

300617
36



UNIVERSIDAD LA SALLE
ESCUELA DE INGENIERIA
INCORPORADA A LA U. N. A. M.

“SISTEMA DE CONTROL DE ACCESO
BASADO EN UN MICROCONTROLADOR”

T E S I S P R O F E S I O N A L
QUE PARA OBTENER EL TITULO DE
INGENIERO MECANICO ELECTRICISTA
CON ESPECIALIDAD EN INGENIERIA ELECTRONICA
P R E S E N T A

FRANCISCO VEGA PRADERE

DIRECTOR DE TESIS :

ING. PATRICIA VASQUEZ AGUILERA

FALLA DE ORIGEN

MEXICO, D.F.

1 9 9 0



UNAM – Dirección General de Bibliotecas Tesis Digitales Restricciones de uso

DERECHOS RESERVADOS © PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis está protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

INDICE.

Introducción..... 1

Objetivos..... 4

CAPITULO I.

MICROCOMPUTADORES, MICROPROCESADORES Y

MICROCONTROLADORES.

I.1 Microcomputadores..... 6

I.1.1 Unidad Central de Proceso..... 8

I.1.2 Memoria..... 11

I.1.3 Puertos de Entrada/Salida..... 15

I.2 Comparación entre micoprocesadores
y microcontroladores..... 19

CAPITULO II.

ANALISIS COMPARATIVO DE LAS DIVERSAS MARCAS DE

MICROCONTROLADORES Y SELECCION DEL CIRCUITO.

II.1 Generalidades 21

II.2 La familia de microcontroladores
de Motorola..... 22

II.3 La familia de microcontroladores
de Intel..... 23

II.4 La familia de microcontroladores
de Texas Instruments..... 24

II.5 La familia de microcontroladores
de National Semiconductor..... 26

II.6 Selección del microcontrolador y
justificación..... 29

CAPITULO III.

DESARROLLO DEL SISTEMA DE CONTROL CON EL COP402.

III.1 Generalidades del COP420.....	31
III.2 El COP402.....	31
III.3 Generalidades del COP402.....	32
III.4 Características del COP402.....	32
III.5 Descripción de funcionamiento del COP402	34
III.5.1 Memoria de programa.....	34
III.5.2 Memoria de datos.....	35
III.5.3 Lógica interna.....	36
III.5.4 Interrupciones.....	40
III.5.5 MICROBUS.....	41
III.5.6 Interfaz con memoria externa.....	42

CAPITULO IV.

PROGRAMACION DEL COP402 Y FUNCIONAMIENTO DEL SISTEMA DE CONTROL.

IV.1 Programación del COP402.....	45
IV.2 Funcionamiento del Sistema.....	48

CAPITULO V.

TEORIA DE OPERACION,

V.1 El oscilador.....	52
V.2 El circuito de inicialización.....	52
V.3 Descripción de terminales del COP402.....	53
V.4 El "DRIVER" DS3658.....	56
V.5 El regulador de voltaje LM7805.....	58
V.6 El EEPROM: NMC9306.....	59

V.7 El PROM 2716.....	60
V.8 La etapa de salida.....	63
Conclusiones.....	66
Apéndice.....	69
Glosario.....	94
Bibliografía.....	98

INTRODUCCION
Y
OBJETIVO

INTRODUCCION

En los últimos años, el mayor avance en el campo de la electrónica se ha tenido por los microprocesadores (circuitos electrónicos capaces de organizar y procesar la información), los cuales nos permiten manejar sistemas muy complejos de una forma mas sencilla y también ocupando menos espacio.

Sin embargo a partir de la segunda mitad de la década de los ochentas, el surgimiento de los microcontroladores (circuitos electrónicos inteligentes que coordinan dispositivos externos por medio de un programa interno) vino a revolucionar mas los diseños que ya se tenían, con una serie de circuitos que son menos costosos y mas poderosos.

Con los microcontroladores se podrán seguir actualizando diseños que ya se habian hecho con microprocesadores siempre y cuando, la capacidad de memoria y los puertos de Entrada/Salida (E/S) se puedan adaptar.

Con el uso de los microprocesadores se han reducido los costos, ya que muchos circuitos integrados (CI) se pueden reemplazar por unos cuantos, y ahora con los microcontroladores se reducirán aún mas por ser menos los circuitos empleados.

Los microprocesadores fueron empleados inicialmente en campos muy complejos aunque después se aplicaron en casi todo, actualmente se emplean en campos como el campo de las

comunicaciones, la industria, instrumentación, computadoras, equipos médicos y aplicaciones militares entre otras, los microcontroladores a su vez, están siendo empleados en aplicaciones tales como: sistemas de seguridad, copiadoras, televisores, impresoras laser, sistemas de control de motores, equipo médico y armamento por mencionar algunos.

Como se puede apreciar, el campo de los microprocesadores está siendo sustituido en la medida de lo posible por los microcontroladores, generando así una nueva revolución en el campo de la electrónica.

Para diseñar un sistema físico con microcontroladores, es necesario emplear elementos de "HARDWARE", los cuales emplean electrónica de potencia para la etapa de salida y la electrónica digital, para hacer llegar las señales a los dispositivos de salida deseados, así como elementos de "SOFTWARE", los cuales se emplean para poder programar al microcontrolador.

Comercialmente existen varias marcas en la fabricación de microcontroladores, en cada una de ellas dichos circuitos poseen un esquema común, aunque siempre existen diferencias tanto en la capacidad que poseen y las unidades de Entrada/Salida (E/S) que los conforman.

En el primer capítulo de esta tesis, se explicará lo que es un microcomputador, así como las similitudes y diferencias que existen entre los microprocesadores y los

microcontroladores.

Posteriormente, en el segundo capítulo se hará un análisis comparativo de las diferentes marcas de microcontroladores que existen, también se justificará la selección del microcontrolador empleado en el desarrollo del sistema de control de acceso a un área determinada.

En el tercer capítulo, se explicarán en detalle las características del circuito seleccionado, lo cual permitirá corroborar la selección que se hizo del mismo.

En el cuarto capítulo, se desarrollan los aspectos fundamentales de la programación del microcontrolador, es decir el desarrollo de la parte de "SOFTWARE" del diseño y también se explica el funcionamiento del sistema de acceso .

En el último capítulo, se desarrolla la parte del "HARDWARE" del diseño y las características de los principales circuitos empleados.

Por último, se añaden las conclusiones obtenidas del desarrollo de esta tesis, así como un glosario y un apéndice con el conjunto de instrucciones del microcontrolador empleado.

OBJETIVO.

El objetivo de desarrollar esta tesis, es el de diseñar un sistema de control de acceso a un área determinada, el cual funcionará comparando una clave dada por el usuario contra otra que está programada de fábrica, en caso de que ésta sea correcta se permitirá al usuario el acceso al lugar deseado y de lo contrario no, llegando incluso a activarse una alarma si en tres ocasiones no se da la clave correcta.

Para diseñar este circuito, es necesario seleccionar circuitos que sean muy confiables y que tengan alta capacidad de integración para ocupar los menos CI posibles, por lo que se ha seleccionado un microcontrolador el cual actuará como cerebro del sistema, éste deberá cumplir óptimamente con la relación funcionamiento-costo, además de brindar eficiencia y confiabilidad.

Debido a que en el desarrollo del sistema de acceso no necesitamos que exista una interfaz de este sistema con un canal de ocho "BITS" (Unidades de E/S), o una unidad de memoria de ocho "BITS" se utilizará un microcontrolador de cuatro "BITS", ya que son menos costosos y se pueden implementar los requerimientos del circuito.

Cualquier otro microcontrolador de mas de cuatro "BITS" que exista comercialmente puede cubrir los requerimientos para que el funcionamiento del sistema sea el adecuado, sin embargo el costo no sería el óptimo, pudiendo así

implementarse un mejor diseño.

El microcontrolador que se utilizará, debe encargarse de comparar los valores tecleados por el usuario y generar las señales adecuadas para que se permita o no el acceso.

CAPITULO I

*MICROCOMPUTADORES, MICROPROCESADORES
Y MICROCONTROLADORES*

I.1 MICROCOMPUTADORES.

Una computadora es una herramienta capaz de procesar información, esta es quizás la definición mas simple pero también la mas clara, si queremos entender lo que hace una computadora.

Existen básicamente tres tipos de computadora de acuerdo a las señales que procesan las cuales son:

Computadoras analógicas, son aquellas que procesan señales continuas o bien, señales que tienen un número infinito de valores en la unidad de tiempo, aquí la señal de salida es exactamente la misma que la de entrada ya una vez procesada.

Las computadoras híbridas, son aquellas que combinan las operaciones analógicas y digitales.

Computadoras digitales son aquellas que procesan señales discretas o bien, señales que tienen un número finito de valores en la unidad de tiempo, en un computador digital siempre existirá un pequeño error porque en la salida de la señal procesada, o se cae en el rango anterior o en el rango posterior, ya que no existen rangos intermedios.

Al utilizar una computadora digital tendremos mas ventajas que con una analógica, por ejemplo:

El computador digital es mas rápido, es reprogramable,

mas fácil de darle servicio, tiene mayor cantidad de integración de funciones y por último, el hecho de que manejar dos variables es mas fácil que manejar un número mayor de éstas.

Se define como un microcomputador digital a aquél que tiene como Unidad Central de Proceso o (CPU) un microprocesador. Este microprocesador o Unidad Central de Proceso se puede implementar en uno o varios circuitos integrados implementando tecnología MOS, TTL, etc.

La Unidad Central de Proceso contiene toda la circuiteria de Control, una unidad aritmética y lógica o ALU, algunos registros y un contador de las direcciones del programa o (PC). Cuando se ejecuta un programa, la Unidad Central de Proceso manda hacia la memoria la dirección de la localidad de la primera instrucción a ejecutarse; al mismo tiempo la CPU manda hacia la memoria una señal que le permita habilitarse , el código de la instrucción va desde la memoria hacia la Unidad Cental de Proceso y esta instrucción se decodifica y ejecuta .

Después de cada operación que se realiza, el contador de programa se incrementa para direccionar la siguiente instrucción o dato que habrá de ser almacenado en memoria.

Podemos entonces decir que en la Unidad Central de Proceso, es el lugar en el que se efectúan los pasos para procesar la información, es en si el cerebro del sistema microcomputador.

Como se muestra en el diagrama 1, que es de bloques las computadoras digitales constan de tres partes principales: Unidad Central de Proceso, Memoria y Puertos de Entrada/Salida (E/S).

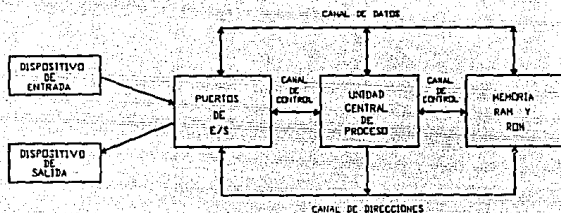


DIAGRAMA 1

I.1.1.- UNIDAD CENTRAL DE PROCESO.

La Unidad Central de Proceso o CPU, es la parte principal de la computadora en la cual se procesa la información que llega en forma de datos y consta a su vez de tres partes:

1) Unidad de Control.- Es la que se encarga de dirigir u organizar lo que sucederá en el microprocesador, es la cual determina cuando se termina una instrucción y cual es

la siguiente instrucción a ejecutarse, dicha Unidad de Control está a su vez constituida de tres partes, las cuales son:

Un registro de instrucciones, el cual es el primer registro al que llegan las instrucciones ya que los ha leído la CPU, éstas llegan aquí en forma de datos; debido a la secuencia de los eventos, los cuales pueden ser lecturas o escrituras en memoria o en periféricos, la CPU sabe si se trata de un dato o de una instrucción. Ya que se encuentra la instrucción o dato en el registro de instrucciones, éste decodifica la instrucción y determina cuantos pulsos se lleva.

Existe también un decodificador de instrucciones y codificador de ciclos de máquina, aquí se determinan los pasos que son necesarios para ejecutar la operación; como cada ciclo se lleva varios pulsos esta sección se encarga de generarlos.

Por último posee un bloque de control y tiempos, en él se generan las señales apropiadas para poder ejecutar las operaciones deseadas. Podemos concluir que la Unidad de Control es el cerebro dentro del cerebro del microcomputador.

2) Unidad Aritmética y Lógica (ALU).- Como su nombre lo indica, ésta es el área del microprocesador donde se efectúan todas las operaciones aritméticas y lógicas necesarias dentro de nuestro proceso, éstas operaciones son

por ejemplo: sumar, restar, comparar, rotar, etc.

Esta unidad aritmética y lógica esta constituida por varias partes, las cuales son:

El acumulador, (1) éste se considera como el registro principal, generalmente contiene uno de los operandos y el resultado de la operación que se efectuó.

