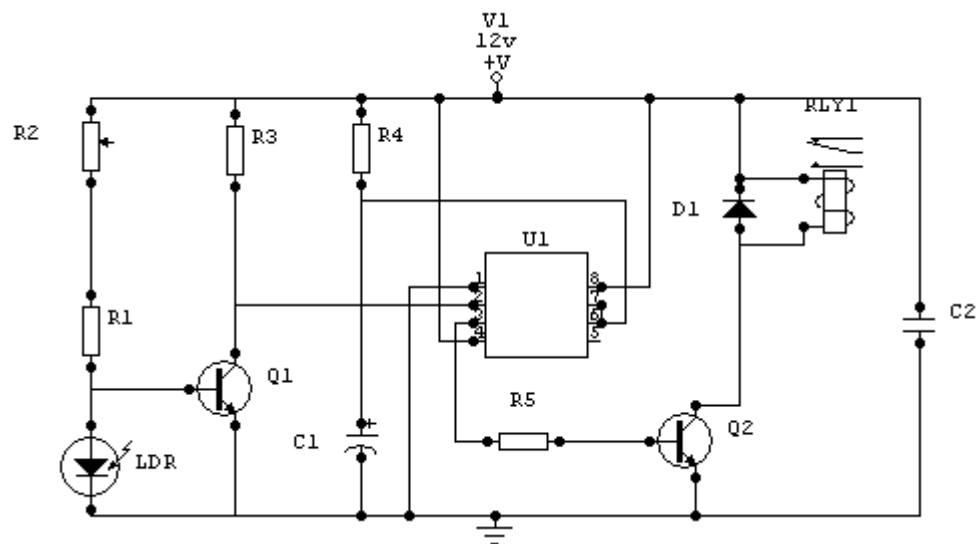


Figura 9. Circuito sensor de presencia láser.

El alcance de este sensor es de unos 300 m. entre el láser y la LDR, la luz del láser debe incidir en la foto resistencia de forma que cuando el haz es interrumpido se produce la activación del sensor, en caso de que el láser sea muy potente deberá proteger la LDR con un dispositivo que amortigüe el haz luminoso

Componentes:

R1 100 k Ω	C1 100 μ F	<u>LDR FR-27</u>
R2 2.2 M Ω	C2 220 μ F	TR1 <u>BC548</u>
R3 47 k Ω		TR2 <u>BD135</u>
R4 1 k Ω		D1 <u>1N4004</u>
R5 1 k Ω		IC1 <u>LM555</u>
		Láser de 1,0 Mw.

Tabla 5. Componentes del circuito de la figura 13.

Como podrá verse el alcance de este circuito es más que suficiente para el uso que necesitamos. Los componentes no son ni costosos ni difíciles de conseguir, la alimentación puede ser menor a 12 volts, en este caso podemos emplear una tensión de 5 volts para utilizar una misma fuente.

4.3.- IMPLEMENTACIÓN DEL SENSOR LÁSER

Así que se compraron los componentes y se hicieron las pruebas necesarias en el lugar donde se utilizaría. Al hacer estas pruebas se decidió poner una mayor cantidad de fotorresistencias en serie para aumentar la respuesta del circuito cuando el haz láser de este sea interrumpido y el receptor (fotorresistencias) no perciba la luz del láser, evitando con esto que cualquier otra luz, como la del día por ejemplo sea captada por las fotorresistencias y afecte el sensado.

El circuito funciona de la siguiente forma:

Se utiliza una configuración poco frecuente en el TRIGGER (disparo) del integrado NE555. El circuito integrado NE555 se utiliza para configurar un disparador. Una característica importante de este circuito es el hecho de no sufrir esas desagradables oscilaciones de los circuitos convencionales cuando

la iluminación llega al umbral de disparo. La transición del punto de espera al disparo es inmediata y única sin oscilaciones.

Se aplica al pin 5 del NE555 una tensión de referencia, aproximadamente igual a la mitad de la tensión de alimentación. La tensión aplicada en el pin 2 depende del valor de la LDR, y cuando sea menor que la mitad de la tensión de alimentación (la del pin 5) se activara la salida (pin 8). En nuestro circuito, fijamos la tensión de referencia por un divisor formado por R2 y R3 y ajustamos el disparo en función de la luz que incide en el LDR. El ajuste se realiza en función de la luz del láser, ajustando R1 hasta que se dispare el circuito.

Un ejemplo práctico del cual se tomó y modifíco para nuestro uso particular este circuito aparece en el marco teórico.

4.4.- IMPLEMENTACIÓN DEL PROGRAMA EN EL MICROCONTROLADOR.

Después de probar ya con la placa el circuito y comprobar su correcto funcionamiento se continuó a hacer el programa que recibirá la señal del sensor y procesará este dato.

Para esto se empleo la programación en lenguaje C con la ayuda del software mikroC v8. Y para su simulación se utilizo el software de Proteus. Ya que el elemento principal en esta etapa donde se recibe la señal del sensor es un microcontrolador, se seleccionaron estos dos programas.

El programa hará básicamente lo siguiente:

Al momento en que el sensor detecte una interrupción, es decir, cuando un saco se interponga en el haz del láser se enviara un cambio de nivel bajo a nivel alto (0v a 5v) a uno de los pines del microcontrolador (RB1).

Al recibir la señal el microcontrolador deberá guardarlo en una variable que ira sumando de acuerdo al número de veces que reciba la señal este incremento obviamente será de uno en uno.

Esta variable será guardada en la memoria EEPROM interna del microcontrolador.

Cuando el microcontrolador reciba cierto comando del programa hecho en la computadora a través del puerto serial por medio de los pines RC6 y RC7 para transmisión USART y recepción USART respectivamente el programa del microcontrolador regresará el valor de la variable que esta siendo incrementada por el sensor.

Para que posteriormente el programa de la computadora manipule esa variable recibida desde la tarjeta de adquisición de datos.

El código del programa compilado en mikroC y los comentarios de cada renglón se muestra a continuación:

```
unsigned short comando=0; // Declaración de variables
```

```
unsigned short cont = 0; // Declaración de variables
```

```
void main()
```

```
{
```


ADCON1=6; \\ Instrucción indispensable para usar el puerto A y el puerto E como entrada o salida de datos digitales

TRISB=0b11111111; // Todo el puerto B como entrada de datos digitales

TRISC.F6=0; // Transmisión serial USART

TRISC.F7=1; // Recepción serial USART

TRISC=128;

//TXSTA=0b00100100; // Asincrono, alta velocidad, 8 bits, transmisión hab.

//RCSTA=0b10010000; // Habilita el puerto y la recep de 8 bits,

TRISA=0b00000000; // Puerto A como salida de datos digitales

// Initialize USART module (8 bit, 2400 baud rate, no parity bit..)

Usart_Init(19200); // Velocidad de transmisión y recepción en bauds

cont=0; // se inicializa el contador en cero

while(1) // Inicio del ciclo infinito

{

PORTA.F1 = PORTB.F1; // Indicador en el Puerto A1 que se enciende cada que el Puerto B1 recibe un dato

if (PORTB.F1 ==1) // si el Puerto B1 es igual a 1, es decir, 5volts enviados por el sensor

```
{  
  
    while(PORTB.F1==1) // mientras este valor se mantenga  
  
        {;} // espera hasta que cambia el estado de alto a bajo, no importa el  
tiempo que esto tarde  
  
    Delay_ms(100); // NOP  
  
    cont++; // Incremento del contador  
  
    EEprom_Write(1,cont); //Se escribe en la memoria EEprom en la posición  
numero 1 el valor del contador  
  
}  
  
if(Usart_Data_Ready()) // Si se lee un dato en la Usart  
  
    {  
  
        comando=Usart_Read(); //Leer el comando que se recibió  
  
        if(comando==97) // si el comando que se envió de la computadora es  
igual a la letra "a" en codigo ASCII  
  
            {  
  
                Usart_Write(cont+32); //se envía el valor de la variable cont mas  
32 para que no se envíe basura ya que en ASCII los primeros 32 caracteres  
no son legibles
```

```
comando=0; //el comando se reinicia en ceros

cont = 0; // también el contador se reinicia para que no se repita el
dato enviado

}

if(comando==101) // si el comando que se envió de la computadora
es igual a la letra "e" en código ASCII

{

    Usart_Write(105); //se envía el valor de la letra "i" en ASCII

    comando=0; // se reinicia el comando nuevamente

}

}

} //ciclo infinito

} //void main
```

4.5.- DISEÑO E IMPLEMENTACIÓN DE LA TARJETA DE ADQUISICIÓN DE DATOS.

