



UNIVERSIDAD NACIONAL AUTÓNOMA
DE MÉXICO

FACULTAD DE INGENIERÍA

TESIS

“AUTOMATIZACIÓN DE UNA MÁQUINA DE
SOPLADO DE PLÁSTICO”

QUE PARA OBTENER EL TÍTULO DE
INGENIERO EN ELÉCTRICA - ELECTRÓNICA

PRESENTA:

GUSTAVO GUILLÉN VALLE



DIRECTOR DE TESIS:
M.I. ROLANDO ALBERTO CARRERA MÉNDEZ

CIUDAD UNIVERSITARIA

MÉXICO D.F.

2009



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



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 esta 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.

AGRADECIMIENTOS

Este trabajo no habría sido posible sin la ayuda de mi familia que ha sido el principal pilar de toda mi carrera, gracias madre por enseñarme a mirar siempre adelante, padre por brindarme siempre todo tu apoyo, hermano porque sé que siempre cuento contigo. También quiero agradecer a mis dos grandes amigas quienes me apoyan y están ahí cuando las necesito, gracias Yadira y Michelle.

Índice general

1. Introducción	3
2. Estado actual de la máquina de soplado de plástico	5
3. Automatización de la máquina de soplado de plástico	9
3.1. Circuito neumático	9
3.2. Electrónica del control de la máquina de soplado	15
3.3. Programación del control de la máquina de soplado de plástico	24
4. Control de temperatura	27
4.1. Operación del sistema de calentamiento	27
4.2. Termopar y pirómetros	27
4.3. Electrónica del control de temperatura	31
4.4. Programación del control de temperatura	37
5. Interfaz de comunicación e implementación final	41
5.1. Electrónica de la interfaz de comunicación	41
5.2. Programación de la Interfaz de comunicación	44
5.3. Implementación de los módulos	46
5.4. Presentación final	50
6. Operación del control	52
6.1. Conexión del control	52
6.2. Operación de la Interfaz	54
6.3. Manipulación manual	58
7. Resultados y Conclusiones	60
A. Mantenimiento	62
B. Diagramas de Flujo	64
C. Diagramas de circuito	68

D. Tablas y gráficas

71

Bibliografía

74

Capítulo 1

Introducción

El objetivo de este proyecto es manipular una máquina de soplado de plástico de forma automática, así como controlar la temperatura a la que opera ésta. Las razones por las que se busca automatizar la máquina son: incrementar la cantidad de piezas de plástico elaboradas durante una jornada normal de trabajo, simplificar la operación de forma que el operador no requiera de grandes conocimientos para realizar el proceso productivo y obtener una producción constante para producir las cantidades necesarias en el tiempo estipulado.

El operador de la máquina ejecuta una secuencia de pasos de control mediante la manipulación de válvulas manuales encargadas de activar actuadores neumáticos, los cuales realizan el desplazamiento del carro, de la prensa y el soplador. En cuanto a la temperatura de la máquina, ésta es controlada a través de pirómetros analógicos. Por lo anterior se tienen dos problemas a solucionar: el primero es buscar una solución al problema de ejecutar la secuencia de pasos de forma automática y el segundo es controlar la temperatura a la que se desea que opere. Anteriormente la ejecución de la secuencia se resolvía con un control basado en un banco de relevadores, los cuales eran activados manualmente para que realizaran la secuencia deseada. Estos controles son confiables y eficientes, sin embargo tienen grandes inconvenientes: es muy tardado conectarlos. Por lo que si el proceso necesitaba ser modificado, en algunos casos era más rentable comprar un banco nuevo de relevadores que detener las máquinas para realizar las conexiones pertinentes; además la mayoría de los bancos son colocados en gabinetes grandes por lo que requieren un espacio considerable para su instalación.

En la actualidad los controladores lógicos programables (PLC, por sus siglas en inglés) se han convertido en una buena opción para automatizar cualquier proceso debido a su bajo costo, tamaño práctico y bajo consumo de energía. Los PLC son dispositivos electrónicos diseñados para trabajar con tensiones industriales y en ambientes ruidosos. Además cuentan con ciertas funciones para ampliar sus aplicaciones, como entradas analógicas y salidas analógicas, permitiendo interactuar no sólo con interruptores sino también con sensores.

En este trabajo se presenta otra alternativa empleando microcontroladores AVR para automatizar una máquina de soplado de plástico. Los microcontroladores AVR son circuitos integrados programables muy versátiles que permiten realizar operaciones complejas. Los microcontroladores AVR cuentan con diversos módulos que permiten interactuar con diversos periféricos y sensores. Para simplificar el tablero de la máquina de soplado y aprovechar las capacidades que los AVR ofrecen, se diseñó un módulo para controlar la temperatura a la cual trabaja la

máquina de soplado de plástico.

El presente trabajo se integra de la siguiente manera:

El capítulo dos plantea las condiciones en las que se encontraba la máquina de soplado de plástico, su anterior funcionamiento y control de temperatura.

El capítulo tres da solución al problema de controlar la operación de la máquina, mediante el acoplamiento del circuito neumático con la electrónica diseñada para controlar las electroválvulas, así como la programación del microcontrolador.

El capítulo cuatro abarca con mayor profundidad los elementos que requieren calentamiento, da una descripción del control de temperatura que anteriormente utilizaba y plantea la nueva solución al problema de controlar el calentamiento de la máquina. Explica como realizar la compensación de los termopares y el diseño del control de dos posiciones dentro del microcontrolador.

En el capítulo cinco se desarrolla la interfaz de comunicación interna, a través de la cual el operador podrá controlar las funciones tanto del control de la máquina como de la temperatura. Se muestra el diagrama de bloques y programación para lograr la interacción entre los tres módulos y finalmente su presentación final.

El capítulo seis trata sobre la forma que debe conectarse el control para su operación, así como los menús que contiene la interfaz para su operación y finalmente como manipular la máquina en modo manual.

El capítulo siete da los resultados obtenidos del presente trabajo, así como las conclusiones a las que se llegó.

Por último para finalizar este trabajo se agregan cuatro apéndices sobre mantenimiento, diagramas de flujo, circuitos impresos y algunas tablas de referencia usadas en este trabajo.

Capítulo 2

Estado actual de la máquina de soplado de plástico

La máquina de soplado de plástico (figura 2.1) cuenta con varios sistemas que se pueden automatizar como son: la manipulación del proceso de producción, el sistema de calentamiento, el enfriamiento del molde, la presión de la compresora, entre otros. En este texto sólo se presenta la automatización de proceso de producción de las piezas de plástico y el sistema de calentamiento.

La máquina de soplado de plástico operaba de forma manual, es decir, era manipulada a través de cuatro válvulas manuales, de las cuales tres accionaban actuadores neumáticos acoplados a la máquina para realizar el desplazamiento del carro, de la prensa y del soplador; la cuarta válvula permitía la apertura o cierre del soplador, el cual introduce aire a presión a la pieza para que tome la forma que se encuentra en el molde.

Existen varios tipos de actuadores neumáticos, sin embargo, los que se utilizan en esta máquina son actuadores de doble efecto con amortiguador (figura 2.2), los cuales pueden realizar trabajo en ambos sentidos. Para desplazar el vástago del actuador al frente se deja pasar aire a presión por la conexión de la cámara trasera y se desfoga por la conexión de la cámara delantera; pero si se quiere que el vástago retroceda, se realiza el proceso inverso, se deja pasar aire a presión por la conexión de la cámara delantera y desfoga por conexión de la parte trasera.

Para poder realizar esta conmutación de aire se emplean las válvulas, de las cuales igualmente hay una amplia gama dependiendo del número de vías y posiciones que tengan. En esta máquina se utilizaban válvulas 5/2, en donde el 5 corresponde al número de vías (ductos o conexión externas a las válvulas) de la válvula y el 2 al número de posiciones que puede tener. Cada posición es representada por un cuadro. Este tipo de válvulas es ideal para manejar actuadores de doble efecto porque permiten conmutar el aire para desplazar el actuador en ambos sentidos (figura 2.3)

Para comenzar la elaboración de las piezas de plástico, primero debe colocarse la máquina en un estado inicial: la prensa debe estar abierta, el carro desplazado al fondo, el actuador del soplador desplazado hacia arriba y el aire apagado; una vez en su estado inicial se procede a ejecutar la siguiente secuencia:

1. El operador espera a que el parison (tubo de material que sale por el cabezal de la máquina, figura 2.4) llegue al fondo del molde.
2. Cierra la prensa.
3. Corta con un cuchillo el parison.

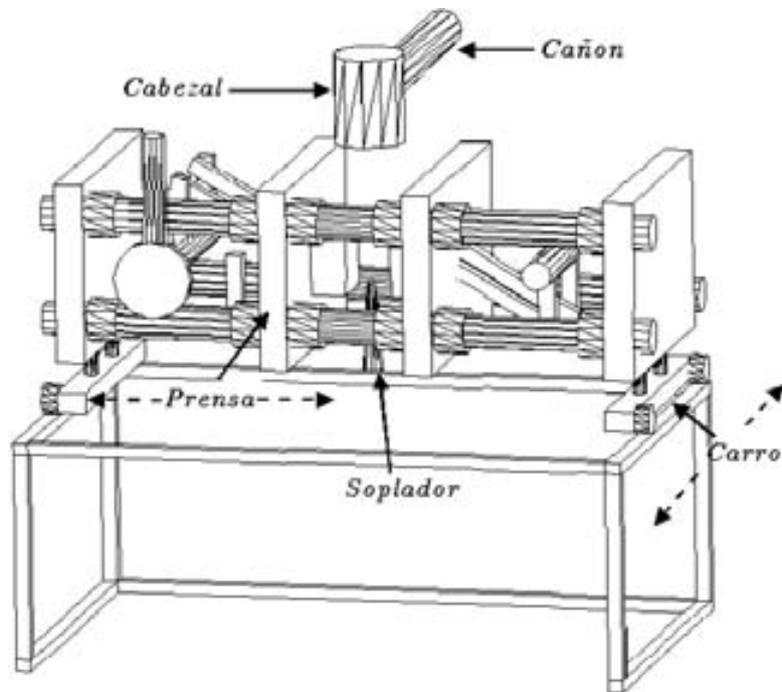


Figura 2.1: Máquina de Soplado de Plástico

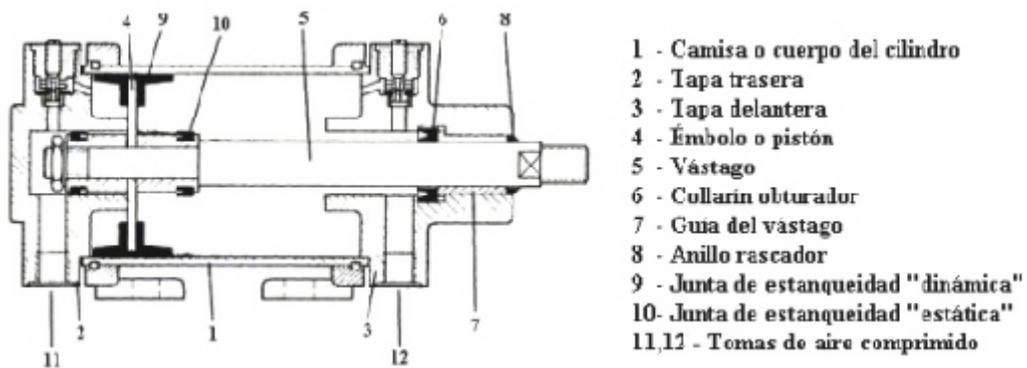


Figura 2.2: Actuador de doble efecto con amortiguamiento



Figura 2.3: *Válvula 5/2*

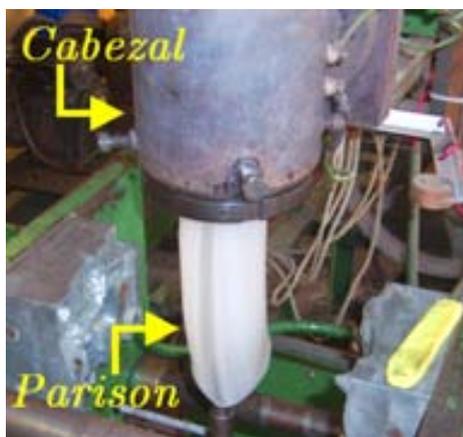


Figura 2.4: *Párison*

4. Desplaza el carro al frente.
5. Baja el actuador que sostiene al soplador.
6. Inyecta aire a la pieza durante un determinado tiempo.
7. Sube el actuador que sostiene al soplador.
8. Espera a que el aire inyectado a la pieza desfogue.
9. Abre la prensa.
10. El operador recoge la pieza de plástico del interior de la prensa.
11. Regresa nuevamente el carro al fondo e inicia nuevamente el proceso.

La secuencia mencionada es ejecutada cada vez que se elabora una pieza de plástico y aunque la secuencia parece muy simple, se debe tener precaución en los pasos 5, 6, 7 y 8. El paso 6 trata sobre la inyección de aire durante un determinado tiempo, el cual es variable, dependiendo de la figura que se esté realizando, pues entre más grande es la figura que se quiere realizar se debe

inyectar aire durante más tiempo; igualmente, el tiempo de espera del paso 8 depende de la pieza que se este fabricando, aunque en este paso la diferencia de tiempo es casi nula, puede variar pero en mucho menor proporción que en el de inyección de aire. Por último los pasos 5 y 7 puede que no sean necesarios en algunas ocasiones, pues depende del molde que se encuentre colocado, esto se debe a que la máquina que se automatizó maneja varios tipos de moldes y cada molde puede soplarse desde diferentes ángulos. Por ejemplo, algunos se soplan por arriba, otros de lado y otros por abajo. Cuando se soplan de lado y por arriba el proceso es casi el mismo, es decir se efectúan los 11 pasos pero cuando se sopla el molde por abajo, se elimina una válvula y un actuador pues únicamente se espera a que el parison llegue al soplador y no se requiere de un actuador que lleve al soplador hasta el molde, por lo tanto el proceso sufre una modificación que de cierta forma no es tan distinta, pues únicamente se eliminan dos paso de los once antes mencionados.

Finalmente el sistema de calentamiento de la máquina es por medio de cuatro resistencias de 1400W a 220V, conectadas en estrella, las cuales envuelven el cañón y cabezal de la máquina. El cañón de la máquina es un tubo largo a través del cual fluye el plástico para ser calentado. El cabezal es la parte final del cañón, por el cual sale el parison (esto se describe con más detalle en el capítulo 4). La temperatura originada por el calentamiento de las resistencias era controlada a partir de tres pirómetros analógicos en los que se medía la temperatura con tres termopares, dos colocados en diferentes partes del cañón y uno sobre el cabezal.

Capítulo 3

Automatización de la máquina de soplado de plástico

3.1. Circuito neumático

La automatización de la máquina se solucionó implementando un circuito neumático con los actuadores instalados (actuadores neumáticos de doble efecto con amortiguamiento) y agregando un actuador de doble efecto para efectuar el corte del párison que anteriormente realizaba el operador. Antes de poder explicar el acoplamiento y secuencia de los actuadores, es necesario dar algunas convenciones en relación con los circuitos neumáticos.

- Los actuadores neumáticos se nombran con letras mayúsculas (A, B, C).
- Cuando el vástago del actuador se desplaza al frente su posición se indica con A+
- Cuando el vástago del actuador regresa al fondo su posición se indica con A-

Ahora que se han establecido las convenciones, puede nombrarse a los actuadores y describirlos.

Actuador	Nombre
Prensa	A
Carro	B
Corte	C
Soplador	D

El actuador A se encarga del cierre y apertura de la prensa, está acoplado en la parte de enfrente, sobre la palanca (figura 3.1), de modo que al adquirir el actuador la posición A+, permita el cierre y al adquirir A- permita la apertura (figura 3.2).

El actuador B se encarga de desplazar el carro al frente y atrás. El actuador se acopló a la mitad de la parte baja del carro; así, al adquirir la posición B+ el carro se desplaza al frente (figura 3.3), y al adquirir la posición B- el carro vuelve atrás (figura 3.4).

El actuador C es un nuevo elemento agregado a la máquina, el cual se encarga de efectuar el corte del párison que anteriormente realizaba el operador con un cuchillo. Se colocó este nuevo actuador debajo del cañón de la máquina, cerca del cabezal con la finalidad de efectuar el corte

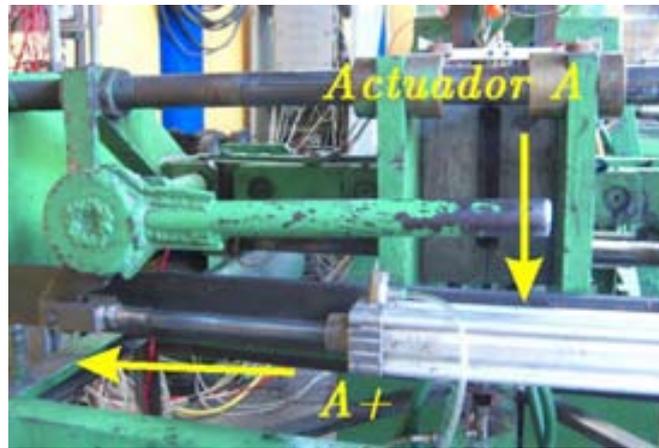


Figura 3.1: *A+ cierra prensa*

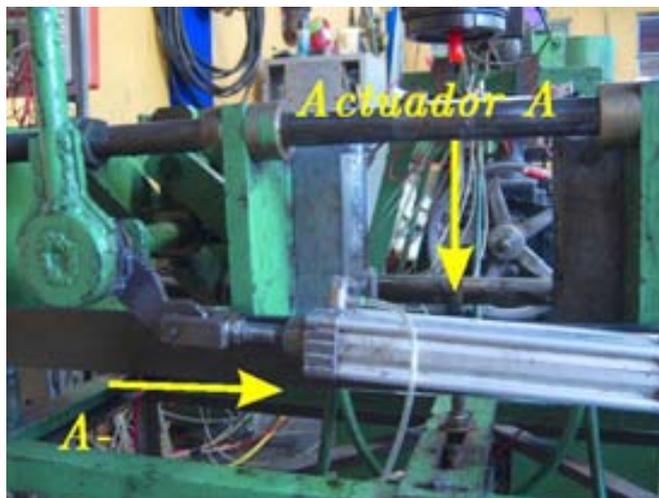


Figura 3.2: *A- abre prensa*

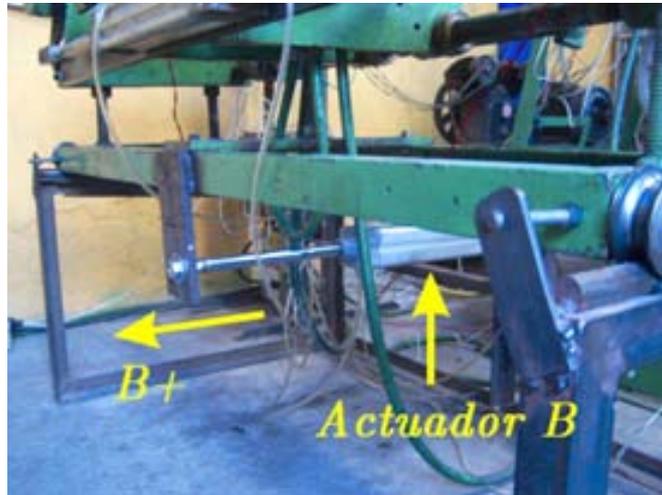


Figura 3.3: *B+* avanza el carro

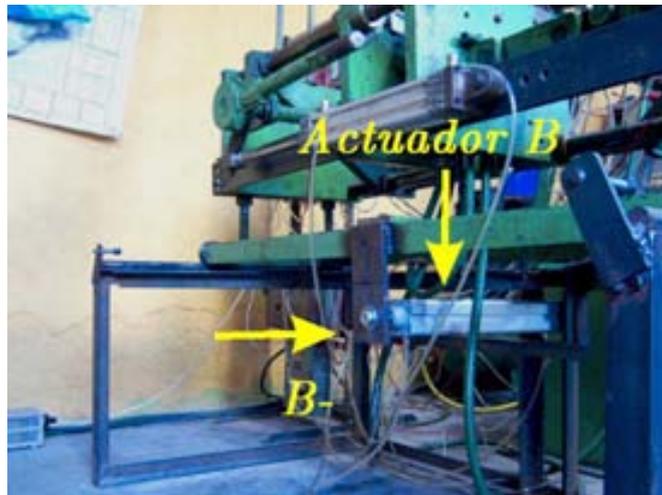


Figura 3.4: *B-* regresa el carro

con una carrera (longitud del vástago) corta. El actuador C cuenta con dos guías y una placa de metal al final del vástago, ideal para acoplar y soportar el peso de la cuchilla de corte.