El bloque de control de la ALU, el cual indica en si que operación se tiene que efectuar.

Las banderas, las cuales nos indican las condiciones del resultado de cada operación: si es cero, positivo, etc.

Un registro temporal, aquí se carga la información de cualquiera de los registros para efectuar una operación específica.

3) Arreglo De Registros.- Existen básicamente dos tipos de registros en la CPU:

Registros de propósito general, son aquellos registros a los que tiene acceso el usuario, en estos registros podemos guardar un dato por ejemplo, o ya una vez que tenemos un resultado en el acumulador moverlo a uno de estos registros, etc.

Registros de propósito específico, son aquellos que el microprocesador utiliza para si exclusivamente, ya que a nivel de usuario no es útil conocerlos, estos registros son básicamente el apuntador de pila y el contador de programa.

El apuntador de pila, es un registro empleado para guardar una dirección que corresponde al "STACK", el cual es un area de memoria RAM reservada por el usuario, aqui se almacenará información que es útil al microprocesador como lo es por ejemplo:

La dirección a la que habrá de regresar el programa después de que se llamó a una subrutina, o el hecho de almacenar el contenido de un registro que no queremos que cambie o se destruya.

Por otra parte el contador de programa, es un registro que se encarga de almacenar la dirección de la instrucción que se esta ejecutando y que en ese momento habrá de ser leída o ejecutada por la CPU. Después de cada instrucción el contador de programa se incrementa automáticamente, así que nos apuntará a la localidad de la siguiente instrucción o dato en memoria.

I.1.2.- MEMORIA.

La memoria se utiliza para almacenar secuencias de instrucciones o programas que habrán de ejecutarse en la computadora y también para guardar datos que se obtengan del proceso.

Las memorias que se utilizan en los microcomputadores son del tipo no destructivo, es decir que al obtener la información de una posición determinada, ésta no se pierde por el hecho de leerla, cuando al leer una palabra se

destruye la información que contiene en sus localidades se la conoce como memoria destructiva.

La memoria de la computadora está formada por dos grupos:

1) MEMORIA DE ACCESO ALEATORIO (RAM).

Es una memoria del tipo volátil es decir, un tipo de memoria que en el momento que se desconecta la alimentación del circuito se pierden los datos que se tenían, estas memorias se emplean para la lectura y escritura de datos; lo que significa que através de la CPU se puede leer y escribir en este tipo de memorias, a su vez existen dos tipos de memoria RAM:

Memoria RAM Estática, la cual está constituida por un grupo de "FLIP-FLOPS" agrupados en forma de matriz o arreglo y éstos conservarán la información tanto tiempo como la energía este presente en el circuito, el nombre de RAM estática viene del hecho, de que cada localidad en la matriz puede ser accesada, ya sea leída o escrita sin que se afecte otra localidad.

Sin embargo existe un tipo de memoria RAM llamada NOVRAM la cual consiste en combinar una memoria RAM estática y una EEPROM (Memoria Programable De Solo Lectura Borrable Eléctricamente) en el mismo CI, se creará un dispositivo o circuito que puede funcionar como una RAM normal, pero en caso de que haya una falla en la fuente de alimentación, los datos que estaban en la RAM se almacenarán en el EEPROM y

cuando se reestablezca la alimentación al circuito, se puede regresar la información a la memoria RAM.

La ventaja de emplear este tipo de circuitos, es que no se requiere de una alimentación auxiliar en caso de que falle la alimentación principal, con esta innovación se lograría que la memoria RAM fuera del tipo no volátil.

Por su parte la memoria RAM Dinámica, es un tipo de memoria que almacena datos en forma de una pequeña carga que se presenta en un capacitor, sin embargo se necesita lo que se llama un ciclo de refresco ("REFRESH"), cada dos milisegundos o menos.

La ventaja de utilizar este tipo de memorias, es que son mas simples, mas pequeñas, cuestan menos y además requieren menos energía para funcionar, este ciclo de refresco requiere casi siempre el cinco por ciento del tiempo total del procesamiento de un microcomputador.

2) MEMORIA DE SOLAMENTE LECTURA (ROM)

Es una memoria del tipo no volátil, es decir, un tipo de memoria que conserva los datos aún después de desconectar la fuente de alimentación, este tipo de memoria se emplea para la lectura de datos, lo que significa que la CPU solamente podrá leer los datos que se encuentren aquí almacenados, y no podrá escribir otros nuevos.

De acuerdo a la forma en que se programan las memorias

ROM existen varios tipos:

Memorias ROM programadas de fábrica (ROMS), las cuales se mandan programar, y ya una vez programadas no se pueden alterar, es necesario mandar a la fábrica la tabla de verdad y en base a esto se genera un programa que será el definitivo, se necesita mandar a hacer una cantidad considerable para que resulte costeable.

Memorias ROM programables (PROMS), son programadas por el usuario y ya una vez hecho esto no se puede modificar. La diferencia es que aquí el usuario, es el que programa en la memoria lo que él quiere, no necesariamente es algo que ya venía de fábrica o que tuvo que mandar a una fábrica, ya que esto generalmente resulta mas costoso.

El caso de las OTPROMS es un caso particular de memoria PROM, pero en este caso, únicamente se puede programar una vez el circuito y ya no puede ser borrado de ninguna forma a diferencia de los PROMS en los cuales si se pueden borrar por medio de un borrador especial de PROMS, son de la fábrica de MOTOROLA.

Memorias ROM programables y borrables (EPROMS) utilizan tecnología MOS y en estas memorias, una vez que se han programado los datos grabados, se pueden borrar empleando luz ultravioleta aproximadamente veinte minutos; esto se hace através de una pequeña ventana de cuarzo la cual está montada sobre una oblea de silicio, por lo cual esta memoria es reprogramable, esto nos sirve en caso de que deseemos

hacer alguna modificación en nuestro programa.

Por último las memorias ROM programables y borrables eléctricamente (EEPROMS) son igual que en el caso anterior, memorias que ya una vez programadas podemos borrarlas, sin embargo en este caso en vez de emplear luz ultravioleta, utilizamos un voltaje entre veinte y veinticinco volts aproximadamente, ya sea para programar o para borrar; en este caso también empleamos tecnología MOS.

Debido a las características mencionadas, generalmente se trabaja con PROMS, EPROMS o EEPROMS, ya que se tiene el programa definitivo y el volumen de producción lo amerita se mandan programar las ROMS, ésto se hace con el fin de no desperdiciar memorias o hacer gastos innecesarios.

I.1.3.- PUERTOS DE ENTRADA/SALIDA (E/S).

Otra parte de la computadora la constituye la conexión de ésta con el exterior, en general casi todos los sistemas digitales ya sean computadoras o no necesitan unidades de intercambio de información con éste.

Mediante un puerto de entrada podemos recibir datos desde un teclado u otro medio físico y lo que buscamos es que los datos o señales estén bajo el control de la CPU. Un puerto de salida por su parte se utiliza para enviar datos a algún dispositivo de salida tal como un teletipo, una pantalla de video, etc. Cuando se requiere enviar los datos el control se hace através de la CPU y los puertos de

salida.

Existen dos tipos de dispositivos de E/S, que permiten que la información sea controlada por la CPU, estos dispositivos se clasifican de acuerdo a la forma en la que viaja la información en ellos, estos dispositivos son:

1) DISPOSITIVOS DE E/S SERIE.

En éstos la información viaja "BIT" por "BIT", el puerto serie mas empleado es del tipo asincrono, el cual emplea tres tipos de "BITS":

Los "BITS" de inicio, son aquéllos que indican el inicio de una palabra y pueden durar mas de un pulso de información.

Los "BITS" de dato, contienen la información y se pueden tomar como unidad dependiendo de la velocidad del puerto.

Los "BITS" de fin, indican cuando se terminó de enviar una palabra. Todos estos "BITS" se deben enviar por cada palabra que se transmita.

En todo puerto serie existe un circuito convertidor, para que ya que se recibe la información y se manejen señales del exterior, estas puedan convertirse a paralelo.

Dicho circuito convertidor permitirá también, que esas señales que se manejan interiormente en paralelo puedan convertirse a serial para mandarse al exterior.

Por último es necesario un bloque de control que rijá

todas las operaciones del dispositivo de acuerdo a las señales que reciba la CPU y dispositivos de E/S paralelo.

2) DISPOSITIVOS DE E/S PARALELO.

En el Puerto de E/S paralelo la información es enviada o recibida al mismo tiempo, es decir viajan simultáneamente todos los "BITS" , aquí se pueden tener dos o mas puertos de cuatro "BITS" y uno o dos puertos de ocho "bits" los cuales pueden ser de entrada, de salida o bidireccionales , según lo seleccione el programador. Los bloques básicos de un puerto en paralelo son:

Un bloque que interconecte dispositivos de E/S con el sistema para realizar las transferencias de información , además un bloque de control que rija las operaciones del dispositivo y de la CPU .

Es necesario también un bloque para cada uno de los puertos en paralelo para que éstos se manejen como fue establecido, (entrada, salida o bidireccionales); los bloques anteriormente establecidos están interconectados para que la información pueda ser transferida internamente en base a las señales de control.

Como se puede observar en el diagrama 1, se puede definir la estructura de comunicación por medio de tres canales que son:

Canal de direcciones.- Cuando la CPU manda la dirección

de una localidad de memoria que será leída o escrita, lo hace a través de este canal, el cual también lo utiliza la CPU para seleccionar un puerto de entrada o salida en particular. Este canal puede consistir en cuatro, ocho o más líneas en paralelo, es unidireccional ya que solamente la CPU es capaz de direccionar tanto a la memoria como a los periféricos.

Canal de datos.- Como se puede apreciar, este canal es bidireccional, ya que el flujo de datos puede ir o venir de la CPU hacia la memoria y los puertos. Cualquier dispositivo de salida conectado en el canal de datos, debe manejar un tercer estado (estado de alta impedancia), para que pueda estar flotando excepto cuando el dispositivo esté siendo direccionado y leído. El canal de datos consiste en cuatro, ocho o más líneas en paralelo.

Canal de control.- Se encarga de manejar señales que nos permitan habilitar memoria o puertos de E/S para la operación propia del circuito, estas operaciones pueden ser tales como: lectura de memoria, escritura de memoria, lectura de un puerto de E/S, o escritura de un puerto de E/S.

I.2 COMPARACION ENTRE MICROPROCESADORES Y MICROCONTROLADORES.

Como se ha desarrollado a lo largo de este capítulo los microprocesadores también conocidos como CPUs son el cerebro de los sistemas de microcomputadores, es decir son los elementos centrales, los cuales pueden estar constituidos por varios circuitos integrados o en un solo CI.

En el microprocesador se manejan señales de datos, de direccionamiento y señales de control, además las otras líneas del microprocesador representan señales de alimentación y del circuito de temporización.

En los microprocesadores el programador indica la secuencia de operaciones que deben de ocurrir y coloca el programa en la memoria del sistema, el cual inicia en una dirección o localidad determinada donde se encuentra la primer instrucción.

Por su parte los microcontroladores son sistemas completos de microcomputador, que están constituidos en un solo CI e incluyen a un microprocesador, memoria ROM y memoria RAM, puertos de E/S, una lógica interna y circuitos de temporización.

Las diferencias básicas entre los microprocesadores y los microcontroladores son las siguientes:

A) Los microcontroladores poseen una arquitectura de canal dual, tienen un canal de comunicación para cada

dispositivo interno (ROM, RAM, puertos) mientras que los microprocesadores poseen la tradicional arquitectura "VON NEUMANN", (2) la cual tiene cinco características básicas que son:

1.- Un solo elemento de cómputo incorporando una unidad de control, puertos y memoria.

2.- Una organización lineal de localidades de memoria de tamaño fijo.

3.- Un lenguaje de máquina de bajo nivel con instrucciones que realizan operaciones simples en operandos elementales.

4.- Un control de cómputo centralizado secuencial.

5.- Poseen solamente un canal de direcciones con el cual se comunica con los elementos de memoria y puertos.

B) Para aplicaciones de control los microcontroladores son generalmente más eficientes que los microprocesadores.

C) El conjunto de instrucciones entre ambos es diferente ya que las instrucciones del microcontrolador están diseñadas para manejar los periféricos internos, mientras que en el microprocesador se manejan los periféricos externamente.

D) Los microcontroladores son invariablemente circuitos de un solo CI, mientras que los microprocesadores son generalmente circuitos con más de un solo CI.

CAPITULO II

*ANALISIS COMPARATIVO DE LAS DIVERSAS MARCAS
DE MICROCONTROLADORES Y SELECCION DEL CIRCUITO*

II.1 GENERALIDADES.

Los microcontroladores son circuitos electrónicos inteligentes, que pueden ejecutar un programa o una secuencia de instrucciones grabadas internamente y que cuentan con los elementos necesarios para controlar dispositivos externos.

