



REPÚBLICA DEL ECUADOR

Escuela Politécnica Nacional

" E SCIENTIA HOMINIS SALUS "

La versión digital de esta tesis está protegida por la Ley de Derechos de Autor del Ecuador.

Los derechos de autor han sido entregados a la "ESCUELA POLITÉCNICA NACIONAL" bajo el libre consentimiento del (los) autor(es).

Al consultar esta tesis deberá acatar con las disposiciones de la Ley y las siguientes condiciones de uso:

- Cualquier uso que haga de estos documentos o imágenes deben ser sólo para efectos de investigación o estudio académico, y usted no puede ponerlos a disposición de otra persona.
- Usted deberá reconocer el derecho del autor a ser identificado y citado como el autor de esta tesis.
- No se podrá obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

El Libre Acceso a la información, promueve el reconocimiento de la originalidad de las ideas de los demás, respetando las normas de presentación y de citación de autores con el fin de no incurrir en actos ilegítimos de copiar y hacer pasar como propias las creaciones de terceras personas.

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA ELÉCTRICA Y ELECTRÓNICA

DISEÑO Y CONSTRUCCIÓN DE UNA IMPRESORA BRAILLE

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO EN ELECTRÓNICA Y CONTROL

JULIO CÉSAR CAMINO IPIALES

jkmino@hotmail.com

LILIANA MARIELA LIGÑA PILLAJO

lilylignia@yahoo.es

DIRECTOR: NELSON SOTOMAYOR, MSc.

nelson.sotomayor@epn.edu.ec

Quito, Agosto 2011

DECLARACIÓN

Nosotros, Julio César Camino Ipiales y Liliana Mariela Ligña Pillajo, declaramos bajo juramento que el trabajo aquí descrito es de nuestra autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que hemos consultado las referencias bibliográficas que se incluyen en éste documento.

A través de la presente declaración cedemos nuestros derechos de propiedad intelectual correspondientes a éste trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Julio César Camino Ipiales

Liliana Mariela Ligña Pillajo

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por Liliana Mariela Ligña Pillajo y Julio César Camino Ipiales, bajo mi supervisión.

NELSON SOTOMAYOR, MSc.

DIRECTOR DEL PROYECTO

AGRADECIMIENTO

A mi amadísimo Dios por brindarme la gran oportunidad de vivir, por su infinito amor y comprensión por ser un padre excelente, cálido, misericordioso y por sobre todo fiel, por brindarme sus dones y la habilidad suficiente como para llevar a cabo todos los proyectos en mi vida.

A mis amados y queridos padres, Julio y Mery, y a mi amada hermana Gaby por aguantar todo a mi lado, por impulsarme a seguir pese a las dificultades, por nunca abandonar los sueños e ilusiones de los seres que amamos, por estar ahí prestos a brindarme un hombro cálido, un abrazo sincero, un consejo oportuno, gracias papi, mami y ñaña por absolutamente todo, que mi Dios se los pague.

A mis tías Rosita y Teresita por ser grandes auténticas y bondadosas, que Dios las prospere y bendiga.

A mi amado abuelito por tu impulso, tu forma de ser y tu valioso legado, que mi Dios te siga bendiciendo aunque ahora las bendiciones te llegan mucho más rápido, siempre te amaré papá Cesítar.

A mis amadas tías Zoily y Tere, por su buen corazón y ayuda desinteresada a lo largo de las épocas difíciles en mi vida familiar, a mis queridos y amados primos; Dani, Xavi y Pati por ser mis hermanos mayores con todo lo que esto implica.

A mi amada hija Mary ya que tú eres el mayor de mis aciertos, sin tu inocencia mi vida y éste proyecto estarían incompletos.

A Sra. Sory quien a lo largo de la carrera siempre se mostró diáfana para ayudar y brindar un consejo oportuno.

A Luchito Campoverde, ya que gracias a tu apoyo éste proyecto llegó a feliz término, gracias por ser un amigo incondicional.

A todas las personas que me brindaron una puerta abierta, ayuda diáfana, y un corazón cálido, que Dios las prospere y bendiga.

Julio César Camino Ipiales

AGRADECIMIENTO

Agradezco a Dios por permitirme tener vida y concluir una etapa mas de la vida estudiantil reflejada en éste proyecto, por levantarme todos los días con la ilusión de seguir adelante y vencer las dificultades que se presentan día a día.

A la virgencita por escuchar mis rezos cuando he necesitado de su ayuda.

A mi mamita Susy por todo el apoyo brindado, por su paciencia, por escucharme y aconsejarme en toda situación, gracias por ser mi amiga y mi madre,

A mi hermano Wily y su esposa Angie por abrirme las puertas de su casa y por las malas noches para que éste proyecto pueda llegar a su fin, gracias por ser parte de mi familia.

Al Ing. Nelson Sotomayor por aceptar dirigir éste proyecto.

A mi tío Emilio y a su familia por el cariño y por su ayuda oportuna.

A mis amigas Belén R, Lore M, Katy A, Pao E, Day V, Gaby S, Taty P, Doris J, Dany P , Eli V , Vero G , Moni H, Eve G, Mary I, Jesy B, Mica S, Paty P y Laury A por escucharme las mismas historias con mucha paciencia, por aconsejarme muy bien por formar parte de mi vida y permitir que yo forme parte de su vida también.

A mis amigos Willian R, Cris LI, Diego G, Wilo S, Beno B, Wagner P, Santy D, Micky C, Franz V, Cris O, Xavi A y Orlando C con quienes compartí mucho de mi vida estudiantil, gracias por estar ahí en los altos y bajos de la vida.

A la Señora Sory quien a lo largo de la carrera ha encontrado la manera de ayudarnos y las palabras adecuadas para aconsejarnos.

Un agradecimiento especial a Lore M, Julio P, Omar T, Belén R y Andrés B por que sin su apoyo no hubiera podido concluir éste sueño, gracias por el interés en que termine el proyecto y concluya una etapa mas de mi vida.

Liliana Ligña P

DEDICATORIA

Éste trabajo está dedicado a mi familia en especial a mi padre y madre por sus grandes sacrificios a lo largo de su vida, por perseguir y apoyar los sueños de los seres que aman.

A mi hermana Gaby, por ser más que familia en la sangre ya que compartimos la salvación de nuestro papi Dios.

A mi amada hija Mary, a su madre Cecy y su familia por aceptar sin reproches y con mucha paciencia todo lo que significa perseguir un sueño.

A Omar Trujillo por creer en un sueño conjunto.

A la familia de Liliana, por su invaluable apoyo y ayuda en todo momento.

Julio César Camino Ipiales

DEDICATORIA

Éste trabajo dedico a mi mamita Susy por todo el amor y el apoyo que siempre me ha dado, porque ella se esforzó mucho para que tuviera la oportunidad de educarme.

A mi Andrés quien siempre ha estado en mi corazón y me ha ayudado a vencer las dificultades pese a ya no estar con vida.

A mi hermano Wily quien ha estado conmigo en todo momento y ha hecho posible que éste proyecto tenga un fin.

A mi nena Lendy porque ha sido mucho tiempo que no he podido verla.

A Angie Hoyos por su paciencia y su apoyo cuando mas lo he necesitado.

A Omar Trujillo por creer en nosotros y en el proyecto.

A Carlos A. Falconí por ser un gran recuerdo y no haber cambiado en estos quince años.

A Xavi Gómez por enseñarme a tener paciencia.

Liliana Ligña P

CONTENIDO

RESUMEN	III
PRESENTACIÓN	IV
CAPÍTULO 1	1
FUNDAMENTOS BÁSICOS	1
1.1 ANTECEDENTES HISTÓRICOS DE LA IMPRESIÓN BRAILLE	2
1.2 LA CELDA O MATRIZ BRAILLE	3
1.2.1 DIMENSIONES DE LA CELDA BRAILLE	4
1.2.2 SEPARACIÓN ENTRE CARACTERES [3]	5
1.2.3 TAMAÑO DE LOS CARACTERES BRAILLE [3]	6
1.4 SIGNOGRAFÍA, ALFABETO Y NUMERACIÓN BRAILLE	6
1.4.1 NÚMEROS CARDINALES	6
1.4.2 NÚMEROS ORDINALES [3]	8
1.4.3 NÚMEROS ROMANOS	10
1.4.4 EL ALFABETO BRAILLE [3]	11
1.4.5 SIGNOS ESPECIALES [3]	13
1.5 LECTURA BRAILLE	14
1.5.1 LOS SIGNOS ESPECÍFICOS DEL ESPAÑOL	14
1.6 IMPRESORAS BRAILLE COMERCIALES	15
1.6.1 MODELO ROMEO Y ROMEO PRO [4]	15
1.6.2 IMPRESORA VIEWPLUS CUB [4]	16
1.6.3 GEMINI [4]	18
CAPÍTULO 2	20
DISEÑO Y ENSAMBLAJE DEL PROTOTIPO	20
2.1 DISEÑO MECÁNICO	20
2.1.1 ADECUACION Y REEMPLAZO DE COMPONENTES	22
2.2 DISEÑO ELECTRÓNICO	31
2.2.1 MÓDULO MOTOR A PASOS	31
2.2.2 MÓDULO MOTOR DC	39
2.2.3 MÓDULO PUNZÓN	46
2.2.4 MÓDULO REPRODUCTOR DE AUDIO	54
2.2.5 MÓDULO MICROCONTROLADOR	56
2.2.6 FUENTE DE ALIMENTACIÓN	68
CAPITULO 3	72
DESARROLLO DEL PROGRAMA DE CONTROL	72
3.1 DESCRIPCIÓN GENERAL	72
3.2 PROGRAMACIÓN EN EL MICROCONTROLADOR	73
3.2.1 REPRODUCCION DE VOZ	74
3.2.2 COMUNICACIÓN USB	76
3.2.3 POSICIONAMIENTO DE LA HOJA	81
3.2.4 IMPRESIÓN MECÁNICA	84

3.3 PROGRAMACIÓN DE LA INTERFAZ DE USUARIO	86
3.3.1 PANTALLA	87
3.3.2 PROGRAMACIÓN INTERNA	89
CAPITULO 4	95
PRUEBAS Y RESULTADOS	95
4.1 PRUEBAS MECANICAS	95
4.1.1 PUNZÓN	95
4.1.2 BANDEJA DE APOYO DE IMPRESIÓN	100
4.1.3 POSICIONAMIENTO DEL PAPEL ENTRE LA BANDEJA Y EL RODILLO	101
4.2 PRUEBAS DE SOFTWARE	101
4.2.1 MICROCONTROLADOR	101
4.3 PRUEBAS MEDIANTE SOFTWARE DE CONTROL	105
4.3.1 COMUNICACIÓN USB	105
4.3.2 RESPUESTA DEL PROTOTIPO AL ENVIO DE CARACTERES	107
4.4 COSTO DEL PROYECTO	119
CAPITULO 5	123
CONCLUSIONES Y RECOMENDACIONES	123
5.1 CONCLUSIONES	123
5.2 RECOMENDACIONES	126
REFERENCIAS BIBLIOGRÁFICAS	129

RESUMEN

El presente proyecto expone el diseño y ensamblaje de una Impresora Braille que tiene por objeto a través de un punzón plasmar en el papel los puntos que conforman los caracteres en Braille que son enviados desde la PC.

Las subrutinas que son parte de los programas de control se han desarrollado de tal manera que primero se detecte que el Módulo Impresora Braille se encuentre conectado, una vez realizado éste proceso se abre el documento y se procede al envío de caracteres de control para asegurar que la comunicación se ha realizado correctamente; se emiten mensajes de voz y de texto en el módulo para que el usuario tenga conocimiento de que se está ejecutando correctamente las diferentes subrutinas.

Al ejecutarse el programa residente en la computadora se despliega una pantalla donde se da la opción de explorar el documento a ser impreso, al abrir él mismo se despliega en una ventana y posteriormente su equivalencia en Braille en la ventana contigua.

Se procede entonces al envío de datos, los mismos que se almacenan en el microcontrolador previamente a ser impresos.

PRESENTACIÓN

En el Capítulo Uno “Fundamentos Básicos” se presenta una introducción de cómo se desarrolló el Código Braille a lo largo de la Historia, sus principales normas y características. Aquí se indican las equivalencias de la escritura convencional en el código Braille.

En el Capítulo Dos “Diseño y Ensamblaje del Prototipo” se expone todo el desarrollo en la adecuación de la parte mecánica de la impresora, así como el diseño del punzón que permite plasmar en el papel los puntos que forman los caracteres Braille.

En el Capítulo Tres “Desarrollo del Programa de Control” se expone la programación general y se explican las subrutinas con los diagramas de flujo de cada una de éstas, se detalla cómo el programa desarrollado en la PC consigue enviar los caracteres que son parte del documento que se imprimen en el Módulo “Impresora Braille” a través del puerto USB y cómo éste a su vez interpreta la información y la plasma en el papel.

En el Capítulo Cuatro “Pruebas y Resultados” se presentan las tareas que se realizaron para tener un resultado adecuado acorde al alcance del proyecto, se expone además las soluciones que se dieron a los inconvenientes presentados a lo largo del desarrollo de éste prototipo. En éste capítulo además se presenta un cuadro donde se indica el listado de elementos y materiales que se utilizaron así como el costo de los mismos.

En el Capítulo Cinco “Conclusiones y Recomendaciones” se puede verificar el trabajo realizado para tener el prototipo final ya que aquí se presentan algunas sugerencias después de la experiencia a lo largo del desarrollo del presente proyecto y se destacan aspectos importantes que fueron cruciales a la hora de tomar decisiones para conseguir los resultados esperados.

CAPÍTULO 1

FUNDAMENTOS BÁSICOS

El perpetuo avance de la tecnología parece relegar a ciertos sectores de la población, entre los cuales se encuentran las personas con capacidades especiales específicamente las que padecen deficiencia visual, tomando en cuenta su lucha incansable por igualdad de derechos y oportunidades se han desarrollado varias propuestas que mejoran su calidad de vida, una de éstas se podría encasillar en la creación del alfabeto o código Braille, un conjunto de caracteres en alto relieve que por medios táctiles les ayudan a identificar, escribir, expresarse y crear.

Desde el nacimiento del código Braille, las primeras matrices en alto relieve y por supuesto los punzones, se han desarrollado varias formas de confeccionar de una manera más automatizada un texto legible que pueda articular en código Braille las diversas ideas que por su limitación estas personas no podrían escribir.

Las máquinas de escribir mecánicas en código Braille fueron las precursoras de varios formatos de prototipos electro mecánicos que literalmente no vieron la luz por su alto costo, poca aplicación y efectividad, debido en gran parte al apareamiento de nuevas tecnologías como las computadoras personales, las impresoras Braille tienen gran acogida en la actualidad. Debido al alto costo que poseen estos modelos comerciales únicamente están destinados para brindar apoyo a instituciones gubernamentales que a su vez ofrecen sus servicios a un limitado grupo del gran conglomerado que componen las personas con deficiencia visual [1].

Las impresoras Braille son máquinas altamente útiles e invaluable si se considera que gracias a ellas es posible un avance mucho más rápido en el aprendizaje y preservación de ideas que las personas con deficiencia visual pueden expresar, sin embargo al ser un artículo altamente costoso limita en gran medida su desarrollo integral.

1.1 ANTECEDENTES HISTÓRICOS DE LA IMPRESIÓN BRAILLE

A mediados del siglo XIII, Al-Imam Al-Amadi, un vendedor de libros ciego en Arabia fue el primero en utilizar caracteres táctiles para la lectura, desarrolló caracteres en relieve, los colocaba sobre cada libro, esto le permitía reconocer su título y precio [1].

En 1786 en París, Valentin Haüy, un filántropo francés fundó el primer colegio para ciegos, él sentía que la ceguera no debía asociarse obligatoriamente a la ignorancia y la dependencia de los demás. Haüy también desarrolló, el primer método de impresión de libros para ciegos éste consistía en presionar una cartulina mojada sobre caracteres de gran tamaño hechos de plomo por lo que los libros eran demasiado grandes y pesaban alrededor de nueve kilos. La lectura era muy lenta, debían recorrer con la yema del dedo cada caracter hasta reconocerlo, hasta aquí era posible la lectura pero no la escritura [2].

Fue Charles Barbier en 1808 quien propuso la utilización de puntos en relieve para la escritura de textos, la llamó "escritura nocturna". En 1819, su sueño era desarrollar un sistema para los no videntes, el sistema se basaba en la utilización de una matriz de 6 líneas y 6 columnas para representar las letras del alfabeto y ciertos sonidos habituales. Por éste motivo, su sistema se ganó el apelativo de "sonográfico" de Barbier. En 1821 Charles Barbier inventó un sistema de letras de puntos, código alfabético basado en grupos de puntos. Louis Braille adaptó el sistema de Barbier utilizando grupos de 1 a 6 puntos. La conferencia de Londres de educadores de invidentes de 1932, estableció el modelo para los sistemas de puntos, adoptando un alfabeto conocido como Standard English Braille. Éste es idéntico al Braille original con pequeñas modificaciones. Éste sistema se usa en todo el mundo angloparlante y ha sido adaptado a la mayoría de las lenguas, incluido el chino. Braille también diseñó un sistema de caracteres de puntos para la escritura musical.

Entre los primeros instrumentos que se diseñaron para esta escritura destaca la pizarra o pizarrón de surcos, que consiste en una hoja metálica o de madera labrada en forma de pequeños cuadrados. El papel se colocaba sobre la pizarra, y el lápiz, guiado por los surcos, marcaba un relieve sobre el papel, obteniéndose

una caligrafía manual cuadrangular que se podía ver y tocar. En la actualidad se utilizan máquinas de escribir adaptadas para comunicarse con las personas invidentes, y pizarras para guiar al escritor en la formación de caracteres Braille. Además se dispone de la máquina de escribir Hall Braille, una modificación de la máquina de escribir normal, que crea caracteres Braille en el papel en lugar de imprimir en él caracteres visuales.

Los pacientes invidentes y sordos, con o sin capacidad de hablar, deben ser tratados de manera individualizada. Su rehabilitación requiere profesores especiales y siempre depende de la comprensión personal y de la paciencia.

1.2 LA CELDA O MATRIZ BRAILLE

Los caracteres Braille se fundamentan en una matriz de 6 puntos, distribuidos en 2 columnas y 3 filas como se observa en la Figura 1.1.



Figura 1.1 Signo Generador del Braille

Cada punto que forma el signo generador se encuentra numerado según su posición, como puede observarse en la Figura 1.2 esto ayuda a identificar cada punto.



Figura 1.2 Signo Generador con los Puntos Numéricos

La combinación de estos 6 puntos permiten obtener los distintos caracteres en braille en total 64 combinaciones incluido el espacio en blanco, las 63 combinaciones de puntos que pueden ser realizadas con el *signo generador* son los signos Braille.

Algunos signos Braille pueden contener más de un caracter, de éste modo se forman las letras del alfabeto, los números, los signos de puntuación, etc. [3].

1.2.1 DIMENSIONES DE LA CELDA BRAILLE

Las medidas de la celda Braille puede observarse en la Figura 1.3.

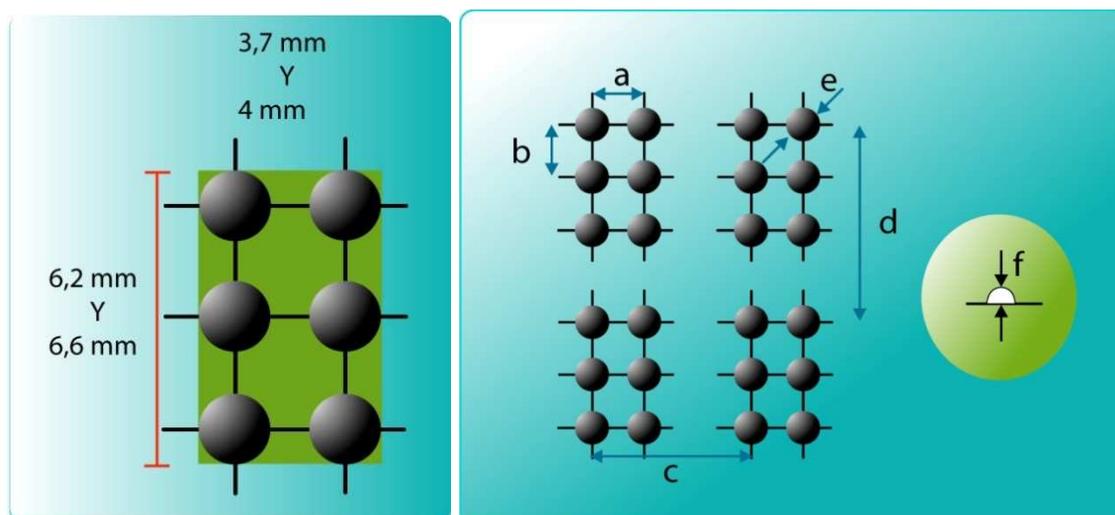


Figura 1.3 Dimensiones de la celda Braille, tomado de [3]

a = Distancia horizontal entre los centros de puntos contiguos de la misma celda de 2.5 a 2.6 mm.

b = Distancia vertical entre los centros de puntos contiguos de la misma celda de 2.5 a 2.6 mm.

c = Distancia entre los centros de puntos en idéntica posición de celdas contiguas de 6.0 a 6.1 mm.

d = Distancia entre centros de puntos en idéntica posición de líneas contiguas de 10.0 a 10.8 mm.

e = Diámetro de la base de los puntos: entre 1.2 y 1.5 mm.

f = Altura del relieve de los puntos: de 0.5 a 0.65 mm.

El espacio en blanco que separa palabras debe ser considerado como un carácter Braille y, por tanto, reproducir las dimensiones del mismo aunque no contenga ningún punto en relieve.

A continuación se recogen diferentes ejemplos de rótulos considerados correctos (en color amarillo) e incorrectos (en color gris).

En éste ejemplo no se ha dejado espacio entre las palabras “sala” y “de”, el cual debe contemplarse aunque el texto ocupe más de un renglón.

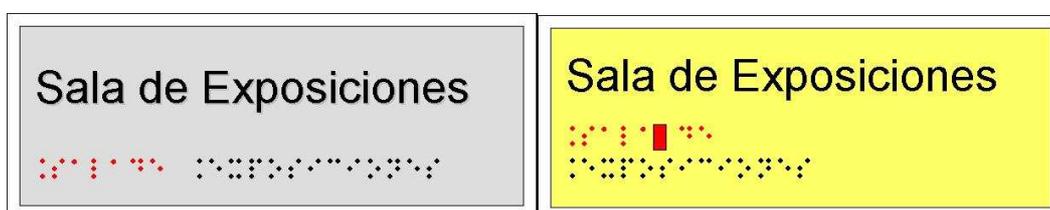
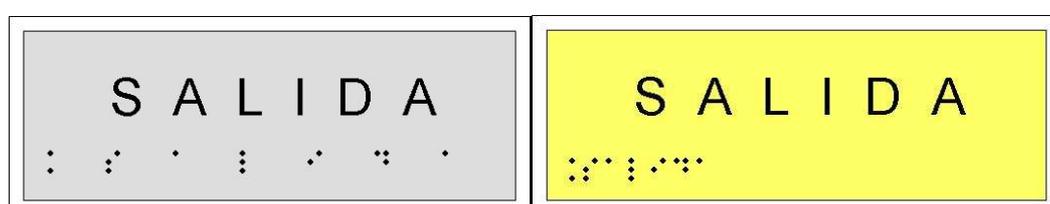


Figura 1.4 Ejemplo de escritura en Braille

1.2.2 SEPARACIÓN ENTRE CARACTERES [3]

La separación entre caracteres ha de ser respetada según las medidas de los parámetros dimensionales en la Figura 1.3.

En el ejemplo siguiente, al escribir en Braille se ha utilizado, de modo incorrecto, el mismo recurso que a la hora de diseñar los caracteres gráficos: se han separado los caracteres para ocupar todo el espacio de la señal. Es un efecto visual de diseño que en Braille no tiene ningún sentido: el lector braille las consideraría palabras formadas por un solo carácter. Además, al haber una separación mayor entre carácter y carácter que la de un espacio en blanco Braille, se corre el riesgo de que la lectura se detenga al leer el primer carácter creyendo que no existe más texto, pues el dedo no encuentra ningún carácter braille a la distancia habitual de separación entre palabras.



(a) Incorrecto (b) Correcto

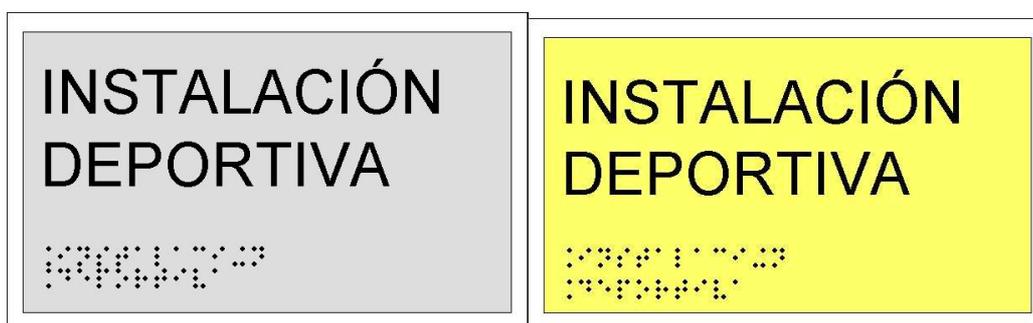
Figura 1.5 Ejemplo de la separación de caracteres en la escritura Braille

1.2.3 TAMAÑO DE LOS CARACTERES BRAILLE [3]

Un caracter Braille tiene el tamaño aproximado de la yema del dedo ya que si éste es más pequeño que el referido en los parámetros dimensionales, no se percibe la separación entre puntos y caracteres y resulta imperceptible, como una “mancha” en relieve, al contrario si el tamaño es demasiado grande la yema del dedo no abarca el caracter y también se tienen problemas para su comprensión.

1.3 DISTANCIA ENTRE RENGLONES [3]

Es importante respetar la distancia entre renglones, de lo contrario pueden confundirse los caracteres del renglón superior con los del inferior.



(a) Incorrecto (b) Correcto

Figura 1.6 Ejemplo de la separación entre renglones en escritura Braille

1.4 SIGNOGRAFÍA, ALFABETO Y NUMERACIÓN BRAILLE

1.4.1 NÚMEROS CARDINALES

Los números se escriben según la siguiente norma:

Al igual que ocurre con las mayúsculas, para escribir los números se utiliza un signo delante de las primeras diez letras.

➤ **Signo de número**

Éste signo está formado por los puntos 3, 4, 5 y 6.

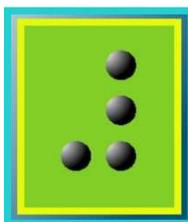


Figura 1.7 Signo de número

De éste modo, los diez dígitos se muestran a continuación en la Tabla 1.1.

Tabla 1.1 Números en Braille del 0 al 9, tomado de [3]

a	•	1	•••	f	••	6	••••
b	••	2	••••	g	•••	7	•••••
c	•••	3	•••••	h	••••	8	••••••
d	••••	4	••••••	i	•••••	9	•••••••
e	•••••	5	•••••••	j	••••••	0	••••••••

El signo de número se escribe una sola vez delante del primer dígito que forme el número, y no delante de cada una de sus cifras como en el siguiente ejemplo.



Figura 1.8 Ejemplos de escritura para números en Braille

1.4.2 NÚMEROS ORDINALES [3]

Los números ordinales se escriben siguiendo las normas que el Braille tiene para tal fin: signo de número (puntos 3, 4, 5 y 6), número en posición baja (combinaciones de puntos 2, 3, 5 y 6) y el calificativo masculino del ordinal (puntos 1,3 y 5 para la “o”).

La escritura de los números ordinales tiene un tratamiento distinto a la escritura de los números cardinales, se utiliza la misma combinación de puntos que conforma cada dígito pero bajando una fila su posición natural, dejando siempre libre la primera fila del signo generador formada por los puntos 1 y 4. En Braille esto se denomina *números en posición baja*. Se escriben de acuerdo a la Tabla 1.2.

Como se puede ver en la Tabla 1.2, cuando el número tiene más de una cifra, todas se escriben en *posición baja*.

A continuación del *número en posición baja* se escribe la letra correspondiente al calificativo del ordinal, es decir “o”.

Los 10 primeros ordinales se presentan en la Tabla 1.3.

Tabla 1.2 Números en posición baja, tomado de [3]

	NÚMERO CARDINAL	NÚMERO EN POSICIÓN BAJA
1		
2		
3		
4		
5		
6		
7		
8		
9		
10		

Tabla 1.3 Números ordinales, tomado de [3]

1°	
2°	
3°	
4°	
5°	
6°	
7°	
8°	
9°	
10°	

Aunque en caracteres visuales aparezca el preceptivo punto entre el número y el calificativo (por ejemplo: 10.^o), en Braille no se escribirá nunca ese punto.

En la Figura 1.9 se puede observar un ejemplo de escritura Braille para un número ordinal

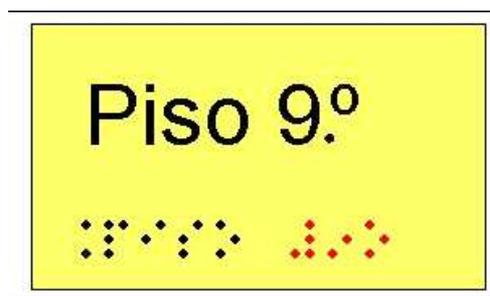


Figura 1.9 Ejemplo de números ordinales en Braille

1.4.3 NÚMEROS ROMANOS

Los números romanos se escriben con los signos Braille que correspondan a las letras que contengan, precedidos por un solo signo de mayúscula. Es decir, se tratarán como si fuera una palabra escrita en mayúsculas, como se puede apreciar en la Tabla 1.4 [3].

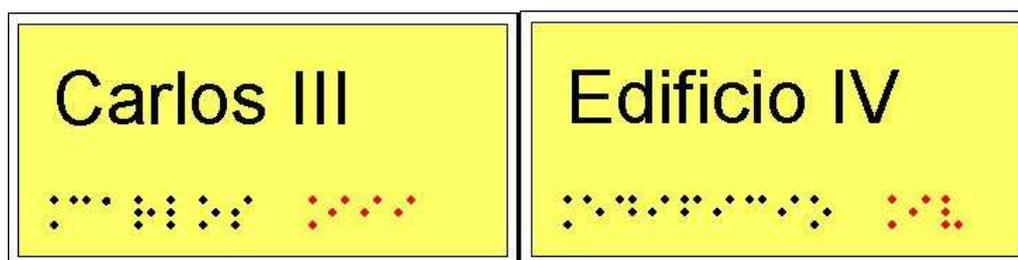


Figura 1.10 Ejemplos de números romanos en Braille

Tabla 1.4 Números romanos, tomado de [3]

I	
V	
X	
C	
L	
D	
M	
XIV	
XXIII	

1.4.4 EL ALFABETO BRAILLE [3]

El sistema o código Braille no es un idioma en sí, con el Braille pueden representarse las letras, los signos de puntuación, los números, la grafía científica, los símbolos matemáticos, la música, etc.

Existen signografías Braille para representar taquigrafía (generado con una máquina que marca los puntos sobre una cinta de papel) y para representar notaciones matemáticas, también llamado Código Matemático Unificado, y musicales.

Con la introducción de la informática, el Braille se amplió a un código de ocho puntos, de tal manera que una letra individual puede ser codificada con una sola celda, pudiendo representar una celda cualquier carácter ASCII. Las 256

combinaciones posibles de los ocho puntos están codificadas según el estándar Unicode.

El Braille puede ser reproducido usando una plancha y un punzón, de forma que cada punto sea generado desde el dorso de la página, escrito en una imagen a la inversa (como la que se obtiene al mirar por un espejo), hecho a mano o impreso con una máquina de escribir Braille, por una impresora Braille conectada a una computadora, o mediante un dispositivo Braille.

A continuación se presenta el alfabeto Braille y los signos de puntuación.

Tabla 1.5 Alfabeto Braille, tomado de [3]

a	b	c	d	e	f	g	h	i	j
⠁	⠃	⠉	⠋	⠑	⠖	⠗	⠈	⠎	⠊
1	12	14	145	15	124	1245	125	24	246
k	l	m	n	o	p	q	r	s	t
⠅	⠇	⠍	⠏	⠕	⠞	⠟	⠞	⠚	⠞
13	123	134	1345	135	1234	12345	1235	234	2345
u	v	w	x	y	z	ALFABETO BRAILLE 			
⠥	⠦	⠪	⠨	⠣	⠞				
136	1236	2456	1346	13456	1356				

Tabla 1.6 Signos de Puntuación, tomado de [3]

PUNTO ORTOGRÁFICO PUNTO DE MILLAR PUNTO DE AGREVIATURA	COMA	PUNTO Y COMA	DOS PUNTOS
.	,	;	:
• 3	• 2	•• 23	••• 25
ABRIR Y CERRAR INTERROGACIÓN	ABRIR Y CERRAR ADMIRACIÓN	APOSTROFO	ABRIR Y CERRAR COMILLAS
¿ ?	¡ !	`	" "
•• 26	••• 235	• 3	••• 236

1.4.5 SIGNOS ESPECIALES [3]

En Braille hay signos especiales que no tienen su correspondencia en caracteres visuales. Estos son el “signo de mayúscula” y el “signo de número”.

➤ **SIGNO DE MAYÚSCULA**

Está formado por los puntos 4 y 6.

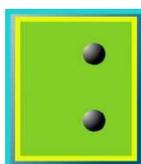


Figura 1.11 Signo de Mayúscula

Éste signo se escribe delante de cada letra para indicar que está escrita en mayúscula. Ejemplo: la letra “a mayúscula” se escribiría como se muestra en la Figura 1.12.



Figura 1.12 Ejemplo de de una letra mayúscula en Braille

El primer caracter Braille corresponde al “signo de mayúscula” y el segundo a la letra “a”.

1.5 LECTURA BRAILLE

La lectura en Braille a través del tacto se realiza por medio de las yemas de los dedos, ya que es una de las zonas más sensibles del cuerpo humano para discriminar estímulos táctiles.

El sentido de la escritura Braille es el mismo que el que se sigue en la lectura visual: de izquierda a derecha.

1.5.1 LOS SIGNOS ESPECÍFICOS DEL ESPAÑOL

El texto debe estar escrito con los signos Braille específicos en español. Las vocales acentuadas y las letras específicas de cada lengua tienen signos especiales que no están recogidos en el alfabeto común, como se muestra a continuación.

La letra *eñe* (**ñ**).

La “u” con diéresis (**ü**).

Las vocales acentuadas (**á, é, í, ó, ú**).

Estos signos en Braille se muestran en la Tabla 1.7.

Tabla 1.7 Signos específicos del español, tomado de [3]

LETRA	SIGNO BRAILLE	PUNTOS
ñ		12456
ü		1256
á		12356
é		2346
í		34
ó		346
ú		23456

1.6 IMPRESORAS BRAILLE COMERCIALES

El costo de una impresora Braille fluctúa entre 1800 y 5000 dólares americanos.

A continuación se presentan algunos ejemplos de modelos comerciales y sus principales características.

1.6.1 MODELO ROMEO Y ROMEO PRO [4]

La impresora “Romeo” posee un potente sistema mecánico de relieve con la ventaja de ser portátil, el modelo “Romeo PRO” a más de las características señaladas también está dotado de un sistema de voz traducido a varios lenguajes, lo que lo hace altamente amigable con cualquier tipo de usuario, se puede apreciar un equipo estándar en la Figura 1.13.



Figura 1.13 Impresora Romeo, tomado de [4]

1.6.1.1 Características

Esta impresora posee una velocidad promedio de 15 caracteres por segundo, tiene la posibilidad de escribir una línea con 32 caracteres Braille, acepta una alimentación de papel continuo y los caracteres Braille pueden ser matrices de 6 puntos u 8 puntos con lo que se pueden crear nuevas combinaciones y más caracteres Braille a futuro.

Su conectividad se realiza mediante el puerto USB (Universal Serial Bus), aunque la misma puede ser serie o paralelo.