El actuador C actúa diferente a los demás pues su posición cambia cada que se elabora una pieza. Para aclarar la idea anterior se ilustra su funcionamiento mediante un ejemplo: al empezar, la cuchilla se encuentra en la posición C- (estado inicial), una vez que el parison ha llegado al final del molde se cierra la prensa. Para realizar el corte el actuador C toma la posición C+ y permanece en esa posición mientras el proceso continúa. Una vez concluidos los demás pasos, la máquina retorna nuevamente al estado inicial a excepción del actuador C que está en la posición C+, esperando a que nuevamente el parison alcance el fondo del molde para iniciar el proceso. Cuando esto sucede, igualmente se cierra la prensa, pero en esta ocasión para realizar el corte, el actuador toma la posición C- y permanece en esa posición hasta que termina el proceso. Como se observa, este actuador se va alternado de C+ a C- cada vez que se efectúa un corte de parison.

El corte de parison puede realizarse de dos formas: con una cuchilla fría o con una caliente. Ambas presentan ventajas y desventajas. La primera efectúa el corte sin ser calentada, presentando la ventaja de que no se requiere gastar energía para su funcionamiento y la desventaja de que requiere estar muy afilada, por lo cual se tiene que estar afilando constantemente, además el sistema requiere mayor presión a la hora de realizar el corte; en cambio, la segunda, como su nombre lo menciona, requiere ser calentada, esta es la desventaja de este tipo de corte, pues se necesita gastar energía para su calentamiento, la ventaja es que no requiere tanto mantenimiento, es decir, no hay que afilarla tan seguido y se puede trabajar con una presión menor, pues al calentarse puede efectuarse un corte sin ejercer mucha fuerza.

En esta máquina se optó por realizar el corte con cuchilla caliente, por las ventajas antes mencionadas. El problema para construir la cuchilla fue buscar como calentarla. Esto se solucionó con una resistencia plana de 100W, que comúnmente usan las selladoras de bolsa. Esta resistencia se colocó debajo de la cuchilla hecha con un tramo de sierra cinta con filo piramidal de dimensiones 19x2.5cm. La cuchilla se colocó sobre una base trapezoidal construida con ángulo de aluminio para permitir el paso del parison cuando el actuador se encuentra en la posición C+. Finalmente para evitar que el calor pasara a la base y de la base al actuador se tuvo que aislar colocando dos tramos de celorón en cada lado de la base. El celorón se escogió por ser un aislante térmico y eléctrico que soporta temperaturas hasta de 135°, lo que es más que suficiente para realizar el corte del parison. La colocación final y operación se muestra en la figura 3.5

Finalmente el actuador D se encarga de transportar el soplador que inyecta aire a la pieza de plástico. Este actuador puede no utilizarse dependiendo de la forma en que se sople el molde. En los moldes que son soplados por arriba, el actuador D se encuentra sobre una base colocada encima del carro (figura 3.6), de esta forma cuando el actuador B está en B+ el actuador D toma la posición D+ para llevar el soplador hasta el molde e inyectar aire a la pieza, cuando termina, el actuador vuelve a la posición D-. En cambio, cuando el molde es soplado de lado, el actuador se encuentra sujetado a un costado del molde (figura 3.7) e igualmente cuando el actuador B está en la posición B+ el actuador D adquiere la posición D+ para llevar el soplador al molde. Por lo tanto, ese actuador mantiene su función que es desplazar el soplador hasta el molde, lo único que varía es su posición en función de la pieza que se esté produciendo.

Por último, cuando la pieza se sopla por abajo no es necesario el actuador D, debido a que no hay necesidad de transportar el soplador hasta el molde, únicamente el soplador se coloca debajo del molde y se espera a que el parison llegue (figura 3.8).



Figura 3.5: *Movimiento del corte del párison*

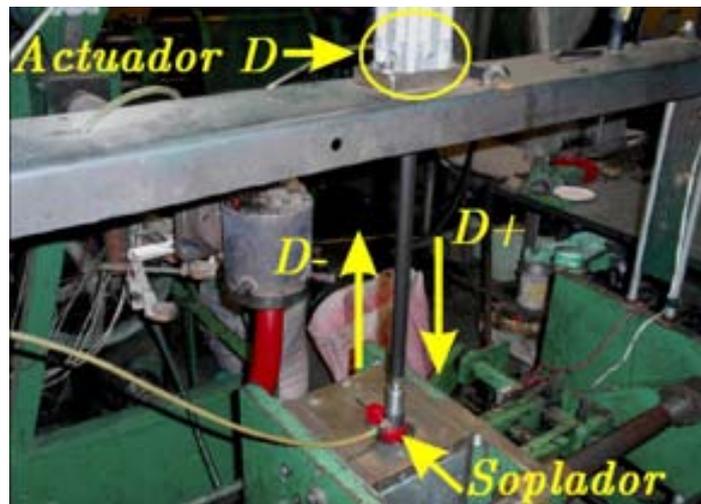


Figura 3.6: *Pieza soplada por arriba*



Figura 3.7: *Pieza soplada de lado*



Figura 3.8: *Pieza soplada por abajo*

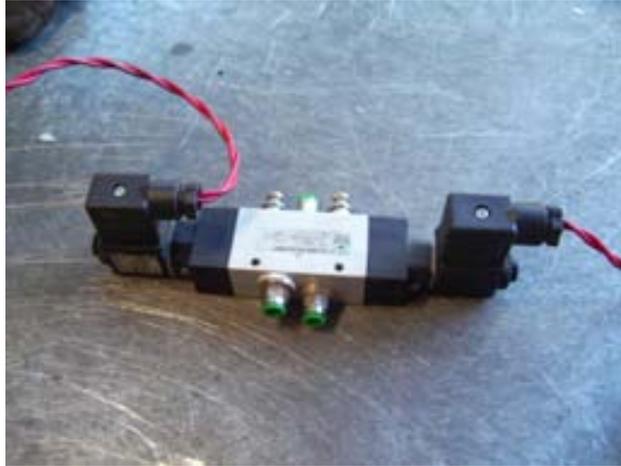


Figura 3.9: *Electroválvula 5/2*

Antes, para manipular el desplazamiento de estos actuadores de doble efecto se utilizaban válvulas manuales, sin embargo, para conseguir una manipulación automática de estos actuadores se optó por sustituir las válvulas por electroválvulas. Las electroválvulas realizan la misma función que una válvula manual sólo que su pilotaje (el pilotaje es el medio físico, eléctrico o neumático que permite la conmutación de una válvula de una posición a otra) es eléctrico. Las electroválvulas están formadas por una bobina y una válvula, al aplicar un voltaje a la bobina ésta convierte la energía eléctrica en mecánica para conmutar la válvula. Hay electroválvulas para CD y CA, a diferentes voltajes, que pueden ser biestables o monoestables. Cuando es biestable, está compuesta por dos bobinas de voltaje. Al dar un pulso a una de ellas, la válvula conmuta y permanece en esa posición, si se quiere regresar a su posición original se da un pulso a la otra bobina; en cambio, si es monoestable cuenta solo con una bobina, que al energizar la válvula conmuta y permanecerá en ese estado mientras se mantenga el pulso, cuando éste se retira, la electroválvula regresa a su posición original por acción de un resorte que se encuentra en su interior. Las electroválvulas utilizadas para el circuito neumático de la máquina fueron electroválvulas 5/2 biestables (figura 3.9) a 24 [V] de corriente directa.

Una vez que se conocen los elementos neumáticos utilizados, se implementó el circuito neumático mostrado en la figura 3.10, para manipular la secuencia de pasos que debe realizar.

3.2. Electrónica del control de la máquina de soplado

Establecido el circuito neumático en la sección anterior, el problema de automatizar se convierte en establecer la secuencia de pulsos que será dada a las bobinas de las electroválvulas. Como se mencionó son biestables, por lo tanto para controlar los cuatro actuadores tendremos ocho bobinas y dos extras para el soplador, encargadas de manipular la apertura y cierre del aire, entonces en total hay diez bobinas a controlar. Las electroválvulas se escogieron biestables para evitar el exceso de corriente pues su conmutación se consigue aplicando un pulso de 24 V durante 250 ms, con lo que una fuente de 24 V a 1 A es suficiente para alimentar a las diez

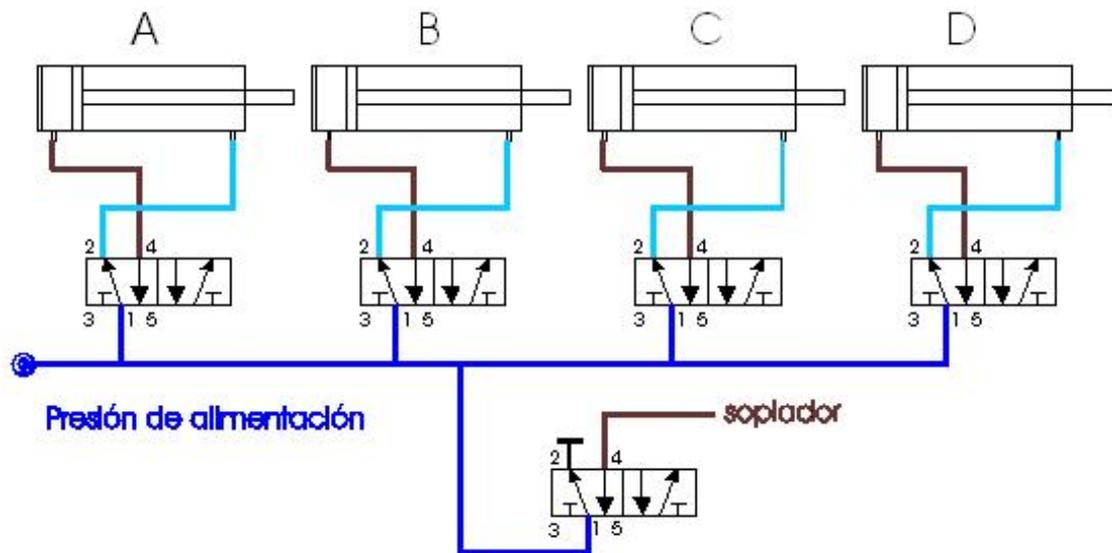


Figura 3.10: Circuito Neumático

bobinas.

El primer paso para el diseño fue buscar la forma de mandar la secuencia de pulsos a las bobinas. Actualmente en la industria utilizan el PLC para solucionar este tipo de problemas, sin embargo en este trabajo se propone otra alternativa que es utilizar microcontroladores AVR de la familia ATmega, con la finalidad de diseñar un control que manipule la máquina. El diseño del control de la máquina de soplado de plástico fue realizado con el microcontrolador ATmega32, el cual se escogió por las siguientes razones:

- Interfaz TWI (*Two-Wire Interface*) para comunicarse con una memoria serial EEPROM
- Cuatro puertos bidireccionales de 8 bits cada uno
- 32 Kb en memoria flash y 1 Kb en memoria EEPROM interna
- Software de desarrollo libre (Compilador y Editor)

Una vez seleccionado el microcontrolador ATmega32 como dispositivo de control, debe implementarse la electrónica correspondiente para interactuar con los sensores y bobinas de las electroválvulas, pues el microcontrolador por si solo no sirve para controlar la máquina, por lo tanto es necesario realizar una serie de acoplamientos a las entradas y salidas de nuestro control. Los acoplamientos tienen que hacerse previniendo siempre el ruido eléctrico, sin subestimar las condiciones adversas de un ambiente industrial, pues si se desea diseñar un sistema confiable y seguro tiene que protegerse completamente, debido a que el microcontrolador es un dispositivo electrónico extremadamente sensible a estos ruidos.

En la industria es casi imposible determinar los tipos y fuentes de ruidos que pueden afectar nuestro sistemas, por lo tanto, la forma de combatirlos es protegiendo al microcontrolador de las

fuentes más comunes de ruidos, lo cual se logra siguiendo una serie de reglas sencillas que son de gran ayuda para llevar a cabo un buen diseño e ignorarlas podría originar serios problemas sobre la aplicación. Estas reglas se mencionarán conforme se vaya avanzando en el desarrollo del control.

La protección del microcontrolador comienza desde su alimentación, pues a través de ella se pueden introducir los llamados ruidos conducidos, que son una de las principales causas del mal funcionamiento de nuestro sistema. Este tipo de ruido se origina por las cargas inductivas que hay en la industria como son los motores, relevadores y contactores, que al conmutar o arrancar provocan severas variaciones en la línea, originando picos de tensión por un par de milisegundos con niveles muy superiores a la tensión normal. Los efectos mas comunes que provoca este tipo de ruido al sistema son:

- Bloquea al microcontrolador
- Reinicios aleatorios del programa
- Saltos no previstos en el contador de programa
- Genera interrupciones falsas
- Disminuye la vida útil del microcontrolador

La solución a este problema pareciera ser muy simple, ya que se podría sugerir alimentar al sistema con una fuente que contenga un transformador de potencia para aislar al sistema de los posibles ruidos en la línea, puesto que su respuesta en frecuencia a lo mucho alcanza 400 Hz, sin embargo esto no es del todo cierto, a frecuencias altas como las contenidas en los picos de tensión, el transformador se comporta como un capacitor siendo sus placas los bobinados primario y secundario, por lo que se convierte en un camino muy fácil para las perturbaciones, en consecuencia la solución no es tan obvia como se piensa.

Una de las reglas que hay que tomar en cuenta es proteger el transformador de la fuente que alimentará al microcontrolador, tanto en el primario como en el secundario para evitar problemas de ruido. La forma de proteger el lado del primario (127 V) es colocando un filtro de red para evitar que las sobre tensiones alcancen nuestro transformador; sin embargo, en nuestro caso evitamos el diseño del filtro de red utilizando como protección un regulador de voltaje con supresor de picos, para alimentar el lado primario de nuestra fuente. En cuanto al lado secundario se protegió implementando un filtro LC a la salida del rectificador de onda. Las dos reactancias forman un divisor de voltaje el cual reduce significativamente los posibles picos que se pudieran introducir al lado del secundario. De esta forma conseguimos una fuente (figura 3.11) libre de ruidos de línea.

El ATmega32 cuenta con puertos bidireccionales, por lo que programando bits de un puerto como entrada podemos interactuar con nuestros sensores; no obstante, hay que tener mucho cuidado en la forma en que se va a introducir la señal, pues las entradas son otro medio perfecto para meter ruido al microcontrolador. Una regla que hay que seguir para proteger al microcontrolador de posibles ruidos a través de los pines de entrada es el uso de opto acopladores. Un opto acoplador es un circuito compuesto por un diodo LED y un fototransistor, ambos se encuentran físicamente separados dentro del encapsulado, su funcionamiento es simple, sólo se hace pasar

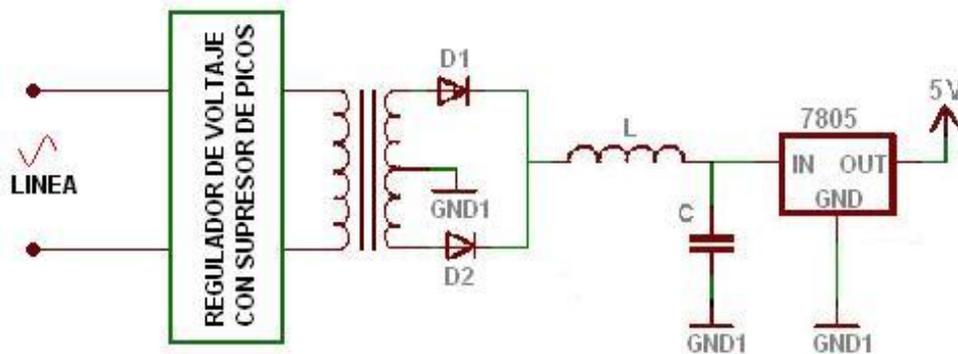


Figura 3.11: Fuente de alimentación

una corriente por el LED el cual emite una luz que al ser captada por el fototransistor permite la conducción. Los propósitos de colocarlo entre el interruptor y el pin de entrada son:

- Aislar la alimentación del circuito de control de la alimentación del circuito de potencia (entradas y salidas del sistema) para evitar los posibles ruidos.
- Acoplar los voltajes TTL del microcontrolador con los 24V de alimentación del interruptor.
- Máxima seguridad, un fallo en cualquiera de los elementos no implica que la corriente de uno circule por el otro.
- Proteger al microcontrolador de los golpes de tensión.

Para poder ofrecer mayor inmunidad ante el ruido, el PCB (Printed Circuit Board) se dividió en dos partes: el primer PCB contiene toda la electrónica de control que requiere la máquina y el segundo PCB contiene las entradas y salidas del sistema, ambos circuitos se comunican a través de un cable plano para mandar y recibir las señales entre ellos.