Existen diversos fabricantes de microcontroladores, los cuales se han encargado de la producción y fabricación de dichos circuitos, estos fabricantes emplean diversas tecnologías en la fabricación de microcontroladores, siendo las mas utilizadas: NMOS, CMOS, PMOS, TTL etc.

Las marcas mas importantes en la fabricación de microcontroladores son: Motorola, Intel , Texas Instruments y National Semiconductor

Todos los microcontroladores de cada una de las marcas poseén: Una lógica interna, memoria RAM, memoria ROM y los puertos de E/S, necesarios para implementar funciones de control que tienen un amplio rango de aplicaciones. Aunque dentro de los microcontroladores existen diferentes nomenclaturas, la lógica de ejecución y las instrucciones que emplean son semejantes en todos ellos.

II.2 LA FAMILIA DE MICROCONTROLADORES DE MOTOROLA. (3)

Esta es una familia de microcontroladores que proveen al usuario de una estructura con memoria RAM, memoria ROM, puertos de E/S, temporizadores y un canal de expansión.

Dentro de las características principales de esta familia es que existen mas de veinte tipos de microcontroladores para que puedan adaptarse específicamente a una aplicación determinada. También tienen la ventaja de que existen en versiones de memoria EPROM externa y de ROM programable una sola vez, con lo cual se evita el tener que hacer una mascarilla y reducir así el costo.

También tienen la característica de tener instrucciones para el desarrollo de "SOFTWARE" como son: temporizadores de instrucciones e instrucciones de "SINGLE STEP".

Dentro de esta familia existen versiones de microcontroladores que incluyen convertidores analógico-digitales y puertos seriales con lo que se logra que el diseño siga basado en un solo CI.

La mayoría de estos microcontroladores tienen un oscilador interno que permite la utilización de varias configuraciones de oscilación. Teniendo también un rango de alimentación de 3 a 6 V.

Toda la familia de microcontroladores manejan un mismo conjunto de instrucciones permitiendo así la transportación

de un programa a cualquier miembro de la familia, lo cual brinda al usuario versatilidad, ya que los diseños realizados para un microcontrolador determinado, pueden transferirse a otro de la misma familia sin mayores problemas.

La principal diferencia de la familia de microcontroladores de Motorola con respecto a las otras marcas es que éstos poseen un programa interno de fábrica que permite que sean autoprogramados, esto significa que por el puerto serial le sea transmitido el programa que será grabado en la ROM. También se incluye dentro de este programa una rutina de autoprueba.

II.3 LA FAMILIA DE MICROCONTROLADORES DE INTEL. (4)

Existen dos familias de microcontroladores de Intel, la MCS-48 y la MCS-51. La estructura de estos microcontroladores incluye características como son: una CPU de 8 "BITS", memoria ROM y RAM interna, puertos de E/S y temporizador de 8 "BITS". Tienen un ciclo de instrucciones promedio de 2.5 a 5 microsegundos y un conjunto de mas de noventa instrucciones de uno o dos ciclos máximo cada una.

Esto hace que estas familias tengan un funcionamiento rápido y eficiente, ya que también un setenta por ciento de las instrucciones son de un solo "BYTE" y el resto de dos "BYTES" sin haber de tres "BYTES".

Dentro de esta familia existen miembros con ROM interna, EPROM o ROM externa. Esta familia de microcontroladores maneja un canal multiplexado por lo cual los hace compatibles con los periféricos de la familia de microprocesadores MCS-85.

También los miembros de esta familia requieren de una fuente de alimentación sencilla de 5 volts, poseen compatibilidad en el encapsulado en las versiones de ROM y EPROM.

Tienen además como característica la de tener instrucciones en "SINGLE STEP" por "HARDWARE". Posee ocho niveles de "STACK" y configuraciones de oscilación en: cristal de cuarzo ó oscilador TTL externo.

Intel ofrece sistemas de desarrollo como los emuladores ICS-48 o ICS-51, así como simuladores de ejecución económicos y con programador de EPROMS integrado. Además ofrece para sus sistemas de desarrollo ensamblador y lenguajes de alto nivel como PL/M y PASCAL.

II.4 LA FAMILIA DE MICROCONTROLADORES DE TEXAS INSTRUMENTS.

La familia TMS-7000 es de microcontroladores de un solo CI que maneja ocho "BITS"(5). Esta familia de microcontroladores incorpora CPU, memoria ROM, RAM y EPROM, puertos de E/S, puerto de comunicaciones seriales, temporizadores, manejo de interrupciones y un canal externo de interfaz todo esto en el mismo CI.

En esta familia de microcontroladores existen mas de quince miembros con distintas características, para ser adaptados a una aplicación específica, ya que existen miembros que tienen hasta convertidor analógico-digital (A/D) y dos temporizadores de instrucciones.

Los miembros de esta familia tienen un ciclo de instrucción de 200 nanosegundos, tienen una capacidad de hasta 16 "Kbytes" de ROM. Algunos incluyen también convertidor A/D de 8 canales, dos temporizadores y un temporizador de instrucciones.

Además los microcontroladores de Texas Instruments poseen un puerto serie asincrono y hasta 55 líneas de E/S, que podemos emplear en diversas aplicaciones.

Texas Instruments ofrece módulos de desarrollo como emuladores y sistemas en prototipo, incluyendo para éstos, ensambladores y lenguajes de alto nivel como C y Pascal.

La familia TMS-7000 maneja un conjunto de sesenta y un instrucciones, las cuales son compatibles entre los diversos miembros de las familias.

Debido a que estos microcontroladores son muy poderosos se utilizan en aplicaciones donde se requiere alta velocidad de ejecución y alta velocidad en la transferencia de datos como por ejemplo, en la encriptación y decodificación de señales de video via satélite.

II.5 LA FAMILIA DE MICROCONTROLADORES DE NATIONAL. (6)

Esta familia de microcontroladores provee al usuario de una estructura con memoria ROM, RAM, Puertos de E/S, sistema de tiempo y lógica interna lo cual conforma en sí un sistema completo de microcomputador.

Para una aplicación específica existen más de cuarenta tipos de microcontroladores y la elección dependerá de los requerimientos y necesidades del usuario.

Para cada uno de los microcontroladores de National existe una versión con memoria ROM externa, lo cual nos permite implementar estos circuitos en los diseños en los que el volumen de producción es bajo, evitando así la fabricación de una mascarilla, lo cual generalmente resulta costoso.

En los microcontroladores de ocho "BITS" existen circuitos que incorporan convertidores analógico-digitales, otros incorporan UARTs (Receptor y Transmisor Serial Universal Asíncrono), y otros incluyen PWM (Moduladores por ancho de pulso).

Para toda la familia de microcontroladores existe un conjunto de instrucciones común, así que los programas escritos para un microcontrolador pueden ser transferidos a otro (con el mismo número de "BITS"), incluyendo un pequeño cambio, o sin realizar ninguna modificación.

Además de la gran cantidad de dispositivos que existen, todos los microcontroladores tienen un número de opciones de E/S especificadas al mismo tiempo que el programa, que le permiten al usuario aprovechar al máximo las características de E/S del microcontrolador dentro del sistema.

El sistema de desarrollo es muy poderoso, ya que es capaz de emular a todos los miembros de la familia de cuatro "BITS" sin hacerle modificaciones de "HARDWARE", lo mismo sucede con el emulador de ocho "BITS".

La principal ventaja que presenta esta familia, es que poseé microcontroladores de cuatro "BITS", mediante los cuales se optimiza la relación funcionamiento-costo y además se pueden emplear ampliamente en aplicaciones de control.

A continuación se muestra una tabla que resume las principales características de cada una de las marcas.

1	2	3	4	5	6	7	8	9	10	11	12	13
MOTOROLA	NO	SI	SI	SI	SI	SI	SI	SI	SI	SI	SI	NO
INTEL	NO	SI	SI	SI	SI	SI	SI	NO	NO	NO	SI	SI
TEXAS INSTRUMENTS	NO	SI	SI	NO	SI	SI	SI	SI	NO	NO	SI	SI
NATIONAL SEMICONDUCTOR	SI	SI	SI	SI	SI	SI	SI	SI	NO	NO	SI	NO

1. FABRICANTE

2. PRODUCE MICROCONTROLADORES DE 4 'BITS'

3. PRODUCE MICROCONTROLADORES DE 8 'BITS'

4. SUS CIRCUITOS SE ENCUENTRAN EN UN SOLO CI.

5. FABRICA CIRCUITOS CON ROM EXTERNA

6. FABRICA SISTEMAS DE DESARROLLO

7. FABRICA CIRCUITOS CON PUERTO SERIAL

8. ES NECESARIA FUENTE DE ALIMENTACION SENCILLA

9. CONJUNTO DE INSTRUCCIONES COMUN

10. MICROCONTROLADORES AUTOPROGRAMABLES

11. MICROCONTROLADORES CON Rutina de AUTOPRUEBA

12. VARIOS NIVELES DE 'STACK'

13. EMPLEA LENGUAJES DE ALTO NIVEL EN SUS CIRCUITOS

II.6 SELECCION DEL MICROCONTROLADOR Y JUSTIFICACION.

Para centrar el problema se presentaron anteriormente cuatro marcas distintas de microcontroladores: Motorola, Intel , Texas Instruments y National Semiconductor.

Existen varios factores que se deben considerar al elegir un microcontrolador, siendo el factor decisivo la relación funcionamiento-costo.

El sistema de acceso requiere un total de 15 líneas de E/S y como existen microcontroladores de cuatro "BITS" que tienen hasta 23 líneas de E/S, se puede designar una línea para cada una de las entradas y salidas del circuito teniendo líneas que sobran para probables expansiones del sistema.

También es importante considerar, que es mas sencillo tener un control individual de cada línea en un microcontrolador de cuatro "BITS" que en uno de ocho "BITS" y lo mas importante: Para que pagar mas si se puede hacer lo mismo por menos dinero, en base a esto se ha decidido seleccionar un microcontrolador con dichas características.

Debido a que National Semiconductor es la unica marca que maneja microcontroladores de cuatro "BITS", se tiene el sistema de desarrollo en la Universidad la Salle, y las características de los circuitos cumplen con los requerimientos del sistema de acceso, además de que el costo resulta ser menor, se ha decidido hacer el sistema de acceso

con un microcontrolador de esta familia.

Dentro de la familia de microcontroladores (COPS) de National Semiconductor, se eligió el COP420 debido a las siguientes características:

- Existe un circuito idéntico a éste, el COP402, el cual posee las líneas de interconexión necesarias para conectarle una memoria ROM externa.
- Es el circuito con memoria ROM externa mas pequeño que se puede encontrar en la familia de los COPS.
- 1K x 8 en memoria ROM
- 64 x 4 localidades en memoria RAM
- 23 líneas de E/S.
- 3 niveles de "STACK"
- Utiliza una fuente de alimentación sencilla de 4.5 a 6.3 V
- Ciclo de instrucción de 16 microsegundos.
- Salidas de propósito general que manejan un tercer estado.
- Rango de temperatura de 0 °C a 70 °C.
- Compatibilidad en "SOFTWARE/HARDWARE" con otros miembros de la familia COP400.

Además de que posee todas estas características, el COP420 cumple con los requerimientos necesarios, es decir: Capacidad de memoria tanto ROM como RAM, conjunto de instrucciones , líneas de E/S que permiten emplearlo en esta aplicación y su costo es muy bajo comparándolo con cualquier otro microcontrolador comercial que se pudiera conseguir en México con algún representante de estas compañías.

CAPITULO III

*DESARROLLO DEL SISTEMA DE ACCESO
CON EL COP402*

III. 1 GENERALIDADES DEL COP420.(7)

El COP420 es un microcontrolador canal N contenido en un solo CI fabricado en base a la tecnología MOS, entre sus características además de poseer las de todo microcontrolador de National que son: Memoria ROM, RAM, sistema de tiempo, lógica interna y unidades de E/S, necesarias para aplicaciones de control requiere de una fuente de alimentación sencilla, un conjunto de instrucciones, una arquitectura interna, un esquema de E/S que permite el manejo de una entrada de teclado y manipulación en BCD (Código de decimal a binario).

Posee 23 líneas de E/S y un rango de temperatura de 0° a 70° C.

III.2 EL COP402.

El sistema físico se desarrolló en base al COP402, que es idéntico al COP420 pero con la ROM externa debido a que si se desea emplear el COP420, sería necesario mandarle a fabricar su mascarilla para esta aplicación y como se mencionó anteriormente cuando es bajo el volumen de producción o cuando el programa está teniendo constantes modificaciones, es mejor emplear un microcontrolador con la ROM externa.

III.3 GENERALIDADES DEL COP402.

El COP402 es un microcontrolador fabricado con canal N y tecnología MOS. Tanto la CPU, la RAM, y las unidades de E/S son idénticas a las del COP420 excepto que la ROM es externa.