Sus dimensiones permiten un transporte fácil, la versión PRO posee la capacidad de emitir órdenes o información en forma audible y en varios idiomas además de permitir varios formatos de hoja. El software que la acompaña posee un traductor de texto incorporado además de los drivers necesarios para la conectividad con cualquier computadora personal, sin embargo su fabricante sugiere un traductor de texto a Braille como Duxbury Braille Translator para “mejorar” el sistema de traducción a las diferentes normas implementadas en cada país.

1.6.2 IMPRESORA VIEWPLUS CUB [4]

Esta impresora posee una alta resolución en cuanto al relieve que producen sus percutores, solamente funciona bajo entorno Windows, tiene la capacidad de generar a través del software proporcionado por el fabricante, imágenes táctiles,

en alto relieve de fotografías y figuras con la limitante que las mismas no deben ser de alta resolución, posee un peso adecuado para ser considerado portable y según su fabricante por las características de ruido emitido esta impresora no necesita de una caja aislante de ruido, básicamente está concebida para aplicaciones en el hogar, un modelo presentado por el fabricante como estándar se puede apreciar en la Figura 1.14.



Figura 1.14 Impresora VIEWPLUS, tomado de [4]

1.6.2.1 Características

Puede generar una velocidad máxima de 50 caracteres por segundo, y tiene la posibilidad de aceptar papel continuo y una amplia variedad de tamaños de papel con una diversidad amplia de gramaje de los mismos gracias a que posee la capacidad de regular la fuerza y el relieve de los puntos para generar el caracter Braille, el módulo tiene un peso aproximado de 20 a 28 lb, tiene una potente herramienta en programación que la hace capaz de imprimir en relieve figuras o fotografías de baja resolución.

La interfaz de conexión es USB, con la posibilidad de conectarlo en red (conector RJ45), el fabricante recomienda un software especializado de traducción como el TSS Braille, el software de gráficos lo provee la casa fabricante, además también es compatible con Duxbury Braille Translator.

Éste modelo también es capaz de realizar una impresión en Braille a doble cara, la emisión de ruido es baja por lo que no se recomienda el uso de cajas aislantes de sonido.

1.6.3 GEMINI [4]

Tiene la capacidad de producir impresiones con el sistema Braille con matrices de 6 puntos, fue creada para hacerla accesible, rápida y silenciosa pues solo emite 55 dB de ruido, además es una de las más rápidas en el mercado pues según su fabricante puede fácilmente superar la velocidad de otros modelos inclusive en 2 o 4 veces.



Figura 1.15 Impresora GEMINI, tomado de [4]

1.6.3.1 Características

Tiene una velocidad de impresión de 45 caracteres por segundo, posee un ablandador en base a tinta, que cuando es escogido por el usuario, aletarga la velocidad a 32 caracteres por segundo, Puede imprimir en una línea de 32 caracteres.

La conexión con la computadora personal puede ser por medio serial, paralelo o USB.

Para el ensamblaje y construcción de un prototipo que se pueda designar como “Impresora Braille” se toman en cuenta varios aspectos que constan como características relevantes entre los módulos antes citados que se deducen “populares” o “más adquiridos” entre las organizaciones de ayuda para personas no videntes, básicamente se podrían reconocer; el tipo o interfaz de

comunicación, en nuestro caso USB 2.0, volumétricamente transportable y amigable para varios tipos de usuario pues se podrían presentar casos en los cuales el usuario final podría ser una persona con discapacidad visual o no, pese a que los documentos generados por éste prototipo están orientados a éste sector de personas con capacidades especiales.

Se busca realizar el prototipo de manera tal que sea económicamente asequible para la mayor parte de la población que sufre discapacidad visual, y sobre todo con elementos electrónicos y mecánicos presentes en el mercado local.

Para esto se adapta la parte mecánica de una impresora, reemplazando los segmentos electrónicos de la misma para que sean adecuados a los propósitos del presente proyecto.

También se desarrolla una interfaz de usuario en la cual se facilita la búsqueda, apertura y tratamiento para el posterior envío de los datos existentes en un archivo de texto, estos datos serán exteriorizados a través del puerto de comunicación USB para posteriormente realizar el relieve en una hoja cumpliendo con las medidas del estándar conocido como caracter Braille.

CAPÍTULO 2

DISEÑO Y ENSAMBLAJE DEL PROTOTIPO

En éste capítulo se aborda las diferentes modificaciones que se realizan a una impresora para utilizar de ésta sus componentes mecánicos, así como el ensamble de un módulo que producirá o ejecutará un relieve o punción para plasmar un caracter Braille que sea aceptablemente legible.

A continuación se presenta el diseño de las diferentes etapas electrónicas para comandar los módulos que componen el prototipo.

2.1 DISEÑO MECÁNICO

Para éste diseño se ha utilizado como base la parte mecánica de una impresora HP-95C, el motor que mueve al rodillo de avance de papel es un motor a pasos unipolar de serie PM55L-048 [7], el motor que mueve el carrete en donde están alojados, entre otros elementos, el punzón es un motor DC con serie c2162-6006 [8], en el carrete se encuentra alojado un encoder óptico Q9874 [9], los elementos antes mencionados, conjuntamente con su alojamiento mecánico, moto reductores y carcasa externa no han sufrido mayor modificación en la construcción del prototipo "Impresora Braille".

En la Figura 2.1 se puede apreciar el diagrama de ensamblaje que tutela éste modelo de impresora, éste diagrama resulta muy útil pues algunos componentes son reemplazados, para realizar esta modificación es necesario un desmontaje total, por dos motivos fundamentales; limpieza general y reemplazo de componentes.

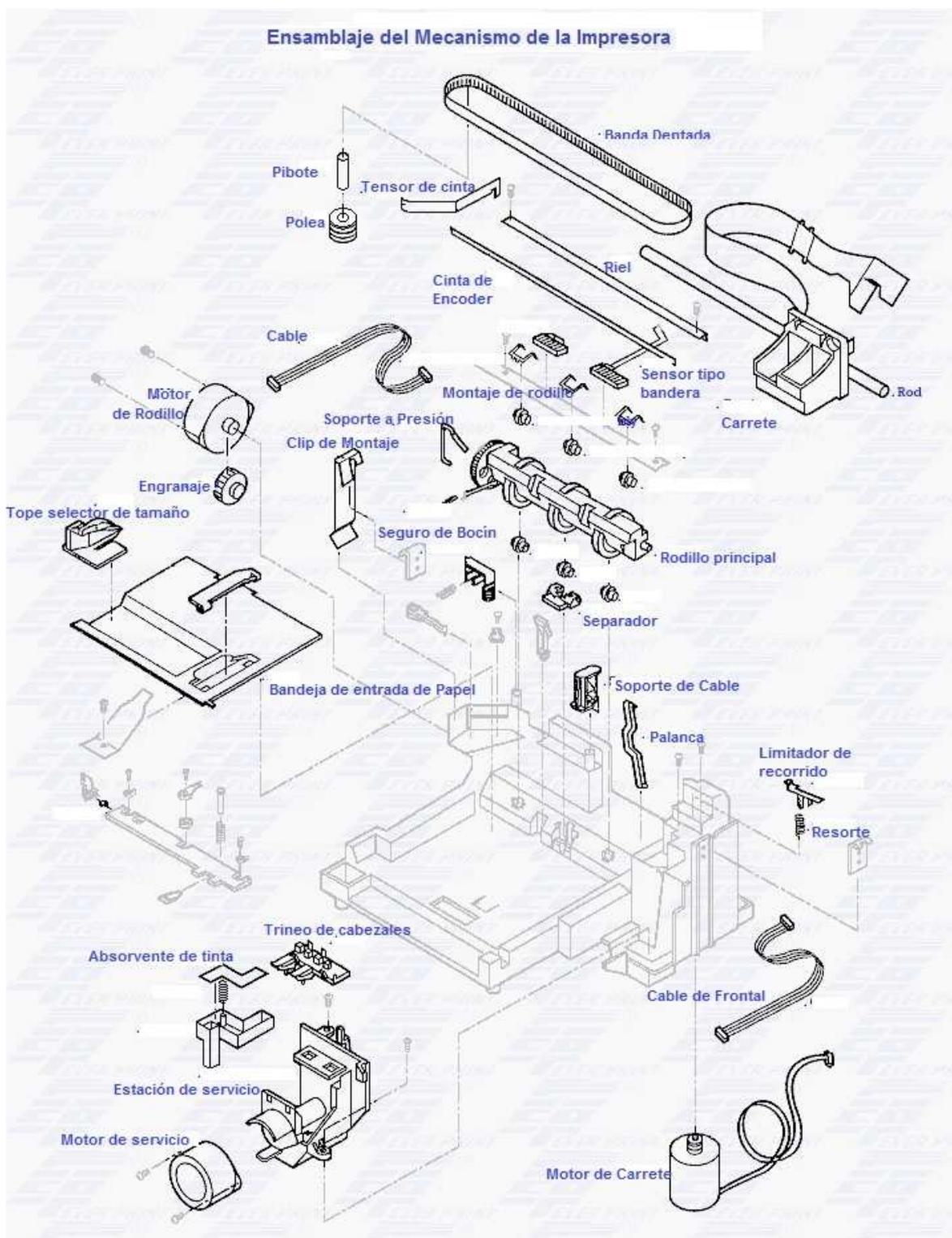


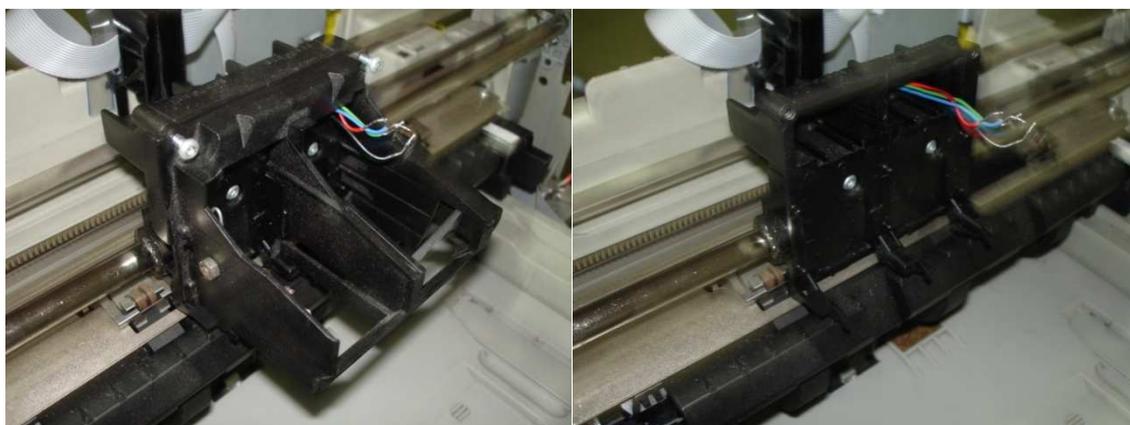
Figura 2.1 Esquema de ensamblaje de la impresora HP95C, tomado de [10]

2.1.1 ADECUACION Y REEMPLAZO DE COMPONENTES

Una vez realizada una limpieza general de la impresora y a su vez verificar que todos los motores y mecanismos involucrados en el desarrollo del presente prototipo se encuentran en buen estado y sobre todo funcionales, se procede a cambiar algunos componentes de la misma.

2.1.1.1 Carrete de alojamiento de cartuchos de tinta

La remoción de éste elemento puede verse en la Figura 2.2, en la parte (a) se observan los sujetadores y en la parte (b) el espacio disponible, cabe anotar que es precisamente éste lugar en donde se colocará el módulo de punzado.



(a)

(b)

Figura 2.2 Carrete de cartuchos de tinta

En la Figura 2.2 (b) se visualiza cuatro cables que se han soldado al encoder óptico presente en éste módulo para detectar y posicionar al carrete de manera exacta, (Figura 2.3).

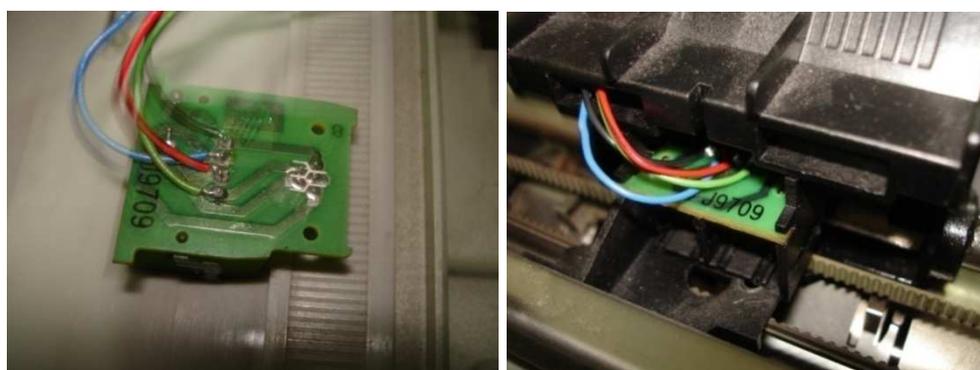


Figura 2.3 Encoder óptico Q9874

2.1.1.2 Base o soporte de impresión

Para realizar una correcta impresión es necesario proveer de un apoyo con base rígida recubierto con un material esponjoso que posibilite el hundimiento y punción que requiere la impresión en Código Braille, por lo que en la base longitudinal de la bandeja de alimentación de papel de la impresora se construye y fija un soporte de aluminio, éste soporte se muestra en la Figura 2.4.

La base de soporte para la impresión se encuentra atornillada a la carcasa interna de la impresora como se puede observar en la Figura 2.5.



Figura 2.4 Base de impresión



Figura 2.5 Base de impresión adaptada en la carcasa de la impresora

Cabe mencionar que el material de color azul es una plancha esponjosa recubierta de una lámina de las mismas dimensiones en acetato, esta combinación de materiales se determina experimentalmente por prueba y error en las diferentes impresiones realizadas para brindar un soporte ideal a los relieves de los caracteres Braille.

2.1.1.3 Punzón

En las siguientes figuras se detalla el proceso de ensamblaje del módulo destinado para la impresión mecánica que forma en relieve un carácter Braille, como primeros pasos se detallan en la Figura 2.6 las partes que componen el

acoplamiento al solenoide del punzón, cabe anotar que el vástago móvil retorna a su posición inicial por medio de un resorte acoplado al mismo, esto se evidencia en la Figura 2.7 que presenta al punzón acoplado en el solenoide.

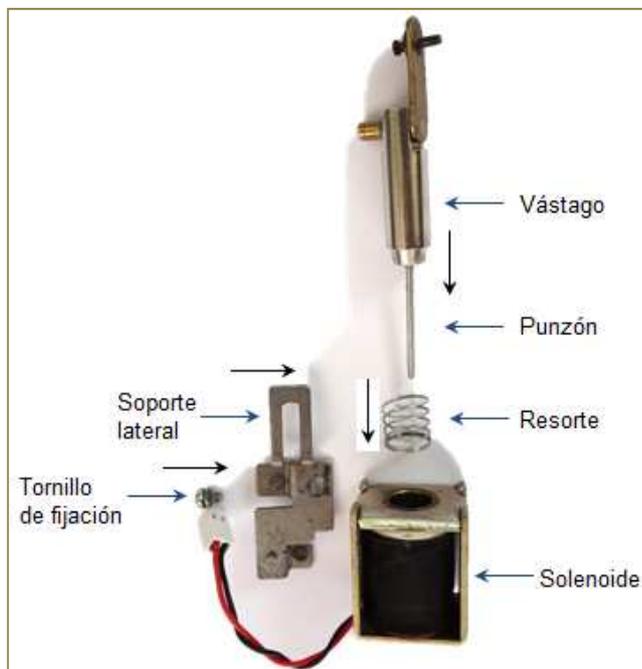


Figura 2.6 Acondicionamiento mecánico del conjunto solenoide – punzón



Figura 2.7 Detalle de solenoide acoplado al punzón

En la Figura 2.8 se observa la placa inferior del soporte del conjunto solenoide – punzón, como se aprecia posee una lámina flexible construida en base a acetato,

esta lámina asegura que si el papel se atasca con el punzón al realizar el relieve la misma liberará al papel empujándolo hacia la base de impresión.

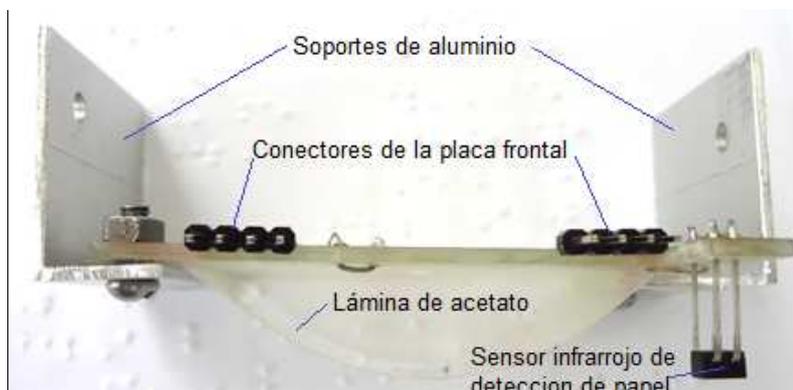


Figura 2.8 Detalle de la placa inferior del soporte del solenoide

En la Figura 2.9 se aprecia la placa frontal del soporte del solenoide que además contiene los elementos de control del mismo, en la Figura 2.10 se detalla el ensamblaje de las partes que componen el soporte del solenoide y su placa de control, las placas que le brindan firmeza al conjunto se construyen en aluminio; se elige éste material debido a su bajo peso, ya que el solenoide es un elemento que aporta gran cantidad de carga (179 gramos) y el conjunto mecánico dispuesto para movilizar el carrito no puede superar los 210 gramos de carga aproximadamente [10].

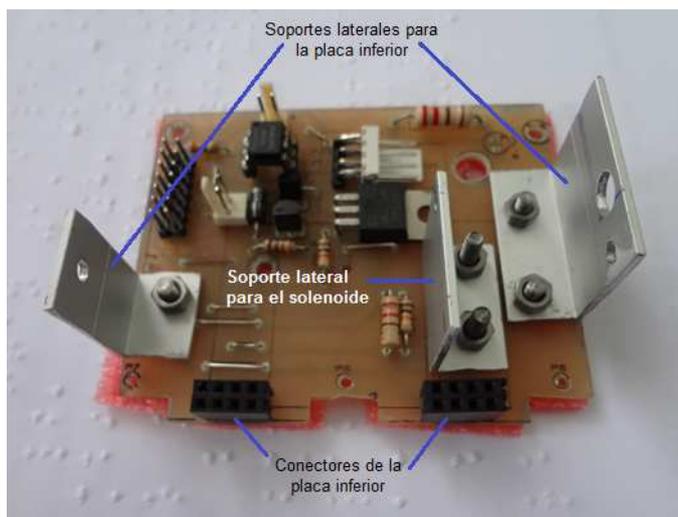


Figura 2.9 Detalle de la placa frontal del soporte del solenoide

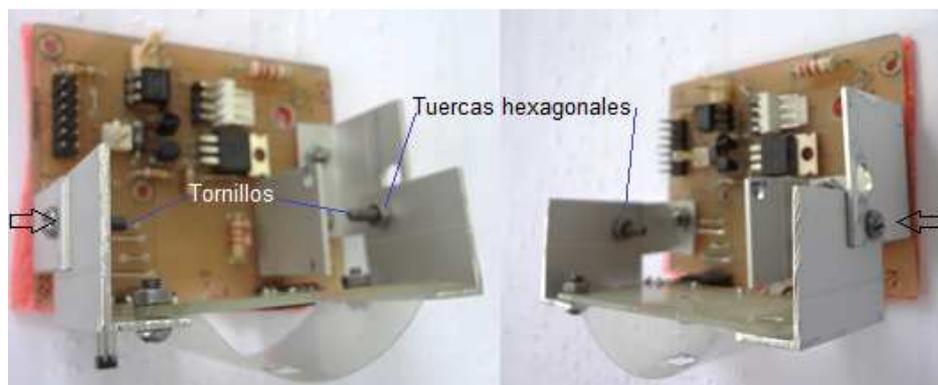


Figura 2.10 Ensamblaje del soporte de solenoide

Para colocar el solenoide en su posición final, su soporte previamente armado debe ser fijado al carrete de la impresora mediante tres tornillos de cabeza de estrella, debido a que el cuerpo del solenoide obstaculiza la labor antes mencionada; así mismo se enchufan los tres conectores externos quedando libre únicamente el conector de alimentación al solenoide, en la Figura 2.11 se puede observar el detalle de la fijación del soporte.

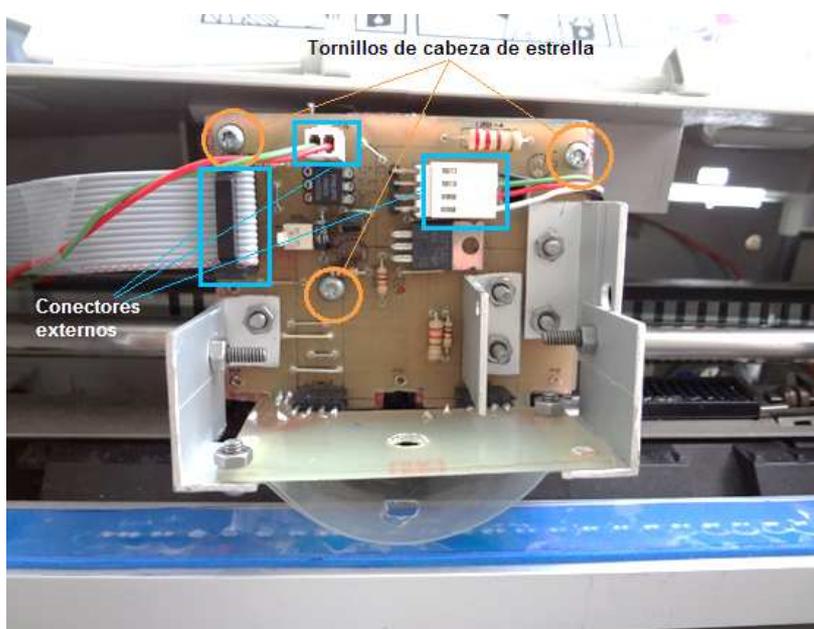


Figura 2.11 Detalle de la fijación del soporte del solenoide

En la Figura 2.12 se ilustra la manera en que se debe colocar el solenoide y fijarlo con un solo tornillo lateral a su base, cabe anotar que para introducir dicho tornillo es necesario utilizar un paso ubicado en una placa de fijación lateral.

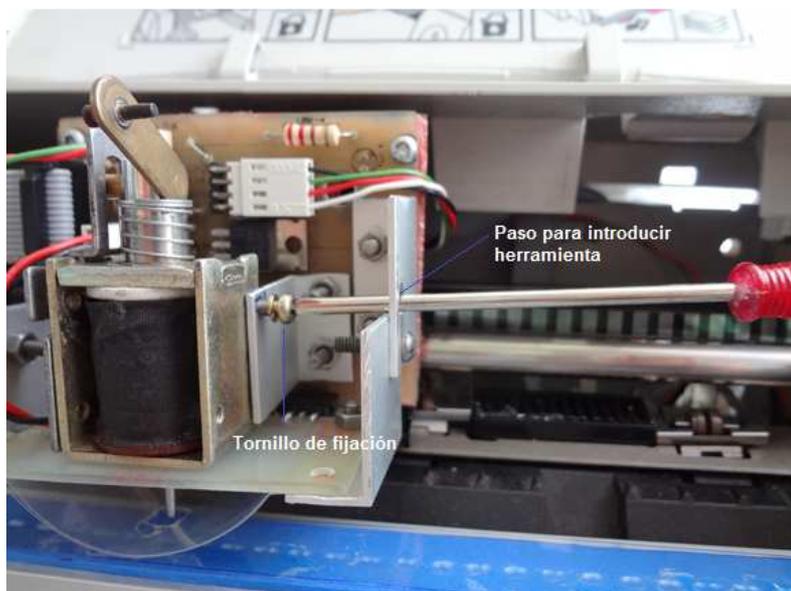


Figura 2.12 Detalle de la fijación del solenoide

Como paso final se conecta el cabezal de alimentación al solenoide, en la Figura 2.13 se aprecia el módulo punzón acoplado al carrete de la impresora después de haber realizado el proceso de ensamblaje.

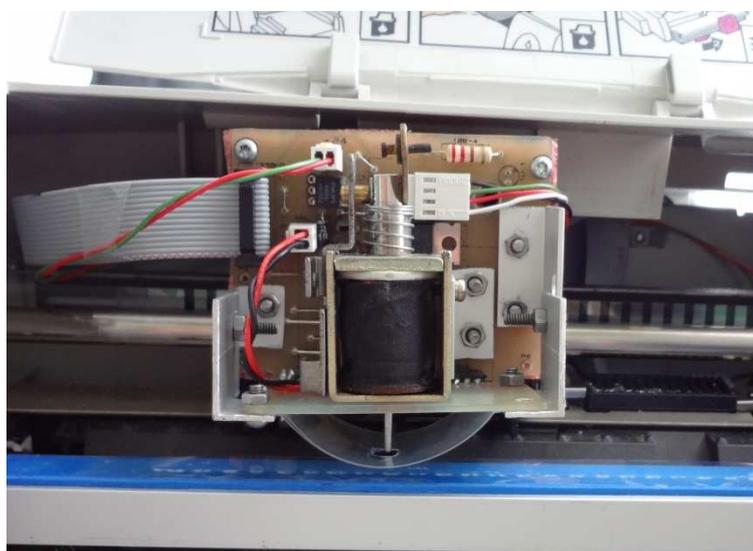


Figura 2.13 Módulo punzón fijado al carrete de la impresora

2.1.1.4 Emplazamiento placa de control principal

Las medidas de la placa de control se toman del alojamiento metálico que se encuentra en la impresora (Figura 2.14), de manera que dicha placa entre adecuadamente en el alojamiento como se observa en la Figura 2.16.

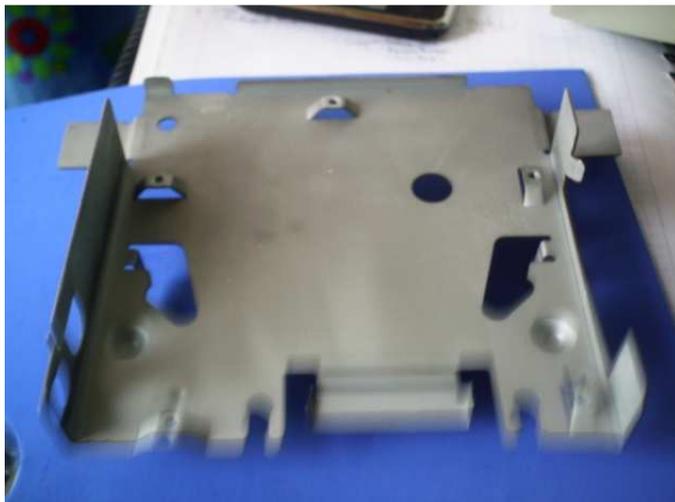


Figura 2.14 Alojamiento de placa de control principal

La placa de control colocada en el alojamiento metálico se puede observar en la Figura 2.15.

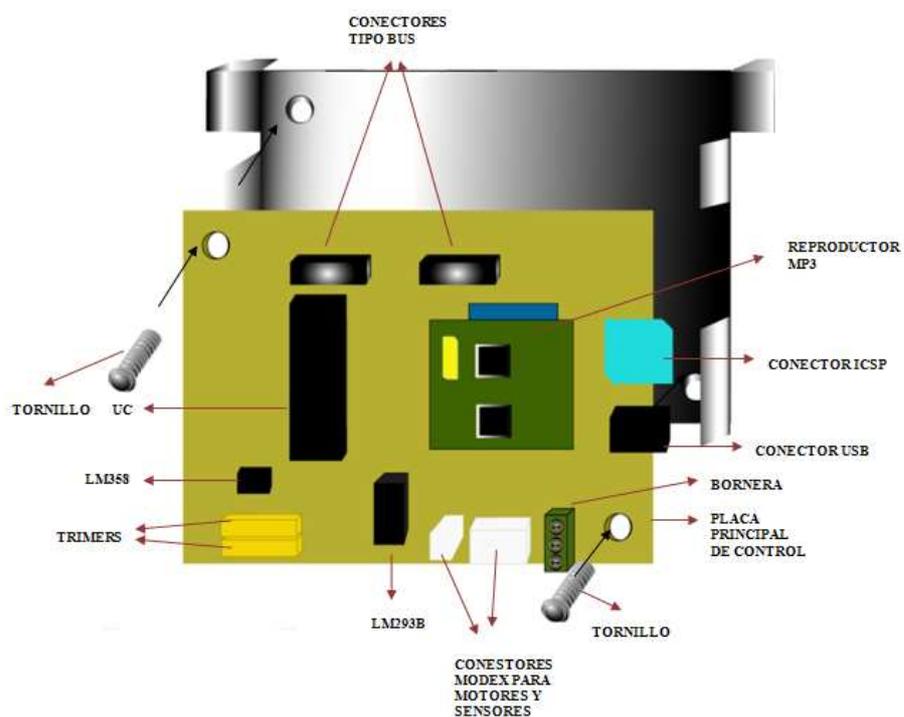


Figura 2.15 Adaptación de la placa de control principal al soporte metálico



Figura 2.16 Placa de control principal insertada en el alojamiento metálico

2.1.1.5 Módulo frontal

En la Figura 2.17 se observan las modificaciones que se realizan en la carcasa del módulo para poder alojar las nuevas placas de control que permitirá al usuario final interactuar con el prototipo.



Figura 2.17 Modificaciones en la parte frontal

En la Figura 2.18 se muestra el resultado final del emplazamiento de la placa frontal.



Figura 2.18 Módulo frontal

Para tener un resultado como el que se observa en la Figura 2.18 se realiza el procedimiento que se indica en la Figura 2.19

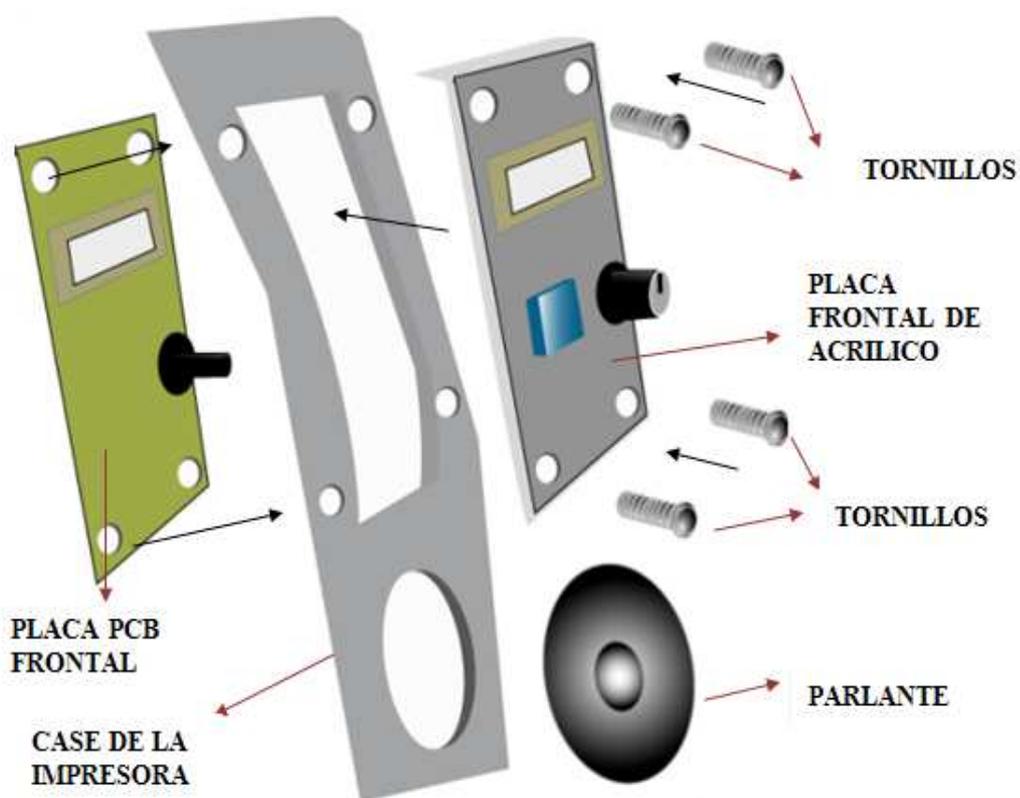


Figura 2.19 Ensamblaje de la parte frontal

2.2 DISEÑO ELECTRÓNICO

El diseño electrónico del prototipo “Impresora Braille” se ha desarrollado en varios módulos para facilitar su estudio e implementación como partes del prototipo final. El detalle de estos módulos se expone en el siguiente diagrama de bloques (Figura 2.20):

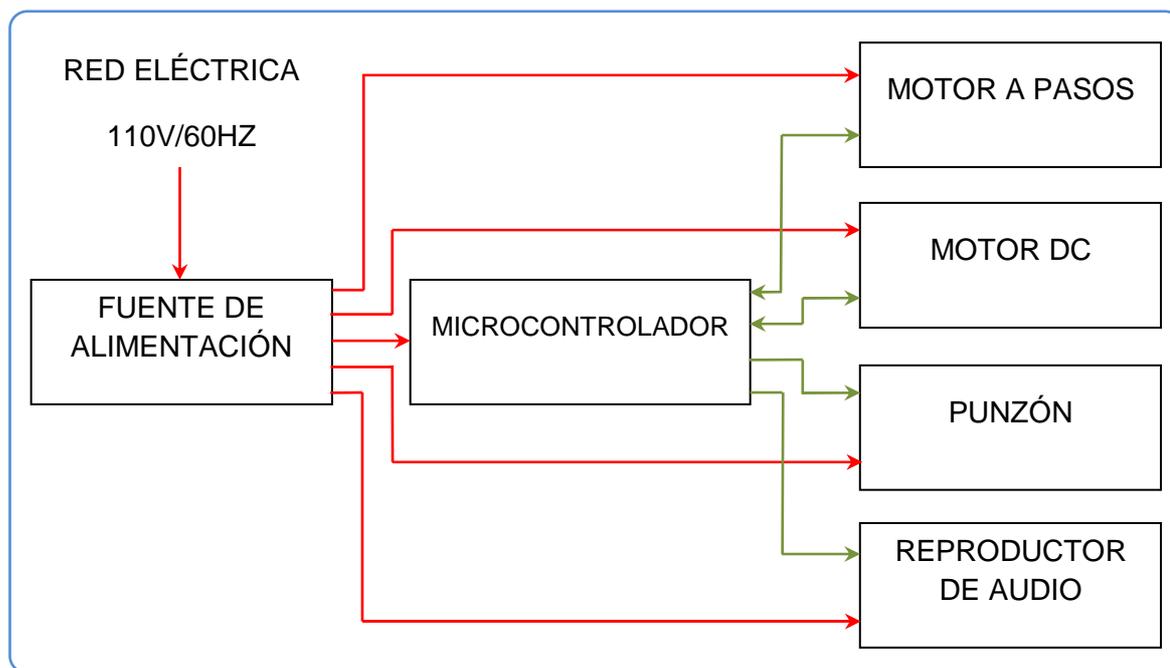


Figura 2.20 Diagrama de bloques del prototipo “Impresora Braille”

Cada módulo consta de varias etapas que se presentan en los siguientes apartados.

2.2.1 MÓDULO MOTOR A PASOS

Éste módulo consta del motor a pasos encargado de mover al rodillo principal el mismo que mueve la hoja de papel a ser impreso, el motor antes mencionado necesita de un driver para su control, debido a que su potencia de operación es diferente a los parámetros considerados en un sistema microcontrolado (voltaje normalmente comprendido entre 3.3 Vdc a 5 Vdc e intensidad de corriente entre 15 mA. a 200 mA. [12]), de igual manera para conocer si no se encuentra atascado o sufre de fallos en su operación el módulo está dotado de un encoder basado en un encapsulado emisor – receptor de rayos infrarrojos, la Figura 2.21 ilustra las partes de las que consta éste módulo.