Las entradas se diseñaron con opto acopladores 4N25 colocados sobre el PCB de entradas y salidas. La alimentación de estos es a 24 V de corriente directa para aprovechar la fuente de alimentación de las electroválvulas y evitar el diseño de una nueva. Así, sólo es necesario calcular las resistencias de *pull up* para el optoacoplador. No obstante, el ATmega32 cuenta internamente con resistores *pull up* en todos sus puertos, sólo hay que activarlas mediante la programación, para así poder conectar el colector del fototransistor directamente al pin de entrada. Con esto se reduce el número de elementos sobre nuestro circuito. La configuración de las entradas se construyó considerando un común, es decir un pin con 24 V, de esta manera solo hay que poner en corto al común con el pin de entrada que se desee introducir al microcontrolador (figura 3.12).

Los sensores del control son interruptores colocados en diferentes partes de la máquina con la finalidad de poder conocer la posición de los actuadores. Como cada actuador puede adquirir dos posiciones, se necesita de ocho entradas para poder detectar todas las posiciones posibles; sin embargo, nuestro sistema sólo considera cinco entradas, una de las cuales es para iniciar la

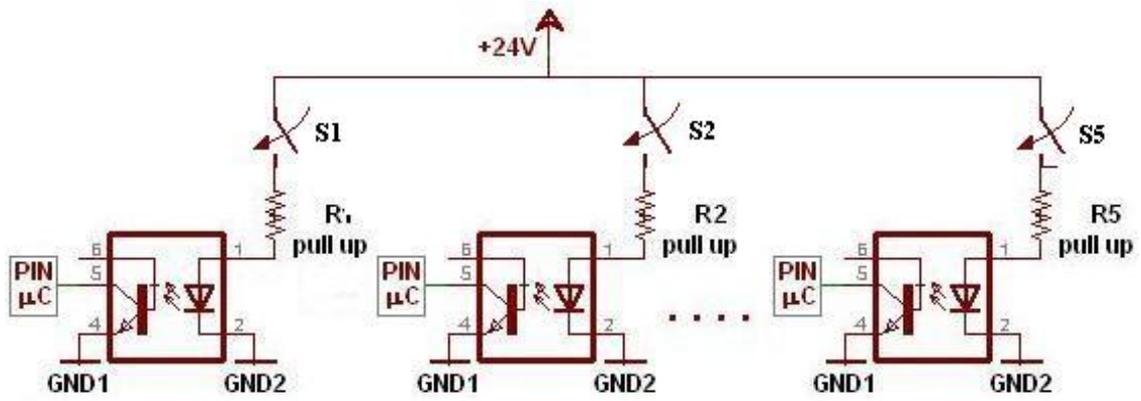


Figura 3.12: Configuración de las entradas



Figura 3.13: Botón de inicio

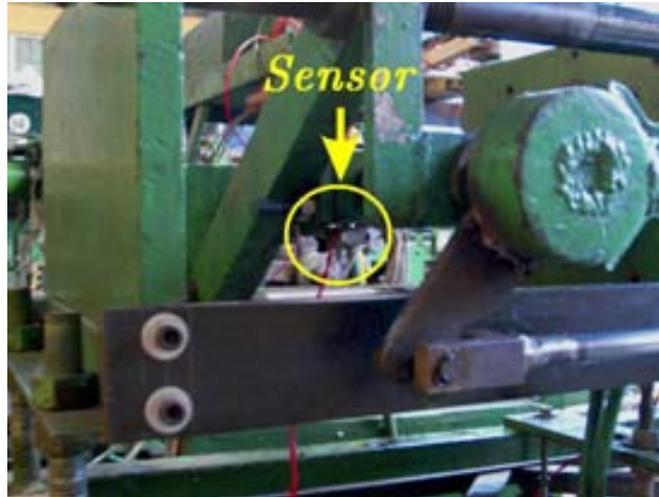


Figura 3.14: *Sensor A+*

secuencia por medio de un botón normalmente abierto (figura 3.13). Cada vez que se presiona este botón se inicia la secuencia y una vez terminada, se espera hasta volver a ser presionado.

Otras dos entradas (figura 3.14 y 3.15) son para detectar la posición del actuador A por medio de interruptores mecánicos colocados sobre la prensa.

Las últimas dos entradas son para conocer la posición del actuador B, por medio de sensores magnéticos en lugar de mecánicos, a causa de los ajustes mecánicos que constantemente se realizan sobre el desplazamiento del carro (figura 3.16 y 3.17).

En los actuadores C y D se evitó el uso de sensores y se optó por programar retardos para permitir que los actuadores alcancen a desplazarse. Esta decisión se prefirió en primer lugar por su ubicación, aunque pudo utilizarse interruptores magnéticos, su uso incrementaba bastante el costo del diseño y su presencia no es tan indispensable y en segundo lugar a causa de que esta máquina no realiza una producción en línea, es decir, no siempre realiza la misma pieza, el actuador D puede no estar presente dependiendo del molde de la pieza que se esté produciendo; así, con la finalidad de no tener que estar reprogramando la acción de los sensores del actuador D constantemente, se evitó su uso, de esta manera únicamente se reprogramará el microcontrolador cuando en realidad sea indispensable.

Las salidas del control son contactos de un relevador compacto, cuando se activa una salida el contacto del relevador cierra y cuando se apaga el contacto abre. Las bobinas de los relevadores son controladas por puertos del microcontrolador programados como salidas, no obstante hay que realizar un acoplamiento entre el pin del microcontrolador y la bobina del relevador, pues el microcontrolador no proporciona la suficiente corriente para activar la bobina, además, no hay que olvidar que los contactos de los relevadores producen arcos eléctricos al conmutar, provocando un ruido severo que puede afectar al microcontrolador. Entonces otra regla para evitar ruidos es realizar un aislamiento entre ambos dispositivos, por lo que hubo que auxiliarse ahora de los opto acopladores 4N26 siguiendo la recomendación sugerida por el *datasheet* para salidas digitales, pues requiere menor corriente para activar su fototransistor. Así, al colocar

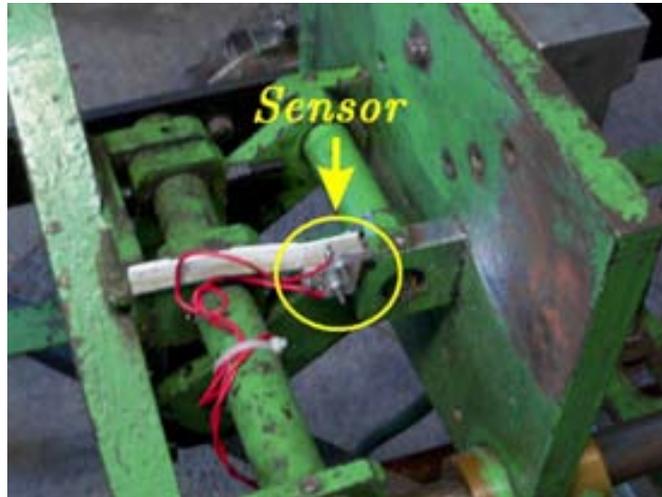


Figura 3.15: *Sensor A-*



Figura 3.16: *Sensor B+*



Figura 3.17: *Sensor B-*

el opto acoplador, el microcontrolador sólo verá que prende un LED, mientras del lado del fototransistor se activará el relevador.

El fototransistor por si solo no puede activar la bobina del relevador pues no soporta la corriente de 80 mA que se necesita para activarla, que en este caso es a 5 V, por lo que se utilizó el transistor PN2907. Sólo hay que calcular la resistencia de base (R_B , figura 3.18) que proporcione la suficiente corriente para permitir la conmutación del relevador. También hay que considerar en el acoplamiento que al energizar la bobina del relevador, ésta almacenan energia que será liberada en forma de corriente al apagarse el relevador, por lo que se coloca un diodo 1N4007 para evitar que esta corriente pueda descargarse sobre el transistor y dañarlo, así el diodo proporciona un camino a la corriente almacenada en la bobina, protege al transistor y evita el ruido. Sin embargo eso no es suficiente para proteger al microcontrolador del ruido, además la fuente que alimenta a los relevadores debe ser independiente a la del circuito de control por lo que se diseñó una tercera fuente dedicada a alimentar las bobinas de los relevadores.

Al igual que las entradas, las salidas se construyeron considerando un común, es decir, los contactos comunes de todos lo relevadores se unieron y el otro se dejó libre. De esta manera, al cerrar el relevador, se deja pasar los 24 V a la carga (figura 3.19). En este caso la carga es la bobina de la electroválvula, por lo que al igual que en la etapa anterior hay que colocar un diodo 1N4007,

Para la instalación del molde o ajustes mecánicos es necesario poder manipular cada uno de los pasos de forma independiente, por lo que se agregó un interruptor que polariza una de las dos líneas; cuando el interruptor está en la posición 1, la alimentación es para los relevadores por lo que el microcontrolador tiene el control de la máquina; pero si se coloca en la posición 2, la alimentación se va a los botones (B1 - B10), colocados sobre la carátula del control, mediante los cuales puede activarse independientemente cualquier bobina. Así puede colocarse la máquina en la posición que se desee. Algo importante que se tiene que recordar es que los sensores son alimentados con la misma fuente de 24 V, es decir, están conectados a la línea 1 (automático),

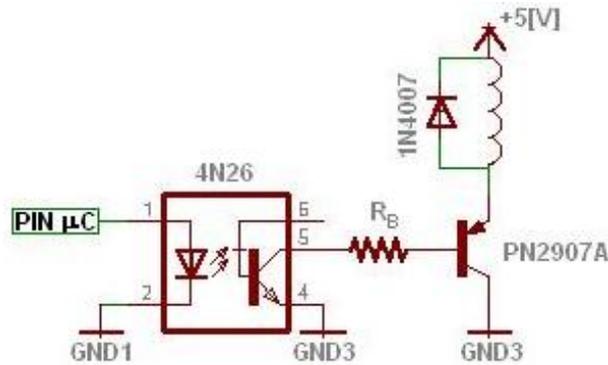


Figura 3.18: Acoplamiento entre el pin del μC y el relevador

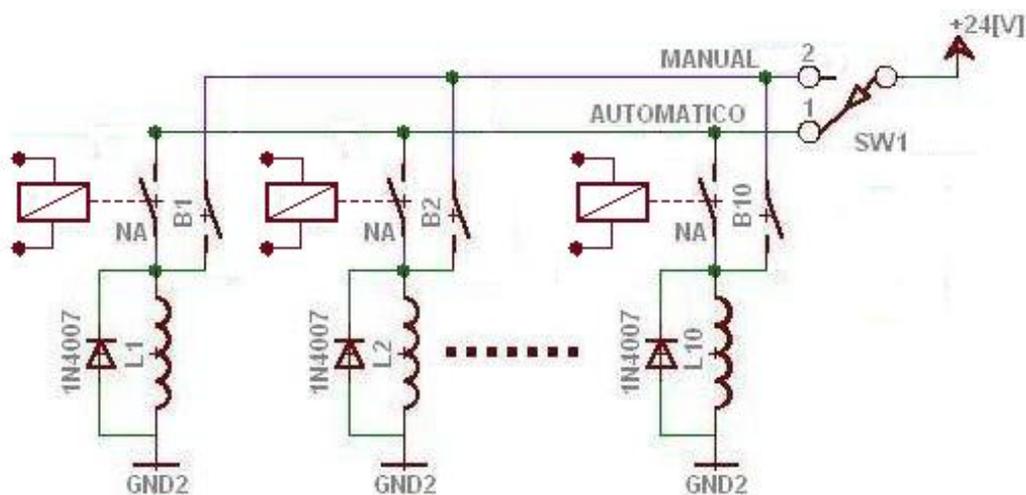


Figura 3.19: Configuración de las salidas

de esta forma al cambiar el interruptor a la posición 2, los sensores son desconectados al mismo tiempo que las salidas, así evitamos introducir señales al microcontrolador.

Como se mencionó en el capítulo dos, la secuencia tiene ciertos aspectos a considerar. El paso 6 trata de un tiempo, *tiempo de soplado*, que depende de la pieza que se está fabricando: Si es grande, el tiempo puede durar de 4 a 6 segundos, pero si es pequeña quizás 1 ó 2 segundos. Igualmente en el paso 8 se menciona otro tiempo que hay que dar para desfogar el aire introducido por el soplador, *tiempo de apertura*, pues si la prensa se abre al instante se corre el riesgo que la figura se rompa, que igualmente depende de la pieza; este tiempo generalmente no cambia mucho, puede estar entre 1 y 3 segundos. Estas dos variables contienen información que cambia constantemente, por lo que es el operador quien establece los tiempos a través de una interfaz de comunicación (capítulo 5). Los valores de estas variables son almacenadas en una memoria AT24C512, que tiene la capacidad de guardar 512 KB, su tamaño es práctico (encapsulado DIP de 8 pines), puede ser escrita ó borrada eléctricamente (EEPROM) y mantiene la información

aún sin alimentación, por lo cual es ideal para la aplicación.

El microcontrolador se comunica con la memoria por medio del protocolo *Two Wire Interface* ó por sus siglas en ingles TWI (el protocolo TWI es compatible con I^2C desarrollado por PHILIPS) que sólo requiere de dos líneas SDA (serial data) y SCL (serial clock) para realizar la transferencia de información entre la memoria y el microcontrolador. La memoria tiene 65536 localidades y cada una puede guardar 8 bits. Para realizar el control se ocupan tres localidades que perminten almacenar el tiempo de soplado, el tiempo de apertura y una variable, *Activoc*. Esta variable se usa para poder encender o apagar el control de forma digital. Cuando su valor es cero el control esta apagado, pero si es uno el control estará activado. Esta es una medida de precaución, así al conectar el control, éste no hará nada hasta que se active.

Localidad	Dato
0x000A	Tiempo de soplado
0x000B	Tiempo de apertura
0x0013	Activoc

Tabla 3.1 Mapa de memoria para el control

3.3. Programación del control de la máquina de soplado de plástico

Antes de empezar la programación es recomendable elaborar un diagrama de flujo que muestre gráficamente los pasos que realizará la máquina, para ello se necesito de los 11 pasos descritos en el capítulo 2. La figura 3.20 muestra el diagrama general de como se lleva acabo el control. Los diagramas de flujo con más detalle del proceso pueden ser consultados en el apéndice B.

En la figura 3.20 se pueden observar dos diagramas de flujo, el del lado izquierdo corresponde al programa principal, mientras que el otro hace referencia al programa que se ejecuta cada que ocurre una interrupción externa. La flecha sólo indica que puede ejecutarse en cualquier parte del programa. El programa principal comienza inicializando el microcontrolador, es decir, programa los puertos como entradas ó salidas, activa las resistencias *pull up* e interrupciones, enseguida desactiva el control poniendo la variable *Activoc* a cero y manda un pulso (protocolo sección 5.3), después permanece en ese estado hasta que por medio de la interfaz se cambie el valor de *Activoc* a uno; cuando esto sucede, lo primero que hace el control es leer cada uno de los sensores de la máquina para determinar la posición de sus actuadores y así poder colocar la máquina en condiciones iniciales. Una vez que la máquina está lista para iniciar el proceso, el programa espera que el botón de inicio sea presionado por el operador y cuando esto ocurre se ejecuta la secuencia de elaboración de la pieza. Una vez terminada, espera nuevamente a que se presione el botón para iniciar un nuevo ciclo. La secuencia fue colocada en un ciclo infinito, de tal forma que cada que se termina de ejecutar, revisa la variable *Activoc*, para saber su valor y determinar si el control se encuentra activo o no, pues en caso de estar desactivado se queda esperando en un ciclo hasta que su valor cambie nuevamente.

El programa de servicio a la interrupción se encarga de actualizar el valor de las variables. Cada que ocurre una interrupción, el microcontrolador realiza una lectura a los registros de la memoria para conocer su valor y aplicarlo al programa. El siguiente paso es transformar los

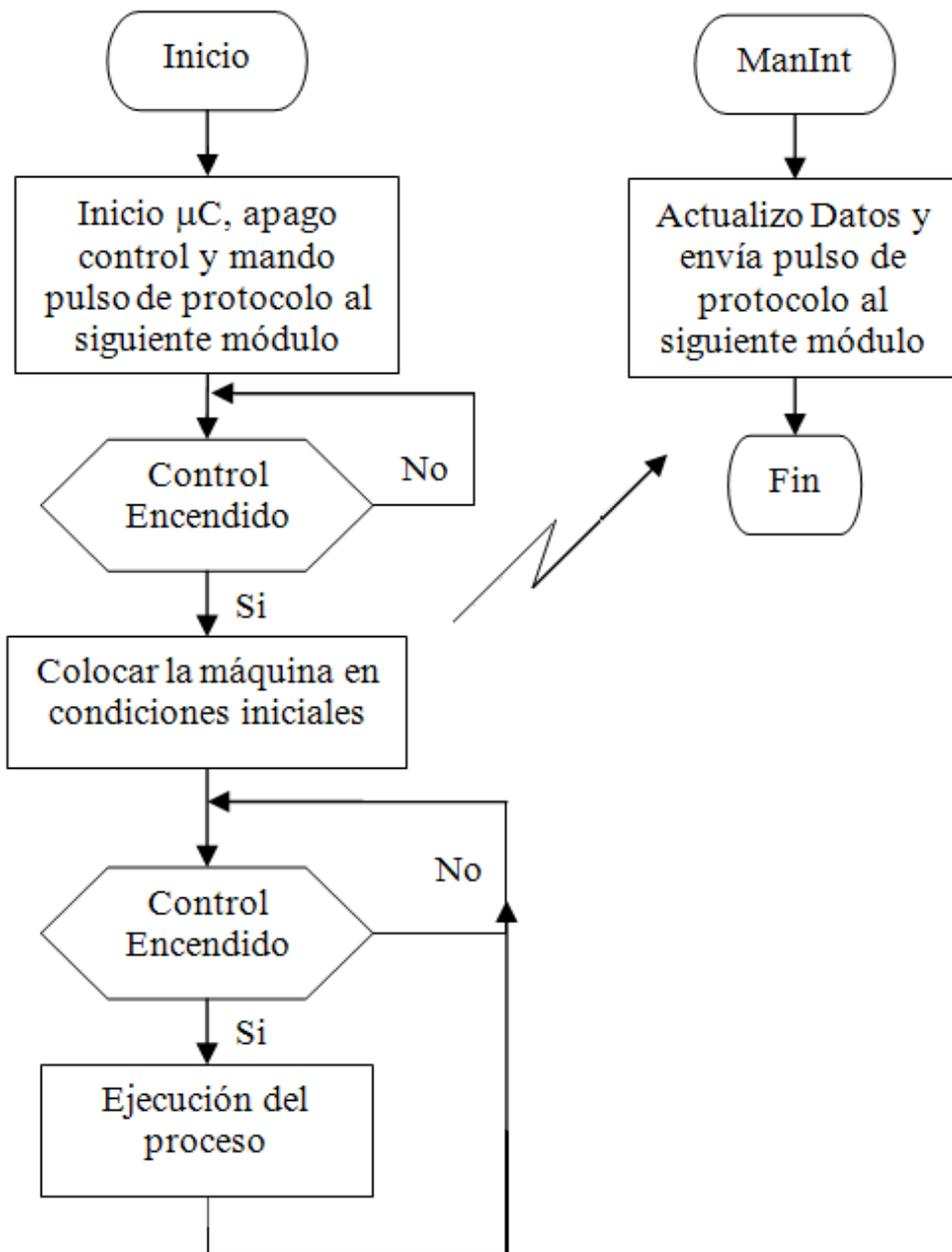


Figura 3.20: Diagrama de flujo del control de la máquina

diagramas a un lenguaje de programación, para ello se utilizó el winAVR (se pronuncia Whenever) un software completamente libre, que permite escribir la programación (código fuente) en lenguaje C gracias a su compilador (avr-gcc) encargado de generar el archivo hexadecimal.