Se han agregado algunas terminales extras para direccionar la ROM y para introducir los datos de la misma. En un sistema físico el COP402 actuaría exactamente como lo haría el COP420; este beneficio facilita el desarrollo y la programación del COP420; ya una vez que se tiene el programa final y una producción considerable, es recomendable mandar a hacer la mascarilla para la aplicación.

El COP402M es una versión del COP402 con la variación de que posee la interfaz serial de National conocida como MICROBUS.

El COP402 es un microcontrolador que se puede emplear para emular no solamente al COP420 sino que también puede emular al COP410L y al COP411L, esto siempre y cuando se reduzca apropiadamente la frecuencia del reloj.

III.4 CARACTERISTICAS DEL COP402 .

Como se podrá apreciar a continuación, el COP402 posee las mismas características que el COP420, las cuales fueron fundamentales para la elección del microcontrolador que habría de emplearse, éstas son:

COP-120
BAJO COSTO
1 K X 8 DE ROM
64 X 4 LOCALIDADES DE RAM
23 LINEAS DE E/S
3 NIVELES DE 'STACK'
FUENTE SENCILLA DE 4 A 6.3 V
CICLO DE INSTRUCCION 4 US
SALIDAS DE PROPOSITO GENERAL
RANGO DE TEMP: 0 A 70 C
PODEROSO CONJUNTO DE INST
COMPATIBILIDAD DE 'HARDWARE/ SOFTWARE' CON LOS OTROS
COP-100

COP402
BAJO COSTO
1 K X 8 DE ROM
64 X 4 LOCALIDADES DE RAM
23 LINEAS DE E/S
3 NIVELES DE 'STACK'
FUENTE SENCILLA DE 4 A 6.3 V
CICLO DE INSTRUCCION 4 US
CIRCUITO EQUIVALENTE AL COP420 PERO SIN OSCILADOR INTERNO
RANGO DE TEMP: 0 A 70 C
PODEROSO CONJUNTO DE INST.
COMPATIBILIDAD DE 'HARDWARE/ SOFTWARE' CON LOS OTROS
COP400

III.5 DESCRIPCION DEL FUNCIONAMIENTO DEL COP402.

III.5.1 MEMORIA DE PROGRAMA.

La memoria de programa consiste en una memoria externa de 1024 localidades típicamente PROM.

El direccionamiento de la ROM se obtiene por medio de un contador de programa de 10 "BITS". El valor binario del contador selecciona una de las 1024 localidades contenidas en memoria ROM.

Cada ciclo de instrucción el contador de programa se carga con una nueva dirección, a menos que la instrucción siguiente se genere de una transferencia de control por medio de otra instrucción.

Se han implementado tres niveles de subrutinas anidadas por medio de los registros SA, SB y SC, que proveen un "STACK". Las instrucciones que se encuentran en la ROM son buscadas, decodificadas y ejecutadas por el decodificador de instrucciones y la circuitería lógica de control y salto.

Internamente el microcontrolador tiene una estructura semi-transparente en la ROM formada por páginas, bloques y capítulos. Una página se compone de 64 localidades continuas de la ROM, un bloque se compone a su vez de cuatro páginas continuas y solamente existen capítulos cuando se tienen más de 2048 localidades de memoria ROM.

III.5.2 MEMORIA DE DATOS.

La memoria para datos consiste de 256 localidades de memoria RAM organizada en cuatro registros para datos de 16 dígitos de cuatro "BITS" cada uno.

El direccionamiento de la RAM se lleva a cabo por medio de un registro llamado B (de seis "BITS") cuyos dos "BITS" superiores pertenecen a la parte Br del registro B, esta parte selecciona uno de los cuatro registros para dato y los cuatro "BITS" inferiores conforman la parte Bd del registro B, mediante la cual se selecciona uno de los 16 dígitos de cuatro "BITS" que se encuentran en cada uno de los cuatro registros para datos.

Mientras que el contenido del dígito de cuatro "BITS" seleccionado de memoria RAM, es usualmente cargado desde el acumulador o intercambiado con él, también a su vez puede ser cargado o intercambiado con el registro Q.

El registro Bd, sirve también como un registro fuente para enviar cuatro "BITS" de datos, directamente a las salidas del puerto D.

Como se puede apreciar la memoria de datos es organizada de forma matricial, en la cual cada renglón es llamado registro y cada columna es llamada dígito, por lo cual la memoria RAM en el COP402 y en general en todos los microcontroladores de National, no es direccionada por el contador de programa sino que tiene su propio registro para

direccionarla, que es el registro B.

III.5.3 LOGICA INTERNA.

El acumulador o registro A; es un registro de cuatro "BITS" el cual se utiliza como fuente y destino de la mayoría de las entradas, salidas , operaciones aritméticas y lógicas así como las operaciones de acceso de datos en memoria.

El acumulador también puede utilizarse para cargar al registro B; para cargar y leer cuatro de los ocho "BITS" del "LATCH" Q, para leer cuatro de los 8 "BITS" del puerto L de E/S y para generar intercambio de datos con el registro SIO.

Poseé además un sumador de cuatro "BITS" que se encarga de realizar las funciones aritméticas y lógicas del COP y almacenar los resultados. (También escribe un "BIT" de acarreo al registro C el cual generalmente indica la condición aritmética de desbordamiento, "OVERFLOW").

Existen además cuatro entradas de propósito general que van de INPUT 3 a INPUT 0 (IN3 a IN0), las cuales pueden ser seleccionadas por una opción de programación mascarillable, tal como llamado de lectura, selector de circuito y llamado de escritura, siendo éstas IN1, IN2 e IN3 respectivamente, las cuales se utilizan en las aplicaciones de la interfaz serial.

El registro D proveé cuatro salidas de propósito general y es utilizado como el registro para el contenido de

Bd de cuatro "BITS".

El registro G provee cuatro terminales de propósito general bidireccionales como puertos de E/S y G0 se puede emplear como una señal de "READY", en el caso de las aplicaciones de la interfaz serial o "MICROBUS".

El registro Q es un registro interno con "LATCH", utilizado para retener datos cargados hacia Bd y el acumulador desde el puerto L de E/S o viceversa, esto se podrá siempre y cuando los "DRIVERS" de L sean habilitados por medio del programa de control.

Los ocho "DRIVERS" de L cuando son habilitados, habilitan el contenido del "LATCH" Q hacia el puerto L de E/S. También el contenido de L puede ser leído directamente en los registros A y Bd.

El registro SIO funciona como un registro de corrimiento de cuatro "BITS" SISO (Entrada serie-salida serie) o como un contador binario dependiendo del contenido del registro EN. El contenido del registro SIO puede ser intercambiado con el acumulador permitiendo así la entrada o salida de datos en forma serial.

El registro EN es un registro interno de cuatro "BITS" que se carga bajo el control del programa por medio de la instrucción LEI (Carga el registro EN inmediatamente) . El estado de cada "BIT" de este registro selecciona o deselecta una característica en particular asociada con

cada "BIT" de este registro:

1.-El "BIT" menos significativo del registro EN, EN0, selecciona el registro SIO ya sea como un registro de corrimiento de cuatro "BITS" o como un contador también de cuatro "BITS". Cuando la terminal ENABLE 0 (EN0) se pone en uno lógico, SIO es un contador binario asíncrono, decrementando su valor cada vez que hay una transición de uno a cero en la entrada SERIAL INPUT (SI), cada pulso debe ser de al menos dos ciclos de instrucción de ancho, siendo el ancho de cada uno de 4 microsegundos (us).

La salida Serial Output (SO) es igual al valor presente en la terminal ENABLE 3 (EN3), cuando EN0 se pone en cero lógico el registro SIO es un registro de corrimiento serial que recorre cada "BIT" hacia la izquierda por cada ciclo de instrucción. El dato presente en la terminal SI va dentro del "BIT" mas significativo del registro SIO.

2.-Con la terminal ENABLE 1 (EN1) en uno lógico IN1 es habilitada como una interrupción de entrada, inmediatamente después que se genera una interrupción, EN1 se pone en cero para deshabilitar interrupciones posteriores.

3.-Con la terminal ENABLE 2 (EN2) en uno lógico los "DRIVERS" del puerto L son habilitados para enviar los datos de Q hacia este puerto y pone al puerto L en un estado de alta impedancia. Si se utiliza la opción del "MICROBUS" EN2 no afecta a los "DRIVERS" de L.

4.-EN3 junto EN0 afecta a la salida SO, la tabla que se da a continuación nos da un resumen de los modos de operación que se tienen al trabajar conjuntamente EN0 y EN3:

EN3	EN0	SI0	SI	SO
0	0	REGISTRO DE CORRIMIENTO	ENTRADA AL REGISTRO	0
1	0	REGISTRO DE CORRIMIENTO	ENTRADA AL REGISTRO	SALIDA SERIAL
0	1	CONTADOR BINARIO	ENTRADA AL CONTADOR	0
1	1	CONTADOR BINARIO	ENTRADA AL CONTADOR	0

III.5.4 INTERRUPCIONES.

Las siguientes características están asociadas con el procedimiento de la interrupción IN1 y el programador debe considerarlas cuando utilice interrupciones:

a. Cuando se generó la interrupción se guarda la siguiente dirección del contador de programa en el "STACK" y esto provoca que se vaya recorriendo el contenido de los otros registros del "STACK" al siguiente nivel inferior, es decir; (PC+1 -> SA -> SB -> SC) el contenido previo de SC se pierde, como se observa a continuación, en el diagrama 2.

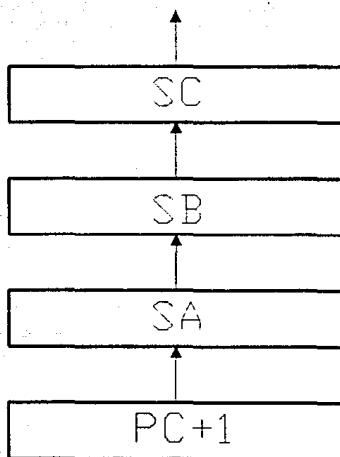


DIAGRAMA 2

b. Una interrupción será reconocida solamente después de que se presenten las siguientes condiciones:

- 1.-EN1 debe de estar en uno lógico
- 2.-Debe de ocurrir una transición de uno a cero de al menos dos ciclos de instrucción en la entrada IN1.
- 3.-La instrucción que se está ejecutando se debe de completar

c. La primera instrucción de la rutina de interrupción la cual se debe de situar en la dirección OFFh debe de ser un NOP.

d. La instrucción LEI (Carga al registro EN inmediatamente), se debe de colocar antes de la instrucción de retorno de interrupción; RET, para reactivar las interrupciones.

III.5.5 INTERFAZ SERIAL DE NATIONAL (MICROBUS).

El COP402M puede ser utilizado como un dispositivo periférico IN1, IN2 y IN3 son entradas de propósito general que al hacerlas compatibles con el "MICROBUS" se comportan como llamadas de lectura, selector de circuito y llamada de escritura respectivamente.

INPUT 1 (IN1) viene siendo una señal de RD\ y un cero lógico en esta entrada causará que el "LATCH" Q habilite al puerto L como entrada al microprocesador.

INPUT 2 (IN2) es una señal de CS\ y un cero lógico en esta línea selecciona al COP402M como un dispositivo

periférico para habilitar la operación de las líneas de RD\ y WR\ lo cual permitirá la selección para varios componentes periféricos.

INPUT 3 (IN3) se utiliza como una señal de WR\ y un uno lógico en esta línea, permitirá que entren datos desde el puerto L a través de los "LATCHES" de Q hacia el microcontrolador.

Lo que se mencionó se puede observar en el diagrama 3 :

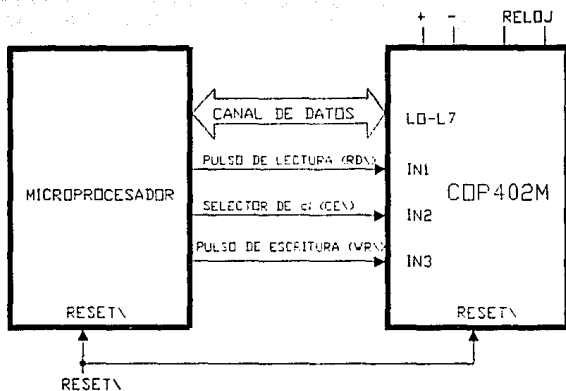


DIAGRAMA 3

III.5.6 INTERFAZ CON MEMORIA EXTERNA.

El COP402 y COP402M están diseñados para utilizarse con una memoria de programa externa. Esta memoria puede implementarse utilizando cualquier tipo de dispositivo con

las siguientes características:

1. Acceso aleatorio.
2. Salidas compatibles con TTL y tercer estado.
3. Entradas compatibles con TTL.
4. Tiempo de acceso de un microsegundo máximo.