Después de hacer las pruebas en el simulador y comprobar que funcionaba correctamente se prosiguió a diseñar la tarjeta de adquisición de datos que contienen la entrada del puerto serial, el circuito Max 232 y el microcontrolador con los puertos de entrada. El circuito de la tarjeta de acuerdo a la configuración de estos tres elementos es el siguiente:

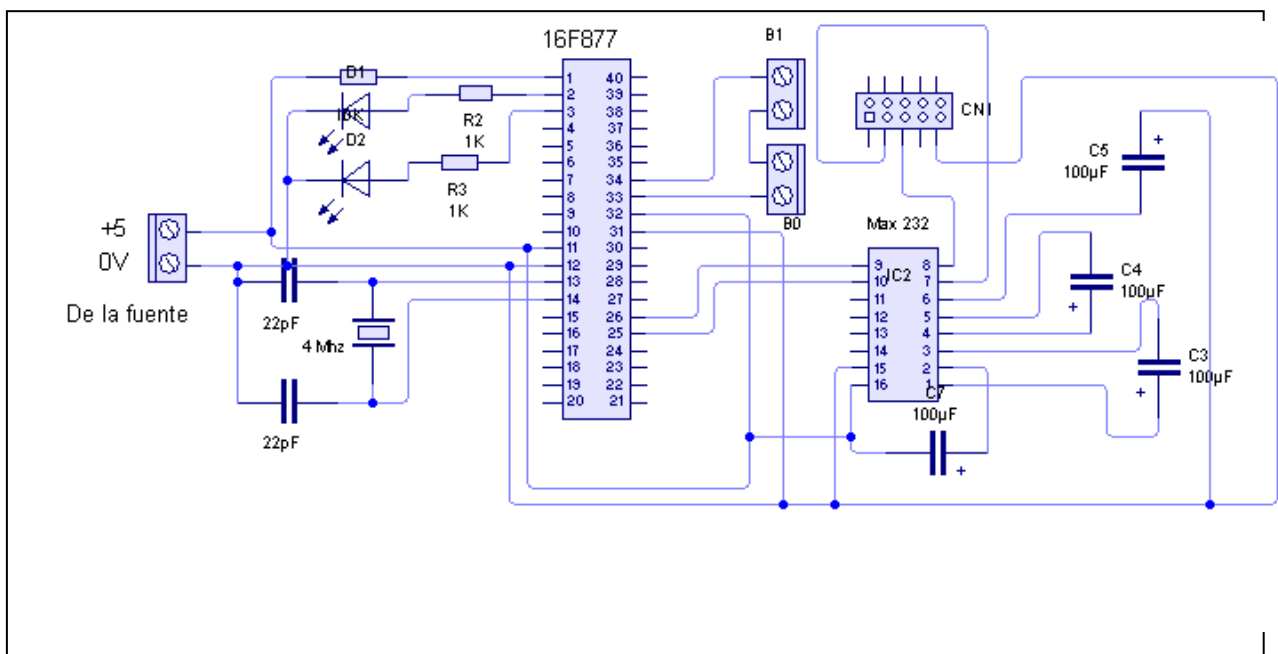


Figura 10. Circuito de la tarjeta de adquisición de datos.

El programa del microcontrolador se grabó con la ayuda del software PIC-PROG USB EDUTRÓNICA REV 1.9 que utiliza un útil quemador de PIC's USB.

El circuito Max 232 soluciona los problemas de niveles de voltaje cuando se requiere enviar unas señales digitales sobre una línea RS-232.

Este chip se utiliza en aquellas aplicaciones donde no se dispone de fuentes dobles de +12 y -12 Volts. El MAX 232 necesita solamente una fuente de +5V para su operación, internamente tiene un elevador de voltaje que convierte el voltaje de +5V al de doble polaridad de +12V y -12V. Cabe mencionar que existen una gran variedad de CI que cumplen con la norma RS-232 como lo son:

MAX220, DS14C232, MAX233, LT1180A.

Para mayor explicación sobre este circuito y la norma RS 232 buscar en el fundamento teórico.

4.6.- IMPLEMENTACIÓN DEL PROGRAMA EN VISUAL BASIC.

Después de hacer las pruebas necesarias con la placa en una simulación de mikroC y comprobar que enviaba y recibía datos correctamente se procedió a hacer el programa que el usuario o encargado de la bodega manipulará.

Como bien ya se mencionó, el programa esta hecho en Visual Basic y hace lo siguiente:

Al abrir el programa se comprueba si el dispositivo (la tarjeta de adquisición de datos) esta correctamente conectada.

Si se comprueba que esta correctamente conectado se puede hacer lo siguiente: visualizar en que numero de puerto de comunicación serial se encuentra conectado (COM) para su correcto funcionamiento al momento de hacer la comunicación entre tarjeta y computadora.

Se introduce el numero de puerto serial que se esta utilizando

Muestra la hora y la fecha actual del sistema.

Un botón que inicia el conteo de forma automática, esto lo hace por un intervalo de 5 segundos, para que el dato recibido por el microcontrolador se envíe constantemente a la computadora y la información se actualice.

Un botón que detiene este conteo automático, dejando de mostrar en pantalla el valor del conteo hasta ese momento pero sin que el programa pierda la cuenta, ya que el conteo simplemente no se muestra pero este sigue llegando.

Un botón que escanea la cantidad exacta que hasta ese momento ha pasado por el sensor según el conteo que se haya iniciado.

Un comando que guarda el conteo final en un fichero de texto (extensión .txt o bloc de notas), con la hora y la fecha en que se guardo.

Un comando que manda los mismos datos de Guardar a un archivo de texto en Word para su posible edición o impresión.

El código de este programa en Visual Basic y los comentarios se muestran en las siguientes líneas:

Private Sub administrador_Click()

Shell "C:\WINDOWS\system32\mmc.exe

C:\WINDOWS\system32\devmgmt.msc", 1 -> Manda a ejecutar el

administrador de dispositivos de Windows que sirve para saber que numero de Puerto serial se esta utilizando

End Sub Finaliza esta subrutina

Private Sub Command1_Click() -> Botón escanear

MSComm1.CommPort = Text1.Text ->El número de Puerto es igual al numero que se introducirá en la caja de texto 1

MSComm1.PortOpen = True -> Manda a abrir el Puerto serial

MSComm1.Output = "a" -> Envía el commando "a" que el microcontrolador recibirá

RTB1.Text = MSComm1.Input -> Manda a RTB1 los caracteres del buffer receptor

RTB1.Text = Asc(RTB1.Text) - 32 ->RichTextBox1 es igual al valor en ASCII del mismo menos 32

RTB2.Text = Val(RTB1.Text) + Val(RTB2.Text) -> Esto se usa para que el dato que llegue no se sobrescriba y se pierda el anterior sino que se sume y vaya haciéndose el total de sacos que entran, así también de esta manera el envío del microcontrolador no se desborda por la limitante de solo poder enviar hasta el numero $255-32 = 223$

MSComm1.PortOpen = False -> Cierra el Puerto serial por si otra aplicación la llegara a necesitar no cause conflictos

End Sub

Private Sub Command2_Click()

CD1.Filter = "Ficheros de Texto |*.txt"

CD1.ShowSave Despliega la ventana para guardar la informacion en un archive .txt

If CD1.FileName = "" Then Si no existe ningun archive despues de haber dado clic en Guardar

CD1.FileName = "C:\Error.txt" Crea un archive.txt en la unidad C:, con el nombr error.

-> Guarda en la direccion C:\Error.txt un archivo de bloc de notas en caso de que se le de clic en el comando guardar y después no se guarde nada sino que se cancele la operación.

End If Fin del if

Open CD1.FileName For Append As #1 Crea un archive con el nombre que se le asigne por el usuario para ser guardado.

Print #1, Text2.Text

Print #1, RTB2.Text

Print #1, Text3.Text

Print #1, RTB4.Text

Print #1, Text4.Text

Print #1, RTB5.Text

Print #1, Text5.Text

La información que contienen todos estos RTB's y Text's son los que se guardan en el archivo que se creó.

Close #1

-> Guarda en un archivo de bloc de notas todos los RichTextBox y TextBox que aqui se mencionan, conteniendo estos los datos necesarios de hora, fecha y numero de sacos del conteo final

End Sub

Private Sub Command3_Click()

Timer1.Enabled = True Habilita el Timer1

Option1.Value = True

-> Activa el timer1 que a su vez activa el conteo automatico y pone como verdadero el OptionButton1

End Sub

Private Sub Command4_Click()

Timer1.Enabled = False Desactiva el Timer1

Option1.Value = False Desactiva el OptionButton1

MSComm1.CommPort = Text1.Text El número de Puerto serial es el mismo que el numero que esta escrito en el Text1.

MSComm1.PortOpen = True Se abre el Puerto serial

MSComm1.Output = "a" -> Escribe una cadena sobre el buffer Transmisor

-> Desactiva el timer1 que a su vez desactiva el conteo automatico y el OptionButton1

RTB1.Text = MSComm1.Input -> Manda a RTB1 los caracteres del buffer receptor

RTB1.Text = Asc(RTB1.Text) - 32 -> RichTextBox1 es igual al valor en ASCII del mismo menos 32

RTB2.Text = Val(RTB1.Text) + Val(RTB2.Text) -> Esto se usa para que el dato que llegue no se sobrescriba y se pierda el anterior sino que se sume y vaya haciéndose el total de sacos que entran, así también de esta manera el envío del microcontrolador no se desborda por la limitante de solo poder enviar hasta el numero $255-32 = 223$

MSComm1.PortOpen = False -> Cierra el Puerto serial por si otra aplicación la llegara a necesitar no cause conflictos

End Sub

Private Sub Command5_Click()

Shell "C:\WINDOWS\system32\mmc.exe

C:\WINDOWS\system32\devmgmt.msc", 1

End Sub

Private Sub Command6_Click()

Open "C:\imprimir.doc" For Append As #1

Print #1, Text2.Text

Print #1, RTB2.Text

Print #1, Text3.Text

Print #1, RTB4.Text

Print #1, Text4.Text

Print #1, RTB5.Text

Print #1, Text5.Text

Close #1

Shell "C:\Archivos de programa\Microsoft

Office\OFFICE11\WINWORD.exe C:\imprimir.doc", 1

-> Genera el archivo .doc en word con los datos que están en los TextBox y RichText Box aqui mencionados

End Sub

Private Sub detectar_Click()

MSComm1.CommPort = Text1.Text

MSComm1.PortOpen = True

MSComm1.Output = "a" -> Escribe una cadena sobre el buffer Transmisor
-> Comando que se envía para comprobar que el dispositivo esta correctamente conectado

RTB10.Text = MSComm1.Input Recibe en el RTB10 lo que el pic le esta enviando.