Para la elaboración del código fuente se ocuparon las librerías de C más la librería twi0.h que fue desarrollada para comunicarse con la memoria serial. El código fuente compilado genera un archivo hexadecimal. Para descargarlo al microcontrolador hay varios métodos que se describen a continuación.

El primer método es utilizar un programador universal como TOP MAX, que permite descargar el programa sobre una amplia gama de dispositivos de diversas compañías. El segundo método es posible a causa de que los microcontroladores AVR pueden ser programados por medio de su puerto SPI (*Serial Peripheral Interface*) a través del puerto LPT de una PC. Existen diversos tipos de programadores en la red que muestran como implementarlos, los cuales en general son bastante buenos y económicos, al utilizar integrados cuyos costos son bastante accesibles. El tercer método requiere forzosamente del segundo pero únicamente una vez, debido a que es necesario utilizar un programador para grabar en una sección de la memoria flash un pequeño programa llamado BOOTLOADER. El bootloader permite descargar el programa a través del módulo UART (*Universal Asynchronous Receiver - Transmitter*) del microcontrolador, utilizando como interfaz al integrado MAX232, el cual convierte los voltajes de la PC a voltajes TTL, permitiendo la comunicación. Las principales ventajas son poder descargar el programa por el puerto serie sin necesidad de algún programador y solo requiere tres líneas de comunicación Tx, Rx y GND.

De todas estas opciones para esta aplicación se optó por cuestiones prácticas utilizar el programador universal TOP MAX, pues permite programar una amplia gama de microcontroladores AVR y cuenta con un software muy amigable por lo que facilita la programación de estos dispositivos.

Capítulo 4

Control de temperatura

4.1. Operación del sistema de calentamiento

La máquina de soplado de plástico necesita calentar el plástico (polietileno) introducido en el cono para generar el parison, las partes de la máquina que requieren calentamiento son: el cañón (figura 4.1) y el cabezal (figura 4.2).

El cañón es un tubo largo, dentro del cual está el husillo (figura 4.3) que está fijo en un extremo a un motor que lo hace girar, su principal función es cortar y empujar el flujo de plástico a través del tubo. El cabezal es la parte final del cañón por donde sale el parison, dentro de él se encuentra el corazón y la campana (figura 4.4) con los que se define el ancho y grosor del parison.

Ambas partes son calentadas por medio de resistencias de Nicromel (aleación de níquel y cromo) de 1400 W a 220 V. Sobre el cañón se encuentra tres resistencias (R_1 , R_2 y R_3) y sobre el cabezal una (R_C). Las resistencias son a 220 V para disminuir la corriente y evitar el uso de un calibre grande en la conexión, las resistencias se encuentran conectadas en estrella (figura 4.5), para tratar de repartir la carga sobre las tres fases, pero como se tienen cuatro resistencias, la estrella estará desequilibrada, por lo que se debe tener en cuenta el calibre del lado desequilibrado pues por el circulará una mayor corriente.

4.2. Termopar y pirómetros

La temperatura producida por las resistencias debe ser controlada para permitir el flujo del plástico a través del cañón, pues si no es así, podría trabajarse con una temperatura baja y el material no tendría la consistencia necesaria para ser empujado por el husillo correctamente, lo que provocaría un esfuerzo en el motor que puede dañarlo. Por otra parte si está muy caliente el plástico puede volverse líquido, con lo que se hace imposible trabajarlo. Para evitar estas situaciones, la temperatura es controlada por medio de tres pirómetros que ocupan como sensores los termopares.

Los termopares son sensores de temperatura (figura 4.6) basados en el efecto Seebeck que establece que la unión de dos metales diferentes genera una tensión que es función de la diferencia de temperaturas entre ambos extremos, uno caliente y otro frío. Los termopares son ampliamente utilizados en la industria como sensores de temperatura, por ser económicos y soportar un amplio rango de temperaturas, sin embargo, hay que tener cierto cuidado con ellos para no cometer



Figura 4.1: *Cañon*



Figura 4.2: *Cabezal*



Figura 4.3: Husillo



Figura 4.4: Corazón y Campana

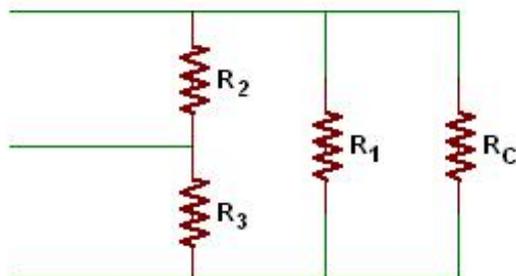


Figura 4.5: Conexión de las resistencias



Figura 4.6: *Termopar*

errores en las mediciones. Hay que considerar que cualquier conexión de metal crea una nueva unión, por lo tanto se producirá una tensión que puede sumarse o restarse a la tensión de la unión caliente, provocando que nuestra medición sea errónea. Su salida es del orden de los mV, por lo que es propenso a los ruidos eléctricos y, por último, para poder realizar mediciones con ellos se tiene que realizar la compensación de unión fría para compensar el hecho que el extremo frío se encuentra a temperatura ambiente.

Existe una gran variedad de termopares, diseñados para distintas aplicaciones, cada uno de los cuales ha sido estandarizado y caracterizado por el National Institute of Standards and Technology. En la tabla 1 del Apéndice D se muestran los tipos y características de los termopares más comunes.

Un pirómetro (Figura 4.7) es un control de 2 posiciones (ON - OFF), que se encarga de controlar la temperatura en un rango de $50^{\circ}C$ a $450^{\circ}C$. Éste dispositivo utiliza un termopar como sensor para medir la temperatura y dentro de él se encuentra un pequeño relevador que cierra mientras la temperatura es inferior a la deseada y abre en cuanto es rebasada.

Con la ayuda de estos dos elementos se establece una temperatura sobre el cañón y cabezal, que no es uniforme a lo largo de estas dos partes, sino se encuentra fraccionada en tres zonas para ir calentando el plástico conforme fluye al cabezal. La zona tres es la parte inicial del cañón, en esta zona se empieza a calentar el plástico por lo regular está entre $100^{\circ}C$ y $130^{\circ}C$. En esta zona no debe aplicarse tanto calor, debido a que se encuentra en una zona muy cercana al cono donde se deposita el plástico, por lo que si llega a calentarse demasiado puede formarse una bola de plástico que obstruya el flujo del mismo. La zona dos se localiza a la mitad del cañón y se encuentra a una temperatura entre los $135^{\circ}C$ y $150^{\circ}C$, un poco más elevada que al inicio para comenzar a preparar la salida. La zona uno es la parte final del cañón, junto con el cabezal, y se encuentra a una temperatura entre $160^{\circ}C$ y $180^{\circ}C$. Esta es la zona de mayor temperatura por



Figura 4.7: *Pirómetro*

ser la salida final del plástico. Como puede verse, el calentamiento es paulatino, para permitir un flujo uniforme.

Sobre cada zona descrita se coloca un termopar para medir la temperatura y un pirómetro para conectar y desconectar la resistencia; sin embargo, los relevadores contenidos dentro de los pirómetros no se conectan directamente sobre la resistencia debido a que su relevador soporta sólo 7A de corriente y si tenemos en cuenta que las resistencias son de 220V a 1400W, entonces aplicando la ley de Joule $P = VI$, se obtiene que la corriente que circula por ellas es de aproximadamente 6.36A un valor cercano a su valor máximo, así que por protección y para prolongar más la vida útil de nuestros pirómetros, colocamos como intermediario un contactor (dispositivo eléctrico que corta la corriente eléctrica a la carga) de 30A de esta manera el pirómetro solo cierra su bobina y él se encarga de conectar o desconectar la resistencia.

La figura 4.8 muestra el circuito de control donde P1, P2 y P3 son los contactos del relevador de los pirómetros y L1, L2 y L3 las bobinas de los contactores. La figura 4.9 muestra el circuito de potencia, los contactores utilizados son bipolares, por lo que al cerrar conecta la resistencia entre dos fases.

4.3. Electrónica del control de temperatura

Los pirómetros son controles muy eficientes, sin embargo, con la finalidad de reducir espacio en el tablero y poder manejar la temperatura a través de nuestra interfaz de comunicación (capítulo 5), se desarrolló un control de temperatura empleando electrónica digital.

El circuito de potencia (figura 4.9) se mantiene igual, lo único que se pretende es desarrollar un control de temperatura que sustituya a los pirómetros. El primer paso para solucionar el problema es seleccionar la manera de poder manipular las resistencias de forma que mantengan la temperatura constante, aprovechando que el primer módulo encargado de manipular las

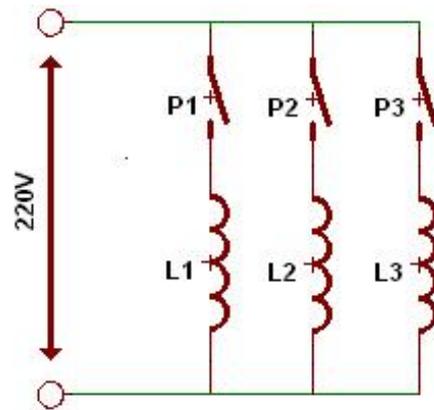


Figura 4.8: *Circuito de control*

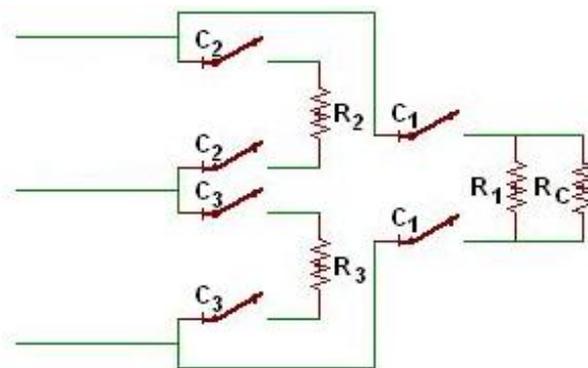


Figura 4.9: *Circuito de Potencia*

electroválvulas fue desarrollado sobre el microcontrolador ATmega32, este nuevo módulo encargado de controlar la temperatura se desarrolló sobre otro dispositivo de la misma familia para establecer fácilmente una comunicación entre ambos.

Para este módulo se utilizó el Atmega8 por las siguientes características:

- 8 Kbyte en memoria Flash
- 512 Kbyte en memoria EEPROM
- Interface TWI
- Tres puertos bidireccionales
- ADC (Convertidor Analógico a Digital) de 10 bits
- Oscilador Interno

Al igual que en el módulo anterior hay que implementar la electrónica correspondiente para interactuar con los sensores y contactores, tomando en cuenta que las condiciones adversas del ambiente industrial pueden originar los problemas descritos en la sección 3.2.

La construcción de este control resulta un poco más sencillo, gracias a que pueden utilizarse algunos acoplamientos y configuración de la sección 3.2. Lo primero que puede reutilizarse es la fuente de alimentación construida para el primer módulo. El ATmega8 no va a generar ningún ruido que pueda perturbar al ATmega32 por lo que ambos microcontroladores pueden alimentarse de la misma fuente (figura 3.11).

Para este sistema de control de temperatura se usan termopares, pero para usar el termopar hay que acondicionar la señal considerando las precauciones descritas en la sección 4.2.

Todas las precauciones que pueden utilizarse para realizar una buena medición con el termopar son importantes, pero en esta aplicación sólo nos centraremos en dos:

- Realizar la compensación de unión fría.
- Amplificar la señal del termopar que es de milivolts.

Retomando el funcionamiento del termopar, éste genera una diferencia de potencial (efecto Seebeck) proporcional a la diferencia de temperaturas entre ambas uniones, por lo que si deseamos conocer la temperatura de la unión caliente, necesitamos conocer la temperatura de la unión de referencia.

Como se observa en la figura 4.10 el voltaje que entrega el termopar incluye la temperatura T_2 , así que se necesita colocar la unión en una zona donde se conozca la temperatura. La forma de facilitar la medición es colocar la sobre un baño de hielo, esto hace que $T_2 = 0^\circ C$, entonces la diferencia de potencial solo será función de T_1 , a esto se le conoce como compensación de unión fría (figura 4.11).

En nuestro caso hay que realizar la compensación para medir el voltaje Seebeck, a causa de las dos uniones que se crean al conectar el termopar a nuestro circuito. Estas dos uniones producen voltajes opuestos pero si se considera que ambas uniones están a la misma temperatura tendrán el mismo valor, por lo tanto el voltaje del termopar es la suma en la unión en medición más las dos uniones creadas.

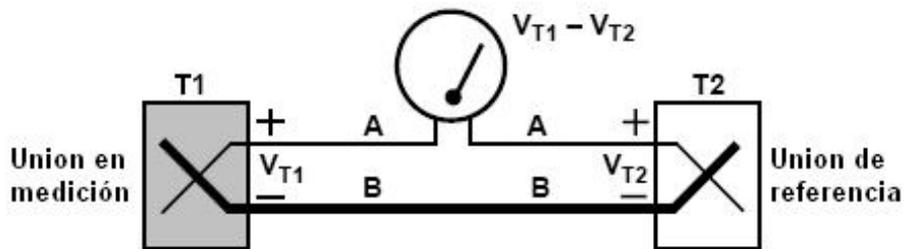


Figura 4.10: *Efecto Seebeck*

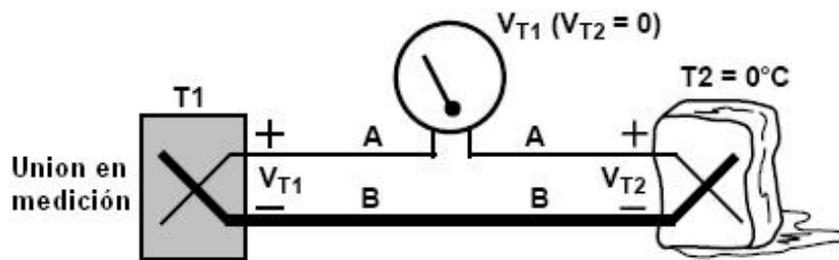


Figura 4.11: *Compensación de unión Fría*

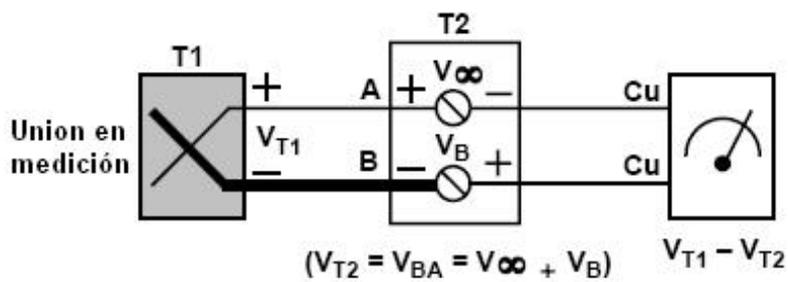


Figura 4.12: *Medición del voltaje del termopar*

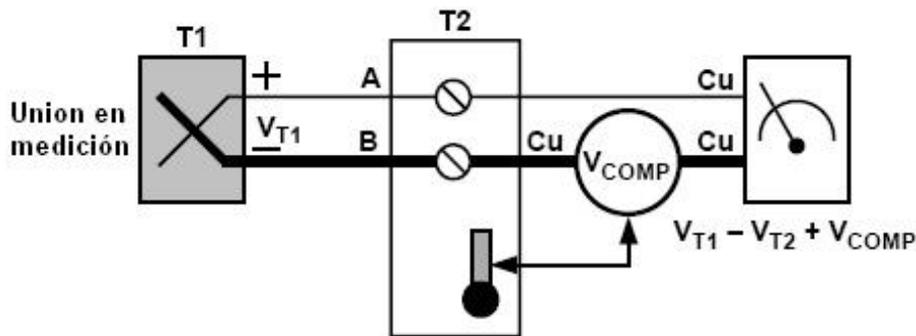


Figura 4.13: *Compensación*

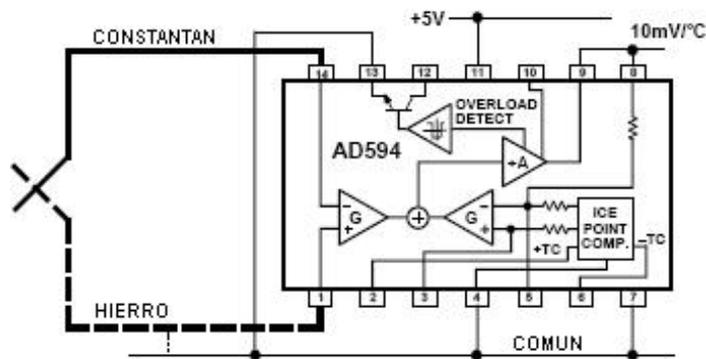


Figura 4.14: *Conexión básica*

En la práctica no se utiliza el baño de hielo para compensar el circuito (figura 4.12), lo que se hace es agregar al lazo del termopar un voltaje (V_{COMP}) igual pero opuesto al de las uniones (figura 4.13), de forma que $V_{COMP} = V_{T2}$, así solo se medirá la temperatura en $T1$.

La pregunta ahora sería ¿cómo se agrega este voltaje?. Hay varios circuitos integrados acondicionadores de señal que solucionan este problema, en esta aplicación se utilizó el AD594AQ de Analog Devices, tal como se muestra en el diagrama a bloques y conexión de la figura 4.14. El AD594AQ es un acondicionador de señal para termopares tipo J formado por la unión del Hierro con el Constantan (aleación formada de cobre y níquel), internamente contiene un amplificador de instrumentación diferencial, donde se obtiene y amplifica la señal del termopar. Además, incluye un compensador de unión fría que va a generar un voltaje en función de la temperatura local. Este voltaje es amplificado y sumado al del termopar para realizar la compensación. Finalmente el voltaje resultante es nuevamente amplificado para obtener una salida de $10\text{ mV}/^{\circ}\text{C}$. El compensador cuenta además con las conexiones $+TC$ y $-TC$ que mantienen a la misma temperatura las uniones formadas por el termopar y el cobre del PCB, para evitar errores en la medición.