Normalmente estos requerimientos se cumplen al utilizar PROMS bipolares o MOS.

Durante la operación del circuito, la dirección de la siguiente instrucción se envía através de P9, P8 y de IP7 a IP0, durante el tiempo que la señal AD/DATA\ tiene un uno lógico.

La dirección en las líneas de IP se guardan en un "LATCH" externo en la transición de uno a cero de la línea de AD/DATA\, P9 y P8 son líneas destinadas a direccionar lo cual evita que se tengan que mandar al "LATCH".

Cuando la señal de AD/DATA\ tiene un cero lógico la salida de la memoria se maneja por compuertas de IP7 a IP0 formando un canal de entrada, esta señal tiene un ciclo de instrucción del cincuenta por ciento y se especifica si las líneas de IP como direcciones de salida o como canal de entrada de datos. Un diagrama de bloques simplificado se muestra a continuación:

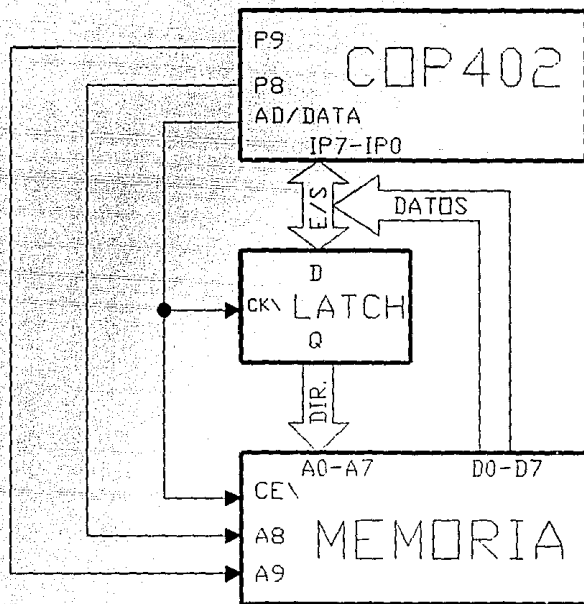


DIAGRAMA 4

CAPITULO IV

*PROGRAMACION DEL COP402 Y FUNCIONAMIENTO
DEL SISTEMA DE CONTROL DE ACCESO*

IV.1 PROGRAMACION DEL COP402.

Durante el desarrollo del sistema de acceso se elaboraron las siguientes subrutinas, de las cuales se explica a continuación su funcionamiento:

Rutina para inicializar la memoria RAM.- Debido a que al alimentar al sistema no se puede asegurar que todos los valores en las localidades de la RAM sean cero, es conveniente inicializar esta área para evitar un posible error causado por un valor previo en las localidades de memoria.

A continuación se da un ejemplo de como emplear el conjunto de instrucciones para que se ejecute una rutina como lo es la rutina de inicializar las localidades de memoria RAM.

```
                CLRA
R0:             LBI      0,0
R1:             LBI      1,0
R2:             LBI      2,0
R3:             LBI      3,0
A:             CLRA
                XIS
                JP      A:
                RET
```

- CLRA es la primera instrucción que debe llevar todo programa, debido a la estructura interna del

microcontrolador.

- LBI 0,0 apunta al primer registro y primer dígito de la memoria RAM, es necesario apuntar a esta localidad para ir inicializando la memoria registro por registro, debido a las características de la instrucción (ver apéndice), la siguiente instrucción a ejecutar debe ser distinta de LBI.

- CLRA inicializa nuevamente el acumulador, pero en esta ocasión para intercambiar su contenido con cada una de las localidades de la memoria RAM, ya que ésta se irá incrementando con la siguiente instrucción.

- Con XIS la porción Bd del registro B se incrementa en uno, apuntando ahora al segundo dígito o columna del primer registro, posteriormente se intercambia el contenido del acumulador con el contenido de la localidad de memoria RAM direccionada por el registro B.

Debido a lo anterior era necesario almacenar un cero en el acumulador, ya que no sabemos lo que había en la memoria RAM y al intercambiar nuevamente el acumulador podría ya no haber un cero y no se inicializaría la memoria RAM.

- Con JP A: regresamos a la etiqueta A, lo cual permitirá seguir intercambiando el contenido del acumulador que siempre será cero con lo que se tenga en memoria RAM, esta condición persistirá hasta que la porción Bd sea mayor a quince, ya que por características de la instrucción en el momento que ésto se cumpla saltará la siguiente instrucción

que en este caso es JP A:

- Con RET regresará la ejecución al programa principal, el cual hará que se ejecute ahora la subrutina R1, limpiando el registro uno, después R2 y por último R3 lo cual hará que se hayan inicializado los dieciséis dígitos de los cuatro registros que conforman el total de la memoria RAM.

Rutina de lectura del teclado.- En esta rutina se efectúa un barrido constante del teclado, para detectar si alguna de las teclas está siendo oprimida.

Rutina de conteo de cinco segundos.- Fue generada para funcionar como un temporizador que hará que si en un tiempo mayor de cinco segundos no se oprimen dos teclas consecutivamente se reinicie el conteo del sistema perdiendo la información que se había insertado.

Rutina de lectura del EEPROM.- En esta rutina se establece una comunicación entre el microcontrolador y el EEPROM en la cual el COP habrá de leer la clave correcta para compararla con la que insertó el usuario , de ser así habilitará las otras funciones del sistema de acceso.

Rutina de escritura del EEPROM.- Se generó para que el microcontrolador pudiera escribir en el EEPROM una clave personal que el usuario suministró, una vez que tecleó la clave fija adecuadamente.

Además de estas subrutinas hubo instrucciones que al realizar el programa permitieron habilitar las salidas y que

el microcontrolador pudiera generar las señales apropiadas para que el sistema de acceso pudiera funcionar bien.

IV.2 FUNCIONAMIENTO DEL SISTEMA.

El sistema de acceso opera de la siguiente manera:

El usuario teclea una clave de seis dígitos de un teclado de con ocho teclas, si deja pasar mas de cinco segundos entre oprimir una tecla y la siguiente, el sistema se reiniciará y se apagará el teclado siendo necesario volver a darle todos los datos.

Al oprimir la primer tecla, el teclado se iluminará y se apagará si pasan cinco segundos sin que se oprima otra, el teclado permanecerá iluminado mientras se teclea la clave, si ésta se teclea erróneamente la iluminación del teclado se apagará inmediatamente, por el contrario si la clave es correcta el teclado permanecerá encendido cinco segundos después de que se dió el último dígito.

Si la clave que se tecleó es la correcta tendrá cuatro segundos para empujar la puerta y poder entrar al lugar deseado, si en esos segundos no empuja la puerta será necesario volver a teclear la clave.

Se tienen tres oportunidades para dar la clave correctamente, esta cuenta se mantendrá aunque la alimentación del sistema se desactive, si el usuario se equivoca en tres ocasiones, la señal de alarma se activará.

Si antes de que se llegue a las tres veces se teclea la clave correcta, la cuenta se reiniciará, teniendo otras tres oportunidades antes de que se active la alarma.

Para agregar una clave personal será necesario en primer lugar dar la clave fija correctamente, después de esto, es necesario oprimir la primer tecla y a continuación dar la nueva clave.

La clave personal que se programó se podrá borrar dando nuevamente la clave, ya sea la fija o la temporal y oprimiendo la tecla uno sin oprimir ninguna secuencia de dígitos después, independientemente de que se dé una clave personal, la clave programada de fábrica seguirá existiendo por tiempo indefinido.

Si al teclear la clave se oprimen dos o más teclas simultáneamente, para considerar la secuencia de números el sistema tomará a la que se haya oprimido primero y perderá los valores que se hayan tecleado después, esto se debe a que para introducir cada uno de los valores es necesario oprimir una tecla y hasta que ésta no se deja de oprimir no se reconoce otra.

Debido a lo anterior, cuando se oprime una tecla por tiempo indefinido el sistema de acceso solamente reconoce un valor y no una secuencia de los mismos.

Además de usar el sistema de acceso, el usuario tiene la opción de poder entrar al área deseada utilizando la

llave del lugar, si se presenta un problema de tipo eléctrico que interrumpa el suministro de energía al sistema éste dejará de operar, sin embargo cuando se reestablezca el suministro, la información no se perderá manteniendo la cuenta, inclusive de las veces que una persona se ha equivocado al introducir la clave y la también conservará la clave temporal.

Para evitar el problema de que el sistema de acceso deje de funcionar cuando se ha suspendido el suministro de energía, es posible añadir como una alternativa en el diseño, una batería extra, la cual entrará en operación cuando se detecte que no hay energía en el sistema, y esto nos permitirá que el sistema opere en forma indefinida.

Este sistema de acceso es altamente confiable ya que lo único que podría causar un error cuando se oprime la clave, es que resultara afectado por un campo electromagnético o que algún dispositivo cercano introdujera ruido eléctrico, sin embargo aunque esto se presentara el sistema posee protecciones en cada una de las conexiones que tiene con el medio externo y de no ser por esta causa el sistema es prácticamente infalible.

Se pueden fabricar tantos sistemas de acceso como se deseen ya que los dispositivos que conforman al sistema se encuentran comercialmente, e inclusive se encuentran piezas de fabricación nacional como es el caso de relevadores de 12V-6A.

Para el usuario el sistema responde prácticamente en forma instantánea, ya que en cuanto se termina de teclear la clave el microcontrolador ejecuta la subrutina de comparación de la clave con la clave programada de fábrica y si ésta es correcta, inmediatamente da las señales de salida necesarias para habilitar a los mecanismos de la etapa final y poder así acceder al lugar deseado.

CAPITULO V

TEORIA DE OPERACION.

EL OSCILADOR.

La circuitería de reloj se fija de tal forma que la CPU tenga un ciclo de instrucción de 4 us, para lograr esto, inicialmente se tiene un cristal de cuarzo que genera una frecuencia de 3.996 MHz, la cual se retrasa en dos ocasiones debido a un CI 74LS04, este contiene inversores y se pasa posteriormente al CI 7493, el cual es un contador binario de cuatro "BITS" que posee cuatro "FLIP-FLOPS", este circuito se encarga de dividir la frecuencia que genera el cristal entre cuatro, con la cual trabajará el microcontrolador.

V.2 EL CIRCUITO DE INICIALIZACION.

La lógica de "RESET" hará que el COP inicie su funcionamiento siempre y cuando el tiempo inicial de suministro de la fuente sea mayor de un microsegundo y menor de un milisegundo, si el tiempo de suministro es mayor de un milisegundo, es necesario agregar una circuitería externa que genere dicha señal en la terminal de RESET\, la cual estará formada por una resistencia, un diodo y un capacitor tal y como se muestra en el diagrama 5.

La entrada de "RESET\" esta configurada como una entrada de "SCHMITT TRIGGER" (debido a que se fija un umbral) y en caso de que no se utilice debe conectarse a VCC

La inicialización se producirá siempre y cuando esté presente un cero lógico en la entrada de "RESET\", si esto sucede el contador de programa se carga con el valor cero

indicando así la dirección cero de la ROM y a su vez los registros se inicializan (se cargan con cero).

La salida del circuito se habilita como una señal de sincronía y proveerá un pulso cada ciclo de instrucción, no hay que olvidar que la primera instrucción que debe existir siempre en la dirección cero debe ser CLRA (Cargar el acumulador con cero).

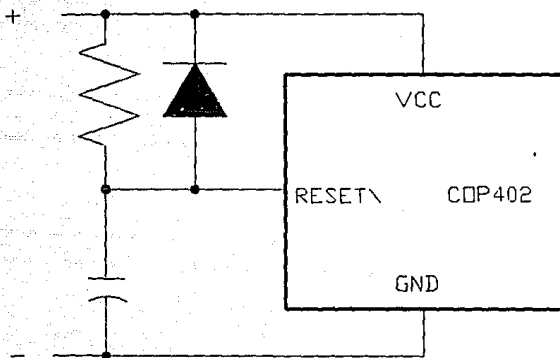


DIAGRAMA 5

V.3 DESCRIPCION DE LAS TERMINALES DEL COP402.

Como se puede observar en el diagrama 6 el cual es del COP402, las terminales de IP0 a IP7 se emplean inicialmente para mandar las direcciones de las localidades de memoria ROM que habrán de ser accedadas por el COP y posteriormente será através de estas líneas donde será mandado el dato contenido en dicha localidad de memoria; todo depende del

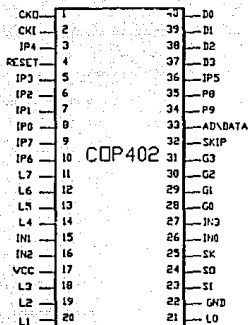


DIAGRAMA 6

estado de la terminal AD/DATA\ del COP.