MSComm1.PortOpen = False Se cierra el Puerto serial por si se requiere usar en otra aplicación.

If RTB10.Text = "i" Then Si lo que se recibio del pic es igual a la letra i.

Check1.Value = 1 El marcador del Check1.Value se activa

Else De lo contrario

Check1.Value = 0 Permanece desactivado.

End If Fin del if

End Sub Fin de la subrutina

Private Sub Form_Load()

MSComm1.Settings = "19200,n,8,1" -> 19200 bps, sin paridad, 8 bits, 1 bit de paro

RTB1.Visible = False

RTB3.Visible = False

-> Oculta el RTB1 y el RTB3

End Sub

Private Sub Label3_Click()

End Sub

Private Sub Guardar_Click()

CD1.Filter = "Ficheros de Texto |*.txt"

CD1.ShowSave

If CD1.FileName = "" Then

CD1.FileName = "C:\Error.txt"

End If

Open CD1.FileName For Append As #1

Print #1, Text2.Text

Print #1, RTB2.Text

Print #1, Text3.Text

Print #1, RTB4.Text

Print #1, Text4.Text

Print #1, RTB5.Text

Print #1, Text5.Text

Close #1

-> Guarda en un fichero de texto los datos contenidos en estos RTB's y
Text's

End Sub

Private Sub imprimir_Click()

Open "C:\imprimir.doc" For Append As #1

Print #1, Text2.Text

Print #1, RTB2.Text

Print #1, Text3.Text

Print #1, RTB4.Text

Print #1, Text4.Text

Print #1, RTB5.Text

Print #1, Text5.Text

Close #1

Shell "C:\Archivos de programa\Microsoft Office\OFFICE11\WINWORD.exe C:\imprimir.doc", 1

->Manda a imprimir o editar todos estos datos

End Sub

Private Sub reiniciar_Click()

RTB2.Text = "0"

-> Reinicia el contador

End Sub

Private Sub salir_Click()

End

End Sub

Private Sub seleccionar_Click()

Text1.Text = InputBox("Teclear el No. de Puerto Serial", "HARINERA DE CHIAPAS S.A. DE C.V.", "1")

-> Aparece una ventana para escribir el numero de puerto serial que se necesita usar

End Sub

Private Sub Timer1_Timer()

MSComm1.CommPort = Text1.Text

MSComm1.PortOpen = True

MSComm1.Output = "a"

RTB1.Text = MSComm1.Input

RTB1.Text = Asc(RTB1.Text) - 32

RTB2.Text = Val(RTB1.Text) + Val(RTB2.Text)

MSComm1.PortOpen = False

End Sub

Private Sub Timer2_Timer()

RTB4.Text = Date

RTB5.Text = Time

-> Muestra la hora y la fecha actual que hay en la computadora

End Sub

End Sub

Private Sub version_Click()

**MsgBox "SISTEMA DE CONTEO 1.0" & vbCrLf & " J.A.M.C.",
vbQuestion, "HARINERA DE CHIAPAS S.A. DE C.V."**

-> Despliega la ventana que muestra información del programa

End Sub

NOTA: Algunos códigos se repiten porque están como botones y como comandos del menú. Por eso no se explicaron cada una de las líneas, pero si se observa correctamente podrán ver las similitudes.

5.- PRUEBAS Y RESULTADOS

5.1.- PRUEBAS CON LOS CIRCUITOS DE LOS SENSORES ENCONTRADOS.

Como era necesario asegurarnos de que los circuitos investigados para los sensores eran funcionales se tomó la decisión de hacer las pruebas necesarias con los dos circuitos que más reincidencia tuvieron en las páginas encontradas en la etapa de investigación antes de seguir con los demás elementos del sistema.

El primer circuito es similar al que se muestra en la siguiente figura.

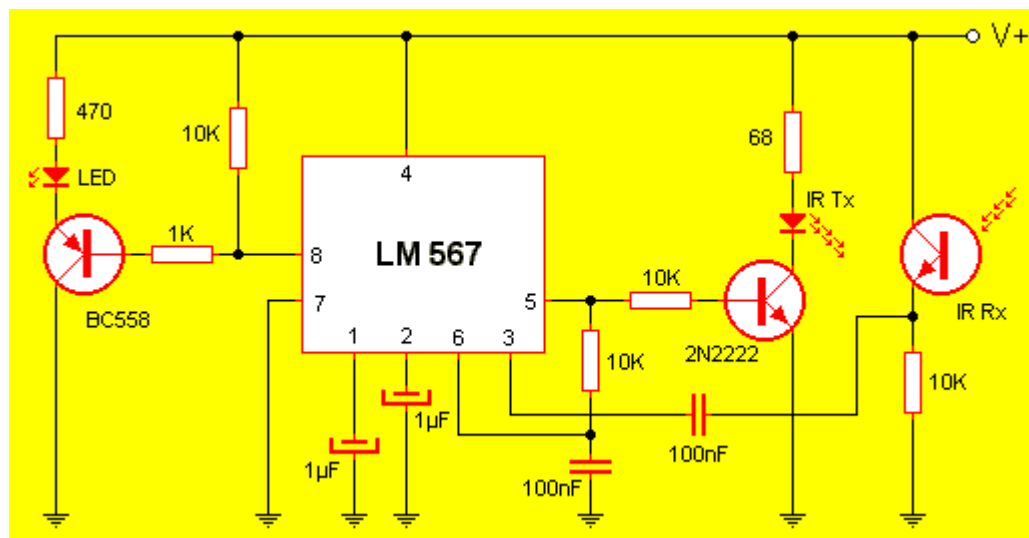


Figura 11. Detector de proximidad infrarrojo 1.

El funcionamiento del circuito se basa en emitir una ráfaga de señales luminosas infrarrojas las cuales al rebotar contra un objeto cercano se reciben por otro componente. Al ser recibidas el sistema detecta proximidad con lo que el led de salida se acciona (brilla).

El circuito integrado es un generador/decodificador de tonos que bien cumple con las necesidades de este diseño. Tanto el fotodiodo como el fototransistor deberán estar situados con unidades de enfoque adecuadas para mejorar el alcance.

La alimentación de este circuito puede ser cualquier tensión comprendida entre 5 y 9 volts, se uso de 5 volts para que fuese compatible con los demás dispositivos que se usarán más adelante.

Con este circuito se obtuvo un alcance aproximado de un metro y medio. El segundo circuito que se implemento fue el que se muestra en la siguiente figura.

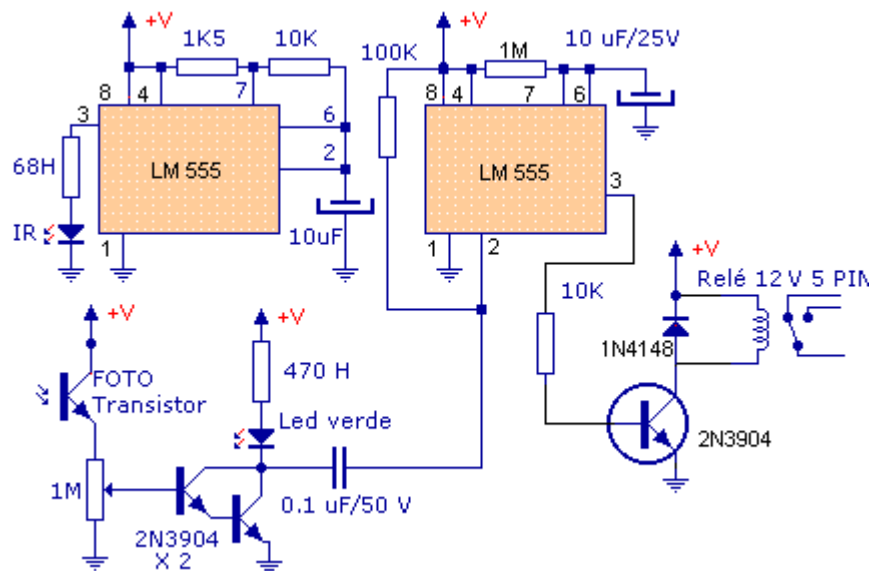


Figura 12. Detector de proximidad infrarrojo 2.

El principio de funcionamiento de este circuito es similar al anterior: generamos una ráfaga de pulsos de alta intensidad con el LM555 a baja frecuencia y los transmitimos por el led infrarrojo. Luego los recibimos en un fototransistor colocado de tal manera que solo los reciba cuando un objeto refleje los pulsos.

Para poder procesar la señal colocamos un fototransistor de tal manera que cuando haya una superficie que refleje los pulsos, bien sea una mano, un objeto cualquiera, a una distancia de unos 10 cm, este los pueda recibir y enviar a un amplificador de corriente, en este caso un par de transistores en configuración darlington.

Con este circuito probado en protoboard como con el anterior también, se alcanzó una distancia entre emisor y receptor sin perder la señal menor a un metro.

El ancho que se midió en las bandas era de 70 centímetros, por lo tanto, con el primer circuito se lograrían los resultados deseados.