Figura 2.21 Diagrama de bloques “Módulo motor a pasos”

2.2.1.1 Driver motor a pasos unipolar

Para controlar el motor a pasos se implementa la secuencia denominada “paso simple” pues gracias al moto reductor presente en la parte mecánica del rodillo, el torque que el motor proporciona con esta secuencia es suficiente para poder mover el papel presente en la bandeja de entrada.

El motor utilizado en la impresora posee los siguientes datos de placa (Anexo C.1):

Motor unipolar

Voltaje nominal = 24 Vdc

Intensidad máxima = 800 mA.

Resolución = 7.5° por paso

Se coloca el driver integrado de transistores en configuración Darlington “ULN2003” porque sus parámetros de funcionamiento están acorde con los requerimientos del motor según los datos de su fabricante (Anexo C.4), además se elige esta opción por ahorro de espacio en la construcción del circuito impreso y su fácil reemplazo en caso de daños.

Sin embargo la intensidad de salida del integrado “ULN2003” es de 600 mA. y esta difiere con 200 mA. respecto a los datos de placa del fabricante del motor a pasos, por lo que en pruebas realizadas en el conjunto motor pasos – moto reductor el motor demanda 400 mA. por bobinado si se lo alimenta con 12 Vdc. aproximadamente, con esta disminución en el voltaje de operación los valores

nominales del driver son adecuados para el motor y mediante pruebas realizadas se verifica que el torque proporcionado por el motor en estas nuevas condiciones es el adecuado para desplazar el papel dentro del rodillo.

En la Figura 2.22 se detalla la forma en la que se comanda el motor a pasos con su respectivo driver:

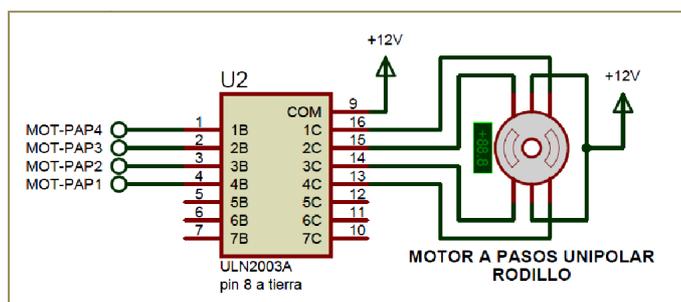


Figura 2.22 Conexión de motor a pasos y driver.

2.2.1.2 Encoder de rodillo

Éste circuito se implementa únicamente para verificar que el rodillo tenga movimiento, esto facilita la detección de atascamientos de papel y posibles roturas de los engranajes presentes en el moto reductor acoplado entre el rodillo y el motor a pasos.

El encoder en su parte electrónica está basado en un encapsulado emisor receptor de infrarrojo CNY 70 (Anexo C.10), para el dimensionamiento de la resistencia que limita la corriente en el led infrarrojo emisor se usa un parámetro denominado *tasa de transferencia de corriente* o CTR (current transfer ratio), éste parámetro relaciona las corrientes de colector y de led interno expresada en porcentaje como se observa en la Ecuación 1.

$$CTR = \frac{I_c}{I_f} * 100\% \quad \text{Ecuación 1}$$

Donde:

CTR = tasa de transferencia de corriente

I_c = intensidad de corriente de colector del transistor

I_f = intensidad de corriente que circula por el led infrarrojo

El valor de “ I_c ” está dado por el fabricante en función de la distancia en la que se coloque el sensor de la superficie reflectante según la Figura 2.23, cabe anotar que la distancia a la que se coloca el sensor de la superficie reflectante en el prototipo es de 1 mm. aproximadamente.

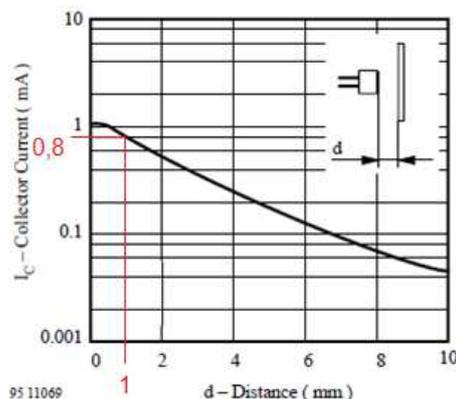


Figura 2.23 Gráfico de corriente de colector vs. distancia de superficie reflectante para el sensor CNY 70, tomado de [14]

De la Figura 2.23 se tiene que $I_c = 0,8 \text{ mA}$, además el valor de CTR para éste sensor es de 5 % [14], reemplazando estos datos en la Ecuación 1 se obtiene:

$$CTR = \frac{I_c}{I_f} * 100\%$$

$$5\% = \frac{0,8 \times 10^{-3}}{I_f} * 100\%$$

Despejando I_f se tiene:

$$I_f = \frac{100 * 0,8 \times 10^{-3}}{5}$$

$$I_f = 16 \text{ mA}$$

Considerando una caída de voltaje de 1.5 V. para un Led infrarrojo se tiene que:

$$R = \frac{V_{cc} - V_{led}}{I_f} \quad \text{Ecuación 2}$$

Donde:

R = resistencia

V_{cc} = voltaje directo (fuente)

V_{led} = voltaje de consumo de led

I_f =intensidad o corriente que circula por el led

$$V_{cc} = 5V$$

$$I_f = 16 \text{ mA}$$

$$R = \frac{(5 - 1.5)V}{0.016 \text{ A.}}$$

$$R = 218,75 \Omega$$

Aplicando un factor de seguridad del 10% se tiene:

$$R = 218,75 \Omega * 1.1 = 240.6 \Omega$$

Según los valores de resistencias comerciales se escoge el inmediato inferior de:

$$R = 220 \Omega$$

Para el ramal de carga, el cual consta de un foto transistor npn dentro del encapsulado, se tiene que elegir un valor de resistencia que esté dentro de un rango delimitado por la resistencia calculada en base a $I_c = 0,8 \text{ mA}$ (valor anterior) e I_c de saturación = $0,1 \text{ mA}$ [14], éste rango se calcula a continuación:

$$R_l = \frac{V_{cc} - V_{ce}}{I_c} \quad \text{Ecuación 2}$$

Donde:

R_l = resistencia de carga en el transistor

V_{cc} = voltaje directo (fuente)

V_{ce} = voltaje existente entre el colector y el emisor

I_c =intensidad o corriente que circula por el colector

$$V_{cc} = 5V$$

$$I_c = 0,8 \text{ mA}$$

$$Rl_{min} = \frac{(5 - 0,6)V}{0.0008 \text{ A.}}$$

$$Rl_{min} = 5,5 \text{ K}\Omega$$

Aplicando un factor de seguridad del 10% se tiene:

$$Rl_{min} = 5,5 \text{ K}\Omega * 1.1 = 6,05 \text{ K}\Omega$$

Para el cálculo de la resistencia máxima se usa el valor de I_c de saturación que según el fabricante es de 0,1 mA [14].

$$V_{cc} = 5V$$

$$I_{c_{sat}} = 0,1 \text{ mA}$$

$$Rl_{max} = \frac{(5 - 0,6)V}{0.0001 \text{ A.}}$$

$$Rl_{max} = 44 \text{ K}\Omega$$

Aplicando un factor de seguridad del 10% se tiene:

$$Rl_{max} = 44 \text{ K}\Omega * 1.1 = 48,4 \text{ K}\Omega$$

De los valores anteriores se tiene que:

$$48,4 \text{ K}\Omega < Rl < 6,05 \text{ K}\Omega$$

Según los valores de resistencias comerciales se escoge un valor relativamente cercano al mínimo por lo que se tiene:

$$Rl = 10 \text{ K}\Omega$$

La salida de éste sensor es análoga con lo que se procede a tratar la señal de detección con un circuito comparador basado en un amplificador operacional, que permite calibrar de cierta manera el disparo de detección de una superficie blanca

en contraste de una negra, de esta manera el esquemático del circuito final se observa en la Figura 2.24.

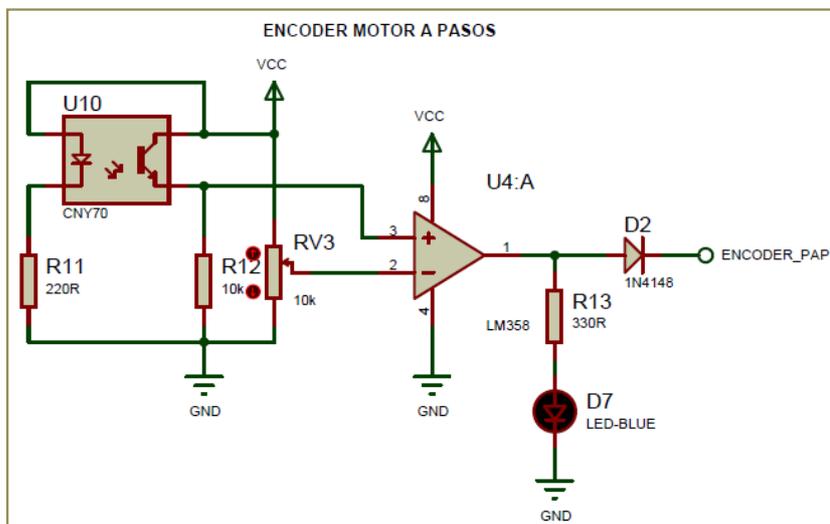


Figura 2.24 Acondicionamiento de señal encoder motor a pasos

En la entrada positiva del comparador se encuentra la señal del sensor, y en la entrada negativa se encuentra el nivel de voltaje al cual se dispara la señal en alto, cuyo máximo valor estará determinado por V_{cc} en éste caso +5 Vdc.

A la salida del comparador se tiene un indicador, el cual se encenderá en la detección de una superficie de color blanco, cumpliendo también la labor de proporcionar la calibración correcta, para dimensionar la resistencia limitadora del led se aplica la ecuación 1 reemplazando datos referentes a voltaje y corriente para los leds normales, se tiene:

$$R = \frac{V_{cc} - V_{led}}{I} \quad \text{Ecuación 2}$$

$$V_{cc} = 5V$$

$$V_{led} = 2.3V$$

$$I_{led} = 10 \text{ mA}$$

$$R = \frac{(5 - 2.3)V}{0.01 \text{ A}}$$

$$R = 270 \Omega$$

Aplicando un factor de seguridad del 10% se tiene:

$$R = 270 \, \Omega * 1.1 = 297 \, \Omega$$

Según los valores de resistencias comerciales se escoge el inmediato superior de:

$$R = 330 \, \Omega$$

El diodo 1N4148 de conmutación rápida ubicado a la salida del amplificador operacional impide que al momento de la grabación del microcontrolador, misma que se realiza sin sacarlo de la placa de circuito impreso, el voltaje de operación del grabador disminuya considerablemente al tener que encender el led monitor ubicado en la salida del amplificador operacional, esta caída de voltaje ocasiona un error en el programa de grabación el mismo que impide la transferencia del programa generado en el computador al microcontrolador presente en el prototipo.

El encoder como tal está formado de una rueda segmentada impresa en un papel fotográfico adherido a una rueda de plástico de la misma dimensión del estator del motor a pasos, está fijado al eje que sobresale en la parte posterior de su alojamiento y se encuentra dentro de la carcasa de la impresora por lo que solo es posible apreciarlo cuando la misma es desmontada totalmente, la implementación de éste punto se observa en la Figura 2.25 en la que también se puede apreciar el circuito impreso que contiene al opto detector CNY70.

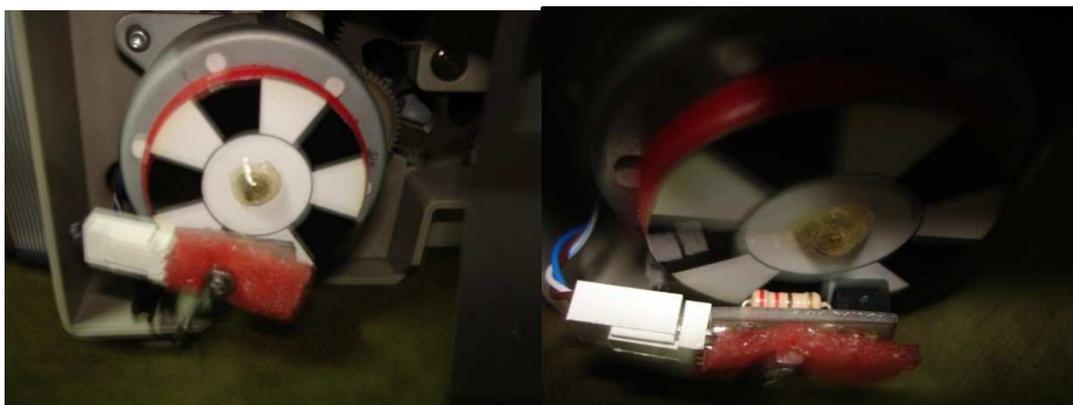


Figura 2.25 Detalle de implementación del encoder presente en el motor a pasos

2.2.1.3 Señales de Control

Éste módulo es regido por cuatro señales digitales de 0 o 5 Vdc., estas señales comandan la secuencia llamada “en olas” o paso simple para el control del motor unipolar a pasos presente en el rodillo de la impresora.

2.2.1.4 Tipo de respuesta

En la parte final del punto 2.2.1.2 se observa que el tipo de respuesta es una señal de tipo on/off, misma que es proporcionada por la salida del comparador implementado con el amplificador operacional, que básicamente se pondrá en alto (+5 Vdc) cuando exista la presencia del color blanco en la rueda del encoder incremental acoplado al eje del rodillo.

2.2.1.5 Requerimientos de alimentación

El módulo necesita:

$$V_{cc} = 5 \text{ Vdc}$$

$$I_{cc} = 50 \text{ mA}$$

Para el acondicionamiento de señales del encoder presente en el motor a pasos unipolar, y:

$$V_m = 12 \text{ Vdc}$$

$$I_m = 800 \text{ mA}$$

Para el funcionamiento del motor acoplado al rodillo de la impresora.

2.2.2 MÓDULO MOTOR DC

El motor de DC presente en éste módulo está encargado del movimiento de izquierda a derecha del carrete de la impresora en donde está alojado el punzón, de igual manera para su control es necesario el uso de un driver que permita el cambio de giro y control de velocidad.

Para poder ubicar de una manera precisa el carrete a lo largo de su recorrido horizontal es necesario el uso de un encoder que informe al microcontrolador el avance o retroceso en una unidad de medida aceptable para éste proyecto, de igual manera para delimitar el recorrido del carrete es necesario colocar finales de

carrera en los extremos del eje de operación del mismo, por lo antes expuesto éste módulo consta de las partes mostradas en la Figura 2.26.



Figura 2.26 Diagrama de bloques “Módulo motor DC”

2.2.2.1 Driver motor DC

Para poder ubicar el carrito que lleva el punzón a lo largo de la hoja a ser impresa es necesario que el motor de DC se mueva de izquierda a derecha y viceversa con velocidades variables para disminuir el tiempo del proceso de impresión, por lo que se implementa un control de inversión de giro con variación de velocidad a través de un puente H.

Por los niveles de voltaje y corriente que maneja, la facilidad en su reemplazo en caso de fallas y por sus dimensiones reducidas se emplea el driver integrado “L293D” basado en transistores, el mismo que cuenta en su encapsulado con diodos de protección para sus salidas, así mismo es capaz de controlar dos motores con inversión de giro y variación de velocidad siempre y cuando estos tengan el mismo voltaje de operación o a su vez controlar la velocidad de giro en una sola dirección de cuatro motores de DC, sus datos de operación (Anexo C.5) según su fabricante se pueden ver a continuación:

Voltaje máximo de alimentación = 36 Vdc

Voltaje mínimo de entrada = 1.5 Vdc

Intensidad pico no repetitivo = 1.2 A.

Intensidad de salida por cada canal= 600 mA

El motor DC presente en la impresora posee los siguientes datos de placa (Anexo C.2):

Motor DC:

Voltaje nominal = 24 Vdc.

Intensidad máxima = 2.5 A.

Intensidad en vacío = 0.15 A.

Considerando que de cada canal del driver L293D solamente se pueden drenar 600 mA, se realizan pruebas con el motor de DC alimentándolo a diferentes voltajes, cargando al carrete con un peso de 198 gr. (todo el conjunto de módulo punzón alojado en el carrete posee éste peso), el motor DC no supera la cantidad de corriente máxima del driver operando a 12 Vdc.

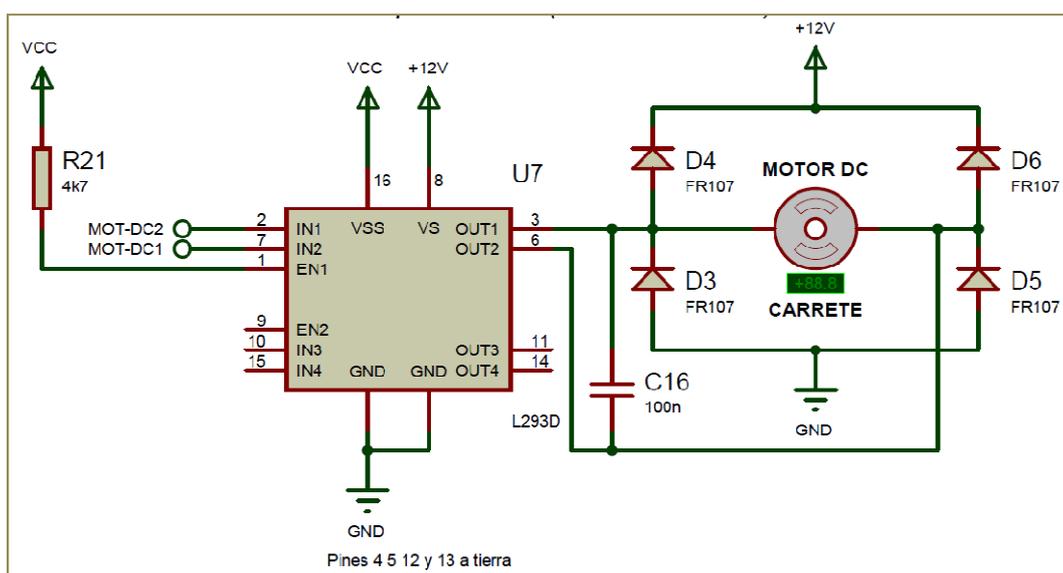


Figura 2.27 Conexión de motor DC y driver.

La resistencia marcada como R21 tiene un valor recomendado por el fabricante para la activación de un solo canal de control. El condensador cerámico presente entre los terminales del motor reduce el efecto de interferencia que podría producir el mismo, éste efecto ocasionaría resets esporádicos del sistema microprocesado, de ahí la importancia de atenuar o minimizar sus efectos.

El driver presenta diodos internos en cada canal para proteger sus salidas, sin embargo el fabricante recomienda colocar un puente de diodos externos

rectificadores con capacidades de voltaje y corriente adecuados para los datos de placa del motor DC utilizado, como se implementa un control de velocidad basado en un PWM sus tiempos de conmutación y la corriente drenada justifican el uso de diodos de conmutación rápida de silicio como el diodo FR107GS (Anexo C.13).

2.2.2.2 Encoder de motor DC acoplado al carrete

En la impresora se encuentra un sensor foto eléctrico con un principio de interrupción de rayo, acoplado a una cinta transparente con divisiones que se muestran en la Figura 2.28.

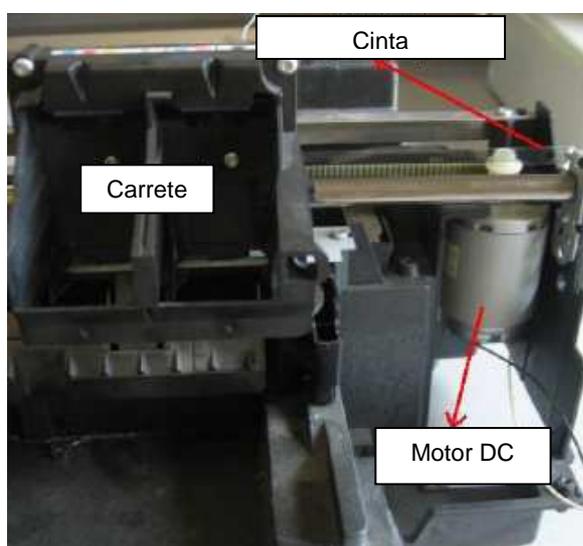


Figura 2.28 Detalle físico motor DC en impresora.

El sensor Q9874 solamente necesita un voltaje de alimentación de 5 Vdc y en sus dos salidas presenta, según su paso por la cinta de encoder, pulsos defasados 90 grados. La cinta original y el sensor se detallan en la Figura 2.29.

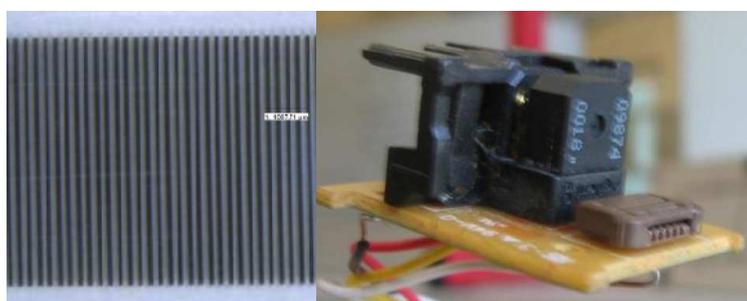


Figura 2.29 Cinta transparente y sensor Q9874 (encoder).

Para éste proyecto es necesario que las medidas del encoder cumplan con las dimensiones de las celdas Braille especificadas en el Capítulo 1 para efectuar una impresión de caracteres que cumplan con las normativas, por lo que se construye una cinta con marcas que definen estas medidas.

Se reemplaza la cinta o grilla presente en la impresora por la grilla que posee las dimensiones de los caracteres Braille en una fila, esta banda se puede observar en la Figura 2.30.



Figura 2.30 Grilla construida con las dimensiones de los caracteres Braille

2.2.2.3 Final de carrera

Por su reducido espacio de ocupación y la versatilidad en su implementación, se elige un final de carrera foto eléctrico basado en un sensor GP2S40 (Anexo C.9) cuyos datos de operación son parecidos al sensor CNY70 por lo que su diseño no se diferencia de éste y se ha desarrollado en el apartado 2.1.2, su esquema se puede observar en la Figura 2.31.

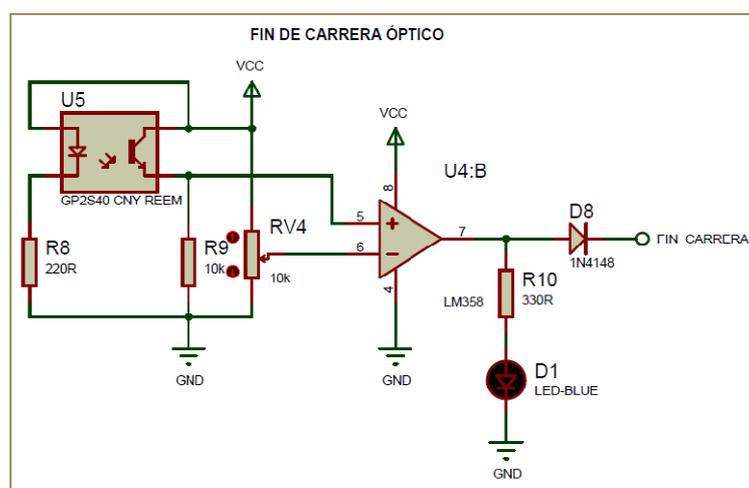


Figura 2.31 Esquema del final de carrera óptico basado en un sensor GP2S40

En el prototipo se implementan marcas de contraste blanco - negro que delimitan el recorrido del carrito al generar señales de detección en el sensor, estas marcas se aprecian en la Figura 2.32.



Figura 2.32 Marcas de contraste para el fin de carrera óptico GP2S40

2.2.2.4 Señales de Control

Para controlar el motor DC en su velocidad es necesario implementar una señal PWM con una frecuencia determinada por la velocidad de conmutación del driver implementado.

Se toma en cuenta la Figura 2.33 que proporciona el fabricante [15], en esta figura se pueden apreciar los tiempos de switcheo propios del driver.

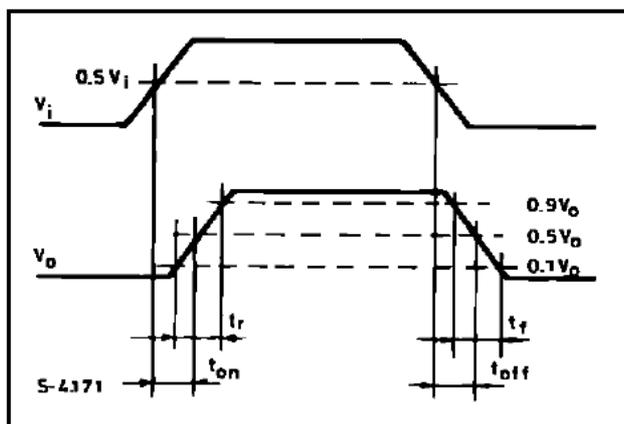


Figura 2.33 Diagrama de tiempos de conmutación para el driver L293D, tomado de [15]

El fabricante proporciona los siguientes datos de tiempos para éste driver:

$$tr = 250 \text{ nS}$$

$$tf = 250 \text{ nS}$$

$$ton = 750 \text{ nS}$$

$$toff = 200 \text{ nS}$$

La determinación de la frecuencia de operación según estos datos y tomando en cuenta las formas de onda referentes al tiempo (Figura 2.33) se calcula en base a la Ecuación 3 y 4, así:

$$T = ton + \frac{tr}{2} + toff + \frac{tf}{2} \quad \text{Ecuación 3}$$

Donde:

T = período de tiempo de operación

ton = tiempo de encendido

tr = tiempo de encumbramiento o crecimiento

$toff$ = tiempo de apagado

tf = tiempo de caída

$$T = \left[750 + \frac{250}{2} + 200 + \frac{250}{2} \right] \times 10^{-9} \text{S}$$

$$T = 1,2 \mu\text{S}$$

Con el dato del período calculado se puede aplicar la Ecuación 4 para obtener el valor de la frecuencia permisible por éste driver, así:

$$f = \frac{1}{T} \quad \text{Ecuación 4}$$

Donde:

f = frecuencia

T = período de tiempo de operación

$$f = \frac{1}{1,2 \mu\text{S}}$$

$$f = 833,33 \text{ KHz}$$

El valor anterior muestra la máxima frecuencia a la cual puede operar el driver antes mencionado. El microcontrolador debe encargarse también de atender continuamente la comunicación USB, en otras palabras debe mantener “vivo” el canal de comunicación, por lo que se elige una frecuencia de operación de **500 Hz.** para el PWM, que es relativamente baja, con esto el microcontrolador no dejará de producir ningún pulso mientras atiende el proceso de comunicaciones.

2.2.2.5 Tipo de respuesta

El encoder implementado produce, según sea el movimiento del carrete, pulsos con un nivel máximo de +5 Vdc. que serán contabilizados en el microcontrolador determinando o informando la ubicación del punzón para realizar o no la impresión mecánica según sea el caso.

2.2.2.6 Requerimientos de alimentación

El módulo necesita:

$$V_{cc} = 5 \text{ Vdc}$$

$$I_{cc} = 200 \text{ mA}$$

Estos valores son los típicamente demandados en sistemas microcontrolados [12], y:

$$V_m = 12 \text{ Vdc}$$

$$I_m = 600 \text{ mA}$$

Para el funcionamiento del motor DC acoplado mecánicamente al carrete que moviliza al punzón.

2.2.3 MÓDULO PUNZÓN

Para realizar la impresión mecánica de los puntos en relieve que requiere el sistema Braille se construye un mecanismo que produce de una manera eficiente un relieve de 0,5 mm. en el papel según las normativas de la celda Braille [3].

Así se tiene un sistema basado en un principio de percusión con punta roma (sin filo), esta punta avanza dentro del núcleo del solenoide según lo determine la

excitación del mismo alimentado con voltaje DC y retrocede por la ausencia de voltaje en el bobinado junto con la acción de un resorte acoplado al eje principal del mismo, para controlar la activación del solenoide se implementa un circuito basado en un Mosfet y tomando otro nivel de voltaje para la parte de alimentación al microcontrolador ya que los requerimientos de voltaje para el microcontrolador se toman entre GND y +5V y para la activación del Mosfet se toman 12V de entre -12V y GND y para la bobina se requieren 24V que se toman de entre -12V y +12V de la fuente ATX. Para separar los niveles de voltaje requeridos se utiliza un opto acoplador cuyo diseño se detalla más adelante, la Figura 2.34 ilustra las partes de las que consta éste apartado.



Figura 2.34 Diagrama de bloques “Módulo Punzón”

2.2.3.1 Driver de Solenoide

Para producir un avance del vástago del solenoide acoplado al punzón es necesario excitar la bobina con un nivel de voltaje DC, cabe anotar que el solenoide se recicla de una fotocopiadora antigua de la que no se posee la referencia de marca o modelo, únicamente se distingue en el cuerpo del elemento el voltaje nominal de operación y éste es **12 Vdc**. Mediante pruebas realizadas con éste elemento se logra conseguir un relieve aceptable para las normativas Braille excitando la bobina a 24 voltios, es decir al doble del voltaje nominal marcado en su case, en estas condiciones se observa un consumo de corriente de aproximadamente 532 a 533 mA en estado estable, sin embargo por las características altamente inductivas del elemento; éste consume una corriente pico de valor:

$$I_p = 8 * I_n \quad \text{Ecuación 5}$$

Donde:

I_p = corriente pico

I_n = corriente nominal de operación

Es necesario aclarar que al no tener un dato del fabricante respecto al consumo de corriente nominal del elemento y al alimentarlo con el doble de su voltaje nominal, para diseñar su driver es necesario adjudicar como I_n al valor obtenido experimentalmente de 533 mA., así:

$$I_p = 8 * (533 \text{ mA})$$

$$I_p = 4,26 \text{ A}$$

Al emplear 24 Vdc para excitar la bobina del solenoide por períodos de tiempo muy cortos (7,6 mS.) se discurre que la reducción de vida útil del elemento no se encuentra comprometida en un valor considerable.

En la Figura 2.35 se detalla el circuito esquemático empleado para activar y desactivar el elemento solenoide, mismo que está basado en un semiconductor mosfet, y cuyo diseño esta explicado más adelante.

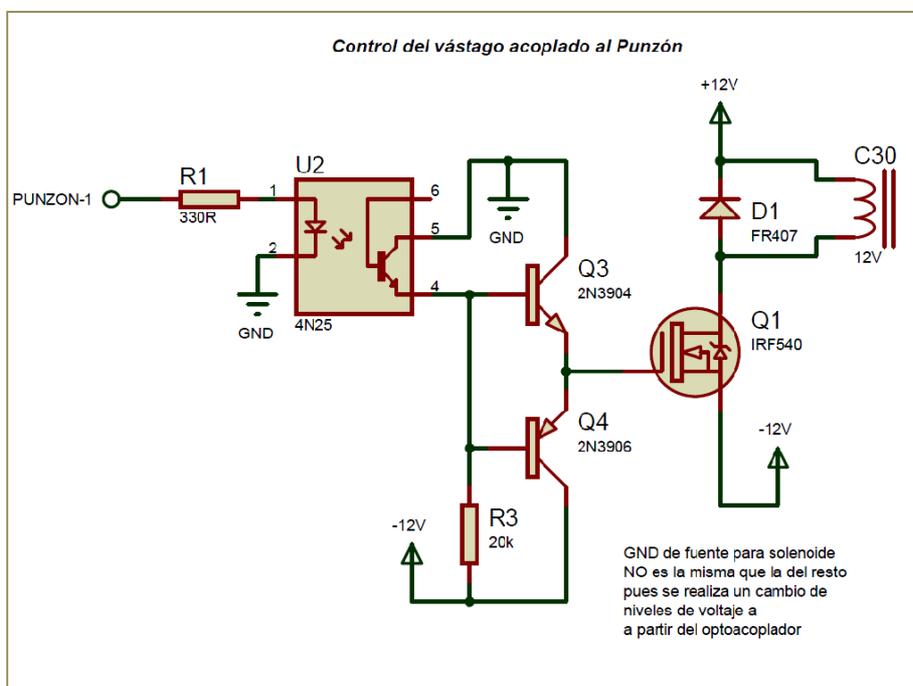


Figura 2.35 Esquema de conexión del driver del solenoide acoplado al punzón

El solenoide que se emplea para realizar el relieve del caracter Braille posee los siguientes datos, obtenidos experimentalmente:

Voltaje de operación = 24 Vdc.

Intensidad de operación = 533 mA.

Intensidad pico = 4,26 A.

Al ser considerada una carga altamente inductiva se requiere un diodo colocado en anti paralelo para asegurar que la corriente almacenada en el solenoide no se distribuya al resto del circuito sino que más bien recircule entre éste dispositivo y el diodo, tomando en cuenta esto será necesario que el diodo soporte la corriente pico del solenoide y sus características de switcheo estén acordes con los tiempos en los que se activa o desactiva la carga, debido a los puntos anteriores se elige un diodo rápido FR107 cuyas características están acordes con esta implementación y se detallan a continuación:

Voltaje de operación = 50 Vdc.

Intensidad de pico no repetitivo = 30 A.

Intensidad máxima de operación = 1 A.

Para activar o desactivar el solenoide se emplea un semiconductor mosfet por su alta capacidad de manejo de corriente y bajos tiempos de switcheo pues para realizar un relieve acorde a los parámetros Braille experimentalmente se determina un tiempo de activación del solenoide no mayor a 7,6 ms. y un tiempo de apagado de 800 ms., cabe mencionar que el tiempo de apagado es mucho mayor debido a que el retorno del vástago depende de la ausencia de voltaje en la bobina y la acción de un resorte acoplado al eje de avance del mismo, debido a que el tiempo de encendido es el más crítico se elige un dispositivo acorde con estos parámetros siendo el FR540 la opción más adecuada, sus datos (Anexo C.3) se detallan a continuación:

$Td(on) = 60 \text{ nS.}$

$Tr = 40 \text{ nS.}$

$ID = 22 \text{ A.}$

$Vdss = 100 \text{ V.}$

Tomando en cuenta que los requerimientos del sistema dictan que el solenoide sea encendido cuando se presente una señal en alto, se implementa un circuito de activación basado en dos transistores de señal el uno npn y el restante pnp para determinar un camino seguro a la corriente que fluye al activar el semiconductor como se observa en la Figura 2.35, cabe anotar que para el activado de estos semiconductores se requiere aplicar entre la compuerta y la referencia un voltaje típico de entre 12 y 15 Vdc. en éste caso se aplican 12 Vdc.