Através de las terminales P8 y P9 se mandan las dos cifras mas significativas de la localidad de memoria que se está direccionando, debido a que estas dos direcciones no se utilizan para algo mas, no necesitan de un "LATCH" como en el caso de las otras líneas que van de IP0 a IP7.

Como ya se mencionó las terminales de CLCK INPUT (CKI) CLOCK OUTPUT (CKO) son la entrada y la salida de reloj respectivamente, y entre las terminales de VCC y GND es donde se suministra el voltaje para el COP.

Por otra parte las terminales de SERIAL INPUT (SI), SERIAL OUTPUT (SO) y SERIAL CLOCK (SK) se emplean para la comunicación serial; SI es la terminal de entrada serial o entrada del contador, SO es la terminal de salida serial y

SK es la terminal donde se encuentra la señal de reloj controlada lógicamente, SO y SK pueden emplearse como señales de propósito general. En la terminal de SKIP es en la que se indica si se presentó una condición de salto.

Por último tenemos cuatro puertos, los cuales son:

El puerto L, el cual es un puerto bidireccional de ocho "BITS", es bidireccional de E/S de acuerdo a como lo programe el usuario y maneja un tercer estado, en este puerto L es en el cual se registrará si alguna de las teclas está siendo oprimida, a través del microcontrolador se realizará el barrido del teclado de forma constante, es decir, en todo momento existen una serie de valores en cada una de las terminales del puerto L, estos valores se mantendrán mientras no se oprima ninguna tecla.

El puerto IN, es un puerto que posee cuatro entradas de propósito general, aquí se encuentra el común del teclado o bien la línea a través de la cual se cerrará circuito en el teclado indicando así que una tecla ha sido oprimida, en esta terminal siempre existirá un uno lógico presente, en el momento que se oprima una tecla se registrará un cero lógico que será aprovechado a nivel "SOFTWARE" en la lógica interna del circuito.

El puerto L y el puerto IN están relacionados, ya que todas las teclas se encuentran en el puerto L, sin embargo la línea del COP que detecta si alguna de las teclas fue oprimida se encuentra en el puerto IN, como se puede observar en el plano que se encuentra al final del capítulo,

cada una de las terminales llevan una protección de una resistencia y un diodo zener, el cual actuará como un regulador de voltaje y evitará que si en un momento dado se induce un voltaje elevado, éste dañe al COP.

Existe además un puerto G, el cual es un puerto bidireccional con cuatro líneas de E/S, en este puerto estarán las señales que controlan la iluminación del teclado y estará también la señal de alarma; estas señales se deben conectar a un "DRIVER" el cual nos permitirá conectar las señales del microcontrolador con la etapa de salida.

Finalmente tenemos el puerto D, este puerto posee cuatro salidas de propósito general, una de estas se empleará para la señal de abrir la puerta una vez que se ha dado la clave correctamente, esta señal también recibirá ganancia en corriente para acoplarse a la etapa de potencia del sistema.

EL "DRIVER" DS358. (8)

En el sistema de acceso se cuenta con un "DRIVER" el cual es un CI, el DS3658 de National, el cual nos dará ganancia en corriente para poder acoplar las salidas del COP, la corriente de salida que suministra el circuito puede llegar a ser hasta de 600 mA por cada salida, como se tienen cuatro en cada DS3658 el circuito es capaz de proveer hasta 2.4 amperes.

Este circuito se emplea en aplicaciones que requieren bajo voltaje de alimentación y alta corriente de salida, este circuito posee además salidas a "COLECTOR ABIERTO", las

cuales como su nombre lo indica nos permite fijar el voltaje de salida a un valor que nosotros deseemos siempre y cuando no se excedan setenta volts, por características eléctricas de fabricación.

En el caso del diseño del sistema de acceso el "DRIVER" se emplea para que las salidas del COP que manejan 2 o 3 mA. típicamente, puedan adaptarse a las entradas de la etapa de potencia las cuales requieren aproximadamente cuarenta miliamperes para funcionar, además el circuito posee la característica, de que para incrementar la capacidad de corriente de salida, las salidas pueden colocarse juntas.

Podemos observar que existen diversas terminales de las cuales es importante mencionar:

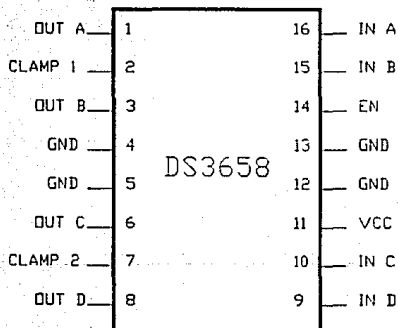


DIAGRAMA 7

Através de las terminales de IN del circuito se colocan las señales de salida del COP y por medio de las terminales de OUT obtenemos las señales que ya tienen una ganancia en corriente y se mandarón a la etapa de salida.

Las terminales de CLAMP pueden conectarse al positivo de la fuente de alimentación si se desea una ganancia de hasta 800 mA, por último las terminales de VCC y GND son en las cuales se conecta la alimentación.

V.5 EL REGULADOR DE VOLTAJE LM7805 (9)

El sistema de acceso cuenta además con un regulador de voltaje, el LM7805, este regulador tiene tres terminales y entrega 5 volts a la salida como su nomenclatura lo indica, también posee un limitador de corriente interno y entrega un máximo de 0.5 Amperes, estos reguladores pueden ser empleados en una amplia gama de aplicaciones, y tienen la ventaja de que pueden ser utilizados con componentes externos para obtener voltajes y corrientes ajustables a la salida.

EL LM7805 tiene un rango de temperatura de $0^{\circ}\text{C} < T_A < 125^{\circ}\text{C}$, el voltaje de alimentación varía de 5 a 24 volts por características eléctricas de fabricación, y como podemos observar en el diagrama 8; en la terminal IN es en la cual se suministra el voltaje de entrada al circuito, la terminal GND debe conectarse directamente a tierra o potencial cero y en la terminal de OUT tendremos la salida del mismo.

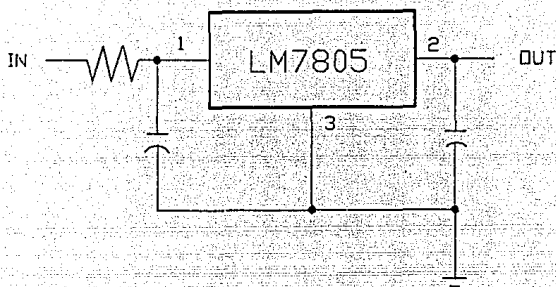


DIAGRAMA 8

EL EEPROM: NMC9306. (10)

Para almacenar las claves del sistema de acceso se emplea un circuito de EEPROM: el NMC9306, el cual, como se observa en el diagrama 9, es un circuito de ocho terminales que se utiliza para establecer comunicación serial con el COP.

Dentro de las terminales que conforman el EEPROM, la terminal de CS se empleará cuando éste vaya a ser leído o escrito por el COP; mediante la terminal de DO, se envían los datos que habrán de entrar al COP; a través de la terminal de DI, entrarán los datos al EEPROM que envía el COP y con la pata de SK se sincronizarán las dos señales de

SK, tanto la del COP como la del EEPROM.

Por medio de las terminales de VCC y VSS se alimenta al circuito con el voltaje de positivo y tierra respectivamente, finalmente existen dos terminales denotadas como NC, en las cuales, como su nombre lo indica, no se utilizarán.

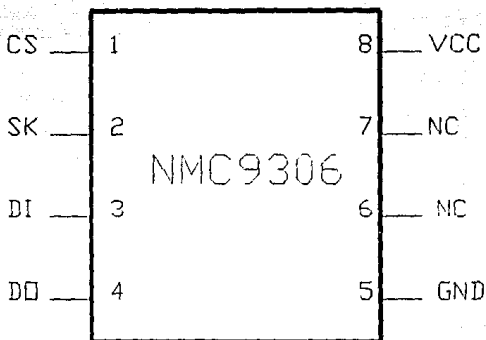


DIAGRAMA 9

V.7 EL EPROM 2716.

En el sistema de acceso existe además una memoria EPROM que por ser un COP con ROM externa es en el cual residirá el programa a ejecutarse; este circuito es el 2716, esta EPROM es una memoria de 2K x 8, la cual tiene un tiempo de acceso bajo, aproximadamente 450 ns, además tiene bajo poder de

disipación alrededor de 525 mW, se utiliza a una tensión de +5 volts únicamente, es completamente estático y compatible con TTL durante la lectura y programa.

El 2716 es una EPROM programable eléctricamente y borrable con luz ultravioleta, el 2716 es además la primera EPROM que presenta un modo de reserva estático, lo cual reduce la potencia disipada sin aumentar el tiempo de acceso.

El 2716 ofrece uno de los modos mas simple y rápido para programar EPROM, el cual consiste en programar con un pulso TTL, es decir no hay necesidad de pulsos con mayor voltaje ya que todos los controles de programación están dirigidos por las señales TTL.

En el diagrama 10 podemos observar que dentro de las terminales que conforman a este circuito, se encuentran las que van de la salida 0 a la 7 (00 a 07), éstas son las salidas de la EPROM através de las cuales se envían los datos de las señales direccionadas.

Las terminales del EPROM de A0 a A11, se utilizan para direccionar los 2K de memoria ROM; por medio de la expresión 2^n obtenemos el número de localidades de memoria disponibles, aquí n representa el número de direcciones contenidas en el circuito, para el desarrollo del sistema de acceso se emplearán solamente de A0 a A10 por tener únicamente 1 K de memoria de programa de acuerdo al COP que se empleó.

Cuando hay un cero lógico en la terminal de CE\ y un

uno lógico en la terminal de OE\ el microcontrolador habilita al dispositivo para ser leído, cuando por el contrario CE\ está en uno lógico y OE\ tiene un cero lógico se seleccionará al circuito para programarlo.

Por su parte las terminales de VCC y GND son en las cuales se debe suministrar el voltaje de alimentación, mientras que en la terminal de VPP es en la que se suministra el voltaje, (generalmente de cinco volts) necesario para programar la memoria.

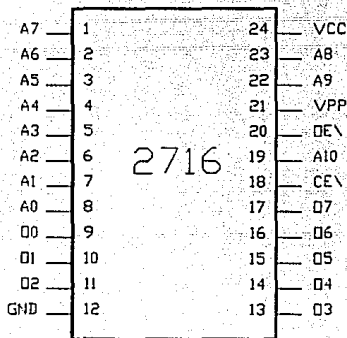


DIAGRAMA 10

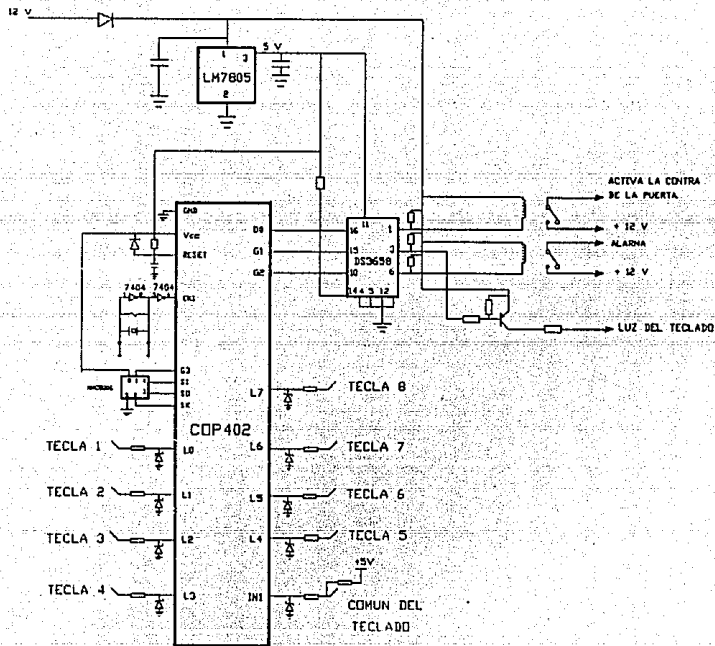
V.8 LA ETAPA DE SALIDA.

La última parte que integra el sistema de acceso es la etapa de salida, la cual está formada por relevadores que operan a 12 volts y estos son capaces de soportar hasta 6 amperes, los relevadores están polarizados como se puede observar en el plano que se encuentra al final del capítulo, de tal forma que al presentarse las señales que salen de los "DRIVERS", quedan polarizados adecuadamente, generándose así una diferencia de potencial, la cual hará que se forme un campo magnético, llevando al contacto normalmente cerrado y al contacto normalmente abierto al estado opuesto.

Al polarizar los relevadores adecuadamente, la señal de iluminación del teclado se activará y la que activa al mecanismo para que se abra la puerta se encontrará presente a la salida, demandando la cantidad de corriente necesaria para que se pueda abrir la puerta y acceder al lugar deseado, desde luego siempre y cuando se haya tecleado la clave correctamente.