5.2.- IMPLEMENTACIÓN DEL CIRCUITO SENSOR INFRARROJO

Se procedió a hacer el diseño de la placa del primer circuito para poder hacer pruebas más exactas en el lugar donde quedaría instalado.

El diseño de la placa se realizó primeramente con la ayuda del software Livewire donde se hizo el esquema del circuito como se ve en la figura 10 y después se convirtió dentro del mismo al circuito placa con la ayuda del software PCB Wizard y quedó de la siguiente forma:

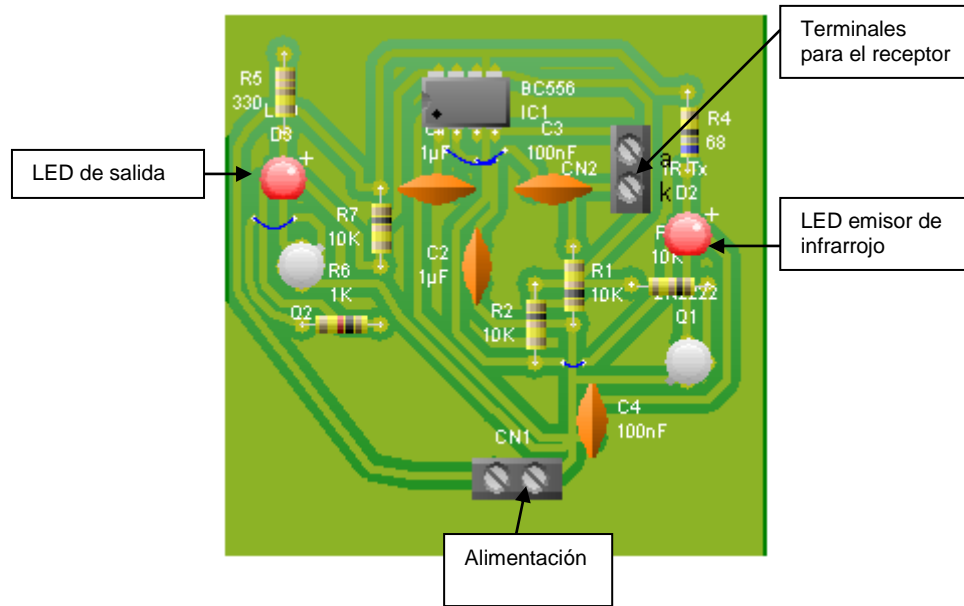


Figura 12. Placa del detector de proximidad infrarrojo 1.

Se puede apreciar en la figura el conector de la parte superior de la placa que es de donde salen los cables que conectan al receptor infrarrojo (fototransistor) ya que este fue el que se escogió para estar al otro extremo de la banda donde sería colocada la placa con el emisor infrarrojo (LED), así también el LED que aparece en el extremo derecho de la placa es el LED emisor infrarrojo.

5.3.- PRUEBAS CON EL SENSOR INFRARROJO

Después de terminado el sensor en placa se llevo al lugar de instalación para hacer las pruebas, que es en las bandas transportadoras de los sacos de producto.

Al llegar hacer las pruebas en el lugar de trabajo se observó que la iluminación del lugar así como las vibraciones de la banda complicaría la instalación. Por lo que se comprobó para mala fortuna que aunque instalándolo de manera independiente a las bandas para evitar cualquier

vibración posible este circuito no se podría implementar en ese lugar ya que la luz del lugar que era muy poca afectaba el sensado, además de que en horarios nocturnos la luz artificial también afectaría.

De la misma forma ningún sensor infrarrojo de los que se pueden encontrar en las tiendas de electrónica cercanas servirían ya que son de muy bajo alcance y cualquier otra luz los llega afectar, y para el uso que se necesita específicamente en este proyecto no servirían.

Por lo tanto se buscaron otros sensores infrarrojos por Internet en diferentes lugares del país y el extranjero, pero estos además de tener un precio superior al presupuesto aprobado por la empresa no garantizaban la funcionalidad y esto representaba un riesgo al momento de pedirlos.

Se llegó a la conclusión de que cualquiera de los circuitos que emplean emisor y receptor infrarrojo funciona, pero sólo en situaciones ideales y donde la luz del exterior no presente demasiadas vibraciones, es decir, todo el trabajo avanzado hasta el momento fue desechado ya que para este uso específicamente no suplía las necesidades.

Aunque todo esto no se utilizó para el proyecto tomé la decisión de incluirlo ya que ocupó buena parte del tiempo hacer todo esto, y principalmente porque puede ser de utilidad para algún lector que necesite implementar un detector de proximidad infrarrojo con menores exigencias que para el caso de este proyecto que esta implementado en un campo industrial, por ejemplo podría usarse para maquetas ilustrativas donde se necesita una distancia mas corta.

La siguiente etapa como se afirma en el desarrollo del proyecto fue buscar otro sensor que cumpliera con las necesidades del lugar y distancia.

El sensor por el que nos inclinamos debido a su menor precio y buena funcionalidad fue el que tiene como emisor de un lado un láser y como receptor a fotorresistencias (el desarrollo de esto se describió anteriormente, en este momento solo nos enfocaremos en las pruebas y resultados).

Se procedió a probar el sensor láser, el cual funcionó correctamente en el lugar de trabajo. Dando como valor un 0 lógico (0volts) mientras el haz no era interrumpido y un 1 lógico (5 volts) cuando algún objeto se interponía entre el emisor y el receptor.

La placa del circuito detector por láser se muestra en la siguiente figura:

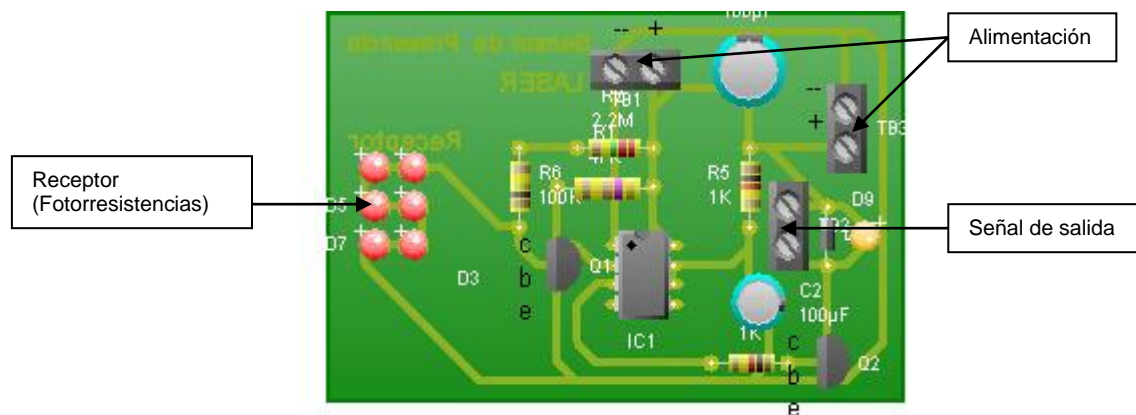


Figura 13. Placa del circuito sensor de presencia láser.

5.4.- PRUEBAS CON EL PROGRAMA HECHO EN MIKROC

De la misma forma después de pasar la etapa de sensado del sistema se continuó con el programa de la tarjeta de adquisición de datos hecho en mikroC. Se comprobó el funcionamiento del programa en Proteus.

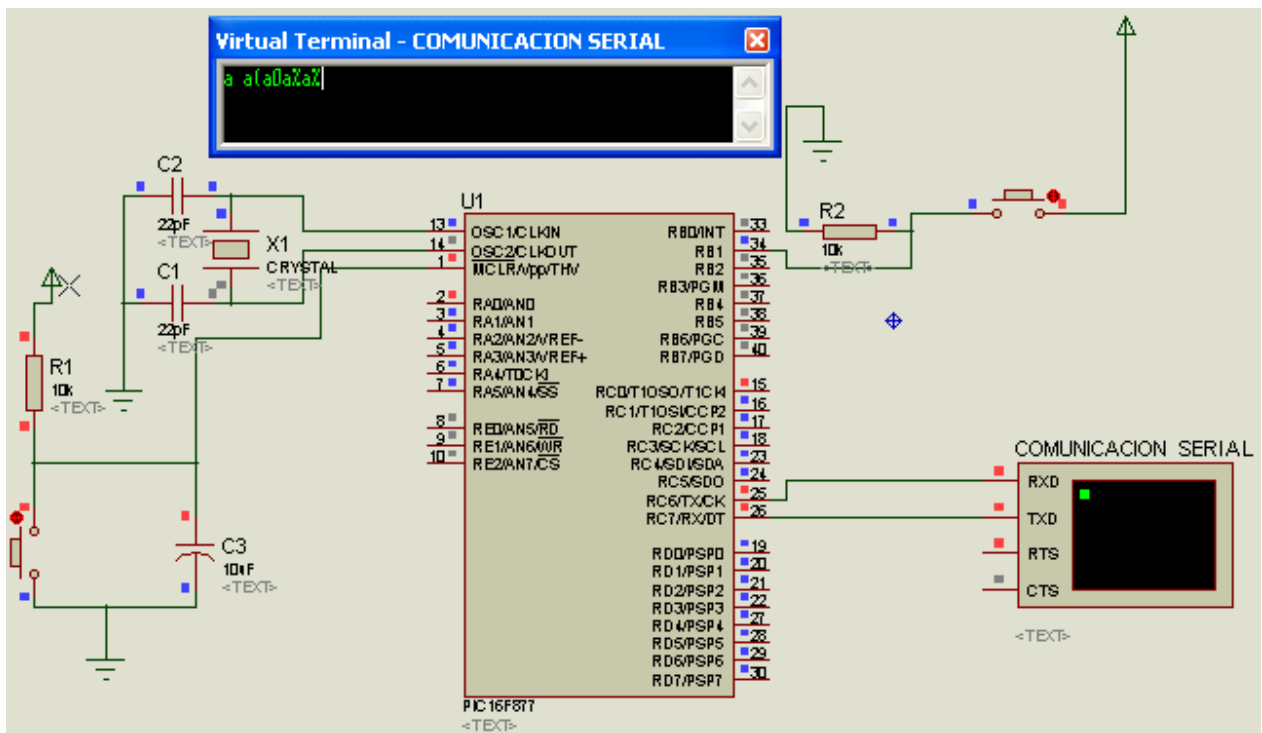


Figura 14. Simulación en Proteus.