Debido a la diferencia entre los niveles de voltaje en los que operan el módulo punzón y el microcontrolador, y a los diferentes requerimientos de corriente se toman diferentes niveles de referencia de la fuentes ATX mediante el optoacoplador 4N25 (Anexo C.14) se consigue separar estos niveles de voltaje, para el dimensionamiento de la resistencia que limita la corriente en el led emisor interno se usa del mismo modo que el dimensionamiento de los parámetros del integrado CNY 70 explicado anteriormente, el parámetro denominado *tasa de transferencia de corriente* o CTR (current transfer ratio) que se puede observar en la Ecuación 1.

$$CTR = \frac{I_c}{I_f} * 100\% \quad \text{Ecuación 1}$$

Donde:

CTR = tasa de transferencia de corriente

I_c = intensidad de corriente de colector del transistor

I_f = intensidad de corriente que circula por el led interno

El valor de “*I_c*” está dado por el fabricante, para el caso de DC se cuantifica en 1 mA, además el valor de CTR para éste sensor es de 10%, reemplazando estos datos en la Ecuación 1 se obtiene:

$$CTR = \frac{I_c}{I_f} * 100\%$$

$$10\% = \frac{1 \times 10^{-3}}{I_f} * 100\%$$

Despejando I_f se tiene:

$$I_f = \frac{100 * 1 \times 10^{-3}}{10}$$

$$I_f = 10 \text{ mA}$$

Considerando una caída de voltaje de 3 V. para un Led según el fabricante, se tiene que:

$$R = \frac{V_{cc} - V_{led}}{I_f} \quad \text{Ecuación 2}$$

Donde:

R = resistencia

V_{cc} = voltaje de fuente

V_{led} = voltaje de led

I_f =intensidad o corriente que circula por el led

$$V_{cc} = 5V$$

$$I_f = 10 \text{ mA}$$

$$R = \frac{(5 - 3)V}{0,01 A.}$$

$$R = 200 \Omega$$

Aplicando un factor de seguridad del 10% se tiene:

$$R = 200 \Omega * 1.1 = 220 \Omega$$

Según los valores de resistencias comerciales se escoge un valor superior de:

$$R = 330 \Omega$$

Para el ramal de carga, el cual consta de un foto transistor npn, se tiene que elegir un valor de resistencia que esté dentro de un rango delimitado por la resistencia

calculada en base a $I_c = 10 \text{ mA}$ (valor anterior) e I_c de saturación = $0,1 \text{ mA}$, éste rango se calcula a continuación:

$$Rl = \frac{V_{cc} - V_{ce}}{I_c}$$

Donde:

Rl = resistencia ubicada entre base y referencia de los transistores

Vcc= voltaje directo (fuente de activación)

Vce= voltaje existente entre el colector y el emisor del integrado

Ic=intensidad o corriente que circula por el colector

$$V_{cc} = 12V$$

$$I_c = 10 \text{ mA}$$

$$Rl_{min} = \frac{(12 - 0,6)V}{0.01 \text{ A.}}$$

$$Rl_{min} = 1,14 \text{ K}\Omega$$

Para el cálculo de la resistencia máxima se usa el valor de I_c de saturación que según el fabricante es de $0,1 \text{ mA}$.

$$V_{cc} = 12V$$

$$I_{c_{sat}} = 0,1 \text{ mA}$$

$$Rl_{max} = \frac{(12 - 0,6)V}{0.0001 \text{ A.}}$$

$$Rl_{max} = 114 \text{ K}\Omega$$

De los valores anteriores se tiene que:

$$1,14 \text{ K}\Omega < Rl < 114 \text{ K}\Omega$$

Según los valores de resistencias comerciales se escoge un valor ubicado en la mitad de éste rango por lo que se tiene:

$$Rl = 20 K\Omega$$

Con todos los valores determinados se implementa el circuito detallado en la Figura 2.35.

2.2.3.2 Señales de control

Debido a que el solenoide es activado por pulsos según lo determinen los datos recibidos, se realiza el control por medio de dos señales que toman valores de 5 o 0 Vdc determinando así el avance o retroceso del vástago que lleva acoplado el punzón.

2.2.3.3 Requerimientos de alimentación

El módulo necesita:

$$V_{cc} = 5 Vdc$$

$$I_{cc} = 200 mA$$

Para la alimentación de la señal de control determinada por el optoacoplador 4N25, y:

$$V_{sol} = 24 Vdc$$

$$I_{sol} = 4,26 A$$

Para el funcionamiento del solenoide, además de los requerimientos para la activación del driver basado en un mosfet de:

$$V_{act} = 12 Vdc$$

$$I_{act} = 10 mA$$

De los requerimientos mencionados anteriormente cabe destacar que el voltaje de alimentación del solenoide es el doble de los datos marcados en su carcasa, sin embargo éste valor no es aplicado de forma continua sino por periodos relativamente cortos de tiempo, con lo que la vida útil del elemento no se ve seriamente comprometida.

2.2.4 MÓDULO REPRODUCTOR DE AUDIO

Éste proyecto está orientado a personas con deficiencia o discapacidad visual, para éste tipo de personas es indispensable que los artículos que facilitan su estilo de vida posean ayudas auditivas para su correcta utilización, los mensajes sonoros deben ser claros y directos además de modificables según los requerimientos de los diferentes usuarios que tenga el prototipo “Impresora Braille”, de esta forma se implementa en el mismo un reproductor MP3 comercial de audio controlable por un sistema microcontrolado y con salida amplificada, pues también es necesario controlar su volumen de reproducción, así en la Figura 2.38 se ilustran las partes de las que consta éste módulo.

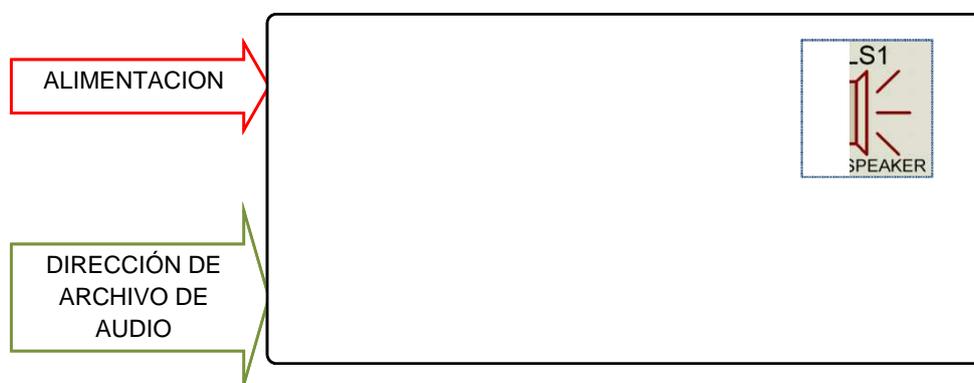


Figura 2.38 Diagrama de bloques “Módulo Reproductor de Audio”

2.2.4.1 Módulo reproductor MP3

El módulo comercial “WTM SD v1.3” (Anexo C.12) tiene la capacidad de leer y proyectar archivos de audio en formato mp3, estos archivos se encuentran alojados en una memoria tipo SD (Storage Device) bajo ciertas consideraciones recomendadas por el fabricante.

La manera en la que se controla éste módulo puede ser: paralelo, serial o por teclas, en éste caso la forma de control que se implementa es la paralela por su rápida respuesta en cuanto a selección de archivos a reproducir, el protocolo utilizado lo especifica el fabricante.

2.2.4.2 Amplificador de señal de audio

El diseño de éste módulo se basa en el amplificador operacional TDA 2003 que se utiliza precisamente para aplicaciones en señales audibles, de acuerdo a las recomendaciones del fabricante quien propone un circuito específico para el

correcto funcionamiento de éste amplificador, se implementa el mismo diseño sin realizar ningún cambio [11], El esquemático del circuito implementado puede observarse en la Figura 2.39.

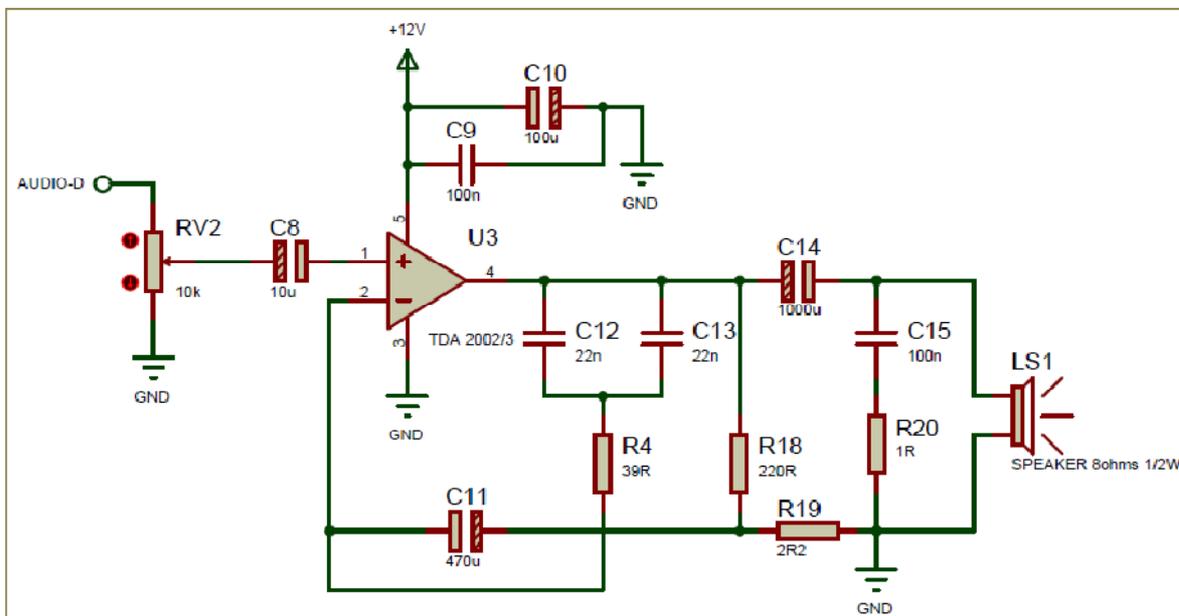


Figura 2.39 Esquemático de implementación del amplificador operacional de audio TDA 2003, tomado de [11]

2.2.4.3 Dirección de archivo de audio

La señal de control que el reproductor de mp3 “WTM-SD v1.3” (Anexo C.12) necesita para operar en paralelo es de tres bits de dirección y para que el reproductor acepte la dirección expuesta se necesita de un flanco negativo producido a su vez por un bit en un solo pin de control.

2.2.4.4 Requerimientos de alimentación

El módulo necesita de un voltaje de alimentación que será el de saturación del amplificador operacional:

$$V_{cc} = 12 \text{ Vdc}$$

Para el valor de la corriente necesaria, es conveniente tener en cuenta la carga del amplificador, en éste caso es un parlante de 8 Ohms a ½ vatio, con lo que:

$$P = V * I * \sqrt{2} \quad \text{Ecuación 6}$$

Donde:

P = potencia disipada

V = voltaje aplicado

I = intensidad de corriente

Reemplazando la Ecuación 2 en la Ecuación 6 se tiene:

$$P = R * I * I * \sqrt{2}$$

$$P = R * I^2 * \sqrt{2}$$

Despejando I:

$$I = \sqrt{\frac{P}{R * \sqrt{2}}}$$

$$I = \sqrt{\frac{0.5 W}{8\Omega * \sqrt{2}}}$$

$$I = 0.210 A = 210 mA$$

Del análisis anterior se concluye que el módulo requiere una corriente de aproximadamente 210 mA.

2.2.5 MÓDULO MICROCONTROLADOR

El prototipo tiene varios sensores de tipo On / Off para ahorrar los pines de lectura y concentrarlos en uno solo del tipo análogo, se implementa un módulo lector de estados en el cual a medida que se enciendan o apaguen los diferentes sensores se formaran divisores de tensión los que producen un valor de voltaje comprendido entre 0 a 5 Vdc que a su vez son leídos por un conversor análogo – digital.

Considerando que no solamente usuarios con discapacidad o deficiencia visual utilizarán el prototipo se coloca un display de cristal líquido en el cual se proyectan mensajes de ayuda y estado de operación del mismo, de igual manera para controlar los diferentes módulos que componen la impresora y para realizar la comunicación con el computador se utiliza un microcontrolador capaz de controlar

y gestionar de una manera eficaz los diferentes requerimientos del prototipo, así en la Figura 2.40 se ilustran las partes de las que consta el presente módulo.

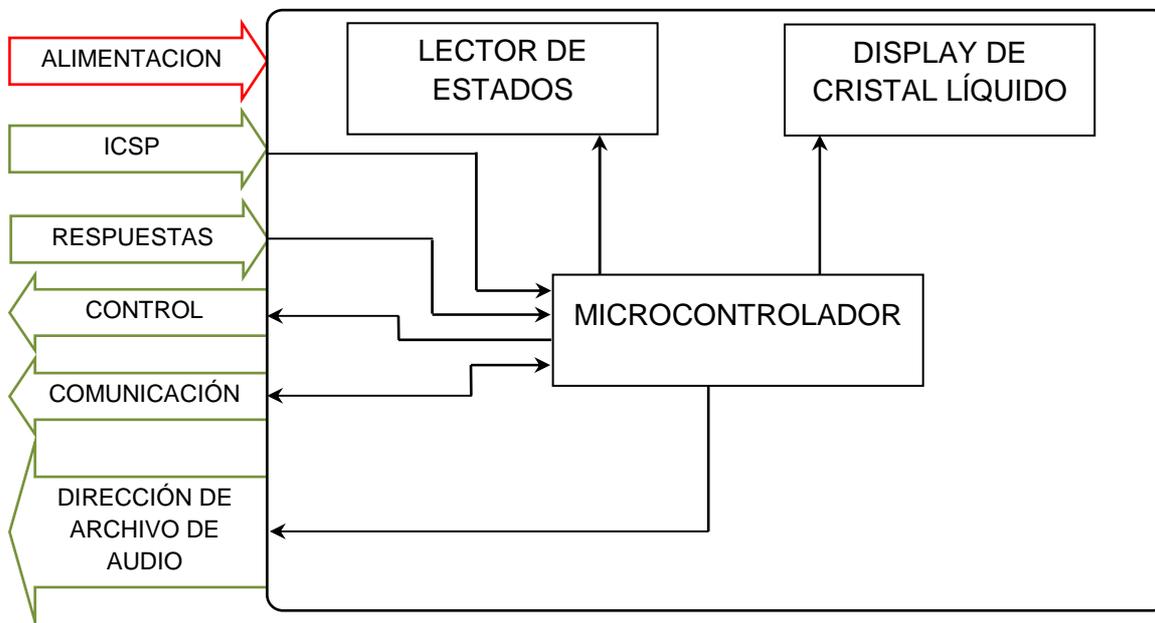


Figura 2.40 Diagrama de bloques “Módulo Microcontrolador”

2.2.5.1 Display de cristal líquido

Para mostrar mensajes de configuración o estado, se utiliza un display de cristal líquido de 2 filas por 16 columnas cuyo esquema de conexión se presenta en la Figura 2.41.

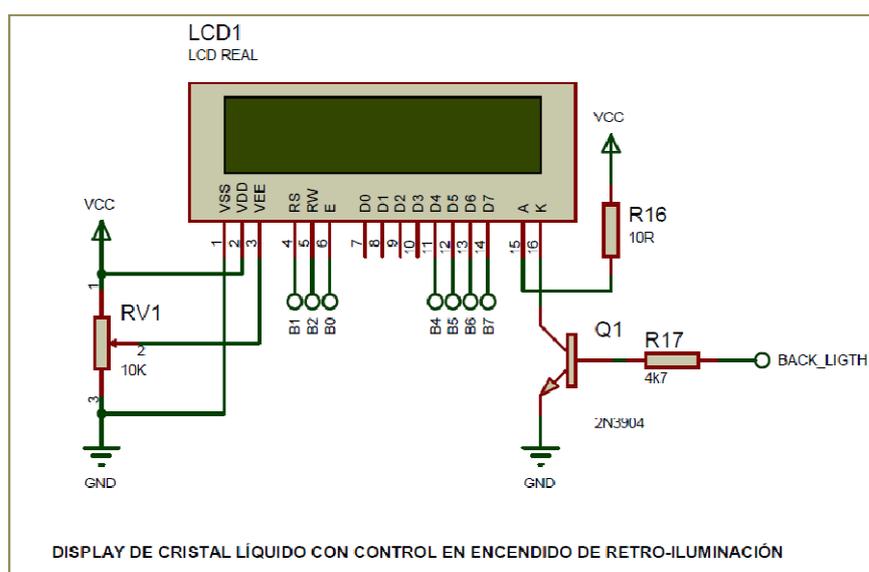


Figura 2.41 Esquema de conexión del display de cristal líquido

El valor de la resistencia variable RV1 es recomendada por el fabricante, así como el valor de la resistencia limitadora del led back light R16 (Anexo C.10) conectado a la patilla 15 del dispositivo (ánodo del led de retro iluminación).

Para controlar el encendido y apagado del led retro iluminador o “back lighth” se coloca un transistor npn para abrir o cerrar el paso del cátodo del led hacia tierra, la corriente del led es de 110 mA según su fabricante (Anexo C.10), al ser la corriente del led la que circula por el colector del transistor, se elige al 2N3904 ya que según su fabricante éste valor de corriente no excede al máximo soportado por el elemento.

Al tener éste transistor una ganancia hfe de 300, en su base se tiene una corriente de:

$$hfe = \frac{Ic}{Ib} \quad \text{Ecuación 7}$$

Donde:

hfe = ganancia del transistor

Ic = intensidad de corriente del colector

Ib = intensidad o corriente de la base

Reemplazando datos se tiene:

$$300 = \frac{110 \text{ mA}}{Ib}$$

$$Ib = 0.366 \text{ mA}$$

Para garantizar que el transistor va a conmutar (entrar en saturación), tomando en cuenta que el dato calculado llega a 0.366 mA, se asume un valor mucho mayor del mismo, así para los cálculos posteriores $Ib = 1 \text{ mA}$ éste valor es menor que los 25 mA que generalmente se pueden drenar de una patilla de microcontrolador con lo que se garantiza un correcto funcionamiento.

Reemplazando éste valor en la Ecuación 2 se tiene:

$$Rb = \frac{V_{cc} - V_{ce}}{Ib}$$

Donde:

Rb = resistencia de base

Vcc = voltaje que suministra el micro controlador

Vce = voltaje colector emisor

Ib = intensidad de corriente que circula por la base del transistor

$$Rb = \frac{5\text{ V} - 0.6\text{ V}}{1\text{ mA}}$$

$$Rb = 4400\ \Omega$$

Normalizando el dato anterior se tiene que la resistencia que aparece como R17 en la Figura 2.41 y tiene un valor de:

$$Rb = 4.7\text{ K}\Omega$$

2.2.5.2 Lector de estados

En la Figura 2.42 se observa cómo se obtiene la lectura de un estado análogo generado por varios sensores externos del tipo digital, dicho estado es único para el caso de activación de cada uno de los sensores involucrados, pues los valores de resistencias conectados a cada sensor generan divisores de voltaje únicos para cada caso.

Se agrupan varios sensores para el lector de estados, así los botones “F1” que en el prototipo tiene la función de sacar la hoja presente en la bandeja de alimentación además de cancelar la impresión, los encoders ópticos asociados con el motor a pasos (rodillo) y con el “Fin de Carrera” (presencia de papel y fin de recorrido horizontal del carrete), son los estados de activación más importantes que se anexan a éste módulo.

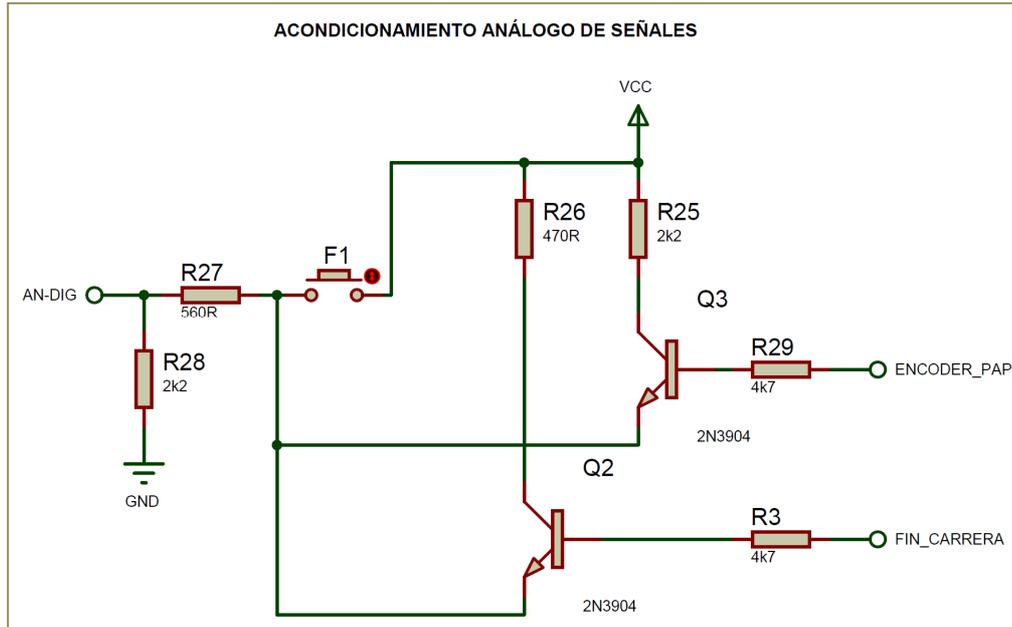


Figura 2.42 Esquema implementado acondicionamiento de señales en el lector de estados

2.2.5.2.1 Ejemplo de cálculo en el lector de estados

Se toma como patrón el cierre del sensor “Fin Carrera”, así se tiene:

En primer lugar se toma en cuenta el divisor de voltaje generado, obteniéndose en el punto AN-DIG lo siguiente:

$$V_{an} = \frac{V_{cc} * R_{28}}{(R_{26} + R_{27} + R_{28})} \quad \text{Ecuación 8}$$

Donde:

V_{an} = voltaje análogo en resistencia 28

V_{cc} = voltaje de alimentación

R = resistencia

Reemplazando valores se tiene:

$$V_{an} = \frac{5 \text{ V} * 2200 \ \Omega}{(470 + 560 + 2200) \ \Omega}$$

$$V_{an} = 3.41 \text{ V}$$

Finalmente se realiza la conversión de nivel de voltaje a valor digital considerando una resolución de conversor A/D de 8 bits, así:

$$Dg = \frac{Van}{Vcc} * 2^{Res} \quad \text{Ecuación 9}$$

Donde:

Dg =valor digital

Van =voltaje análogo

Vcc = voltaje de alimentación

Res =resolución

$$Dg = \frac{3.41 V}{5 V} * 2^8$$

$$Dg = 174.59$$

Del valor obtenido se toma en cuenta, según su aproximación, un dato entero por lo tanto:

$$Dg = 175$$

2.2.5.3 Comunicación

La interfaz de comunicación que se implementa es USB (Universal Serial Bus) en su versión 2.0, ya que la tendencia de los computadores personales es incluir solamente éste tipo de comunicación eliminando paulatinamente los conectores y comunicación del tipo paralelo y RS 232.

2.2.5.3.1 Interfaz Física [13]

La interfaz física está formada por cuatro hilos dos para la alimentación 5v (Rojo) GND (Negro) y dos para datos D+ (verde) y D- (Blanco), los signos + y – hacen referencia a una señal diferencial cuyo valor depende de la velocidad del bus (3,3 V para low-speed y 400 mV para high-speed por ejemplo).

Existen muchas clases de conectores físicos para la interfaz USB, sin embargo los más difundidos y los que se utilizan en el presente proyecto son los que se denominan de tipo A y B, una imagen de dichos conectores se pueden observar en las Figuras 2.43 y 2.44.



Figura 2.43 Conector USB tipo A

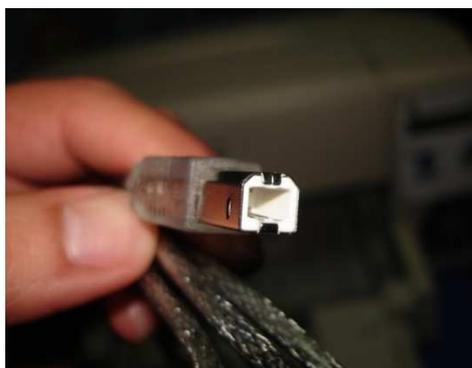


Figura 2.44 Conector USB tipo B

2.2.5.3.2 *Velocidades del Bus USB [13]*

El módulo denominado Host perteneciente a la PC es el que controla la velocidad en la que circulan los datos en el bus USB y que al ser un bus compartido, su velocidad real depende de la cantidad de dispositivos que se conecten a él en un momento determinado (máximo 127 incluyendo al Host [13]). Los datos siguientes son teóricos y de referencia.

- **Low speed:** 1,5 Mbps. Soportado por las especificaciones USB 1.1, 2.0 y 3.0. Es la velocidad utilizada por dispositivos como teclados, ratones, joystick, etc.
- **Full speed:** 12 Mbps. Soportado por USB 1.1, 2.0 y 3.0. Un ejemplo en donde se utilizan estas velocidades es en transmisiones de audio.
- **High speed:** 480 Mbps. Solamente USB 2.0 y 3.0. Ejemplo en transmisiones de video.
- **Super speed:** 5 Gbps solo soportado en dispositivos USB 3.0.

2.2.5.3.3 Tipos de transferencias en el Bus USB [13]

Cabe anotar que no existe un formato único de transferencia, la especificación del bus USB permite cuatro tipos de las mismas:

- **Control:** es utilizada para configurar y enviar comandos, por ejemplo en la enumeración del dispositivo.
- **Bulk (masivas):** es un tipo más rápido de transferencia, sin embargo no hay garantía de que los datos se transmitan en un tiempo determinado. Dispone de un sistema de corrección de errores (CRC) la misma que garantiza la llegada de todos los paquetes de datos enviados, esta transferencia solo la pueden utilizar dispositivos que soporten velocidades del tipo Full y High Speed. Éste tipo de transferencia es utilizada por dispositivos como discos duros, pen drivers, escaners, etc.
- **Isócronas:** es usada por dispositivos que transmiten señales de audio y de vídeo en tiempo real.
- **Interrupción:** por su tipo garantiza la verificación de que los datos se han transmitido con éxito. Se utiliza en dispositivos como: teclados, mouse, sensores, pantallas táctiles.

Los dispositivos que se comunican mediante el bus USB usan uno o más tipos de transferencias, sin embargo la de control es utilizada por todos los dispositivos en el proceso de enumeración en el que el Host es el encargado de detectar cualquier dispositivo que se conecta al bus. Cuando un dispositivo es detectado el Host necesita obtener información sobre él, a éste proceso es al que se le llama enumeración.

La información que necesita el Host se encuentra implementada en el dispositivo en los llamados descriptores. Los descriptores son datos que se guardan en la memoria no volátil del microcontrolador y contienen la siguiente información; el identificador del vendedor (VID) o vendor ID y del producto (PID) o product ID, la clase utilizada, entre otros datos [12].

2.2.5.3.4 Clases de dispositivos USB [13]

Las clases de dispositivos más utilizadas con microcontroladores son:

- **HID (Human Interface Device):** ejemplos de dispositivos que utilizan esta clase son: teclados, ratones, pantallas táctiles, joystick, etc. Velocidad low-speed (64 KB/s de velocidad máxima), tipos de transferencias soportadas: de control y de Interrupción. Una de las características interesantes al utilizar esta clase es que no se necesita instalar un driver específico en el sistema operativo para poder utilizar el dispositivo que maneje esta clase, se utiliza uno estándar que ya está incluido en el sistema.
- **MSD (Mass Storage Device Class):** como su nombre indica son dispositivos de almacenamiento masivo como discos duros, memorias flash, cámaras digitales, dispositivos ópticos externos como lectores y grabadoras de CD, DVD, etc. Esta clase solo se utilizan en dispositivos que soporten Full y High Speed. El tipo de transferencias utilizadas es Bulk o una combinación formada por transferencias del tipo Control, Bulk e Interrupt. No se necesita la instalación de un driver específico, esta clase utiliza drivers genéricos instalados ya en los sistemas operativos.
- **CDC (Communications Device Class):** Un ejemplo de dispositivo que utiliza esta clase son los Modems, en éste proyecto se utiliza esta clase para comunicar el prototipo con la PC. La velocidad máxima al utilizar esta clase será de 80 Kb por segundo y el tipo de transferencias soportadas son Interrupción y Bulk. Utiliza también un driver específico proporcionado para esta clase, en éste se puede modificar los strings de presentación ya que los identificadores no se recomienda realizar cambios debido a que pueden existir otros dispositivos que comparten los datos de VID y PID produciéndose un mal funcionamiento del dispositivo desarrollado.

Cuando se utiliza esta clase de dispositivo en la PC se crea un puerto serie virtual y la comunicación entre el dispositivo y la aplicación de escritorio se realiza a través de éste al igual que se haría con un puerto serie físico, esto supone una gran ventaja a la hora de diseñar la aplicación de escritorio, ya que el lenguaje de programación visual que se usa dispone de un componente o librería que permite el acceso al puerto serie.

- **CUSTOM CLASS:** existe una clase genérica llamada "**Custom Class**" que se utiliza cuando el dispositivo desarrollado o a desarrollarse no se asemeja a las características de ninguno de los miembros pertenecientes a otras clases.

2.2.5.4 Microcontrolador

En éste punto se analiza las diferentes posibilidades de elección en cuanto a empresas fabricantes de microcontroladores, para determinar la opción más adecuada se realiza un estudio de periféricos que consta en la Tabla 2.1:

Tabla 2.1 Requerimientos de periféricos totales del sistema

PERIFÉRICO	# DE PINES
Comunicación USB	2
Motor a Pasos (Rodillo)	4
Encoder Motor a pasos	1
Motor DC (Carrete)	2
Fin de carrera óptico	1
Encoder Motor DC	1
Motores DC (Punzones)	3
Fin de carrera mecánico (Verifica punzón)	1
Teclado análogo	1
LCD	8
Reproductor MP3	4
TOTAL	28

De la tabla anterior se nota que el sistema requiere 28 pines de uso, entre los cuales deberán existir al menos un convertidor análogo a digital, y proveer de emulación o módulo interno que se encargue de la comunicación USB de clase CDC.

Con esta premisa y ya que en el mercado local existen dos de los principales fabricantes de microcontroladores como son: Microchip y Atmel que en gamas altas de sus productos satisfacen las necesidades del proyecto, se toma la decisión de participar de los productos Microchip específicamente en su serie 18Fxxxx que consta con un módulo de comunicación USB y por supuesto con canales análogo digitales configurables. Otra razón para elegir éste fabricante es

que provee una versión gratuita, tanto software como firmware, de grabador vía USB y los elementos sugeridos en el esquema del mismo son fáciles de conseguir en el mercado local.

Antes de proceder a especificar el modelo de microcontrolador es conveniente realizar la siguiente tentativa de utilización de pines, así:

Tabla 2.2 Tipos de pines requeridos en el sistema

PERIFÉRICOS	PINES			
	ENTRADA	SALIDA	A/D	DIFERENCIAL
Comunicación USB				2
Motor a Pasos (Rodillo)		4		
Encoder Motor a pasos	1			
Motor DC (Carrete)		2		
Fin de carrera óptico	1			
Encoder Motor DC	1			
Solenoides (Punzón)		1		
Fin de carrera mecánico	1			
Teclado análogo			1	
LCD		8		
Reproductor MP3		4		
TOTAL	4	19	1	2

De la gama 18Fxxxx de microcontroladores específicamente se elige el modelo 18F4550 que en particular posee 40 pines, esto permite prescindir trabajar con modo extendido ya que éste sistema es más costoso por los elementos electrónicos que se requiere utilizar.

En la Figura 2.45 se observa la distribución de pines y en la Tabla 2.3 los recursos utilizados.

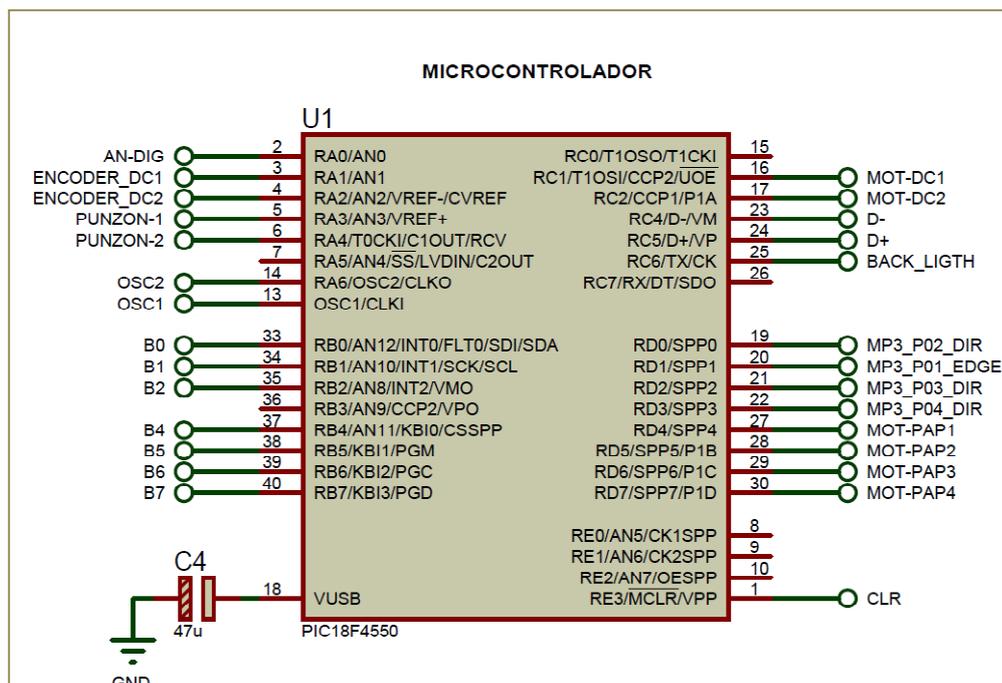


Figura 2.45 Distribución de pines en el microcontrolador 18F4550

En la patilla 18, que cumple con la función VUSB (voltaje de operación USB) es necesario colocar un condensador electrolítico comprendido entre 4.7 uF a 47 uF pues según su fabricante si se utiliza el módulo de comunicación USB es necesario completar un circuito interno que regula el voltaje que emplea la interfaz USB 2.0 presente en el microcontrolador de aproximadamente 3,3 Vdc. [12].

Tabla 2.3 Recursos utilizados del microcontrolador 18F4550

DESCRIPCIÓN	PINES		
	NÚMERO	ASIGNACION	TIPO
Alimentación	4	Vss - Vdd	Entrada
MCLR	1	RE3	Entrada
Oscilación	2	OSC1-OSC2	Entrada
Comunicación USB	2	RC4-RC5	Diferencial
Motor a Pasos (Rodillo)	4	RD4-RD7	Salida
Encoder Motor a pasos	1	RA0	Entrada
Motor DC (Carrete)	2	RC1-RC2	Entrada
Fin de carrera óptico	1	RA0	Entrada
Encoder Motor DC	1	RA1-RA2	Entrada

Solenoide (Punzón)	1	RA3-RA4	Salida
Fin de carrera mecánico	1	RC6	Entrada
Entrada análoga	1	RA0	A/D
LCD	8	RB0-RB7	Salida
Reproductor MP3	4	RD0-RD3	Salida
TOTAL	33		

2.2.5.5 Modulo ICSP

El módulo ICSP (In Circuit Serial Programming) propio de Microchip está pensado para desarrollar dispositivos y grabarlos sin sacarlos del circuito impreso o del sistema del que forman parte, de esta manera no es necesario desmontar los sistemas para grabarlos o actualizarlos.

Está implementado siguiendo las recomendaciones que da el fabricante [12], se construye con un jack del tipo RJ45 en la placa principal debido a su fácil manejo y reducidas dimensiones, en la Figura 2.46 se puede observar el circuito implementado.

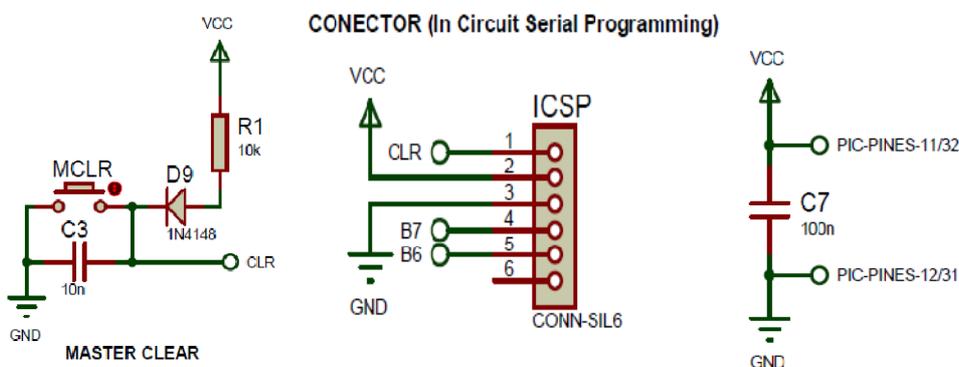


Figura 2.46 Esquema implementado "ICSP", tomado de [12]

2.2.6 FUENTE DE ALIMENTACIÓN

Todos los periféricos que componen el módulo Impresora Braille, así como su tarjeta de control funcionan con voltajes de corriente continua, sin embargo los requerimientos en cuanto a niveles de corriente y valores de voltaje difieren según el caso, así:

Tabla 2.4 Requerimientos ideales de voltaje y corriente

Módulo	Voltaje [V dc]	Corriente [mA]
Control motor a pasos	5	500
	12	800
Control de motor DC	5	500
	12	600
Control de punzón	5	200
	12	10
	24	4260
Amplificador de audio	12	210
Microcontrolador	5	400

De la Tabla 2.4 se evidencia que ciertos módulos comparten requerimientos en cuanto a voltaje pero difiriendo en cuanto a su consumo de corriente, específicamente estos módulos son: “Control motor a pasos”, “Control de motor DC” y “Control de Punzones”, por lo que se toma en cuenta el mayor consumo de corriente de los mismos ya que en su funcionamiento del prototipo “Impresora Braille” solamente requiere que uno de los tres módulos se encuentre en funcionamiento a la vez.