En todo el sistema se utilizan dispositivos discretos como lo son resistencias y capacitores, con el fin de generar las protecciones y que existan los voltajes debidos en las terminales de los circuitos; dentro de los componentes activos que se emplearon, están los diodos, los cuales hacen que la corriente vaya solamente en un sentido y que no pueda regresarse (por ser unidireccionales), lo cual podría dañar a los circuitos; como ya se mencionó, existen

también diodos zeners de 4.7 volts, los cuales evitan que se induzcan voltajes mayores a éste, que puedan dañar al sistema y específicamente al microcontrolador.



CONCLUSIONES

CONCLUSIONES.

El hecho de diseñar un sistema de control con microcontroladores debe de considerarse como una de las mejores opciones sino es que la mejor.

La forma en la que se diseñó este sistema de acceso es la que ocupa menos espacio; además utiliza la tecnología mas avanzada que existe actualmente en el mercado y también es la mas confiable y segura que se puede tener.

Si se desea diseñar un sistema como éste sin aplicar microcontroladores, resulta muy difícil y laborioso, además de que muy probablemente no serian tan confiables, sin embargo al emplear microcontroladores se simplifica bastante el grado de dificultad del diseño y el costo también se reduce.

Es importante mencionar que cuando se desea diseñar un sistema de control como en este caso el emulador, es una herramienta muy poderosa que permite que el diseño sea mas rápido, mas eficiente y poderoso al mismo tiempo.

Al utilizar conjuntamente diversas áreas de la electrónica y la programación del microcontrolador se obtienen mejores resultados, que si se empleara solamente una de dichas áreas, llegando casi a ser imposible el poder realizar un diseño de con estas características.

Si se emplea adecuadamente el conjunto de instrucciones

del microcontrolador y además se aprovechan las características de las instrucciones así como la Lógica de Control y de Salto se pueden lograr programas muy versátiles y ahorrar la memoria del programa disponible, lo cual nos permite utilizar circuitos de memoria con menor capacidad y reducir también con esto el costo del diseño.

Cuando se realiza un diseño con microcontroladores es necesario proteger las patas del microcontrolador cuando éstas tengan relación con el medio externo, ya que de no realizarlo se pueden inducir voltajes que dañen al circuito.

Para acoplar la etapa de potencia con la etapa lógica, que es la que genera el microcontrolador, es necesario que se suministre la corriente necesaria, lo cual se logra mediante los "DRIVERS" o mediante los transistores; ya que de no ser así la etapa de potencia demandará mas corriente y debido a sus características eléctricas el microcontrolador no podrá suministrarla trayendo con esto la posible destrucción del circuito.

Cada que se vaya a escoger un microcontrolador, se deben de considerar características importantes como las siguientes:

- Número de "BITS".
- Costo del circuito (Es directamente proporcional al número de "BITS").
- Disponibilidad en el mercado.
- Tamaño físico del empaquetado.

- Capacidad de memoria suficiente para nuestra aplicación.
- Número de líneas de E/S para atender a todos los circuitos periféricos.

Si alguno de los puntos anteriores no se consideran como un primer acercamiento al microcontrolador que se requiere, el diseño que se realice no será el óptimo, por lo cual se debe considerar el mas completo y menos costoso de todos los circuitos que existan en el mercado.

APENDICE

CONJUNTO DE INSTRUCCIONES DEL COP420.

El conjunto de instrucciones básicas de los COPS, está diseñado para obtener las máximas ventajas de la arquitectura de canal dual que tiene. Este conjunto de instrucciones, junto con la arquitectura de éstos, provee al usuario con el poderío, versatilidad y eficiencia para lograr el máximo funcionamiento y capacidad en el mínimo de memoria.

Como los microcontroladores COPS no son dispositivos de mapeo de memoria, la mayoría de las instrucciones no tienen un campo de direcciones. Por lo tanto la mayoría de las instrucciones son de un "BYTE" de longitud, lo cual incrementa la eficiencia del programa. De esta forma, el espacio de ROM está dedicado a realizar las funciones, en lugar de ocuparse apuntando a campos de direcciones.

Es muy común que las instrucciones de los COP contengan una multiplicidad de funciones. Esto, obviamente aumenta la eficiencia del programa al realizar en una sola instrucción un número de funciones, que de otra manera requerirían varias instrucciones.

DESCRIPCION DETALLADA DE LAS INSTRUCCIONES

Para propósito de discusión y explicación, las instrucciones de los COPS se agruparon, a grandes rasgos, en las siguientes seis categorías:

1. Instrucciones aritmético/lógicas.
2. Instrucciones de transferencia de control.
3. Instrucciones de referencia a memoria.
4. Instrucciones de referencia a registros.
5. Instrucciones de prueba.
6. Instrucciones de entrada/salida.

Esta sección provee una descripción detallada de todas las instrucciones de los COPS. Esta descripción contiene la siguiente información:

- * El mnemónico (abreviatura) de la instrucción.
- * Una descripción escrita de la instrucción.
- * El flujo de datos o del programa asociado con la instrucción.
- * El código de operación en hexadecimal.
- * El tiempo de ejecución de la instrucción expresado en ciclos de instrucción.
- * Las condiciones de salto asociadas con la instrucción.
- * Cualquier restricción en la instrucción o en su uso así como cualquier efecto especial de la instrucción.

La siguiente lista define los símbolos usados en la descripción de las instrucciones:

- | | |
|----|--|
| A | Acumulador de cuatro "BITS" |
| B | Registro de dirección de RAM |
| Br | Parte alta del registro B, dirección de registro |

Bd Parte baja del registro B, dirección de dígito
 C Registro de acarreo de 1 "BIT"
 D Puerto de salida de datos de 4 "BITS"
 EN Registro de habilitación
 G Registro de 4 "BITS" para enlazar los datos para el puerto de entrada/salida G
 IL Dos "LATCHES" de 1 "BIT" asociados con las entradas IN3 e IN0.
 IN Puerto de entrada de 4 "BITS".
 L Puerto E/S de 8 "BITS" con tercer estado
 M El contenido de 4 "BITS" de la RAM direccionada por B.
 N Apuntador de pila de subrutina
 PC Registro de dirección de ROM, contador de programa
 Q Registro de 8 "BITS" para enlazar los datos del puerto L de E/S
 SA,SB,SC "STACK" A,B,C
 SIO Registro de corrimiento de 8 "BITS" y contador
 SK Salida de reloj controlada lógicamente
 T Registro contador binario de 8 "BITS"
 RAM(B) El contenido de 4 "BITS" de la RAM direccionada por B
 RAMN Contenido de la localidad de RAM direccionada por el apuntador de la pila N
 ROM(t) Contenido de la localidad de ROM direccionada por t
 PCa:b Bits del a al b del contador de programa (PC).

INSTRUCCIONES LOGICO-ARITMETICAS.

ASC

Suma binaria del registro C con el acumulador y con el contenido de la localidad de memoria direccionada por el registro B. El resultado se coloca en el acumulador, si el resultado es mayor de 15 (en decimal) se genera un salto.

A \leftarrow A + C + RAM(B) C: Se establece o reestablece de acuerdo al "CARRY" del "BIT" 3.

Codigo hexadecimal: 30

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Si 1 \rightarrow C, salta

Restricciones: Ninguna

ADD

Suma binaria del acumulador con la localidad de memoria especificada por el registro B. El resultado se coloca en el acumulador.

A \leftarrow A + RAM(B) C: No es afectada o no se usa

Codigo hexadecimal: 31

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

ADT

Suma binaria de 10 decimal al acumulador. Esta instrucción

se usa para ajuste decimal.

A <-- A + 10 (dec) C: No es afectada o no se usa

Codigo hexadecimal: 4A

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

AISC y

Suma binaria del valor inmediato "y" al acumulador y coloca el resultado en el acumulador. Se genera un salto si existe un acarreo del "BIT" 3.

A <-- A + y C: No es afectada o no se usa

Codigo hexadecimal: 5y

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Si hay acarreo del "BIT" 3 salta

Restricciones: y <> 0, 0 < y

CASC

Suma binaria del registro C, con el complemento a uno del acumulador y con el dato de la localidad de memoria especificada por el registro B. Genera un salto si el resultado es mayor a 15 (decimal). Esta es la instrucción básica de sustracción.

A <-- /A + RAM(B) + C

Codigo hexadecimal: 10

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Si 1 --> C, salta

Restricciones: Ninguna

CLRA

Limpia el acumulador

A \leftarrow 0 C: No es afectada

Código hexadecimal: 00

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

COMP

Reemplaza el valor en A con su complemento a uno

A \leftarrow /A C: No es afectada

Código hexadecimal: 40

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

NOP

No hay operación C: No es afectada

Código hexadecimal: 44

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

RC

Reestablece (Coloca en un cero lógico) el registro C de 1 "BIT".

C <-- 0 A: No es afectada

Código hexadecimal: 32

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

SC

Establece (Coloca en uno lógico) el registro C de 1 "BIT".

C <-- 1 A: No es afectada

Código hexadecimal: 22

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

XOR

Or exclusivo, "BIT" por "BIT", del acumulador con el contenido de la localidad de memoria especificada por el registro B. El resultado se coloca en el acumulador.

A <-- A XOR RAM(B) C: No es afectada

Código hexadecimal: 02

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

INSTRUCCIONES DE TRANSFERENCIA DE CONTROL.

JID

Salto indirecto. Esto involucra una modificación de dos

pasos del contador de programa. Primero, carga los ocho "BITS" menos significativos del contador de programa con el contenido del acumulador, (los cuatro "BITS" mas significativos) y la localidad de memoria especificada por el registro B.

El dato direccionado por este contador de programa modificado, es entonces cargado en los ocho "BITS" inferiores del contador de programa. La ejecución continúa en esta segunda dirección.

- (1) PC <-- PC + 1
- (2) PC7:0 <-- A, RAM(B)
- (3) PC <-- PC + 1
- (4) PC7:0 <-- ROM(PC10:8,A, RAM(B))

Codigo hexadecimal: FF

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: Si se utiliza esta instrucción para saltar a la última palabra de un bloque debe considerarse al siguiente vector de direcciones el cual se encuentra a partir de la siguiente instrucción.

JMP a

Salto directo. Carga el contador de programa (11 "BITS" inferiores) con la dirección especificada en la instrucción. Continúa la ejecución del programa en esta dirección.

PC10:0 <-- a C:No es afectada

A: No es afectada

Código hexadecimal: 6-
- (varía dependiendo del salto)

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: JMP en las dos últimas palabras de un capítulo salta al siguiente capítulo

JP a

Salto dentro de página.

(1) PC <-- PC + 1 C: No es afectada

(2) PC6:0 <-- a- páginas 2,3 únicamente A: No es afectada

o

(3) PC5:0 <-- a- todas las demás páginas

Código hexadecimal: - (varía dependiendo del salto)

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: No se puede saltar a la última palabra de una página. Una instrucción JP en la última palabra de una página salta a la siguiente página.

JSRP a

Salto a una subrutina dentro de la página 2.

(1) PC <-- PC + 1

(2) SC <-- SB <-- SA <-- PC

Codigo hexadecimal: - (varía dependiendo del salto)
Tiempo de ejecución: 1 Ciclo de instrucción
Condiciones de salto: Ninguna
Restricciones: No puede ser utilizada dentro de las páginas 2, 3. No puede saltar a la última palabra de la página 2.

JSR a

Salto directo a subrutina, carga los 11 "BITS" inferiores del contador de programa con la dirección a. Realiza un "PUSH" al "STACK" de subrutina. Continúa la ejecución en la dirección especificada por la instrucción.

- (1) PC \leftarrow PC + 2
- (2) SC \leftarrow SB \leftarrow SA \leftarrow PC
- (3) PC10:0 \leftarrow a

Codigo hexadecimal: 6-
- (varía dependiendo del salto)

Tiempo de ejecución: Dos ciclos de instrucción
Condiciones de salto: Ninguna
Restricciones: A10=0 en dispositivos de 1kb

RET

Regreso de subrutina y regreso del control al programa principal en la instrucción que sigue a JSR, JSRP o JSRL

PC \leftarrow SA \leftarrow SB \leftarrow SC

PC \leftarrow RAMN

Codigo hexadecimal: 48

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

RETSK

Regreso de una subrutina. Regreso del control al programa principal y siempre salta la instrucción que sigue a JSR, JSRP o JSRL.

PC <-- SA <-- SB <-- SC

Código hexadecimal: 49

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Siempre salta cuando regresa

Restricciones: Ninguna

INSTRUCCIONES DE REFERENCIA A MEMORIA.