En esta figura se simulan la entrada de la señal del sensor y la comunicación serial del pic. El push ubicado en RB1 es la entrada del sensor láser. Los pines RC6 y RC7 son de transmisión y recepción serial del pic. Como puede verse en la terminal virtual, al momento de introducir la letra "a" (que será el comando que enviaremos desde la computadora para que el pic nos regrese el valor del contador), el valor que te regresa son símbolos y no números,

esto es porque estamos viendo el código ASCII de los números; posteriormente en el programa hecho en visual se convertirá ese código.

También en los anexos se pone una tabla de los valores de estos caracteres.

5.5.- IMPLEMENTACION DE LA TARJETA DE ADQUISICION DE DATOS

Ahora con la programación del pic hecha se continúa con la implementación de la tarjeta de adquisición de datos.

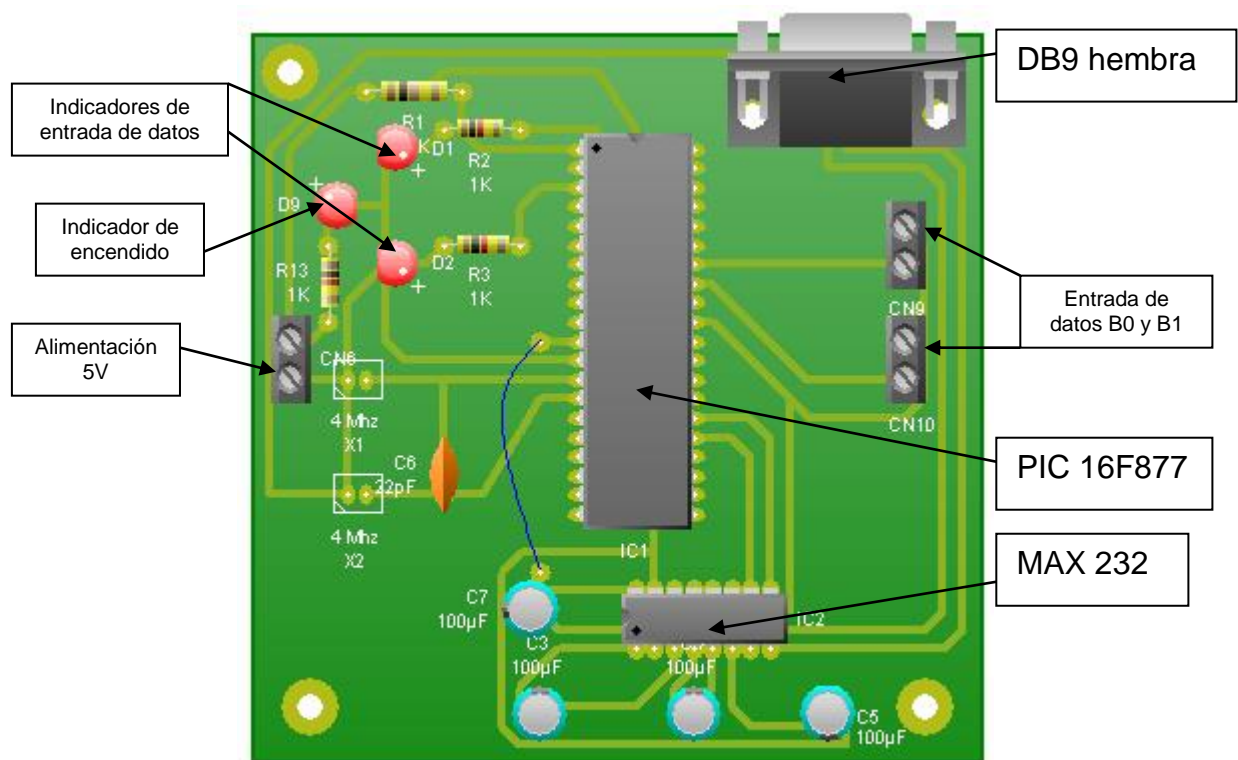


Figura 15. Placa de la tarjeta de adquisición de datos.

Esta tarjeta se probó con mikroC con la ayuda del elemento que simula la comunicación de la terminal serial para ver si enviaba y recibía correctamente los datos.

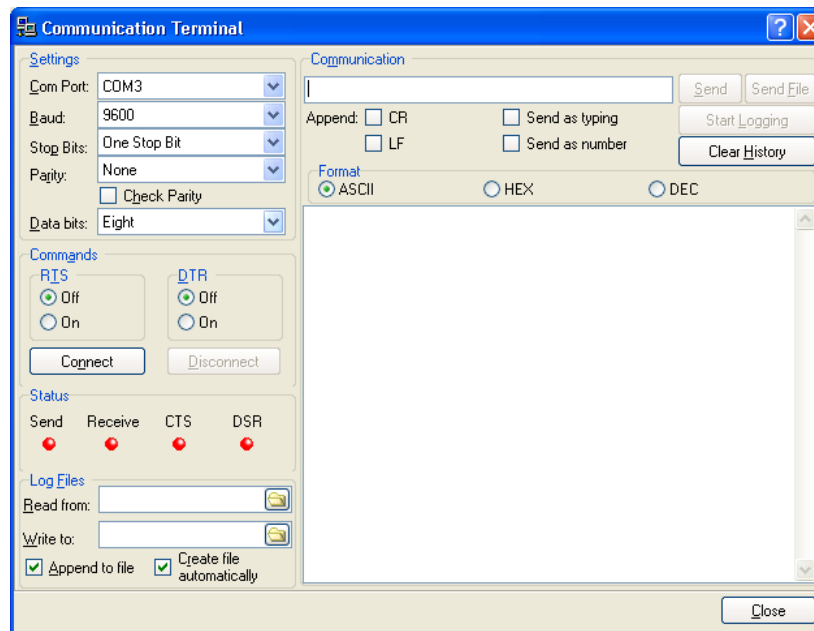


Figura 16. "Communication Terminal" de mikroC

5.6.- PRUEBAS CON EL PROGRAMA EN VISUAL BASIC

Ya por último se probó con todos los elementos el funcionamiento del programa que manipulará el usuario.

Como bien ya se mencionó, el programa esta hecho en Visual Basic y hace lo siguiente:

Al abrir el programa se comprueba si el dispositivo (la tarjeta de adquisición de datos) esta correctamente conectada.

Si se comprueba que esta correctamente conectado se puede hacer lo siguiente: visualizar en que numero de puerto de comunicación serial se encuentra conectado (COM) para su correcto funcionamiento al momento de hacer la comunicación entre tarjeta y computadora.

Se introduce el numero de puerto serial que se esta utilizando

Muestra la hora y la fecha actual del sistema.

Un botón que inicia el conteo de forma automática, esto lo hace por un intervalo de 5 segundos, para que el dato recibido por el microcontrolador se envíe constantemente a la computadora y la información se actualice.

Un botón que detiene este conteo automático, dejando de mostrar en pantalla el valor del conteo hasta ese momento pero sin que el programa pierda la cuenta, ya que el conteo simplemente no se muestra pero este sigue llegando.

Un botón que escanea la cantidad exacta que hasta ese momento ha pasado por el sensor según el conteo que se haya iniciado.

Un comando que guarda el conteo final en un fichero de texto (extensión .txt o bloc de notas), con la hora y la fecha en que se guardo.

Un comando que manda los mismos datos de Guardar a un archivo de texto en Word para su posible edición o impresión.