Una vez realizada esta consideración y tomando en cuenta un factor de seguridad del 10% en la corriente se tienen los valores evidenciados en la Tabla 2.5.

Tabla 2.5 Requerimientos finales de voltaje y corriente

Fuente	Voltaje [V dc]	Corriente [mA]
Motores a pasos, DC	12	880
Punzones	12	880
	24	4686
Amplificador de audio	12	231
Tarjeta de control (microcontrolador)	5	550

Según los datos especificados en la Tabla 2.5 el máximo requerimiento en cuanto a corriente es de aproximadamente 5 A., debido a esto y a los valores de voltaje normalizados presentes se requiere una fuente tipo ATX de computador, de la cual se aprovechan los valores presentes en sus terminales estabilizados y al ser una fuente conmutada los servicios de corriente que provee están dentro de los valores requeridos por el prototipo impresora Braille, en la Figura 2.47 se muestra un esquema de la utilización de los terminales de una fuente ATX.

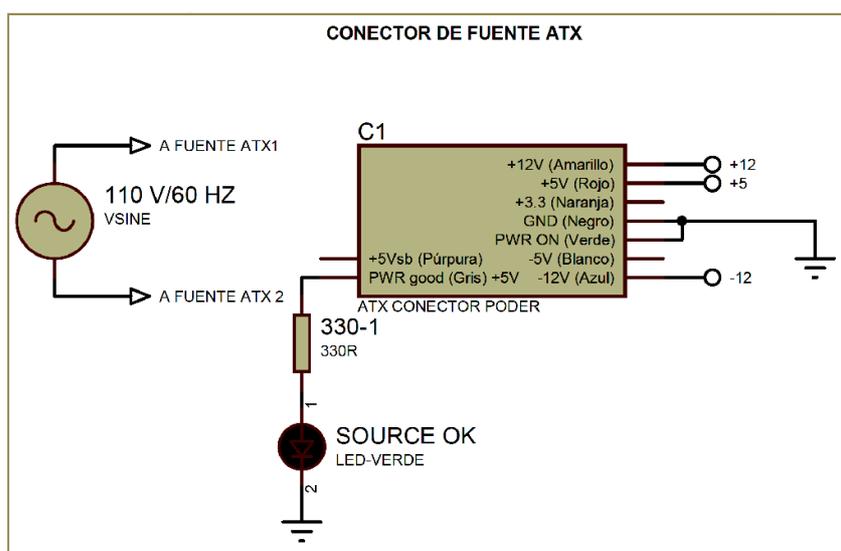


Figura 2.47 Esquema de fuente de alimentación ATX

Cabe mencionar que los valores normalizados que provee una fuente de éste tipo son: -12 Vdc., +12 Vdc., +5 Vdc., +3,3 Vdc., en modelos de años anteriores también se encuentra el valor de -5 Vdc. sin embargo éste último ya no es implementado en fuentes construidas en la actualidad.

En cuanto a consumo de corriente todos los niveles de voltaje proveen la misma cantidad de potencia, por lo que se puede combinar niveles de voltaje sumándolos con el cuidado de no colocar el mismo nivel de referencia para alimentar los diferentes módulos, en la Figura 2.48 se aprecian las conexiones de los niveles de voltaje.

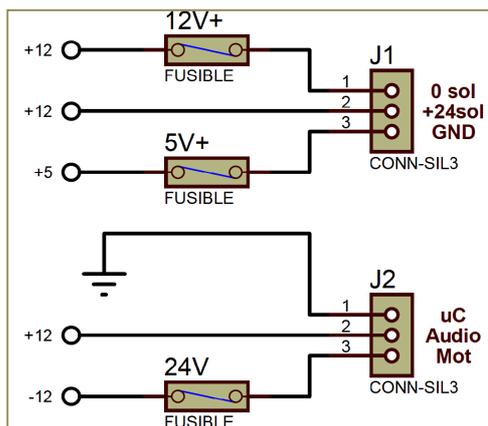


Figura 2.48 Esquema de conexión de fuente de alimentación

En los conectores marcados como J1 y J2 se aprecian los niveles de voltaje y los dispositivos a los cuales se conectan los mismos, como se observa en algunos casos es necesario colocar fusibles de protección pues debido a que puede drenar una cantidad considerable de corriente antes de detectar una falla por cortocircuito es necesario implementar esta protección adicional.

CAPITULO 3

DESARROLLO DEL PROGRAMA DE CONTROL

En éste capítulo se abordan las diferentes etapas de la programación desarrollada para el correcto funcionamiento del prototipo “Impresora Braille”, se expone además la configuración general de la interfaz gráfica para la comunicación entre el usuario y el módulo. Para éste propósito se utiliza el programa de interfaz gráfica LabVIEW en su versión 10, en el cual se implementa una pantalla donde el usuario pueda seleccionar el documento con extensión .doc a ser impreso.

Para la ejecución de la impresión en relieve y la comunicación con la PC, se utiliza el microcontrolador 18f4550 cuyo programa es desarrollado en el lenguaje PIC C versión 2.44.

3.1 DESCRIPCIÓN GENERAL

La programación tiene dos partes, las mismas que trabajarán conjuntamente a través de la comunicación USB, la primera está desarrollada en el lenguaje de programación PIC C para el microcontrolador 18f4550, éste programa contiene subrutinas que permiten el manejo de motores para posicionar la hoja a ser impresa, desarrollándose además las subrutinas para la impresión mecánica y otra para la adquisición de datos a través del puerto USB en configuración CDC (Communication Device Class o dispositivo de comunicación por emulación de trama serial).

La segunda parte es la implementación y programación de la interfaz visual de usuario en la PC, éste programa se realiza en el entorno LabVIEW en su versión 10 y consta de una pantalla con dos partes principales; una para el documento a ser impreso y otra con su equivalencia o traducción gráfica en Braille. En el mismo entorno se realiza la preparación del archivo para ser enviado al módulo impresora por partes (líneas con un número definido de caracteres), se realiza

una subrutina que mantiene “vivo” el puerto de comunicación USB para el envío de datos de control y los caracteres que conforman el documento a ser impreso.

El algoritmo que se implementa para la utilización del prototipo al realizar una impresión posee la siguiente secuencia:

- A. Encender la impresora.
- B. Diagnóstico de estado de prototipo y ejecución de mensaje de voz en el que se escucha “Sistema encendido, realizando diagnóstico”.
- C. Inicio de la comunicación USB con la PC.
- D. Implementación de transmisión y recepción de información hasta que todos los caracteres que forman el documento hayan sido enviados.
- E. Cada treinta caracteres se procede a la impresión de los mismos en una fila acompañados tanto al inicio como al final de caracteres o comandos de control.
- F. Al finalizar la impresión se ejecuta un mensaje de voz “Impresión Terminada”, todas las variables involucradas retornan a sus valores iniciales.
- G. La hoja presente en el rodillo sale completamente del mismo terminando el proceso.

En los siguientes puntos se detallan los programas desarrollados tanto en el microcontrolador como en la PC.

3.2 PROGRAMACIÓN EN EL MICROCONTROLADOR

Éste programa recibe los datos que se envían a través del puerto USB, los mismos que son almacenados y comparados con una tabla implementada, esta tabla contiene las equivalencias en puntos en relieve de todos y cada uno de los caracteres reconocidos en idioma castellano por las normativas Braille, para luego ser impresos respetando sus distancias de separación, mismas que también las dicta esta normativa; para éste proceso se utilizan tres motores uno a pasos y dos motores de DC.

Para la parte de la impresión mecánica se diseña un punzón que usa un principio de percusión en base a un solenoide como se indica en el capítulo anterior.

Esta primera parte de la programación o implementación del control del prototipo se divide en las siguientes fases:

- A. Reproducción de Voz.
- B. Comunicación USB.
- C. Posicionamiento de la hoja.
- D. Impresión mecánica.

3.2.1 REPRODUCCION DE VOZ

En esta fase se controla un módulo de reproducción de voz en formato MP3, el mismo que permite grabar varios mensajes de ayuda al usuario al que está dirigido éste proyecto.

3.2.1.1 Grabación de mensajes

- A. Se graban los mensajes de voz requeridos en formato MP3, para esto se utiliza cualquier medio que permita grabar voz en cualquier computadora por su entrada auxiliar de micrófono. Para el presente proyecto en particular se utiliza el programa “a Tube Catcher” (Figura 3.1)

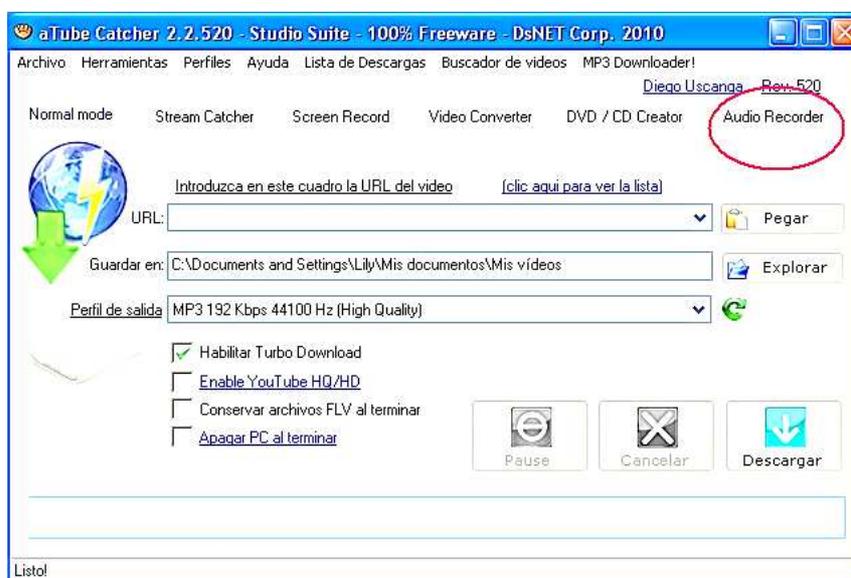


Figura 3.1 Pantalla Principal del programa “a Tube Catcher”

- B. Una vez grabados los mensajes requeridos se transfieren a una carpeta con el nombre de “advert00”, esta especificación la da el fabricante,

posteriormente se graba en una memoria de tipo SD, la misma que será leída por el reproductor.

El formato para especificar la dirección de cada mensaje que el fabricante señala es el siguiente:

00X.mp3

Donde X es el número en orden de reproducción del mensaje a amplificar.

Para éste proyecto en específico los mensajes grabados se detallan en la Tabla 3.1.

Tabla 3.1 Mensajes de voz grabados en el reproductor MP3

Item.	Dirección	Descripción
1	001.MP3	Sistema encendido, realizando diagnóstico
2	002.MP3	La computadora no está conectada
3	003.MP3	Conexión USB realizada
4	004.MP3	No existe papel en bandeja
5	005.MP3	Comenzando impresión
6	006.MP3	Impresión terminada

3.2.1.2 Control de reproducción de mensajes grabados

El control de reproducción de los mensajes previamente grabados se lo realiza mediante una interfaz paralela en la cual 3 bits determinan la dirección del mensaje a interpretar y 1 bit determina el control de reproducir / detener mediante un flanco negativo.

Se reproduce un mensaje cuando ocurren eventos específicos como los indicados en la tabla anterior, dichos eventos se citan más adelante, en el detalle de las diferentes subrutinas de programación.

3.2.2 COMUNICACIÓN USB

En esta fase el microcontrolador envía continuamente sus identificadores a través del puerto USB. Estos identificadores se componen de dos números Hexadecimales, los mismos que provee Microchip para que la PC identifique la clase de dispositivo que se está conectando.

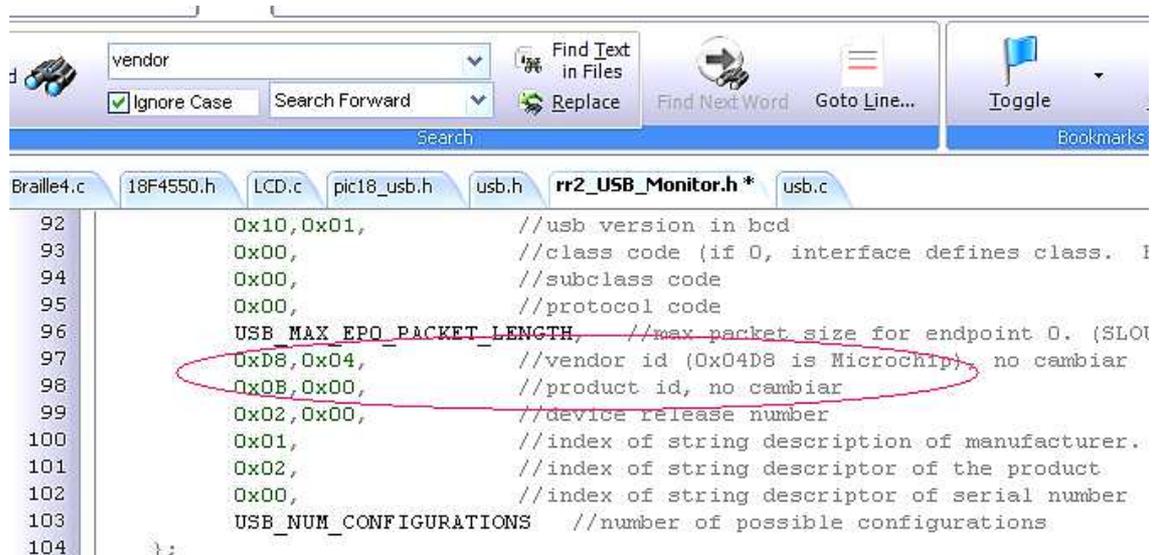
Para el presente proyecto lo que detecta la PC es; que se encuentra conectado un prototipo de investigación que utiliza un microcontrolador microchip de la familia 18fXXXX.

Los especificadores o identificadores provistos para éste microcontrolador y utilizados por éste prototipo son los siguientes:

Vendor ID (Identificador del Fabricante) = 0h04D8

Product ID (Identificador del producto realizado) = 0h000B

Para utilizar estos identificadores en el lenguaje de programación PIC C es necesario que los mismos sean separados en dos bytes y a continuación escritos intercambiando bytes superiores por inferiores como se muestra en la Figura 3.2.



```

92     0x10,0x01,        //usb version in bcd
93     0x00,            //class code (if 0, interface defines class. I
94     0x00,            //subclass code
95     0x00,            //protocol code
96     USB_MAX_EP0_PACKET_LENGTH, //max packet size for endpoint 0. (SLOT
97     0xD8,0x04,        //vendor id (0x04D8 is Microchip), no cambiar
98     0x0B,0x00,        //product id, no cambiar
99     0x02,0x00,        //device release number
100    0x01,            //index of string description of manufacturer.
101    0x02,            //index of string descriptor of the product.
102    0x00,            //index of string descriptor of serial number
103    USB_NUM_CONFIGURATIONS //number of possible configurations
104    };
  
```

Figura 3.2 Formato de los identificadores de Microchip en PIC C para la serie 18FXXXX.

El cambio anteriormente descrito se realiza en la librería de PIC C implementada para realizar la identificación de la comunicación, a esta librería se la conoce con el nombre de “rr2_USB_Monitor.h”, cabe mencionar que la misma es desarrollada por PIC C y es modificable según los requerimientos del proyecto.

3.2.2.1 Algoritmos de Comunicación USB

3.2.2.1.1 Inicialización

La inicialización del puerto de comunicación consiste en abrir el mismo y mantenerlo “vivo” en el microcontrolador, para lo cual se utilizan las librerías usb.c y usb.h presentes en el compilador PIC C.

Una vez abierto el puerto de comunicación, el microcontrolador envía a la PC los identificadores Vendor ID y Product ID.

El microcontrolador espera que la PC le envíe la numeración del puerto, si esto no sucede el prototipo emite un mensaje de voz; “La computadora no está conectada”.

Una vez identificado el módulo envía una palabra o string de identificación propia del prototipo, esto es “Tesis, Impresora Braille”, esto indica que el dispositivo está conectado y listo para ser utilizado.

Luego la PC enumera un puerto por el cual se va a transmitir la información y se ejecuta un lazo de espera hasta tener una interrupción generada por la transmisión de un dato desde la PC.

En la Figura 3.3 se presenta el diagrama de flujo correspondiente a la inicialización de la comunicación USB.

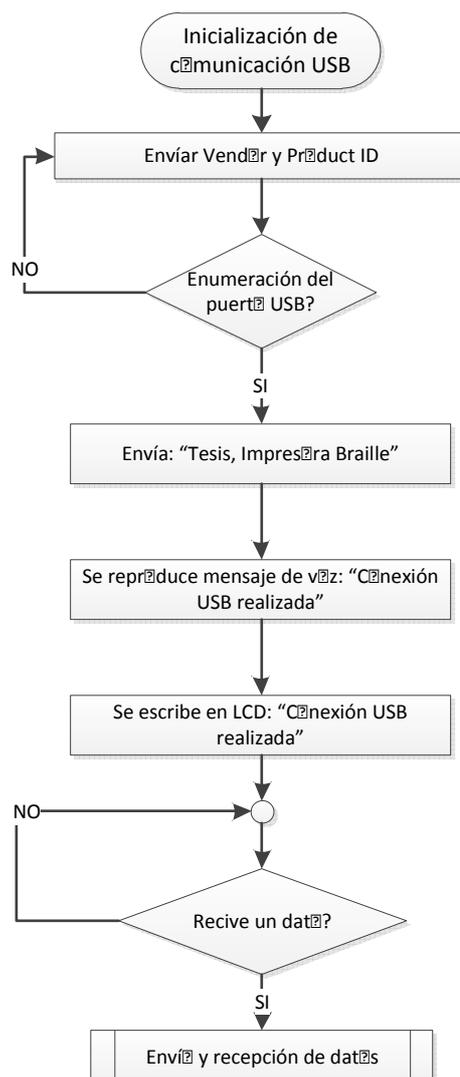


Figura 3.3 Inicialización de la comunicación USB

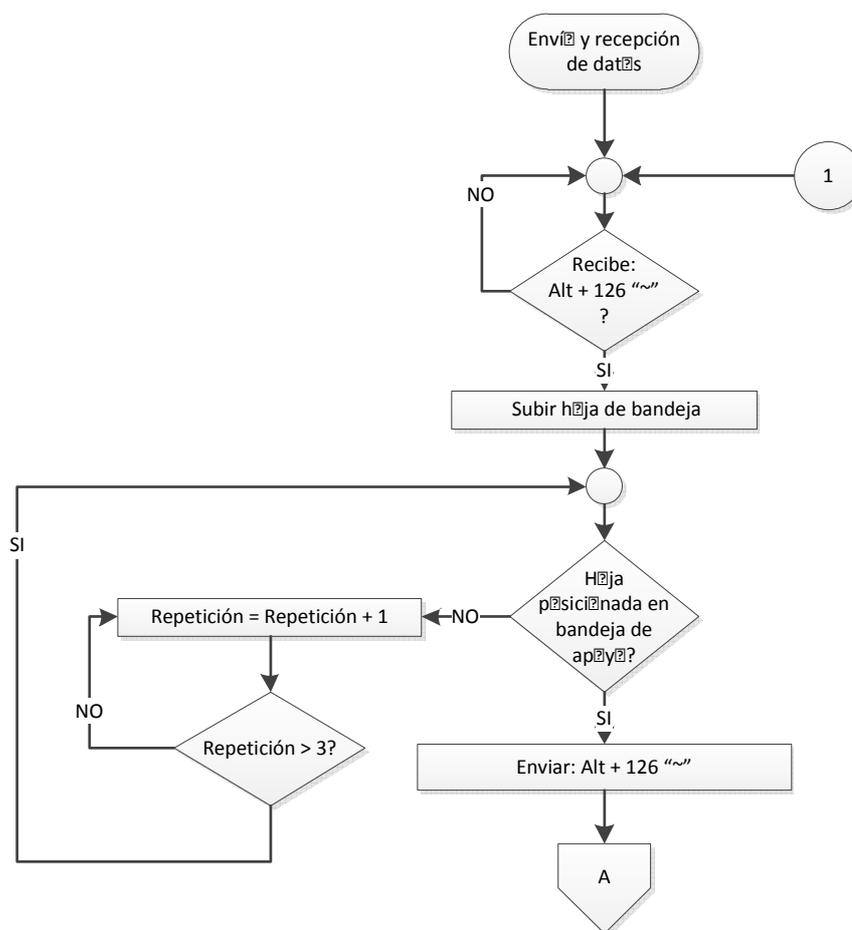
3.2.2.1.2 Envío y recepción de datos

Al momento en que la PC envía un dato (caracter tipo char) en el microcontrolador se genera una interrupción por recepción.

El primer dato recibido (Alt + 126 "~") genera un eco del mismo para confirmar que se tiene una orden de impresión, éste eco se produce exactamente después de ejecutar las acciones correspondientes para subir la hoja presente en la bandeja de alimentación, de no existir papel en bandeja se reproduce el mensaje correspondiente y se intenta repetir éste proceso hasta encontrar un papel en el que se realizará una impresión. Tanto los datos subsecuentes como el primer dato se almacenan en una localidad de memoria para luego ser comparados con una

base de datos, la cual asigna el caracter al que corresponde en código Braille, éste procedimiento se repite hasta completar el número de elementos correspondientes a una fila (30 caracteres), una vez completa la impresión mecánica se procede a enviar un caracter de control, en éste caso la letra “F” que indica en la PC que el prototipo se encuentra listo para recibir los siguientes caracteres.

El procedimiento se repite hasta completar todas las filas que conforman una página (22 filas), para finalizar la impresión el microcontrolador espera un caracter de control (Alt + 175 “»”) que le informa que no existen más datos a transmitir como se indica en la Figura 3.4 en la que se muestra el diagrama de flujo de éste subproceso.



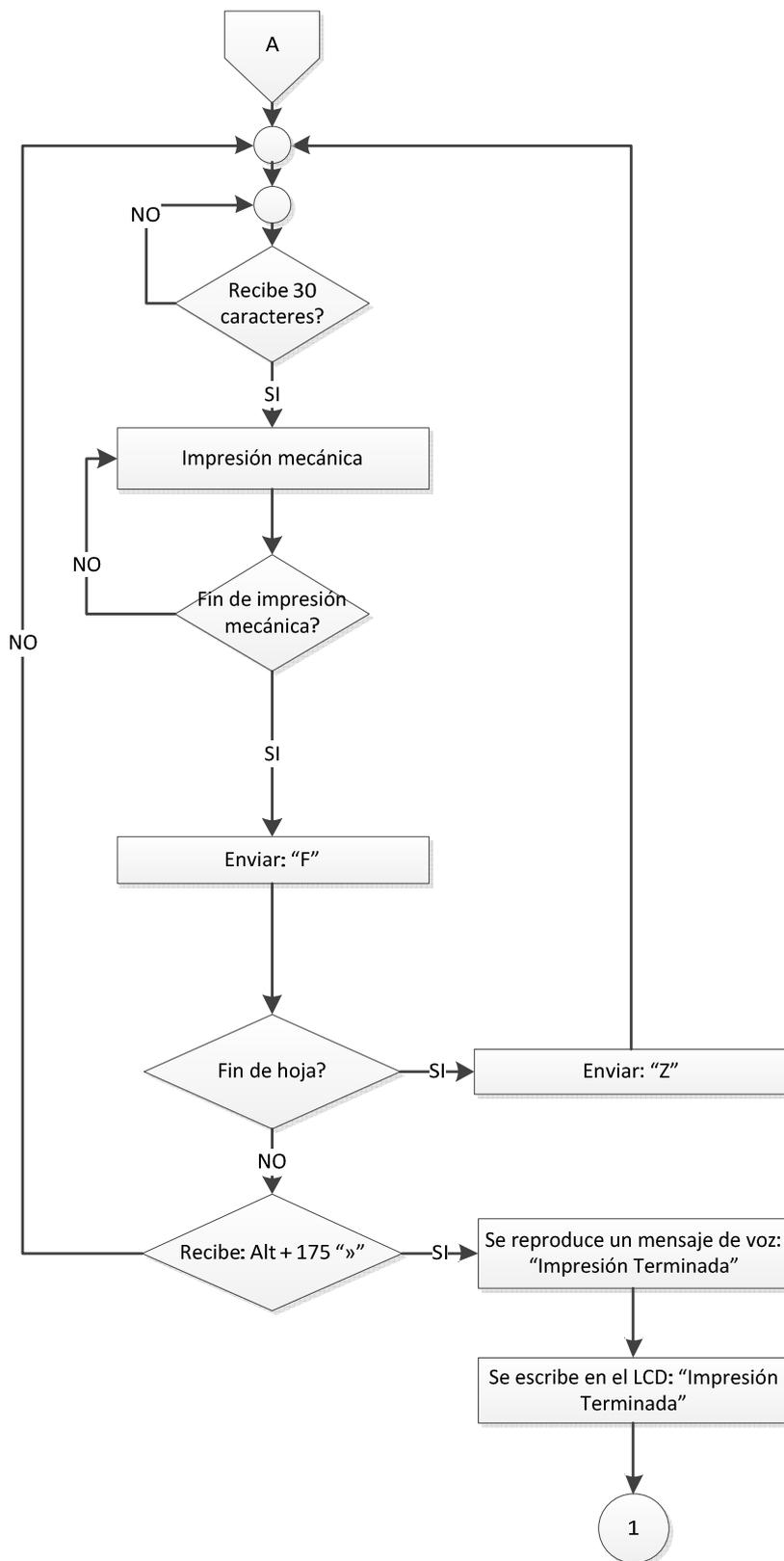


Figura 3.4 Diagrama de flujo de envío y recepción de datos entre la PC y el microcontrolador

3.2.3 POSICIONAMIENTO DE LA HOJA

Para el posicionamiento de la hoja se requiere el control de dos motores, un motor de DC que permite el movimiento del carrete, mismo que acciona un dispositivo mecánico que enclava la parte móvil de la bandeja al engranaje principal del rodillo, lo que posibilita que esta suba y por la acción del rodillo, la hoja es atraída hacia el rodillo, esta aproximación de la hoja se consigue por el acople de engranajes presentes en el rodillo, los mismos que se observan en la Figura 3.5.

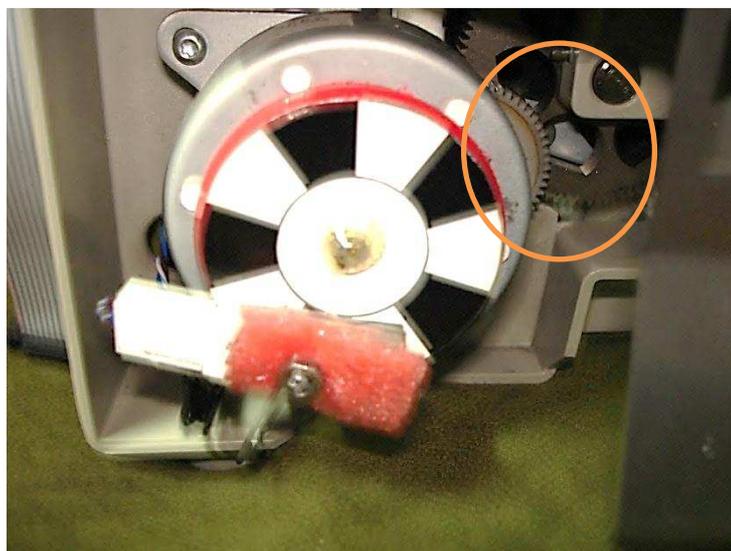


Figura 3.5 Acople mecánico para levantar la bandeja

El segundo motor (a pasos) controla el movimiento rotacional del rodillo para que la hoja pueda moverse conjuntamente con el mismo proporcionando así una ubicación adecuada de la hoja para proceder a realizar la impresión.

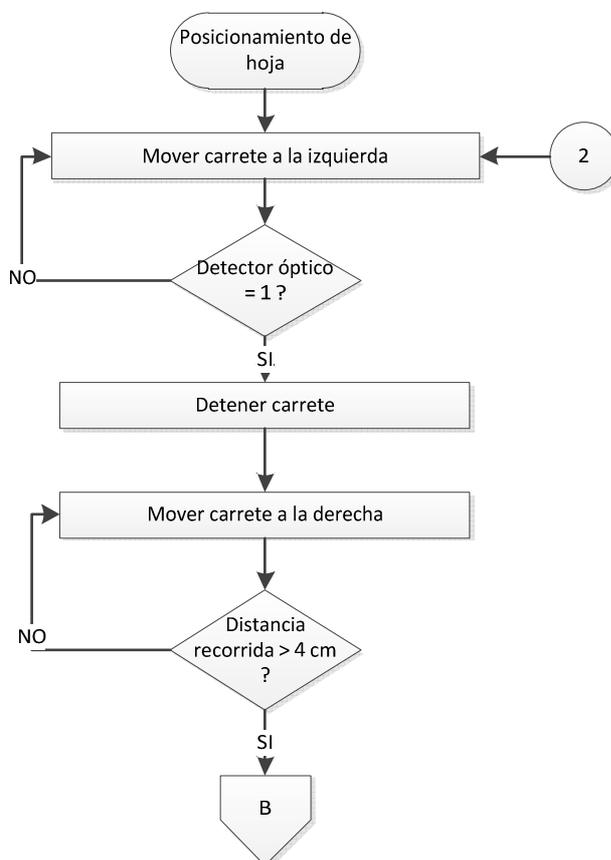
El motor de DC viaja lentamente hacia la posición extrema izquierda hasta que el fin de carrera óptico genere una señal para detener su movimiento.

A continuación el mismo motor viaja una distancia de aproximadamente cuatro centímetros en dirección derecha, al llegar a esa posición se detiene y regresa al extremo izquierdo rápidamente asegurando que el mecanismo enganche los dos engranajes (el de rotación del rodillo y el de subida de la bandeja), para mantener esta posición se genera una señal de PWM de 500 hz con un ancho de pulso en alto del 10 % de su totalidad para evitar que el mecanismo se desacople, luego se

mueve el motor a pasos para que la hoja ingrese a medida que se mueva el rodillo, con esto se logra que se produzca un levantamiento de la bandeja y la entrada de la hoja en el rodillo, posteriormente el motor de DC se mueve 4 cm hacia la derecha, éste movimiento desengancha el acople del levantamiento de la bandeja.

El movimiento del motor a pasos seguirá hasta que el encoder óptico situado en el carrete detecte la presencia del papel, una vez generada esta señal (1 lógico) el movimiento del motor a pasos continúa hasta que exista entre el filo del papel y la punta del punzón 2 cm.

El motor de DC produce un movimiento del carrete hacia la izquierda hasta cuando se pierda la señal generada por el fin de carrera óptico, se detecta así el extremo izquierdo del papel, luego el detector de grilla posterior informa al microcontrolador cuando llega el inicio de los treinta caracteres a imprimir, con esto el carrete se encuentra listo para realizar la impresión de los 30 caracteres que componen una fila. El procedimiento detallado anteriormente se puede observar en el siguiente diagrama de flujo (Figura 3.6).



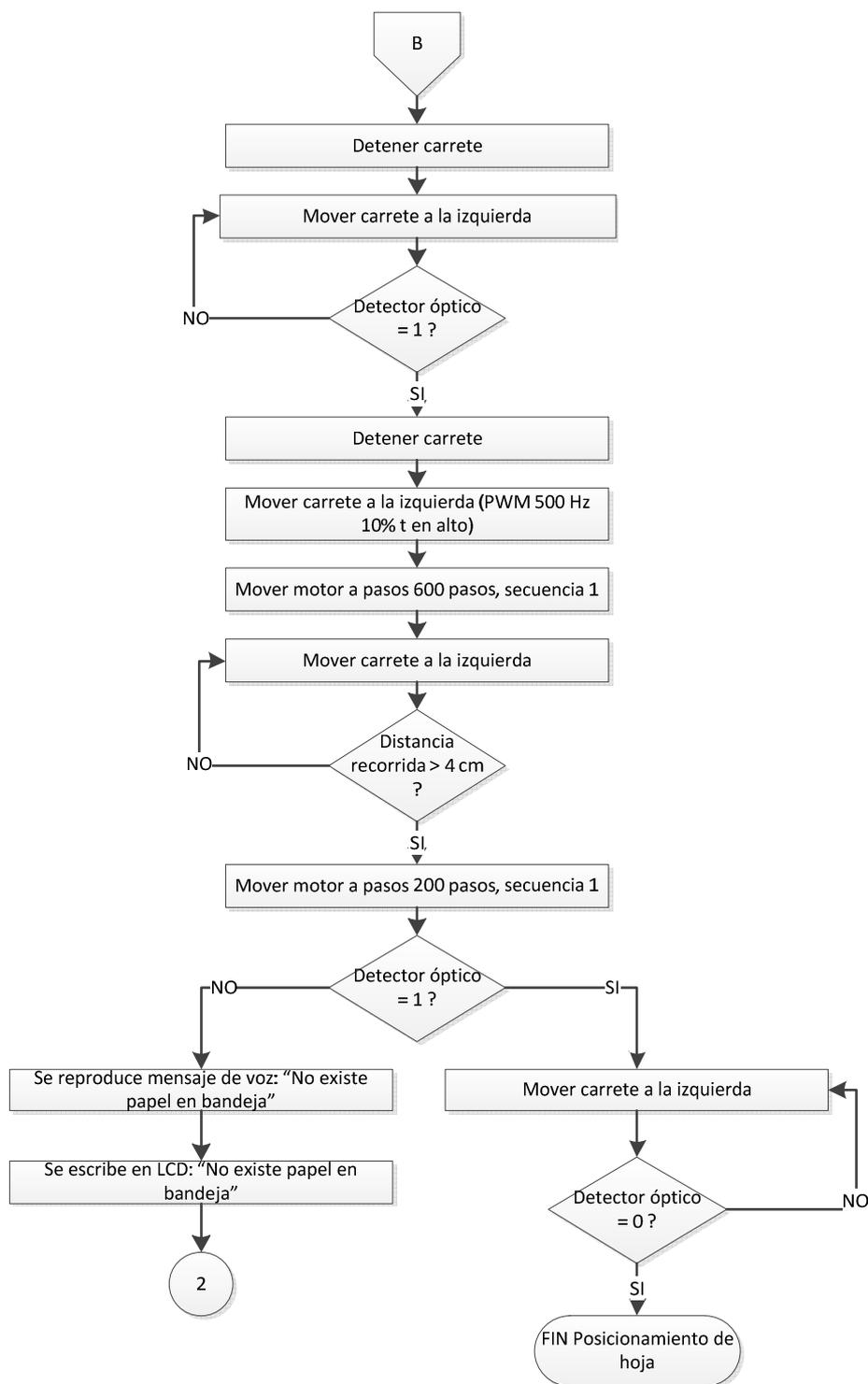


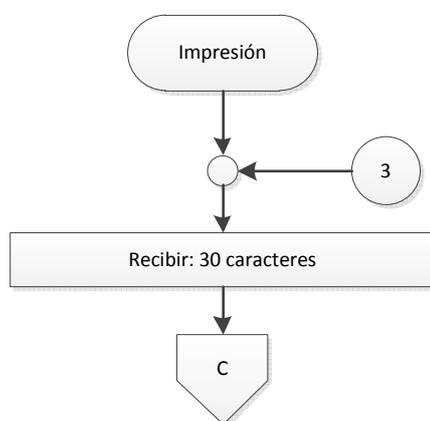
Figura 3.6 Diagrama de Flujo del posicionamiento de la hoja a ser impresa.