El AD594AQ satisface los dos problemas que se tenían, proporciona la compensación de unión fría y amplifica la señal, solo falta cubrir otro punto importante que debe considerarse.

Los termopares no son del todo lineales, por lo que hay que linealizar la salida. Esto se puede hacer de dos maneras: mediante circuitos integrados que linealizan la señal o por software aprovechando la capacidad del ATmega8. Ambas opciones son buenas pero revisando la tabla 2 del apéndice D puede verse que al inicio es casi lineal, el error se incrementa conforme aumenta la temperatura. Para esta aplicación la temperatura usual esta debajo de los $200^{\circ}C$ aunque se da un rango de más para tener control hasta los $280^{\circ}C$. En la región donde se piensa trabajar, el error máximo es casi de $7^{\circ}C$ y puede alcanzar los $14^{\circ}C$ al final de la escala, no obstante, se observa que sobre la región de trabajo puede desprejarse el error, debido a que no es un sistema que exija tanta precisión, entonces el microcontrolador puede trabajar directamente la salida del acondicionador.

La configuración de las entradas se construyó considerando el hierro del termopar como común de las entradas, tal como se establece en su *datasheet*, la posición de los sensores sobre la máquina se mantuvo, conectándose cada uno a un AD594AQ para obtener las señales de las tres zonas. El ATmega8 cuenta con un ADC de seis canales de 10 bits, entonces utilizando tres de ellos podemos introducir las señales analógicas al microcontrolador para su procesamiento. Cada acondicionador puede operar con una alimentación de 5 a 30 Volts, pero como el hierro del termopar es conectado a tierra, esta debe ser la misma del ATmega8 para que pueda realizar la conversión, por lo tanto los acondicionadores son alimentados con la misma fuente que los microcontroladores, esto no ocasiona ningún problema pues no producen ningún ruido que los pueda perturbar.

Las salidas del control son a relevadores, lo que permite reutilizar el acoplamiento de la figura 3.18. La configuración de las salidas se construyó de misma manera que el control de la máquina, es decir, considerando un común de 220V, sólo que esta vez no tiene como carga una bobina de corriente directa sino una de alterna por lo que el diodo no sirve en esta ocasión para eliminar el ruido de los arcos eléctricos.

Lo que se hizo para evitar este ruido fue construir un PCB externo con relevadores a 12V de corriente directa de tal manera que las salidas del control cerrarán las bobinas del PCB externo y estos a su vez activarán las bobinas de los contactores (figura 4.15), así este PCB va a servir como un filtro, porque el control únicamente verá que cierra la bobina de un relevador de corriente directa mientras del otro lado activa la bobina del contactor. Además, ahora la carga ya son bobinas de relevadores y puede eliminarse el ruido colocando un diodo.

La electrónica se ha vuelto a diseñar de manera que el ATmega8 se encuentra encapsulado para protegerlo de cualquier ruido, sin embargo falta mencionar al AT25C512 otro elemento que interactúa con el microcontrolador.

Anteriormente la temperatura de cada zona era seleccionada en el pirómetro (figura 4.7) por medio de una perilla y su carátula graduada; así, para seleccionar una temperatura solo se giraba hasta posicionarla en la temperatura indicada sobre la carátula. Ahora que se ha eliminado este elemento, necesitamos almacenar la temperatura de referencia de cada zona, para no tener que estar ajustándola cada que se prende la máquina, para eso se utiliza la memoria serial del primer módulo.

En la memoria, la temperatura de cada zona se guarda en dos registros, porque en uno puede almacenar como máximo una temperatura $255^{\circ}C$, pero si utilizamos dos registros puede almacenarse hasta $65535^{\circ}C$, más que suficiente para la escala que va de $1^{\circ}C$ a $280^{\circ}C$. Al igual que el módulo anterior se utilizará una variable llamada *Activot* para poder prender o apagar

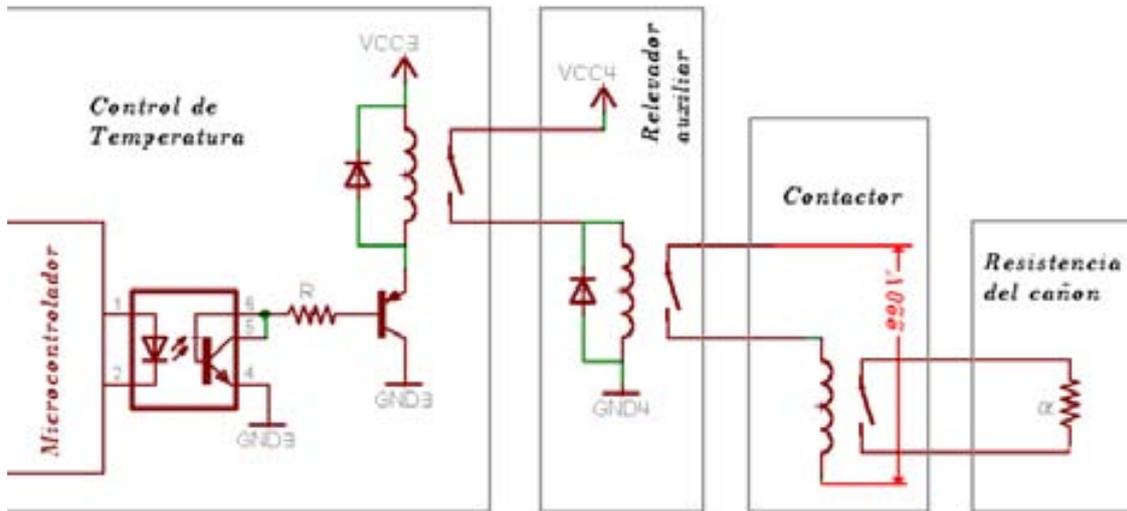


Figura 4.15: Diagrama de conexión del pin del μC con la resistencia

el control a través de la interfaz de comunicación, funciona exactamente como la otra variable, cuando su valor es uno el módulo trabaja normalmente y cuando es cero deja de operar hasta que cambie su valor, esta variable tiene como propósito desactivar las resistencias cuando se estén realizando ajustes mecánicos.

Localidad	Dato
0x000D	Zona1 Byte Bajo
0x000E	Zona1 Byte Alto
0x000F	Zona2 Byte Bajo
0x0010	Zona2 Byte Alto
0x0011	Zona3 Byte Bajo
0x0012	Zona3 Byte Alto
0x0014	Activot

Tabla 4.1 Mapa de memoria para la Temperatura

4.4. Programación del control de temperatura

En el microcontrolador ATmega8 se programaron tres controles de dos posiciones. Este tipo de control accionador sólo tiene dos posiciones: prendido o apagado (ON - OFF). Su funcionamiento es simple: consideremos una señal $v(t)$ (señal de salida del sistema) y una señal de error $e(t)$ (señal producida por la diferencia de la señal de referencia y la señal de salida), la salida va a permanecer en un valor máximo o mínimo según la señal de error sea positiva o negativa, de esta forma:

$$\begin{aligned}
 v(t) &= M_1 && \text{para } e(t) > 0 \\
 v(t) &= M_2 && \text{para } e(t) < 0
 \end{aligned}$$

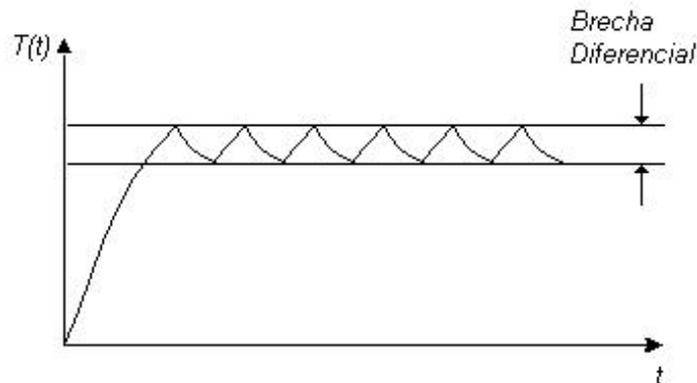


Figura 4.16: Temperatura $T(t)$ en función del tiempo

En este caso $M_1 = 5V$ y $M_2 = 0V$, así M_1 conecta la resistencia y M_2 la desconecta. El control tiene programado deliberadamente un rango para que la señal de error actuante pueda desplazarse antes de producir la conmutación conocida como brecha diferencial. La ventaja de agregar un rango es evitar el constante accionamiento de los mecanismos.

En la figura 4.16 puede verse que entre más pequeña se la brecha diferencial más cercano del valor de referencia se estará, sin embargo se incrementa el número de conmutaciones del elemento, lo que provoca disminución en su vida útil.

El diagrama de flujo general de la programación de los controles se muestra en la figura 4.17, para mayor detalle puede consultarse el apéndice B. El programa principal comienza inicializando al microcontrolador, programando los puertos como salidas o entradas y activando el módulo ADC. Cuando termina, lo primero que verifica es el registro *Activot* para saber si está prendido o no; de estar apagado, desactiva los relevadores de las tres zonas y espera hasta que cambie su valor, en caso contrario empieza la ejecución del programa.

El primer control en ejecutarse corresponde a la zona uno, lo que hace el control es leer su termopar a través del canal cero del microcontrolador. Una vez realizada la medición de la temperatura obtiene la señal de error para compararla, si es menor de cero y esta fuera de la brecha diferencial desconecta la resistencia pero si el error es mayor a cero y se encuentra fuera de la brecha diferencial la conecta. El siguiente control en ejecutarse pertenece a la zona dos, su función es la misma sólo que lee su termopar por el canal uno. El control de la zona 3 es el último en ejecutarse y funciona igual que los dos anteriores pero con la diferencia de que la lectura de su termopar es por el canal dos. Al final del programa se genera un retardo de 4 segundos antes de volver a muestrear la señal.

El programa correspondiente a la interrupción actualiza los valores de temperatura cuando estos han sido cambiados por la interfaz de comunicación, dentro de la interrupción se realiza un escalamiento para manejar los datos, se sabe que el acondicionador entrega $10mV/^\circ C$, mientras que la relación de conversión del ADC del microcontrolador se obtiene a partir del voltaje de referencia (5 V) y el número de bits que puede representar el ADC(10 bits), por lo tanto para conocer el voltaje escalado basta con multiplicar el voltaje analogico de entrada por la relación de escalamiento que en este caso es $\frac{1024}{5} \text{ escala}/V$ ó $0.2048 \text{ escala}/mV$, entonces para saber

cuanto representa en escala $1^{\circ}C$ se multiplican ambas relaciones (ADC y acondicionador), con lo que se obtiene $2.048 \text{ escala}/^{\circ}C$. Esta es la relación de conversión por la que debe multiplicarse la temperatura para saber el valor en escala que requiere el microcontrolador para representarla. La relación anterior es un número flotante, algo complicado, más no imposible de manejar en los AVR, sin embargo para simplificar su manipulación se programa como $\frac{256}{125} \text{ bit}/^{\circ}C$, de esta forma primero se multiplica y se después divide.

La brecha diferencial para los tres controles es de $7^{\circ}C$ menor a partir del valor de referencia establecido, es decir, si el operador coloca una temperatura de $160^{\circ}C$, los resistencias estarán conectadas mientras estén por debajo de la temperatura, una vez alcanzada la temperatura, la resistencia no se volverá a conectar hasta que la temperatura sea menor a $153^{\circ}C$, este rango fue elegido pensando que la temperatura no es una variable que cambie rápidamente, lo que permite dar un amplio rango.

Como se mencionó el ATmega 8 cuenta con un oscilador interno para evitar el uso del cristal de cuarzo, que se activa cambiando los *fuse bits*, los cuales son bits del microprocesador que determinan ciertos aspectos del modo de funcionamiento, para el ATmega8, estos bits no pueden cambiarse en tiempo de ejecución sino sólo a través de un programador (por ejemplo TOPMAX). Algunas de las características que controlan estos *fuse bits* son:

- Habilitar Bit 6 del puerto C como I/O ó RESET
- Habilitar perro de guardia (watchdog)
- Selección del tamaño de espacio para el bootloader
- Selección del dispositivo de reloj

Todas estas opciones son configuradas por medio de los *fuse bits*, mediante el encendido ó apagado de estos (mayor información *datasheet* ATmega8). Para esta aplicación se configuró "Selección del dispositivo de reloj" mediante el cual se activó el oscilador interno a una frecuencia de 1000 Hz .

La escritura y compilación del código fuente fue hecha nuevamente con winAVR y programado con el TOPMAX, por último cabe mencionar que aparte de la librería *twi0.h* fue desarrollada la librería *sensor.h* para manejar el ADC.

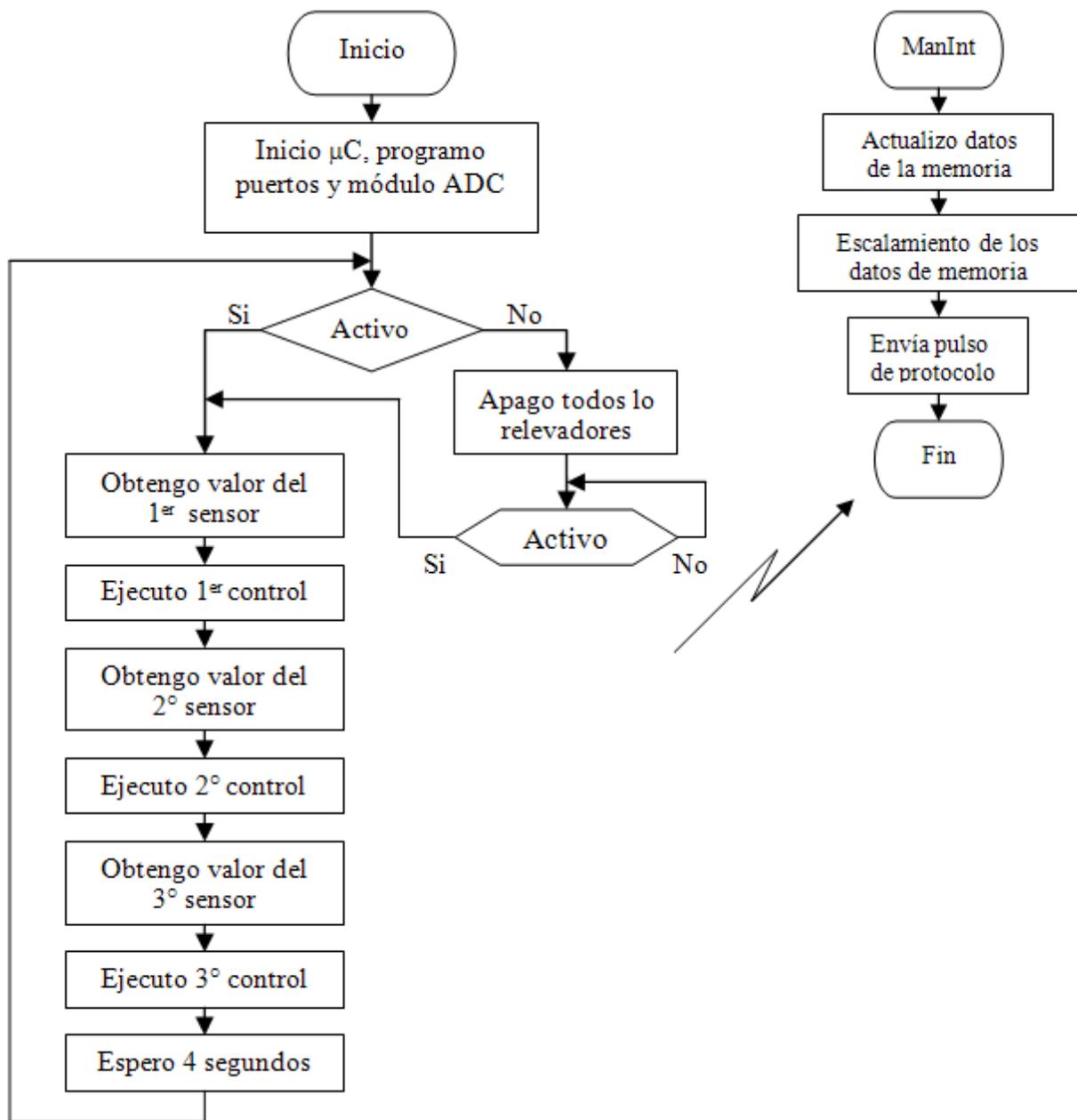


Figura 4.17: Diagrama de flujo del control de temperatura

Capítulo 5

Interfaz de comunicación e implementación final

5.1. Electrónica de la interfaz de comunicación

El propósito de crear una interfaz de comunicación es facilitar al usuario la comunicación con el sistema. Las interfaces se componen por dispositivos de entrada (ratones o teclado) para introducir datos o instrucciones y dispositivos de salida (pantallas, sonidos o luces) que sirven de realimentación al usuario para comprobar que la información está siendo correctamente proporcionada al sistema.

La interfaz en esta aplicación fue desarrollada para que el usuario pueda ajustar parámetros de los controles en función de la pieza que está elaborando. Este tercer módulo fue diseñado con base en un microcontrolador ATmega8, debido a las características ya mencionadas en la sección 4.2 y para facilitar la realización de un protocolo que permita comunicar a los tres microcontroladores AVR.

Para poder realizar una comunicación hombre - máquina, hay que establecer cual va a ser el medio de entrada de los datos y cual va a ser el medio de salida que despliegue la información. La solución a estos problemas se encuentra en los periféricos que los AVR son capaces de manejar como los LCD, teclados matriciales, memorias, etc.

El primer periférico a ser utilizado en la interfaz es un LCD (Liquid Crystal Display), el cual va a ser el elemento de salida que desplegará la información. En general existen dos tipos de LCD:

- LCD alfanuméricos
- LCD gráficos

Los LCD alfanuméricos (figura 5.1) son aquellos que únicamente pueden visualizar números, caracteres y símbolos definidos por el usuario, mientras que los LCD gráficos (figura 5.2) no sólo permiten desplegar caracteres sino imágenes, esto es posible porque los LCD gráficos se componen por puntos y no por renglones, por ejemplo un LCD grafico de 128x64, nos indica que posee 128 puntos de longitud y 64 puntos de altura, en donde cada punto puede tomar dos estados prendido ó apagado, lo cual permite realizar combinaciones de estados sobre la pantalla para originar imágenes.



Figura 5.1: *LCD alfanumérico*



Figura 5.2: *LCD gráfico*

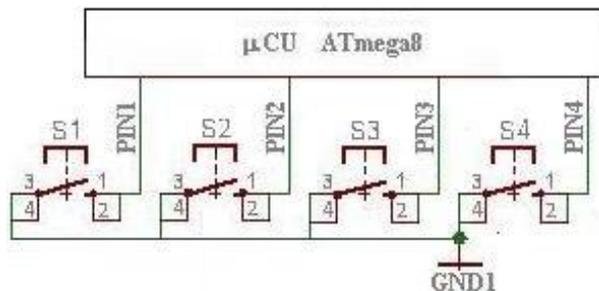


Figura 5.3: Teclado

Dado que los LCD alfanuméricos se restringen sólo a caracteres, se ocupa un gráfico (JHD12864A) para no limitar el diseño en caso de querer realizar alguna modificación a futuro. La forma de comunicarse con los LCD gráficos es a través de sus 18 pines. Algunos LCD gráficos cuentan con dos pines extras cuya función es activar el *backlight* encargado de prender una iluminación en el fondo de la pantalla que permita visualizar la información cuando haya poca luz.