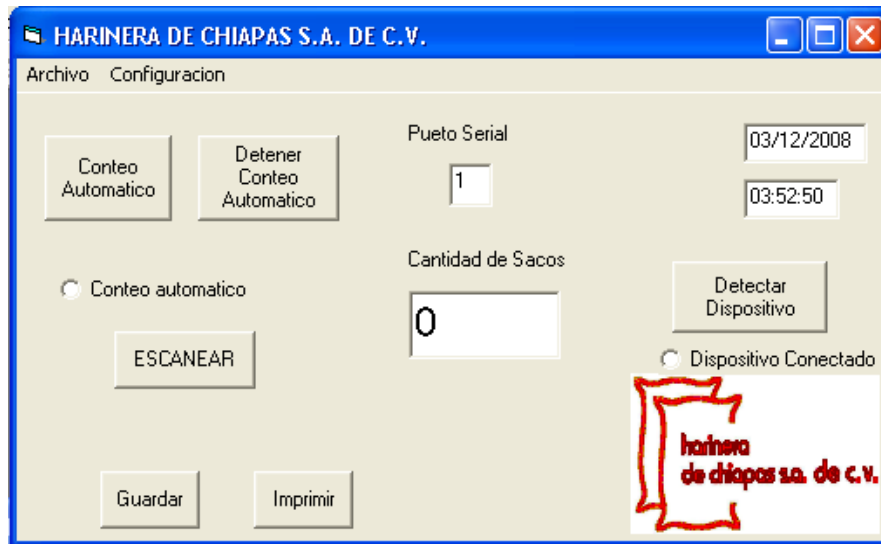


Figura 17. Ventana principal del programa

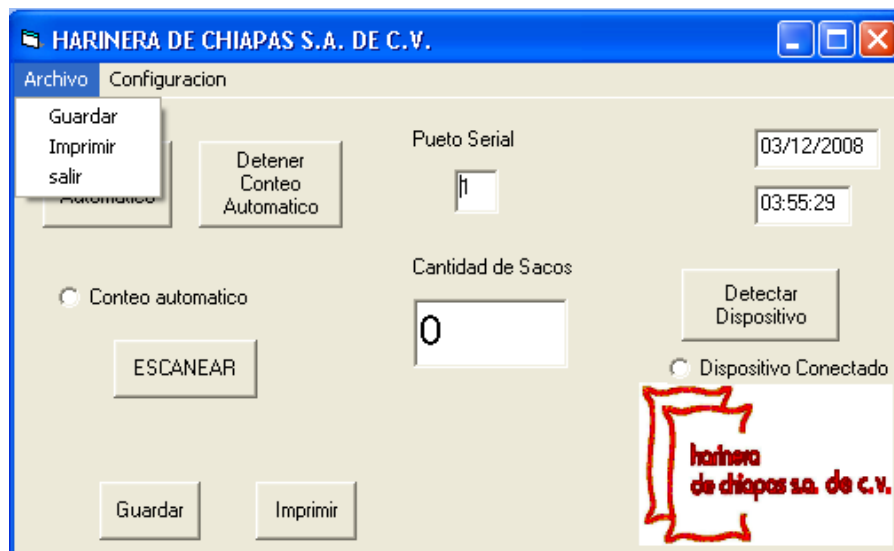


Figura 18. Menú Archivo.

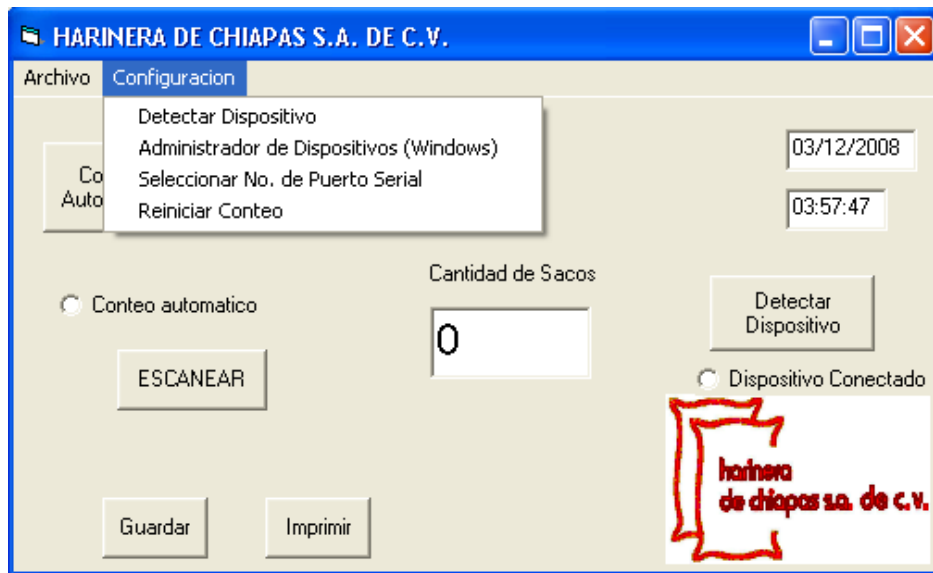


Figura 19. Menú Configuración

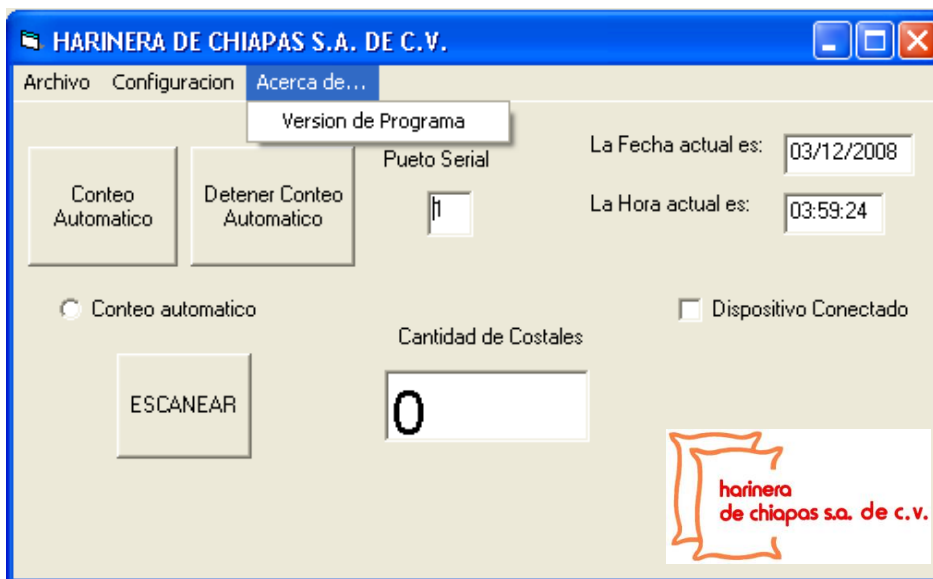


Figura 20. Menú Acerca de...

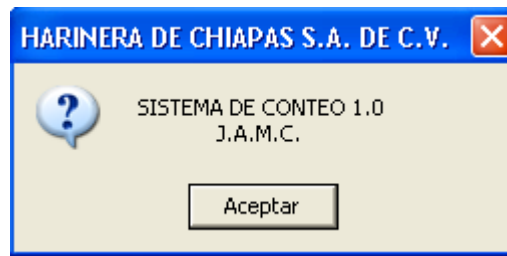


Figura 21. Menú Acerca de... Versión de Programa

6.- CONCLUSIONES

Se comprobó que el sensor óptico más útil para este proyecto es el de un emisor láser ya que por la distancia que se necesita es mas que suficiente y no afecta ninguna de las luces externas que estén sobre el receptor (LDR's).

Así también la comunicación computadora – microcontrolador de manera serial y no paralela es mas ahorrativa, fácil de manejar y con una respuesta bastante rápida ya que no necesitamos transmitir/recibir muchos datos al mismo tiempo.

Lo dicho anteriormente se comprueba con los resultados obtenidos ya que como se menciona en este trabajo se buscaron diversos sensores ópticos, y se investigó también que el caso del puerto paralelo solo es mejor en el caso de querer transmitir varios datos al mismo tiempo.

Aunque no tiene muchas complicaciones este sistema demostró ser funcional para la empresa ya que satisfizo esa necesidad específica.

Se ahorró al implementar este sistema, no solo a mediano plazo porque evitará pérdidas sino que también este sistema tiene un menor costo que

uno que se compra con algún proveedor externo, ya que el presupuesto usado para este proyecto es mínimo.

7.- BIBLIOGRAFÍA

- [1] Schmelkes Corina, Manual para la presentación de anteproyectos e informes de investigación, 2da Edición, Oxford, 1998.
- [2] Pallás, Ramón ; Valdés, Fernando, MICROCONTROLADORES: FUNDAMENTOS Y APLICACIONES CON PIC, 1ª edición (10/02/2007), MARCOMBO EDICIONES TÉCNICAS
- [3] <http://www.senet.com.au/~cpeacock>
- [4] <http://www.lvr.com>
- [5] <http://www.neoteo.com/ldr-fotorresistencias.neo>
- [6] <http://www.monografias.com/>
- [7} <http://wk3.iespana.es/serial.htm>
- [8] www.steren.com.mx

8.- ANEXOS

8.1.- DATOS GENERALES DE LA EMPRESA

Nombre:

HARINERA DE CHIAPAS S.A. DE C.V.



Domicilio:

Planta y Oficinas en Arriaga:

Av. Central Poniente Norte S/N.

Arriaga, Chiapas.

Tels.: 01 (966) 662-01-40 y 662-01-98.

Ventas: 01 (966) 662-21-69 y 662-21-70.

Fax: 01 (966) 662-07-67.

Oficinas en Tuxtla:

Calle 4a. Sur Poniente No. 1805.

Tuxtla Gutiérrez, Chiapas.

Tels. y Fax: 01 (961) 602-54-97, 602-54-98 y 602-54-99.

Giro:

Producción y comercialización de distintas marcas de harina y sus derivados (salvado, salvadillo y cemita).