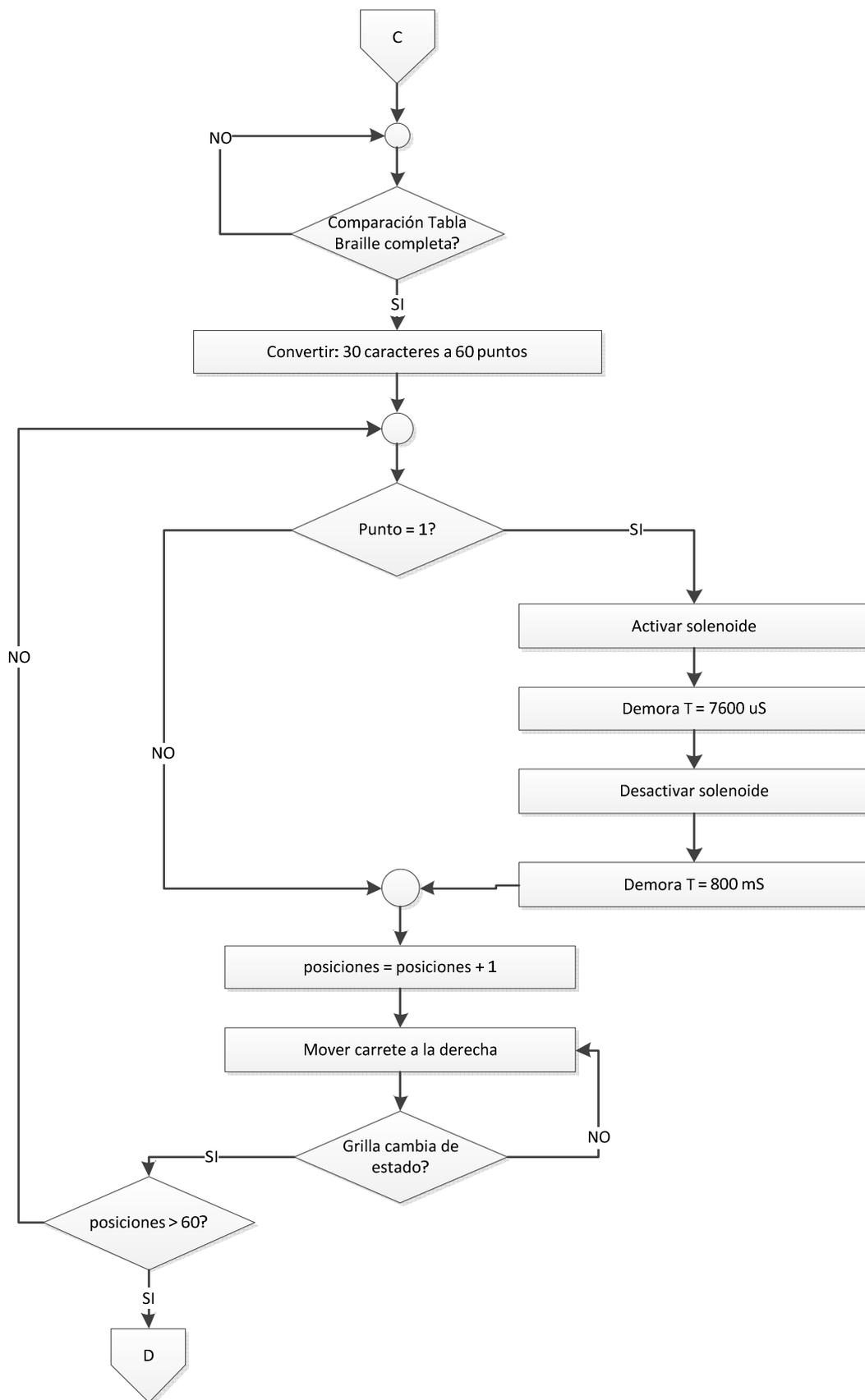
3.2.4 IMPRESIÓN MECÁNICA

Los datos previamente almacenados son comparados uno a uno con una base de datos, al determinar a qué carácter corresponde se asigna los puntos que deberán punzarse para obtener así el relieve de la impresión en código Braille de cada carácter.

Una vez que se establecen los puntos a ser punzados comienza el movimiento del carrete que aloja el punzón, éste movimiento se produce en el eje horizontal respetando las distancias determinadas por la grilla posterior con la finalidad de posicionar el punzón en el sitio exacto donde se marcará el punto que forma parte del carácter Braille, después de haber realizado un recorrido completo, es decir de reconocer 60 espacios el carrete regresa a su posición extrema izquierda para luego continuar con las 2 filas de puntos restantes y de esta forma completar una fila de 30 caracteres, luego de los cuales el microcontrolador envía a través del puerto USB el carácter "F", esperando recibir los siguientes 30 a ser impresos.

Éste proceso se repite hasta completar 22 filas que componen la hoja, luego de lo cual se envía otro carácter a la PC para informar, que si el documento no ha terminado, se repita todo el proceso anterior después de posicionar una nueva hoja, si por el contrario se recibe un carácter de confirmación de terminación de documento el motor a pasos se mueve provocando que la hoja existente salga del mismo y se reproduzca un mensaje de voz informando que el proceso de impresión a terminado. En la Figura 3.7 se muestra el diagrama de flujo correspondiente a la subrutina de impresión.





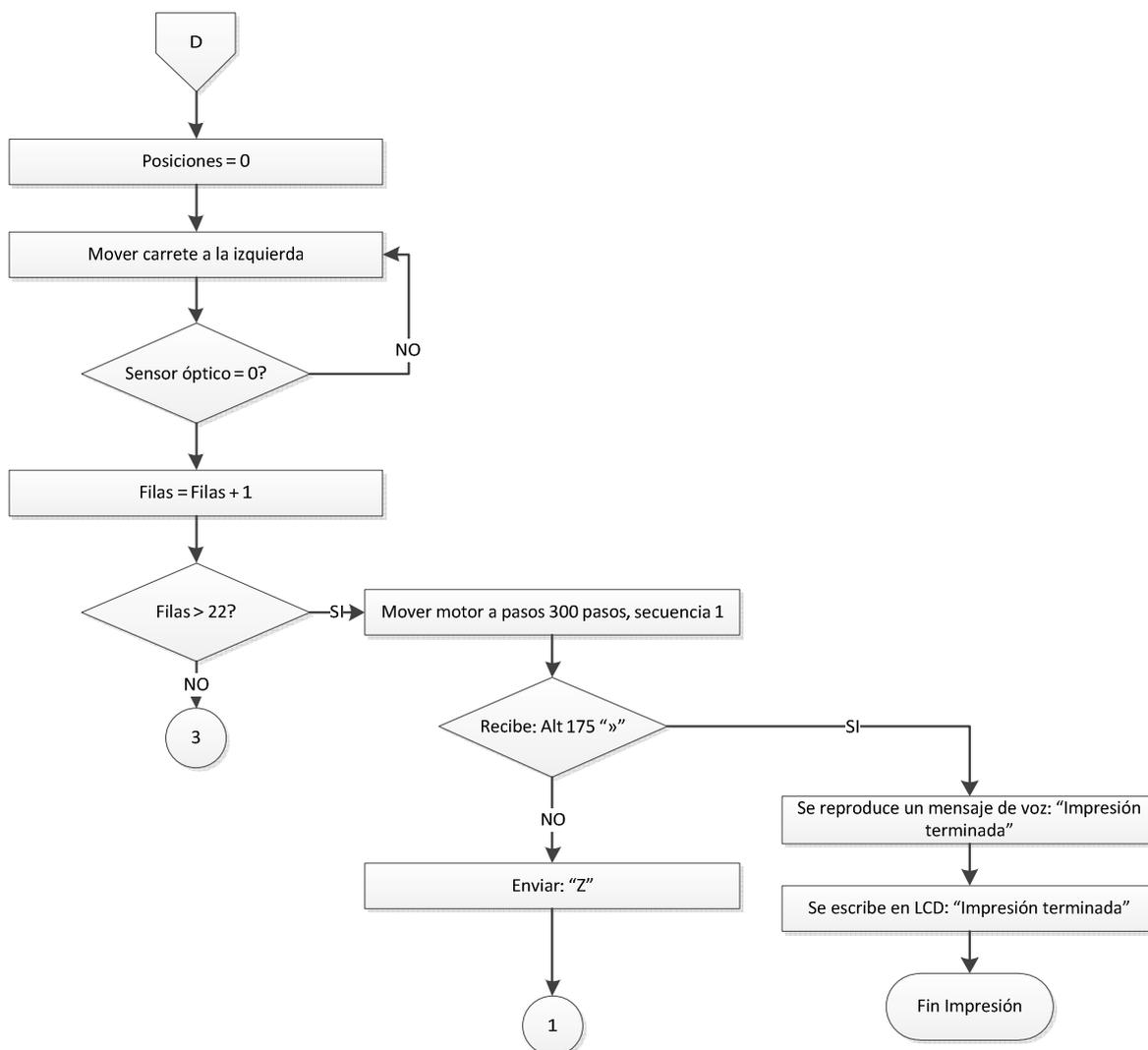


Figura 3.7 Diagrama de flujo del subproceso “impresión” en código Braille

3.3 PROGRAMACIÓN DE LA INTERFAZ DE USUARIO

Esta instancia de la programación permite la comunicación del usuario con el módulo “Impresora Braille”, consta de una pantalla donde se escoge el archivo a ser impreso, y una vez elegido el documento se reescribe adecuando el mismo a las reglas de la escritura Braille para luego traducir el nuevo documento a un tipo de Font diseñado en base a la combinación de puntos a los que cada caracter corresponde en la normativa Braille.

Luego de ser reescrito se procede a enviar el documento fragmentándolo en partes de 30 en 30 caracteres que formaran las diferentes filas de la impresión.

Esta programación consta de dos partes fundamentales, las mismas que se detallaran a continuación:

3.3.1 PANTALLA

Se implementa una pantalla que permite al usuario buscar el archivo a ser impreso y en la que se observa cómo se traduce el mismo a puntos negros que representan los relieves propios del código Braille; además de escoger el número del puerto por el cual se produce la comunicación, así como indicadores del tratamiento del documento a ser enviado mediante comunicación USB, en la Figura 3.8 se detallan las partes que componen la pantalla principal del programa desarrollado en la PC a través de LabVIEW 10.

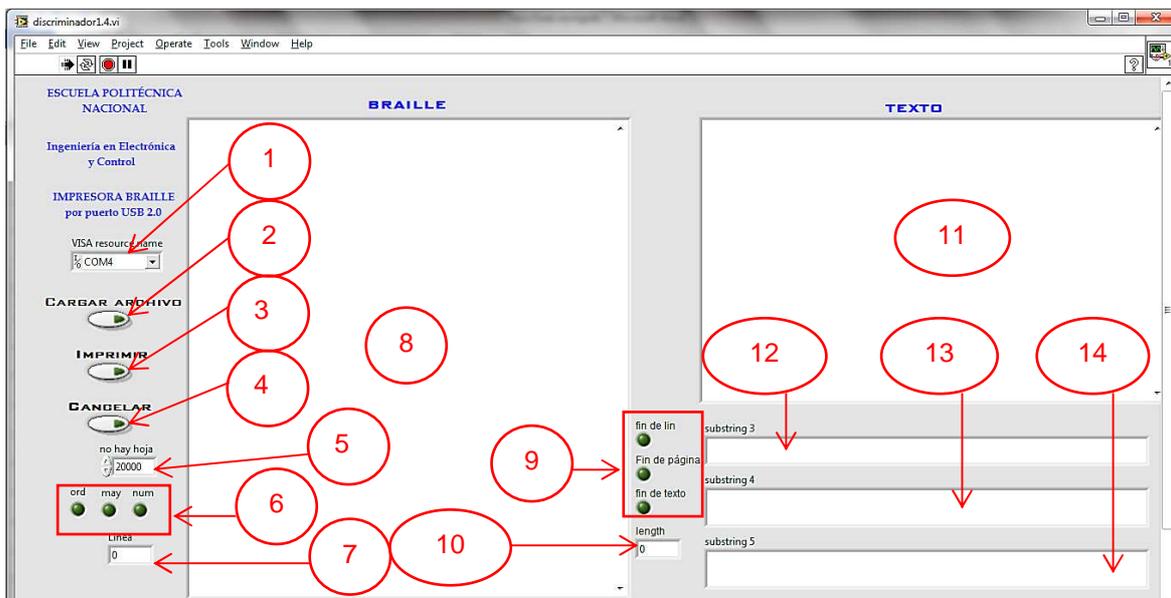


Figura 3.8 Pantalla principal de la interfaz de usuario

Como se observa en la Figura 3.8 las partes que componen la pantalla principal del programa de comunicación son varias y se detallan a continuación:

1. Numeración del puerto de comunicación.
2. Botón de carga de archivo.
3. Botón que ordena la impresión.
4. Botón de cancelación de la impresión del documento abierto.

5. Tiempo en ms que el modulo espera una respuesta si la primera y segunda carga de papel han fallado.
6. Indicadores de tratamiento de caracter: Ordinales, Mayúsculas y Números.
7. Indicador de número de línea enviado.
8. Pantalla de traducción a Braille del documento cargado en la pantalla "TEXTO".
9. Indicadores generales de: Fin de línea, Fin de hoja o página y Fin de texto.
10. Indicador de la dimensión de la palabra tratada.
11. Pantalla en donde aparecen los caracteres originales del documento a ser impreso.
12. "Substring3": en donde se almacena la palabra a ser tratada y en caso de sobrepasar los 30 caracteres que componen una fila es en esta variable en donde permanece hasta la carga de los demás caracteres que forman la siguiente fila.
13. "Substring4": en donde se forma la fila de 30 caracteres a ser enviados a través del puerto USB.
14. "Substring5": en donde se coloca la fila a ser enviada mientras las variables anteriores continúan tratando datos a ser enviados.

En la Figura 3.9 se puede observar un documento y su traducción a código Braille una vez cargado el mismo se encuentra listo para ser impreso.

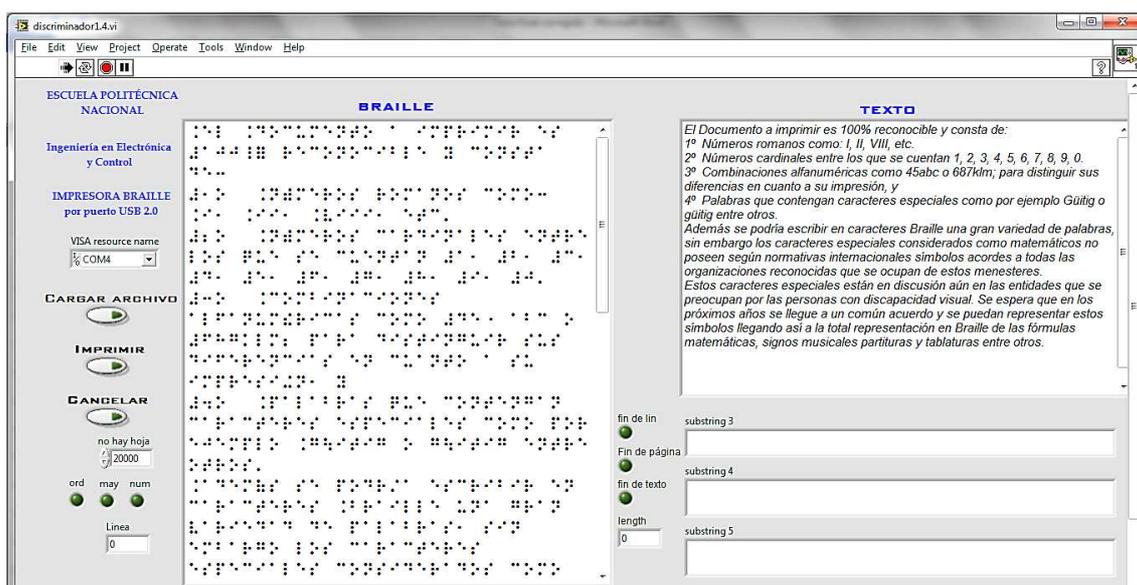


Figura 3.9 Documento previo a la Impresión

3.3.2 PROGRAMACIÓN INTERNA

3.3.2.1 Programa para el tratamiento de caracteres

Se desarrolla un programa que realiza una comparación y clasificación de todos los caracteres que forman parte del documento a imprimir, luego de abrirlo mediante el botón destinado a éste objetivo se compara caracter a caracter todos los componentes de una palabra, la cual está determinada entre un espacio en blanco y otro o entre un inicio de fila y un espacio en blanco. Cada palabra, al ser tratada y comparada caracter a caracter, puede clasificarse en letras mayúsculas, minúsculas y números, la misma se logra gracias a que el programa convierte cada letra a su equivalente en número decimal según su ubicación dentro de una tabla de caracteres ASCII, de esta manera se pueden clasificar dentro de ciertos rangos que determinan su ubicación en la misma por ejemplo: para las letras mayúsculas se compara que estén dentro del rango de 65 y 90, letras minúsculas entre 97 a 122, números entre el rango de equivalencia decimal ASCII de 48 a 57, etc. Algunas de las principales equivalencias ASCII se pueden observar en la Tabla 3.2.

Tabla 3.2 Principales equivalencias ASCII

DEC	HEX	CAR.	DEC	HEX	CAR.	DEC	HEX	CAR.
32	20	Espacio	70	46	F	107	6B	k
33	21	!	71	47	G	108	6C	l
34	22	"	72	48	H	109	6D	m
35	23	#	73	49	I	110	6E	n
37	25	%	74	4A	J	111	6F	o
40	28	(75	4B	K	112	70	p
41	29)	76	4C	L	113	71	q
42	2A	*	77	4D	M	114	72	r
43	2B	+	78	4E	N	115	73	s
44	2C	,	79	4F	O	116	74	t
45	2D	-	80	50	P	117	75	u
46	2E	.	81	51	Q	118	76	v

48	30	0	82	52	R	119	77	w
49	31	1	83	53	S	120	78	x
50	32	2	84	54	T	121	79	y
51	33	3	85	55	U	122	7A	z
52	34	4	86	56	V	160	A0	á
53	35	5	87	57	W	161	A1	í
54	36	6	88	58	X	162	A2	ó
55	37	7	89	59	Y	164	A4	ñ
56	38	8	90	5A	Z	165	A5	Ñ
57	39	9	97	61	a	167	A7	o
58	3A	:	98	62	b	168	A8	¿
59	3B	;	99	63	c	173	AD	i
63	3F	?	100	64	d	174	AE	«
64	40	@	101	65	e	175	AF	»
65	41	A	102	66	f			
66	42	B	103	67	g			
67	43	C	104	68	h			
68	44	D	105	69	i			
69	45	E	106	6A	j			

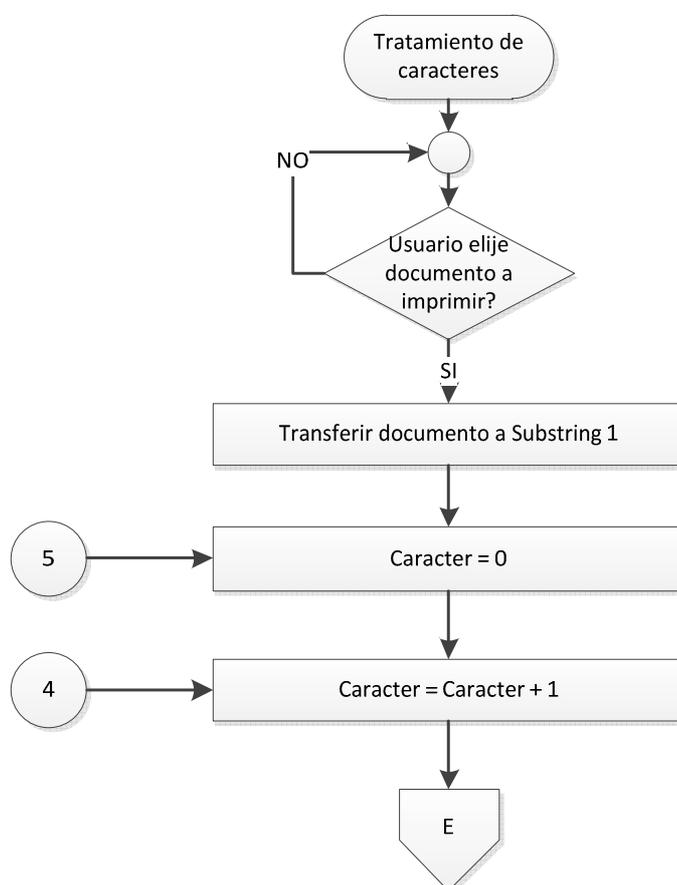
En el caso de encontrar una palabra con letras mayúsculas se aumenta un signo generador de las mismas en Braille previo a la escritura de la palabra, lo que no sucede si al contrario se posee palabras íntegramente escritas en minúsculas.

De encontrar un número debe aumentarse un signo generador del mismo incrementando la longitud de la palabra tratada en un caracter.

En el caso de hallar un número ordinal el programa realiza un cambio de caracter identificador del número, pues estos deben escribirse en posición baja (Poseen las mismas características del caracter Braille que corresponde a su número pero desplazándolo una fila hacia abajo dentro de la celda 3 x 2), cabe anotar que su terminación será la letra “o” y conservan previo a su escritura el signo generador de número.

Una vez realizada esta discriminación se procede a guardar el nuevo archivo con los respectivos aumentos que determina la codificación Braille en un Substring que en la pantalla posee el nombre de "BRAILLE" (Internamente corresponde a Substring 2), enviándose de éste todos los caracteres existentes respetando el número de 30 de estos por fila y 22 filas que componen una hoja hasta que el documento termine, al ocurrir esto se envía hacia el módulo un caracter de control (Alt + 175 "»") que informa al prototipo la ausencia de caracteres a imprimir.

En la Figura 3.10 se observa el flujo grama implementado.



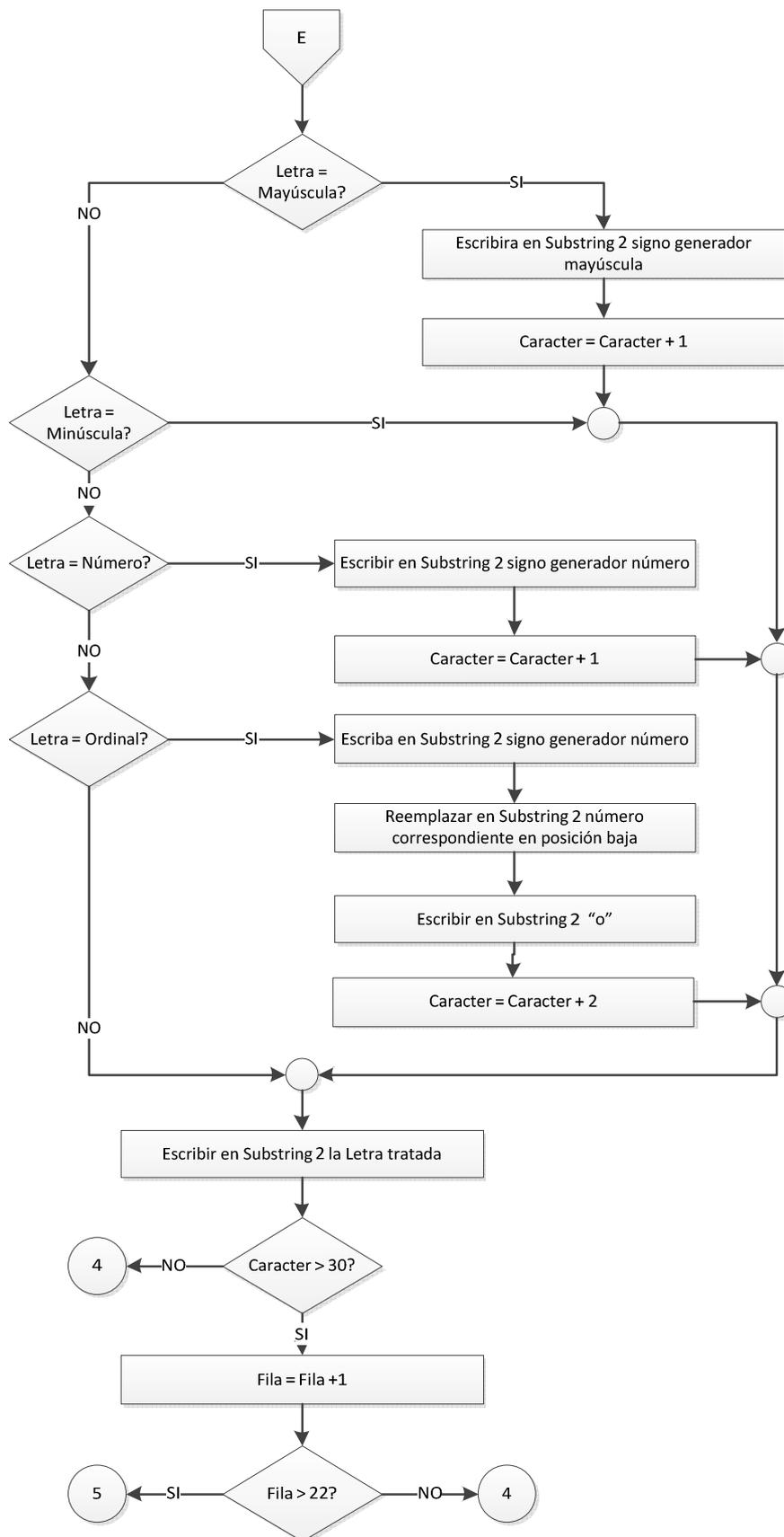


Figura 3.10 Diagrama de Flujo de tratamiento de caracteres

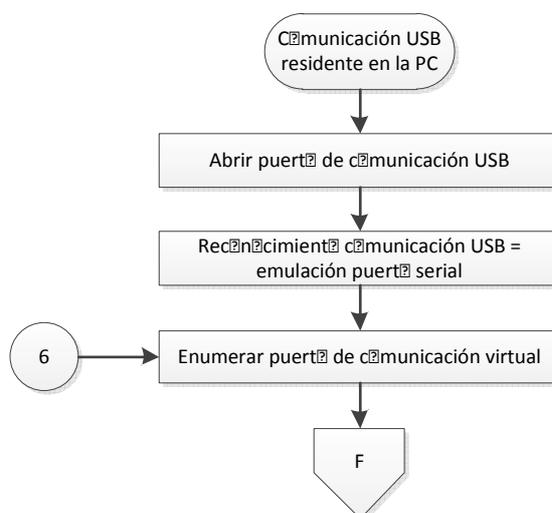
3.3.2.2 Envío y recepción de datos en la interfaz de usuario

3.3.2.2.1 Comunicación por el Puerto USB

En un inicio la PC reconoce la conexión y asigna el número del puerto serial virtual en el cual se desarrollará la comunicación, luego envía el primer carácter de control al microcontrolador, una vez realizado esta transmisión entra en espera para recibir el eco del carácter antes enviado, confirmando así que el prototipo ha cargado una hoja y se encuentra listo para recibir los caracteres que componen el documento a imprimir.

Luego de comparar y almacenar los caracteres recibidos el microcontrolador procede a realizar la impresión mecánica, una vez terminado éste proceso emite hacia la PC un carácter de control "F", el cual indica que la PC puede enviar los siguientes 30 caracteres hasta completar una hoja o hasta terminar el documento, si por el contrario recibe el carácter "Z" en vez de enviar los 30 caracteres correspondientes, debe esperar a recibir el carácter de confirmación de que la hoja se encuentra cargada para proceder a enviarlos hasta que el documento termine.

Al finalizar el documento se envía un carácter (Alt + 175 "»") para indicar que la transferencia de datos finalizó, el diagrama de flujo de éste proceso se muestra en la Figura 3.11.



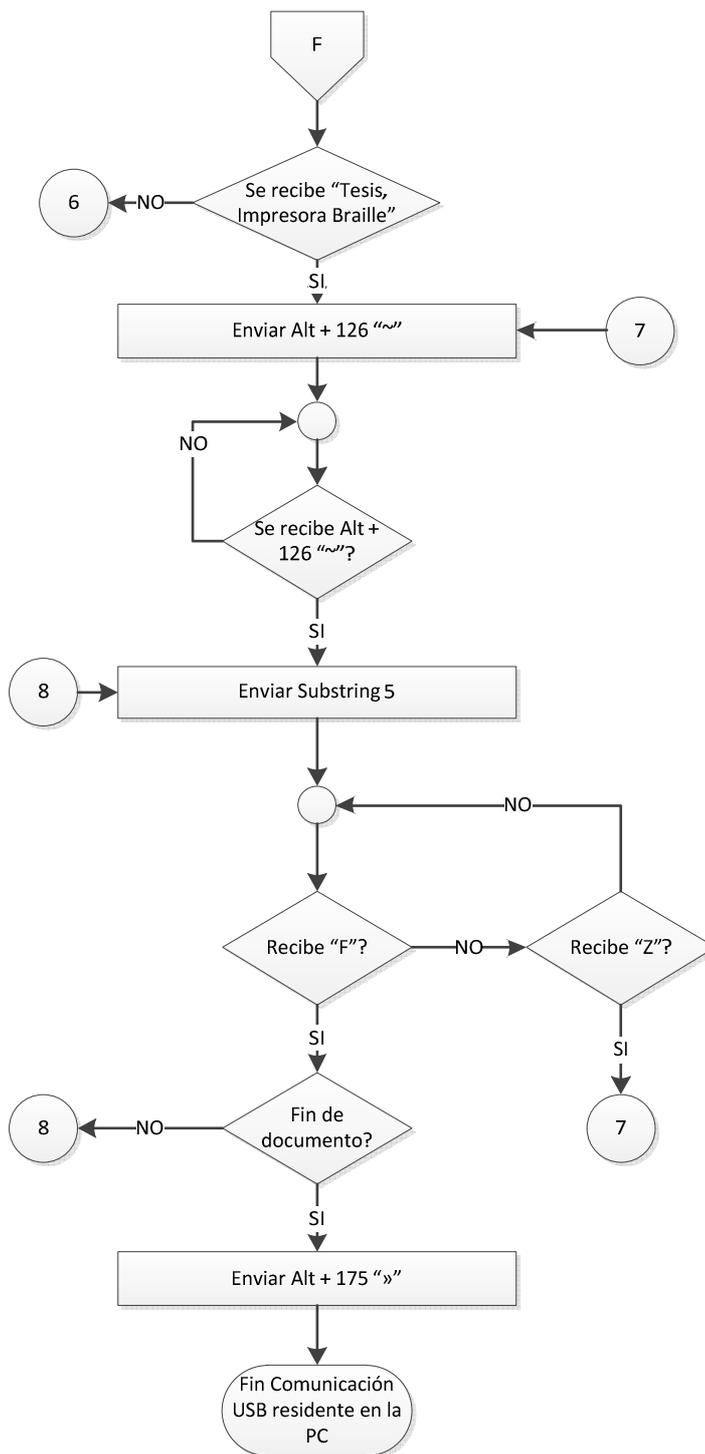


Figura 3.11 Comunicación desde la PC a través del puerto USB

En el siguiente capítulo se tratan las pruebas realizadas al prototipo, así como un análisis de los resultados de las mismas.

CAPITULO 4

PRUEBAS Y RESULTADOS

En éste capítulo se muestra las diferentes pruebas realizadas al prototipo “Impresora Braille”.

Para tener un resultado final adecuado que cumpla los alcances propuestos en el presente proyecto, se realizan pruebas que permiten verificar la respuesta de las partes que integran el mismo. Se realizan en primer lugar las pruebas mecánicas independientemente de la interfaz USB para posteriormente verificar la misma y las pruebas en conjunto módulo – software con el programa implementado en la PC verificando así el cumplimiento del objetivo planteado.

4.1 PRUEBAS MECANICAS

4.1.1 PUNZÓN

4.1.1.1 Principio de Percusión (Motor DC)

Para la construcción del punzón, la primera idea impregnada utiliza un motor de DC sin motoreductor acoplado directamente al mecanismo de percusión; sin embargo los resultados no fueron los esperados debido a que la fuerza producida por el eje del motor no era suficiente para realizar un movimiento en el vástago que permite la impresión de un punto en el papel como se observa en la Figura 4.1.

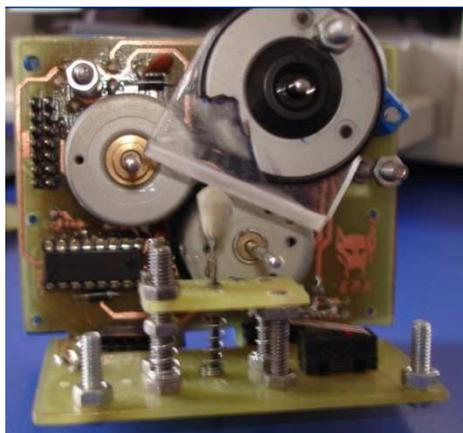


Figura 4.1 Primer punzón implementado

Al implementar el principio anterior añadiendo un motoreductor tampoco se observan resultados favorables ya que, al disminuir la velocidad solamente ocasiona que el papel se arrugue (Figura 4.2).



Figura 4.2 Resultado obtenido con el punzón basado en un motor de DC con motoreductor

4.1.1.2 Principio de taladrado

Se implementa un sistema cuyo principio es taladrar con la ayuda de un motor DC con motoreductor presente en un servomotor de modelismo, como se muestra en la Figura 4.3.

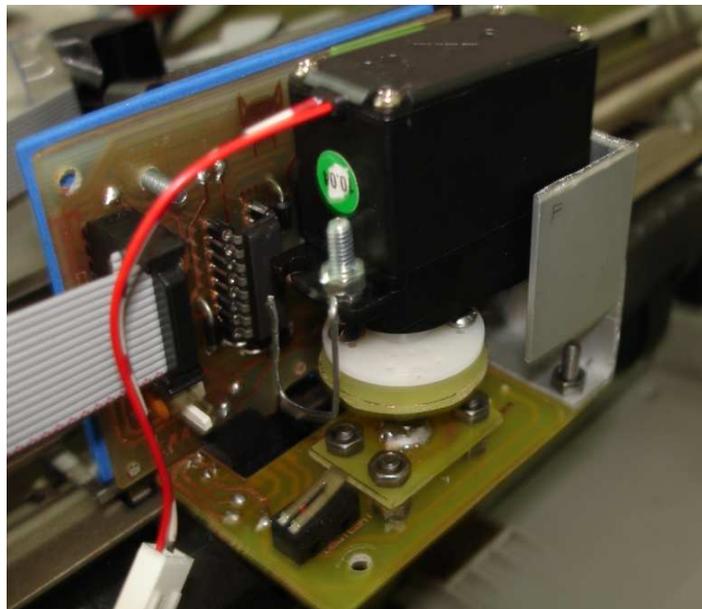


Figura 4.3 Punzón implementado con un servomotor de modelismo (principio de taladrado)

Para tener rigidez en el nuevo prototipo de punzón se ubican piezas de aluminio en la estructura de éste, ya que el material es sumamente resistente y liviano, lo que ayuda a desplazarse sobre el bocín al carrete sin tener problemas de atascamiento o rozamiento excesivo, esta implementación se puede ver en la Figura 4.4.