La forma de poder dibujar una letra o imagen es por medio del protocolo establecido en su hoja de especificaciones, del cual no se dará detalle por no ser objeto de este trabajo.

El LCD gráfico trabaja con voltajes TTL ($5 V_{DC}$) por lo que los pines del LCD pueden ser conectados directamente a los pines del microcontrolador, sin ningún acoplamiento.

El segundo periférico es un teclado, el cual va a ser el elemento de entrada que permitirá introducir los datos que el usuario dé al sistema.

El teclado para la interfaz fue diseñado con botones nombrados de la siguiente manera:

- Aumentar (S1)
- Disminuir (S2)
- Aceptar (S3)
- Cancelar (S4)

Como su nombre lo indica los botones *aumentar* y *disminuir* se utilizan para incrementar o reducir el valor de alguna variable de los controles, mientras que *aceptar* y *cancelar* sirven para indicarle al sistema cuando algún dato que ha cambiado es rechazado o admitido. Los botones fueron colocados sobre un PCB externo y conectados al ATmega8, por medio de un cable plano. El teclado diseñado (figura 5.3) puede ser conectado directamente sobre los pines del ATmega8 sólo hay que recordar activar su resistencias *pull up* internas para cada pin que reciba señal del teclado.

El tercer y último periférico que va utilizar la interfaz es la memoria serial AT24C512. La información de los parámetros de ambos controles es almacenada allí, para ser consultada cada que se prende sistema o se actualizan los valores. La función de la interfaz es sobrescribir los registros de la memoria para cambiar los parámetros, por lo que debe tener comunicación con ella.

5.2. Programación de la Interfaz de comunicación

La programación del ATmega8 fue hecha para manejar tres periféricos. La idea general del código es desplegar menús y submenús para cambiar los valores de las variables. La estructura del programa puede verse en el diagrama de flujo de la figura 5.4.

El programa principal comienza iniciando los puertos como entradas o salidas y configurando los periféricos (LCD, teclado y memoria), después despliega sobre el LCD la pantalla principal, donde se encuentran agrupados por categorías los parámetros del control; enseguida hay una instrucción *switch*, que contiene una expresión única cuyo valor es comparado de izquierda a derecha para seleccionar alguno de los posibles procesos. Para este programa se evalúa la variable llamada *opción*, la cual es modificada por medio del teclado colocado sobre la carátula del gabinete (figura 5.11).

La primera opción ejecuta una pantalla que despliega las variables *tiempo de soplado* y *tiempo de apertura*, a continuación se encuentra un *switch* para seleccionar alguna de las dos, dependiendo de la variable que se elija será el registro que modifique la interfaz.

La segunda opción despliega las variables *Tzona1*, *Tzona2* y *Tzona3*, correspondientes a la temperatura de cada zona, después se encuentra otro *switch* para selección cualquiera de las 3 zonas para modificar su registro.

La tercera y última opción despliega las variables *Activoc* y *Activot* encargadas del prendido o apagado de los controles, nuevamente se encuentra a continuación un *switch* para seleccionar alguna de las dos variables para la modificación de su registro.

Como se observa, todos los bloques convergen a una pregunta lógica *if-else*, su función es determinar si el usuario en realidad desea cambiar los valores o no, esta pregunta es importante dado que a lo mejor se equivocó de opción y para no alterar el valor de las variables se hace la pregunta, en caso de ser cierta genera un pulso aproximadamente de 20 ms, este pulso consiste en activar un pin de salida (5 V) por 20ms y después desactivarlo (0 V), para avisar que ha terminado de leer la memoria y el siguiente módulo puede acceder a ella (protocolo sección 5.3), si es falsa despliega la pantalla principal y se queda esperando hasta que se elija nuevamente una opción.

El código fuente fue, al igual que los dos módulos anteriores, escrito y compilado con la ayuda del winAVR, sin embargo este código no sólo utiliza las librerías de C y la *twi0.h*, sino también se incluyeron librerías de *procyon*, estas librerías son completamente libres y fueron desarrolladas para ahorrar tiempo de diseño. Estas librerías proporcionan métodos (funciones de C) que permiten controlar diversos periféricos, sólo hay que incluirlas al programa para poder utilizarlas. Para esta interfaz se ocupa sólo las librerías *global.h*, *ks0108.h* y *gled.h* que permiten controlar el LCD gráfico.

La documentación de las librerías de *procyon* pueden ser consultada o descargadas de Internet, la dirección del sitio está en la bibliografía. Al igual que el módulo anterior, el ATmega8 de la interfaz se le configuró sus *fuse bits* para activar su oscilador interno a una frecuencia de 1000 Hz. La descarga del archivo hexadecimal y programación de *fuse bits* fue hecha con el programador TOP MAX.

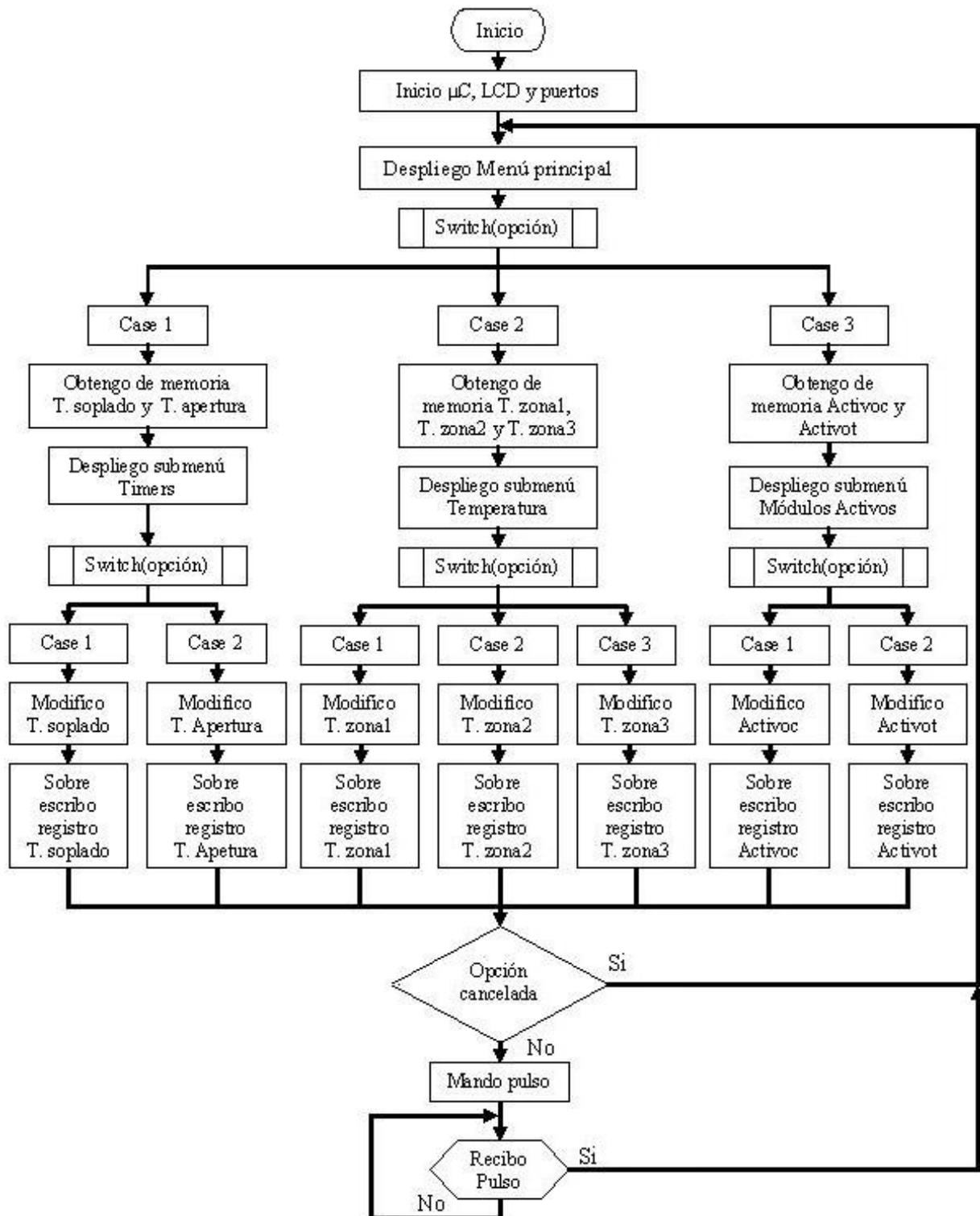


Figura 5.4: Diagrama de Flujo de la Interface

5.3. Implementación de los módulos

En capítulos anteriores se han mostrado los elementos y configuraciones de los controles, ahora que ya se ha explicado toda la electrónica y su programación, se mostrará en este apartado la forma en que se conectaron estos tres módulos (control de la máquina, control de temperatura e Interfaz) para su interacción.

A continuación se enlistan los bloques que se han mencionado a lo largo de este trabajo:

1. Fuente de 5 V_{DC} con filtro LC a la salida (Fuente1)
2. Fuente de 24 V_{DC} (Fuente 2)
3. Fuente de 5 V_{DC} (Fuente 3)
4. Fuente de 12 V_{DC} (Fuente 4)
5. Regulador de voltaje con supresor de picos
6. Control de la máquina (ATmega32)
7. Control de Temperatura (ATmega8)
8. Interfaz de comunicación (ATmega8)
9. Memoria (AT24C512)
10. Teclado
11. LCD gráfico
12. Entradas opto acopladas
13. Salidas a relevador
14. Relevadores Auxiliares
15. Acondicionadores de señal (AD594AQ)

La conexión de todos estos bloques puede verse en la figura 5.5, en ella puede observarse las entradas y salidas de cada uno de estos elementos. Como se ha mencionado, para lograr la comunicación de módulos se desarrolló un protocolo. Un protocolo es un lenguaje de comunicación que se requiere para lograr que dos o más dispositivos se entiendan.

En este caso es necesario implementar uno, porque cada vez que es alterada una variable por la interfaz, ésta debe avisarle a los demás módulos para que actualicen sus valores. La comunicación comienza cuando se altera algún valor de la interfaz, entonces ésta manda un pulso (figura 5.5, señal P1) al control de la máquina; este pulso es recibido en el pin de interrupción externa, lo que provoca que el microcontrolador se interrumpa, el programa de interrupción manda a leer los registros correspondientes y actualiza los valores de las variables, al finalizar, el control de la máquina igualmente envía un pulso (figura 5.5, señal P2) que es recibido por el pin de interrupción externa del control de temperatura, por lo que nuevamente se ejecuta el programa

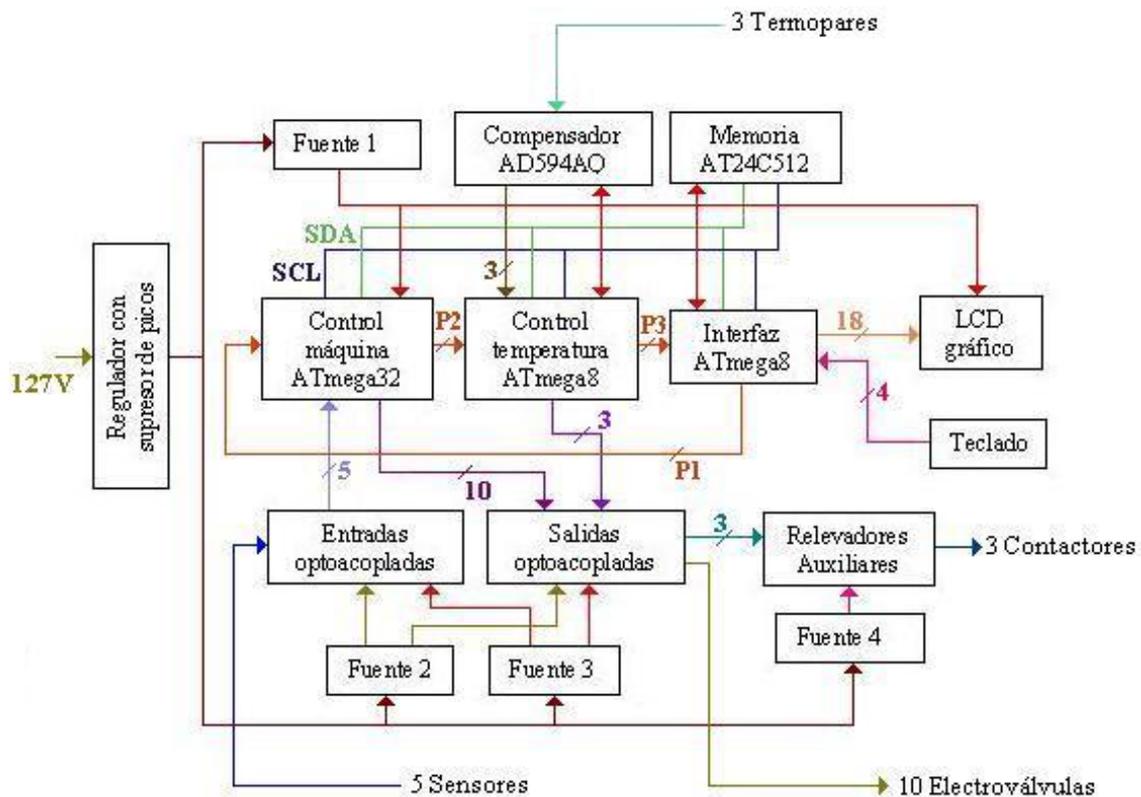


Figura 5.5: Diagrama a bloques

de interrupción que manda a leer los registros de las variables del control de temperatura para actualizar su valores; al terminar la subrutina de interrupción del control de temperatura envía un pulso (señal P3) que es recibido por una entrada del microcontrolador de la interfaz, ya que el programa está esperando un pulso, que corresponde al enviado por el control de temperatura. La interfaz, al recibir el pulso reconoce que los módulos han actualizado sus valores y está lista para realizar otra operación, por lo que despliega nuevamente el menú principal.

La idea principal es que la interfaz sobrescriba los registros de las variables, una vez que ha terminado, avise al control de la máquina que han cambiado los valores de la memoria, entonces el control lee sus registros para actualizarlos. Al terminar avisa al control de temperatura para que haga lo mismo con sus valores y finalmente el control de temperatura avisa a la interfaz que ambos se han actualizado. El objetivo del protocolo es principalmente evitar las colisiones de datos, permitir una lectura ordenada de la memoria y leer la memoria sólo cuando ha sufrido un cambio.

Ya teniendo todos los elementos y su forma de comunicación, el siguiente paso en el diseño fue probarlo en una tarjeta protoboard. Una vez comprobado el funcionamiento del sistema, el siguiente paso fue diseñar los PCB, estos pueden ser consultados en el apéndice C. Los primeros PCB que se diseñaron fueron la fuente 3 de 5V (figura 5.6) para alimentar las bobinas de los relevadores y la de fuente 2 de 24V (figura 5.7) para los interruptores y electroválvulas.



Figura 5.6: *Fuente 3*



Figura 5.7: *Fuente 2*

Los siguientes PCB fueron diseñados con base en lo descrito en la sección 3.2, dividido en dos partes. La primera parte del PCB (figura 5.8) es la electrónica del control, compuesta por los siguientes elementos:

- Fuente 1 de 5V con filtro LC
- Control de temperatura
- Control de la máquina
- Interfaz de comunicación
- Compensadores
- Memoria
- LCD gráfico
- Teclado

Mientras que la segunda parte del PCB esta compuesto por las entradas (digitales) y salidas del sistema (figura 5.9), ambas partes se comunican por medio de un cable plano.

El último PCB diseñado corresponde a los relevadores auxiliares utilizados para aislar al sistema del ruido de los contactores (figura 5.10)

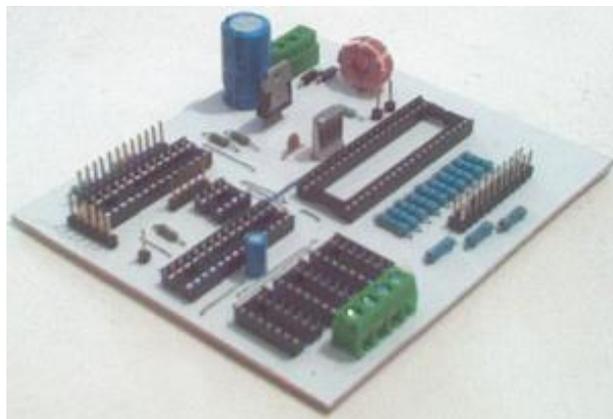


Figura 5.8: *PCB de control*

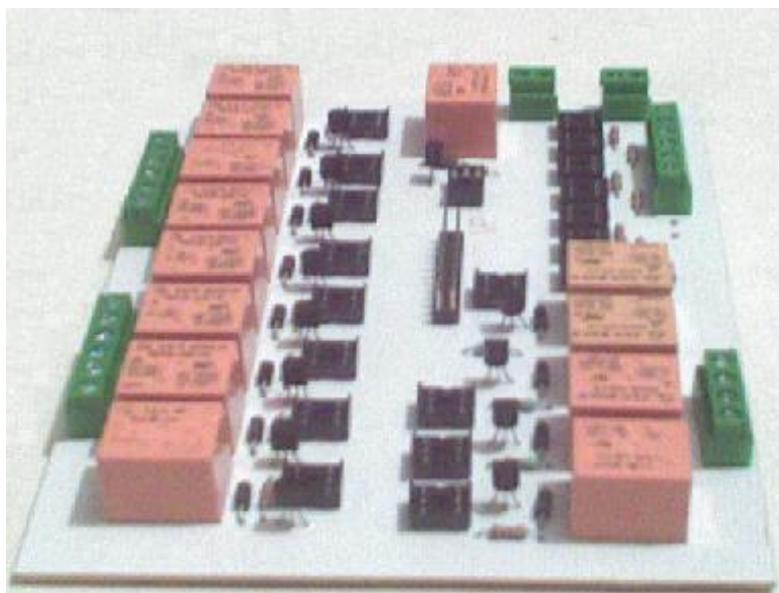


Figura 5.9: *PCB de entradas y salidas*

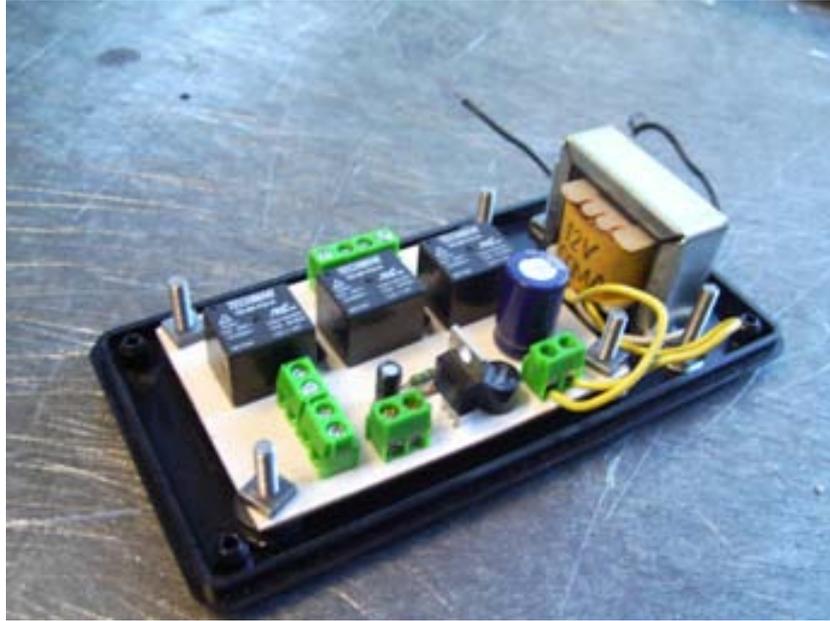


Figura 5.10: *PCB auxiliar*

5.4. Presentación final

Para la presentación final se optó por ubicar las tarjetas en forma horizontal, dentro de un gabinete de plástico de dimensiones 32x21cm.