Organigrama:

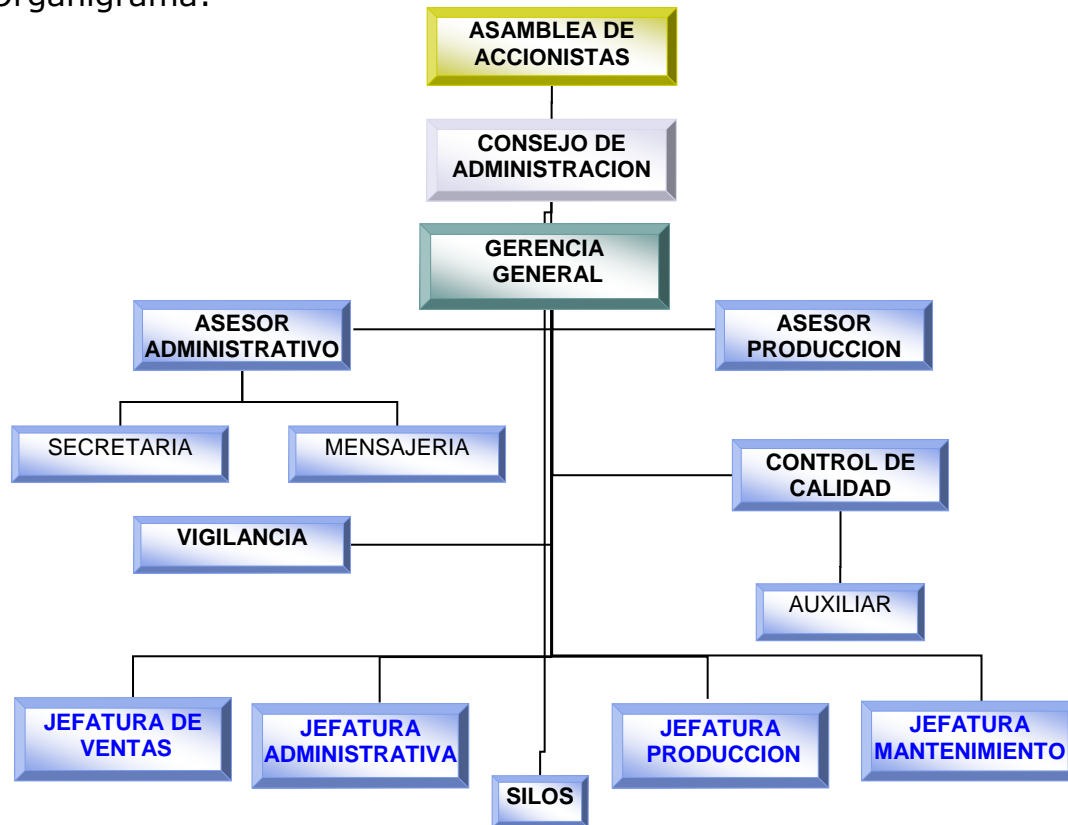


Figura 1. Organigrama de la empresa.

8.2.- BREVE DESCRIPCIÓN DE LA EMPRESA.

Su Historia.

Harinera de Chiapas.- Desde 1958 elabora productos de notable calidad gracias a sus estrictos controles de producción y a la estratégica combinación de trigos nacionales e importados. Su producción inicial fue de 40 tons. por día.

En el año de 1970, se modifica su capacidad de producción a 80 tons./día. En el año de 1978 se eleva nuevamente su capacidad hasta 120 tons. y posteriormente en el año 2001 a 140 tons./día.

En el año de 1988 se instala otra sección de molienda de 120 tons./día, siendo actualmente su capacidad total de 260 toneladas por día.

Adicional a la producción de Harinas y Sub-Productos, nuestra empresa cuenta con una infraestructura muy importante en almacenaje.

De inicio se cuenta con una capacidad en almacenaje en bodegas de 5,000 toneladas.

En el año de 1988 se construyen 10 Silos y 4 Intersilos con una capacidad total de 14,000 toneladas, adicionales a las bodegas, lo que proporciona un servicio a terceros para el almacenaje de sus granos, en un lugar muy estratégico por su ubicación en el estado de Chiapas.

Nuestra planta está ubicada en la ciudad de Arriaga, Chiapas, teniendo oficinas instaladas en Tuxtla Gutiérrez.

Arriaga es la primera ciudad en la costa chiapaneca, donde confluyen diversas vías de comunicación (carreteras y ferrocarril), posicionando a nuestra empresa en un lugar privilegiado para distribuir sus productos.

Tiene comunicación con el norte y con la costa de Chiapas, hasta llegar a Guatemala; con los litorales de Oaxaca, Guerrero y Veracruz; existiendo proyectos de nuevas carreteras que agilizarán los tiempos de entrega.

Nuestro proceso, rigurosamente controlado por medio de equipos modernos, nos permiten estandarizar nuestra calidad, desde la preparación de los granos hasta el empaclado y su distribución.

Nuestras marcas de harina se distinguen por su alta calidad:

- Flor de Chiapas.
- San Cristóbal.
- Monte Bello.
- Agua Azul.

Adicionalmente contamos con una producción permanente de sub-productos:

- Salvado.
 - Salvadillo.
 - Cemita.
- Flor de Chiapas.- En presentación de 44 Kgs., por ser una harina extrafina, es propicia para la elaboración de pan dulce, bollos, biscochos, hojaldres y teleras, para trabajar con equipo mecánico. En presentación de 10 Kgs. es propicia para la panadería artesanal.
- San Cristóbal.- En presentación de 44 Kgs., por ser una harina extrafina con alto contenido en proteína, es propicia para la elaboración de bolillos, teleras, pan de caja y todo tipo de pan dulce, para trabajar con equipo mecánico. En presentación de 10 y 20 Kgs. es propicia para la panadería artesanal.

- Montebello.- Es una harina semifuerte, propicia para utilizarse en la elaboración de bizcochería, bollería, pastelería y panadería artesanal. Presentación: 10 y 44 Kgs.
- Agua Azul.- Es una harina fina y suave, ideal para la elaboración de galletas y tortillas. Presentación: 10 y 44 Kgs.
- Salvado.- Por ser un producto con alto contenido de fibra, es propicio para la elaboración de panes especializados y para fabricas y tiendas de alimentos balanceados. Presentación: 25 Kgs.
- Salvadillo.- Es un producto similar al salvado, con un granulado más fino. Utilizado para fábricas y tiendas de alimentos balanceados. Presentación: 25 Kgs.
- Cemita.- Es un producto similar al salvadillo, con granulados mucho más fino. Propicio para la elaboración de panes especializados y alimentos balanceados. Presentación: 30 Kgs.

Nuestra Misión

"Contribuir al desarrollo nutricional social, económico y humano de la región sureste de México y Centroamérica, satisfaciendo las expectativas de las partes interesadas, mediante la elaboración y comercialización con calidad de harina de trigo y sus derivados".

8.3 HOJA DE DATOS DEL CIRCUITO LM555

Electrical Characteristics $T_A = 25^\circ\text{C}$, $V_{CC} = +5\text{V}$ to $+15\text{V}$, (unless otherwise specified) (Continued)								
Parameter	Conditions	Limits						Units
		LM555			LM555C			
		Min	Typ	Max	Min	Typ	Max	
Output Voltage Drop (Low)	$V_{CC} = 15\text{V}$		0.1	0.15		0.1	0.25	V
	$I_{SINK} = 10\text{ mA}$		0.4	0.5		0.4	0.75	V
	$I_{SINK} = 50\text{ mA}$		2	2.2		2	2.5	V
	$I_{SINK} = 100\text{ mA}$		2.5			2.5		V
	$V_{CC} = 5\text{V}$		0.1	0.25				V
	$I_{SINK} = 8\text{ mA}$ $I_{SINK} = 5\text{ mA}$					0.25	0.35	V
Output Voltage Drop (High)	$I_{SOURCE} = 200\text{ mA}$, $V_{CC} = 15\text{V}$	13	12.5			12.5		V
	$I_{SOURCE} = 100\text{ mA}$, $V_{CC} = 15\text{V}$	3	13.3		12.75	13.3		V
	$V_{CC} = 5\text{V}$		3.3		2.75	3.3		V
Rise Time of Output			100			100		ns
Fall Time of Output			100			100		ns

Note 1: For operating at elevated temperatures the device must be derated above 25°C based on a $+150^\circ\text{C}$ maximum junction temperature and a thermal resistance of $164^\circ\text{C}/\text{w}$ (TO-8), $108^\circ\text{C}/\text{w}$ (DIP) and $170^\circ\text{C}/\text{w}$ (SO-8) junction to ambient.

Note 2: Supply current when output high typically 1 mA less at $V_{CC} = 5\text{V}$.

Note 3: Tested at $V_{CC} = 5\text{V}$ and $V_{CC} = 15\text{V}$.

Note 4: This will determine the maximum value of $R_A + R_B$ for 15V operation. The maximum total ($R_A + R_B$) is 20 M Ω .

Note 5: No protection against excessive pin 7 current is necessary providing the package dissipation rating will not be exceeded.

Note 6: Refer to RET555X drawing of military LM555H and LM555J versions for specifications.