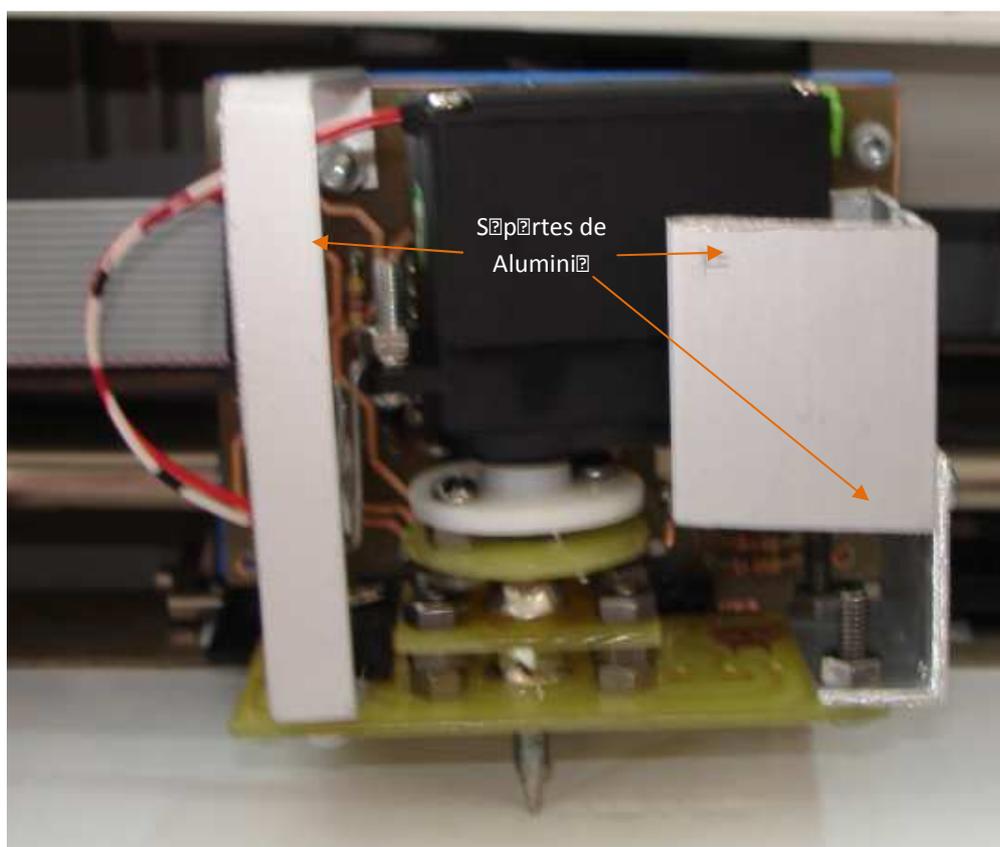


Figura 4.4 Soportes de aluminio acoplados al módulo punzón

En las pruebas realizadas de impresión en el papel se observa que no se obtiene el relieve esperado de 0.5 mm, pues solamente alcanza medidas de 0.3 mm debido a que el cabeceo que se produce en el carrete al momento de la impresión, los resultados de esta prueba se visualizan en la Figura 4.5 (a). Para solucionar éste inconveniente se adecua una pieza de aluminio que sostiene al carrete en su parte posterior cuya implementación se muestra en la Figura 4.6.



(a)



(b)



(c)

Figura 4.5 Pruebas de impresión con el punzón basado en el principio de taladrado

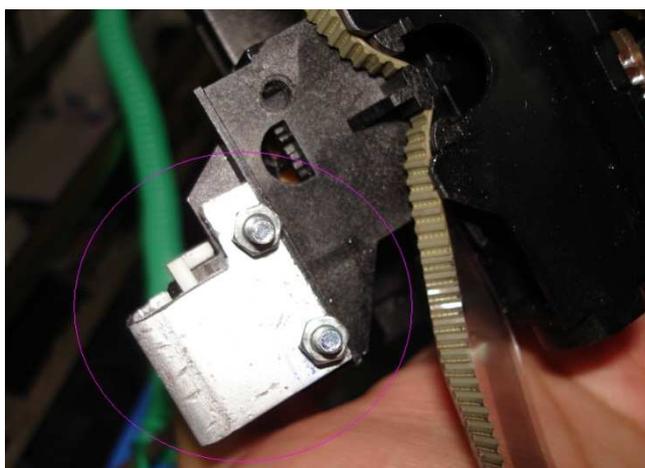


Figura 4.6 Soporte posterior del carrete

Para el accionamiento del final de carrera que activa el movimiento de retorno del vástago se coloca un cable semirrígido (clip), sin embargo la flexibilidad del mismo

ocasiona que no siempre la profundidad de impresión sea la misma pues esta varía entre 0.3 mm y 0.4 mm, para corregir éste error se reemplaza el cable de clip por un ángulo de aluminio, con esta modificación se consigue el resultado esperado en cuanto a profundidad de punto de 0.5 mm. La profundidad de perforación en los caracteres es la misma y se encuentra dentro de las normas Braille (se considera aceptable y legible si consta dentro del rango de entre 0.5 y 0.6 mm). Éste resultado se puede visualizar en la Figura 4.5 (c).

Pese a que los resultados en cuanto a dimensiones de relieves y separación son los aceptados por las normativas Braille, en pruebas de impresión se determina la velocidad del dispositivo para imprimir cada celda Braille (3 x 2 puntos) en 4,5 minutos aproximadamente, con lo que un texto completo podría tardar en imprimirse varias horas. Con éste resultado se construye otro dispositivo que provea de velocidad y fuerza para realizar los relieves necesarios en la normativa Braille, se retorna al principio de percusión pero en esta ocasión el actuador es un solenoide de vástago móvil alimentado mediante DC, cuyos resultados se detallan en el siguiente apartado.

4.1.1.3 Principio de Percusión (Solenoid)

Con éste tipo de punzón es necesario excitar a la bobina para producir un avance drástico del vástago que se encuentra en el núcleo, y para su retorno simplemente se quita la alimentación al solenoide y retorna a su estado inicial mediante la acción de un resorte, el módulo construido puede apreciarse en la Figura 4.7.



Figura 4.7 Punzón de percusión en base a un solenoide

Mediante esta implementación se logra que los caracteres Braille sean aún más legibles que con el punzón del literal anterior (profundidad de caracteres entre 0.5 y 0.55 mm) y sus tiempos de ejecución son reducidos, pues mediante pruebas de impresión se promedia que cada caracter Braille (celda completa 3 x 2 puntos) es realizada en 7 segundos aproximadamente, cumpliendo claro está con las medidas de separación y relieve que dicta las normativas que rigen esta codificación.

4.1.2 BANDEJA DE APOYO DE IMPRESIÓN

Se propone que la bandeja de apoyo para la impresión sea de un material flexible, como es el acrílico, con el propósito de que al momento de la impresión éste material ceda y no dañe el punzón ni la hoja a imprimirse, al instante de realizar pruebas de esta aplicación el apoyo sufre deformaciones considerables con lo que las impresiones de los relieves en unos casos no se realizaban sobre todo en el punto medio del soporte.

Posteriormente se reemplaza éste material por uno más resistente como es el aluminio, para conseguir que la impresión no dañe el punzón por las características propias del material en el que se construye el apoyo, se coloca una capa de material esponjoso (Fomix). Éste cambio se observa en la Figura 4.8.

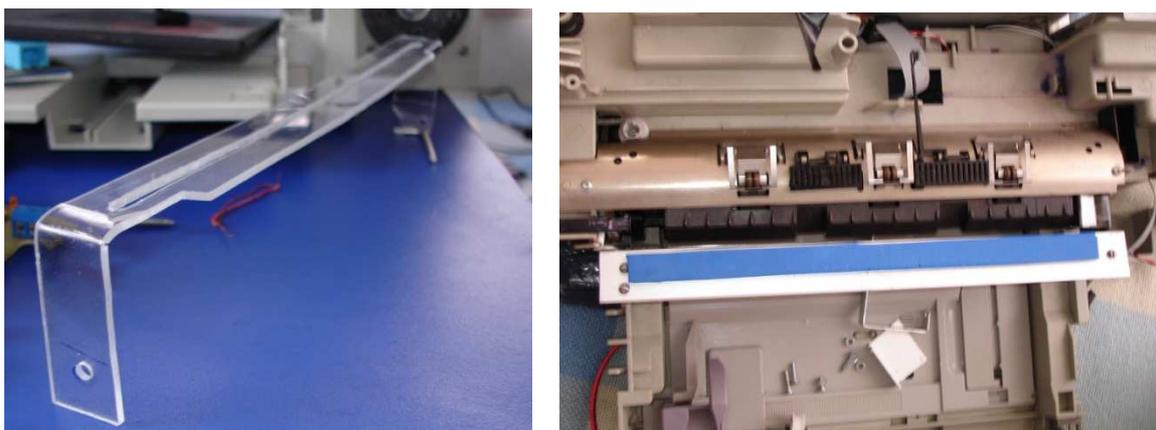


Figura 4.8 Reemplazo de la Bandeja de Apoyo de Impresión

En pruebas realizadas se observa que éste material por sí solo no provee de un molde adecuado para el caracter a ser impreso, pues en muchos casos el punzón perfora completamente la hoja, debido a esto se añade a la capa de material

esponjoso una lámina de acetato en su parte superior, adjuntando así rigidez al molde de punzado formado por el material esponjoso. Con esta modificación el soporte de impresión cumple con su objetivo, el de brindar apoyo a la impresión y también servir de molde para los relieves a ser realizados.

4.1.3 POSICIONAMIENTO DEL PAPEL ENTRE LA BANDEJA Y EL RODILLO

Luego de un análisis de cómo una impresora convencional ubica el papel previamente a una impresión, se desarrolla un programa en el microcontrolador en lenguaje PIC C que accione los motores necesarios (motor DC para el movimiento del carrete y un motor a pasos para el movimiento del rodillo) para movilizar el papel desde la bandeja de alojamiento del mismo hasta que la hoja a ser impresa se mueva conjuntamente con el rodillo.

En un inicio debido a la elevada velocidad del motor de DC, el carrete sufre un atascamiento al final de su recorrido de derecha a izquierda lo que ocasiona una sobre exigencia de corriente para el driver que lo comanda provocando que el sistema se reinicie, para solucionar esta contrariedad se modifica el programa de control afinando el ancho de pulso de activación, de esta manera se logra que inclusive al llegar al final de su recorrido, manteniendo ese mismo ancho de pulso solo mantenga presionado el mecanismo de acoplamiento entre el rodillo y el levantamiento de la bandeja de alojamiento de papel, y el proceso de subir el papel hacia el rodillo se produzca sin contratiempos.

4.2 PRUEBAS DE SOFTWARE

4.2.1 MICROCONTROLADOR

Se verifica que los caracteres que se envían desde la PC son correctos, es decir que el módulo recibe todos y cada uno de los datos que la PC le envía, esta prueba se realiza mediante el uso del simulador "SIOW" de PIC C, para una verificación en el módulo se modifica el programa para que estos datos aparezcan en el LCD como se muestra en la Figura 4.9.



Figura 4.9 Recepción de caracteres en el módulo

Al desarrollar una tabla de comparación entre los caracteres a recibirse desde la PC y aquellos que constan en una tabla convencional de equivalencias de caracteres ASCII se observa que el lenguaje de programación PIC C tiene limitado el número de caracteres desde el 32 correspondiente a “espacio en blanco” hasta el 126 correspondiente a “~”, esto quiere decir que el único intervalo que reconoce como datos validos el PIC si se tratan de variables tipo “char” o caracter se encuentra dentro de éste rango. El microcontrolador al no tener elementos con los cuales comparar toma los datos fuera de éste rango como si fuesen espacios en blanco por lo que no se visualizan en el LCD y tampoco se imprimen en el papel. Para solucionar éste inconveniente se manejan los elementos restantes que conforman la tabla de equivalencias en Braille, como una variable numérica (tipo byte) en lugar de manejarla como una variable de tipo

“char”, al realizar esta implementación todos los caracteres enviados desde la PC son reconocidos como se puede observar en la Figura 4.10 donde se muestra que al enviar el código que identifica un carácter que se encuentra fuera del rango de operación de la tabla de PIC C (carácter Alt + 175 o “»”), pero que está implementado como carácter de control, en el LCD del módulo “Impresora Braille” se identifica con la equivalencia creada desde la programación, es decir no se observa el carácter pero se ejecuta la orden ligada al mismo.

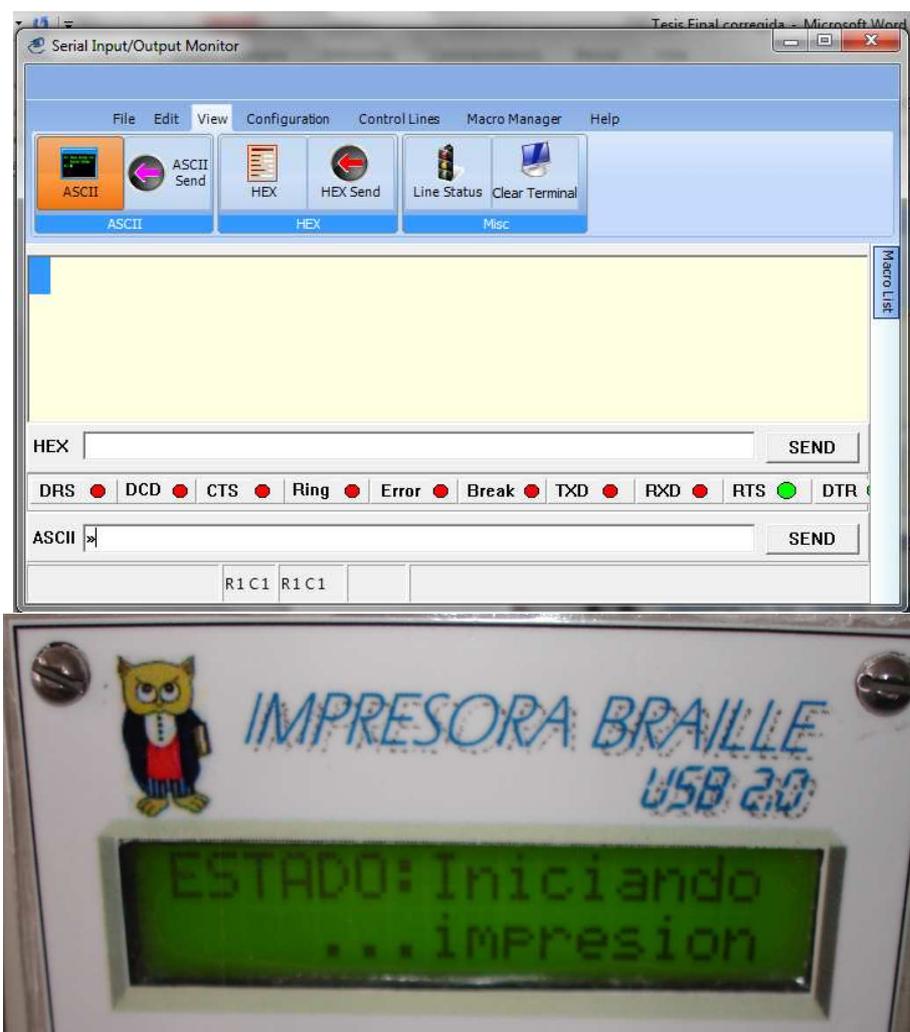


Figura 4.10 Visualización en el LCD de caracteres que se encuentran fuera de los caracteres reconocibles por PIC C

4.2.2.1 Importación del Documento con extensión .doc

Siendo LabVIEW el entorno en el cual se desarrolla la aplicación para el computador se realiza una prueba, mediante la cual se abre un documento de Word con extensión .doc para verificar su contenido, sin embargo al ejecutarla se

observa que el documento no solo está compuesto de las letras que se grabaron en el sino también de un encabezado y una parte final que indican algunas de las características propias con las que se guarda el documento, éste encabezado no es de utilidad para la impresión, la cantidad de estos caracteres es de 2560 y para éste tipo de extensión de documentos permanece constante en todos los casos, por lo que se procede a eliminarlos para aprovechar solamente los caracteres presentes en el documento. En la Figura 4.12 se aprecian tanto el documento .doc abierto con la totalidad de sus caracteres y después de implementar un offset para cortar o eliminar los 2560 caracteres que no se usan.

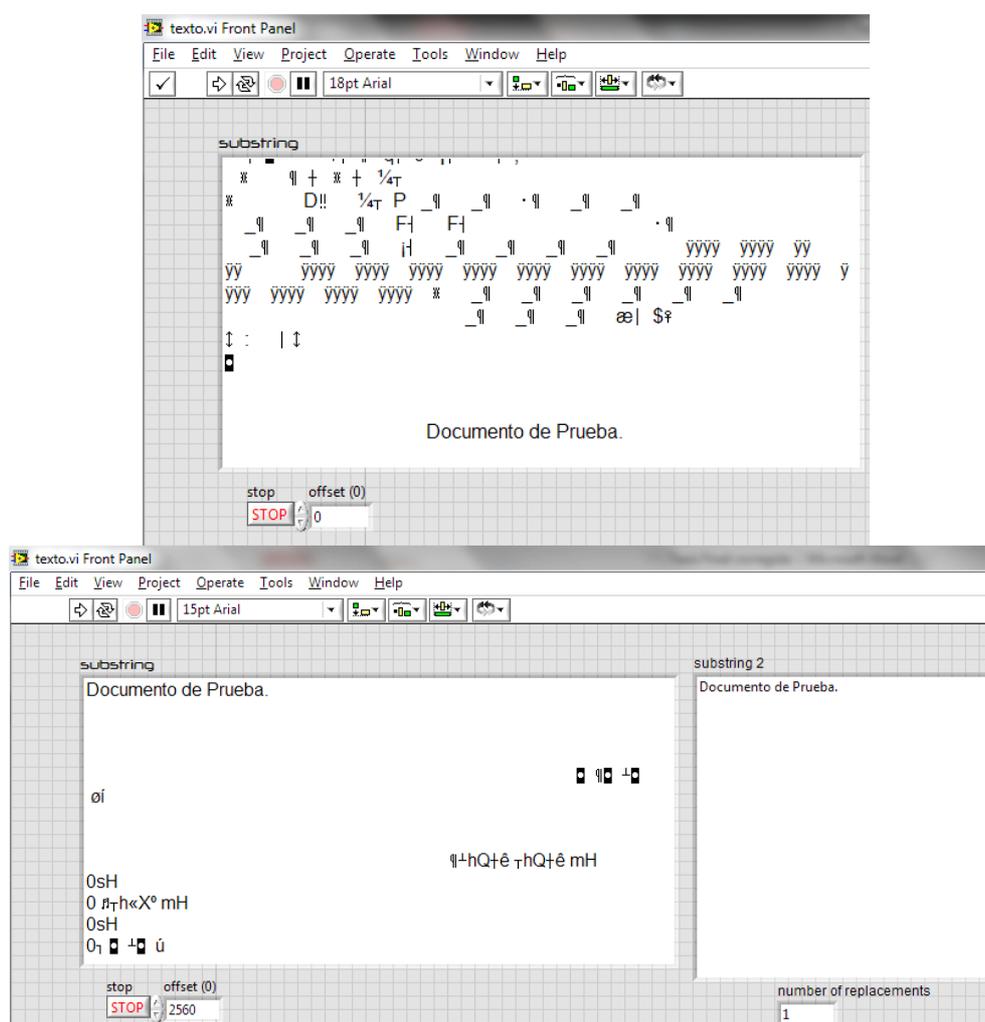


Figura 4.12 Eliminación de encabezado en un documento .doc previo a la impresión

4.3 PRUEBAS MEDIANTE SOFTWARE DE CONTROL

4.3.1 COMUNICACIÓN USB

Se verifica el funcionamiento del Driver de Identificación proporcionado por Microchip para dispositivos de clase CDC (communications device class) conectando el módulo impresora al puerto USB de la PC.

La primera vez que se conecta el dispositivo el sistema operativo Windows pide la ubicación del Driver para habilitar el puerto serial virtual como se observa en la Figura 4.13.

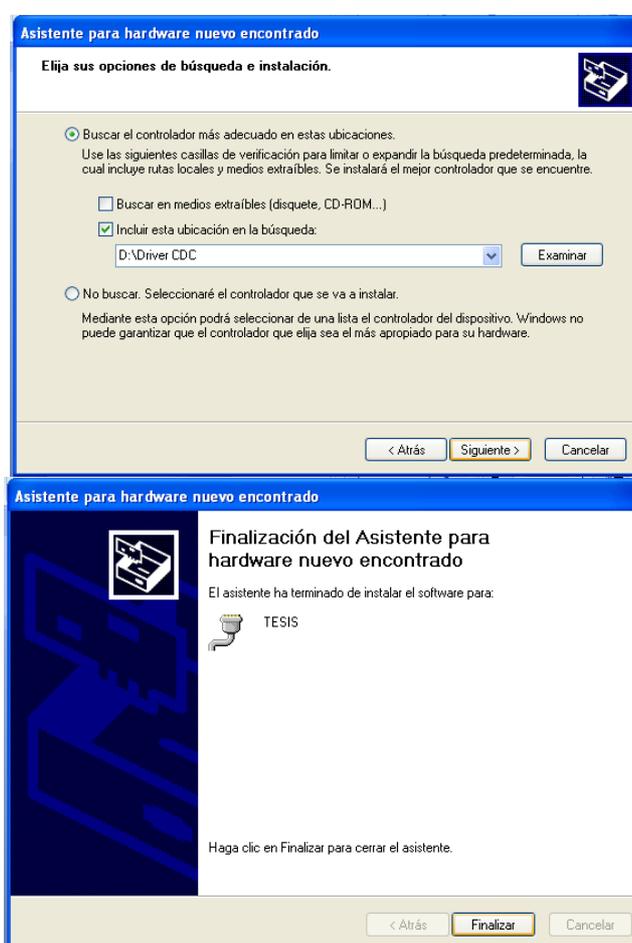


Figura 4.13 Instalación del Driver para el módulo Impresora Braille

Una vez instalado el Driver se tiene un mensaje de confirmación a través del LCD y del reproductor de voz como se muestra en la Figura 4.14.



Figura 4.14 Mensaje de conformación de conexión realizada correctamente

4.3.1.1 Pruebas con el Simulador SIOW de PIC C

Se envían distintos caracteres de control para verificar el comportamiento del prototipo. En la Figura 4.15 se nota la respuesta del prototipo al caracter de control correspondiente a la numeración “B5” en hexadecimal, en éste proyecto éste comando ordena que el papel se posicione en la bandeja de apoyo de impresión y que el prototipo espere los caracteres a ser impresos.



Figura 4.15 Mensaje de concordancia de comando utilizando emulador de comunicación SIOW de PIC C

4.3.2 RESPUESTA DEL PROTOTIPO AL ENVIO DE CARACTERES

Las pruebas realizadas reflejan concordancia con los puntos que generan los diferentes caracteres aceptados en el idioma castellano, estos caracteres constan en la Tabla 4.1, cabe recalcar que existen combinaciones de puntos que no tienen correspondencia con ninguna letra reconocida, sin embargo en otros idiomas la poseen de acuerdo con la conveniencia de la organización encargada de estos menesteres en los diferentes países que tienen otros idiomas.

Tabla 4.1 Combinaciones de puntos y sus equivalencias en código Braille

TABLA DE CARACTERES BRAILLE	
	
PUNTOS DE LA CELDA	CARACTER
	Espacio en blanco
1	1 a
12	2 b
13	k
14	3 c
15	5 e
123	l
124	6 f
125	8h
126	(
134	m
135	o
136	u
145	4 d
1234	p
1235	r
1236	v

1245	7 g
1256	ü
1345	n
1346	x
1356	z
12345	q
12356	á
12456	ñ
13456	y
2	,
23	;
24	9 i
25	:
26	¿?
234	s
235	¡
236	"
245	J
246	□
2345	T
2456	w
23456	Ú
3	.
34	í
36	-
345)
356	Ordinal 0
3456	#
46	Signo generador de mayúsculas
5	@

4.3.2.1 Impresión de caracteres del alfabeto

Como se observa en la Figura 4.17 se imprimen los caracteres del documento de la Figura 4.16 en el cual se tiene el alfabeto en letras minúsculas y a continuación el mismo alfabeto en letras mayúsculas como se espera se tiene impreso el caracter generador para indicar al lector que a continuación se tienen solamente letras mayúsculas, esta prueba se ilustra en las Figuras 4.18 y 4.19.

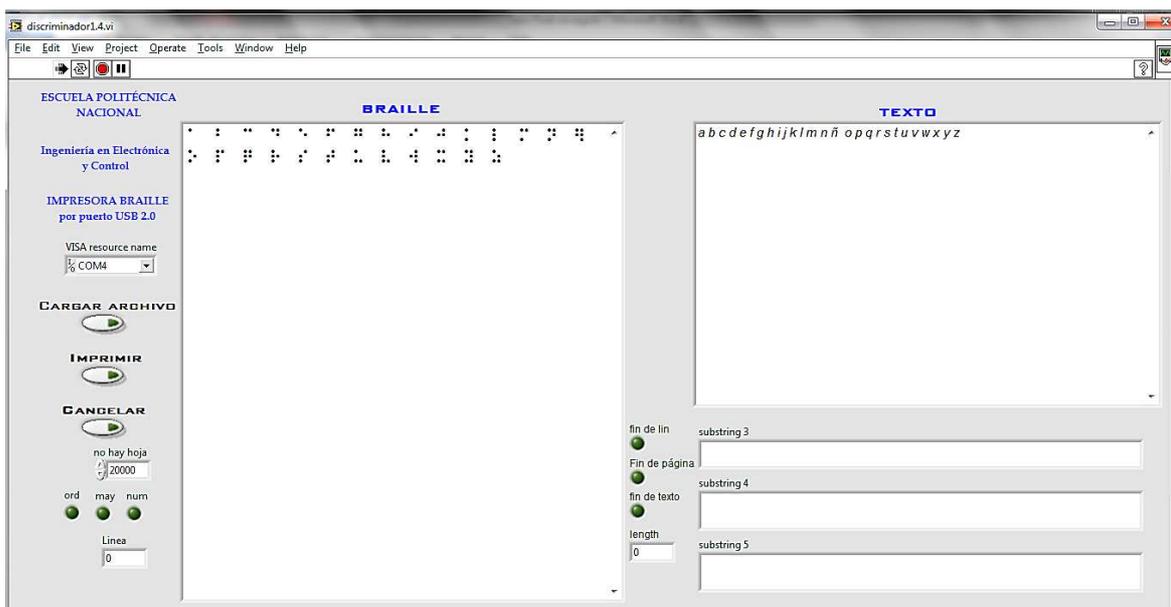


Figura 4.16 Preparación de documento a imprimir “alfabeto en minúsculas”

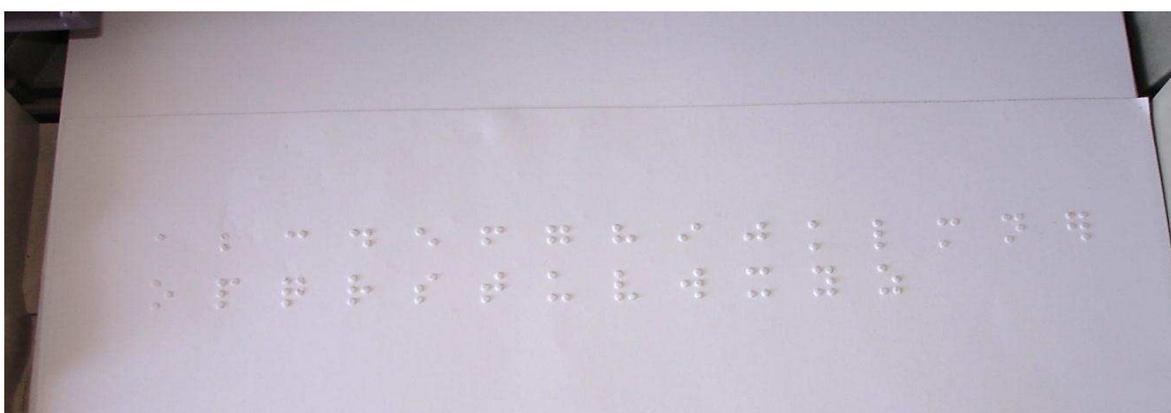


Figura 4.17 Impresión de documento “alfabeto en minúsculas”

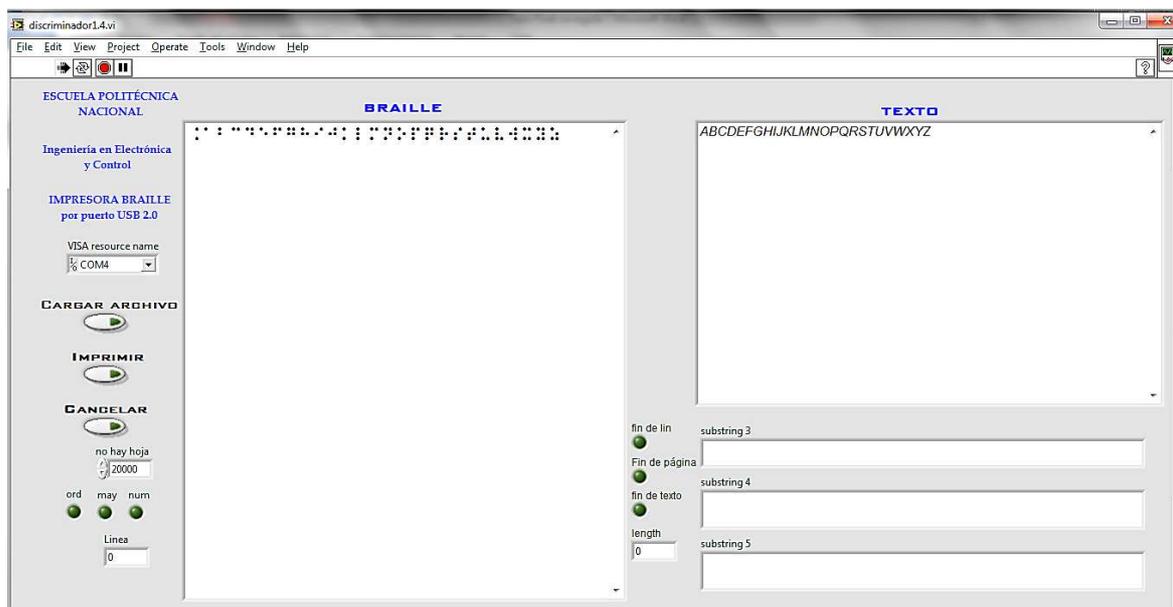


Figura 4.18 Preparación de documento a imprimir “alfabeto en mayúsculas”

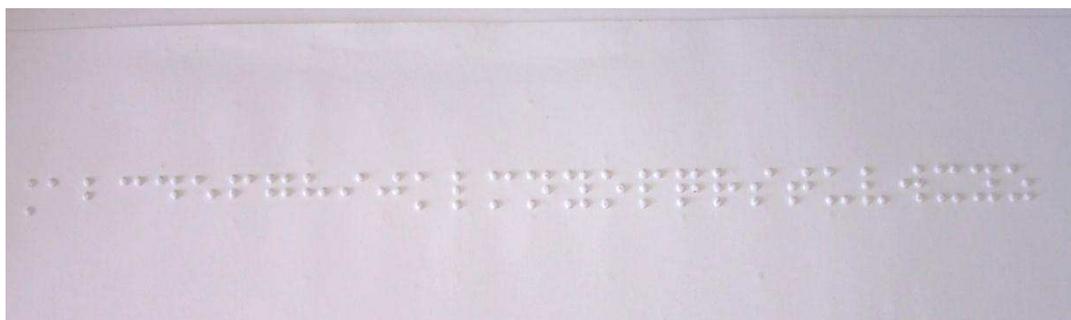


Figura 4.19 Impresión de documento “alfabeto en mayúsculas”

4.3.2.1.1 Impresión de caracteres especiales

En las Figuras 4.20 y 4.21 se observan los resultados obtenidos en la impresión de caracteres que se consideran especiales como es el caso de las vocales acentuadas y también la letra “ü”. La letra “ñ” también se la considera dentro de éste grupo, sin embargo la impresión de éste carácter se realiza previamente en las impresiones del alfabeto.

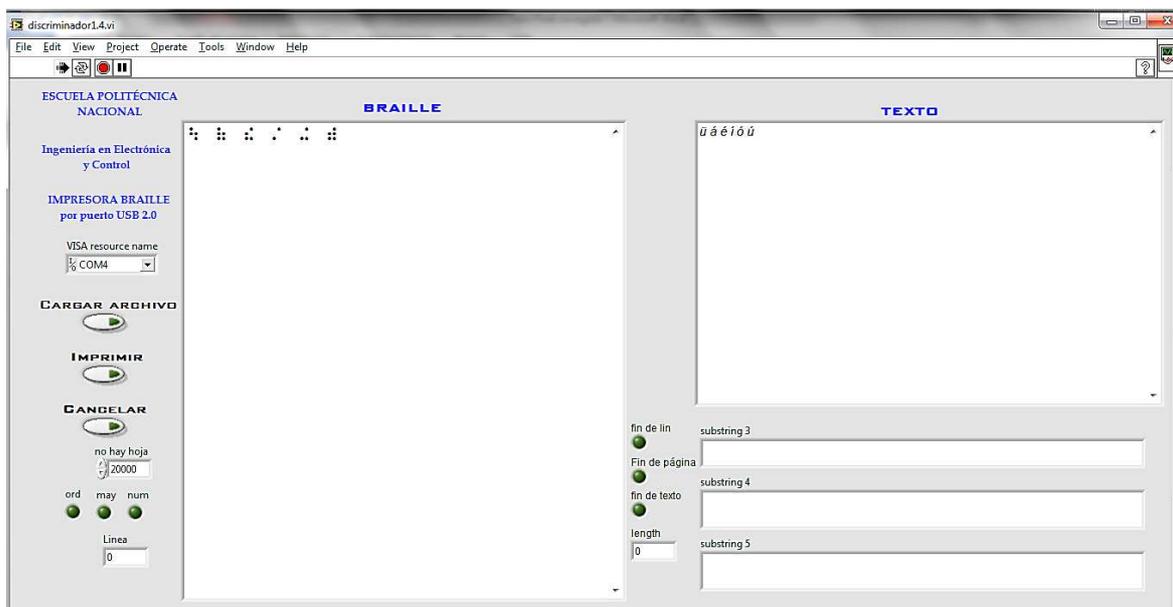


Figura 4.20 Preparación de documento a imprimir “caracteres especiales”



Figura 4.21 Impresión de documento “caracteres especiales”

4.3.2.2 Impresión de caracteres numéricos

En la Figura 4.23 se muestra el resultado de la impresión de números enviados (Figura 4.22) los mismos que se encuentran precedidos del caracter indicador o generador de números.

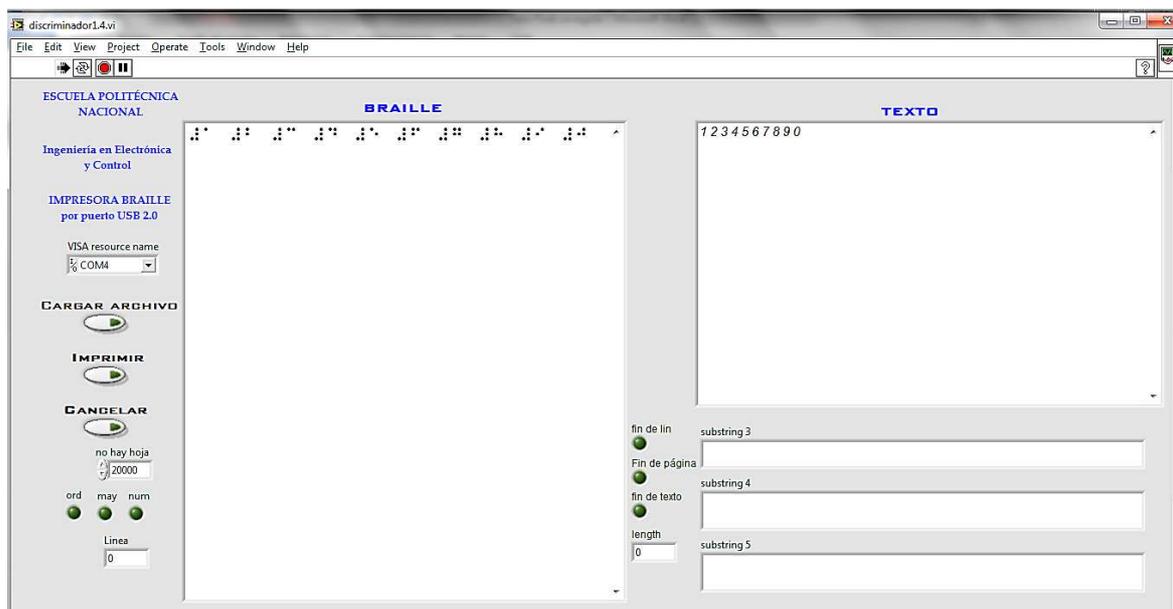


Figura 4.22 Preparación de documento a imprimir “números”



Figura 4.23 Impresión de documento “números”

4.3.2.2.1 Impresión de números ordinales

La impresión de los números ordinales desde el 1 al 10 femeninos se pueden observar en las Figuras 4.24 y 4.25.