CAMQ

Copia el contenido de ocho "BITS" del acumulador y de la localidad de memoria direccionada por el registro B al registro de ocho "BITS" Q. Esta es la función inversa a la instrucción CQMA, en función y con respecto a los cuatro "BITS" de Q con los cuales A y RAM(B) se comunican.

Q7:4 <-- A3:0 A: No es afectada

Q3:0 <-- RAM(B)3:0 C: No es afectada

Código hexadecimal: 33

3C

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

ESTA TESIS NO DEBE
SALIR DE LA BIBLIOTECA

RMB 0, RMB 1, RMB 2, RMB 3

Reestablece el "BIT" especificado en la instrucción en la localidad de memoria direccionada por el registro B.

RAM(B)n <-- 0 C: No es afectada

n=0,1,2,3 A: No es afectada

Código hexadecimal: RMB 0 4C

 RMB 1 45

 RMB 2 42

 RMB 3 43

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

SMB 0, SMB 1, SMB 2, SMB 3

Establece el "BIT" especificado en la instrucción en la localidad de memoria direccionada por el registro B.

RAM(B)n <-- 1 C: No es afectada

n=0,1,2,3 A: No es afectada

Código hexadecimal: SMB 0 4D

 SMB 1 47

 SMB 2 46

 SMB 3 4B

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

STII y

Almacena el valor inmediato de "y" en la localidad de

C:No es afectado

Código hexadecimal: BF

23

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna.

XDS r

Intercambia el contenido del acumulador con el contenido de la localidad de memoria direccionada por el registro B.

Reemplaza Br por el OR exclusivo de Br y r. Decrementa Bd en

1. Genera un salto si Bd se decrementa de 0 a 15.

A <--> RAM(B) C:No es afectada

Br <-- Br XOR n

Bd <-- Bd - 1

Código hexadecimal: r7

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Genera un salto si Bd-1 = 15

Restricciones: r=0,1,2,3 o 4 unicamente

XIS r

Intercambia el contenido del acumulador con el contenido de la localidad de memoria direccionada por el registro B.

Reemplaza Br con el OR exclusivo de Br y r. Incrementa Bd en uno. Genera un salto si Bd se incrementa de 15 a 0.

A <--> RAM(B) C:No es afectada

Br <-- Br XOR n, Bd <-- Bd+1.

Codigo hexadecimal: r4
r = 0,1,2,3
Tiempo de ejecución: 1 Ciclo de instrucción
Condiciones de salto: Genera un salto si $Bd + 1 = 0$
Restricciones: r = 0,1,2 o 3 unicamente

INSTRUCCIONES DE REFERENCIA A REGISTRO

CAB

Copia el contenido del acumulador a los cuatro "BITS" inferiores del registro B.

Bd <-- A A: No es afectada
 C: No es afectada
 Br: No es afectada

Codigo hexadecimal: 50
Tiempo de ejecución: 1 Ciclo de instrucción
Condiciones de salto: Ninguna
Restricciones: Ninguna

CBA

Copia el contenido de los cuatro "BITS" inferiores de B al acumulador.

A <-- Bd C: No es afectada
 B: No es afectada

Codigo hexadecimal: 4E
Tiempo de ejecución: 1 Ciclo de instrucción
Condiciones de salto: Ninguna
Restricciones: Ninguna

LBI r,d

Carga el registro B inmediatamente con los valores de r (para la porción alta del registro B). Salta todas las instrucciones LBI subsecuentes, hasta que se encuentra una instrucción distinta a LBI.

Br <-- r A:No es afectada

Bd <-- d C:No es afectada

Código hexadecimal: --

33

Tiempo de ejecución: 3 Ciclos de instrucción (3 "BYTES")

Condiciones de salto: Salta hasta que no haya un LBI

Restricciones: Ninguna.

LEI y

Carga el registro de habilitación con el valor inmediato de "y".

EN3:0 <-- y A:No es afectada

 C:No es afectada

Código hexadecimal: 33

6y

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

XABR

Intercambia el contenido del acumulador con el contenido de la parte alta del registro B (Br). Si Br es menor que 4 "BITS" de ancho, se colocan ceros en los "BITS" sobrantes

del acumulador.

Br <-- A, A3 <-- 0, A2 <-- 0

Codigo hexadecimal: 12
Tiempo de ejecución: 1 Ciclo de instrucción
Condiciones de salto: Ninguna
Restricciones: Ninguna

INSTRUCCIONES DE PRUEBA.

SKC

Si el registro de carry de un "BIT" (C) es igual a 1, salta a la siguiente instrucción del programa.

A: No es afectada

C: No es afectada

Codigo hexadecimal: 20
Tiempo de ejecución: 1 Ciclo de instrucción
Condiciones de salto: Salta si C=1
Restricciones: Ninguna

SKE

Si el contenido del acumulador es igual al contenido de la localidad de memoria direccionada por el registro B, salta a la siguiente instrucción del programa.

A: No es afectada

C: No es afectada

Codigo hexadecimal: 21
Tiempo de ejecución: 1 Ciclo de instrucción
Condiciones de salto: Salta si A=RAM(B)

Restricciones: Ninguna

SKGZ

Si las cuatro lineas G son cero lógico, salta a la siguiente instrucción del programa.

A: No es afectada

C: No es afectada

G: No es afectada

Codigo hexadecimal: 33

21

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Salta si G3:0 = 0

Restricciones: Ninguna

SKGBZ n, n=0,1,2,3

Si G(n) es cero, salta a la siguiente instrucción del programa.

A,C,G: No es afectada

Codigo hexadecimal: SKGBZ 0 33 01

SKGBZ 1 33 11

SKGBZ 2 33 03

SKGBZ 3 33 13

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Salta si el "BIT" especificado en G es cero.

Restricciones: Ninguna

SKMBZ n n=0,1,2,3

Si el "BIT" especificado en la localidad de memoria direccionada por el registro B es cero, salta la siguiente instrucción del programa.

A,C, RAM(B): No es afectada

Codigo hexadecimal:	SKMBZ 0	01
	SKMBZ 1	11
	SKMBZ 2	03
	SKMBZ 3	13

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Salta si $RAM(B)n = 0$

Restricciones: Ninguna

SKT

Si ha ocurrido un "CARRY" (desbordamiento) en el contador T desde la última prueba (último SKT), salta la siguiente instrucción del programa. Reestablece el "LATCH". (El acarreo o desbordamiento del timer establece el "LATCH" SKT. La instrucción SKT prueba y reestablece este "LATCH").

SKTL \leftarrow 0 A,C,T: No es afectada

Codigo hexadecimal:	41
---------------------	----

Tiempo de ejecución: 1 Ciclo de instrucción

Condiciones de salto: Salta si SKTL=1

Restricciones: Ninguna

INSTRUCCIONES DE ENTRADA/SALIDA

ING

Copia el estado del puerto G de E/S dentro del acumulador.

A <-- G C:No es afectada

G:No es afectada

Codigo hexadecimal: 33

2A

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

ININ

Copia el estado interno de las cuatro líneas IN dentro del acumulador.

A <-- IN C:No es afectada

Codigo hexadecimal: 33

28

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

INIL

Copia el estado de los "LATCHES" IL y de la entrada CKO al acumulador. Reestablece después los "LATCHES" IL.

A 3:0 <-- IL 3, CKO, "0", IL, 0

Codigo hexadecimal: 33

29

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: Si CKO no es seleccionado como una entrada general, se carga un 1

dentro de A2. Los "LATCHES"
permanecen encendidos hasta que
llega otro INIL.

INL

Copia el estado del puerto de ocho "BITS" L a la localidad
de memoria direccionada por el registro B y por el
acumulador.

Codigo hexadecimal: 33

2E

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

OBD

Copia el contenido de los cuatro "BITS" inferiores del
registro B (Bd) al puerto de salida D.

D <-- Bd A:No es afectada

B:No es afectada

C:No es afectada

Codigo hexadecimal: 33

3E

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

OGI y

Saca el valor inmediato del valor "y" por el puerto G de

cuatro "BITS"

G <-- y A:No es afectada

C:No es afectada

Codigo hexadecimal: 33

5y

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

OMG

Copia el contenido de la localidad de memoria direccionada por el registro B al puerto G de cuatro "BITS".

G <-- RAM(B) A:No es afectada

C:No es afectada

Codigo hexadecimal: 33

3A

Tiempo de ejecución: 2 Ciclos de instrucción

Condiciones de salto: Ninguna

Restricciones: Ninguna

XAS

Intercambia el contenido del acumulador con el contenido del registro SIO. Copia el contenido del registro C de un "BIT" con el latch SK. Esta es la instrucción básica de interfaz para la comunicacion serial .

A <--> SIO C:No es afectada

SKL <-- C

Codigo hexadecimal: 4F

Tiempo de ejecución	1 Ciclo de instrucción
Condición de salto:	Ninguna
Restricciones:	Ninguna

GLOSARIO

GLOSARIO.

Bit	Digito binario considerado como unidad, es la unidad de información mas pequeña que se puede tener.
Hardware	Parte física o componentes físicos que conforman un circuito electrónico.
Software	Programas elaborados que constituyen la parte inmaterial del circuito electrónico, sin embargo son las que regulan el funcionamiento del Hardware.
Stack	Area de memoria RAM o registros dedicados a almacenar la dirección de memoria a la que se debe regresar cuando se ha ejecutado una subrutina.
RAM	Tipo de memoria de lectura y escritura, es un tipo de memoria de acceso aleatorio, es del tipo volátil, es decir al desconectar la alimentación se pierde la información.
ROM	Tipo de memoria de solamente lectura, es del tipo no volátil.

PROM	Unidad de memoria ROM programable por el usuario.
EPROM	Unidad de memoria ROM programable y borrrable, generalmente con luz ultravioleta por el usuario.
EEPROM	Unidad de memoria ROM programable y borrrable eléctricamente por el usuario.
Flip-flop	Unidad de memoria mas pequeña que se puede tener, se puede decir que es la unidad minima de memoria.
BCD	Código digital para conversión binaria-digital.
Carry	Dígito producido al efectuar una suma y el resultado es mayor que la base del sistema de notación en el cual se expresan los sumandos.
COP	Siglas en inglés de la palabra Procesador Orientado al Control.
CPU	Siglas en inglés de la palabra Unidad Central de Proceso o microprocesador.

Byte	Agrupamiento de ocho bits que forman una palabra.
Driver	Dispositivo electrónico que aumenta la corriente de las señales, para que pasen a otra etapa del circuito.
Latch	Dispositivo que retiene algún estado lógico eléctrico.
E/S	Abreviatura de Entrada/Salida.
Microbus	Estandar de comunicación en paralelo en ocho bits con las líneas mínimas necesarias para establecer la comunicación.
Single-Step	Ejecución de las instrucciones que conforman un programa de una por una.
Overflow	Condición aritmética de acarreo, cuando se excede la base del sistema.
PC	Abreviatura en inglés de la palabra Contador de Programa.
Refresh	Ciclo de efectuado para mantener el

estado de los capacitores que conforman las memorias RAM dinámicas.

MOS

Tecnología de fabricación de circuitos lógicos metal-óxido semiconductor.

TTL

Tecnología de fabricación de circuitos lógicos transistor-transistor lógico,

BIBLIOGRAFIA

BIBLIOGRAFIA.

1.- "Microprocessors and Digital Systems"

Hall Douglas V.

Ed: Mc Graw Hill

ed: segunda, cuarta reeimpresión, 1987

pags: 194-199.

2.- "Systems Design with Advanced Microprocessors"

Freer John.

Ed: Howard W. Sams & Company

ed: segunda, 1987

pags: 40-60

3.- Manual técnico de microcontroladores.

"Single-Chip Microcomputer Data"

Ed: Motorola Inc.

ed: segunda, 1984.

pags: 1-3 a 1-12.

4.- Manual técnico de microcontroladores.

" TMS 7000 Family, Data Manual"

Ed: Texas Instruments Inc.

ed: primera, 1989.

pags: 1-1 a 1-7.

5.- Manual técnico de microcontroladores.

" Embedded Controller Handbook"

Ed: Intel Corporation, 1988.

pags: 1-1 a 1-17.

6.- Manual técnico de microcontroladores.

" Microcontroller Databook"

Ed: National Semiconductor Inc., 1987

pags: 4-149 a 4-170.

7.- Manual técnico de microcontroladores.

"Microcontroller Databook"

Ed: National Semiconductor Inc. 1989.

pags: 1-260 a 1-277.

8.- Manual técnico.

" Interface Databook"

Ed: National Semiconductor Inc. 1989.

pags: 3-23 a 3-25.

9.- Manual técnico.

" Memory Databook"

Ed: National Semiconductor Inc. 1989

pags: 3-4 a 3-10.

10.- Manual técnico

" General Purpose Linear Devices"

Ed: National Semiconductor Inc. 1989

pags: 1-318 a 1-324.