Connection Diagrams

Metal Can Package

Top View
Order Number LM555H or LM555CH
See NS Package Number H09C

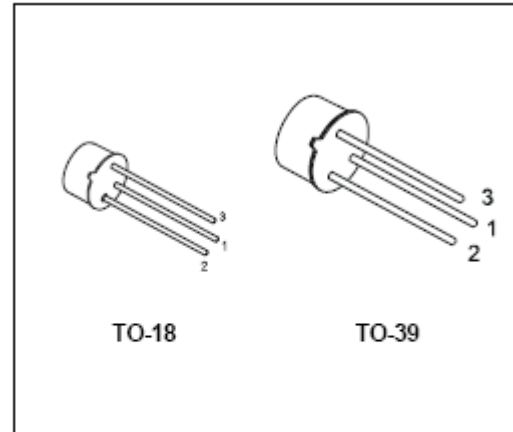
Dual-In-Line and Small Outline Packages

Top View
Order Number LM555J, LM555CJ,
LM555CM or LM555CN
See NS Package Number J08A, M08A or N08E

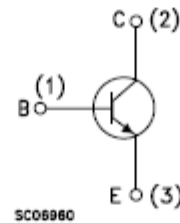
8.4.- HOJA DE DATOS DEL CIRCUITO 2N2222

DESCRIPTION

The 2N2219A and 2N2222A are silicon Planar Epitaxial NPN transistors in Jedec TO-39 (for 2N2219A) and in Jedec TO-18 (for 2N2222A) metal case. They are designed for high speed switching application at collector current up to 500mA, and feature useful current gain over a wide range of collector current, low leakage currents and low saturation voltage.



INTERNAL SCHEMATIC DIAGRAM



ABSOLUTE MAXIMUM RATINGS

Symbol	Parameter	Value	Unit
V_{CB0}	Collector-Base Voltage ($I_E = 0$)	75	V
V_{CE0}	Collector-Emitter Voltage ($I_B = 0$)	40	V
V_{EB0}	Emitter-Base Voltage ($I_C = 0$)	6	V
I_C	Collector Current	0.6	A
I_{CM}	Collector Peak Current ($t_p < 5$ ms)	0.8	A
P_{tot}	Total Dissipation at $T_{amb} \leq 25$ °C for 2N2219A for 2N2222A at $T_C \leq 25$ °C for 2N2219A for 2N2222A	0.8	W
		0.5	W
		3	W
		1.8	W
T_{stg}	Storage Temperature	-85 to 175	°C
T_J	Max. Operating Junction Temperature	175	°C

8.5.- HOJA DE DATOS DEL CIRCUITO BC 558

PNP general purpose transistors

BC556; BC557; BC558

FEATURES

- Low current (max. 100 mA)
- Low voltage (max. 65 V).

APPLICATIONS

- General purpose switching and amplification.

DESCRIPTION

PNP transistor in a TO-92; SOT54 plastic package.
NPN complements: BC546, BC547 and BC548.

PINNING

PIN	DESCRIPTION
1	emitter
2	base
3	collector

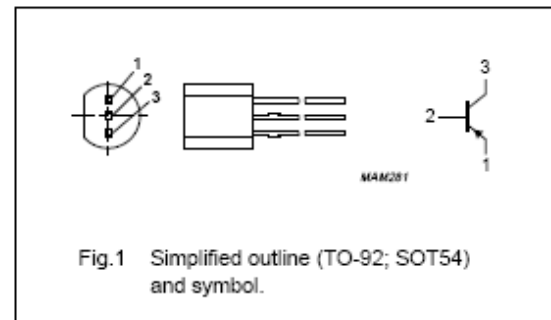


Fig.1 Simplified outline (TO-92; SOT54) and symbol.

QUICK REFERENCE DATA

SYMBOL	PARAMETER	CONDITIONS	MIN.	MAX.	UNIT
V_{CB0}	collector-base voltage	open emitter			
	BC556		-	-80	V
	BC557		-	-50	V
V_{CE0}	collector-emitter voltage	open base			
	BC556		-	-65	V
	BC557		-	-45	V
	BC558		-	-30	V
I_{CM}	peak collector current		-	-200	mA
P_{tot}	total power dissipation	$T_{amb} \leq 25\text{ }^{\circ}\text{C}$	-	500	mW
h_{FE}	DC current gain	$I_C = -2\text{ mA}; V_{CE} = -5\text{ V}$			
	BC556		125	475	
	BC557; BC558		125	800	
f_T	transition frequency	$I_C = -10\text{ mA}; V_{CE} = -5\text{ V}; f = 100\text{ MHz}$	100	-	MHz

8.6.- HOJA DE DATOS DEL CIRCUITO MAX 232

MAX232, MAX232I
DUAL EIA-232 DRIVERS/RECEIVERS

SLLS0-471 - FEBRUARY 1989 - REVISED OCTOBER 2002

- Meet or Exceed TIA/EIA-232-F and ITU Recommendation V.28
- Operate With Single 5-V Power Supply
- Operate Up to 120 kbit/s
- Two Drivers and Two Receivers
- ± 30 -V Input Levels
- Low Supply Current . . . 8 mA Typical
- Designed to be Interchangeable With Maxim MAX232
- ESD Protection Exceeds JESD 22 - 2000-V Human-Body Model (A114-A)
- Applications
 - TIA/EIA-232-F
 - Battery-Powered Systems
 - Terminals
 - Modems
 - Computers

MAX232 . . . D, DW, N, OR NS PACKAGE
 MAX232I . . . D, DW, OR N PACKAGE
 (TOP VIEW)

description/ordering information

The MAX232 is a dual driver/receiver that includes a capacitive voltage generator to supply EIA-232 voltage levels from a single 5-V supply. Each receiver converts EIA-232 inputs to 5-V TTL/CMOS levels. These receivers have a typical threshold of 1.3 V and a typical hysteresis of 0.5 V, and can accept ± 30 -V inputs. Each driver converts TTL/CMOS input levels into EIA-232 levels. The driver, receiver, and voltage-generator functions are available as cells in the Texas Instruments LinASIC™ library.

ORDERING INFORMATION

TA	PACKAGE†		ORDERABLE PART NUMBER	TOP-SIDE MARKING
0°C to 70°C	PDIP (N)	Tube	MAX232N	MAX232N
		Tube	MAX232D	MAX232
	SOIC (D)	Tape and reel	MAX232DR	MAX232
		Tube	MAX232DW	MAX232
	SOIC (DW)	Tape and reel	MAX232DWR	MAX232
SOP (NS)	Tape and reel	MAX232NSR	MAX232	
-40°C to 85°C	PDIP (N)	Tube	MAX232IN	MAX232IN
		Tube	MAX232ID	MAX232I
	SOIC (D)	Tape and reel	MAX232IDR	MAX232I
		Tube	MAX232IDW	MAX232I
	SOIC (DW)	Tape and reel	MAX232IDWR	MAX232I

† Package drawings, standard packing quantities, thermal data, symbolization, and PCB design guidelines are available at www.ti.com/sc/package.

8.7.- VALOR DE LOS CARACTERES ASCII.

ASCII Chart																	
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	NUL 0	SOH 1	STX 2	ETX 3	EOT 4	ENQ 5	ACK 6	BEL 7	BS 8	HT 9	LF 10	VT 11	FF 12	CR 13	SO 14	SI 15	
1	DLE 16	DC1 17	DC2 18	DC3 19	DC4 20	NAK 21	SYN 22	ETB 23	CAN 24	EM 25	SUB 26	ESC 27	FS 28	GS 29	RS 30	US 31	
2	SPC 32	! 33	" 34	# 35	\$ 36	% 37	_ 38	' 39	(40) 41	* 42	+ 43	, 44	- 45	. 46	/ 47	
3	0 48	1 49	2 50	3 51	4 52	5 53	6 54	7 55	8 56	9 57	: 58	; 59	< 60	= 61	> 62	? 63	
4	@ 64	A 65	B 66	C 67	D 68	E 69	F 70	G 71	H 72	I 73	J 74	K 75	L 76	M 77	N 78	O 79	
5	P 80	Q 81	R 82	S 83	T 84	U 85	V 86	W 87	X 88	Y 89	Z 90	[91	\ 92] 93	^ 94	_ 95	
6	` 96	a 97	b 98	c 99	d 100	e 101	f 102	g 103	h 104	i 105	j 106	k 107	l 108	m 109	n 110	o 111	
7	p 112	q 113	r 114	s 115	t 116	u 117	v 118	w 119	x 120	y 121	z 122	{ 123	 124	} 125	~ 126	DEL 127	
8	€ 128	□ 129	, 130	f 131	„ 132	… 133	† 134	‡ 135	^ 136	%o 137	Š 138	< 139	œ 140	□ 141	ž 142	□ 143	
9	□ 144	' 145	' 146	" 147	" 148	• 149	- 150	- 151	~ 152	™ 153	š 154	> 155	œ 156	□ 157	ž 158	Ÿ 159	
A		i 160	¢ 161	£ 162	¤ 163	¥ 164	¦ 165	§ 166	¨ 167	© 168	ª 169	« 170	¬ 171	- 172	® 173	¯ 174	175
B	° 176	± 177	² 178	³ 179	´ 180	µ 181	¶ 182	· 183	, 184	¡ 185	º 186	» 187	¼ 188	½ 189	¾ 190	¿ 191	
C	À 192	Á 193	Â 194	Ã 195	Ä 196	Å 197	Æ 198	Ç 199	È 200	É 201	Ê 202	Ë 203	Ì 204	Í 205	Î 206	Ï 207	
D	Ð 208	Ñ 209	Ò 210	Ó 211	Ô 212	Õ 213	Ö 214	× 215	Ø 216	Ù 217	Ú 218	Û 219	Ü 220	Ý 221	Þ 222	ß 223	
E	à 224	á 225	â 226	ã 227	ä 228	å 229	æ 230	ç 231	è 232	é 233	ê 234	ë 235	ì 236	í 237	î 238	ï 239	
F	ð 240	ñ 241	ò 242	ó 243	ô 244	õ 245	ö 246	÷ 247	ø 248	ù 249	ú 250	û 251	ü 252	ý 253	þ 254	ÿ 255	