La figura 5.11 muestra la presentación final, en la parte inferior izquierda de la misma imagen hay otro gabinete de plástico de dimensiones 15x7.5cm, dentro de él se colocó el PCB con los relevadores auxiliares, la razón de no incluirlo en el gabinete principal es evitar que el ruido de los contactores no altere al resto de la circuitería, dado que la tensión que maneja es elevada respecto a las demás salidas.

Por último hay que mencionar que todas las entradas y salidas del PCB de la figura 5.11 fueron conectadas a una tira con conectores molex y terminales tornillos para facilitar la instalación del diseño.

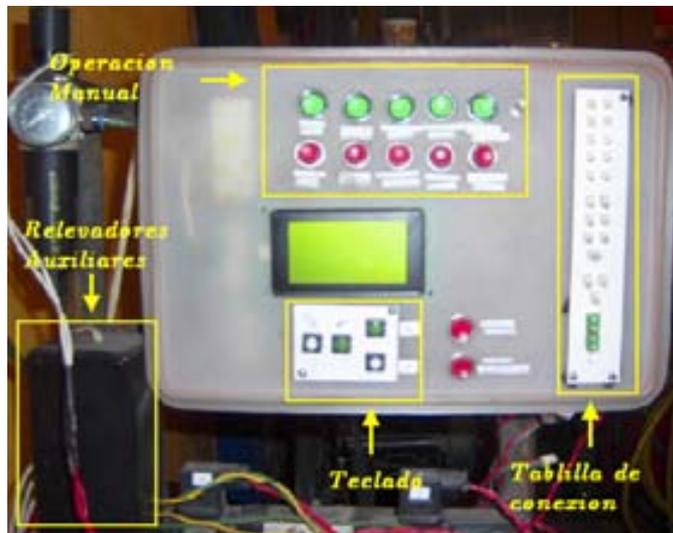


Figura 5.11: *Diseño final*

Capítulo 6

Operación del control

El propósito de este capítulo es dar al lector información básica sobre la conexión del control. Dado que surgen imprevistos como cambiar alguna pieza neumática, lo cual muchas veces obliga a desconectar sensores o electroválvulas; por lo tanto este capítulo ilustra su conexión, también da información de la modificación de los parámetros del control y manipulación de la máquina mediante su interfaz de comunicación.

6.1. Conexión del control

Esta sección sirve de referencia para la instalación del control, tanto en la parte electrónica como la neumática, se inicia con las conexiones neumáticas y finalmente la electrónica. Antes de iniciar la conexión del circuito neumático hay que definir el código de números de las diferentes vías para facilitar su localización, ésta es la siguiente:

Presión de Alimentación	1
Escape	3, 5, 7, 9
Alimentación hacia el actuador	2, 4, 6, 8
Pilotajes neumáticos	12, 14, 16, 18

Estos números se encuentran al lado de cada vía. Ahora para simplificar nombres, se llamará en este texto a la cámara delantera del actuador CDA y a la cámara trasera del actuador CTA, para hacer una referencia simplificada a estas partes del actuador. Con esto puede iniciarse la conexión de las electroválvulas. Todos los actuadores en su CDA van conectados a una electroválvula en la vía 2, en la vía 4 se conecta la CTA y la vía 1 va conectada directamente a la presión de alimentación, las vías 3 y 5 son escapes por donde desfoga el aire, por lo que se colocaron silenciadores con regulador de velocidad para controlar la velocidad de desplazamiento del actuador, para aclarar la idea la figura 6.1 muestra un ejemplo de conexión.

En cuanto a la electroválvula del soplador la configuración cambia, la vía 2 no tiene conexión solo se coloca un tapón para evitar que el aire se fugue, la vía 4 se conecta directamente al soplador, la vía 1 a la presión de alimentación y las vías 3 y 5 nuevamente se colocan reguladores de velocidad para el desfogue del aire de la pieza, con esto termina lo correspondiente a la conexión neumática. Las electroválvulas son 5/2 biestables, por lo que cuenta con dos bobinas. Para reconocerlas hay que apoyarse en el código de números, la bobina que conecta la vía 1 con

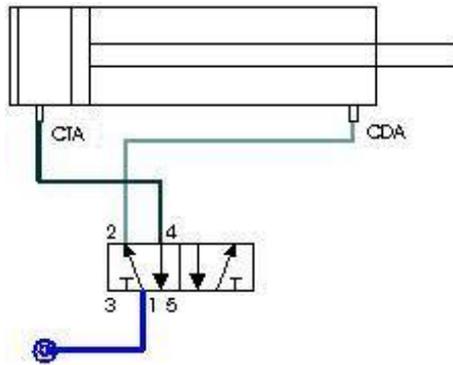


Figura 6.1: *Conexión Neumática*

la 2, se encuentra del mismo lado que la vía 2 e igualmente la bobina que conecta la vía 1 con la 4 se encuentra del mismo lado que la vía 4, así pueden ser identificadas. Para conectarlas al sistema de control, las entradas y salidas fueron conectadas a una tira con conectores molex y terminal tornillo, la tabla 6.1 ilustra la configuración de la tira.

Con esto la conexión electrónica es simple, por la configuración neumática manejada, las bobinas A+, B+, C+ y D+ están del lado de la vía 4 y las A-, B-, C- y D- del lado de la vía 2, solo hay que conectar cada una el lugar indicado por la tabla 6.1, igualmente los sensores y termopares, aunque se conoce la ubicación de cada bobina y de más elementos en la tablilla de conexión por la tabla 6.1, a todos las conexiones se les identifico mediante un número colocado en su cable, para facilitar su conexión, así solo hay que colocar el número correspondiente de cable al correspondiente en la tablilla, para la conexión del PCB auxiliar hay que considerar que tiene un pin de $12 V_{DC}$ que se conecta al común (21) de la terminal tornillo, las otras tres entradas cierran respectivamente a su relevador auxiliar (22 - 24). También hay que tomar en cuenta que el actuador D puede o no estar presente, dado que no siempre se ocupa para llevar el soplador hasta el molde, así que cuando no se requiera debe evitarse su conexión, tanto de la alimentación de presión (vía 1) como de la conexión de sus bobinas (D+, D-) en la tablilla de conexión. Finalmente en la tabla 6.1 pueden observarse dos entradas extras llamadas Aux1 y Aux2, estas entradas se agregaron al diseño final en caso de que alguno de los dos actuadores sin sensor lo requiera. Finalmente al momento de realizar las conexiones siempre hay que considerar que apesar de que los elementos se encuentran identificados, debe verificarse que estos correspondan a los sensores ó bobinas mediante la identificación de la electroválvula por medio de la vía.

N°	ELEMENTO	ELEMENTO	N°
1	Bobina A+	Bobonia A-	2
3	Bobina C+	Bobina C-	4
5	Bobina B+	Bobina B-	6
7	Bobina D-	Bobina D-	8
9	Bobina abrir aire	Bobina cerrar aire	10
11	Sensor A-	Sensor A+	12
13	Sensor B+	Sensor B-	14
15	Aux1	Aux2	16
17	Botón		
18	Termopar Zona 1	Termopar Zona 2	19
20	Termopar Zona 3		
21	Común		
22	Relevador Zona 1		
23	Relevador Zona 2		
24	Relevador Zona 3		

Tabla 6.1 Molex del control

6.2. Operación de la Interfaz

La finalidad de la interfaz es variar los parámetros de los controles, por lo que esta sección está dedicada a su operación. Las variables del control de la máquina son:

- *Tsoplado* (Tiempo de soplado)
- *Tapertura* (Tiempo de apertura)
- *Activoc* (Activación del control de la máquina)

Y las del control de temperatura son:

- *Tzona1* (Temperatura Zona1)
- *Tzona2* (Temperatura Zona 2)
- *Tzona3* (Temperatura Zona 3)
- *Activot* (Activación del control de temperatura)

Los datos se introducen a la interfaz a través del teclado (figura 6.2) cuenta con dos botones verticales (botón aumentar de color verde y botón disminuir de color blanco) que tienen dos funciones: la primera es incrementar o reducir el valor uno por uno si son pulsados instantáneamente o la segunda función, si se mantiene el botón presionado por dos segundos, comienza

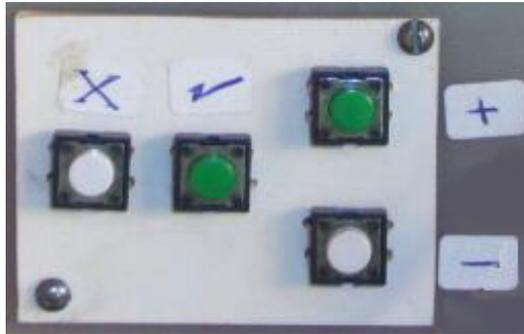


Figura 6.2: Teclado



Figura 6.3: Menú principal

a incrementarse o decrementarse el valor hasta que el botón sea soltado. Cuando la variable llega a su valor máximo permitido, el siguiente número será su valor mínimo permitido, igualmente cuando la variable se reduce hasta su valor mínimo, el siguiente valor será el máximo. Los botones colocados horizontalmente son para aceptar (color verde) o rechazar (color blanco) un valor, cuando se presiona el botón cancelar (botón marcado con una equis) regresamos a la pantalla principal, pero si aceptamos (botón marcado con una paloma) los registros de la memoria son actualizados. Una vez que se conoce el funcionamiento del teclado se puede comenzar a navegar por las pantallas de la interfaz.

La pantalla inicial de la interfaz despliega el menú principal, en ella se agrupan por categorías las variables que pueden seleccionarse.

En la figura 6.3 la última etiqueta con nombre opción es donde se va indicar el submenú al cual se quiere ingresar, para ello se usan los botones aumentar (+) ó disminuir (-). Se selecciona el número de acuerdo al submenú a ingresar, cuando es seleccionado se presiona el botón aceptar.

El primer submenú se refiere a los timers (figura 6.4), el nombre se eligió en inglés para facilitar la comprensión de la opción, dado que en la industria el nombre de timer es muy utilizado para referirse a un retardo en el proceso. Dentro del submenú se encuentran las primeras dos variables (tiempo de soplado y tiempo de apertura) seguidos de sus respectivos valores. En la



Figura 6.4: Submenú Timers



Figura 6.5: Elección del Tiempo

parte de abajo se encuentra nuevamente la etiqueta opción, donde se elige la variable a modificar.

Una vez seleccionada la variable, aparece enseguida valor actual (figura 6.5), entonces se procede a ingresar su nuevo valor con el teclado (+, -), recordando que las unidades son segundos. Cuando se coloca el valor requerido aceptamos o cancelamos el valor para regresar a la pantalla principal. El valor máximo que puede introducirse es 150 segundos y el mínimo de 1 segundo.

El segundo submenú se refiere a la temperatura de las diferentes zonas de la máquina (figura 6.6), por lo que dentro de él se despliegan las temperaturas de referencia en grados centígrados.

El procedimiento es el mismo, con el número de opción se elige la zona, enseguida aparece su valor para ser modificado (figura 6.7), cuando se acepta o rechaza el valor volvemos a la pantalla principal. La temperatura máxima que puede introducirse es $280^{\circ}C$ y el mínimo de $1^{\circ}C$.

El tercer submenú corresponde a la activación de los controles (figura 6.8), dentro de él se despliegan las variables *activoc* y *activot*, seguidos de su estado actual.

Por medio de la opción, se elige la variable, a continuación aparece su estado (figura 6.9), donde ON es para prender y OFF para apagar, al aceptar o rechazar la operación se vuelve a la pantalla inicial.



Figura 6.6: *Submenú Temperatura*



Figura 6.7: *Elección de la Temperatura*

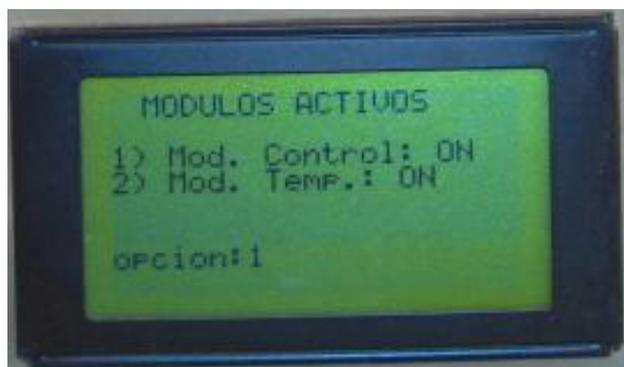


Figura 6.8: *Activación de Controles*



Figura 6.9: Elección del Estado

Hay que mencionar que la interfaz tiene programado un *timeout*, lo que significa que si no se entra a algún opción del submenú seleccionado o no se completó el proceso de actualización en un lapso de 11 minutos, la interfaz regresa a la pantalla inicial sin modificar ningún valor.

6.3. Manipulación manual

En la sección 3.2 se mencionó la posibilidad de controlar cada uno de los pasos que ejecuta la máquina de forma independiente. La forma de poder hacerlo es a través del juego de botones colocado en la parte superior del sistema (figura 6.10).

Los botones se encuentran etiquetados con la acción que realiza cada uno, en la esquina superior derecha de la figura 6.10 puede observarse un interruptor, con él se hace el cambio de modo de operación, cuando esta posicionado hacia abajo el control es automático, pero cuando el interruptor es posicionado hacia arriba, se cambia a manual, por lo que cada actuador se puede mover independientemente, sólo basta presionar el botón pertinente.

Debe tenerse en cuenta que el modo manual está hecho sólo para ajustes mecánicos, por lo que cuando se activa quitamos la alimentación a los relevadores de los contactores para evitar que se gaste energía. Por eso, una vez terminados los ajustes debe regresarse el interruptor a modo automático, si no aunque en la interfaz se muestren activos ambos controles, ninguno de ellos funcionará.



Figura 6.10: *Manipulación Manual*

Capítulo 7

Resultados y Conclusiones

Los resultados obtenidos en este proyecto fueron satisfactorios al cumplirse con los siguientes objetivos propuestos:

1. Construcción de un control seguro y confiable para la máquina de soplado de plástico.
2. Construcción de un control de temperatura que sustituye a los pirómetros y es manejado de forma digital.
3. Reducción de espacio al ocupar el sistema una dimensión de 32x21cm.
4. Fácil instalación.
5. Reducción de costos en la elaboración del proyecto.
6. No requiere de grandes conocimientos para su operación, dado que sólo se requiere oprimir un botón para iniciar el proceso.

Por la amplia variedad de productos que se elaboran en la máquina, fue complicado tratar de hacer un control estándar para todos los casos, no obstante, únicamente cuando el molde es soplado por arriba puede eliminarse el botón que inicia el proceso, por lo que se deja como propuesta la colocación de una fotocelda al final del molde para que detecte la llegada del parison e iniciar el proceso de forma automática. De esta forma el operador sólo tendría que recoger la pieza y ya no preocuparse por accionar el botón, dado que la fotocelda se encargaría de mandar la señal al control para arrancar el proceso.

Otro punto importante en el desarrollo de este proyecto aparte del problema del ruido eléctrico, fue buscar el lugar donde colocar las tarjetas diseñadas, pues una mala elección puede ocasionar ruido sobre los circuitos. Lo más importante que debe considerarse al colocarlas es evitar que la tierra física de la red eléctrica tenga contacto con el negativo de nuestras fuentes diseñadas, dado que la tierra puede introducir ruido o dañar las tarjetas de forma permanente.

Finalmente, para concluir este trabajo, sólo resta mencionar que los sistemas desarrollados con base en microcontroladores son una buena opción para reducir costos y espacio, sin embargo, este tipo de diseño sólo es recomendable cuando el tiempo de diseño no cuesta o no es tan caro, pues como pudo verse a lo largo de este trabajo, el ruido que puede estar presente depende de la carga que vaya accionar así como también el lugar donde vaya a ser colocado, por lo que si

no se puede perder tiempo en analizar estas situaciones o no se tiene la experiencia necesaria, lo más recomendable es comprar un PLC para evitar problemas.

Apéndice A

Mantenimiento

El aire que hay en el ambiente contiene impurezas y humedad que son almacenadas en el compresor, por lo que antes de poder utilizarlo en nuestro sistema debe pasar por la unidad de mantenimiento que se encargará de eliminar todo lo anterior. Las unidades de mantenimiento están compuestas por:

1. *La filtración* del aire para eliminar impurezas sólidas o líquidas.
2. *La regulación* de la presión de alimentación de los elementos neumáticos.
3. *La lubricación* del aire, para mantener lubricadas válvulas y cilindros al momento de alimentarlos.

Con esto puede asegurarse mayor durabilidad y fiabilidad de los elementos neumáticos. Para el circuito neumático instalado en este trabajo, su unidad de mantenimiento sólo cuenta con los primeros dos puntos, pues para las electroválvulas y cilindros instalados no es necesaria una lubricación, ya que están diseñados para tener una vida útil de aproximadamente diez años. La unidad de mantenimiento (figura A.1) esta provista de un recipiente transparente colocado en la parte de abajo donde se almacena toda la humedad de la compresora. También tiene un manómetro (medidor de presión) y una válvula en la parte superior para regular la presión de alimentación del sistema, que en este caso está alrededor de los 6 bar.