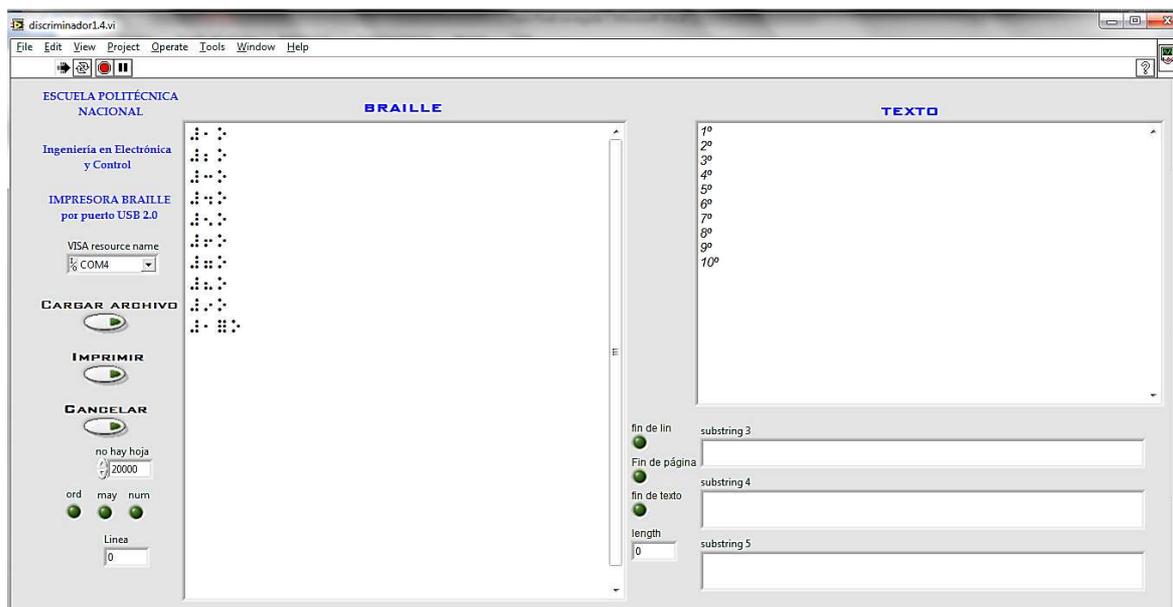


Figura 4.24 Preparación de documento a imprimir “números ordinales”

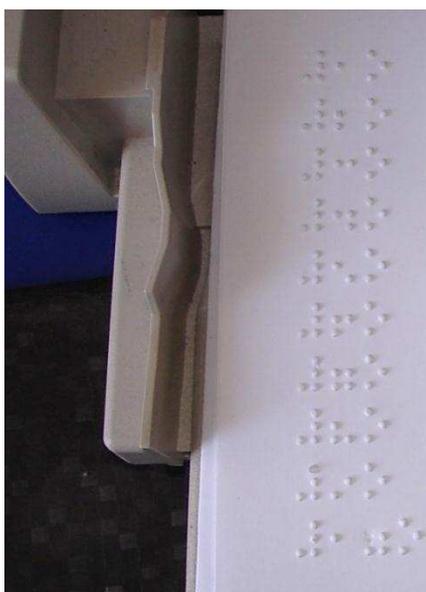


Figura 4.25 Impresión de documento “números ordinales”

4.3.2.2.2 Impresión de números romanos

Algunos números romanos son incluidos en el documento que se envía a imprimir y esta prueba se evidencia en las Figuras 4.26 y 4.27.

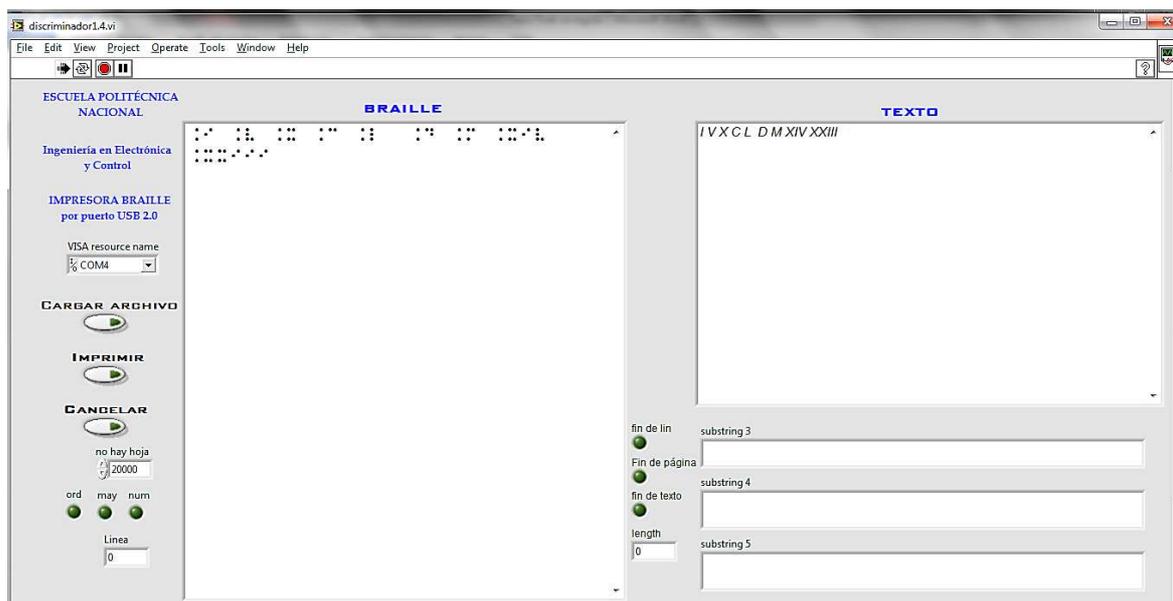


Figura 4.26 Preparación de documento a imprimir “números romanos”



Figura 4.27 Impresión de documento “números romanos”

4.3.2.3 Impresión de símbolos

Los símbolos que son considerados en esta prueba de impresión son considerados los más importantes dentro de la puntuación, sus resultados se evidencian en las Figuras 4.28 y 4.29.

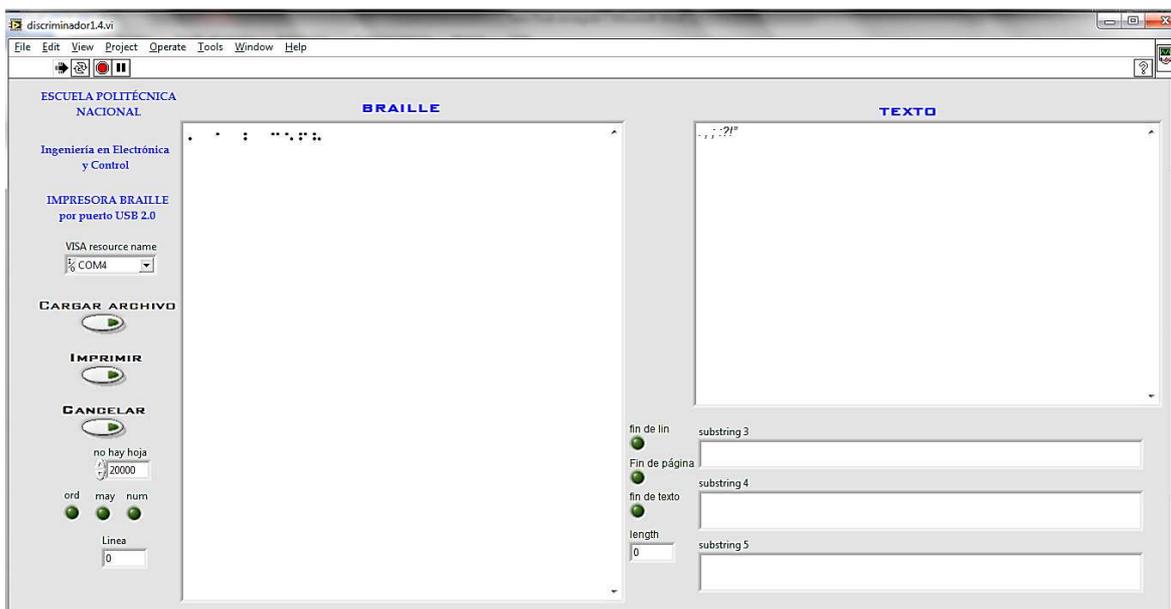


Figura 4.28 Preparación de documento a imprimir “símbolos de puntuación”



Figura 4.29 Impresión de documento “símbolos de puntuación”

En las pruebas realizadas anteriormente referentes a impresiones de caracteres Braille se muestra la concordancia entre la traducción Braille de puntos realizada en la pantalla de LabVIEW y los puntos impresos físicamente después de realizadas las mismas verificando la efectividad de los caracteres grabados en la tabla interna del microcontrolador.

4.3.2.4 Cálculo de errores relativos a las pruebas de impresión realizadas

En la Tabla 4.2 se tabulan los diferentes resultados con sus respectivos errores absolutos y relativos de la distancia horizontal entre los centros de puntos contiguos de la misma celda o dimensión “a”, las medidas se realizaron utilizando un calibrador pie de rey, tomando en cuenta las dimensiones más significativas respecto a los caracteres Braille generados, estas pruebas de medida se ilustran en la Figura 4.30.

En la Figura 4.31 se ilustran las medidas más significativas a tomar en cuenta en las normativas que rigen el código Braille.

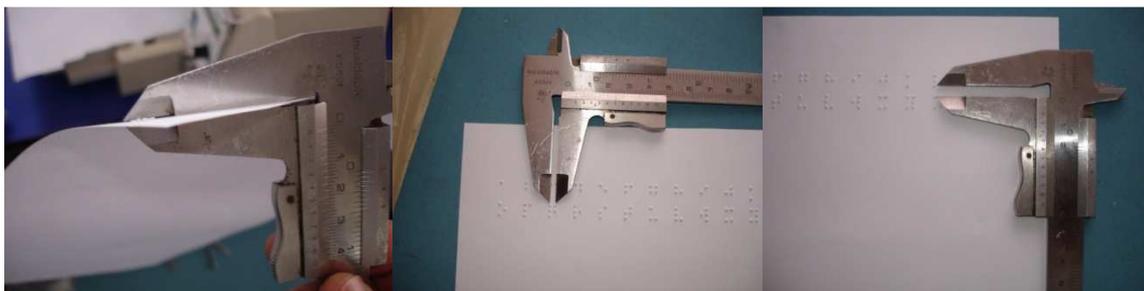


Figura 4.30 Toma de medidas de las pruebas de impresión realizadas

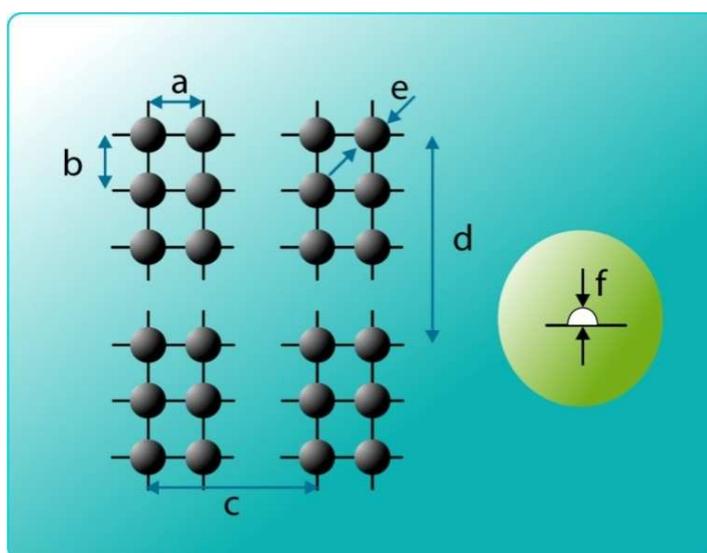


Figura 4.31 Medidas significativas en las celdas Braille

Tabla 4.2 Tabulación de datos relativos a la distancia "a"

PRUEBAS	DISTANCIA DESEADA (mm)	DISTANCIA MEDIDA (mm)	ERROR ABSOLUTO	ERROR RELATIVO (%)
1	2,5	2,4	0,1	4,17
2	2,5	2,5	0,0	0,00
3	2,5	2,6	0,1	3,85
4	2,5	2,6	0,1	3,85
5	2,5	2,4	0,1	4,17
6	2,5	2,6	0,1	3,85
7	2,5	2,7	0,2	7,41

En las Tablas 4.3, 4.4, 4.5, 4.6 y 4.7 se tabulan los diferentes resultados de las medidas tomadas correspondientes a; "b" la distancia vertical entre los centros de puntos contiguos de la misma celda, "c" la distancia entre los centros de puntos en

idéntica posición de celdas contiguas, “d” la distancia entre centros de puntos en idéntica posición de líneas contiguas, “e” el diámetro de la base de los puntos y “f” la altura del relieve de los puntos.

Tabla 4.3 Tabulación de datos relativos a la distancia “b”

PRUEBAS	DISTANCIA DESEADA (mm)	DISTANCIA MEDIDA (mm)	ERROR ABSOLUTO	ERROR RELATIVO (%)
1	2,5	2,5	0,0	0,00
2	2,5	2,4	0,1	4,17
3	2,5	2,4	0,1	4,17
4	2,5	2,5	0,0	0,00
5	2,5	2,5	0,0	0,00
6	2,5	2,4	0,1	4,17
7	2,5	2,5	0,0	0,00

Tabla 4.4 Tabulación de datos relativos a la distancia “c”

PRUEBAS	DISTANCIA DESEADA (mm)	DISTANCIA MEDIDA (mm)	ERROR ABSOLUTO	ERROR RELATIVO (%)
1	6,0	6,2	0,2	3,23
2	6,0	6,5	0,5	7,69
3	6,0	6,1	0,1	1,64
4	6,0	6,3	0,3	4,76
5	6,0	6,3	0,3	4,76
6	6,0	6,1	0,1	1,64
7	6,0	6,1	0,1	1,64

Tabla 4.5 Tabulación de datos relativos a la distancia “d”

PRUEBAS	DISTANCIA DESEADA (mm)	DISTANCIA MEDIDA (mm)	ERROR ABSOLUTO	ERROR RELATIVO (%)
1	10,0	10,1	0,1	0,99
2	10,0	10,2	0,2	1,96
3	10,0	10,1	0,1	0,99
4	10,0	10,1	0,1	0,99
5	10,0	10,2	0,2	1,96
6	10,0	10,3	0,3	2,91
7	10,0	9,9	0,1	1,01

Tabla 4.6 Tabulación de datos relativos a la distancia “e”

PRUEBAS	DISTANCIA DESEADA (mm)	DISTANCIA MEDIDA (mm)	ERROR ABSOLUTO	ERROR RELATIVO (%)
1	1,2	1,1	0,1	9,09
2	1,2	1,2	0	0,00
3	1,2	1,1	0,1	9,09
4	1,2	1,1	0,1	9,09
5	1,2	1,1	0,1	9,09
6	1,2	1,2	0	0,00
7	1,2	1,1	0,1	9,09

Tabla 4.7 Tabulación de datos relativos a la distancia “f”

PRUEBAS	DISTANCIA DESEADA (mm)	DISTANCIA MEDIDA (mm)	ERROR ABSOLUTO	ERROR RELATIVO (%)
1	0,5	0,5	0,0	0,00
2	0,5	0,5	0,0	0,00
3	0,5	0,5	0,0	0,00
4	0,5	0,5	0,0	0,00
5	0,5	0,5	0,0	0,00
6	0,5	0,5	0,0	0,00
7	0,5	0,5	0,0	0,00

De los datos anteriormente tabulados se observa que el máximo error absoluto en cuanto a las medidas realizadas de las distancias en los caracteres impresos no supera el valor de 0.5, además el máximo valor de error relativo es de 7.69 %, estos valores muestran la eficacia del prototipo realizado en cuanto a dimensiones de los caracteres impresos.

4.3.2.5 Cálculo del tiempo de impresión para un caracter Braille

Se modifica la programación del prototipo para realizar filas completas de caracteres que se consideran llenos, es decir que se imprimen o realizan el relieve de todos los 6 puntos de la matriz de 3 x 2 que conforman una celda Braille, a continuación se toma el tiempo en que se imprimen cada una de las filas compuestas por 30 caracteres, además se realiza la impresión de 3 filas completas promediando el tiempo en que toma imprimir un caracter lleno,

determinándose al final la velocidad aproximada de impresión del prototipo, así en la Tabla 4.8 se tabulan los tiempos requeridos.

Tabla 4.8 Tabulación de datos relativos la velocidad de impresión de cada caracter del prototipo “Impresora Braille”

PRUEBAS	CARACTERES LLENOS IMPRESOS	TIEMPO TOTAL (s)	TIEMPO DE IMPRESION DE CADA CARACTER (s)
1	30	222	7,4
2	30	231	7,7
3	30	237	7,9
VELOCIDAD PROMEDIO:			7,6

De los valores de la Tabla 4.8 se observa que la velocidad promedio aproximada del prototipo es de 7.6 segundos por cada caracter, lo que implica que el prototipo no es tan veloz como sus contrapartes comerciales, pero definitivamente es una alternativa viable para implementarlo comercialmente.

4.4 COSTO DEL PROYECTO

En la Tabla 4.9 que se muestra a continuación, se presenta los costos de materiales mismos que se invirtieron en la realización del presente proyecto, para éste estudio se toma en cuenta entre otras cosas los materiales empleados tanto electrónicos como mecánicos y de construcción, de igual manera se incluye un análisis de ingeniería en el que se destaca la participación efectiva en horas de investigación y desarrollo que se han invertido en el prototipo.

Tabla 4.9 Análisis de costos del proyecto

DESCRIPCIÓN	CANTIDAD	V. UNITARIO	V. TOTAL
<i>ELEMENTOS ELECTRÓNICOS</i>			
Microcontrolador 18F4550	1	13,00	13,00
Condensador cerámico 15pF	2	0,08	0,16
Condensador cerámico 10nF	2	0,08	0,16
Condensador cerámico 22nF	2	0,08	0,16
Condensador cerámico 100nF	24	0,10	2,40

Condensador cerámico 470nF	4	0,10	0,40
Condensador electrolítico 4,7uF	1	0,10	0,10
Condensador electrolítico 10uF	4	0,10	0,40
Condensador electrolítico 22uF	1	0,10	0,10
Condensador electrolítico 47uF	1	0,10	0,10
Condensador electrolítico 100uF	1	0,28	0,28
Condensador electrolítico 470uF	3	0,49	1,47
Condensador electrolítico 1000uF	1	0,60	0,60
Condensador electrolítico 4700uF	2	0,60	1,20
Cristal de 20MHz	1	0,60	0,60
Resistencia 10 Ω $\frac{1}{4}$ W	1	0,02	0,02
Resistencia 39 Ω $\frac{1}{2}$ W	1	0,03	0,03
Resistencia 220 Ω $\frac{1}{2}$ W	4	0,03	0,12
Resistencia 2,2 Ω $\frac{1}{2}$ W	1	0,03	0,03
Resistencia 1 Ω $\frac{1}{2}$ W	1	0,03	0,03
Resistencia 330 Ω $\frac{1}{4}$ W	4	0,02	0,08
Resistencia 470 Ω $\frac{1}{4}$ W	1	0,02	0,02
Resistencia 560 Ω $\frac{1}{4}$ W	1	0,02	0,02
Resistencia 1 K Ω $\frac{1}{4}$ W	1	0,02	0,02
Resistencia 1,2 K Ω $\frac{1}{4}$ W	1	0,02	0,02
Resistencia 2,2 K Ω $\frac{1}{4}$ W	2	0,02	0,04
Resistencia 4,7 K Ω $\frac{1}{4}$ W	6	0,02	0,12
Resistencia 10 K Ω $\frac{1}{4}$ W	4	0,02	0,08
Trimmer 10 K Ω	2	0,78	1,56
Potenciómetro 500 K Ω	1	0,25	0,25
Potenciómetro araña 10 K Ω	1	0,17	0,17
Diodo 1N4148	3	0,08	0,24
Diodo 1N4007	9	0,05	0,45
Diodo FR107	8	0,12	0,96
Led 3mm	6	0,07	0,42
Jack USB "B"	1	0,68	0,68
Cable USB "A" a "B"	1	2,45	2,45
Conector tipo espadín hembra	2	0,98	1,96
Conector tipo espadín macho	1	0,60	0,60
Conector maquinado	4	1,10	4,40
Jack RJ45	1	0,66	0,66
Bornera 3 tornillos	3	0,30	0,90
Bornera 2 tornillos	7	0,25	1,75
Conector tipo BUS plano 14 pin	4	2,45	9,80

Cable tipo bus plano	2	1,30	2,60
Cable de red flexible	1	1,45	1,45
Conector Molex 2 pines	6	0,35	2,10
Conector Molex 3 pines	2	0,35	0,70
Conector Molex 5 pines	3	0,60	1,80
Módulo mp3 wtm	1	60,00	60,00
Transistor 2N3904	3	0,08	0,24
Amplificador operacional LM358	1	0,96	0,96
Puente "h" integrado L293	2	3,35	6,70
Driver motor a pasos ULN2004	1	0,67	0,67
LCD 2x16	1	7,50	7,50
Amplificador de audio TDA2003	1	0,69	0,69
Pulsador 4 pines	1	0,22	0,22
Regulador 7805	2	0,42	0,84
Regulador 7812	3	0,52	1,56
Regulador LM317	1	0,54	0,54
Fin de carrera 3 pines	1	0,22	0,22
Sensor óptico GP2S40	1	4,56	4,56
Sensor óptico CNY70	1	4,23	4,23
Puente rectificador 2W005	2	0,47	0,94
Interruptor retroreflectivo	1	2,45	2,45
Jack de fusible	2	0,28	0,56
Fusible	2	0,10	0,20
Ventilador 12 V	1	2,90	2,90
Conector de 4 vias tipo automovil	1	0,45	0,45
Transformador 12V toma central	1	5,35	5,35
Jack de poder 3 vias	1	0,41	0,41
Servomotor Hitech	1	15,09	15,09
Cable de poder	1	3,20	3,20
PCB simple cara fibra 200 cm2	1	34,00	34,00
PCB doble cara fibra 75 cm2	1	45,00	45,00
PCB simple cara fibra 6 cm2	1	5,00	5,00
PCB doble cara fibra 30 cm2	1	45,00	45,00
PCB doble cara fibra 12 cm2	1	35,00	35,00
Subtotal (USD)			342,14
<i>COMPONENTES MECÁNICOS Y CONSTRUCTIVOS</i>			
Caja de proyectos PBC negro	1	6,08	6,08
Impresora HP descartada	1	15,00	15,00
Perfileria variada de aluminio	1	15,00	15,00

Tornillos y tuercas variados	1	6,00	6,00
Alquiler de taladro y menajes varios	1	20,00	20,00
Subtotal (USD)			62,08
<i>COSTO DE INGENIERÍA</i>			
Horas efectivas de trabajo	100	15,00	1500,00
Subtotal (USD)			1500,00
TOTAL (USD)			1904,22

En el siguiente capítulo se detallan las diversas conclusiones y recomendaciones desarrolladas a partir de la culminación del presente proyecto.

CAPITULO 5

CONCLUSIONES Y RECOMENDACIONES

5.1 CONCLUSIONES

- Se puede construir un prototipo para plasmar caracteres Braille en una hoja de papel a partir de una impresora de inyección a tinta modificando la parte mecánica y diseñando la parte electrónica en hardware y software. Para el software se puede utilizar cualquier tipo de programa que permita almacenar instrucciones en un microcontrolador procedente de cualquier casa comercial y para el Hardware se pueden diseñar y construir placas modulares que operen conjuntamente, de esta forma se tiene facilidad de reemplazo en caso de presentarse fallas, futuras implementaciones y mejoras.
- Para proyectos en los que se desarrolle comunicación USB se puede utilizar un microcontrolador como el 16f741, 18f2550 pero la aparente ventaja del integrado 18f4550, que es el que se emplea en el desarrollo de éste proyecto, es que supera la capacidad de memoria de sus antecesores así como el número de periféricos que se pueden manejar sin necesidad de utilizar modo extendido; lo que posibilita que los tiempos de ejecución de las sentencias para el control sean menores.
- Para un funcionamiento adecuado es necesario que previo a realizar las sentencias de comunicación USB se ejecuten otras, que dejen el sistema mecánico listo para una tarea de impresión, de manera que de producirse un error éste sea corregido o no persista al reinicializar el sistema. Para éste proyecto en particular estas sentencias se enfocan a verificar la presencia de papel en el rodillo ya que de existir éste sin tener las sentencias de diagnóstico implementadas se provoca un recalentamiento

en el driver que comanda al motor encargado del movimiento del rodillo, pudiendo ocasionar daños mayores en el sistema.

- No existe un consenso absoluto que se pueda seguir para realizar una impresión en caracteres Braille esto se debe básicamente a que se han creado varios conversores que traducen en su mayoría textualmente lo que se encuentra en un documento, sin embargo hay que tener en cuenta que se tiene una limitación en cuanto al número de caracteres existentes para la codificación Braille y que en reiteradas ocasiones un caracter tiene varios significados que dependen de otros caracteres adjuntos que le dan un significado propio de acuerdo al contexto del documento. En vista de que no todos los caracteres tienen un mismo significado en todos los idiomas. Cada organización encargada de regir las normativas referentes a la escritura en Braille trata de conseguir un consenso general con las demás, éste proceso tarda muchos años y aún no se llegan a acuerdos definitivos por lo tanto el presente proyecto ha tomado en cuenta las normas que utiliza la ONCE (Organización Nacional de Ciegos Españoles) ya que las mismas están siendo utilizadas en la mayoría de países de habla hispana.
- El módulo de reproducción de voz WTM – SD V1.3 posee ciertas ventajas sobre otros módulos de las mismas características, por un lado el control de éste dispositivo se lo puede realizar utilizando varios tipos de comunicación como RS 232, paralelo o por teclado lo que permite una elección adecuada de acuerdo a la aplicación en la que esté involucrado éste dispositivo, para éste proyecto se realiza el control en paralelo ya que su velocidad de ejecución y control es elevada respecto a RS 232, otro punto importante es la posibilidad de poder grabar cualquier tipo de mensaje siempre y cuando tenga un formato MP3, con lo que la posibilidad de emplear mensajes de ayuda para la utilización de dispositivos por personas con discapacidad visual es relativamente ilimitado.

- Debido a que en un proyecto se tiene cambios imprevistos como por ejemplo en los niveles de voltaje requeridos para la alimentación de dispositivos, es importante que a la hora de realizar la elección de la fuente se tome en cuenta que se deben tener valores diferentes de voltaje regulables, por los requerimientos de éste proyecto es necesario la utilización de una fuente ATX por su gran versatilidad en cuanto a niveles de voltaje fijos que provee y además por la elevada cantidad de corriente que se le puede drenar a éste tipo de dispositivos, solamente hay que tener en cuenta la correcta distribución de voltajes en sus conectores para poder aprovechar al máximo sus características.
- Microchip proporciona los identificadores necesarios para que el dispositivo desarrollado en comunicación USB sea correctamente identificado por el computador, esto debido en gran parte a que Microsoft mantiene continuamente actualizados los drivers que identifican a un dispositivo USB de esta casa fabricante, esto se evidencia al momento de conectarse a la PC, esta lo identifica como un dispositivo que está utilizando elementos de esta casa, hay que tener en cuenta que los identificadores que microchip facilita no son de uso comercial si cabe el termino, pues se emplean únicamente para fines de desarrollo e investigación, si se desea desarrollar un dispositivo de manera comercial es necesario adquirir los mismos realizando un importante aporte económico a la organización encargada de proveer los identificadores para comunicación de la PC con dispositivos externos a través del puerto USB.
- El lenguaje o entorno de programación para microcontroladores PIC C es adecuado para el desarrollo de prototipos como éste ya que provee de librerías como usb.h, cdc.h, dcd.c, que facilitan la programación y el manejo de periféricos gracias a la comunicación USB siendo éste el principal propósito del presente proyecto.
- En cuanto a la implementación de un dispositivo que realice el relieve de un caracter Braille la mejor opción es decantarse por elementos que empleen

un principio de percusión sea cual fuere el actuador que se elija. Los tiempos del proceso son reducidos en comparación con otros métodos, la calidad de la impresión es mucho mayor sin embargo la mayoría de dispositivos que se emplean como son los solenoides poseen un peso considerable, incrementando de esta forma el rozamiento y desgastando los elementos mecánicos involucrados en su desplazamiento, pese a esto es uno de los mejores elementos a emplear en éste tipo de proyectos.

5.2 RECOMENDACIONES

- Se recomienda que al implementar un placa electrónica o circuito impreso donde se aloja un microcontrolador se considere la construcción de un circuito de grabación ICSP (In Circuit Serial Programming) para facilitar las modificaciones futuras en la programación ya que el procedimiento de extraer el microcontrolador en repetidas ocasiones de la placa en donde se aloja puede hacer que éste se deteriore o sufra daños físicos que repercutan en el funcionamiento adecuado del mismo implicando gastos adicionales al tener que reemplazarlo por deterioro.
- Es recomendable que el voltaje de la fuente de alimentación al dispositivo sea verificado previamente a ser enchufado, para ello se confirma el estado de los terminales que alimentan al prototipo, pues de esta manera se evitara posibles daños en los elementos.
- Es recomendable que los soportes para el punzón y la bandeja de impresión se realicen de algún material resistente ya que de hacerlo en uno flexible puede romperse o no cumplir con el objetivo para el que fue diseñado, proveer de soporte a la vez de ser molde y realizar un relieve con especificaciones Braille, en el presente proyecto se emplean aluminio, fomix y acetato, la combinación de ellos permite cumplir con el objetivo planteado.

- Para la programación en la que se requiere establecer algún tipo de comunicación con la PC se recomienda emplear un lenguaje de alto nivel y sobre todo realizar pruebas preliminares de funcionamiento para observar el comportamiento, principales características, virtudes y defectos del entorno de programación, en el presente proyecto se emplea y recomienda ampliamente PIC C como lenguaje de programación de microcontroladores pues el mismo provee de librerías que al final permiten que el desarrollo del proyecto sea más amigables, por otro lado en cuanto a la programación de una interfaz de comunicación entre la PC y medios externos se emplea LabVIEW por su versatilidad y alta confiabilidad al momento de desarrollar programas que involucren comunicación con medios externos sea por medio de interfaces de comunicación propias del computador o mediante una de sus tarjetas de adquisición de datos.
- Se recomienda que las placas electrónicas se realicen en fibra de vidrio en lugar de hacerlas en baquelita ya que la fibra de vidrio no absorbe humedad que podría generarse por el sitio en donde se encuentre en operación el módulo, otra de sus características es la flexibilidad lo que ayuda a que el nivel de resistencia a las tensiones mecánicas sea mayor, provee además durabilidad a las pistas de cobre por las características propias de la fibra de vidrio que fijan de mejor manera éste material en la placa.
- Se recomienda engrasar el bocín y el eje en el cual está acoplado ya que al no existir lubricación en el mismo se pueden producir atascamientos del carrete provocando sobrecalentamiento en el driver que comanda al motor de DC, generando así una sobre exigencia de corriente lo que podría abocar en un reseteo del sistema.
- La excesiva sensibilidad a la interferencia estática del módulo reproductor de voz lo hacen un elemento delicado dentro del prototipo por lo que se recomienda que la manipulación del mismo sea muy cuidadosa y por sobre

todas las cosas se tenga una pulsera antiestática para evitar que al manipular éste elemento se produzca algún daño irreparable.

- Para la detección del papel se recomienda un sensor óptico basado en la emisión de señal de acuerdo al contraste de colores ya que otro tipo de detección que pudiese ser mecánica como es el caso de los finales de carrera puede generar falsos positivos y ocasionar la operación inadecuada del módulo en general.
- Para éste tipo de aplicaciones se puede utilizar la clase de comunicación Bulk o CDC. La comunicación de transferencia masiva Bulk permite transmitir paquetes de información lo que no ocurre con la comunicación tipo CDC en la que cada caracter se almacena en un buffer y permanece ahí hasta que sea enviado o recibido según sea el caso, pero considerando los tiempos de comunicación, la velocidad de transferencia, y el tiempo que transcurre hasta que el módulo realice el proceso se implementa la comunicación USB CDC.
- El entorno de programación visual de LabVIEW permite que el desarrollo del proyecto sea amigable y que el despliegue de la información en la pantalla sea didáctica, esto permite que se realicen modificaciones de manera rápida y sin afectar el resto del programa desarrollado. Al ser un lenguaje visual la programación se vuelve sencilla e intuitiva en muchos casos.

REFERENCIAS BIBLIOGRÁFICAS

- [1] ONCE, “*El Alfabeto Braille*,” 2010, Disponible: <http://www.discapnet.es/Castellano/areastematicas/tecnologia/productos%20de%20apoyo/Guias%20tecnicas/Paginas/alfabetobrilie.aspx>
- [2] Cabrerizo & Eutiquio, “*El Braille, mucho más que un sistema de lectura para los ciegos*,” U.T.L.A.I., 2002, Disponible: <http://www.Nodo50.org/utlai/lucer11.htm>
- [3] ONCE (Organización Nacional de Ciegos Españoles), “*Características de la rotulación para personas con discapacidad visual, Edición Multilingüe*,” Primera edición, España 2007.
- [4] Enable Mart, “*Braille embossers*,” 2010, Disponible: <http://www.enablemart.com/Braille-Embossers>
- [5] Martin Daniel, “*Sensores Aplicados a la Robótica*,” Robótica & μ Controladores Pic, 2009, Disponible: <http://www.x-robotics.com/>
- [6] DYNAMO, “*Sensores y Motores*,” 2010, Disponible: <http://www.dynamoelectronics.com/home.html>
- [7] Minebea-Matsushita Motor Corporation, “*PM55L-048*,” 2010, Disponible: http://www.eminebea.com/content/html/en/motor_list/pm_motor/
- [8] Pelletier Alain, “*Motor C2162-6006 HP*,” 2010, Disponible: <http://www.motors.wrobots.com/C2162-6006.php>
- [9] Radek Chris, “*HP Encoder Modules*,” 2010, Disponible: <http://www.timeguy.com/cradek/01148758851>
- [10] Even Print, “*Print Mechanism Assembly HP*,” 2007, Disponible: <http://www.evenprint.com/diagrams-printers/09999923htrHP95C.html>
- [11] VISHAY semiconductors, “*Audio Application TDA 2003*,” 2009, Disponible: <http://www.datasheetcatalog.com/>
- [12] MICROCHIP, “*18FXXXX Family*,” 2011, Disponible: <http://www.microchip.com/>
- [13] BIBLIOMAN, “*Prácticas con microcontroladores PIC en lenguaje C*,” 2010, Disponible: <http://aquihayapuntes.com/indice-practicas-pic-en-c/comunicacion-usb-pic18f4550-utilizando-la-clase-cdc.html?start=1>

- [14] VISHAY semiconductors, "*Optoelectronics CNY70*," 2008, Disponible: <http://www.datasheetcatalog.com/>
- [15] SGS Thomson Microelectronics, "*Push Pull Four Channel Driver L293D*," 1996, Disponible: <http://www.datasheetcatalog.com/>