En su interior se encuentra el filtro, el cual debe cambiarse por lo regular cada seis meses o quizás en menos tiempo. Todo está en función de la humedad del ambiente de trabajo, así como el uso constante que se le dé. Por último hay que recordar purgar (limpiar impurezas) a diario el recipiente transparente, para impedir que éste se llene y pueda introducirse el líquido al sistema.



Figura A.1: *Unidad de mantenimiento*

Apéndice B

Diagramas de Flujo

Diagrama de flujo (figura B.1) para colocar la máquina de soplado de plástico en condiciones iniciales. Diagrama de flujo del proceso de la producción (figura B.2). En la figura B.3 se muestra el diagrama de flujo del control de temperatura para la zona 1, los demás son exactamente igual a diferencia de la zona y referencia que controlan.

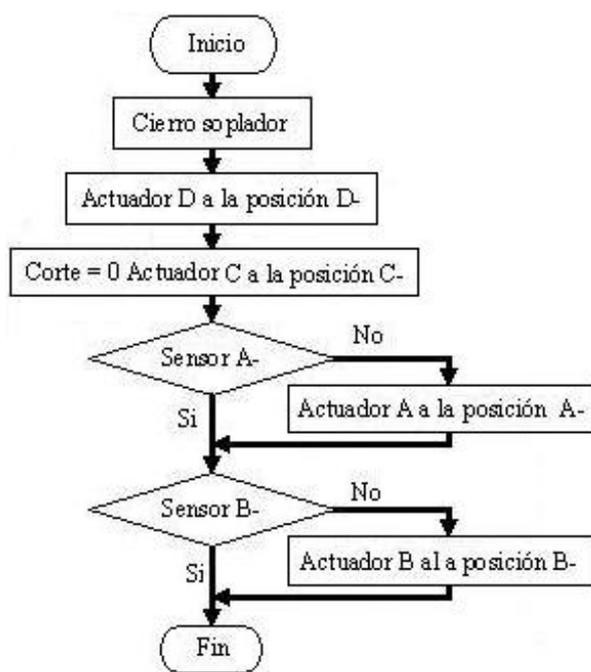


Figura B.1: *Condiciones Iniciales*

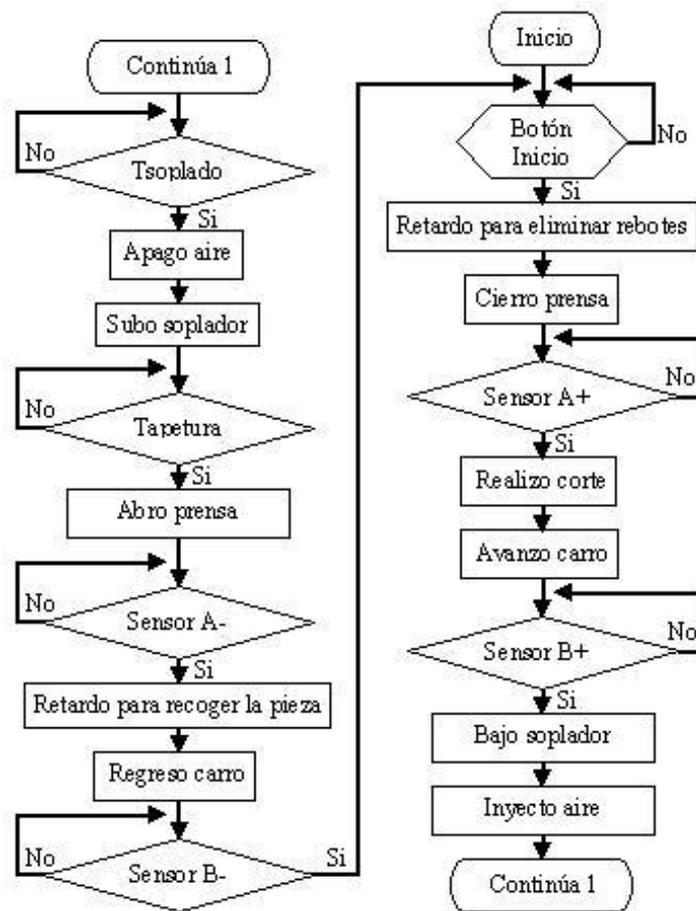


Figura B.2: *Secuencia de la máquina*

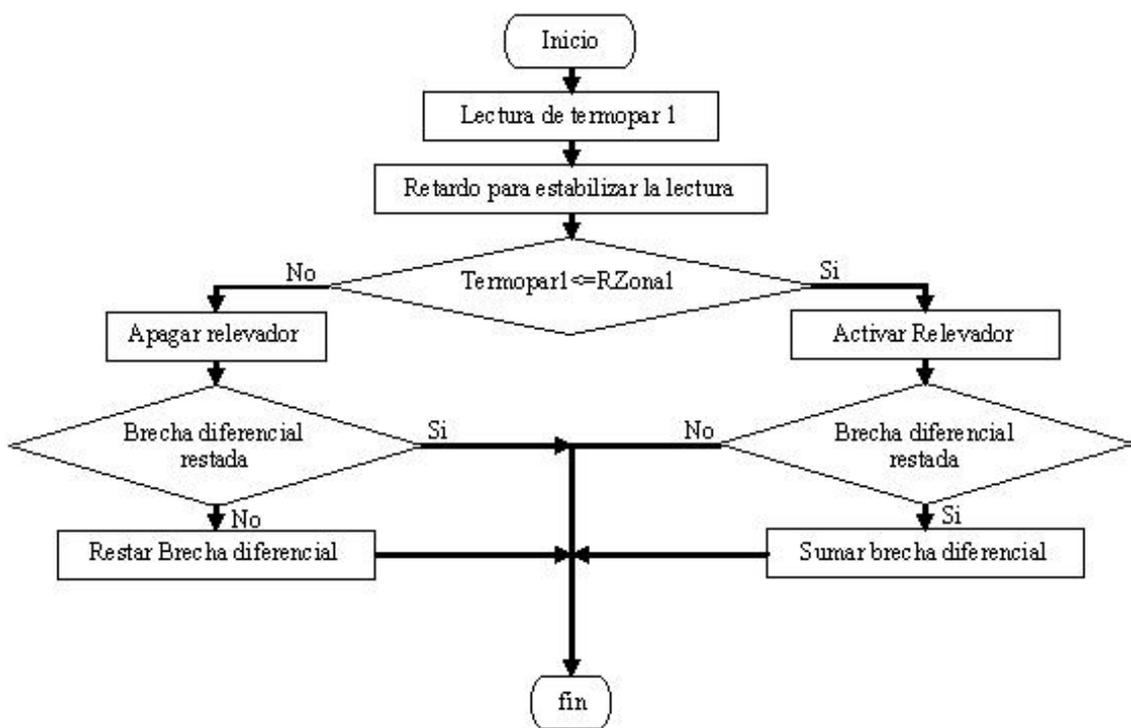


Figura B.3: *Control de Temperatura*

Apéndice C

Diagramas de circuito

Los diagramas tan sólo son demostrativos no están a escala de diseño

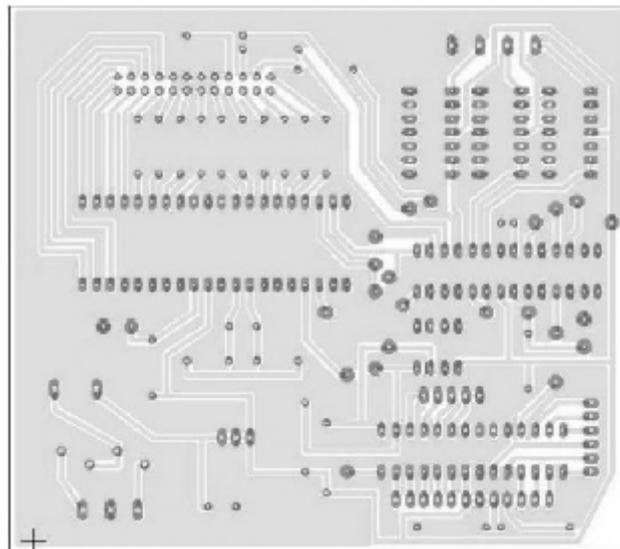


Figura C.1: *PCB del control*

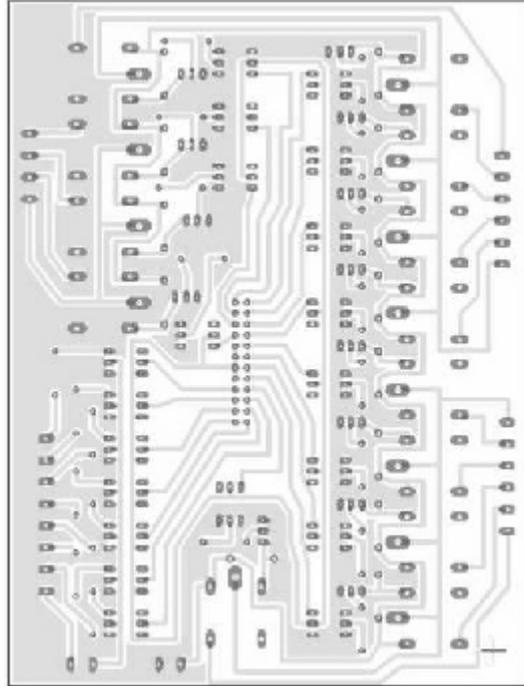


Figura C.2: *PCB entradas y salidas*

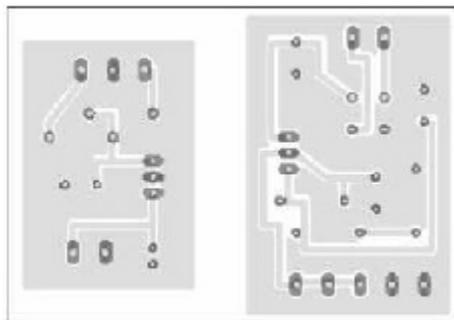


Figura C.3: *PCB fuentes de alimentación*

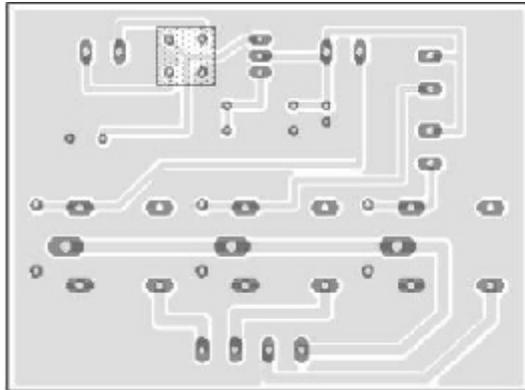


Figura C.4: *PCB relevadores auxiliares*

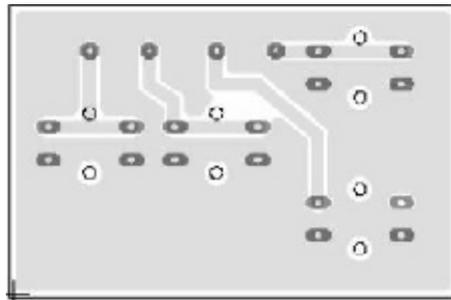


Figura C.5: *PCB del teclado*

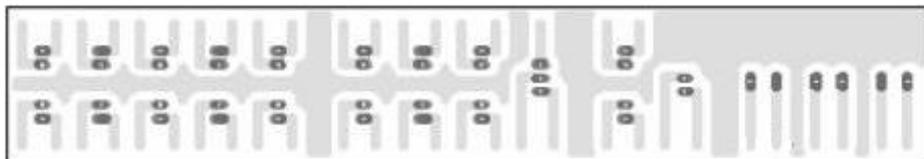


Figura C.6: *PCB molex*

Apéndice D

Tablas y gráficas

ANSI Code	Alloy Combination	Maximum Temperature Range	mV Output
B	Platinum/Rhodium	0°C to +1700°C	0 to +12.426
E	Chromel/Constantan	-200°C to +900°C	-8.824 to +68.783
J	Iron/Constantan	0°C to +750°C	0 to +42.283
K	Chromel/Alumel	-200°C to +1250°C	-5.973 to +50.633
N	Nicrosil/Nisil	-270°C to +1300°C	-4.345 to +47.502
R	Platinum/Rhodium Platinum	0°C to +1450°C	0 to +16.741
S	Platinum/Rhodium Platinum	0°C to +1450°C	0 to +14.973
T	Copper/Constantan	-200°C to +350°C	-5.602 to +17.816

Tabla D.1 Características de los termopares

Thermocouple Temperature °C	Type J Voltage mV	AD594 Output mV	Type K Voltage mV	AD595 Output mV
-200	-7.990	-1525	-5.891	-1454
-180	7.402	-1428	5.550	-1370
-160	-6.821	-1316	-5.141	-1269
-140	-6.150	-1188	-4.666	-1152
-120	-5.426	-1046	-4.138	-1021
-100	-4.632	-895	-3.553	-876
-80	3.785	-729	2.920	-719
-60	2.892	-556	2.243	-552
-40	-1.960	-376	-1.527	-375
-20	-0.995	-189	-0.777	-180
-10	-0.501	-94	-0.392	-94
0	0	3.1	0	2.7
10	.507	101	.397	101
20	1.019	200	.798	200
25	1.277	250	1.000	250
30	1.536	300	1.203	300
40	2.058	401	1.611	401
50	2.585	505	2.022	503
60	3.115	606	2.436	605
80	4.186	813	3.266	810
100	5.268	1022	4.095	1015
120	6.350	1233	4.916	1219
140	7.457	1445	5.733	1420
160	8.560	1659	6.539	1620
180	9.667	1875	7.338	1817
200	10.777	2087	8.137	2015
220	11.887	2302	8.938	2213
240	12.998	2517	9.745	2413
260	14.108	2732	10.560	2614
280	15.217	2946	11.381	2817
300	16.325	3160	12.207	3022
320	17.432	3374	13.039	3227
340	18.537	3588	13.874	3434
360	19.640	3801	14.712	3641
380	20.743	4015	15.552	3849

Tabla D.2 Voltaje de salida AD594 y AD595 contra la temperatura

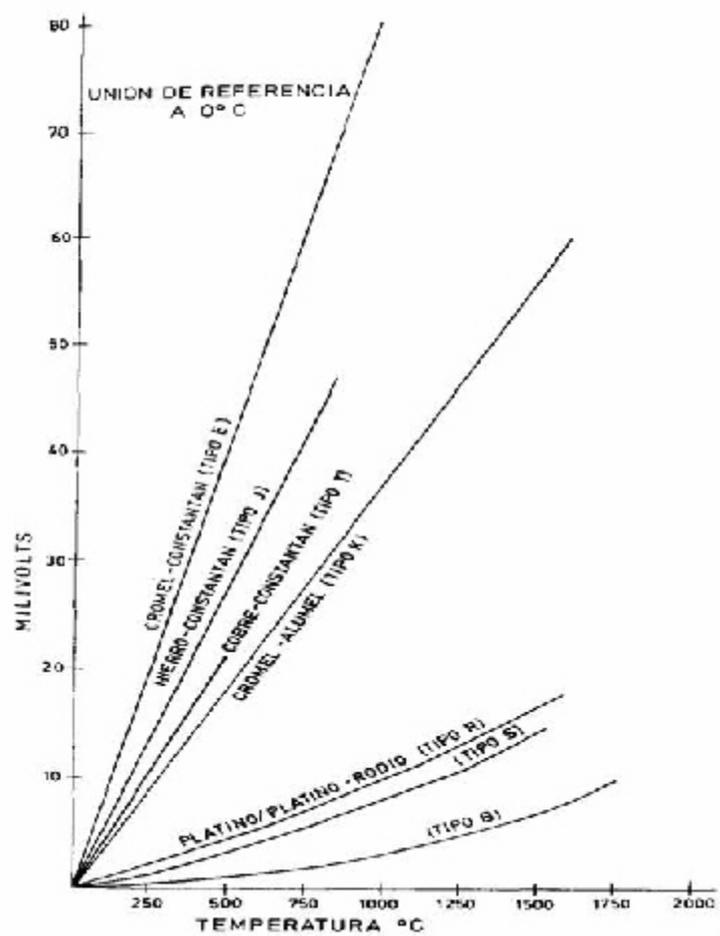


Figura D.1: Gráfica de respuesta de los termopares

Bibliografía

- ATMEL, Datasheet AT24C512, Two-Wire Serial EEPROM 512Kbytes
http://www.atmel.com/dyn/resources/prod_documents/doc1116.pdf
- ATMEL, Datasheet ATmega8, 8-bit AVR Microcontroller with 8KBytes In-System Programmable Flash http://www.atmel.com/dyn/resources/prod_documents/doc2486.pdf
- ATMEL, Datasheet ATmega32, 8-bit AVR Microcontroller with 32KBytes In-System Programmable Flash http://www.atmel.com/dyn/resources/prod_documents/doc2503.pdf
- Boylestad Robert L. y Nashelsky Louis, *Electrónica: Teoría de circuitos y Dispositivos electrónicos*, Ed. Pearson Educación, 2003.
- Datasheet JHD12864A, LCD gráfico <http://www.agspecinfo.com/pdfs/J/JHD12864.PDF>
- Di Lella Daniel, “Ruidos en los sistemas con microcontroladores”¿Cómo controlarlos?, Comentario Técnico, Dto Técnico Electrocomponentes S.A.
http://www.ingenieria-inversa.com.ar/uploads/files/BC908_3.pdf
http://www.ingenieria-inversa.com.ar/uploads/files/BC908_4.pdf
http://www.ingenieria-inversa.com.ar/uploads/files/BC908_5.pdf
http://www.ingenieria-inversa.com.ar/uploads/files/BC908_6.pdf
http://www.ingenieria-inversa.com.ar/uploads/files/BC908_7.pdf
http://www.ingenieria-inversa.com.ar/uploads/files/BC908_8.pdf
- Floyd Thomas L., *Dispositivos Electrónicos*. Ed. LIMUSA, 2004.
- Gottfried Byron, *Programación en C*, Ed Mc. Graw Hill, 1997.
- Marcin Joe, ANALOG DEVICES, AD594/AD595 Application Note, Thermocouple Signal Conditioning Using the AD594/595
http://www.analog.com/static/imported-files/application_notes/34661261AN369.pdf#xml=http://search.analog.com/search/pdfPainter.aspx?url=http://www.analog.com/static/imported-files/application_notes/34661261AN369.pdf&fterm=ad594&fterm=application&fterm=note&fterm=ad594_application_note&la=en

- Ogata Katsuhiko, Ingeniería de control moderna, Ed. Prentice-Hall Inc.1970.
- Procyon, Librerías para microcontroladores AVR de ATMEL y documentación de las mismas. <http://www.mil.ufl.edu/~chrisarnold/components/microcontrollerBoard/AVR/avrlib/>
- Solórzano Palomares J. Fernando, *Introducción a la programación estructurada y al lenguaje C*, Tomo II, Ed. Facultad de ingeniería,1995.